IBM Tivoli Workload Scheduler

Administration

Version 9.2

SC11-6396-05

IBM Tivoli Workload Scheduler

Administration

Version 9.2

SC11-6396-05

Important
Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques», à la page 465.

Réf. US : SC23-9113-07

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- http://www.fr.ibm.com (serveur IBM en France)
- http://www.can.ibm.com (serveur IBM au Canada)
- http://www.ibm.com (serveur IBM aux Etats-Unis)

Compagnie IBM France
Direction Qualité
17, avenue de l'Europe
92275 Bois-Colombes Cedex

Cette édition s'applique à la version 9.2.0 de Tivoli Workload Scheduler (numéro de programme 5698-WSH) et à toutes les éditions et modifications ultérieures, sauf mention contraire dans les nouvelles éditions.

Table des matières

Chapitre 4. Configuration de l'autorisation des utilisateurs (fichier de sécurité) 155

Chapitre 5. Configuration de l'authentification 199

Chapitre 6. Administration du réseau 227

Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.

OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
(Pos1)		Home
Fin	Fin	End
(PgAr)		PgUp
(PgAv)		PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
(Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

Figures

Tableaux

A propos de cette publication

Le présent manuel *IBM® Tivoli Workload Scheduler - Administration* fournit des informations relatives à l'administration des principaux composants d'IBM Tivoli Workload Scheduler (souvent appelé *moteur*).

Nouveautés de cette édition

Découvrez les nouveautés de la présente édition.

Pour plus d'informations à propos des fonctions nouvelles ou modifiées, voir la section *Récapitulatif des améliorations* du manuel *Tivoli Workload Automation - Présentation*.

Pour plus d'informations sur les APAR résolus par la présente édition, voir les Notes sur l'édition de Tivoli Workload Scheduler à l'adresse http://www-01.ibm.com/support/docview.wss?rs=672&uid=swg27041032 et les Notes sur l'édition de Dynamic Workload Console à l'adresse http://www-01.ibm.com/support/docview.wss?rs=672&uid=swg27041033.

Nouveautés de cette publication

Informez-vous sur les nouveautés de cette publication.

Les sections suivantes ont été ajoutées ou modifiées depuis version 8.5.1 :

- «Règles d'utilisation d'un Registre d'utilisateurs fédéré avec Tivoli Workload Scheduler», à la page 201
- «Configuration du serveur Dynamic Workload Broker sur le gestionnaire de domaine maître et le gestionnaire de domaine dynamique», à la page 67
- «Configuration pour planifier des types de travail avec options avancées», à la page 86
- «Configuration de l'accès à Dynamic Workload Console», à la page 104
- «Utilitaires d'audit», à la page 350
- «Modification d'un gestionnaire de domaine ou d'un gestionnaire de domaine dynamique», à la page 373.
- «Modification du nom d'hôte ou de l'adresse IP du poste de travail», à la page 400

Pour plus d'informations sur les fonctions nouvelles ou modifiées de cette version, voir *Tivoli Workload Automation - Présentation*.

Le texte modifié ou ajouté par rapport à la version précédente est marqué d'une barre verticale dans la marge de gauche.

Public visé

Informez-vous sur les utilisateurs concernés par cette publication.

Cette publication fournit des informations relatives à l'administration du produit au quotidien. Elle vise à aider l'administrateur informatique ou l'administrateur de Tivoli Workload Scheduler chargé de faire en sorte que le produit fonctionne de

manière fluide et correctement. Ils trouveront des informations sur la façon de procéder à des modifications courantes de la configuration, par exemple pour ajouter un utilisateur et des informations sur les procédures périodiques qui garantissent l'intégrité du produit, telles que les sauvegardes.

Le lecteur doit être un programmeur système expert ayant une bonne compréhension de l'infrastructure Tivoli Workload Scheduler et des interactions entre les composants.

Publications

Le produit Tivoli Workload Automation fait l'objet de plusieurs publications.

Pour connaître la liste des publications disponibles dans la bibliothèque du logiciel Tivoli Workload Automation, voir *Publications* sous *Référence* dans la documentation du produit.

Pour connaître la liste des termes utilisés dans le produit Tivoli Workload Automation, voir *Glossaire* sous *Référence* dans la documentation du produit.

Accessibilité

Les fonctions d'accessibilité permettent aux personnes souffrant d'un handicap physique (par exemple, une mobilité réduite ou une déficience visuelle) de pouvoir utiliser les logiciels.

Avec ce produit, vous pouvez utiliser les technologies d'assistance pour parcourir l'interface à l'aide de messages sonores. Vous pouvez également utiliser le clavier au lieu de la souris pour toutes les fonctions de l'interface graphique.

Pour obtenir des informations complètes sur Dynamic Workload Console, reportez-vous à l'annexe Accessibilité du manuel *IBM Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Formation technique Tivoli

Tivoli propose une formation technique.

Pour plus d'informations sur la formation technique Tivoli, reportez-vous au site Web IBM Tivoli Education suivant :

http://www.ibm.com/software/tivoli/education

Informations sur le support

IBM vous propose plusieurs façons d'obtenir de l'aide lorsque vous êtes confronté à un problème.

Si vous rencontrez un problème avec un logiciel IBM, vous pouvez le résoudre rapidement. IBM vous permet d'obtenir l'assistance que vous souhaitez de plusieurs manières :

- En faisant des recherches dans les bases de connaissances : elles contiennent un grand nombre de problèmes recensés et de solutions, de remarques d'ordre technique et autres informations adéquates.

- En vous procurant des correctifs : vous trouverez les versions les plus récentes disponibles pour votre produit.
- En contactant le service de support logiciel IBM : si les deux solutions ci-dessus ne vous ont pas permis de résoudre votre problème, vous pouvez contacter directement un technicien IBM de plusieurs manières.

Pour plus d'informations sur ces trois manières de résoudre un incident, voir l'annexe relative aux informations de support dans le manuel *Tivoli Workload Scheduler - Guide d'identification et de résolution des problèmes*.

Chapitre 1. Initiation à l'administration

Cette publication explique comment réaliser des tâches d'administration sur Tivoli Workload Scheduler et Dynamic Workload Console. La plupart des procédures qu'elle décrit impliquent d'identifier un fichier dans le chemin d'installation du produit et de son composant. Ces fichiers peuvent toutefois figurer dans des chemins d'installation différents en fonction des différents composants ou sur des systèmes d'exploitation différents, comme indiqué dans la section «Emplacement de l'installation des produits et des composants».

Emplacement de l'installation des produits et des composants

Cette section commence par une brève introduction sur Tivoli Workload Automation et inclut des détails sur les chemins d'installation et les composants de Tivoli Workload Scheduler.

Tivoli Workload Automation

Tivoli Workload Automation est le nom d'une famille de produits et de composants comprenant :

- Tivoli Workload Scheduler
- Tivoli Workload Scheduler pour z/OS
- Tivoli Workload Scheduler for Applications
- Dynamic Workload Console
- Tivoli Workload Scheduler for Virtualized Data Centres
- Tivoli Workload Scheduler Plug-in for Informatica PowerCenter

Un grand nombre de composants Tivoli Workload Scheduler sont installés dans une *instance Tivoli Workload Automation*.

Chemins d'installation

Cette section décrit les chemins d'installation des composants Tivoli Workload Scheduler :

Chemin d'installation du répertoire principal de TWA

Selon la description ci-dessus, un grand nombre de composants sont installés dans une instance Tivoli Workload Automation. Bien qu'il s'agisse d'une structure notionnelle, elle est représentée sur l'ordinateur sur lequel vous installez les composants de Tivoli Workload Automation par un répertoire commun désigné dans la documentation par *TWA_home*. Le chemin de ce répertoire est déterminé lorsque vous installez un composant Tivoli Workload Scheduler pour la première fois. Vous pouvez choisir ce chemin lorsque vous effectuez l'installation pour la première fois, mais si vous acceptez le chemin par défaut, il se présente comme suit :

Linux `/opt/IBM/TWA<n>`

UNIX `/opt/ibm/TWA<n>`

Windows

`C:\Program Files\IBM\TWA<n>`

où <n> est un entier situé entre <null> pour la première instance installée, 1 pour la deuxième, et ainsi de suite.

Dans les publications, ce chemin est nommé *TWA_home*. Pour obtenir des détails à propos des répertoires créés en dehors de *TWA_home*, voir *Tivoli Workload Scheduler : Planification et installation*.

Chemin d'installation de Tivoli Workload Scheduler

Vous pouvez installer plusieurs composants Tivoli Workload Scheduler (gestionnaire de domaine maître, gestionnaire de domaine maître de sauvegarde, gestionnaire de domaine ou gestionnaire de domaine de secours) sur un système, mais chaque composant est installé sur une instance distincte de Tivoli Workload Automation, comme indiqué ci-dessus.

Le chemin d'installation de Tivoli Workload Scheduler est le suivant :

TWA_home/TWS

Chemin d'installation de l'agent Tivoli Workload Scheduler

L'agent utilise également la même structure de chemin par défaut, mais il a son propre répertoire d'installation distinct :

TWA_home/TWS/ITA/cpa

Remarque : L'agent installe aussi certains fichiers hors de ce chemin. Si vous devez partager, mapper ou copier les fichiers de l'agent (par exemple lors de la configuration de la prise en charge des clusters), partagez, mappez ou copiez également ces fichiers :

Systèmes d'exploitation UNIX et Linux

```
/etc/teb/teb_tws_cpa_agent_<TWS_user>.ini
/opt/IBM/CAP/EMICPA_default.xml
/etc/init.d/tebctl-tws_cpa_agent_<TWS_user>
   (sur Linux et Solaris)
/etc/rc.d/init.d/tebctl-tws_cpa_agent_<TWS_user>
   (sur AIX)
/sbin/init.d/tebctl-tws_cpa_agent_<TWS_user>
   (sur HP-UX)
```

Systèmes d'exploitation Windows

```
%windir%\teb\teb_tws_cpa_agent_&lt;tws_user>.ini
%ALLUSERSPROFILE%\Application Data\ibm\CAP\EMICPA_default.xml
```

L'agent utilise les fichiers de configuration suivants qu'il vous faudra peut-être modifier :

JobManager.ini

Ce fichier contient les paramètres indiquant à l'agent comment exécuter les travaux. Vous ne devriez modifier les paramètres que si la documentation Tivoli Workload Scheduler le recommande ou si le service de support logiciel IBM vous invite à le faire. Le chemin de ce fichier est :

TWA_home/TWS/ITA/cpa/config/JobManager.ini

JobManagerGW.ini

Lorsqu'un agent dynamique est installé et **-gateway** `local|remote` est spécifié, ce fichier contient alors les mêmes paramètres que le fichier `JobManager.ini`, à l'exception des différences suivantes :

- Le paramètre **ResourceAdvisorUrl** pointe vers le the Dynamic Workload Broker et non sur le gestionnaire de domaine maître.

Le fichier `JobManagerGW.ini` est installé à l'emplacement suivant :

| *TWA_home*/TWS/ITA/cpa/config/JobManagerGW.ini

ita.ini Ce fichier contient les paramètres qui déterminent le comportement de l'agent. Modifier ces paramètres peut compromettre le bon fonctionnement de l'agent et nécessite sa réinstallation. Vous ne devriez modifier les paramètres que si la documentation Tivoli Workload Scheduler le recommande ou si le service de support logiciel IBM vous invite à le faire. Le chemin de ce fichier est :

TWA_home/TWS/ITA/cpa/ita/ita.ini

Chemin d'installation des fichiers fournissant la fonction de planification dynamique

Les fichiers fournissant la fonction de planification dynamique sont installés dans le chemin suivant :

TWA_home/TDWB

Chemin d'installation de Dynamic Workload Console

Vous pouvez installer Dynamic Workload Console dans le chemin de votre choix, mais le chemin d'installation par défaut est le suivant :

Sous Windows

C:\Program Files\IBM\TWAUI

Sous UNIX

/opt/IBM/TWAUI

Chemin d'installation de WebSphere Application Server

WebSphere Application Server est automatiquement installé lorsque vous créez une nouvelle instance *Tivoli Workload Automation*. Vous pouvez indiquer n'importe quel chemin pour l'installation. Le chemin d'installation par défaut est :

TWA_home/WAS

Pour Dynamic Workload Console : C:\Program Files\IBM\JazzSM

Chemin d'installation de client de ligne de commande

Le client de ligne de commande est installé à l'extérieur de toutes les instances de *Tivoli Workload Automation*. Son chemin par défaut est le suivant :

UNIX /opt/ibm/TWS/CLI

Windows

C:\Program Files\IBM\TWS\CLI

Chemin d'installation des outils du serveur d'applications

Because the WebSphere Application Server n'étant pas fourni avec une interface graphique d'administration, de nombreuses tâches d'administration s'effectuent via des outils fournis avec Tivoli Workload Scheduler, qui apportent les modifications nécessaires à la configuration. Ces outils portent le nom de *wastools*. Ils sont installés à l'emplacement suivant :

TWA_home/wastools

Toutefois, les informations ci-dessus ne contiennent que les chemins *par défaut*. Pour déterminer les chemins réels des produits et composants installés dans les instances Tivoli Workload Automation, voir «Détermination des éléments installés dans les instances Tivoli Workload Automation», à la page 4

Détermination des éléments installés dans les instances Tivoli Workload Automation

Si vous n'avez pas la charge d'installer Tivoli Workload Scheduler et ses composants, vous pouvez ne pas connaître les composants installés et leurs instances Tivoli Workload Automation d'installation. Suivez la procédure suivante pour déterminer cela :

1. Accédez au répertoire suivant :

   Systèmes d'exploitation UNIX et Linux
   `/etc/TWA`

   Systèmes d'exploitation Windows
   `%windir%\TWA`

2. Répertoriez le contenu du répertoire. Chaque instance de Tivoli Workload Automation est représentée par un fichier appelé : `twainstance<numéro_instance>.TWA.properties`. Ces fichiers sont supprimés lorsque tous les produits ou composants d'une instance sont désinstallés. Le nombre de fichiers présents indique donc le nombre d'instances valides en cours d'utilisation.

3. Ouvrez un fichier dans un visualiseur de texte.

   Avertissement : Ne modifiez pas le contenu de ce fichier, sauf si vous êtes invité à le faire par le service de support logiciel IBM. Si vous modifiez ce ficher, votre environnement Tivoli Workload Scheduler peut devenir invalide.

   Le contenu est identique à ce qui suit :

```
TWS_version=9.2.0.0
DB2_basePath=/home/db2inst1/sqllib
DB2_IS_SERVER=TRUE
EWas_basePath=/opt/IBM/WebSphere/AppServer
DB2_INSTANCE_PORT=50000
TWS_counter=1
EWas_counter=1
TWA_path=/opt/tws/tws
TWS_server_name=bvtserver
DB2_ADMINISTRATOR_NAME=db2inst1
TWS_instance_type=MDM
EWas_profile_path=/opt/tws/tws/Appserver/profiles/TWSProfile
EWas_node_name=TWSNode
TWS_basePath=/opt/tws/tws/TWS
EWas_user=tws
EWas_cell_name=TWSCell
EWas_version=8.5.0.1
DB2_version=10.1.0.0
EWas_server_name=server1
EWas_update_installer_dir=
TWS_LAST_COMMITED_LEVEL_KEY=9.2.0.00
TWS_user_name=tws
TWS_FIX_LIST_KEY=
DB2_INSTANCE_NAME=db2inst1
DB2_counter=1
TWA_componentList=TWS,EWas,DB2
EWas_isc_version_key=8.5.0.1
EWas_profile_name=BVTProfile
EWas_service_name=IBMWAS85Service - tws
```

   Il est important d'interpréter les éléments suivants dans ce fichier :

   TWA_path
   Chemin de base, où l'installation a ajouté un ou plusieurs répertoires parmi les suivants, selon les éléments installés :

TWS Emplacement d'installation du composant Tivoli Workload Scheduler

TWAUI Emplacement d'installation de Dynamic Workload Console

WAS Emplacement d'installation de WebSphere Application Server

wastools
Emplacement d'installation des outils que vous utilisez pour configurer la WebSphere Application Server

ssm Emplacement d'installation de l'agent de surveillance SSM Netcool (utilisé dans le cadre de la gestion des événements)

TWA_componentList
Répertorie les composants installés dans l'instance de Tivoli Workload Automation

TWS_counter
Indique si un composant Tivoli Workload Scheduler est installé dans cette instance de Tivoli Workload Automation (lorsque value=1)

TWS_instance_type
Indique le composant Tivoli Workload Scheduler installé dans cette instance :

MDM Gestionnaire de domaine maître

BKM gestionnaire de domaine maître de secours

DDM gestionnaire de domaine dynamique

BDDM
gestionnaire de domaine dynamique de secours

FTA Agent tolérant aux pannes ou gestionnaire de domaine

TDWC_counter
Indique si une instance de Dynamic Workload Console est installée dans cette instance de Tivoli Workload Automation (lorsque value=1)

EWas_counter
Indique le nombre d'applications installées dans cette instance de Tivoli Workload Automation et qui accèdent à WebSphere Application Server.

TWS_user_name
ID de l'<*utilisateur_TWS*> du composant Tivoli Workload Scheduler.

EWas_user
ID de l'utilisateur d'administration de la WebSphere Application Server. Pour une installation par défaut, il est identique à celui du <*utilisateur_TWS*>.

Le seul composant de Tivoli Workload Scheduler installé dans une instance Tivoli Workload Automation, mais pas indiqué de manière explicite, est le connecteur. Pour déterminer s'il a été installé, consultez les combinaisons de touches suivantes :

Agent installé sans connecteur

```
TWS_counter=1
EWas_counter=
TWS_instance_type=FTA
TDWC_counter=
TWA_componentList=TWS
```

Agent installé avec connecteur

```
TWS_counter=1
EWas_counter=1
TWS_instance_type=FTA
TDWC_counter=
TWA_componentList=TWS,EWas
```

Agent installé sans connecteur et sans Dynamic Workload Console

```
TWS_counter=1
EWas_counter=1
TWS_instance_type=FTA
TDWC_counter=1
TWA_componentList=TWS,EWas,TDWC
```

Agent installé avec connecteur et sans Dynamic Workload Console

```
TWS_counter=1
EWas_counter=2
TWS_instance_type=FTA
TDWC_counter=1
TWA_componentList=TWS,EWas,TDWC
```

Remarque : La seule différence entre les deux derniers éléments réside au niveau de `EWas_counter`, égal à 2 au lieu de 1.

Chapitre 2. Personnalisation et configuration de Tivoli Workload Scheduler

Après avoir installé le produit, vous pouvez le personnaliser en fonction de vos besoins opérationnels. Vous pouvez aussi modifier les valeurs personnalisées à tout moment. Le présent chapitre décrit les étapes de personnalisation facultatives pour Tivoli Workload Scheduler. Il comprend les sections suivantes :

- «Définition des options globales»
- «Définition des options locales», à la page 30
- «Définition des options d'utilisateur», à la page 51
- «Configuration du serveur Dynamic Workload Broker sur le gestionnaire de domaine maître et le gestionnaire de domaine dynamique», à la page 67
- «Configuration de l'agent», à la page 52
- «Configuration de l'authentification pour l'accès au client de ligne de commande», à la page 91
- «Invites et messages d'écran de Tivoli Workload Scheduler», à la page 94
- «Activation de la fonction de fuseau horaire», à la page 95
- «Configuration de l'utilisation des commandes de rapport», à la page 96

Remarque : Pour plus d'informations sur l'automatisation du cycle de production et la gestion de l'environnement de production, voir *Guide d'utilisation et de référence*.

Définition des options globales

Définissez les options globales à l'aide de la commande **optman**.

optman

Gère les options globales de Tivoli Workload Scheduler. Vous pouvez les répertorier, les afficher et les modifier.

Autorisation

vous devez disposer des droits d'accès suivants pour le fichier d'options globales dans le fichier de sécurité Tivoli Workload Scheduler pour utiliser cette commande :

- Pour `optman ls` ou `optman show` :

  ```
  FILE NAME=GLOBALOPTS ACCESS=DISPLAY
  ```

- Pour `optman chg` :

  ```
  FILE NAME=GLOBALOPTS ACCESS=MODIFY
  ```

Pour plus d'informations sur le fichier de sécurité, voir Chapitre 4, «Configuration de l'autorisation des utilisateurs (fichier de sécurité)», à la page 155.

Syntaxe

optman [-u | -v]
optman [<*connectionParams*>] chg {<*option*> | <*shortName*>} = <*value*>
optman [<*connectionParams*>] ls
optman [<*connectionParams*>] show {<*option*> | <*shortName*>}

Arguments

<connectionParams>

Si vous utilisez la commande **optman** à partir du gestionnaire de domaine maître, les paramètres de connexion ont été configurés à l'installation et n'ont pas besoin d'être indiqués, sauf si vous ne souhaitez pas utiliser les valeurs par défaut.

Si vous utilisez la commande **optman** à partir du client de ligne de commande sur un autre poste de travail, les paramètres de connexion peuvent être obtenus via une ou plusieurs méthodes parmi les suivantes :

- Stockés dans le fichier `localopts`
- Stockés dans le fichier `useropts`
- Fournis à la commande dans un fichier de paramètres
- Fournis à la commande comme partie intégrante de la chaîne de commande

Pour obtenir les détails complets sur les paramètres de connexion, voir «Configuration de l'authentification pour l'accès au client de ligne de commande», à la page 91.

chg {*<option>* | *<shortName>*} **=** *<valeur>*

Remplacez la valeur d'une option par la nouvelle valeur fournie. L'option peut être identifiée par son nom complet ou par son nom abrégé. Voir Options globales - récapitulatif pour obtenir un tableau présentant toutes les options avec leurs noms complets et leurs noms abrégés, les plages de valeurs et les valeurs par défaut. Pour une description complète de chaque option, voir «Options globales - Description détaillée», à la page 14.

ls Affiche la liste des valeurs en cours de toutes les options globales.

show {*<option>* | *<shortName>*}

Affiche la valeur en cours de l'option indiquée. L'option peut être identifiée par son nom complet ou par son nom abrégé. Voir Options globales - récapitulatif pour obtenir un tableau présentant toutes les options avec leurs noms complets et leurs noms abrégés, les plages de valeurs et les valeurs par défaut. Pour une description complète de chaque option, voir «Options globales - Description détaillée», à la page 14.

Commentaires

Certaines modifications sont effectives immédiatement, mais d'autres nécessitent une action spécifique, telle que l'exécution de **JnextPlan** et le redémarrage du WebSphere Application Server. Ces actions sont indiquées dans les descriptions des options. Reportez-vous au document*Tivoli Workload Scheduler - Guide d'utilisation et de référence* pour plus d'informations sur la commande **JnextPlan**.

Exemples

Exemple 1 : afficher la liste des options globales

Pour afficher la liste de toutes les options globales, lorsque vos paramètres de connexion sont fournis via les fichiers `localopts` et `useropts`, entrez la commande suivante :

```
optman ls
```

Exemple 2 : afficher la valeur d'une option globale

Pour afficher la valeur en cours de l'option globale `enCarryForward`, en l'identifiant avec son nom abrégé, entrez la commande suivante :

```
optman show cf
```

Exemple 3 : modifier la valeur d'une option globale

Pour modifier la valeur de l'option globale `enCarryForward`, en l'identifiant avec son nom complet, entrez la commande suivante :

```
optman chg enCarryForward no
```

Options globales - récapitulatif

Cette section récapitule les options globales gérées par **optman**. Les colonnes du tableau signifient ce qui suit :

Description
Brève description de l'option

Nom **Option** telle qu'elle est utilisée dans les commandes **optman**.

Nom abrégé
shortName tel qu'il est utilisé dans les commandes **optman**.

Par défaut
Valeur par défaut appliquée à l'option lors de l'installation (le cas échéant).

Plage Plage ou choix de valeurs que vous pouvez fournir (le cas échéant).

Unités Les unités de la valeur et de la gamme.

Effet Manière de rendre les modifications effectives. Les codes suivants ont été utilisés :

- **E** Si vous activez l'option, lancez le processeur d'événement. Si vous la désactivez, arrêtez le processeur d'événement.
- **Imm** La modification prend immédiatement effet
- **Imm (DB)** La modification prend immédiatement effet dans la base de données uniquement.
- **J** Exécutez **JnextPlan**.
- **J (Plan)** Exécutez **JnextPlan** - cette commande rend le changement effectif dans le plan uniquement.
- **NSJ** La modification prend effet au prochain envoi d'action de flot de travaux.
- **NSM** La modification prend effet au prochain envoi de courrier électronique.
- **W** Redémarrez le serveur WebSphere Application Server.

Le tableau suivant récapitule les options globales de gestion des fonctionnalités et fonctions de Tivoli Workload Scheduler :

Tableau 1. Fonction d'assurance de service de charge de travail

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Activer l'assurance de service de charge de travail	enWorkloadServiceAssurance	wa	yes	yes, no	booléen	J
Approche du décalage tardif	approachingLateOffset	al	120	>=0	secondes	J ou W
Décalage de l'échéance	deadlineOffset	do	2	>=0	minutes	J ou W
Décalage de promotion	promotionOffset	po	120	>=0	secondes	J

Tableau 1. Fonction d'assurance de service de charge de travail (suite)

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Activer le calcul d'heure de début prévue	enForecastStartTime	st	no	yes, no	booléen	imm

Tableau 2. Fonction d'automatisation de charge de travail gérée par événement (EDWA) - généralités

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Activer l'automatisation d'une charge de travail gérée par des événements	enEventDrivenWorkloadAutomation	ed	yes	yes, no	booléen	J ou E
Fréquence de déploiement des règles	deploymentFrequency	df	5	0-60	minutes	Imm
Activer le protocole HTTPS du processeur d'événements	enEventProcessorHttpsProtocol	eh	yes	yes, no	booléen	J
Port de la fonction d'intégration d'événements Tivoli	eventProcessorEIFPort	ee	31131	0 - 65535	numéro du port	W et J
Nom du serveur Sonde EIF (utilisé pour des événements aux formats TEC et TBSM)	TECServerName	th	localhost		nom	J
Port du serveur Sonde EIF (utilisé pour des événements aux formats TEC et TBSM)	TECServerPort	tp	5529	0 – 65535	numéro du port	J

Tableau 3. Fonction d'automatisation de charge de travail gérée par événement (EDWA) - envoi d'événement

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Nom de l'expéditeur du message	mailSenderName	ms	TWS		nom	NSM
Nom du serveur SMTP	smtpServerName	sn	localhost		nom	Imm
Port du serveur SMTP	smtpServerPort	sp	25	0 – 65535	numéro du port	NSM
Module d'extension de messagerie utilise l'authentification SMTP.	smtpUseAuthentication	ua	no	yes, no	booléen	Imm
Nom d'utilisateur SMTP	smtpUserName	un	utilisateur _TWS		nom	Imm
Mot de passe utilisateur SMTP	smtpUserPassword	up				Imm
Module d'extension de messagerie utilise SSL.	smtpUseSSL	us	no	yes, no	booléen	Imm
Module d'extension de messagerie utilise le protocole TLS.	smtpUseTLS	tl	no	yes, no	booléen	Imm

Tableau 4. Fonction d'automatisation de charge de travail gérée par événement - module d'extension Tivoli Workload Scheduler for z/OS

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
nom du serveur distant du connecteur Tivoli Workload Scheduler for z/OS	zOSRemoteServerName	zr			nom	NSJ
nom du serveur du connecteur Tivoli Workload Scheduler for z/OS	zOSServerName	zs	localhost		nom	NSJ
port du serveur de connecteur Tivoli Workload Scheduler for z/OS	zOSServerPort	zp	31217	0 – 65535	numéro du port	NSJ
nom de l'utilisateur du connecteur Tivoli Workload Scheduler for z/OS	zOSUserName	zu	utilisateur _TWS		nom	NSJ
mot de passe de l'utilisateur du connecteur Tivoli Workload Scheduler for z/OS	zOSUserPassword	zw				NSJ

Tableau 5. SSL

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Activer la connexion SSL complète.	enSSLFullConnection	sf	no	yes, no	booléen	J
Activez le chiffrement renforcé des mots de passe.	enStrEncrypt	se	no	yes, no	booléen	J

Tableau 6. Gestion des travaux

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Nombre maximum d'invites après une fin anormale.	baseRecPrompt	bp	1000	0 – 65535	invites	J
Invites supplémentaires après fin anormale.	extRecPrompt	xp	1000	0 – 65535	invites	J
Activer automatiquement la connexion par lot.	enLogonBatch	lb	no	yes, no	booléen	J
Seuil de travail de longue durée.	longDurationThreshold	ld	150	100 - 1000	secondes	J ou W
Utilisateur servant à effectuer la liaison entre le travail reflet et les travaux distants	bindUser	bu	utilisateur _TWS			Imm

Tableau 7. Gestion de flot de travaux

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Flots de travaux sans règle de travail.	enEmptySchedsAreSucc	es	no	yes, no	booléen	J
Empêcher le démarrage des flots de travaux sans dépendance "at"	enPreventStart	ps	yes	yes, no	booléen	J

Tableau 8. Stageman

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Etat des travaux reportés.	carryStates	cs	null		liste des états	J
Activer le report	enCarryForward	cf	all	all, no	booléen	J
Activer le report pour les dépendances interréseaux	enCFinterNetworkDeps	ci	yes	yes, no	booléen	J
Activer le report par quantité de ressources.	enCFResourceQuantity	rq	yes	yes, no	booléen	J
Conserver le nom des travaux de réexécution.	enRetainNameOnRerunFrom	rr	no	yes, no	booléen	J

Tableau 9. Planman

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Durée maximale du plan de préproduction.	maxLen	xl	8	8 - 365	jours	J
Durée minimale du plan de préproduction.	minLen	ml	8	7 - 365	jours	J

Tableau 10. Audit et consignation

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Fréquence de nettoyage du journal.	logCleanupFrequency	lc	5	0 - 60	minutes	J
Période d'historique du journal.	logHistory	lh	10	>=0	jours	J
Logman - Règles des phases d'exécution minimum et maximum.	logmanMinMaxPolicy	lm	both		literal	J
Logman - règle de calcul du temps d'exécution normal.	logmanSmoothPolicy	lt	-1	0 - 100	factor	J
Activer l'audit de base de données.	enDbAudit	da	0	0, 1	booléen	Imm

Tableau 10. Audit et consignation (suite)

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Type de stockage à utiliser pour consigner les enregistrements d'audit de base de données	auditStore	as	file	db, file, both		Imm
Période d'historique d'audit	auditHistory	ah	180	>=1	jours	Imm

Tableau 11. Dépendances croisées

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Nombre de jours durant lequel le système réessaiera d'envoyer des notifications concernant des changements de statut de travail au moteur distant si la notification échoue.	notificationTimeout	nt	5	1-90	Nombre	Imm

Tableau 12. Open Services for Lifecycle Collaboration (OSLC)

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Description du fournisseur de services d'automatisation de Tivoli Workload Scheduler	oslcAutomationDescription	ad			nom	Imm
Titre du fournisseur de services d'automatisation de Tivoli Workload Scheduler	oslcAutomationTitle	à			nom	Imm
Nom d'hôte du fournisseur de services Tivoli Workload Scheduler (nom d'hôte du gestionnaire de domaine maître actif)	oslcProviderUri	pu			nom	Imm
Description du fournisseur de services d'application de Tivoli Workload Scheduler	oslcProvisioningDescription	pd			nom	Imm
Titre du fournisseur de services d'application de Tivoli Workload Scheduler	oslcProvisioningTitle	pt			nom	Imm
Mot de passe associé à l'utilisateur qui se connecte aux services de registre	oslcRegistryPassword	rp			nom	Imm
Adresse des services de registre	oslcRegistryUri	cu			nom	Imm
Utilisateur qui se connecte aux services de registre	oslcRegistryUser	ru			nom	Imm

Tableau 13. SmartCloud Control Desk

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Adresse des SmartCloud Control Desk	sccdUrl	du			nom	Imm
Utilisateur qui se connecte aux SmartCloud Control Desk	sccdUserName	dn			nom	Imm
Mot de passe associé à l'utilisateur qui se connecte aux SmartCloud Control Desk	sccdUserPassword	dp			nom	Imm

Tableau 14. Général

Description	Nom	Nom abrégé	Par défaut	Plage	Unités	Effet
Nom de la société	companyName	cn			nom	J
Activer la sécurité centralisée	enCentSec	ts	no	yes, no	booléen	J
Activer l'ID de flot de travaux précédent	enLegacyId	li	no	yes, no	booléen	J
Evaluation de début de journée	enLegacyStartOfDayEvaluation	le	no	yes, no	booléen	J
Activer la vérification de sécurité de la liste	enListSecChk	sc	no	yes, no	booléen	J (Plan) Imm (DB)
Activer l'audit du plan.	enPlanAudit	pa	0	0, 1	booléen	J
Activer le gestionnaire de commutateurs tolérant aux pannes	enSwfaultTol	sw	no	yes, no	booléen	J
Activer les fuseaux horaires	enTimeZone	tz	yes	yes, no	booléen	J (Plan) Imm (DB)
Ignorer les agendas	ignoreCals	ic	no	yes, no	booléen	J
Heure de début du jour de traitement	startOfDay	sd	0000	0000 –2359	HHMM	J
Période d'historique des statistiques de travaux	statsHistory	sh	10	>=0	jours	J (Plan) Imm (DB)

Options globales - Description détaillée

Cette section propose des descriptions détaillées des options globales gérées par **optman** :

approachingLateOffset | al

Approche du décalage tardif. Utilisée dans l'assurance de service de charge de travail. L'heure de début critique d'un travail dans le réseau critique est l'heure la plus tardive à laquelle le travail peut débuter sans provoquer la fin du travail critique après l'heure d'échéance. Dans la plupart des cas, un travail commence bien avant l'heure de début critique afin que, s'il se prolonge plus longtemps que sa durée estimée, la situation ne devienne pas immédiatement critique. En conséquence, si un travail n'a

pas commencé et que l'heure de début critique n'est qu'à quelques minutes, l'achèvement du travail dans les délais est considéré comme potentiellement menacé.

L'option *approachingLateOffset* permet de déterminer combien de temps avant l'heure de début critique d'un travail du réseau critique vous devez être informé de ce risque potentiel. Si un travail n'a toujours pas démarré dans l'intervalle de secondes indiqué avant l'heure de début critique, le travail est ajouté à une liste d'accès direct qui peut s'afficher dans Dynamic Workload Console.

Remarque : Pour qu'un travail soit ajouté à la liste d'accès direct, toutes les dépendances de temps et de suivi doivent avoir été résolues.

Cette option n'est active que si *enWorkloadServiceAssurance* est défini sur *yes*.

La valeur par défaut est de *120* secondes.

Remarque : Quelle que soit la valeur que vous définissez pour cette option, si Tivoli Workload Scheduler perd la connexion avec sa base de données, la valeur par défaut est appliquée au traitement des travaux critiques et le message d'avertissement AWSJCO135W est émis pour vous indiquer ce qui s'est passé.

Exécutez **JnextPlan** ou démarrez le WebSphere Application Server (**stopappserver** et **startappserver**) pour rendre ce changement effectif.

auditHistory | ah
Période d'historique d'audit. Option utilisée dans la gestion d'audit. Entrez le nombre de jours de conservation des données d'audit enregistrées. Les enregistrements d'audit sont supprimés sur une base premier entré, premier sorti.

La valeur par défaut est *180* jours. Cette option prend effet immédiatement.

auditStore | as
Type de stockage à utiliser pour consigner les enregistrements d'audit de base de données. Entrez l'un des éléments suivants.

file Pour indiquer qu'un fichier à plat du répertoire `TWA_home/TWS/audit/database` est utilisé pour stocker les enregistrements d'audit (valeur par défaut).

db Pour indiquer que la base de données de Tivoli Workload Scheduler est utilisée pour stocker les enregistrements d'audit.

both Pour consigner les enregistrements d'audit dans le fichier et dans la base de données.

Toute modification apportée à cette valeur prend effet immédiatement.

baseRecPrompt | bp
Nombre maximum d'invites après fin anormale. Indique le nombre maximum d'invites pouvant s'afficher pour l'opérateur suite à l'arrêt anormal d'un travail.

Par défaut, la valeur est *1000*. Exécutez **JnextPlan** pour que ce changement soit effectif.

bindUser | bu
Utilisateur servant à effectuer la liaison entre le travail reflet et les travaux distants. Indiquez l'ID utilisateur utilisé pour lier un travail reflet à

un travail distant lors du contrôle de sécurité des "dépendances croisées". Cet utilisateur doit disposer au minimum des autorisations suivantes dans le fichier de sécurité :

- Accès en *affichage* au *travail* et aux objets de *planification* devant être liés
- Accès à la *liste* aux objets de *travail* devant être liés

Toutefois, l'ID n'a pas à se trouver dans le registre utilisateur du moteur, ni à disposer d'un mot de passe, car cela n'est requis qu'à des fins d'autorisation.

La valeur par défaut est utilisateur_TWS. Toute modification apportée à cette valeur prend effet immédiatement.

carryStates | cs

Etat des travaux reportés. Option de préproduction qui affecte le fonctionnement de la commande *stageman*. Spécifiez les travaux, par état, à inclure dans les flots de travaux reportés. Vous devez délimiter l'état des travaux à l'aide de parenthèses, de guillemets ou d'apostrophes. Les virgules peuvent être remplacées par des espaces. Les états de travaux internes valides sont les suivants :

Tableau 15. Etats de travaux internes valides

abend	*abenp*	*add*	*bound*	*done*	*error*	*exec*
fail	*hold*	*intro*	*pend*	*ready*	*rjob*	*sched*
skel	*succ*	*succp*	*susp*	*wait*	*waitd*	

Voici des exemples de définition de cette option :

```
carryStates="abend,exec,hold,intro"
carryStates='abend,exec,hold,intro'
carryStates="abend, exec, hold, intro"
carryStates='abend, exec, hold, intro'
```

Une liste vide doit être entrée de la façon suivante :

```
carryStates=null
```

La valeur par défaut est *null*, ce qui correspond à la sélection de tous les statuts. Exécutez **JnextPlan** pour que ce changement soit effectif.

companyName | cn

Nom de la société. Spécifiez le nom de votre société. La longueur maximale est de 40 octets. Si le nom contient des espaces, placez-le entre guillemets ("). Si vous utilisez le module de langue japonais (Katakana), placez le nom entre apostrophes ou entre guillemets.

Exécutez **JnextPlan** pour que ce changement soit effectif.

deadlineOffset | do

Décalage de l'échéance. Utilisée dans l'assurance de service de charge de travail. Permet de calculer le début critique d'un travail critique dans le cas où une échéance n'a pas été spécifiée, ni pour le travail, ni pour son flot de travaux . Dans ce cas, l'échéance déterminer par défaut correspond à la date et à l'heure de fin du plan, augmentée du décalage défini, et exprimée en minutes.

Cette option n'est active que si *enWorkloadServiceAssurance* est défini sur *yes*.

La valeur par défaut est *2* minutes.

Remarque :

1. **Important** : lorsque le plan est étendu, l'heure de début des travaux critiques dont l'échéance est calculée à l'aide de ce mécanisme est automatiquement modifiée en conséquence du fait qu'elle doit maintenant correspondre à la nouvelle heure de fin du plan.
2. Quelle que soit la valeur que vous définissez pour cette option, si Tivoli Workload Scheduler perd la connexion avec sa base de données, la valeur par défaut est appliquée au traitement des travaux critiques et le message d'avertissement AWSJCO135W est émis pour vous indiquer ce qui s'est passé.

Exécutez **JnextPlan** ou démarrez le WebSphere Application Server (**stopappserver** et **startappserver**) pour rendre ce changement effectif.

deploymentFrequency | df
Fréquence de déploiement des règles. Utilisée dans la gestion des règles d'événement. Spécifiez la fréquence, en minutes, de vérification des règles afin de détecter les éventuelles modifications à déployer. Toutes les règles actives (pour lesquelles la propriété `isDraft` est définie sur `no` dans leur définition) qui ont été modifiées ou ajoutées depuis le dernier déploiement sont déployées.

Les valeurs valides sont comprises dans la plage *0-60* minutes. Si vous spécifiez *0*, les modifications ne sont pas déployées automatiquement et vous devez utiliser la commande **planman deploy**.

La valeur par défaut est *5* minutes. La modification prend immédiatement effet.

enAddUser | au
Permet d'ajouter automatiquement un utilisateur dans le fichier Symphony. Cette option permet d'ajouter automatiquement un utilisateur au fichier Symphony après l'avoir créé ou modifié dans la base de données. Si vous spécifiez "yes" (oui), l'utilisateur est ajouté automatiquement au plan. Si vous spécifiez "no" (non), l'utilisateur n'est pas ajouté automatiquement au plan.

La valeur par défaut est "yes" (oui). Les changements apportés à ce paramètre prennent effet immédiatement.

Pour plus d'informations sur l'utilisation de cette fonction, voir "Tivoli Workload Scheduler : Guide d'utilisation et de référence".

enAddWorkstation | aw
Permet d'ajouter automatiquement le poste de travail de l'agent dynamique dans le fichier Symphony. Cette option permet d'ajouter automatiquement dans le fichier Symphony un poste de travail d'agent dynamique après avoir créé ce poste de travail dans la base de données. Si vous spécifiez "yes" (oui), le poste de travail de l'agent dynamique est ajouté automatiquement au plan. Si vous spécifiez "no" (non), le poste de travail de l'agent dynamique n'est pas ajouté automatiquement au plan.

La valeur par défaut est "no". Les changements apportés à ce paramètre prennent effet immédiatement.

Pour plus d'informations sur l'utilisation de cette fonction, voir "Tivoli Workload Scheduler : Guide d'utilisation et de référence".

enCarryForward | cf
Activer le report. Option de préproduction qui affecte le fonctionnement de la commande *stageman*. Spécifiez si les flots de travaux non terminés sont reportés depuis l'ancien vers le nouveau plan de

production(Symphony). Entrez `yes` pour que les flots de travaux inachevés soient reportés uniquement si l'option *Carry Forward* est activée dans la définition de Planificateur de travaux. Entrez `all` pour que tous les flots de travaux inachevés soient reportés, indépendamment de l'option *Carry Forward*. Entrez `no` pour désactiver complètement la fonction *Carry Forward*. Si vous exécutez la commande `JnextPlan -for 0000` et que l'option *Carry Forward* est définie sur `yes` ou sur `no`, un message s'affiche, vous informant que les flots de travaux inachevés ne seront pas reportés. Lorsque la commande **stageman -carryforward** est utilisée, elle remplace *enCarryForward*. Pour plus d'informations, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*. Si cette option est définie sur no, les travaux en cours sont déplacés vers le flot de travaux USERJOBS.

Par défaut, la valeur est *all*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enCentSec | ts
Activer la sécurité centralisée. Déterminez la manière dont le fichier de sécurité est utilisé dans le réseau. La sécurité centralisée ne convient pas aux environnements de planification de bout en bout.

Si elle est définie sur *yes*, les fichiers de sécurité de tous les postes de travail du réseau peuvent être créés et modifiés uniquement sur le gestionnaire de domaine maître. Dans ce cas, l'administrateur de Tivoli Workload Scheduler est chargé de leur production, de leur maintenance et de leur distribution.

Si elle est définie sur *no*, le fichier de sécurité de chaque poste de travail peut être géré par l'utilisateur root ou l'administrateur du système. L'utilisateur local peut exécuter la commande *makesec* pour créer ou mettre à jour le fichier.

Pour plus d'informations à propos de la sécurité centralisée, reportez-vous au *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enCFinterNetworkDeps | ci
Activer le report pour les dépendances interréseaux. Option de préproduction qui affecte la manière dont **stageman** gère les dépendances inter-réseau. Elle spécifie si les flux de travaux externes sont reportés de l'ancien vers le nouveau plan de production (fichier Symphony). Entrez *yes* pour que tous les flux de travaux externes soient reportés. Entrez *no* pour qu'aucun flux de travaux externe ne soit reporté.

La valeur par défaut est *yes*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enCFResourceQuantity | rq
Activer le report par quantité de ressources. Option de préproduction affectant la manière dont **stageman** gère les ressources. Entrez *yes* pour reporter la quantité de ressources de l'ancien fichier de production vers le nouveau. Entrez *no* pour ne pas reporter la quantité de ressources. **Stageman** reporte les quantités de ressources uniquement si la ressource est requise par un travail ou un flux de travaux faisant également l'objet d'un report. Dans le cas contraire, les quantités de ressources sont définies sur la valeur d'origine. Pour plus d'informations sur l'utilisation de cette fonction, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

La valeur par défaut est *yes*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enDbAudit | da
Activer l'audit de base de données. Activez ou désactivez l'audit de base de données. Pour désactiver l'audit de base de données, spécifiez *0*. Pour activer l'audit de base de données, spécifiez *1*. Les informations d'audit sont consignées dans un fichier à plat dans le répertoire *TWA_home*/TWS/audit/database, dans la base de données Tivoli Workload Scheduler elle-même ou dans les deux emplacements. Pour effectuer votre sélection, définissez la propriété *auditStore* dans la commande **optman**. Chaque poste de travail Tivoli Workload Scheduler gère son propre journal. Seules les actions sont consignées sans préciser si elles ont abouti ou non. L'installation de gestionnaires de domaine dynamique et de agents n'est pas enregistrée dans les journaux d'audit.

Pour plus d'informations sur l'utilisation de cette fonction, reportez-vous à la section consacrée aux utilitaires d'audit dans *Guide d'identification et de résolution des problèmes*.

Par défaut, la valeur est *0*. Les modifications apportées à ce paramètre prennent immédiatement effet.

enEmptySchedsAreSucc | es
Flots de travaux sans règle de travail. Spécifiez le comportement des flux de travaux ne comprenant pas de travaux. Si cette option est définie sur *yes*, les flux de travaux ne contenant pas de travaux sont définis sur SUCC après que leurs dépendances ont été résolues. Si elle est définie sur *no*, les flux de travaux conservent le statut READY.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enEventDrivenWorkloadAutomation | ed
Activer l'automatisation d'une charge de travail gérée par des événements. Activez ou désactivez la fonction d'automatisation de la charge de travail gérée par les événements. Pour activer cette option, spécifiez *yes*. Pour la désactiver, spécifiez *no*.

La valeur par défaut est *yes*.

Après la désactivation, vous devez exécuter **JnextPlan** et arrêter le serveur de traitement d'événement (avec la commande **conman stopevtp**).

Après l'activation, vous devez exécuter **JnextPlan** et démarrer le serveur de traitement d'événement (avec la commande **conman startevtp**).

enEventDrivenWorkloadAutomationProxy | pr
Activer le proxy d'automatisation d'une charge de travail gérée par des événements. Activez ou désactivez la fonction de proxy d'automatisation de la charge de travail gérée par les événements. Pour activer cette option, spécifiez *yes*. Pour la désactiver, spécifiez *no*.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enEventProcessorHttpsProtocol | eh
Activer le protocole HTTPS du processeur d'événements. Utilisée dans la gestion des règles d'événement. Permet d'activer ou de désactiver l'utilisation du protocole HTTPS pour la connexion au serveur du processeur d'événements. Pour activer cette option, entrez *yes*. Pour la désactiver, entrez *no*.

La valeur par défaut est *yes*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enForecastStartTime | st

Activer le calcul d'heure de début prévue. Applicable uniquement si l'assurance de service de charge de travail est activée (voir *enWorkloadServiceAssurance*). Entrez *yes* pour activer le calcul de l'heure de début prévue pour chaque travail lors de l'exécution d'un plan prévisionnel. L'activation de cette fonction peut affecter de façon négative la durée nécessaire à la génération du plan prévisionnel. Entrez *no* pour désactiver le calcul de l'heure de début prévue pour chaque travail lors de l'exécution d'un plan prévisionnel.

La valeur par défaut est *no*. Tout changement apporté à cette valeur prend effet immédiatement.

Lorsque cette option est définie sur *yes*, l'option globale **enPreventStart** est ignorée au cours de la création des plans prévisionnels.

enLegacyId | li

Activer l'ID de flot de travaux précédent. Déterminez la manière dont les flots de travaux doivent être nommés lors de leur fonctionnement dans des environnements mixtes avec des versions de Tivoli Workload Scheduler antérieures à la version 8.3, gérées par une version 8.5 gestionnaire de domaine maître. Cette option n'est pas prise en charge par le catalogue libre-service qui l'ignore même si sa valeur est définie à YES.Cette option permet de maintenir la cohérence de l'identification de flots de travaux dans le plan. La valeur attribuée à cette option est lue soit lorsque le plan de production est créé ou étendu, soit lors de la soumission du flots de travaux dans la production à l'aide de la commande.

Lorsque le plan est créé ou étendu, si cette option est définie sur *no*, l'instance du Planificateur de travaux reçoit un nouvel ID suivant le mécanisme normal du Tivoli Workload Scheduler. Dans le fichier Symphony, le nom de Planificateur de travaux est identique à cet ID. Si l'option est définie sur *yes*, un ID (ID Symphony) équivalent au nom Planificateur de travaux est affecté à l'instance Planificateur de travaux. Dans le fichier Symphony, le nom du Planificateur de travaux est égal au nom réel du Planificateur de travaux. Si plusieurs instances du même Planificateur de travaux sont présentes, un ID est généré pour chaque instance, avec un alias commençant par le nom du Planificateur de travaux.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enLegacyStartOfDayEvaluation | le

Evaluation de début de journée. Spécifiez comment l'option *startOfDay* doit être gérée dans le réseau Tivoli Workload Scheduler. Si vous définissez cette option sur *yes*, la valeur de *startOfDay* dans le gestionnaire de domaine maître est convertie dans le fuseau horaire local défini pour chaque poste de travail du réseau. Si vous définissez cette option sur *no*, la valeur de *startOfDay* sur le gestionnaire de domaine maître est appliquée en l'état à chaque poste de travail du réseau. Cette option requiert que l'option *enTimeZone* soit définie sur *yes* pour devenir opérationnelle.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enListSecChk | sc

Activer le contrôle de sécurité de liste. Contrôlez les objets dans le plan qu'un utilisateur est autorisé à répertorier lors de l'exécution d'une requête sur la commande Dynamic Workload Console or a **conman show <object>**. Si elle est définie sur *yes*, les objets du plan renvoyés à partir d'une requête ou d'une commande show sont affichés uniquement si l'utilisateur bénéficie des droits d'accès list dans le fichier de sécurité. Si elle est définie sur *no*, tous les objets sont affichés, quels que soient les paramètres du fichier de sécurité.

Remarque : La définition de cette option sur *yes* affecte la manière dont les interfaces utilisateur graphiques fonctionnent pour les utilisateurs définis dans le fichier de sécurité.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que cette modification soit effective pour le plan. Pour la base de données, cette option prend effet immédiatement.

enLogonBatch | lb

Activer automatiquement la connexion par lot. Cela concerne uniquement les travaux sous Windows. Si elle est définie sur *yes*, les utilisateurs de la connexion pour les travaux sousWindows sont automatiquement autorisés à *se connecter par lot*. Si elle a pour valeur *no*, ou si elle n'est pas définie, ce droit doit être accordé manuellement à chaque utilisateur ou groupe. Ce droit ne peut pas être accordé automatiquement aux utilisateurs exécutant des travaux sur un gestionnaire de domaine de secours : vous devez donc octroyer ces droits manuellement.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enPlanAudit | pa

Activer l'audit du plan. Activez ou désactivez l'audit du plan. Pour activer l'audit du plan, spécifiez *1*. Pour désactiver l'audit du plan, spécifiez *0*. Les informations d'audit sont consignées dans un fichier plat du répertoire *TWA_home*/TWS/audit/plan. Chaque poste de travail Tivoli Workload Scheduler gère son propre journal. Pour le plan, seules les actions sont consignées dans le fichier d'audit, mais pas la réussite ou l'échec d'une action. Reportez-vous à *Tivoli Workload Scheduler - Guide d'utilisation et de référence* pour plus d'informations à propos de cette fonction.

La valeur par défaut est *0*. Exécutez **JnextPlan** pour que cette modification soit effective.

enPreventStart | ps

Empêcher le démarrage des flots de travaux sans dépendance "at". Indiquez si le démarrage immédiat des flots de travaux sans dépendance *at* doit être empêché, sans attendre le cycle d'exécution spécifié dans le Planificateur de travaux. Les valeurs valides sont *yes* et *no*.

La valeur par défaut est *yes*. Exécutez **JnextPlan** pour que ce changement soit effectif.

Lorsque l'option **enForecastStartTime** est définie sur *yes*, cette option est ignorée au cours de la création des plans prévisionnels.

enRetainNameOnRerunFrom | rr

Conserver le nom des travaux de réexécution. Option de production qui affecte le fonctionnement de **Batchman**, le processus de contrôle de production de Tivoli Workload Scheduler. Sa valeur détermine si les

travaux qui sont de nouveau exécutés à l'aide de la commande **Conman** *rerun* conservent leur nom de travail d'origine. Pour que les travaux ré-exécutés conservent leur nom d'origine, entrez *yes*. Saisissez *no* pour attribuer un nom *rerun from* aux travaux ré-exécutés.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que la modification soit effective.

enSSLFullConnection | sf
Activer la connexion SSL complète. Spécifiez que Tivoli Workload Scheduler utilise un niveau de connexion SSL supérieur au niveau standard. Pour obtenir des détails complets, voir «Configuration de la sécurité SSL totale», à la page 292. Les valeurs valides sont *yes* pour activer la connexion SSL complète ou *no* pour désactiver la connexion SSL complète.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enStrEncrypt | se
Activer le chiffrement renforcé des mots de passe. Activez ou désactivez le chiffrement renforcé. Activez le chiffrement renforcé en attribuant la valeur *yes* à cette option. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enSwfaultTol | sw
Activer le gestionnaire de commutateurs tolérant aux pannes. Activez ou désactivez la fonction de gestionnaire de commutateurs tolérant aux pannes. Les valeurs valides sont *yes* pour activer le gestionnaire de commutateurs tolérant aux pannes, et *no* pour le désactiver. Cette option n'a pas de fonctions dynamiques et n'est pas conçue pour fonctionner avec des agents de courtier. Pour plus de détails, reportez-vous à *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

La valeur par défaut est *no*. Exécutez **JnextPlan** pour que ce changement soit effectif.

enTimeZone | tz
Activer les fuseaux horaires. Active ou désactive l'option de fuseau horaire. Pour activer les fuseaux horaires dans votre réseau, spécifiez *yes*. Pour désactiver les fuseaux horaires dans le réseau, spécifiez *no*. Voir «Activation de la fonction de fuseau horaire», à la page 95.

La valeur par défaut est *yes*. Exécutez **JnextPlan** pour que ce changement soit effectif dans le plan. Pour la base de données, cette option prend effet immédiatement.

enWorkloadServiceAssurance | wa
Activer l'assurance de service de charge de travail. Active ou désactive l'assurance de service de charge de travail, qui est la fonction qui gère le traitement privilégié des travaux critiques et de leurs prédécesseurs. Spécifiez *yes* pour l'activer ou *no* pour la désactiver.

Remarque : Avant d'utiliser l'assurance de service de charge de travail vous devez configurer *utilisateur_TWS* dans le fichier de sécurité de manière à avoir accès aux objets que cette fonction modifie - voir «*utilisateur_TWS* - Remarques particulières relatives au fichier de sécurité», à la page 190

La valeur par défaut est *yes*. Exécutez **JnextPlan** pour que ce changement soit effectif.

eventProcessorEIFPort | ee
Port de la fonction d'intégration d'événements **Tivoli.** Utilisée dans la gestion des règles d'événement. Spécifiez le numéro de port où le serveur du processeur d'événements reçoit les événements de la fonction d'intégration des événements Tivoli (EIF). Les valeurs valides se situent dans la plage *0-65535*.

La valeur par défaut est *31131*. Si vous modifiez la valeur, redémarrez le WebSphere Application Server (**stopappserver** et **startappserver**) et exécutez **JnextPlan** pour rendre ce changement effectif.

Si vous utilisez un pare-feu de sécurité, vérifiez que ce port est ouvert pour les connexions entrantes et sortantes.

extRecPrompt | xp
Invites supplémentaires après fin anormale. Spécifiez un nombre supplémentaire d'invites pour la valeur définie dans *baseRecPropmt*. Cela s'applique lorsqu'un travail est ré-exécuté après un abandon et que la limite spécifiée dans *baseRecPropmt* a été atteinte.

Par défaut, la valeur est *1000*. Exécutez **JnextPlan** pour que ce changement soit effectif.

ignoreCals | ic
Ignorer les agendas. Option de préproduction qui affecte le fonctionnement de la commande **planman**. Sa valeur détermine si les calendriers d'utilisateur sont copiés dans le nouveau fichier du plan de production (Symphony). Pour empêcher les calendriers d'utilisateur d'être copiés dans le nouveau plan de production, entrez *yes*.

La valeur par défaut est *no*. Voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*. Exécutez **JnextPlan** pour que ce changement soit effectif.

logCleanupFrequency | lc
Fréquence de nettoyage du journal. Option utilisée dans la gestion d'audit et des règles d'événement. Spécifiez la fréquence à laquelle l'apurement automatique des instances du journal est exécuté. Les valeurs valides sont comprises dans la plage *0-60* minutes. Si vous spécifiez *0*, la fonction d'apurement automatique est désactivée.

La valeur par défaut est *5* minutes. Cette option prend effet immédiatement.

logHistory | lh
Période d'historique du journal. Utilisée dans la gestion des règles d'événement. Entrez le nombre de jours pour lequel vous voulez enregistrer l'instance de règle, l'action exécutée et les données du journal des messages. Les instances de journaux sont éliminées selon la règle premier entré, premier sorti.

La valeur par défaut est *10* jours. Cette option prend effet immédiatement.

logmanMinMaxPolicy | lm
Logman - Règles des phases d'exécution minimum et maximum. Spécifiez la manière dont les répétitions minimales et maximales de travaux doivent être consignées et rapportées par **logman**. Les valeurs possibles sont les suivantes :

elapsedtime
Les temps d'exécution minimal et maximal écoulés sont consignés et rapportés.

cputime
Les temps d'exécution minimal et maximal d'unité centrale sont consignés et rapportés.

both Les temps d'exécution minimal et maximal des travaux sont consignés et rapportés.

Pour plus d'informations sur l'utilisation de cette fonction, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Par défaut, la valeur est *both*. Exécutez **JnextPlan** pour que ce changement soit effectif.

logmanSmoothPolicy | lt
Logman - règle de calcul du temps d'exécution normal. Cette option définit le facteur de pondération qui favorise le travail le plus récemment exécuté lors du calcul du délai d'exécution normal (moyen) d'un travail. Le résultat de ce calcul est exprimé sous la forme d'un pourcentage. Par exemple, spécifiez *40* pour appliquer un facteur de pondération de 40 % à l'exécution de travail la plus récente, et 60 % pour la moyenne existante. Reportez-vous à *Tivoli Workload Scheduler - Guide d'utilisation et de référence* pour plus d'informations à propos de l'utilisation de cette option.

La valeur par défaut est *-1*. Exécutez **JnextPlan** pour que ce changement soit effectif.

longDurationThreshold | ld
Seuil de travail de longue durée. Après comparaison de la durée réelle d'un travail par rapport à sa durée estimée, cette option spécifie le seuil au-delà duquel le travail est considéré comme étant de longue durée. La valeur du seuil est exprimée sous forme de pourcentage de la durée estimée. Par exemple, si le seuil est défini sur *150*, et que la durée réelle est supérieure à 150 % de la durée estimée (elle est supérieure de 50 %), le travail est considéré comme étant de "longue durée".

Si la fonction d'assurance de service de charge de travail est activée, les travaux critiques satisfaisant aux critères de longue durée sont automatiquement insérés dans la liste d'accès direct.

Les valeurs valides sont comprises entre :
100 La valeur minimale. Tous les travaux dépassant la durée estimée sont considérés comme étant de longue durée.
1000 La valeur maximale. Seuls les travaux durant dix fois plus longtemps que leur durée estimée sont considérés comme des travaux de longue durée.

La valeur par défaut est *150*.

Remarque : Quelle que soit la valeur que vous définissez pour cette option, si vous avez activé la fonction d'assurance de service de la charge de travail et que Tivoli Workload Scheduler perd la connexion avec sa base de données, la valeur par défaut est appliquée au traitement des travaux critiques et le message d'avertissement AWSJCO135W est émis pour vous indiquer ce qui s'est passé.

Exécutez **JnextPlan** ou démarrez le WebSphere Application Server (**stopappserver** et **startappserver**) pour rendre ce changement effectif.

mailSenderName | ms
Nom de l'expéditeur du message. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action qui envoie des e-mails via un serveur SMTP, spécifiez une chaîne à utiliser en tant qu'expéditeur des e-mails.

La valeur par défaut est *TWS*. Les changements apportés à ce paramètre prennent effet lors de l'envoi du prochain message.

maxLen | xl
Durée maximale du plan de préproduction. Spécifiez la longueur maximale du plan de préproduction en jours après qu'il ait été automatiquement étendu ou créé. La valeur de *maxLen* doit être supérieure ou égale à la valeur de *minLen* et doit figurer dans la plage de *8* à *365*.

La valeur par défaut est *14* jours. Exécutez **JnextPlan** pour que ce changement soit effectif.

minLen | ml
Durée minimale du plan de préproduction. Spécifiez la longueur minimale en jours du plan de préproduction qui peuvent s'écouler après la création ou l'extension du plan de préproduction, sans étendre le plan de préproduction. Si le nombre de jours restant dans le plan de préproduction après un **JnextPlan** est inférieur à la valeur de cette option, le plan de préproduction est automatiquement étendu. La valeur de *minLen* doit être inférieure ou égale à la valeur de *maxLen* et doit se situer dans la plage de *7* à *365*.

La valeur par défaut est *8* jours. Exécutez **JnextPlan** pour que ce changement soit effectif.

notificationTimeout | nt
Expiration du délai de notification Utilisée dans les dépendances croisées. Indiquez pendant combien de jours Tivoli Workload Scheduler tentera de renvoyer au moteur distant des notifications relatives aux modifications d'état de travail, lorsque la notification échoue. A l'expiration de ce délai, l'abonnement à la demande de travail et les notifications de statut associées à ce travail sont supprimés.

Les valeurs valides se situent dans la plage de *1* à *90*. La valeur par défaut est *5* jours. Les changements apportés à ce paramètre prennent effet immédiatement.

oslcAutomationDescription | ad
Description du fournisseur de services d'automatisation de Tivoli Workload Scheduler. Utilisé dans l'intégration OSLC pour enregistrer le fournisseur de services d'automatisation de Tivoli Workload Scheduler dans les services de registre. Cette valeur permet de définir une description pour le fournisseur de services.

Les changements apportés à ce paramètre prennent effet immédiatement.

oslcAutomationTitle | at
Titre du fournisseur de services d'automatisation de Tivoli Workload Scheduler. Utilisé dans l'intégration OSLC pour enregistrer le fournisseur de services d'automatisation de Tivoli Workload Scheduler dans les services de registre. Cette valeur permet d'identifier de manière unique le fournisseur de services d'automatisation. Pour identifier facilement le

fournisseur de services que vous voulez utiliser, utilisez un titre significatif pour chaque fournisseur de services d'automatisation de Tivoli Workload Scheduler enregistré dans les mêmes services de registre.

Les changements apportés à ce paramètre prennent effet immédiatement.

oslcProviderUri | pu
Adresse du fournisseur de services Tivoli Workload Scheduler. Utilisé dans l'intégration OSLC pour enregistrer le fournisseur de services Tivoli Workload Scheduler dans les services de registre. Utilisez le format `https://nom_hôte:port`, ou `nom_hôte` est le nom de l'hôte utilisé pour la connexion au gestionnaire de domaine maître. Par exemple, `https://myProviderHostanme.com:31115`.

Les changements apportés à ce paramètre prennent effet immédiatement.

oslcProvisioningDescription | pd
Description du fournisseur de services d'automatisation de Tivoli Workload Scheduler. Utilisé dans l'intégration OSLC pour enregistrer le fournisseur de services d'automatisation de Tivoli Workload Scheduler dans les services de registre. Cette valeur permet de définir une description pour le fournisseur de services.

Les changements apportés à ce paramètre prennent effet immédiatement.

oslcProvisioningTitle | pt
Titre du fournisseur de services d'application Tivoli Workload Scheduler. Utilisé dans l'intégration OSLC pour enregistrer le fournisseur de services d'application de Tivoli Workload Scheduler dans les services de registre. Cette valeur permet d'identifier de manière unique le fournisseur de services d'application. Pour identifier facilement le fournisseur de services que vous voulez utiliser, utilisez un titre significatif pour chaque fournisseur de services d'application de Tivoli Workload Scheduler enregistré dans les mêmes services de registre.

Les changements apportés à ce paramètre prennent effet immédiatement.

oslcRegistryPassword | rp
Mot de passe de l'utilisateur qui se connecte aux services de registre. Utilisé dans l'intégration OSLC pour enregistrer le fournisseur de services Tivoli Workload Scheduler dans les services de registre.

Les changements apportés à ce paramètre prennent effet immédiatement.

oslcRegistryUri | cu
Adresse des services de registre. Utilisé dans l'intégration OSLC pour enregistrer le fournisseur de services Tivoli Workload Scheduler dans les services de registre. Utilisez le format `https://hostname:port/oslc/pr`.

Les changements apportés à ce paramètre prennent effet immédiatement.

oslcRegistryUser | ru
Utilisateur qui se connecte aux services de registre. Utilisé dans l'intégration OSLC pour enregistrer le fournisseur de services Tivoli Workload Scheduler dans les services de registre.

Les changements apportés à ce paramètre prennent effet immédiatement.

promotionOffset | po
Décalage de promotion. Utilisé dans l'assurance de service de charge de travail. Spécifiez lorsqu'un travail devient éligible pour promotion en termes du nombre de secondes avant que son heure de début critique ne soit atteinte. S'applique uniquement aux travaux marqués comme critiques

dans une définition de flot de travaux et des travaux prédécesseurs. Un travail critique et ses prédécesseurs constituent un réseau critique.

Lorsqu'un prédécesseur met en péril l'achèvement du travail critique dans les délais, il est *promu* ; en d'autres termes, des ressources supplémentaires lui sont affectées et sa soumission devient prioritaire par rapport aux autres travaux qui se situent hors du réseau critique. Des travaux critiques peuvent également être promus.

Le planificateur calcule l'heure de début critique d'un travail critique en soustrayant sa durée estimée de l'échéance. Il calcule l'heure de début critique d'un prédécesseur critique en soustrayant sa durée estimée de l'heure de début critique de son successeur direct. Au sein d'un réseau critique, le planificateur calcule l'heure de début critique du premier travail en premier lieu, puis fonctionne à rebours dans la chaîne de ses prédécesseurs. Ces calculs sont réitérés autant de fois que nécessaire jusqu'à ce que le travail critique ait été exécuté.

Cette option n'est active que si *enWorkloadServiceAssurance* est défini sur *yes*.

La valeur par défaut est de *120* secondes.

Exécutez **JnextPlan** pour que ce changement soit effectif.

sccdUrl | du
Adresse de SmartCloud Control Desk. Utilisée dans la gestion des règles d'événement. L'adresse URL de SmartCloud Control Desk, au format `http://nom_hôte:[port]`. Si vous ne spécifiez pas de numéro de port, la valeur par défaut utilisée est *80*.

Les changements apportés à ce paramètre prennent effet immédiatement.

sccdUserName | dn
Utilisateur qui se connecte à SmartCloud Control Desk. Utilisée dans la gestion des règles d'événement. Identificateur de l'utilisateur qui se connecte à SmartCloud Control Desk.

Les changements apportés à ce paramètre prennent effet immédiatement.

sccdUserPassword | dp
Mot de passe de l'utilisateur qui se connecte à SmartCloud Control Desk. Utilisée dans la gestion des règles d'événement. Le mot de passe qui est associé à l'utilisateur qui se connecte à SmartCloud Control Desk.

Les changements apportés à ce paramètre prennent effet immédiatement.

smtpServerName | sn
Nom du serveur SMTP. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action qui envoie des e-mails via un serveur SMTP, spécifiez le nom du serveur SMTP à utiliser par le plug-in de messagerie.

La valeur par défaut est *localhost*. Les changements apportés à ce paramètre prennent effet immédiatement.

smtpServerPort | sp
Port du serveur SMTP. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action qui envoie des e-mails via un serveur SMTP, spécifiez le numéro de port utilisé par le plug-in de messagerie pour se connecter au serveur SMTP. Les valeurs valides se situent dans la plage *0–65535*.

La valeur par défaut est *25*. Les changements apportés à ce paramètre prennent effet lors de l'envoi du prochain message.

smtpUseAuthentication | ua
Module d'extension de messagerie utilise l'authentification SMTP. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action qui envoie des e-mails via un serveur SMTP, spécifiez si la connexion SMTP doit être authentifiée. Les valeurs sont *yes* ou *no*.

La valeur par défaut est *no*. Les changements apportés à ce paramètre prennent effet immédiatement.

smtpUserName | un
Nom d'utilisateur du serveur SMTP. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action qui envoie des e-mails via un serveur SMTP, spécifiez le nom d'utilisateur du serveur SMTP.

La valeur par défaut est le nom de l'utilisateur Tivoli Workload Scheduler (le utilisateur_TWS) sur le gestionnaire de domaine maître. Les changements apportés à ce paramètre prennent effet immédiatement.

smtpUserPassword | up
Mot de passe d'utilisateur du serveur SMTP. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action qui envoie des e-mails via un serveur SMTP, spécifiez le mot de passe de l'utilisateur du serveur SMTP. Le mot de passe est stocké sous forme chiffrée.

Les changements apportés à ce paramètre prennent effet immédiatement.

smtpUseSSL | us
Module d'extension de messagerie utilise SSL. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles qui implémentent une action d'envoi d'e-mails via un serveur SMTP, indiquez si la connexion SMTP doit être authentifiée via SSL. Les valeurs sont *yes* ou *no*.

La valeur par défaut est *no*. Les changements apportés à ce paramètre prennent effet immédiatement.

smtpUseTLS | tl
Module d'extension de messagerie utilise le protocole TLS. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles mettant en oeuvre une action d'envoi d'e-mails via un serveur SMTP, indiquez si la connexion SMTP doit être authentifié via le protocole TLS (Transport Layer Security). Les valeurs sont *yes* ou *no*.

La valeur par défaut est *no*. Les changements apportés à ce paramètre prennent effet immédiatement.

startOfDay | sd
Heure de début du jour de traitement. Spécifiez l'heure de début du jour de traitement Tivoli Workload Scheduler au format 24 heures : *hhmm* (*0000-2359*).

La valeur par défaut est *0000* (minuit), mais si vous avez mis à niveau votre environnement vers la version V9.2 à partir d'une version antérieure à la version 8.6, la valeur par défaut est *0600* (6h00 le matin). Si vous modifiez cette option, vous devez également modifier l'heure de lancement

du Planificateur de travaux *final*, qui est généralement définie à une minute avant l'heure de début. Exécutez **JnextPlan** pour que le changement de *startOfDay* soit effectif.

statsHistory | sh
Période d'historique des statistiques de travaux. Indiquez le nombre de jours pendant lequel vous voulez conserver les statistiques des travaux. Les statistiques sont éliminées en fonction de la règle premier entré, premier sorti. Par exemple, si vous laissez la valeur par défaut de *10*, les statistiques des 10 derniers jours seront conservées. Cette option n'a aucun effet sur les fichiers de liste standard des travaux, lesquels doivent être supprimés via la commande *rmstdlist*. Reportez-vous au *Tivoli Workload Scheduler - Guide d'utilisation et de référence* pour plus d'informations à propos de la commande *rmstdlist*.

La valeur par défaut est *10*. Exécutez **JnextPlan** pour que ce changement soit effectif dans le plan. Pour la base de données, cette option prend effet immédiatement.

TECServerName | th
Nom du serveur Sonde EIF. Utilisée dans la gestion des règles d'événement. Si vous utilisez des règles implémentant une action qui transfère des événements vers un serveur Tivoli Enterprise Console ou un serveur Tivoli Business Systems Manager (ou toute autre application qui traite des événements au format TEC ou TBSM), indiquez le nom de serveur Sonde EIF. Pour utiliser un serveur Sonde EIF différent, vous pouvez changer cette valeur lorsque vous définissez l'action.

La valeur par défaut est *localhost*. Exécutez **JnextPlan** pour que ce changement soit effectif.

TECServerPort | tp
port du serveur Sonde EIF. Utilisée dans la gestion des règles d'événement. Si vous utilisez des règles implémentant une action qui transfère des événements vers un serveur Tivoli Enterprise Console ou un serveur Tivoli Business Systems Manager (ou toute autre application qui traite des événements au format TEC ou TBSM), indiquez le numéro de port du serveur Sonde EIF. Pour utiliser un serveur Sonde EIF différent, vous pouvez changer cette valeur lorsque vous définissez l'action.

Le numéro de port par défaut est *5529*. Exécutez **JnextPlan** pour que ce changement soit effectif.

workstationLimit | wl
Limite du poste de travail.

Utilisée dans l'enregistrement automatique de l'agent dynamique. Ce paramètre indique la valeur limite du poste de travail de l'agent dynamique que le poste de travail de l'agent dynamique prend en charge après l'ajout du poste de travail au plan. Vous pourrez modifier la valeur limite du poste de travail de l'agent dynamique ultérieurement à l'aide de la ligne de commande **conman** ou de l'Dynamic Workload Console.

Les valeurs valides se situent dans la plage *0-1024*.

La valeur par défaut est *100*. Les changements apportés à ce paramètre prennent effet immédiatement.

zOSRemoteServerName | zr
Nom du serveur distant du connecteur Tivoli Workload Scheduler for z/OS. Utilisée dans la gestion des règles d'événement. Si vous déployez

des règles implémentant une action de soumission de flots de travaux au contrôleur Tivoli Workload Scheduler for z/OS, entrez le nom du contrôleur indiqué comme moteur au connecteur z/OS. Il doit correspondre exactement au nom du moteur de connecteur z/OS et est sensible à la casse.

Une fois la valeur de ce paramètre modifiée, elle devient effective lorsque l'action `submit` suivante est exécutée.

zOSServerName | zs
Nom du serveur du connecteur Tivoli Workload Scheduler for z/OS. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action de soumission de flots de travaux au contrôleur Tivoli Workload Scheduler for z/OS, indiquez le nom ou le nom d'hôte du système sur lequel le connecteur Tivoli Workload Scheduler for z/OS s'exécute. La valeur par défaut est `localhost`.

Une fois la valeur de ce paramètre modifiée, elle devient effective lorsque l'action `submit` suivante est exécutée.

zOSServerPort | zp
Port du serveur du connecteur Tivoli Workload Scheduler for z/OS. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action de soumission de flots de travaux au contrôleur Tivoli Workload Scheduler for z/OS, indiquez le numéro du port d'amorce du serveur de connecteur Tivoli Workload Scheduler for z/OS. Les valeurs valides sont comprises entre 0 et 65535. La valeur par défaut est 31217.

Une fois la valeur de ce paramètre modifiée, elle devient effective lorsque l'action `submit` suivante est exécutée.

zOSUserName | zu
Nom de l'utilisateur du connecteur Tivoli Workload Scheduler for z/OS. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action de soumission de flots de travaux au contrôleur Tivoli Workload Scheduler for z/OS, indiquez le nom d'utilisateur du connecteur Tivoli Workload Scheduler for z/OS obligatoire pour accéder au moteur Tivoli Workload Scheduler for z/OS.

Une fois la valeur de ce paramètre modifiée, elle devient effective lorsque l'action `submit` suivante est exécutée.

zOSUserPassword | zw
Mot de passe de l'utilisateur du connecteur Tivoli Workload Scheduler for z/OS. Utilisée dans la gestion des règles d'événement. Si vous déployez des règles implémentant une action de soumission de flots de travaux au contrôleur Tivoli Workload Scheduler for z/OS, indiquez le mot de passe de l'utilisateur du connecteur Tivoli Workload Scheduler for z/OS obligatoire pour accéder au moteur Tivoli Workload Scheduler for z/OS. Le mot de passe est enregistré sous forme chiffrée.

Une fois la valeur de ce paramètre modifiée, elle devient effective lorsque l'action `submit` suivante est exécutée.

Définition des options locales

Définissez les options locales dans le fichier `localopts`. Les modifications ne prennent pas effet tant que **`netman`** n'est pas arrêté (**conman shut;wait**), puis redémarré (**StartUp**).

Un fichier modèle contenant les paramètres par défaut se trouve dans *TWA_home*/TWS/config/localopts.

Remarque : Tous les paramètres SSL du fichier localopts s'appliquent aux communications réseau, mais pas à Dynamic Workload Console.

Au cours du processus d'installation, une copie de travail du fichier d'options local est installée en tant que *TWA_home*/TWS/localopts.

Les options du fichier localopts sont décrites dans les sections suivantes :

- «Récapitulatif de Localopts»
- «Détails de Localopts», à la page 34
- «Exemple de fichier des options locales», à la page 47

Récapitulatif de Localopts

Attributs généraux du poste de travail :

thiscpu = *workstation*
merge stdlists = *yes*|*no*
stdlist width = *colonnes*
syslog local = *fonction*
restricted stdlists = *yes*|*no*

Attributs du poste de travail pour le processus batchman :

bm check file = *seconds*
bm check status = *seconds*
bm look = *seconds*
bm read = *seconds*
bm stats = *on*|*off*
bm verbose = *on*|*off*
bm check until = *seconds*
bm check deadline = *seconds*
bm late every = *minutes*

Attributs du poste de travail pour le processus jobman :

jm interactive old = *yes*|*no*
jm job table size = *entries*
jm load user profile = *on*|*off*
jm look = *seconds*
jm nice = *value*
jm promoted nice = Priorité du travail critique *UNIX et Linux*
jm promoted priority = priorité du travail critique *Windows*
jm no root = *yes*|*no*
jm read = *seconds*

Attributs du poste de travail pour le processus mailman :

mm planoffset = *HHMM*
mm response = *seconds*
mm retrylink = *seconds*
mm sound off = *yes*|*no*
mm unlink = *seconds*
mm cache mailbox = *yes*|*no*
mm cache size = *octets*

mm resolve master = *yes* | *no*
autostart monman = *yes* | *no*
mm read = *minutes*

Attributs du poste de travail pour le processus netman :

nm mortal = *yes* | *no*
nm port = *numéro de port*
nm read = *seconds*
nm retry = *seconds*

Attributs du poste de travail pour le processus writer :

wr read = *seconds*
wr unlink = *seconds*
wr enable compression = *yes* | *no*

Attributs facultatifs du poste de travail pour les fichiers de base de données distants

mozart directory = *mozart_share*
parameters directory = *parms_share*
unison network directory = *unison_share*

Attributs du poste de travail pour les formats personnalisés

date format = *entier*
composer prompt = *clé*
conman prompt = *clé*
switch sym prompt = *clé*

Attributs du poste de travail pour la personnalisation des E/S sur les fichiers de la boîte aux lettres

sync level = *low* | *medium* | *high*

Attributs du poste de travail pour la mise en réseau

tcp timeout = *seconds*
tcp connect timeout = *seconds*

Attributs du poste de travail pour SSL - Général

ssl auth mode = *caonly* | *string* | *cpu*
ssl auth string = *chaîne*
ssl fips enabled = *yes/no*
nm ssl full port = *value*
nm ssl port = *value*

Attributs OpenSSL du poste de travail - utilisés uniquement si *ssl fips enabled = "no"*

ssl key = **.pem*
ssl certificate = **.pem*
ssl key pwd = **.sth*
ssl ca certificate = **.crt*
ssl random seed = **.rnd*
ssl encryption cipher = *cipher*
cli ssl server auth = *yes* | *no*

cli ssl cipher = *chaîne*
cli ssl server certificate = *file_name*
cli ssl trusted dir = *directory_name*

Attributs GSKit du poste de travail - utilisés uniquement si *ssl fips enabled = "yes"*

ssl keystore file = **.kdb*
ssl certificate keystore label = *nom*
ssl keystore pwd = **.sth*
cli ssl keystore file = **.kdb*
cli ssl certificate keystore label = *nom*
cli ssl keystore pwd = **.sth*

Attributs du poste de travail pour WebSphere Application Server

local was = *yes | no*

Attributs de vérification du serveur d'applications sur le poste de travail

appserver check interval = *minutes*
appserver auto restart = *on | off*
appserver min restart time = *minutes*
appserver max restarts = *nombre*
appserver count reset interval = *heures*
appserver service name = *nom*

L'instance Tivoli Workload Scheduler est un client de ligne de commande

is remote cli = *yes | no*

Attributs de la connexion client de ligne de commande (conman)

host = *host_name*
protocol = *protocole*
port = *numéro de port*
proxy = *serveur proxy*
proxy port = *numéro de port du serveur proxy*
time out = *seconds*
default ws = *master_workstation*
useropts = *fichier_useropts*

Remarque : Les attributs SSL de la connexion du client de ligne de commande dépendent de la méthode SSL utilisée. Ils sont inclus dans la section correspondante et commencent tous par "cli".

Paramètres de gestion d'événements

can be event processor = *yes | no*

Remarque : La syntaxe du fichier `localopts` n'est pas sensible à la casse et les espaces entre les mots dans les noms d'option sont ignorés. Par exemple, pour le paramètre **is remote cli**, vous pouvez écrire :

- is remote cli
- Is Remote CLI
- isremotecli
- ISREMOTECLI
- isRemoteCLI
- ...

Détails de Localopts

 # commentaire
Traite tout ce qui suit le signe # jusqu'à la fin de la ligne comme du commentaire.

appserver auto restart = yes | no
Demande au processus `appservman` de démarrer automatiquement WebSphere Application Server s'il est arrêté. La valeur par défaut est `Yes`.

appserver check interval = *minutes*
Spécifie la fréquence en minutes à laquelle le processus `appservman` doit contrôler que WebSphere Application Server est toujours en cours d'exécution. La valeur par défaut est 5 minutes.

appserver count reset interval = *heures*
Spécifie l'intervalle de temps en heures à l'issue duquel le comptage de redémarrage est réinitialisé à partir du dernier démarrage de WebSphere Application Server. La valeur par défaut est 24 heures.

appserver max restarts = *nombre*
Spécifie le nombre maximal de tentatives de redémarrage que le processus `appservman` peut effectuer avant d'abandonner et de quitter sans redémarrer WebSphere Application Server. Le compteur est réinitialisé si la durée d'exécution de WebSphere Application Server est supérieure à la valeur de `appserver count reset interval`. La valeur par défaut est 5.

appserver min restart time = *minutes*
Spécifie, en minutes, la durée minimale durant laquelle le processus `appservman` doit attendre entre chaque tentative de redémarrage de WebSphere Application Server si ce dernier est arrêté. Si cette valeur est inférieure à `appserver check interval`, le WebSphere Application Server est redémarré dès que le système détecte son arrêt. Si le système détecte qu'il est arrêté avant l'expiration de ce délai (min restart time), `appservman` quitte sans le redémarrer. La valeur par défaut est 10 minutes.

appserver service name = *nom*
Uniquement dans les environnements Windows. Spécifie le nom du service Windows de WebSphere Application Server si celui-ci diffère du nom standard. Généralement, cette zone n'est pas renseignée.

autostart monman = yes | no
Utilisée dans la gestion des règles d'événement. Redémarre automatiquement le moteur de contrôle lors de l'activation du plan de production suivant (sous Windows, également lors du redémarrage de Tivoli Workload Scheduler). La valeur par défaut est `Yes`.

bm check deadline = *seconds*
Nombre minimum de secondes pendant lequel Batchman attend avant de vérifier si un travail n'a pas respecté l'échéance. Le contrôle est exécuté sur tous les travaux et flots de travaux inclus dans le fichier Symphony, quel que soit le poste de travail sur lequel les travaux et les flots de travaux sont définis. Les travaux et flots de travaux dont le délai a expiré sont marqués comme en retard dans le fichier Symphony local. Pour obtenir des informations à jour à propos de l'ensemble de l'environnement, définissez cette option sur le gestionnaire de domaine maître. Les délais d'expiration des travaux critiques sont évalués automatiquement, indépendamment de l'option **bm check deadline**. Pour désactiver l'option sans vérifier les échéances, entrez la valeur zéro (valeur par défaut).

bm check file = *seconds*
Nombre minimum de secondes pendant lequel Batchman attend avant de vérifier l'existence d'un fichier qui est utilisé en tant que dépendance. La valeur par défaut est 120 secondes.

bm check status = *seconds*
Nombre de secondes pendant lequel Batchman attend avant de vérifier l'état d'une dépendance interréseau. La valeur par défaut est 300.

bm check until = *seconds*
Spécifiez le nombre de secondes maximal durant lequel Batchman attend avant de signaler l'expiration d'un délai de réalisation de travail ou de Planificateur de travaux. L'indication d'une valeur inférieure à la valeur par défaut (300) risque d'entraîner une surcharge du système. Si cette option a une valeur inférieure à celle de l'option locale **bm read**, la valeur de **bm read** est utilisée à sa place. La valeur par défaut est 300.

bm look = *seconds*
Nombre minimum de secondes pendant lequel Batchman attend avant d'analyser et de mettre à jour son fichier de contrôle de production. Par défaut, la valeur est 15 secondes.

bm read = *seconds*
Nombre maximal de secondes pendant lesquelles Batchman attend l'arrivée d'un message dans le fichier de messages `intercom.msg`. Si aucun message n'est en file d'attente, Batchman attend jusqu'à l'expiration du délai ou jusqu'à ce qu'un message soit écrit dans le fichier. La valeur par défaut est 10 secondes.

bm stats = on | off
Pour que Batchman envoie ses statistique de démarrage et d'arrêt à son fichier de liste standard, spécifiez **on**. Pour empêcher les statistiques Batchman d'être envoyées à leur fichier de liste standard, spécifiez **off**. Par défaut, la valeur est **off**.

bm verbose = on | off
Pour que Batchman envoie tous les messages de statut des travaux à son fichier de liste standard, spécifiez **on**. Pour empêcher l'ensemble étendu de messages de statut de travaux d'être envoyé au fichier de liste standard, spécifiez **off**. Par défaut, la valeur est **off**.

bm late every = minutes
Lorsqu'un travail **every** ne démarre pas à l'heure de début attendue, **bm late every** spécifie le nombre maximum de minutes qui s'écouleront avant que Tivoli Workload Scheduler n'ignore le travail. Cette option s'applique uniquement aux travaux définis avec l'option **every** ainsi qu'avec la dépendance de temps **at**, elle n'entraîne aucun impact sur les travaux pour lesquels seule l'option **every** est définie.

can be event processor = yes | no
Indique si ce poste de travail peut agir en tant que serveur de traitement d'événement ou pas. La valeur par défaut est **yes** pour les gestionnaire de domaine maître et les serveurs maîtres de secours ; dans les autres cas, cette valeur est **no**.

cli ssl certificate keystore label = *chaîne*
Utilisé uniquement si SSL est défini avec GSKit (`ssl fips enabled="yes"`). Indiquez le label qui identifie le certificat dans le fichier de clés lorsque le client de ligne de commande utilise l'authentification SSL pour communiquer avec le gestionnaire de domaine maître. La valeur par défaut

est `IBM TWS 8.6 workstation`, qui est la valeur du certificat fournie avec le produit à tous les clients. Ce certificat n'est donc pas sécurisé et doit être remplacé par votre propre certificat sécurisé. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

cli ssl keystore file = *file_name*
Utilisé uniquement si SSL est défini avec GSKit (`ssl fips enabled="yes"`). Indiquez le nom du fichier de clés utilisé pour l'authentification SSL lorsque le client de ligne de commande utilise l'authentification SSL pour communiquer avec le gestionnaire de domaine maître. La valeur par défaut est *TWA_home*`/TWS/ssl/TWSPublicKeyFile.pem`. Ce fichier fait partie de la configuration SSL fournie avec le produit à tous les clients. Cette configuration n'est donc pas sécurisée et doit être remplacée par votre propre configuration SSL sécurisée. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

cli ssl keystore pwd = *file_name*
Utilisé uniquement si SSL est défini avec GSKit (`ssl fips enabled="yes"`). Indiquez le fichier de mot de passe du fichier de clés utilisé pour l'authentification SSL lorsque le client utilise l'authentification SSL pour communiquer avec le gestionnaire de domaine maître. Ce fichier fait partie de la configuration SSL fournie avec le produit à tous les clients. Cette configuration n'est donc pas sécurisée et doit être remplacée par votre propre configuration SSL sécurisée. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

cli ssl cipher = *cipher_class*
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`). Indiquez la classe de chiffrement à utiliser lorsque le client de ligne de commande et le serveur utilisent l'authentification SSL. Utilisez l'une des classes de chiffrement communes répertoriées dans le tableau 16. La valeur par défaut est MD5.

Pour utiliser une classe de chiffrement OpenSSL non répertoriée dans le tableau, utilisez la commande suivante pour déterminer si la classe dont vous avez besoin est prise en charge :

```
openssl ciphers <class_name>
```

où *nom_classe* est le nom de la classe que vous souhaitez utiliser. Si la commande renvoie une chaîne de chiffrement, la classe peut être utilisée.

Tableau 16. Classe de chiffrement valides

Classe de chiffrement	Description
SSLv3	SSL version 3.0
TLSv1	TLS version 1.0
EXP	Exportation
EXPORT40	Exportation 40 bits
MD5	Chiffrements utilisant l'historique MD5, la signature numérique, le chiffrement unidirectionnel, l'algorithme de hachage (hash) ou de contrôle (checksum).
LOW	Faible puissance (aucune exportation, DES simple)
MEDIUM	Codes de chiffrement avec chiffrage 128 bits

Tableau 16. Classe de chiffrement valides (suite)

Classe de chiffrement	Description
HIGH	Codes de chiffrement utilisant la norme DES triple
NULL	Codes de chiffrement n'utilisant aucun chiffrage

cli ssl server auth = yes|no
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`) Spécifiez **yes** si l'authentification de serveur doit être utilisée dans les communications SSL avec le client de ligne de commande. Par défaut, la valeur est **no**.

cli ssl server certificate = *file_name*
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`) Spécifiez le fichier, y compris son chemin de répertoire complet, qui contient le certificat SSL lorsque le client de ligne de commande et le serveur utilisent l'authentification SSL pour communiquer. Il n'y a pas de valeur par défaut. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

cli ssl trusted dir = *directory_name*
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`). Indiquez le répertoire contenant un certificat sécurisé SSL hébergé dans des fichiers avec la désignation de hachage (#) lorsque le client de ligne de commande et le serveur utilisent l'authentification SSL pour communiquer. Lorsque le chemin d'accès au répertoire contient des espaces vides, placez-le entre guillemets ("). Il n'y a pas de valeur par défaut.

composer prompt = *invite*
Spécifiez l'invite pour la ligne de commande de composer. L'invite peut contenir jusqu'à 10 caractères. Par défaut, la valeur est un tiret (**-**).

conman prompt = *invite*
Spécifiez l'invite pour la ligne de commande conman. L'invite peut contenir jusqu'à 8 caractères. Par défaut, la valeur est le signe pourcentage (**%**).

date format = 0|1|2|3
Indiquez la valeur qui correspond au format de date que vous souhaitez utiliser. Les valeurs peuvent être les suivantes :

- 0 correspond à *aa/mm/jj*
- 1 correspond à *mm/jj/aa*
- 2 correspond à *jj/mm/aa*
- 3 indique l'utilisation de variables d'environnement NLS (Native Language Support)

Par défaut, la valeur est **1**.

default ws = *manager_workstation*
Poste de travail par défaut lors de l'accès à l'aide d'un client de ligne de commande. Indiquez le poste de travail du gestionnaire de domaine.

host = *nom_d'hôte_ou_adresse_IP*
Le nom ou l'adresse IP de l'hôte lorsque vous accédez au système à l'aide d'un client de ligne de commande.

is remote cli = yes | no
Spécifiez si cette instance de Tivoli Workload Scheduler est installée en tant que client de ligne de commande (yes).

jm interactive old = yes | no
Uniquement pour les systèmes d'exploitation Windows à partir de Vista et versions ultérieures. Pour respecter les restrictions de sécurité introduites avec la version Vista des systèmes d'exploitation Windows, uniquement pour les agents tolérants aux pannes, Tivoli Workload Scheduler exécute des travaux interactifs uniquement si l'utilisateur `streamlogon` a une session interactive valide. Indiquez **yes** pour autoriser **jobman** à démarrer des travaux interactifs même si aucune session n'est active pour l'utilisateur `streamlogon`. Indiquez **no** pour autoriser **jobman** à démarrer des travaux interactifs uniquement si des sessions sont actives pour l'utilisateur `streamlogon`. Par défaut, la valeur est **no**.

jm job table size = *entries*
Indiquez la taille, en nombre d'entrées, de la table des travaux utilisée par Jobman. Par défaut, la valeur est 1024 entrées.

jm load user profile = on | off
Uniquement sur les systèmes d'exploitation Windows. Indiquez si le processus **jobman** charge le profil d'utilisateur et ses variables d'environnement pour l'utilisateur spécifié dans la zone de connexion de chaque travail, avant le lancement du travail sur le poste de travail. Indiquez **on** (activer) pour charger le profil d'utilisateur sur le poste de travail avant d'exécuter des travaux pour l'utilisateur de la connexion ; sinon spécifiez **off** (désactiver). Les profils itinérants ne sont pas pris en charge. La valeur par défaut est **on**.

jm look = *seconds*
Nombre minimum de secondes pendant lequel Jobman attend avant de rechercher les travaux terminés et d'exécuter des tâches de gestion générales des travaux. La valeur par défaut est 300.

jm nice = *valeur_nice*
Pour les systèmes d'exploitation UNIX et Linux uniquement, spécifiez la valeur **nice** à appliquer aux travaux lancés par Jobman pour modifier leur priorité dans le planificateur de kernel. La valeur par défaut est zéro.

Les valeurs limites de **nice** varient selon les plateformes, mais généralement, les valeurs inférieures correspondent aux niveaux de haute priorité et vice versa. La valeur par défaut dépend du système d'exploitation.

S'applique aux travaux planifiés par l'utilisateur root uniquement. Les travaux soumis par tout autre utilisateur héritent la même valeur **nice** que le processus jobman.

Voir aussi jm promoted nice.

jm no root = yes | no
Pour les systèmes d'exploitation UNIX et Linux uniquement, spécifiez **yes** pour empêcher Jobman de lancer les travaux **root**. Indiquez **no** pour permettre à Jobman de lancer des travaux **root**. Par défaut, la valeur est **no**.

jm promoted nice = *valeur_nice*
Utilisée dans l'assurance de service de charge de travail. Pour les systèmes d'exploitation UNIX et Linux uniquement, attribue la valeur prioritaire à un travail critique qui soit être promu afin que le système d'exploitation le

traite avant les autres. S'applique aux travaux critiques ou à leurs prédécesseurs qui doivent être promus afin qu'ils puissent commencer à l'heure critique locale.

Les valeurs limites varient en fonction des plateformes, mais généralement, les valeurs inférieures correspondent aux niveaux de haute priorité et vice versa. La valeur par défaut est -1.

Sachez que :

- Le processus de promotion n'est efficace qu'avec des valeurs négatives. Si vous définissez une valeur positive, le système l'exécute avec la valeur par défaut -1 et consigne un message d'avertissement à chaque démarrage de Jobman.
- Une valeur hors plage (par exemple -200) incite le système d'exploitation à promouvoir automatiquement les travaux dont la valeur **nice** attribuée est la plus basse. Notez que dans ce cas, aucun avertissement n'est consigné.
- L'utilisation abusive du mécanisme de promotion (c'est-à-dire la définition d'un nombre excessif de travaux comme critiques et la définition de la valeur de priorité la plus élevée ici) risque de surcharger le système d'exploitation, entraînant un impact négatif sur les performances générales du poste de travail.

Vous pouvez utiliser cette option et l'option jm nice ensemble. Dans ce cas, n'oubliez pas que même si **jm nice** s'applique uniquement aux travaux soumis par l'utilisateur racine, **jm promoted nice** s'applique uniquement aux travaux présentant une heure de début critique. Lorsqu'un travail répond à ces deux conditions, les valeurs définies pour les deux options s'ajoutent. Par exemple, si sur un agent particulier, le fichier d'options locales présente :

```
jm nice= -2
jm promoted nice= -4
```

lorsqu'un travail critique soumis par le superutilisateur doit être promu, le système lui attribue une valeur prioritaire cumulée de -6.

jm promoted priority = *value*

Utilisée dans l'assurance de service de charge de travail. Pour les systèmes d'exploitationWindows uniquement, cette valeur indique la priorité selon laquelle le système d'exploitation traite un travail critique lorsqu'il est promu.

S'applique aux travaux critiques ou à leurs prédécesseurs qui doivent être promus afin qu'ils puissent commencer à l'heure critique locale.

Les valeurs possibles sont :

- `High`
- `AboveNormal` (valeur par défaut)
- `Normal`
- `BelowNormal`
- `Low` ou `Idle`

Notez que si vous définissez une valeur de priorité inférieure à celle qui peut être attribuée aux travaux non critiques, aucun avertissement n'est envoyé et aucun mécanisme comme celui disponible pour **jm promoted nice** ne rétablit la valeur par défaut.

jm read = *seconds*
Nombre maximal de secondes pendant lesquelles Jobman attend l'arrivée d'un message dans le fichier de messages `courier.msg`. La valeur par défaut est 10 secondes.

local was = yes | no
Pour les gestionnaires de domaine maître et les systèmes maître de secours connectés à la base de données Tivoli Workload Scheduler. Si elle est définie sur **yes**, cette option améliore les performances de Planificateur de travaux et la soumission de travaux de la base de données. La valeur par défaut est **no**.

merge stdlists = yes | no
Indiquez **yes** pour que tous les processus de contrôle Tivoli Workload Scheduler, à l'exception de Netman, envoient leurs messages écran à un seul fichier de liste standard. Ce fichier est intitulé **TWSmerge**. Indiquez **no** pour que les processus envoient les messages à des fichiers de liste standard séparés. Par défaut, la valeur est **yes**.

mm cache mailbox = yes | no
Utilisez cette option pour permettre à Mailman d'utiliser une mémoire cache en lecture pour les messages entrants. Dans ce cas, seuls les messages considérés comme essentiels à la cohérence du réseau sont mis en mémoire cache. Par défaut, la valeur est **yes**.

mm cache size = *messages*
Définissez cette option si vous utilisez également **mm cache mailbox**. La valeur maximale (par défaut) est **512**.

mm planoffset = *HHMM*
HHMM est une période exprimée en heures et minutes. Au démarrage de Tivoli Workload Scheduler, ce temps est utilisé comme un décalage pour vérifier la validité du plan Symphony en fonction de la formule suivante :

```
horodatage_en_cours < (Symphony_end_timestamp - HHMM)
```

Si le résultat est true, en d'autres termes, si l'heure courante est antérieure à l'heure de fin planifiée de Symphony moins le décalage, le plan Symphony est considéré comme valide et Tivoli Workload Scheduler démarre. Si le résultat est false, Tivoli Workload Scheduler ne démarre pas et une erreur est consignée. La valeur par défaut de cet attribut facultatif est une valeur vide ; dans ce cas, Tivoli Workload Scheduler n'effectue aucun contrôle sur la validité du plan. Cette vérification est parfois nécessaire lorsqu'un gestionnaire de domaine s'arrête suite à une indisponibilité non planifiée et redémarre ultérieurement, lorsqu'un nouveau gestionnaire de domaine a été démarré entre-temps, car toutes les procédures de reprise appropriées n'ont pas été exécutées pour l'exclure du réseau Tivoli Workload Scheduler. Par conséquent, deux gestionnaires de domaine s'exécutent simultanément sur le même agent tolérant aux pannes. Cela crée des problèmes de planification sur tous les agents tolérants aux pannes.

mm read = *seconds*
Indiquez le nombre maximal de secondes d'attente de Mailman pour établir une connexion avec un poste de travail distant. Par défaut, la valeur est 15 secondes.

mm resolve master = yes | no
Si cette option est associée à la valeur **yes**, la variable $MASTER est résolue au début de la journée de production. L'hôte de tout agent étendu

est commuté après le **JnextPlan** (commutateur à long terme) suivant. Lorsque cette option est définie sur **no**, la variable $MASTER n'est pas résolue à **JnextPlan** et l'hôte de tout agent étendu peut être commuté après une commande **switchmgr** (commutateur à court et à long terme). Par défaut, la valeur est **yes**. Lorsque vous commutez un gestionnaire de domaine maître et que dans l'original, la valeur de **mm resolve master** est définie sur no alors que sur le serveur de secours, **mm resolve master** est défini sur yes, après la commutation, tout agent étendu hébergé par $MASTER est commuté vers le gestionnaire de domaine. Au redémarrage du gestionnaire de domaine de secours, le mot clé $MASTER est étendu localement par Mailman. La valeur de **mm resolve master** doit rester la même pour les gestionnaires de domaine maître et les gestionnaires de domaines de secours.

mm response = *seconds*
Nombre maximum de secondes pendant lesquelles Mailman attend une réponse avant de signaler qu'un poste de travail ne répond pas. Le délai d'attente minimal d'une réponse est **90** secondes. Par défaut, la valeur est 600 secondes.

mm retrylink = *seconds*
Nombre maximum de secondes pendant lesquelles Mailman attend, après avoir supprimé les liaisons avec un poste de travail ne répondant pas, avant d'essayer à nouveau de créer un lien avec ce poste de travail. Par défaut, la valeur est 600 secondes. Les serveurs mailman **tomserver** facultatifs ne suppriment pas les liaisons vers les agents qui ne répondent pas. La liaison est régulièrement vérifiée toutes les 60 secondes, ce qui correspond à la valeur par défaut de l'option **retrylink** pour ces serveurs.

mm sound off = yes|no
Spécifiez la façon dont Mailman répond à une commande conman **tellop ?** . Indiquez **yes** pour que Mailman affiche des informations concernant chacune des tâches qu'il exécute. Indiquez **no** pour que Mailman envoie uniquement son propre état. Par défaut, la valeur est **no**.

mm symphony download timeout = *seconds*
Indiquez le nombre maximal de minutes d'attente de Mailman après une tentative d'initialisation d'un poste de travail sur un réseau lent. Si le délai expire avant l'aboutissement de l'initialisation du poste de travail, Mailman initialise le poste de travail suivant dans la séquence. Par défaut, il n'y a aucun délai d'attente (0).

mm unlink = *seconds*
Indiquez le nombre maximal de secondes pendant lesquelles Mailman doit attendre avant de supprimer les liaisons avec un poste de travail qui ne répond pas. Le délai d'attente ne doit pas être inférieur au temps de réponse indiqué pour l'option locale **nm response**. La valeur par défaut est 960 secondes.

mozart directory = *directory_name*
Ce paramètre s'applique uniquement aux versions de Tivoli Workload Scheduler antérieures à la version 8.3. Définit le nom du répertoire mozart gestionnaire de domaine maître partagé. La valeur par défaut est *TWA_home*/mozart.

nm mortal = yes|no
Indiquez **yes** pour que Netman se ferme lorsque tous ses processus enfants

se sont arrêtés. Indiquez **no** pour que Netman continue à s'exécuter même lorsque tous ses processus enfants se sont arrêtés. Par défaut, la valeur est **no**.

nm port = *port*
Indiquez le numéro du port TCP auquel Netman répond sur l'ordinateur local. Cette valeur doit correspondre au port TCP/UIP indiqué dans la définition de poste de travail de cet ordinateur. Il doit s'agir d'une valeur 16 bits non signée, comprise entre 1 et 65535. (Les valeurs comprises entre 0 et 1023 sont réservées à des services, tels que FTP, TELNET, HTTP, etc.) La valeur par défaut correspond à la valeur fournie lors de l'installation du produit.

Si vous exécutez la fonction d'automatisation de charge gérée par les événements et que vous disposez d'un pare-feu de sécurité, vérifiez que ce port est ouvert pour les connexions entrantes et sortantes.

nm read = *seconds*
Indiquez le nombre maximal de secondes pendant lesquelles Netman doit attendre une demande de connexion avant de vérifier la présence des commandes **stop** et **start** dans sa file d'attente de messages. La valeur par défaut est 10 secondes.

nm retry = *seconds*
Indiquez le nombre maximal de secondes pendant lesquelles Netman doit attendre avant d'essayer une nouvelle fois d'établir une connexion qui a échoué. Par défaut, la valeur est 800 secondes.

nm ssl full port = *port*
Port utilisé pour écouter les connexions SSL entrantes lorsque l'option full SSL est configurée en attribuant la valeur `yes` à l'option globale `enSSLFullConnection` (pour plus de détails, voir «Configuration de la sécurité SSL totale», à la page 292). Cette valeur doit correspondre à celle définie dans l'attribut **secureaddr** de la définition de poste de travail dans la base de données. Elle doit être différente de l'option locale **nm port** qui définit le port utilisé pour des communication normales.

Remarque :
1. Si vous installez plusieurs instances du produit Tivoli Workload Scheduler sur le même ordinateur, définissez tous les ports SSL sur plusieurs valeurs.
2. Si vous envisagez de ne pas utiliser SSL, définissez la valeur sur 0.

Il n'y a pas de valeur par défaut.

nm ssl port = *port*
Port utilisé pour écouter les connexions SSL entrantes, lorsque l'option full SSL n'est pas configurée (pour plus de détails, voir «Configuration de la sécurité SSL totale», à la page 292). Cette valeur doit correspondre à celle définie dans l'attribut **secureaddr** de la définition de poste de travail dans la base de données. Elle doit être différente de l'option locale **nm port** qui définit le port utilisé pour des communication normales.

Remarque :
1. Si vous installez plusieurs instances du produit Tivoli Workload Scheduler sur le même ordinateur, définissez tous les ports SSL sur plusieurs valeurs.
2. Si vous envisagez de ne pas utiliser SSL, définissez la valeur sur 0.

Il n'y a pas de valeur par défaut.

parameters directory = *directory_name*
Ce paramètre s'applique uniquement aux versions de Tivoli Workload Scheduler antérieures à la version 8.3. Définit le nom du répertoire *TWA_home* partagé des gestionnaires de domaine maître. La valeur par défaut est none.

port = *port*
Le numéro de port TCP/IP du protocole utilisé lors de l'accès à l'aide d'un client de ligne de commande. Par défaut, la valeur est 31115.

protocol = http | https
Le protocole utilisé pour se connecter à l'hôte lors de l'utilisation d'un client de ligne de commande.

proxy = *nom_d'hôte_ou_adresse_IP_serveur_proxy*
Le nom du serveur proxy utilisé lors de l'accès à l'aide d'un client de ligne de commande.

proxy port = *port_serveur_proxy*
Le numéro du port TCP/IP du serveur proxy utilisé pour l'accès à l'aide d'un client de ligne de commande.

restricted stdlists = yes | no
Utilisez cette option pour définir un niveau de sécurité plus élevé pour le répertoire `stdlist` (et ses sous-répertoires) en n'autorisant que les utilisateurs sélectionnés à créer, modifier et lire les fichiers.

Cette option est disponible uniquement sur les postes de travail UNIX. Après l'avoir définie, assurez-vous de supprimer votre répertoire actif `stdlist` (et ses sous-répertoires), puis de redémarrer Tivoli Workload Scheduler. Par défaut, la valeur est `no`.

Si cette option est absente ou si elle a la valeur `no`, le répertoire `stdlist` qui vient d'être créé, ainsi que ses sous-répertoires ne sont pas affectés et leurs droits d'accès sont définis comme suit :

```
drwxrwxr-x  22 twsmdm staff          4096 Nov 09 12:12
drwxrwxr-x   2 twsmdm staff           256 Nov 09 11:40 2009.11.09
drwxrwxr-x   2 twsmdm staff          4096 Nov 09 11:40 logs
drwxr-xr-x   2 twsmdm staff          4096 Nov 09 11:40 traces
```

Si cette option a la valeur `yes`, ces répertoires ont les droits d'accès suivants :

```
drwxr-x--x  5 twsmdm staff             256 Nov 13 18:15
rwxr-x--x   2 twsmdm staff             256 Nov 13 18:15 2009.11.13
rwxr-x--x   2 twsmdm staff             256 Nov 13 18:15 logs
rwxr-x--x   2 twsmdm staff             256 Nov 13 18:15 traces
```

Pour définir et activer cette option, procédez comme suit :

1. Remplacez la ligne `restricted stdlists = no` par `restricted stdlists = yes` dans votre fichier d'options locales.
2. Arrêtez Tivoli Workload Scheduler.
3. Arrêtez WebSphere Application Server, le cas échéant.
4. Supprimez le répertoire `stdlist` (ou du moins ses fichiers et sous-répertoires).
5. Démarrez Tivoli Workload Scheduler.
6. Démarrez WebSphere Application Server, le cas échéant.

ssl auth mode = caonly | string | cpu
Le comportement de Tivoli Workload Scheduler pendant l'établissement d'une liaison SSL est basé sur la valeur de l'option SSL auth mode, comme suit :

caonly Tivoli Workload Scheduler contrôle la validité du certificat et vérifie que le certificat homologue a été délivré par une autorité de certification reconnue. Les informations contenues dans le certificat ne sont pas examinées. Valeur par défaut.

chaîne Tivoli Workload Scheduler contrôle la validité du certificat et vérifie que le certificat homologue a été délivré par une autorité de certification reconnue. Il vérifie également que le nom CN de l'objet du certificat correspond à la chaîne indiquée dans l'option SSL auth string. Voir «ssl auth string = chaîne».

cpu Tivoli Workload Scheduler contrôle la validité du certificat et vérifie que le certificat homologue a été délivré par une autorité de certification reconnue. Il vérifie également que le nom CN de l'objet du certificat correspond au nom du poste de travail qui a demandé le service.

ssl auth string = *chaîne*
Utilisé avec l'option **SSL auth mode** lorsque la valeur "string" est indiquée. La valeur **SSL auth string** (de 1 à 64 caractères) permet de vérifier la validité du certificat. La chaîne par défaut est "tws".

ssl ca certificate = *file_name*
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`). Indiquez le nom du fichier contenant les certificats des autorités de certification (CA - Certification Authority) sécurisés, obligatoires pour l'authentification SSL. Les CA figurant dans ce fichier sont également utilisées pour créer la liste des CA client acceptables communiquées au client lorsque le côté serveur de la connexion demande un certificat de client. Ce fichier est la concaténation, par ordre de préférence, des différents fichiers de certificat d'autorité de certification codés selon la norme PEM.

La valeur par défaut est *TWA_home*`/TWS/ssl/TWSTrustedCA.crt`. Ce fichier fait partie de la configuration SSL fournie avec le produit à tous les clients. Cette configuration n'est donc pas sécurisée et doit être remplacée par votre propre configuration SSL sécurisée. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

ssl certificate = *file_name*
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`). Indiquez le nom du fichier de certificat local utilisé dans la communication SSL.

La valeur par défaut est *TWA_home*`/TWS/ssl/TWSPublicKeyFile.pem`. Ce fichier fait partie de la configuration SSL fournie avec le produit à tous les clients. Cette configuration n'est donc pas sécurisée et doit être remplacée par votre propre configuration SSL sécurisée. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

ssl certificate keystore label = *chaîne*
Utilisé uniquement si SSL est défini avec GSKit (`ssl fips enabled="yes"`). Indiquez le label qui identifie le certification dans le fichier de clés lors de l'utilisation de l'authentification SSL.

La valeur par défaut est `IBM TWS 8.6 workstation`, qui est la valeur du certificat fournie avec le produit à tous les clients. Ce certificat n'est donc pas sécurisé et doit être remplacé par votre propre certificat sécurisé. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

ssl encryption cipher = *cipher_class*
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`). Définissez les codes de chiffrement pris en charge par le poste de travail lors d'une connexion SSL.

Utilisez l'une des classes de chiffrement communes répertoriées dans le tableau 16, à la page 36. La valeur par défaut est **SSLv3**. Pour utiliser une classe de chiffrement OpenSSL non répertoriée dans le tableau, utilisez la commande suivante pour déterminer si la classe dont vous avez besoin est prise en charge :

```
openssl ciphers <class_name>
```

où *nom_classe* est le nom de la classe que vous souhaitez utiliser. Si la commande renvoie une chaîne de chiffrement, la classe peut être utilisée.

ssl fips enabled = yes|no
Détermine si l'ensemble de votre réseau Tivoli Workload Scheduler est activé conformément aux normes FIPS (Federal Information Processing Standards). La conformité aux normes FIPS nécessite l'utilisation de GSKit au lieu de l'option par défaut OpenSSL pour les communications sécurisées. Si vous activez FIPS (*ssl fips enabled="yes"*), vous devez définir les valeurs de tous les attributs SSL applicables à GSKit. Si vous n'activez pas FIPS (*ssl fips enabled="no"*), définissez les valeurs d'OpenSSL. La valeur par défaut est **no**. Pour plus de détails, voir «Conformité aux normes FIPS», à la page 300.

ssl key = *file_name*
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`). Nom du fichier de clés privées.

La valeur par défaut est *TWA_home*`/TWS/ssl/TWSPrivateKeyFile.pem`. Ce fichier fait partie de la configuration SSL fournie avec le produit à tous les clients. Cette configuration n'est donc pas sécurisée et doit être remplacée par votre propre configuration SSL sécurisée. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

ssl key pwd = *file_name*
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`). Nom du fichier contenant le mot de passe pour la clé dissimulée.

La valeur par défaut est *TWA_home*`/TWS/ssl/TWSPrivateKeyFile.sth`. Ce fichier fait partie de la configuration SSL fournie avec le produit à tous les clients. Cette configuration n'est donc pas sécurisée et doit être remplacée par votre propre configuration SSL sécurisée. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

ssl keystore file = *file_name*
Utilisé uniquement si SSL est défini avec GSKit (`ssl fips enabled="yes"`). Indiquez le nom du fichier de clés utilisé pour l'authentification SSL.

La valeur par défaut est *TWA_home*`/TWS/ssl/TWSKeyRing.kdb`. Ce fichier fait partie de la configuration SSL fournie avec le produit à tous les clients. Cette configuration n'est donc pas sécurisée et doit être remplacée par

votre propre configuration SSL sécurisée. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

ssl keystore pwd = *file_name*
Utilisé uniquement si SSL est défini avec GSKit (`ssl fips enabled="yes"`). Indiquez le nom du fichier de mot de passe de clés utilisé pour l'authentification SSL.

La valeur par défaut est *TWA_home*`/TWS/ssl/TWSKeyRing.sth`. Ce fichier fait partie de la configuration SSL fournie avec le produit à tous les clients. Cette configuration n'est donc pas sécurisée et doit être remplacée par votre propre configuration SSL sécurisée. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

ssl random seed = *file_name*
Utilisé uniquement si SSL est défini avec OpenSSL (`ssl fips enabled="no"`). Indiquez le fichier de nombre pseudo-aléatoire utilisé par OpenSSL sur certains systèmes d'exploitation. Sans ce fichier, l'authentification SSL risque de ne pas fonctionner correctement.

La valeur par défaut est *TWA_home*`/TWS/ssl/TWS.rnd`. Ce fichier fait partie de la configuration SSL fournie avec le produit à tous les clients. Cette configuration n'est donc pas sécurisée et doit être remplacée par votre propre configuration SSL sécurisée. Voir «Configuration du protocole de connexion SSL pour le réseau», à la page 292.

stdlist width = *colonnes*
Spécifiez la largeur maximale des messages de la console Tivoli Workload Scheduler. Vous pouvez spécifier un nombre de colonnes dans la plage **1** à **255**. Le retour à la ligne est effectué au niveau de la colonne spécifiée ou avant, selon la présence de caractères de contrôle chariot incorporés. Indiquez un nombre négatif ou nul pour ignorer l'épaisseur de ligne. Sur les systèmes d'exploitation UNIX et Linux, vous devriez ignorer la largeur de ligne si vous activez la connexion au système avec l'option **syslog local**. La valeur par défaut est 0 colonne.

switch sym prompt = *invite*
Indiquez une invite pour la ligne de commande conman après avoir sélectionné un fichier Symphony différent via la commande **setsym**. La longueur est limitée à 8 caractères. La valeur par défaut est **n%**.

sync level = low | medium | high
Débit auquel Tivoli Workload Scheduler synchronise les informations écrites sur le disque. Cette option affecte tous les agents de boîte aux lettres et s'applique uniquement aux systèmes d'exploitation UNIX et Linux. Les valeurs peuvent être :

low Permet au système d'exploitation de gérer ce paramètre.

medium
Vide les mises à jour dans le disque dès qu'une transaction est terminée.

high Vide les mises à jour dans le disque chaque fois que des données sont entrées.

La valeur par défaut est **low**.

syslog local = *value*
Permet d'activer ou de désactiver la connexion système Tivoli Workload Scheduler pour les systèmes d'exploitation UNIX et Linux uniquement.

Indiquez **-1** pour désactiver la journalisation système pour Tivoli Workload Scheduler. Indiquez un nombre compris entre **0** et **7** pour activer la journalisation système et pour que Tivoli Workload Scheduler utilise l'utilitaire local correspondant (LOCAL0 à LOCAL7) pour ses messages. Indiquez tout autre nombre pour activer la journalisation système et pour que Tivoli Workload Scheduler utilise l'utilitaire USER pour ses messages. La valeur par défaut est -1. Voir «Invites et messages d'écran de Tivoli Workload Scheduler», à la page 94.

tcp connect timeout = *seconds*
Spécifiez le nombre maximum de secondes d'attente avant d'établir une connexion via un socket non bloquant. Par défaut, la valeur est 15 secondes.

tcp timeout = *seconds*
Spécifiez le nombre maximum de secondes d'attente d'achèvement d'une requête sur un poste de travail connecté qui ne répond pas. La valeur par défaut est 300.

this cpu = *workstation_name*
Indiquez le nom Tivoli Workload Scheduler de ce poste de travail. Lors du basculement entre le gestionnaire de domaine maître et un gestionnaire de domaine de secours, à l'aide de la commande **switchmgr**, la valeur d'en-tête Symphony pour l'option **this cpu** est remplacée par la valeur **this cpu** du fichier `localopts`. La valeur par défaut est **$(this_cpu)**.

timeout = *seconds*
Le délai en secondes lors de l'accès à l'aide d'un client de ligne de commande. La valeur par défaut est 3600 secondes.

unison network directory = *directory_name*
Ce paramètre s'applique uniquement aux versions de Tivoli Workload Scheduler antérieures à la version 8.3. Définit le nom du répertoire partagé Unison. La valeur par défaut est `<TWA_home>/../unison/network`.

useropts = *file_name*
Si un système est doté de plusieurs instances de Tivoli Workload Scheduler, utilisez cette option pour identifier le fichier *useropts* à utiliser pour stocker les paramètres de connexion de l'instance dans laquelle se trouve ce fichier *localops*. Pour plus d'informations, voir «Multiples instances du produit», à la page 52.

wr enable compression = yes|no
Utilisez cette option sur les agents tolérants aux pannes. Spécifiez si l'agent tolérant aux pannes peut recevoir le fichier Symphony sous forme compressée de la part du gestionnaire de domaine maître. Par défaut, la valeur est **no**.

wr read = *seconds*
Nombre de secondes pendant lequel le processus Writer attend la réception d'un message entrant avant de vérifier l'existence d'une demande d'arrêt émise par Netman Par défaut, la valeur est 600 secondes.

wr unlink = *seconds*
Nombre de secondes pendant lequel le processus Writer attend avant de s'arrêter si aucun message entrant n'est reçu. La valeur minimale est 120 secondes. Par défaut, la valeur est 180 secondes.

Exemple de fichier des options locales

Voici un exemple de fichier par défaut `localopts` :

Remarque : Certains paramètres peuvent apparaître ou non, selon votre version et votre configuration.

```
################################################################################
# Eléments sous licence - Propriété d'IBM(R)
# 5698-WSH
# (C) Copyright IBM Corp. 1991, 2013.   All Rights Reserved
# US Government Users Restricted Rights - Use, duplication, or disclosure
# restricted by GSA ADP Schedule Contract with IBM Corp.
# IBM est une marque enregistrée d'IBM International Business Machines Corporation
# aux Etats-Unis et/ou dans certains autres pays.
################################################################################
#
# Le fichier localopts de Tivoli Workload Scheduler définit les attributs de
# ce poste de travail, pour divers processus.
#
#-------------------------------------------------------------------------------
# Attributs généraux de ce poste de travail :
#
thiscpu            =$(this_cpu)
merge stdlists     =yes
stdlist width    =0
syslog local = -1
restricted stdlists  =no
#
#-------------------------------------------------------------------------------
# Attributs de ce poste de travail pour le processus batchman :
#
bm check file      =120
bm check status=300
bm look = 15
bm read = 10
bm stats      =off
bm verbose      =off
bm check until      =300
bm check deadline    =0
bm late every        =0
#
#-------------------------------------------------------------------------------
# Attributs de ce poste de travail pour le processus jobman :
#
jm job table size = 1024
jm look      =300
jm nice      =0
jm promoted nice   =-1                #UNIX
jm promoted priority =AboveNormal     #WINDOWS
jm no root      =no
jm read      =10
jm load user profile =on
#
#-------------------------------------------------------------------------------
# Attributs de ce poste de travail pour le processus TWS mailman :
#
mm response     =600
mm retrylink     =600
mm sound off      =no
mm unlink      =960
mm cache mailbox   =yes
mm cache size      =512
mm resolve master = yes
autostart monman   =yes
mm symphony download timeout =0
#
#-------------------------------------------------------------------------------
# Attributs de ce poste de travail pour le processus netman :
#
nm mortal      =no
```

```
nm port            =$(tcp_port)
nm read = 10
nm retry      =800
#
#----------------------------------------------------------------------------
# Attributs de ce poste de travail pour le processus writer :
#
wr read                                  =600
wr unlink              =180
wr enable compression                    =no
#
#----------------------------------------------------------------------------
# Attributs facultatifs de ce poste de travail pour les fichiers de base de données distante
#
# mozart directory =         $(install_dir)/mozart
# parameters directory =     $(install_dir)
# unison network directory = $(install_dir)/../unison/network
#
#----------------------------------------------------------------------------
# Attributs de ce poste de travail pour les formats personnalisés :
#
date format             =1 # Les valeurs possibles sont 0-aaaa/mm/jj,
                           # 1-mm/jj/aaaa, 2-jj/mm/aaaa, 3-NLS.
composer prompt =           -
conman prompt =             %
switch sym prompt       =<n>%
#
#----------------------------------------------------------------------------
# Attributs de ce poste de travail pour la personnalisation des E/S sur
# les fichiers de la boîte aux lettres
#
sync level    =low
#
#----------------------------------------------------------------------------
# Attributs de ce poste de travail pour la mise en réseau
#
tcp timeout    =300
tcp connect timeout  =15
#
#----------------------------------------------------------------------------
# Options sécurisées générales
#
SSL auth mode = caonly
#
# Utilisez "SSL auth string" uniquement si "SSL auth mode" a la valeur "string"
#
SSL auth string = tws
#
# La valeur "yes" pour "SSL Fips enabled" force TWS à utiliser GSKIT,
# autrement TWS utilise OpenSSL
# Définie avec la valeur "yes", cette option active les règles FIPS 140-2.
# La valeur par défaut est "no".
#
SSL Fips enabled       = no
#
# Port SSL complet Netman, utilisez "nm SSL full port" si "enSSLFullConnection"
# (Option globale) a la valeur "yes". La valeur "0" signifie que le port est fermé.
#
nm SSL full port        =0
#
# Netman SSL port
# La valeur "0" signifie que le port est fermé
#
nm SSL port = 0
#
# Fin des options générales sécurisées
#----------------------------------------------------------------------------
```

```
#-------------------------------------------------------------------------
# Options OpenSSL, TWS les utilise si l'option "SSL Fips enabled" a la valeur "no"
#(valeur par défaut)
SSL key                 ="$(install_dir)/ssl/TWSPrivateKeyFile.pem"
SSL certificate         ="$(install_dir)/ssl/TWSPublicKeyFile.pem"
SSL key pwd             ="$(install_dir)/ssl/TWSPrivateKeyFile.sth"
SSL CA certificate      ="$(install_dir)/ssl/TWSTrustedCA.crt"
SSL random seed         ="$(install_dir)/ssl/TWS.rnd"
SSL Encryption Cipher   =SSLv3
#
#CLI SSL server auth =
#CLI SSL cipher     = MD5
#CLI SSL server certificate =
#CLI SSL trusted dir =
# Fin des options OpenSSL
#-------------------------------------------------------------------------

#-------------------------------------------------------------------------
# Options GSKIT, TWS les utilise si l'option "SSL Fips enabled" a la valeur "yes"
##
SSL keystore file                    = "$(install_dir)/ssl/TWSKeyRing.kdb"
SSL certificate keystore label       = "IBM TWS 9.2 - poste de travail"
SSL keystore pwd                     = "$(install_dir)/ssl/TWSKeyRing.sth"
#
#
CLI SSL keystore file                    = "$(install_dir)/ssl/TWSKeyRing.kdb"
CLI SSL certificate keystore label       = "IBM TWS 9.2 - poste de travail"
CLI SSL keystore pwd                     = "$(install_dir)/ssl/TWSKeyRing.sth"
#---------------------- Fin des options GSKit ---------------------------------


#-------------------------------------------------------------------------

# L'instance TWS a été installée en tant qu'interface REMOTE CLI
IS REMOTE CLI = no  # yes pour une installation REMOTE CLI, sinon no

#-------------------------------------------------------------------------
# Attributs pour les connexions à l'interface de ligne de commande
#
# Attributs généraux pour les connexions à l'interface de ligne de commande
#
HOST              = 127.0.0.1    # Nom d'hôte maître utilisé lors des tentatives de connexion
PROTOCOL    = https        # Protocole utilisé pour établir une connexion avec le maître.
PORT              = 31116        # Port du protocole
#PROXY      =
#PROXYPORT  =
TIMEOUT           = 3600  # Délai d'attente (en secondes) d'une réponse en provenance du serveur.
#CLI SSL SERVER AUTH = yes

DEFAULTWS         = NC125183
USEROPTS          = useropts_twsuser

#-------------------------------------------------------------------------
# Paramètres de gestion des événements
#
CAN BE EVENT PROCESSOR = yes # yes for MDM and BKM, no otherwise

#-------------------------------------------------------------------------
# Attributs de ce poste de travail pour la version intégrée
# d'IBM WebSphere Application Server
#
LOCAL WAS               =no
#
#-------------------------------------------------------------------------
# Attributs de vérification du serveur d'applications
```

```
Appserver profile path = "$(TWA_home)/WAS/TWSProfile"
Appserver profile name = "TWSProfile"
Appserver cell path = "$(TWA_home)/WAS/TWSProfile/installedApps/TWSNodeCell"
Appserver cell name = "TWSNodeCell"
Appserver node name = "TWSNode"
Appserver server name = "server1"
Appserver installation dir = "/opt/IBM/WebSphere/AppServer"
Appserver check interval        = 5     #minutes
Appserver auto restart          = yes    #yes/no
Appserver min restart time      = 10    #minutes
Appserver max restarts          =  5    #nombre de redémarrages
Appserver count reset interval  = 24   #heures
#Nom du service Appserver          = "IBMWAS85Service - tws920xx"
#-------------------------------------------------------------------------
```

Remarque : "REMOTE CLI" fait référence au client de ligne de commande.

Définition des options d'utilisateur

Définit les options dont vous avez besoin pour chaque utilisateur sur un poste de travail qui doit les utiliser dans le fichier `useropts`. Les modifications sont appliquées uniquement après l'arrêt et le redémarrage de Tivoli Workload Scheduler.

Le concept du fichier `useropts` est de contenir des valeurs pour les paramètres `localopts` qui doivent être personnalisés pour un utilisateur individuel. Les fichiers doivent se trouver dans le répertoire *user_home*`/.TWS` de l'utilisateur. Lorsque Tivoli Workload Scheduler a besoin d'accéder à des données à partir du fichier `localopts`, il cherche d'abord si la propriété dont il a besoin est uniquement ou également stockée dans le fichier `useropts` de l'utilisateur, privilégiant toujours la version du fichier `useropts` de la même valeur que la clé. Si une propriété n'est pas spécifiée lors de l'appel de la commande qui en a besoin, ou dans les fichiers obligatoire`useropts` et `localopts`, un message d'erreur apparaît.

La fonction principale du fichier `useropts` est de stocker les paramètres de connexion spécifiques à l'utilisateur utilisés pour accéder au client de ligne de commande (voir «Configuration de l'authentification pour l'accès au client de ligne de commande», à la page 91). Il s'agit des clés suivantes, qui ne sont pas stockées dans le fichier `localopts` :

username
Nom d'utilisateur utilisé pour accéder au gestionnaire de domaine maître. L'utilisateur doit être défini dans le fichier de sécurité du gestionnaire de domaine maître (voir Chapitre 4, «Configuration de l'autorisation des utilisateurs (fichier de sécurité)», à la page 155)

password
Mot de passe utilisé pour accéder au gestionnaire de domaine maître. La présence de l'étiquette `ENCRYPT` dans la zone du mot de passe indique que le paramètre spécifié a été chiffré ; si cette étiquette n'est pas présente, vous devez quitter le programme de l'interface et y accéder à nouveau pour permettre le chiffrement de cette zone.

Un fichier `useropts` est créé pour le <*utilisateur_TWS*> au cours de l'installation, mais vous devez créer un fichier distinct pour chaque utilisateur qui a besoin d'utiliser des paramètres spécifiques à l'utilisateur sur un poste de travail.

Exemple de fichier useropts

Voici un extrait du contenu d'un fichier `useropts` :

```
#
# Le fichier useropts Tivoli Workload Scheduler définit les attributs de ce poste de travail.
#
#-------------------------------------------------------------------------
# Attributs pour les connexions à l'interface de ligne de commande
USERNAME    = MDMDBE4       # Nom d'utilisateur utilisé lors de la connexion
PASSWORD    = "ENCRYPT:YEE7cEZs+HE+mEHCsdNOfg=="      # Mot de passe utilisé lors de la connexion
#HOST       =               # Nom d'hôte principal utilisé lors de la tentative de connexion.
PROTOCOL    = https         # Protocole utilisé pour établir une connexion avec le maître.
#PROTOCOL   = http          # Protocole utilisé pour établir une connexion avec le maître.
PORT        = 3111              # Port de protocole
#PROXY      =
#PROXYPORT  =
TIMEOUT     = 120           # Délai d'attente d'une réponse du serveur, en secondes
#DEFAULTWS  =

CLI SSL keystore file                    = "$(install_dir)/ssl/MyTWSKeyRing.kdb"
CLI SSL certificate keystore label       = "client"
CLI SSL keystore pwd                     = "$(install_dir)/ssl/MyTWSKeyRing.sth"
```

Les options de configuration SSL du client de ligne de commande dépendent du type de SSL implémenté - ici GSKit est utilisé.

Remarque : Le symbole # est utilisé pour commenter une ligne.

Multiples instances du produit

Dans la mesure où Tivoli Workload Scheduler prend en charge plusieurs instances du produit installées sur le même ordinateur, il peut exister plusieurs instances du fichier `useropts` par utilisateur. On obtient ce résultat en donnant aux fichiers `useropts` destinés à un utilisateur des noms différents pour chaque instance.

Dans le fichier `localopts` de chaque instance, l'option nommée ***useropts*** identifie le nom du fichier `useropts` qui doit être utilisé dans le répertoire *user_home*`/.TWS` pour se connecter à cette instance de l'installation.

Cela signifie, par exemple que si deux instances de Tivoli Workload Scheduler sont installées sur une ordinateur et que l'utilisateur `operator` est un utilisateur des deux instances, vous pouvez définir les données d'identification `useropts` comme suit :

- Dans le fichier `localopts` de la *première* instance, l'option locale *useropts* = `useropts1` identifie le fichier *operator_home*`/.TWS/useropts1` contenant les paramètres de connexion dont l'utilisateur `operator` a besoin pour se connecter à la *première* instance Tivoli Workload Scheduler.
- Dans le fichier `localopts` de la *seconde* instance Tivoli Workload Scheduler, l'option locale *useropts* = `useropts2` identifie le fichier *operator_home*`/.TWS/useropts2` contenant les paramètres de connexion dont l'utilisateur `operator` a besoin pour se connecter à la *seconde* instance Tivoli Workload Scheduler.

Configuration de l'agent

Les paramètres de configuration de l'agent sont contenus dans le fichier `JobManager.ini`, pour obtenir le chemin d'accès à ce fichier, voir «Emplacement de l'installation des produits et des composants», à la page 1. Dans un environnement distribué, si une passerelle est configurée pour permettre au gestionnaire de domaine maître ou au gestionnaire de domaine dynamique de communiquer avec un agent dynamique situé derrière une frontière de réseau, les paramètres de configuration de la passerelle de l'agent sont alors contenus dans le fichier

JobManagerGW.ini. Ce fichier est presque identique au fichier JobManager.ini, cependant, seuls des paramètres des sections [ITA], [Env] et [ResourceAdvisorAgent] requièrent une configuration. Pour ces paramètres, des définitions sont fournies pour les deux fichiers JobManager.ini et JobManagerGW.ini.

Ces fichiers sont constitués de plusieurs sections différentes. Chaque nom de section est placé entre crochets et chaque section comporte une séquence d'instructions variable = value.

Vous pouvez personnaliser les propriétés des éléments suivants :

- Propriétés d'automatisation de charge de travail gérée par événement
- Propriétés de journal
- Propriétés de trace lorsque l'agent est arrêté. Vous pouvez également personnaliser des traces lorsque l'agent est exécuté à l'aide de la procédure décrite dans la section «Configuration des propriétés de trace lorsque l'agent est en cours d'exécution», à la page 57.
- Exécuteur de travail natif
- Exécuteur de travail Java™
- Agent d'assistant de ressources
- Scanner du système

Les messages sont consignés dans le fichier suivant :

Sur les système d'exploitation Windows :
<*TWA_home*>\TWS\stdlist\JM\JobManager_message.log

Systèmes d'exploitation UNIX et Linux :
<*TWA_home*>/TWS/stdlist/JM/JobManager_message.log

Les messages de trace sont écrits dans le fichier suivant :

Sur les système d'exploitation Windows :

- <*TWA_home*>\TWS\stdlist\JM\ITA_trace.log
- <*TWA_home*>\TWS\stdlist\JM\JobManager_trace.log
- <*TWA_home*>\TWS\JavaExt\logs\javaExecutor0.log

Systèmes d'exploitation UNIX et Linux :

- <*TWA_home*>/TWS/stdlist/JM/ITA_trace.log
- <*TWA_home*>/TWS/stdlist/JM/JobManager_trace.log
- <*TWA_home*>/TWS/JavaExt/logs/javaExecutor0.log

Consignation d'informations relatives aux types de travaux avec des options avancées

Le fichier logging.properties permet de configurer le processus de journalisation des types de travaux avec des options avancées, à l'exception des travaux de type Exécutable et Méthode d'accès.

Le fichier logging.properties se trouve sur l'agent Tivoli Workload Scheduler for z/OS, dans le répertoire *TWA_home*/TWS/JavaExt/cfg/logging.properties.

Après avoir procédé à l'installation, ce fichier est comme suit :

```
# Spécifiez les gestionnaires à créer dans le consignateur situé à la racine
# (tous les consignateurs sont des enfants du consignateur situé à la racine)
# La ligne ci-après crée deux gestionnaires
handlers = java.util.logging.ConsoleHandler, java.util.logging.FileHandler
```

```
# Définissez le niveau de consignation par défaut pour le consignateur à la racine
.level = INFO

# Définissez le niveau de consignation par défaut pour les nouvelles
  instances ConsoleHandler
java.util.logging.ConsoleHandler.level = INFO

# Définissez le niveau de consignation par défaut pour les nouvelles
  instances FileHandler
java.util.logging.FileHandler.level = ALL
java.util.logging.FileHandler.pattern
    = C:\TWA_home\TWS\JavaExt\logs\javaExecutor%g.log
java.util.logging.FileHandler.limit
    = 1000000
java.util.logging.FileHandler.count
    = 10

# Définissez le formateur par défaut pour les nouvelles instances ConsoleHandler
java.util.logging.ConsoleHandler.formatter = java.util.logging.SimpleFormatter
java.util.logging.FileHandler.formatter = java.util.logging.SimpleFormatter

# Définissez le niveau de consignation par défaut pour le consignateur
  nommé com.mycompany
com.ibm.scheduling = INFO
```

Vous pouvez personnaliser :

- Le niveau de consignation (de INFO à WARNING, ERROR, ou ALL) des mots clés suivants :

  .level Définit le niveau de journalisation pour le journal d'événements interne.

  com.ibm.scheduling
  Définit le niveau de journalisation pour les types de travaux avec des options avancées. Pour consigner des informations relatives aux types de travaux avec des options avancées, définissez ce mot-clé sur ALL.

- Le chemin d'écriture des journaux, spécifié par le mot clé suivant :

  `java.util.logging.FileHandler.pattern`

Toutes les propriétés des fichiers `JobManager.ini` et `JobManagerGW.ini` ne sont pas personnalisées. Pour obtenir la liste des propriétés configurables, consultez les sections suivantes :

- «Configuration des propriétés des messages de journal [JobManager.Logging.cclog]», à la page 55.
- «Configuration des propriétés de trace lorsque l'agent est arrêté [JobManager.Logging.cclog]», à la page 56.
- «Configuration des propriétés communes des lanceurs de tâches [Launchers]», à la page 60.
- «Configuration des propriétés du lanceur de travaux natif [NativeJobLauncher]», à la page 61.
- «Configuration des propriétés du lanceur de travaux Java [JavaJobLauncher]», à la page 63.
- «Configuration des propriétés de l'agent assistant de ressources [ResourceAdvisorAgent]», à la page 63.
- «Configuration des propriétés du scanner du système [SystemScanner]», à la page 65

- Configuration des propriétés de l'automatisation de la charge de travail commandée par les événements [EventDrivenWorkload]

Configuration des propriétés générales [ITA]

Dans le fichier `JobManagerGW.ini`, il est possible d'ajouter quelques propriétés générales à la section suivante :

`[ITA]`

Vous pouvez ajouter ou modifier la propriété suivante :

http_proxy
Adresse URL du proxy configuré dans un environnement distribué par lequel des agents ou des passerelles communiquent avec le serveur du courtier installé sur le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique. La valeur est `https_proxy =http://<poste_travail_proxy>:<port_poste_travail_proxy>`, où :

- *<poste_travail_proxy>* correspond au nom d'hôte complet du poste de travail sur lequel le proxy est configuré.
- *<port_poste_travail_proxy>* correspond au numéro de port du poste de travail sur lequel le proxy est configuré.

Redémarrez l'agent après la modification de la propriété.

Configuration des propriétés des messages de journal [JobManager.Logging.cclog]

Pour configurer les journaux, éditez la section [JobManager.Logging.cclog] dans le fichier `JobManager.ini`. Cette procédure nécessite l'arrêt et le redémarrage de l'agent Tivoli Workload Scheduler

La section qui contient les propriétés de journal est appelée :

`[JobManager.Logging.cclog]`

Vous pouvez modifier les propriétés suivantes :

JobManager.loggerhd.fileName
Nom du fichier où sont consignés les messages.

JobManager.loggerhd.maxFileBytes
Taille maximale que peut atteindre le fichier journal. La valeur par défaut est de 1024000 octets.

JobManager.loggerhd.maxFiles
Nombre maximum de fichiers journaux stockés. La valeur par défaut est 3.

JobManager.loggerhd.fileEncoding
Par défaut, les fichiers journaux de l'agent ne sont pas codés au format UTF-8. Si vous souhaitez produire le journal dans un autre format, ajoutez cette propriété et spécifiez la page de code requise.

JobManager.loggerfl.level
Quantité d'informations à fournir dans les journaux. La plage de valeurs se situe entre 4000 et 7000. Les valeurs plus faibles correspondent à des journaux plus détaillés. La valeur par défaut est 3000.

JobManager.ffdc.maxDiskSpace
Dépassement de cet espace disque maximum, les fichiers journaux collectés par le mécanisme de capture de données à la première défaillance sont supprimés, en commençant par les fichiers les plus anciens.

JobManager.ffdc.baseDir
Répertoire dans lequel sont copiés des fichiers journaux et de trace collectés par l'outil de capture de données à la première défaillance. Le répertoire par défaut est <*TWA_home*>\TWS\stdlist\JM\JOBMANAGER-FFDC.

JobManager.ffdc.filesToCopy
Fichiers journaux et de trace (JobManager_message.log et JobManager_trace.log) collectés par l'outil de capture de données à la première défaillance se trouvant dans <*TWA_home*>\TWS\stdlist\JM. Par exemple, JobManager.ffdc.filesToCopy = "/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JobManager_message.log" "/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JobManager_trace.log"

Lorsqu'un message est consigné (JobManager.ffdc.triggerFilter = JobManager.msgIdFilter) avec un ID en corrélation avec le modèle "AWSITA*E" (JobManager.msgIdFilter.msgIds = AWSITA*E), lequel correspond à tous les messages d'erreur, les fichiers journaux et de trace (JobManager.ffdc.filesToCopy = "/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JobManager_message.log" "/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JobManager_trace.log") sont copiés (JobManager.ffdc.className = ccg_ffdc_filecopy_handler) to the directory JOBMANAGER-FFDC (JobManager.ffdc.baseDir = /opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JOBMANAGER-FFDC). Si les fichiers copiés dépassent 10 Mo (JobManager.ffdc.maxDiskSpace = 10000000), les fichiers les plus anciens sont alors supprimés les premiers (JobManager.ffdc.quotaPolicy = QUOTA_AUTODELETE).

Configuration des propriétés de trace lorsque l'agent est arrêté [JobManager.Logging.cclog]

Comment configurer les propriétés de trace lorsque l'agent est arrêté.

Pour configurer les propriétés de trace lorsque l'agent est arrêté, éditez la section [JobManager.Logging] dans le fichier JobManager.ini, puis redémarrez agent Tivoli Workload Scheduler.

La section qui contient les propriétés de trace est appelée :
[JobManager.Logging.cclog]

Vous pouvez modifier les propriétés suivantes :

JobManager.trhd.fileName
Nom du fichier de trace.

JobManager.trhd.maxFileBytes
Taille maximale que peut atteindre le fichier de trace. La valeur par défaut est de 1024000 octets.

JobManager.trhd.maxFiles
Nombre maximum de fichiers de trace stockés. La valeur par défaut est 3.

JobManager.trfl.level
Détermine le type des messages trace consignés. Modifiez cette valeur pour choisir de consigner vous-même, ou sur demande du service de support logiciel IBM. Les valeurs valides sont les suivantes :

DEBUG_MAX
Traçage maximum. Chaque message de trace du code est écrit dans les journaux de trace.

INFO Les messages *informatifs*, *d'avertissement*, *d'erreur* et *critiques* sont écrits dans la trace. Valeur par défaut.

WARNING
Les messages d'*avertissement*, d'*erreur* et *critiques* sont écrits dans la trace.

ERROR
Tous les messages de trace *error* et *critical* sont écrits dans la trace.

CRITICAL
Seules les messages provoquant l'arrêt de l'agent sont écrits dans la trace.

La trace de sortie (JobManager_trace.log) est au format XML.

Configuration des propriétés de trace lorsque l'agent est en cours d'exécution

Utilisez la commande **twstrace** pour définir la trace sur l'agent lorsqu'il est en cours d'exécution.

A l'aide de la commande **twstrace** vous pouvez effectuer les actions suivantes sur l'agent lorsque ce dernier est en cours d'exécution :

- «Affichage de la syntaxe de commande et vérification de la version».
- «Activation ou désactivation de la trace», à la page 58.
- Définissez les traces sur un niveau spécifique, indiquez le nombre de fichiers de trace que vous voulez créer et la taille minimale de chacun d'eux. Voir «Définition des informations de trace», à la page 58.
- «Affichage des informations de trace», à la page 58.
- Collectez les fichiers de trace, les fichiers message et les fichiers de configuration dans un fichier compressé. Voir «Collecte des informations de trace», à la page 59.

Vous pouvez également configurer les traces lorsque l'agent n'est pas en cours d'exécution en modifiant la section [JobManager.Logging] du fichier `JobManager.ini`, comme indiqué dans la section Configuration de l'agent. Cette procédure nécessite l'arrêt et le redémarrage de l'agent.

Commande twstrace

Utilisez la commande **twstrace** pour configurer des traces et collecter des fichiers journaux, de traces et de configuration (ita.ini et jobManager.ini) pour les agents. Vous collectez toutes les informations dans un fichier compressé lorsque la commande est en cours d'exécution, sans l'arrêter ni la redémarrer.

Affichage de la syntaxe de commande et vérification de la version

Pour afficher la syntaxe et les options de la commande, utilisez la syntaxe ci-dessous.

Syntaxe

twstrace -u | -v

Paramètres

-u Affiche la syntaxe de la commande.

-v Affiche la version de la commande.

Activation ou désactivation de la trace

Pour définir le niveau de trace maximal ou le désactiver, utilisez la syntaxe ci-dessous.

Syntaxe

twstrace -enable | -disable

Paramètres

-enable
Définit le niveau de trace maximal. Le niveau maximal est **3000**.

-disable
Désactive les traces sur l'agent.

Définition des informations de trace

Pour définir un niveau de trace spécifique, indiquez le nombre de fichiers de trace que vous voulez créer ainsi que la taille maximale des fichiers de trace en utilisant la syntaxe ci-dessous.

Syntaxe

twstrace [-level <*numéro_niveau*> **] [-maxFiles** <*nombre_fichiers*> **] [-maxFileBytes** <*nombre_octets*> **]**

Paramètres

-level <*numéro_niveau*>
Définissez le niveau de trace en indiquant une valeur comprise entre 1000 et 3000.

-maxFiles <*nombre_fichiers*>
Indiquez le nombre de fichiers de trace que vous voulez créer.

-maxFileBytes <*nombre_octets*>
Définissez la taille maximale en octets que peuvent atteindre les fichiers de trace. La valeur par défaut est **1024000** octets.

Affichage des informations de trace

Pour afficher le niveau de trace courant, le nombre de fichiers de trace et leur taille maximale, utilisez la syntaxe ci-dessous.

Syntaxe

twstrace -level | -maxFiles | -maxFileBytes

Paramètres

-level
Affiche le niveau de trace que vous définissez.

-maxFiles
Affiche le nombre de fichiers de trace que vous créez.

-maxFileBytes
Affiche la taille maximale que vous définissez pour chaque fichier de trace

Exemple

L'exemple montre les informations que vous recevez lorsque vous exécutez la commande suivante :

```
twstrace -level -maxFiles -maxFileBytes
AWSITA176I The trace properties are: level="1000",
max files="3", file size="1024000".
```

Collecte des informations de trace

Pour collecter les fichiers de trace, les fichiers message et les fichiers de configuration dans un fichier compressé, utilisez la syntaxe suivante.

Syntaxe

twstrace -getLogs [**-zipFile** <*nom_fichier_compressé*>] [**-host** <*nom_hôte*>] [**-protocol** {*http* | *https* } [**-port** <*numéro_port*>][**-iniFile** <*nom_fichier_ini*>]

Paramètres

-zipFile <*nom_fichier_compressé*>
Indiquez le nom du fichier compressé contenant toutes les informations, à savoir, les fichiers journaux, de trace et de configuration (ita.ini et jobManager.ini) pour l'agent. Le fichier par défaut est **logs.zip**.

-host <*nom_hôte*>
Indiquez le nom d'hôte ou l'adresse IP de l'agent pour lequel vous voulez collecter la trace. La valeur par défaut est **localhost**.

-protocol *http*|*https*
Indiquez le protocole de l'agent pour lequel vous collectez la trace. La valeur par défaut est le protocole indiqué dans le fichier **.ini** de l'agent.

-port <*numéro_port*>
Indiquez le port de l'agent. Par défaut il s'agit du numéro de port de l'agent pour lequel vous exécutez la ligne de commande.

-iniFile <*nom_fichier_ini*>
Indiquez le nom du fichier **.ini** contenant la configuration SSL de l'agent pour lequel vous voulez collecter les traces. Si vous collectez les traces pour un agent distant pour lequel vous avez personnalisé les certificats de sécurité, vous devez importer le certificat sur l'agent local et indiquer le nom du fichier **.ini** contenant ces informations. Pour cela, procédez comme suit :

1. Extrayez le certificat du magasin de clés de l'agent distant.
2. Importez le certificat dans un magasin de clés d'agent local. Vous pouvez créer un magasin de clés ad hoc dont le nom doit être **TWSClientKeyStore.kdb**.
3. Créez un fichier **.ini** dans lequel vous indiquez :
   - **0** dans la propriété **tcp_port** comme suit :
     ```
     tcp_port=0
     ```
   - Le port de l'agent distant dans la propriété **ssl_port** est le suivant :

ssl_port=<*ssl_port*>

- Le chemin d'accès au magasin de clés créé à l'étape 2, à la page 59 dans la propriété **key_repository_path** est le suivant :

  key_repository_path=<*chemin_magasin_clés_agent_local*>

Configuration des propriétés communes des lanceurs de tâches [Launchers]

Dans le fichier JobManager.ini, la section contenant les propriétés communes aux différents lanceurs de tâches (ou exécuteurs) est appelée :

[Launchers]

Vous pouvez modifier les propriétés suivantes :

BaseDir
Chemin d'installation de l'agent Tivoli Workload Scheduler.

CommandHandlerMinThreads
La valeur par défaut est **20**.

CommandHandlerMaxThreads
La valeur par défaut est **100**.

CpaHeartBeatTimeSeconds
Intervalle d'interrogation (en secondes) permettant de vérifier que la processus d'**agent** est toujours actif et en cours d'exécution. S'il est inactif, le produit arrête également le processus **JobManager**. La valeur par défaut est **30**.

DirectoryPermissions
Droits d'accès attribués à l'agent pour la création de répertoires lors de l'exécution de travaux. La valeur par défaut est **0755**. Les valeurs prises en charge sont des entrées au format UNIX en notation hexadécimale.

ExecutorsMaxThreads
La valeur par défaut est **400**.

ExecutorsMinThreads
La valeur par défaut est **38**.

FilePermissions
Droits d'accès attribués à l'agent pour la création de fichiers lors de l'exécution de travaux. La valeur par défaut est **0755**. Les valeurs prises en charge sont des entrées au format UNIX en notation hexadécimale.

MaxAge
Délai de conservation en jours des journaux de travaux (à l'emplacement *TWA_home*/TWS/stdlidst/JM) avant suppression. La valeur par défaut est **30**. La plage des valeurs possibles commence à 1 jour.

NotifierMaxThreads
La valeur par défaut est **5**.

NotifierMinThreads
La valeur par défaut est **3**.

SpoolDir
Chemin du dossier contenant le jobstore et les sorties. La valeur par défaut est :

valeur de BaseDir/stdlidst/JM

StackSizeBytes
Taille de la pile du système d'exploitation (en octets). La valeur par défaut est **DEFAULT**, ce qui signifie que l'**agent** utilise la valeur par défaut pour le système d'exploitation.

Configuration des propriétés du lanceur de travaux natif [NativeJobLauncher]

Dans le fichier `JobManager.ini`, la section contenant les propriétés du lanceur de travaux natif est appelée :

```
[NativeJobLauncher]
```

Vous pouvez modifier les propriétés suivantes :

AllowRoot
S'applique uniquement aux systèmes UNIX. Indique si le superutilisateur peut exécuter des travaux sur l'agent. Ce paramètre peut être défini sur `true` ou `false`. La valeur par défaut est `true`.

CheckExec
Si elle est définie sur `true`, avant de lancer le travail, l'agent vérifie la disponibilité et les droits d'exécution du fichier binaire. La valeur par défaut est `true`.

JobUnspecifiedInteractive
S'applique uniquement aux systèmes d'exploitation Windows. Spécifie si les travaux natifs sont lancés en mode interactif. Ce paramètre peut être défini sur `true` ou `false`. La valeur par défaut est `false`.

KeepCommandTraces
Définissez cette propriété sur `true` afin d'enregistrer les traces de l'appel de méthode pour les actions effectuées sur une définition de travail, par exemple, lors de la sélection d'une liste de réquisition. Ces fichiers sont enregistrés dans le chemin `/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/r3batch_cmd_exec`. Le paramètre par défaut est `false`.

KeepJobCommandTraces
Définissez cette propriété sur `true` pour enregistrer les traces de l'appel de méthode pour les actions effectuées sur une instance de travail, par exemple, afficher une liste de spool. Ces fichiers sont enregistrés dans le fichier .zip de l'instance de travail. Le paramètre par défaut est `true`.

LoadProfile
Spécifie si le profil utilisateur va être chargé. Ce paramètre peut être défini sur `true` ou `false`. La valeur par défaut est `true`.

PortMax
La plage maximum des numéros de port utilisés par le lanceur de tâches pour communiquer avec le gestionnaire de travaux. La valeur par défaut est 0 ; cette valeur indique au système d'exploitation d'attribuer le port automatiquement.

PortMin
La plage minimum des numéros de port utilisés par le lanceur de tâches pour communiquer avec le gestionnaire de travaux. La valeur par défaut est 0 ; cette valeur indique au système d'exploitation d'attribuer le port automatiquement.

PromotedNice

Utilisée dans l'assurance de service de charge de travail. Cette propriété n'est pas prise en charge sur l'Agent for z/OS.

Pour les systèmes d'exploitation UNIX et Linux uniquement, attribue la valeur prioritaire à un travail critique qui soit être promu afin que le système d'exploitation le traite avant les autres. S'applique aux travaux critiques ou à leurs prédécesseurs qui doivent être promus afin qu'ils puissent commencer à l'heure critique locale.

Les valeurs limites varient en fonction des plateformes, mais généralement, les valeurs inférieures correspondent aux niveaux de haute priorité et vice versa. La valeur par défaut est -1.

Sachez que :

- Le processus de promotion n'est efficace qu'avec des valeurs négatives. Si vous définissez une valeur positive, le système l'exécute avec la valeur par défaut -1.
- ne valeur hors plage (par exemple -200) incite le système d'exploitation à promouvoir automatiquement les travaux dont la valeur nice attribuée est la plus basse.
- L'utilisation abusive du mécanisme de promotion (c'est-à-dire la définition d'un nombre excessif de travaux comme critiques et la définition de la valeur de priorité la plus élevée ici) risque de surcharger le système d'exploitation, entraînant un impact négatif sur les performances générales du poste de travail.

PromotedPriority

Utilisée dans l'assurance de service de charge de travail. Cette propriété n'est pas prise en charge sur l'Agent for z/OS.

Pour les systèmes d'exploitationWindows uniquement, cette valeur indique la priorité selon laquelle le système d'exploitation traite un travail critique lorsqu'il est promu. S'applique aux travaux critiques ou à leurs prédécesseurs qui doivent être promus afin qu'ils puissent commencer à l'heure critique locale. Les valeurs valides sont les suivantes :

- `High`
- `AboveNormal` (valeur par défaut)
- `Normal`
- `BelowNormal`
- `Low` ou `Idle`

Si vous définissez une valeur de priorité inférieure à celle qui peut être attribuée aux travaux non critiques, aucun avertissement n'est envoyé.

RequireUserName

Lorsqu'il est défini sur `true`, vous devez ajouter le nom d'utilisateur dans la définition de travail JSDL.

Lorsqu'il est défini sur `false`, il s'exécute avec le nom d'utilisateur utilisé par le gestionnaire de travaux, à savoir :

- *utilisateur_TWS* sur les systèmes UNIX et Linux
- Le compte de système local sur les systèmes Windows

La valeur par défaut est `false`.

ScriptSuffix

Suffixe à utiliser lors de la création des fichiers script. Il s'agit de :

.cmd Pour Windows

.sh Pour UNIX

VerboseTracing
Active la fonction de trace prolixe. Ce paramètre est défini par défaut sur `true`.

Configuration des propriétés du lanceur de travaux Java [JavaJobLauncher]

Dans le fichier `JobManager.ini`, la section contenant les propriétés du lanceur de travaux Java est nommée :

```
[JavaJobLauncher]
```

Vous pouvez modifier les propriétés suivantes :

JVMDir
Chemin d'accès à la machine virtuelle utilisée pour démarrer les types de travail avec options avancées. Vous pouvez modifier le chemin en le remplaçant par un autre compatible avec la machine virtuelle Java.

JVMOptions
Les options à fournir à la machine virtuelle Java pour lancer les types de travail avec options avancées. Les mot clés pris en charge pour établir une connexion sécurisée sont :

- htttps.proxyHost
- https.proxyPort

Les mots clés pris en charge pour établir une connexion non sécurisée sont :

- Dhttp.proxyHost
- Dhttp.proxyPort

Par exemple, pour définir des types de travail avec options avancées, en fonction du protocole http JVM par défaut, sur le serveur proxy non authentifié appelé avec le nom myproxyserver.mycompany.com, définissez l'option suivante :

```
JVMOptions = -Dhttp.proxyHost=myproxyserver.mycompany.com
 -Dhttp.proxyPort=80
```

Configuration des propriétés de l'agent assistant de ressources [ResourceAdvisorAgent]

Dans les fichiers `JobManager.ini` et `JobManagerGW.ini`, la section contenant les propriétés de l'agent assistant de ressources s'intitule :

```
[ResourceAdvisorAgent]
```

Vous pouvez modifier les propriétés suivantes :

BackupResourceAdvisorUrls
La liste des adresses URL retournées par le maître Tivoli Workload Scheduler dans un environnement réparti ou par le gestionnaire de domaine dynamique dans un environnement z/OS ou réparti. L'agent utilise cette liste pour se connecter au maître ou au gestionnaire de domaine dynamique.

CPUScannerPeriodSeconds
L'intervalle de temps où l'agent d'assistant de ressources collecte les informations de ressources concernant l'unité centrale locale. La valeur par défaut est toutes les 10 secondes.

FullyQualifiedHostname
Nom d'hôte qualifié complet de l'agent. Il est configuré automatiquement lors de l'installation et utilisé pour la connexion au maître dans un environnement réparti ou au gestionnaire de domaine dynamique dans un environnement z/OS ou réparti. Modifiez-le uniquement si le nom d'hôte est modifié après l'installation.

NotifyToResourceAdvisorPeriodSeconds
L'intervalle de temps où l'agent d'assistant de ressources transfert les informations des ressources collectées à l'assistant de ressources. La valeur par défaut (et maximum) est toute les 180 secondes.

ResourceAdvisorUrl

JobManager.ini
L'adresse URL du maître dans un environnement réparti ou de gestionnaire de domaine dynamique dans un environnement z/OS ou réparti qui héberge l'agent. Cette URL est utilisée jusqu'à ce que le serveur réponde en renvoyant la liste de ses URL. La valeur est `https://$(tdwb_server):$(tdwb_port)/JobManagerRESTWeb/JobScheduler/resource`, où :

$(*tdwb_server***)**
Nom d'hôte qualifié complet du maître dans un environnement réparti, ou du gestionnaire de domaine dynamique dans un environnement z/OS ou réparti.

$(*tdwb_port***)**
Numéro de port du maître dans un environnement réparti, ou du gestionnaire de domaine dynamique dans un environnement z/OS ou réparti.

Il est automatiquement configuré au moment de l'installation. Modifiez cette valeur uniquement si le nom d'hôte ou le nom de port est modifié après l'installation, ou si vous n'utilisez pas de connexion sécurisée (adresse définie sur `http`). Si vous définissez le numéro de port sur zéro, l'agent d'assistant de ressources ne démarre pas. Le port est défini sur zéro si, au moment de l'installation, vous avez spécifié que vous n'utiliseriez pas le maître dans un environnement réparti ou le gestionnaire de domaine dynamique dans un environnement z/OS ou réparti.

Dans un environnement distribué, si **-gateway** est défini à `local` ou `remote`, il s'agit alors de l'adresse URL du poste de travail de l'agent dynamique dans lequel la passerelle réside et par lequel les agents dynamiques communiquent. La valeur est `https://$(tdwb_server):$(tdwb_port)/ita/JobManagerGW/JobManagerRESTWeb/JobScheduler/resource`, où :

$(*tdwb_server***)**
Nom d'hôte qualifié complet du poste de travail agent dynamique où la passerelle réside et par lequel l'agent dynamique communique avec Dynamic Workload Broker.

$(*tdwb_port***)**
Numéro de port du poste de travail agent dynamique où la passerelle réside et par lequel l'agent dynamique communique avec Dynamic Workload Broker.

JobManagerGW.ini
Dans un environnement distribué, si **-gateway** est défini à `local`, **ResourceAdvisorUrl** est alors l'adresse URL du maître ou du gestionnaire de domaine dynamique. La valeur est `https://$(tdwb_server):$(tdwb_port)/JobManagerRESTWeb/JobScheduler/resource`, où :

$(*tdwb_server***)**
Nom d'hôte qualifié complet du maître ou du gestionnaire de domaine dynamique.

$(*tdwb_port***)**
Numéro de port du maître ou du gestionnaire de domaine dynamique.

ScannerPeriodSeconds
L'intervalle de temps où l'agent d'assistant de ressources collecte les informations sur toutes les ressources du système local autres que les ressources de l'unité centrale. La valeur par défaut est toutes les 120 secondes.

L'agent assistant de ressource analyse les ressources de la machine par intermittence (le système informatique, le système d'exploitation, les systèmes de fichiers et les réseaux) et envoie périodiquement une mise à jour de leur état au maître ou au gestionnaire de domaine dynamique dans un environnement z/OS ou réparti.

L'unité centrale est analysée toutes les `CPUScannerPeriodSeconds` secondes, alors que les autres ressources le sont toutes les `ScannerPeriodSeconds` secondes. Dès qu'une des analyses présente une modification significative de l'état d'une ressource, les ressources sont synchronisées avec le maître dans un environnement réparti ou avec le gestionnaire de domaine dynamique dans un environnement z/OS ou réparti. La politique appliquée par l'agent pour signifier si l'attribut d'une ressource a changé de manière significative est la suivante :

- Une ressource a été ajoutée ou supprimée
- La valeur d'un attribut chaîne a changé
- Une valeur de l'unité centrale a changé de plus de `DeltaForCPU`
- Une valeur du système de fichiers a changé de plus de `DeltaForDiskMB` mégaoctets
- Une valeur de la mémoire a changé de plus de `DeltaForMemoryMB` mégaoctets

S'il n'y a aucune modification significative, les ressources sont synchronisées avec le maître Tivoli Workload Scheduler dans un environnement réparti ou avec le gestionnaire de domaine dynamique dans un environnement z/OS ou réparti, à la fréquence en secondes indiquée dans `NotifyToResourceAdvisorPeriodSeconds`.

Configuration des propriétés du scanner du système [SystemScanner]

Dans le fichier `JobManager.ini`, la section contenant les propriétés du scanner du système est appelée :

```
[SystemScanner]
```

Vous pouvez modifier les propriétés suivantes :

CPUSamples
Le nombre d'exemples utilisés pour calculer l'utilisation moyenne de l'unité centrale. La valeur par défaut est 3.

DeltaForCPU
La modification de l'utilisation de l'unité centrale est considérée comme importante lorsqu'elle excède ce pourcentage (par exemple, la valeur de DeltaForCPU est 20 si l'utilisation de l'unité centrale passe de 10 à 30%). La valeur par défaut est 20%.

DeltaForDiskMB
La modification de l'utilisation des ressources du système de fichiers est considérée comme importante lorsqu'elle excède cette valeur. La valeur par défaut est 100 Mo.

DeltaForMemoryMB
La modification de l'utilisation de la mémoire système est considérée comme importante lorsqu'elle excède cette valeur. La valeur par défaut est 100 Mo.

Configuration des variables d'environnement [ENV]

Ajoutez la section `[Env]` dans le fichier de configuration `JobManagerGW.ini` et insérez les variables d'environnement dont vous avez besoin dans votre environnement de planification dynamique.

Maintenance régulière

La maintenance régulière fait référence à des mécanismes qui sont utilisés sur vos agents dynamiques de postes de travail pour libérer de l'espace de stockage et améliorer les performances.

Contrairement aux agents tolérants aux pannes sur lesquels des tâches de maintenance doivent être exécutées manuellement à l'aide de la commande d'utilitaire **rmstdlist**, vous pouvez bénéficier d'une maintenance régulière qui s'effectue sur vos postes de travail d'agent dynamique afin de maintenir l'espace disque sous contrôle, en configurant les paramètres suivants de manière appropriée.

Tableau 17. Paramètres de configuration de l'agent. Paramètres de configuration pour la maintenance

Fichier	Paramètre	Description
JobManager.ini se trouve à l'emplacement *TWA_home*/TWS/ITA/cpa/config	MaxAge	Délai de conservation en jours des journaux de travaux (à l'emplacement *TWA_home*/TWS/stdlidst/JM) avant suppression. La valeur par défaut est 2. La plage des valeurs commence à partir de 1 jour.
	JobManager.loggerhd.maxFileBytes	Taille maximale que peut atteindre le fichier journal. La valeur par défaut est de 1024000 octets.
	JobManager.loggerhd.maxFiles	Nombre maximum de fichiers journaux pouvant être stockés dans le répertoire stdlist/JM. La valeur par défaut est 3.
	JobManager.ffdc.maxDiskSpace	Espace disque maximal atteint, par les fichiers journaux collectés par l'outil de capture de données à la première défaillance, au-delà duquel les fichiers les plus anciens sont supprimés.
	JobManager.trhd.maxFileBytes	Taille maximale que peut atteindre le fichier journal. La valeur par défaut est de 1024000 octets.
	JobManager.trhd.maxFiles	Nombre maximum de fichiers journaux stockés. La valeur par défaut est 3.
logging.properties se trouve à l'emplacement <TWA_home>/TWS/JavaExt/cfg/ Journaux à des travaux avec options avancées.	java.util.logging.FileHandler.limit	Volume maximal pour l'écriture de messages de journal sur un fichier. La valeur par défaut est 1000000 (octets)
	java.util.logging.FileHandler.count	Nombre de fichiers de sortie à faire défiler. La valeur par défaut est 10.

Configuration du serveur Dynamic Workload Broker sur le gestionnaire de domaine maître et le gestionnaire de domaine dynamique

Vous pouvez effectuer ces tâches de configuration après avoir installé le gestionnaire de domaine maître, le gestionnaire de domaine dynamique et les agents dynamiques, et chaque fois que vous souhaitez modifier ou optimiser des paramètres spécifiques dans votre environnement.

Les paramètres de configuration du serveur Dynamic Workload Broker sont définis par défaut au moment de l'installation. Vous modifiez un sous-ensemble de ces

paramètres à l'aide des fichiers créés lors de l'installation de Dynamic Workload Broker. les fichiers suivants sont créés à l'emplacement :

TWA_home/TDWB/config

ResourceAdvisorConfig.properties
Contient les informations de configuration de l'**assistant de ressources**. Pour plus d'informations, voir «Fichier ResourceAdvisorConfig.properties», à la page 70.

JobDispatcherConfig.properties
Contient les informations de configuration du **répartiteur de travaux**. Pour plus d'informations, voir «Fichier JobDispatcherConfig.properties», à la page 72.

BrokerWorkstation.properties
Contient les informations de configuration du serveur de courtier. «Fichier BrokerWorkstation.properties», à la page 75

CLIConfig.properties
Contient des informations de configuration pour la ligne de commande de Dynamic Workload Broker. Ce fichier est décrit dans *Tivoli Workload Scheduler : Planification dynamique de la charge de travail*.

audit.properties
Contient des options pour configurer les audits sur les événements. Ce fichier est documenté dans *IBM Tivoli Workload Scheduler - Guide d'identification des problèmes*.

Vous pouvez modifier un sous-ensemble de paramètres figurant dans ces fichiers afin de modifier les éléments suivants :

- Signal de pulsation des agents.
- Intervalle de temps pour l'allocation d'un travail aux ressources
- Intervalle de temps pour les notifications concernant les ressources
- Heure d'interrogation lors de la vérification du statut des postes de travail du moteur distant
- Nombre maximum de résultats pour une correspondance de ressources globales
- Chiffrement des mots de passe envoyés dans les définitions JSDL
- Intervalle entre les tentatives pour une opération après un échec du **répartiteur de travaux**
- Intervalle entre les essais pour une opération après l'échec d'une notification client
- Paramètres d'archivage des données de travaux
- Paramètres de l'historique des travaux
- Propriétés de ligne de commande (voir *IBM Tivoli Workload Scheduler : Planification dynamique de la charge de travail*)

Les paramètres éditables sont répertoriés dans les sections suivantes. Si vous modifiez un paramètre non répertorié, il se peut que le produit ne fonctionne pas. Après avoir modifié les fichiers, vous devez arrêter et redémarrer le serveur IBM WebSphere.

Maintenance du serveur Dynamic Workload Broker sur le gestionnaire de domaine maître et le gestionnaire de domaine dynamique

Avec un serveur Dynamic Workload Broker installé avec le gestionnaire de domaine maître et le gestionnaire de domaine dynamique, et un autre serveur installé avec chaque gestionnaire de secours, vous disposez d'au moins deux serveurs dans votre réseau Tivoli Workload Scheduler. Le serveur s'exécutant avec le gestionnaire de domaine maître est le seul serveur actif en permanence. Les serveurs installés dans les gestionnaires de secours sont en veille jusqu'à ce que vous permutiez les gestionnaire et celui qui se troue sur le nouveau gestionnaire devient alors le serveur actif (pour obtenir des informations importantes sur ce scénario, voir «Démarrage, arrêt et affichage du statut de Dynamic Workload Broker», à la page 407). Pour effectuer une transition en douceur d'un serveur vers un autre lorsque vous permutez les gestionnaires, il est important de conserver les mêmes paramètres de configuration dans les fichiers `ResourceAdvisorConfig.properties` et `JobDispatcherConfig.properties` sur tous les serveurs dont vous disposez. Lorsque vous apportez une modification à l'un de ces fichiers de votre serveur Dynamic Workload Broker actif, n'oubliez pas d'appliquer la même modification au serveur Dynamic Workload Broker en veille dans votre gestionnaire de secours.

Certains des paramètres du serveur Dynamic Workload Broker sont stockés dans le fichier **BrokerWorkstation.properties** local ainsi que dans la base de données Tivoli Workload Scheduler. Lorsque vous passez dans le gestionnaire de domaine maître de secours ou dans le gestionnaire de domaine dynamique, les paramètres du serveur Dynamic Workload Broker sont automatiquement mis à jour sur le poste de travail de secours. Pour plus d'informations propos du fichier **BrokerWorkstation.properties**, voir «Fichier BrokerWorkstation.properties», à la page 75.

Remarque : La base de données est automatiquement remplie avec les informations provenant du poste de travail actif, qu'il s'agisse du poste de travail du gestionnaire ou du poste de travail de secours. Par exemple, si vous modifiez les paramètres du serveur Dynamic Workload Broker sur le gestionnaire de domaine maître de secours ou sur le gestionnaire de domaine dynamique, ces modifications sont enregistrées dans la base de données. Lorsque vous repassez au poste de travail du gestionnaire, les modifications sont appliquées dans le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique et les paramètres locaux associés sont écrasés.

Il est important de conserver aussi les données inhérentes à chaque serveur Dynamic Workload Broker à jour. Si vous modifiez le nom d'hôte ou le numéro de pot d'un de vos serveurs Dynamic Workload Broker, utilisez les commandes `exportserverdata` et `importserverdata` à partir de la ligne de commande Dynamic Workload Broker pour enregistrer ces modifications dans la base de données Tivoli Workload Scheduler. Pour plus d'informations à propos de ces commandes, voir *Scheduling Workload Dynamically*.

Les enregistrements de la base de données de vos serveurs poste de travail Workload Broker ont tous LOCALHOST comme nom d'hôte de poste de travail. Laissez l'enregistrement en l'état. Ne remplacez pas LOCALHOST par le nom d'hôte ou l'adresse IP réelle du poste de travail. LOCALHOST est utilisé délibérément pour s'assurer que les travaux soumis à partir de Tivoli Workload

Scheduler sont envoyés avec succès au nouveau Dynamic Workload Broker local lorsque vous permutez le gestionnaire de domaine maître or gestionnaire de domaine dynamique.

Activation de communications non sécurisées avec le serveur Dynamic Workload Broker

Par défaut, le serveur Dynamic Workload Broker utilise des communications sécurisées. Vous devrez peut-être activer les communications non sécurisées, même si cela n'est pas recommandé.

Pour activer les communications non sécurisées avec le serveur Dynamic Workload Broker, procédez comme suit sur le gestionnaire de domaine maître:

1. Exécutez la commande exportserverdata située dans le répertoire *installation_directory*/TDWB/bin:

   ```
   exportserverdata -dbUsr db_instance_admin -dbPwd db_instance_admin_pwd
   ```

2. Ouvrez le fichier server.properties ainsi obtenu dans un éditeur de texte à plat.
3. Copiez la ligne suivante :

   ```
   https://nom_hôte:port/JobManagerRESTWeb/JobScheduler
   ```

4. Modifiez la ligne copiée en remplaçant **https** par **http** :

   ```
   http://nom_hôte:port/JobManagerRESTWeb/JobScheduler
   ```

   Le fichier contient maintenant deux lignes spécifiant le mode de connexion, une ligne spécifiant le type https et une ligne indiquant le mode http.
5. Enregistrez le fichier.
6. Importez les nouvelles données à l'aide de la commande importserverdata située dans *installation_directory*/TDWB/bin :

   ```
   importserverdata -dbUsr db_instance_admin -dbPwd db_instance_admin_pwd
   ```

Pour plus d'informations à propos des commandes exportserverdata et importserverdata, voir *Tivoli Workload Scheduler - Planification dynamique de la charge de travail*.

Fichier ResourceAdvisorConfig.properties

Les paramètres de ce fichier affectent les paramètres de serveur Dynamic Workload Broker suivants :

- Signal de présence des agents
- Intervalle de temps pour l'allocation d'un travail aux ressources
- Intervalle de temps pour les notifications concernant les ressources
- Heure d'interrogation lors de la vérification du statut des postes de travail du moteur distant
- Nombre maximum de résultats pour une correspondance de ressources globales

Vous pouvez modifier les paramètres suivants du fichier `ResourceAdvisorConfig.properties` :

DatabaseCheckInterval
: Spécifie l'intervalle au cours duquel le serveur Tivoli Dynamic Workload Broker vérifie la disponibilité de la base de données. La valeur par défaut est **180**.

ResourceAdvisorURL
: Indique l'adresse URL de l'**assistant de ressources**.

RaaHeartBeatInterval

Indique l'intervalle de temps pendant lequel l'**assistant de ressources** attend un signal de présence de l'agent dynamique. La valeur par défaut est **200** secondes. Une fois le nombre maximal de nouvelles tentatives (spécifié dans le paramètre **MissedHeartBeatCount**) dépassé, l'**assistant de ressources** signale l'ordinateur associé comme étant non disponible. Dans un réseau lent, il peut s'avérer utile de définir ce paramètre sur une valeur plus élevée. Cependant, la définition d'une valeur plus élevée retarde les mises à jour des statuts de disponibilité des systèmes informatique. Si vous réduisez cette valeur ainsi que celle définie pour le paramètre **NotifyToResourceAdvisorPeriodSeconds**, cela peut générer du trafic réseau et augmenter l'utilisation de l'UC lors de la mise à jour des données de la mémoire cache. La valeur de ce paramètre doit correspondre au paramètre **NotifyToResourceAdvisorPeriodSeconds** défini dans le fichier `JobManager.ini`, laquelle définit l'intervalle au cours duquel chaque agent dynamique doit envoyer le signal à l'**Assistant de ressources**.

MissedHeartBeatCount

Spécifie le nombre de signaux de présence après lequel l'ordinateur est répertorié comme non disponible. La valeur par défaut est 2. Dans un réseau lent, il peut s'avérer utile de définir ce paramètre sur une valeur plus élevée.

MaxWaitingTime

Spécifie l'intervalle de temps maximum qu'un travail doit attendre avant qu'une ressource soit disponible. Si l'intervalle expire avant qu'une ressource soit disponible, le statut du travail change en Resource Allocation Failure (Echec d'allocation de la ressource). La valeur par défaut est 600 secondes. Vous pouvez remplacer cette valeur pour chaque travail spécifique en utilisant le paramètre **Temps d'attente maximum de la ressource** défini dans la console de définition de courtage des travaux. Pour de plus amples informations sur le paramètre **Temps d'attente maximum de la ressource**, voir l'aide en ligne de la Console de définition de courtage de travaux. Si vous définissez ce paramètre sur -1, aucun temps d'attente n'est appliqué pour les travaux. Si vous définissez ce paramètre sur 0, l'**assistant de ressources** essaye une seule fois de trouver les ressources correspondantes et, s'il n'en trouve aucune, le statut du travail est modifié en ALLOCATION FAILED. Si vous augmentez cette valeur, tous les travaux soumis conservent plus longtemps le statut EN ATTENTE, l'**assistant de ressources** tente alors de trouver des ressources correspondantes en fonction de la valeur définie pour le paramètre **CheckInterval**.

CheckInterval

Spécifie l'intervalle de temps que l'**assistant de ressources** doit attendre avant de tenter à nouveau de trouver des ressources correspondantes pour un travail qui n'a trouvé aucune ressource dans l'intervalle précédent. La valeur par défaut est 60 secondes.

TimeSlotLength

Spécifie l'intervalle pendant lequel l'**assistant de ressources** alloue des ressources à chaque travail. Les travaux soumis après l'expiration de cet intervalle sont pris en compte dans un nouvel intervalle. La valeur par défaut est 15 secondes. La valeur par défaut est adaptée à la plupart des environnements et ne doit pas être modifiée. Lorsque vous définissez une valeur plus élevée pour ce paramètre, l'**assistant de ressources** affecte les ressources aux travaux à priorité élevée plutôt qu'aux travaux à priorité faible quand tous les travaux tentent d'obtenir la même ressource.

Cependant, cela peut entraîner un ralentissement du traitement de correspondance des ressources de travail et de la mise à jour de l'état de ressources par les agents. Lorsque ce paramètre est défini sur une valeur plus faible, l'**assistant de ressources** traite la correspondance des ressources plus rapidement et, si vous possédez un nombre élevé d'agents avec des mises à jour fréquentes, met immédiatement à jour le référentiel de ressources. Lorsque les configurations de travaux correspondent à plusieurs ressources, des valeurs plus faibles garantissent un meilleur équilibrage de charge. Si la plupart des travaux utilisent une allocation de ressources, ne réduisez pas cette valeur car l'évaluation de l'allocation requiert plusieurs ressources de traitement.

NotifyTimeInterval
Spécifie l'intervalle dans lequel l'**assistant de ressources** tente à nouveau d'envoyer des notifications concernant le statut du travail au **répartiteur de travail** après l'échec d'une notification. La valeur par défaut est 15 secondes. La valeur par défaut est adaptée à la plupart des environnements et ne doit pas être modifiée.

MaxNotificationCount
Spécifie le nombre maximum de fois où l'**assistant de ressources** tente d'envoyer des notifications au **répartiteur de travaux**. La valeur par défaut est 100. La valeur par défaut est adaptée à la plupart des environnements et ne doit pas être modifiée.

ServersCacheRefreshInterval

Spécifie la fréquence (en secondes) à laquelle l'assistant de ressources vérifie la liste des serveurs Dynamic Workload Broker actifs et de sauvegarde pour les mises à jour. Cette liste est créée lors de l'installation du gestionnaire de domaine maître, elle est ensuite mise à jour chaque fois qu'un maître de secours est installé et connecté à la base de données du gestionnaire de domaine maître (le gestionnaire de domaine maître et les maîtres de secours incluent également un serveur Dynamic Workload Broker). Lorsque les agents de l'assistant de ressources envoient leurs données relatives aux ressources reconnues sur chaque ordinateur, ils peuvent automatiquement permuter entre les serveurs de la liste, pour que le serveur Dynamic Workload Broker actif puisse stocker ces données dans son référentiel de ressources. C'est la raison pour laquelle les agents de l'assistant de ressources doivent toujours connaître la liste de tous les serveurs Dynamic Workload Broker. Les valeurs possibles sont comprises entre 300 (5 minutes) et 43200 (12 heures). La valeur par défaut est 600 secondes.

StatusCheckInterval
Indique l'intervalle d'attente en secondes de l'assistant de ressources avant d'interroger le statut d'une ressource. Ce délai d'attente s'applique, par exemple, lors de la vérification du statut d'un moteur distant. La valeur par défaut est de 120 secondes.

Après avoir modifié le fichier, vous devez arrêter et redémarrer WebSphere Application Server.

Fichier JobDispatcherConfig.properties

Les paramètres de ce fichier affectent les paramètres suivants du serveur Dynamic Workload Broker installé sur un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique :

- Chiffrement des mots de passe envoyés dans les définitions JSDL

- Intervalle entre les tentatives pour une opération après un échec du **répartiteur de travaux**
- Intervalle entre les essais pour une opération après l'échec d'une notification client
- Paramètres d'archivage des données de travaux
- Paramètres de l'historique des travaux
- Passerelles et paramètres de connexion du serveur Dynamic Workload Broker.

Après avoir modifié le fichier, vous devez arrêter et redémarrer le serveur IBM WebSphere.

Au cours de la mise à niveau à partir de la version 8.5.1, les valeurs que vous avez définies pour les propriétés de la version 8.5.1 sont préservées. Les valeurs par défaut des propriétés de la version 8.6 sont différentes de celles de la version 8.5.1. Pour utiliser les valeurs par défaut de la version 8.6, vous devez les changer manuellement.

Dans le fichier `JobDispatcherConfig.properties`, les paramètres suivants sont disponibles :

DatabaseCheckInterval
Spécifie l'intervalle au cours duquel le serveur Dynamic Workload Broker vérifie la disponibilité de la base de données. La valeur par défaut est **180**.

EnablePasswordEncryption
Spécifie que tout mot de passe d'utilisateur figurant dans les définitions JSDL doit être chiffré lorsque les définitions sont envoyées au agents. La valeur par défaut est `true`. Si vous définissez cette propriété sur `false`, vous forcez le serveur Dynamic Workload Broker à envoyer les mots de passe en texte normal. Ceci s'applique à toute zone de mot de passe.

RAEndpointAddress
Indique l'adresse URL de l'**assistant de ressources**.

JDURL
Indique l'adresse URL du **répartiteur de travaux**.

FailQInterval
Spécifie le nombre de secondes entre les tentatives pour une opération après les échecs suivants :

- Notification client.
- Demandes d'allocation, de réallocation, d'annulation d'allocation à l'**assistant de ressources**.
- Toute opération de base de données échouée pour des raisons de connectivité.

La valeur par défaut est 30 secondes. L'augmentation de cette valeur améliore la rapidité de reprise après incident mais peut utiliser beaucoup de ressources système si l'opération de reprise est complexe. Par exemple, si le poste de travail Workload Broker traite un nouveau fichier Symphony, cette opération peut prendre du temps. Vous devriez donc définir ce paramètre sur une valeur plus élevée. Si vous n'utilisez pas le poste de travail Workload Broker, ce paramètre peut être défini sur une valeur plus faible.

MaxCancelJobAttemptsCount
Le nombre maximum de tentatives par le répartiteur de travail pour annuler un travail reflet ou un travail s'exécutant sur un agent dynamique

lorsqu'une requête de suppression du travail est effectuée et que cette dernière ne peut pas être traitée immédiatement. La valeur par défaut est 1440 tentatives. Le répartiteur de travail tente d'annuler le travail toutes les 30 secondes selon un nombre de fois maximum spécifié par ce paramètre.

MaxNotificationCount
Spécifie le nombre maximal de nouvelles tentatives après l'échec d'une notification client. La valeur par défaut est 1440. Par exemple, si le poste de travail Workload Broker traite un nouveau fichier Symphony, cette opération peut prendre du temps. Vous devriez donc définir ce paramètre sur une valeur plus élevée. Si vous n'utilisez pas le poste de travail Workload Broker, ce paramètre peut être défini sur une valeur plus faible.

MoveHistoryDataFrequencyInMins
Spécifie la fréquence à laquelle les données des travaux doivent être déplacées dans les tables d'archivage de la base de données du **référentiel des travaux** et les tables dans la base d'archives. La fréquence est exprimée en minutes. La valeur par défaut est 60 minutes. Lorsque cette valeur est augmentée, le **répartiteur de travaux** vérifie moins souvent les travaux à déplacer. Le volume des travaux du **référentiel des travaux** peut ainsi augmenter et les demandes peuvent prendre plus de temps pour s'exécuter. Les serveurs Dynamic workload broker avec un rendement de travail élevé peuvent exiger des valeurs plus faibles, alors que les rendements de travail faibles peuvent requérir des valeurs plus élevées.

SuccessfulJobsMaxAge
Spécifie le délai pendant lequel les travaux achevés ou annulés correctement doivent être conservés dans la base de données du **référentiel des travaux** avant d'être archivés. Le délai est exprimé en heures. La valeur par défaut est 240 heures, c'est-à-dire dix jours.

UnsuccessfulJobsMaxAge
Spécifie le délai pendant lequel les travaux qui ne se sont pas achevés correctement ou dont le statut est inconnu doivent être conservés dans la base de données du **référentiel des travaux** avant d'être archivés. Le délai est exprimé en heures. La valeur par défaut est 720 heures, c'est-à-dire 30 jours.

ArchivedJobsMaxAge
Spécifie le délai pendant lequel les travaux doivent être conservés dans la base d'archives avant d'être supprimés. Le délai est exprimé en heures. La valeur par défaut est 720 heures, c'est-à-dire 30 jours.

AgentConnectTimeout
Spécifie le délai en minutes pendant lequel le serveur Dynamic Workload Broker attend une réponse de l'agent de planification après sa première tentative de connexion à cet agent. Si l'agent ne répond pas dans le délai défini, le serveur n'établit pas la connexion. La plage de valeurs se situe entre 0 et 60 (définissez 0 pour attendre indéfiniment). La valeur par défaut est 3.

AgentReadTimeout
Spécifie l'intervalle en minutes pendant lequel le serveur Dynamic Workload Broker attend pour recevoir des données des connexions établies avec un agent de planification ou une passerelle. Si aucune donnée n'arrive dans l'intervalle spécifié, le serveur ferme la connexion à l'agent. La plage de valeurs se situe entre 0 et 60 (définissez 0 pour attendre indéfiniment). La valeur par défaut est 3.

GatewayPollingTimeout
Spécifie l'intervalle en minutes pendant lequel la passerelle attend pour recevoir des données des connexions établies avec un Dynamic Workload Broker. Si aucune donnée n'arrive dans l'intervalle spécifié, la passerelle ferme la connexion au Dynamic Workload Broker. La plage de valeurs admise est comprise entre 1 et 60. La valeur par défaut est 1 minute.

GatewayConnectionTimeout
Spécifie l'intervalle en secondes pendant lequel le serveur Dynamic Workload Broker attend les données de réception d'une passerelle après que le Dynamic Workload Broker a tenté pour la première fois d'envoyer des données à celle-ci. Si la passerelle ne répond pas dans l'intervalle spécifié, le Dynamic Workload Broker n'établit pas la connexion. La plage de valeurs admise est comprise entre 1 et 60. La valeur par défaut est 10 secondes.

GatewaysNumber
Indique le nombre de passerelles que le serveur Dynamic Workload Broker peut gérer sans perte de performances. La plage de valeurs admise est comprise entre 3 et 100. La valeur par défaut est 3.

Remarque :

Vous pouvez utiliser ce fichier pour configurer le comportement du produit lors de l'archivage des données des travaux. Pour plus d'informations sur les tables d'archivage, voir «Tables de base de données historiques créées pendant l'installation», à la page 77.

Si un pic de charge de travail inattendu se produit et qu'un nettoyage de base de données est nécessaire avant la fin du délai spécifié dans le paramètre `MoveHistoryDataFrequencyInMins`, vous pouvez utiliser la commande **`movehistorydata`** pour effectuer un nettoyage avant le nettoyage planifié.

Fichier BrokerWorkstation.properties

Si vous devez modifier la configuration du serveur de courtier, après l'installation, vous pouvez éditer le fichier `BrokerWorkstation.properties`. Le fichier `BrokerWorkstation.properties` contient les propriétés de configuration suivantes :

DomainManager.Workstation.Name
Nom du poste de travail du gestionnaire de domaine.

DomainManager.Workstation.Port
Port du poste de travail du gestionnaire de domaine.

MasterDomainManager.Name
Nom du poste de travail du gestionnaire de domaine maître.

Broker.Workstation.Name
Nom du serveur du courtier dans le plan de production de Tivoli Workload Scheduler. Ce nom est attribué pour la première fois au moment de l'installation.

MasterDomainManager.HostName
Nom d'hôte du poste de travail du gestionnaire de domaine maître.

MasterDomainManager.HttpsPort
Port HTTPS du poste de travail du gestionnaire de domaine maître.

Broker.Workstation.Port
Port utilisé par le serveur du courtier pour écouter le trafic entrant

(équivalent au port Netman). Il est attribué pour la première fois au moment de l'installation. Ce numéro de port doit être identique pour tous les serveurs de courtier définis dans votre réseau Tivoli Workload Scheduler (un avec le gestionnaire de domaine maître et un pour chaque maître de secours installé) afin de garantir la cohérence lorsque vous permutez des maîtres.

DomainManager.Workstation.Domain
Nom du domaine sur lequel est enregistré le serveur du courtier.

Broker.AuthorizedCNs
Liste des préfixes des noms communs autorisés à communiquer avec le serveur du courtier. Pour plus d'informations sur l'autorisation des connexions au gestionnaire de domaine dynamique, voir «Personnalisation de la connexion SSL entre un gestionnaire de domaine maître et un gestionnaire de domaine dynamique ou son gestionnaire de secours à l'aide de vos certificats», à la page 283.

Broker.Workstation.Enable
Commutateur permettant d'activer ou de désactiver le serveur du courtier. Il peut prendre la valeur `true` ou `false`. La valeur par défaut est `true`.

Définissez cette valeur sur `false` si vous choisissez de ne pas utiliser un serveur de courtier. Si vous n'utilisez pas le serveur du courtier, vous pouvez soumettre des travaux de manière dynamique directement sur Dynamic Workload Broker (à l'aide de la ligne de commande de Dynamic Workload Console ou de Dynamic Workload Broker) sans utiliser les fonctions de planification de Tivoli Workload Scheduler.

Broker.Workstation.CpuType
Type de poste de travail affecté au serveur du courtier. Les valeurs prises en charge sont :

- gestionnaire de domaine maître (maître)
- gestionnaire de domaine maître de secours (agent tolérant aux pannes)
- gestionnaire de domaine dynamique (agent tolérant aux pannes, courtier, agent)
- gestionnaire de domaine dynamique de secours (agent tolérant aux pannes, courtier, agent)

Broker.Workstation.RetryLink
Intervalle en secondes entre deux tentatives consécutives de liaison avec le serveur du courtier. La valeur par défaut est 600.

Remarque : aucune sécurité SSL n'est disponible pour la connexion entre le gestionnaire de domaine maître et le serveur du courtier. Toutes les données se déplaçant entre les deux postes de travail sont envoyées non chiffrées. Si cela entraîne un risque lié à la sécurité dans votre environnement, vous pouvez choisir de ne pas utiliser les fonctions du serveur de courtier, en définissant le paramètre `Broker.Workstation.Enable` sur `false`.

Archivage des données de travaux

Les définitions de travaux créés à l'aide de la Console de définition de courtage de travaux ou de Dynamic Workload Console sont stockées dans la base de données du **référentiel des travaux**. La base de données **référentiel des travaux** stocke également tous les travaux créés lorsque les définitions de travaux sont soumises à Dynamic Workload Broker.

Les informations sur les travaux sont archivées régulièrement. Par défaut, les travaux exécutés correctement sont archivés une fois par jour. Les travaux dont le statut est inconnu ou ayant échoué sont archivés par défaut toutes les 72 heures. Les travaux archivés sont déplacés dans les tables d'historique du **référentiel des travaux**.

Vous pouvez configurer le délai après lequel les données des travaux sont archivées à l'aide des paramètres suivants :

- **MoveHistoryDataFrequencyInMins**
- **SuccessfulJobsMaxAge**
- **UnsuccessfulJobsMaxAge**
- **ArchivedJobsMaxAge**

Ces paramètres sont disponibles dans le fichier JobDispatcherConfig.properties, comme décrit dans «Fichier JobDispatcherConfig.properties», à la page 72. Vous pouvez également utiliser la commande **movehistorydata** pour effectuer un nettoyage avant celui planifié.

Tables de base de données historiques créées pendant l'installation

La création de la base de données diffère selon le fournisseur de base de données que vous utilisez. Si vous utilisez DB2, deux bases de données sont créées par défaut lors de l'installation du serveur Dynamic Workload Broker. Si vous utilisez Oracle, deux schémas sont créés dans la même base de données. Les noms des bases de données et des schémas sont les suivants :

IBMCDB (DB2)/ CDB (Oracle)
: Contient des données gestionnaire d'agents. Le nom est permanent et ne peut être modifié.

TDWB
: Contient des données Dynamic Workload Broker. Vous pouvez modifier le nom.

Les trois tables d'historique suivantes sont créées pendant le processus d'installation dans la base de données **TDWB**. Ces tables sont utilisées pour recueillir les données d'historique relatives aux instances de travaux.

JOA_JOB_ARCHIVES
: Contient des instances de travaux archivées. Voir tableau 18, à la page 78.

JRA_JOB_RESOURCE_ARCHIVES
: Contient des informations de ressource relatives à un travail. Voir tableau 19, à la page 78.

MEA_METRIC_ARCHIVES
: Contient les métriques collectées pour un travail. Voir tableau 20, à la page 79.

Pour améliorer la performance du SGBDR, vous pouvez régulièrement déplacer des données des tables standard vers les tables d'historique. Vous pouvez configurer la maintenance du SGBDR à l'aide du fichier JobDispatcherConfig.properties. Pour plus d'informations, voir «Fichier JobDispatcherConfig.properties», à la page 72. Vous pouvez également utiliser la commande **movehistorydata** pour déplacer les données vers les tables d'historique et supprimer les données archivées.

Tableau 18. Table de base de données JOA_JOB_ARCHIVES

Nom de la colonne	Type de données DB2	Type de données Oracle	Longueur	Null admis	Description
JOA_ID	CHAR () FOR BIT DATA	RAW	16	Non	Contient l'identificateur unique du travail
JOA_START_TIME	TIMESTAMP	TIMESTAMP	26	Oui	L'heure de début, le cas échéant
JOA_END_TIME	TIMESTAMP	TIMESTAMP	26	Oui	L'heure de fin du travail, le cas échéant
JOA_JSDL_INSTANCE	CLOB	CLOB		Non	La JSDL (définition du travail), stocké au format binaire
JOA_SUBMIT_USERNAME	VARCHAR	VARCHAR2	120	Non	L'émetteur
JOA_TIMEZONE	VARCHAR	VARCHAR2	40	Oui	Non utilisé dans cette édition
JOA_STATE	DECIMAL	NUMBER	2	Non	Le code du statut du travail
JOA_RETURN_CODE	DECIMAL	NUMBER	10	Non	Le code retour du travail
JOA_SUBMIT_TIME	TIMESTAMP	TIMESTAMP	26	Non	L'heure de soumission
JOA_NAME	VARCHAR	VARCHAR2	250	Non	Le nom de la définition du travail
JOA_NAMESPACE	VARCHAR	VARCHAR2	250	Oui	L'espace de nom de la définition du travail
JOA_ALIAS_NAME	VARCHAR	VARCHAR2	250	Oui	L'alias de la définition du travail
JOA_SUBMITTER_TYPE	VARCHAR	VARCHAR2	80	Oui	Le type d'émetteur (par exemple, TDWB CLI, TDWB UI)
JOA_UPDATE_TIME	TIMESTAMP	TIMESTAMP	26	Non	Le dernier horodatage mis à jour de la ligne en cours

Tableau 19. Table de base de données JRA_JOB_RESOURCE_ARCHIVES

Nom de la colonne	Type de données DB2	Type de données Oracle	Longueur	Null admis	Description
JOA_ID	CHAR () FOR BIT DATA	RAW	16	Non	Contient l'identificateur unique du travail
JRA_RESOURCE_NAME	VARCHAR	VARCHAR2	250	Non	Le nom interne de la ressource
JRA_RESOURCE_TYPE	VARCHAR	VARCHAR2	30	Non	Le type de ressource (par exemple, système informatique, système de fichiers...)
JRA_RESOURCE_GROUP	DECIMAL	NUMBER	5	Non	Le code du groupe (groupement d'une allocation d'un travail réel)
JRA_DISPLAY_NAME	VARCHAR	VARCHAR2	250	Oui	Le nom affiché
JRA_IS_TARGET	DECIMAL	NUMBER	1	Non	Un indicateur signalant la ressource racine (généralement le système informatique)

Tableau 20. Table de base de données MEA_METRIC_ARCHIVES

Nom de la colonne	Type de données DB2	Type de données Oracle	Longueur	Null admis	Description
JOA_ID	CHAR () FOR BIT DATA	RAW	16	Non	Contient l'identificateur unique du travail
MEA_NAME	VARCHAR	VARCHAR2	80	Non	Le nom métrique (par exemple, JOB_MEMORY_USAGE, JOB_CPU_USAGE, ...)
MEA_TYPE	CHAR	CHARACTER	10	Non	Le type de données métriques (par exemple, DECIMAL, ...)
MEA_VALUE	VARCHAR	VARCHAR2	250	Non	La mesure

Le tableau 21 répertorie le statut des travaux stockés dans les tables d'historique ainsi que les statuts de travaux disponibles dans Dynamic Workload Console et dans la ligne de commande, et les mappe aux options associées dans la commande **movehistorydata**.

Tableau 21. Statuts des travaux dans les tables d'historique

Option movehistorydata	Statut des travaux dans les tables	Dynamic Workload Consoleétat	Statuts de la ligne de commande
SuccessfulJobsMaxAge	43	Completed successfully (Terminé correctement)	SUCCEEDED_EXECUTION
	44	Annulé	CANCELED
UnsuccessfulJobMaxAge	41	Echec d'allocation de ressource	RESOURCE_ ALLOCATION_FAILED
	42	Echec de l'exécution	FAILED_EXECUTION
	45	Inconnu	UNKNOWN
	46	Unable to start (Impossible de démarrer)	NOT_EXECUTED

Configuration pour planifier des travaux J2EE

Le composant Dynamic Workload Broker permet de planifier des travaux J2EE. Pour ce faire, vous devez compléter les tâches de configuration suivantes :

- Configurer l'exécuteur J2EE sur chaque agent auquel vous soumettez les travaux J2EE.
- Configurer l'agent J2EE Job Executor sur un WebSphere Application Server externe

Configuration de l'exécuteur J2EE

Pour planifier de façon dynamique les travaux J2EE, vous devez configurer les fichiers de propriétés suivants sur chaque agent auquel vous soumettez des travaux J2EE :

- J2EEJobExecutorConfig.properties
- logging.properties
- soap.client.props

Ces fichiers sont configurés avec des valeurs par défaut au moment de l'installation. Les valeurs que vous pouvez personnaliser sont indiquées dans la description de chaque fichier.

Fichier J2EEJobExecutorConfig.properties : Le chemin d'accès à ce fichier est *TWA_home*/TWS/JavaExt/cfg/J2EEJobExecutorConfig.properties (*TWA_home*\TWS\JavaExt\cfg\J2EEJobExecutorConfig.properties) sur l'agent.

Les mots clés de ce fichier sont décrits dans la table suivante :

Tableau 22. mots clés du fichier J2EEJobExecutorConfig.properties

Mot clé	Spécifie...	Valeur par défaut	Doit être personnalisé
wasjaas.default	Le chemin vers le fichier de configuration IBM WebSphere (wsjaas_client.conf) utilisé pour s'authentifier sur le WebSphere Application Server externe à l'aide de JAAS security.	*TWA_home*/TWS/JavaExt/cfg/wsjaas_client.conf ou *TWA_home*\TWS\JavaExt\cfg\wsjaas_client.conf	Oui (facultatif), si vous déplacez le fichier vers le chemin que vous avez spécifié.
credentials.mycred	Données d'identification (ID et mot de passe) utilisées pour établir la connexion SOAP au serveur WebSphere Application Server externe lorsque vous utilisez la planification indirecte (le mot de passe doit être un mot de passe chiffré {xor})	wasadmin,{xor}KD4sPjsyNjE\= (ID=wasadmin et password=wasadmin au format chiffré {xor})	Oui, voir «Exécution d'un chiffrement {xor} sur votre mot de passe», à la page 81 pour savoir comment chiffrer votre mot de passe.
connector.indirect	Nom du canal de communication avec WebSphere Application Server. La sélection d'un auteur de l'appel indirect signifie que Tivoli Dynamic Workload Broker utilise une infrastructure de planification WebSphere Application Server existante déjà configurée sur une instance WebSphere Application Server externe cible. Lorsque vous créez la définition de travail, vous pouvez spécifier si vous voulez utiliser un connecteur direct ou indirect dans l'écran **Application J2EE** de la page **Application** du Console de définition de courtage de travaux, ou dans l'élément **auteur de l'appel** dans le fichier JSDL. Pour plus d'informations sur la Console de définition de courtage de travaux, voir l'aide en ligne.	Une seule ligne avec les valeurs suivantes séparées par des virgules : • Mot clé indirect • Nom de l'exécuteur : sch/MyScheduler • Mot clé soap • Nom d'hôte de l'instance WebSphere Application Server externe : washost.mydomain.com • Port SOAP de l'instance WebSphere Application Server externe : 8880 • Chemin vers le fichier soap.client.props : *TWA_home*/TWS/JavaExt/cfg/soap.client.props • Mot clé des données d'identification : mycred	Vous devez personnaliser les éléments suivants : • Le nom de l'exécuteur. Remplacez la chaîne sch/MyScheduler par le nom JNDI du planificateur IBM WebSphere que vous voulez utiliser. • Le nom d'hôte de l'instance WebSphere Application Server externe. • Le port SOAP de l'instance WebSphere Application Server externe.

Tableau 22. mots clés du fichier J2EEJobExecutorConfig.properties (suite)

Mot clé	Spécifie...	Valeur par défaut	Doit être personnalisé

connector.direct	Le nom du canal de communication direct sans utiliser le planificateur WebSphere Application Server. Sélectionnez un auteur d'appel direct pour que Tivoli Dynamic Workload Broker transmette immédiatement le travail aux composants de l'instance WebSphere Application Server externe (EJB ou JMS). Lorsque vous créez la définition de travail, vous pouvez spécifier si vous voulez utiliser un connecteur direct ou indirect dans l'écran **Application J2EE** de la page **Application** du Console de définition de courtage de travaux, ou dans l'élément **auteur de l'appel** dans le fichier JSDL. Pour plus d'informations sur la Console de définition de courtage de travaux, voir l'aide en ligne.	Une seule ligne avec les valeurs suivantes séparées par des virgules :
		• Mot clé `direct`
		• La chaîne suivante :
		`com.ibm.websphere.naming.WsnInitialContextFactory`
		• La chaîne suivante :
		`corbaloc:iiop:washost.mydomain.com:2809`
			Vous devez personnaliser les éléments suivants :
			• Le nom d'hôte de l'instance WebSphere Application Server externe : `washost.mydomain.com`.
			• Le port RMI de l'instance WebSphere Application Server externe : `2809`.

trustStore.path	Le chemin d'accès au fichier de clés certifiées de WebSphere Application Server (ce fichier doit être copié à cet emplacement local à partir de l'instance WebSphere Application Server).	*TWA_home*/TWS/JavaExt/cfg/DummyClientTrustFile.jks	Vous pouvez modifier l'emplacement (*TWA_home*/TWS/JavaExt/cfg), si vous copiez le chemin du fichier de clés certifiées de l'instance WebSphere Application Server externe vers ce chemin.
trustStore.password	Mot de passe du fichier de clés certifiées de WebSphere Application Server.	WebAs	Oui

Exécution d'un chiffrement {xor} sur votre mot de passe :
Pour chiffrer votre mot de passe au format {xor}, utilisez la commande `PropFilePasswordEncoder` figurant dans le répertoire *WAS_home*/bin du serveur WebSphere Application Server externe.

Procédez comme suit :

1. Ouvrez un nouveau fichier texte et entrez la ligne suivante :
   chaîne=votre_password_en_texte_brut
2. Sauvegardez le fichier sous le *file_name* de votre choix.
3. Exécutez `PropFilePasswordEncoder` comme suit :
   `PropFilePasswordEncoder` *file_name chaîne*

   Où :

file_name
Est le nom du fichier avec votre mot de passe.

chaîne Est la *chaîne* que vous utilisez dans le fichier texte. Il peut s'agir de tout mot de votre choix, par exemple `password`, `mypwd`, `joe`, etc.

4. Lorsque la commande s'achève, ouvrez à nouveau le fichier texte. Le contenu a été modifié en :

   chaîne`={xor}`*votre_password_chiffré*
5. Copiez votre mot de passe codé, y compris les caractères `{xor}`, et collez-le à l'endroit indiqué dans vos fichiers de propriétés.

Par exemple, vous voulez chiffrer votre mot de passe `catamaran`. Procédez comme suit :

1. Ouvrez un fichier texte et entrez :

   ```
   mypasswd=catamaran
   ```
2. Sauvegardez le fichier sous le nom `encrfile.txt`.
3. Exécutez :

   ```
   PropFilePasswordEncoder encrfile.txt mypasswd
   ```
4. Ouvrez `encrfile.txt`. Vous trouvez :

   ```
   mypasswd={xor}PD4rPjI+LT4x
   ```
5. Copiez `{xor}PD4rPjI+LT4x` et collez cette chaîne à l'endroit de votre choix.

Fichier logging.properties :
Le chemin d'accès à ce fichier est *TWA_home*`/TWS/JavaExt/cfg/logging.properties` (*TWA_home*`\TWS\JavaExt\cfg\logging.properties`) sur l'agent.

Après avoir procédé à l'installation, ce fichier est comme suit :

```
# Spécifiez les gestionnaires à créer dans le consignateur situé à la racine
 # (tous les consignateurs sont des enfants du consignateur situé à la racine)
 # La ligne ci-après crée deux gestionnaires
 handlers = java.util.logging.ConsoleHandler, java.util.logging.FileHandler

 # Définissez le niveau de consignation par défaut pour le consignateur à la racine
 .level = INFO

 # Définissez le niveau de consignation par défaut pour les nouvelles instances ConsoleHandler
 java.util.logging.ConsoleHandler.level = INFO

 # Définissez le niveau de consignation par défaut pour les nouvelles instances FileHandler
 java.util.logging.FileHandler.level = ALL
 java.util.logging.FileHandler.pattern =
  C:\TWA_home\TWS\JavaExt\logs\javaExecutor%g.log
 java.util.logging.FileHandler.limit = 1000000
 java.util.logging.FileHandler.count = 10

 # Définissez le formateur par défaut pour les nouvelles instances ConsoleHandler
 java.util.logging.ConsoleHandler.formatter = java.util.logging.SimpleFormatter
 java.util.logging.FileHandler.formatter = java.util.logging.SimpleFormatter

 # Définissez le niveau de consignation par défaut pour le consignateur nommé com.mycompany
 com.ibm.scheduling = INFO
```

Vous pouvez personnaliser :

- Le niveau de consignation (de INFO à WARNING, ERROR, ou ALL) des mots clés suivants :

  `.level` Définit le niveau de journalisation pour le journal d'événements interne.

com.ibm.scheduling
Définit le niveau de consignation pour les types de travail avec options avancées. Pour journaliser des informations relatives aux types de travail avec options avancées, définissez cette valeur sur ALL.

- Le chemin d'écriture des journaux, spécifié par le mot clé suivant :
  `java.util.logging.FileHandler.pattern`

Fichier soap.client.props :
Le chemin d'accès à ce fichier est *TWA_home*/TWS/JavaExt/cfg/soap.client.props (*TWA_home*\TWS\JavaExt\cfg\soap.client.props) sur l'agent.

Après avoir procédé à l'installation, ce fichier est comme suit :

```
#-----------------------------------------------------------------------------
 # Activation de sécurité du client SOAP
 #
 # - statut de sécurité activé (false[par défaut], true)
 #-----------------------------------------------------------------------------
 com.ibm.SOAP.securityEnabled=false

 com.ibm.SOAP.loginUserid=wasadmin
 com.ibm.SOAP.loginPassword={xor}KD4sPjsyNjE\=

 #-----------------------------------------------------------------------------
 # Invite de connexion SOAP
 #
 # L'invite automatique ne s'effectue que si toutes les conditions suivantes sont remplies :
 #
 # - Exécution à partir d'un client SOAP
 # - Le serveur est accessible et la sécurité serveur est activée
 # - Le nom d'utilisateur et le mot de passe ne sont fournis ni sur la ligne de commande ni
 #   dans ce fichier
 # - com.ibm.SOAP.loginSource ci-dessous est défini sur "stdin" ou "prompt"
 #
 #   stdin : invite dans la fenêtre de commande
 #   prompt : boîte de dialogue de l'interface graphique ; reprend la valeur stdin si
 #   l'interface graphique n'est pas autorisée
 #   (Donc, pour désactiver l'invite automatique, ne définissez aucune valeur pour loginSource)
 #-----------------------------------------------------------------------------
 com.ibm.SOAP.loginSource=prompt

 #-----------------------------------------------------------------------------
 # Délai d'attente des demandes SOAP
 #
 # - délai d'attente (indiqué en secondes [par défaut 180], 0 signifie aucun délai d'attente)
 #
 #-----------------------------------------------------------------------------
 com.ibm.SOAP.requestTimeout=180

 #-----------------------------------------------------------------------------
 # Alias de configuration SSL référencé dans ssl.client.props
 #-----------------------------------------------------------------------------
 com.ibm.ssl.alias=DefaultSSLSettings
```

Si vous voulez activer la sécurité du client SOAP, vous devez :

1. Modifier `com.ibm.SOAP.securityEnabled` sur `true`
2. Personnaliser :
   - `com.ibm.SOAP.loginUserid` avec le véritable ID utilisateur de l'administrateur WebSphere Application Server.
   - `com.ibm.SOAP.loginPassword` avec le mot de passe administrateur WebSphere Application Server au format chiffré {xor}. Voir «Exécution d'un chiffrement {xor} sur votre mot de passe», à la page 81.

Configuration de l'agent J2EE Job Executor

Pour configurer l'environnement sur le serveur WebSphere Application Server externe, version 7.0 pour l'agent J2EE Job Executor, procédez comme suit :

Créez un bus d'intégration de services

1. Ouvrez la console d'administration WebSphere (par exemple, `http://localhost:9060/admin`, selon le port administrateur que vous avez configuré).
2. Développez **Service Integration** et sélectionnez **Buses**. La fenêtre Buses (Bus) s'affiche.
3. Cliquez sur **Nouveau** pour afficher la fenêtre de configuration des bus.
4. Entrez un nom pour le nouveau bus, par exemple **MyBus**, cliquez sur **Suivant** puis sur **Terminer** pour confirmer.
5. Cliquez sur le nom MyBus ; les propriétés MyBus s'affichent.
6. Dans Topology, cliquez sur **Bus Members**. La fenêtre Buses MyBus Bus members s'affiche.
7. Cliquez sur **Ajouter**, sélectionnez le bouton radio **Serveur**, choisissez **<votre_serveur_d'applications>**, cliquez sur **Suivant**, puis cliquez sur **Terminer**.
8. Lorsque le panneau `Confirm the addition of a new bus member` s'affiche, cliquez sur **Terminer**.
9. Sélectionnez **Intégration de service → Bus → MyBus → Destinations → Nouveau**.
10. Sélectionnez le type **File d'attente** puis cliquez sur **Suivant**
11. Entrez **BusQueue** comme identificateur et affectez la file d'attente à un membre de bus. Cliquez sur **Suivant**. Dans le panneau de confirmation, cliquez sur **Terminer**.

Configurez le service de messagerie par défaut

1. Dans le panneau gauche de la console d'administration WebSphere, développez **Ressources→ JMS→ JMS Providers (Fournisseurs JMS)**, puis cliquez sur **Default messaging (Messagerie par défaut)** comme portée au niveau du serveur.
2. Dans la section **Connection Factories (Fabriques de connexions)**, cliquez sur **Nouveau**.
3. Dans la fenêtre New JMS connection factory (Nouvelle fabrique de connexions JMS), renseignez les zones suivantes :

   Nom MyCF

   JNDI Name (Nom JNDI)
   jms/MyCF

   Bus Name (Nom du Bus)
   MyBus

   Points finaux du fournisseur
   <hostname>:<Basic SIB port number>:BootstrapBasicMessaging;<hostname>:<Secure SIB port number>:BootstrapSecureMessaging,

   où <Basic SIB port number> et< Secure SIB port number> se trouvent en développant **Serveurs**, en sélectionnant **<your_application_server_name>**, puis en sélectionnant **Messaging**

engine inbound transports (Transports entrants de moteur de messagerie) sous **Server Messaging (Messagerie serveur)**.

4. Sélectionnez à nouveau **Ressources -→ JMS-→ JMS Providers (Fournisseurs JMS) → Default Messaging (Messagerie par défaut)** comme portée au niveau du serveur, localisez la section **Destinations** et cliquez sur **Queues (Files d'attente)**. Cliquez sur **New (Nouveau)** et renseignez les zones suivantes comme indiqué :
   Name=MyQueue
   JNDI Name=jms/MyQueue
   Bus name=MyBus
   Queue Name=BusQueue

   Cliquez sur **Ok**.
5. Sélectionnez à nouveau **Ressources -→ JMS-→ JMS Providers (Fournisseurs JMS) → Default Messaging (Messagerie par défaut)** comme portée au niveau du serveur, et localisez la section **Activation Specifications (Spécifications d'activation)**.
6. Cliquez sur **JMS activation specification**. Cliquez sur **New (Nouveau)** et renseignez les zones suivantes comme indiqué :
   Name=MySpecAct
   JNDI Name=eis/MyActSpec
   Bus name=MyBus
   Destination Type=Queue
   Destination JNDI name=jms/MyQueue

   Cliquez sur **Ok**.

Configurez la sécurité Java

1. Sélectionnez **Security (Sécurité) → Secure Administration, applications and infrastructure (Sécuriser l'administration, les applications et l'infrastructure)**.
2. Localisez la section **Authentication (Authentification)**, développez **Java Authentication and Authorization Service (Service JAAS)** et cliquez sur **J2C authentication data (Données d'authentification J2C)**.
3. Cliquez sur **New (Nouveau)** et renseignez les zones suivantes comme indiqué :
   Alias=*usr*
   User ID=*usr*
   Password=*pwd*

   où *utr* est l'ID utilisateur authentifié lorsque vous utilisez la sécurité du connecteur et *pwd* est le mot de passe associé.
4. Cliquez sur **Ok**.

Créez une source de données XA

1. Dans le panneau de gauche, allez dans **Ressources → JDBC..→ JDBCProviders (Fournisseurs JDBC)**. Dans le panneau qui s'affiche à droite, vérifiez que la portée pointe vers **<votre_serveur_d'applications>**.
2. Localisez l'entrée **DERBY JDBC Provider (XA)** et cliquez dessus.
3. Recherchez la section **Additional Properties (Propriétés complémentaires)** et cliquez sur **Data Sources (Sources de données)**.
4. Cliquez sur **New (Nouveau)** et renseignez les zones suivantes comme indiqué :
   Name=MyScheduler XA DataSource
   JNDI Name=jdbc/SchedulerXADS
   Database name=${USER_INSTALL_ROOT}/databases/Schedulers/
   ${SERVER}/SchedulerDB;create=true

5. En haut de la page, cliquez sur **Test connection button (Bouton Tester la connexion)**.
6. Même si vous obtenez un résultat négatif, modifiez la zone **Nom de la base de données** en supprimant la partie `;create=true`. Cliquez sur **Ok**.

Créer un gestionnaire de travaux

1. Dans le panneau de gauche, allez dans **Ressources → Asynchronous beans (Beans asynchrones)→ Work managers (Gestionnaires de travail)** et cliquez sur **Nouveau**.
2. Renseignez les zones suivantes comme indiqué :
   Name=SchedulerWM
   JNDI Name=wm/SchedulerWM
3. Cliquez sur **Ok**.

Créez et configurez un planificateur

1. Dans le panneau de gauche, allez dans **Ressources → Schedulers (Planificateurs)** et cliquez sur **Nouveau**.
2. Renseignez les zones suivantes comme indiqué :
   Name=MyScheduler
   JNDI name=sch/MyScheduler
   Data source JNDI name=jdbc/SchedulerXADS
   Table prefix=MYSCHED
   Work managers JNDI name=wm/SchedulerWM
3. Cliquez sur **Ok**.
4. Sélectionnez **MyScheduler** et cliquez sur **Create tables**.
5. Déployez l'application de test.

Ordre de priorité de sécurité utilisé pour l'exécution des tâches J2EE

Il existe trois façons de vérifier qu'une tâche s'exécute avec les données d'identification correctes de l'utilisateur. Les tâches s'exécutent avec les données d'identification de sécurité spécifiées, via les méthodes suivantes :

1. Contexte de sécurité Java Authentication and Authorization Service (JAAS) sur l'unité d'exécution où la tâche a été créée.
2. Méthode `setAuthenticationAlias` sur l'objet `TaskInfo`.
3. Une identité de sécurité spécifiée sur une méthode EJB `TaskHandler` de tâche `BeanTaskInfo`.

Les méthodes d'authentification sont exécutées dans l'ordre indiqué ci-dessus, de telle façon que si une méthode réussit, les autres contrôles sont ignorés. Cela signifie que les données d'identification *usr* et *pwd* définies dans **Configurer la sécurité** prévalent sur les éventuelles données d'identification fournies dans les tâches elles-mêmes.

Configuration pour planifier des types de travail avec options avancées

Parallèlement à la définition des types de travail avec options avancées avec Dynamic Workload Console ou la commande **composer**, vous pouvez utiliser les fichiers de configuration associés. Les options que vous définissez dans le fichier de configuration s'appliquent à tous les types de travail avec options avancées du même type. Vous pouvez substituer ces options lorsque vous définissez le travail à l'aide de Dynamic Workload Console ou de la commande **composer**.

Les fichiers de configuration sont disponibles sur chaque agent dynamique dans *TWA_home*/TWS/JavaExt/cfg pour les types de travail avec options avancées suivants :

Tableau 23. Fichiers de configuration pour les types de travail avec options avancées

Type de travail	Nom de fichier	Mot clé
• Type de travail de base de données • Travail MSSQL	DatabaseJobExecutor.properties	Utilisez le mot clé `jdbcDriversPath` pour spécifier le chemin d'accès aux pilotes JDBC. Définissez le mot de passe de sorte qu'il pointe vers le répertoire des fichiers jar JDBC, par exemple : `jdbcDriversPath=c:\\mydir\\jars\\jdbc` Les fichiers jar JDBC doivent se trouver dans le répertoire spécifié ou dans ses sous-répertoires. Assurez-vous que vous disposez des autorisations de liste pour le répertoire et ses sous-répertoires. **Remarque :** Pour la base de données MSSQL, utilisez la version 4 des pilotes JDBC.
Type de travail Java	JavaJobExecutor.properties	Utilisez le mot clé `jarPath` pour spécifier le chemin d'accès au répertoire dans lequel les fichiers jar sont stockés. Sont compris tous les fichiers jar stockés dans le répertoire spécifié et tous les sous-répertoires.
Type de travail J2EE	J2EEJobExecutorConfig.properties	Pour plus d'informations sur le type de travail J2EE, voir Configuration pour planifier des travaux J2EE.

Configuration des rôles de sécurité pour les utilisateurs et les groupes

Pendant l'installation de Dynamic Workload Console, de nouveaux rôles et groupes prédéfinis sont créés dans Integrated Solutions Console. Ces rôles permettent de déterminer les fenêtres Dynamic Workload Console disponibles pour l'utilisateur, et donc les activités que ce dernier peut exécuter dans Dynamic Workload Console. Si vous n'attribuez pas de rôle à un utilisateur Integrated Solutions Console, aucune entrée ne s'affiche pour Dynamic Workload Broker dans l'arborescence de navigation lorsque l'utilisateur se connecte. L'accès aux entrées de l'arborescence de navigation ne signifie pas que l'utilisateur peut accéder aux fonctions de produit. Il existe un deuxième niveau d'autorisation, déterminé par un ensemble de rôles de sécurité créés au moment de l'installation du gestionnaire de domaine maître dans WebSphere Application Server. Ces rôles définissent les niveaux d'autorisation requis pour exécuter les fonctions de produit, quelle que soit l'interface utilisée.

Par conséquent, vous devez mapper les utilisateurs et les rôles dans WebSphere Application Server afin de les associer à ceux définis dans Integrated Solutions Console et d'établir la communication entre Dynamic Workload Console et l'instance Dynamic Workload Broker. Cette procédure est décrite dans «Mappage de rôles de sécurité vers des utilisateurs et groupes dans WebSphere Application Server»

Mappage de rôles de sécurité vers des utilisateurs et groupes dans WebSphere Application Server

Lorsque l'instance Dynamic Workload Broker est installée sur votre gestionnaire de domaine maître, les rôles correspondants sont configurés dans WebSphere

Application Server. Par défaut, ces rôles ne sont pas utilisés. Cependant, si vous activez la sécurité globale dans votre environnement, l'autorisation requise pour effectuer toute tâche est toujours validée par WebSphere Application Server. Les utilisateurs doivent fournir leurs données d'identification pour accéder aux tâches de planification dynamique. Ces données d'identification correspondent aux utilisateurs existants définis dans le registre d'utilisateurs du domaine ou sur le serveur LDAP.

Pour permettre aux utilisateurs et aux groupes d'accéder aux fonctions Dynamic Workload Broker lorsque la sécurité globale est activée, ils doivent être mappés aux rôles de sécurité dans WebSphere Application Server. Ce mappage permet aux utilisateurs et groupes d'accéder aux applications définies pour le rôle. Au moment de l'installation, les rôles suivants sont créés dans WebSphere Application Server:

Opérateur
Surveille et contrôle les travaux soumis.

Administrateur
Gère l'infrastructure de planification.

Développeur
Définit les travaux à exécuter en spécifiant les paramètres du travail, les besoins en ressources, etc.

Emetteur
Gère la soumission de ses propres travaux ; surveille et contrôle le cycle de vie des travaux. Il s'agit du rôle classique d'un utilisateur Tivoli Workload Scheduler.

Tivoli Workload Scheduler agit comme un émetteur de travaux sur l'agent dynamique de Tivoli Workload Scheduler.

Configurateur
Responsable de l'exécution des travaux dans un environnement local.

Pour mapper les rôles de sécurité aux utilisateurs et groupes dans WebSphere Application Server, vous devez modifier le fichier **BrokerSecurityProps.properties** à l'aide du script **changeBrokerSecurityProperties**.

Pour éviter de modifier une valeur de configuration par inadvertance, ou d'annuler les dernières modifications, vous devez d'abord créer un fichier contenant les propriétés actuelles, le modifier en définissant les valeurs requises, et appliquer les modifications. Procédez comme suit :

1. Connectez-vous à l'ordinateur sur lequel Tivoli Workload Scheduler est installé en tant que l'utilisateur suivant :

   UNIX root

   Windows
   Tout utilisateur du groupe *Administrators*.
2. Accédez au répertoire : *<TWA_home>*/wastools
3. Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)
4. A partir de ce répertoire, exécutez le script suivant pour créer un fichier contenant les propriétés de sécurité du courtier actuel :

   UNIX **showBrokerSecurityProperties.sh > mon_fichier**

Windows

```
showBrokerSecurityProperties.bat > mon_fichier
```

5. Editez le fichier `mon_fichier` avec un éditeur de texte.
6. Modifiez les propriétés en fonction de vos besoins. Pour chacun des rôles du fichier, vous pouvez définir les propriétés suivantes :

   Everyone? (Tous ?)
   Valeurs possibles :
   - **Oui** : tous les utilisateurs sont autorisés à effectuer des tâches pour le rôle. Aucun contrôle n'est effectué sur le registre d'utilisateur WebSphere Application Server.
   - **Non** : l'accès est refusé aux utilisateurs non définis dans le registre d'utilisateurs WebSphere Application Server.

   All authenticated? (Tous les utilisateurs authentifiés ?)
   Valeurs possibles :
   - **Oui** : tous les utilisateurs appartenant au registre d'utilisateurs WebSphere Application Server actuel et qui ont été authentifiés peuvent accéder aux ressources et aux tâches du rôle. Il s'agit de l'option par défaut.
   - **Non** : l'accès est autorisé uniquement aux utilisateurs et aux groupes définis dans le registre d'utilisateurs WebSphere Application Server et répertoriés dans les propriétés d'utilisateur mappé et de groupe mappé.

   Utilisateurs mappés
   le cas échéant, un ou plusieurs utilisateurs séparés par une barre verticale (`|`). Cette zone peut être laissée vide.

   Groupes mappés
   la cas échéant, un ou plusieurs groupes séparés par une barre verticale (`|`). Cette zone peut être laissée vide.
7. Enregistrez le fichier *mon_fichier*.
8. Exécutez le script :

   Windows

   ```
   changeBrokerSecurityProperties.bat mon_fichier
   ```

   UNIX

   ```
   changeBrokerSecurityProperties.sh mon_fichier
   ```

   où *mon_fichier* est le *chemin qualifié complet* du fichier contenant les nouveaux paramètres.

   Les propriétés sont mises à jour en fonction des règles indiquées dans les descriptions de chaque type de propriété.
9. Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)
10. Assurez-vous que le changement a été implémenté.

Remarque :

1. Si le nom de l'utilisateur ou du groupe mappé contient des espaces vides, la liste complète des utilisateurs et des groupes doit être spécifiée entre guillemets ("). Par exemple, si vous voulez ajouter les utilisateurs John Smith, MaryWhite et DavidC au rôle de développeur, spécifiez-les comme suit :

```
Role: Developer
Everyone?: No
All authenticated?: No
Mapped users:"John Smith|MaryWhite|DavidC"
Mapped groups:
```

2. Dans le fichier se trouve un rôle par défaut supplémentaire nommé **WSClient**, que vous devez laisser tel quel.

Exemples : Pour attribuer le rôle Opérateur aux utilisateurs Susanna et Ann, appartenant au registre d'utilisateur WebSphere Application Server actuel :

```
Role: Operator
Everyone?: No
All authenticated?: No
Mapped users:Susanna|Ann
Mapped groups:
```

Pour attribuer le rôle Administrateur à l'utilisateur Tom et le rôle Développeur au groupe d'utilisateurs MyGroup définis dans le registre d'utilisateurs WebSphere Application Server actuel :

```
Role: Administrator
Everyone?: No
All authenticated?: No
Mapped users:Tom
Mapped groups:

Role: Developer
Everyone?: No
All authenticated?: No
Mapped users:
Mapped groups:MyGroup
```

Fichier de propriétés BrokerSecurityProps.properties

```
#################################################################
# Broker Security Properties
#################################################################


Role: WSClient
Everyone?: No
All authenticated?: Yes
Mapped users:
Mapped groups:

Role: Administrator
Everyone?: No
All authenticated?: Yes
Mapped users:
Mapped groups:

Role: Operator
Everyone?: No
All authenticated?: Yes
Mapped users:
Mapped groups:

Role: Submitter
Everyone?: No
All authenticated?: Yes
Mapped users:
Mapped groups:

Role: Configurator
Everyone?: No
All authenticated?: Yes
```

```
Mapped users:
Mapped groups:

Role: Developer
Everyone?: No
All authenticated?: Yes
Mapped users:
Mapped groups:
```

Configuration de l'authentification pour l'accès au client de ligne de commande

Cette section décrit la procédure à suivre pour reconfigurer la connexion utilisée par le client de ligne de commande.

Le client de ligne de commande est installé automatiquement sur le gestionnaire de domaine maître et peut être installé en option sur tout autre poste de travail. Sur le gestionnaire de domaine maître vous l'utilisez pour exécuter toutes les commandes et les utilitaires.

Sur tout autre poste de travail, vous l'utilisez pour exécuter l'une des commandes suivantes :

- **evtdef**
- **composer**
- **optman**
- **planman**
- **sendevent**

Il est configuré automatiquement par l'assistant d'installation, mais si vous devez modifier les données d'identification qui donnent accès au serveur sur le gestionnaire de domaine maître, ou que vous voulez l'utiliser pour accéder à un différent gestionnaire de domaine maître, modifiez les ***paramètres de connexion*** comme décrit ici.

Remarque :

1. Les ***paramètres de connexion*** ne sont pas nécessaires pour utiliser le programme local **conman** sur un agent tolérant aux pannes.
2. Le client de ligne de commande du gestionnaire de domaine maître utilise exactement le même mécanisme pour communiquer avec le serveur que celui utilisé s'il est installé à distance.

Paramètres de connexion

Les paramètres de connexion peuvent être fournis à l'aide d'une des trois méthodes suivantes :

Définissez-les dans le fichier localopts

Toutes les zones à l'exception de *username* et *password*, peuvent être définies en éditant le fichier de propriétés *TWA_home*/TWS/**localopts** sur l'ordinateur à partir duquel l'accès est requis. Voir «Définition des options locales», à la page 30 pour obtenir une description complète du fichier et des propriétés.

Dans le fichier **localopts**, il y a une section consacrée aux propriétés générales de connexion, qui contient les éléments suivants :

host = *host_name*
protocol = *protocole*
port = *numéro de port*

```
proxy = serveur proxy
proxyport = numéro de port du serveur proxy
timeout = seconds
defaultws = master_workstation
useropts = fichier_useropts
```

De plus, il existe des groupes distincts de paramètres SSL qui varient en fonction de la conformité de votre réseau aux normes FIPS, et qui utilisent GSKit pour SSL, ou qui, dans le cas contraire, utilisent OpenSSL (pour plus de détails, voir «Conformité aux normes FIPS», à la page 300) :

Conformité aux normes FIPS (GSKit)

```
CLI SSL keystore file = keystore_file_name
CLI SSL certificate keystore label = label
CLI SSL keystore pwd = password_file_name
```

Non conformité aux normes FIPS (OpenSSL)

```
CLI SSL server auth = yes|no
CLI SSL cipher = cipher_class
CLI SSL server certificate =file_name_certificat
CLI SSL trusted dir =trusted_directory
```

Stockage intégral ou partiel dans le fichier useropts

Au minimum, les paramètres **username** et **password** peuvent être définis dans le fichier *user_home*/.TWS/**useropts** pour l'utilisateur ayant besoin d'établir la connexion. De même, si vous devez personnaliser l'une des propriétés qui se trouvent habituellement dans le fichier `localopts` pour un utilisateur, ajoutez les propriétés au fichier `useropts`. Les valeurs présentes dans le fichier `useropts` sont prioritaires par rapport à celles du fichier `localopts`. Voir «Définition des options d'utilisateur», à la page 51 pour obtenir une description complète du fichier et des propriétés.

L'ensemble minimal de propriétés que vous trouverez dans le fichier **useropts** est le suivant :

```
username=user_ID
password=password
```

Fournissez-les lorsque vous utilisez la commande

Lorsque vous utilisez une commande, vous pouvez ajouter un ou plusieurs paramètres de connexion à la chaîne de commande. Ces paramètres sont prioritaires par rapport aux paramètres des fichiers **localopts** et **useropts**. Cela vous permet notamment de conserver ces paramètres dans le fichier **localopts** et de demander simplement aux utilisateurs de fournir les paramètres **username** et **password** lorsqu'ils utilisent l'une des commandes, évitant ainsi d'avoir à stocker ces données dans le fichier **useropts** pour chaque utilisateur.

Les paramètres peuvent être fournis dans leur totalité ou en partie dans un fichier, auquel vous vous référez dans la chaîne de commande ou entrés directement dans la chaîne de commande. Voici la syntaxe complète :

```
[-file <fichier_paramètre>
|
 [-host <host_name>]
 [-password <user_password>]
 [-port <port_number>]
 [-protocol {http|https}]
 [-proxy <nom_proxy>]
 [-proxyport <proxy_port_number>]
 [-timeout <dépassement_délai>]
 [-username <username>]
```

-file <*fichier_paramètre*>
Fichier contenant un ou plusieurs paramètres de connexion. Les paramètres de ce fichier sont obsolètes si le paramètre correspondant est entré dans la commande de manière explicite.

-host <*host_name*>
Nom d'hôte ou adresse IP du gestionnaire de domaine maître auquel vous voulez vous connecter.

-password <*user_password*>
Mot de passe de l'utilisateur fourni dans le paramètre `-username`.

-port <*port_number*>
Port d'écoute du gestionnaire de domaine maître auquel vous voulez vous connecter.

-protocol {http|https}
Entrez http ou https, selon que vous voulez établir une connexion sécurisée ou non.

-proxy <*nom_proxy*>
Nom d'hôte ou adresse IP du serveur proxy impliqué dans la connexion (le cas échéant).

-proxyport <*proxy_port_number*>
Port d'écoute du serveur proxy impliqué dans la connexion (le cas échéant).

-timeout <*dépassement_délai*>
Nombre de secondes pendant lesquelles le client de ligne de commande doit attendre pour établir la connexion avant d'émettre une erreur de dépassement du délai d'attente.

-username <*username*>
ID de l'utilisateur qui établit la connexion.

Remarque : A partir de la ligne de commande, vous ne pouvez pas indiquer le poste de travail par défaut ou les paramètres SSL du client de ligne de commande. Ces paramètres doivent être toujours fournis dans le fichier `localopts` (voir «Définition des options locales», à la page 30) ou dans le fichier `useropts` de l'utilisateur (voir «Définition des options d'utilisateur», à la page 51).

Le client de ligne de commande doit regrouper un ensemble complet de paramètres, comme suit :

1. Il commence par rechercher les valeurs fournies en tant que paramètres dans la commande
2. Puis, pour les paramètres qui lui manquent, il recherche les paramètres fournis dans le fichier identifié par le paramètre `-file`
3. Ensuite, il recherche les paramètres qui lui manquent dans le fichier `useropts` de l'utilisateur
4. Enfin, s'il lui manque encore des paramètres, il les recherche dans le fichier `localopts`

Si la valeur d'un paramètre n'est pas indiquée à l'un de ces emplacements, une erreur s'affiche.

Saisie de mots de passe

La sécurité du mot de passe est traitée comme suit :

Mot de passe entré dans le fichier useropts
Vous saisissez le mot de passe de connexion dans le fichier `useropts` sous forme chiffrée. La première fois que vous accédez à l'interface, elle est chiffrée. Il s'agit de la méthode recommandée.

Mot de passe saisi dans le fichier de paramètres utilisé par la commande
Vous saisissez le mot de passe de connexion dans le fichier de paramètres sous forme non chiffrée. Il n'est pas chiffré avec la commande. Supprimez le fichier après emploi pour assurer la sécurité du mot de passe.

Mot de passe entré à l'aide du paramètre -password dans la commande
Vous saisissez le mot de passe dans la chaîne de commande sous forme non chiffrée. Il reste visible dans la fenêtre de commande jusqu'à ce que vous en effaciez le contenu.

Remarque : Sur les postes de travail Windows, lorsque vous spécifiez un mot de passe contenant des guillemets (") ou tout autre caractère spécial, assurez-vous qu'une séquence d'échappement est définie pour ces caractères. Si par exemple le mot de passe utilisé est `tws11"tws`, spécifiez-le sous la forme `"tws11\"tws"` dans **useropts**.

Invites et messages d'écran de Tivoli Workload Scheduler

Les processus de contrôle de Tivoli Workload Scheduler (Netman, Mailman, Batchman, Jobman et Writer) écrivent leurs messages de statut (appelés messages de console) dans les fichiers de liste standard. Ces messages comprennent les invites utilisées en tant que travaux et dépendances de Planificateur de travaux. Sur les systèmes d'exploitation UNIX et Linux, les messages peuvent également être dirigés vers le daemon**syslog** (**syslogd**) et vers un terminal exécutant le gestionnaire de console Tivoli Workload Scheduler. Ces fonctions sont décrites dans les sections suivantes.

Définition de sysloglocal sous UNIX

Si vous associez **sysloglocal** à une valeur positive dans le fichier d'options locales, les processus de contrôle de Tivoli Workload Scheduler envoient leurs messages d'écran et leurs messages d'invite au daemon **syslog**. La définition de la valeur **-1** pour cette option désactive cette fonction. Si vous affectez un nombre positif à cette option pour activer la journalisation système, vous devez également affecter la valeur **0** ou un nombre négatif à l'option locale **stdlistwidth**.

Les messages d'écran de Tivoli Workload Scheduler correspondent aux niveaux **syslog** suivants :

LOG_ERR
Les messages d'erreur, tels que les arrêts anormaux des processus de contrôle et les erreurs de système de fichier.

LOG_WARNING
Les messages d'avertissement telles que les erreurs de liaison et les flots de travaux bloqués.

LOG_NOTICE
Les messages spéciaux, tels que les invites et les tellops.

LOG_INFO
Les messages d'information tels que les lancements de travaux et les modifications d'état des Planificateur de travaux.

La définition d'un nombre positif pour l'option **sysloglocal** détermine l'utilitaire syslog utilisé par Tivoli Workload Scheduler. Par exemple, une valeur de **4** indique à Tivoli Workload Scheduler d'utiliser l'utilitaire local LOCAL4. Après avoir défini cette valeur, vous devez effectuer les entrées appropriées dans le fichier **/etc/syslog.conf** et reconfigurer le daemon syslog. Pour utiliser LOCAL4 et pour que les messages Tivoli Workload Scheduler soient envoyés à la console système, entrez la ligne suivante dans **/etc/syslog.conf** :

```
local4    /dev/console
```

Pour que les messages d'erreur Tivoli Workload Scheduler soient envoyés aux utilisateurs **maestro** et **root**, entrez la commande suivante :

```
local4.err    maestro,root
```

Les zones de sélecteur et d'action doivent être séparées par au moins une tabulation. Après avoir modifié **/etc/syslog.conf**, vous pouvez configurer le daemon **syslog** en entrant la commande suivante :

```
kill -HUP `cat /etc/syslog.pid`
```

Commande console

Vous pouvez utiliser la commande **console** de conman pour définir le niveau de message de Tivoli Workload Scheduler et pour acheminer les messages jusqu'à votre terminal. La définition du niveau de message concerne uniquement les messages Batchman et Mailman, lesquels sont les plus nombreux. Elle définit également le niveau des messages écrits dans le ou les fichiers de liste standard et dans le daemon **syslog**. Par exemple, la commande suivante définit le niveau des messages Batchman et Mailman à **2** et envoie les messages à votre ordinateur :

```
console sess;level=2
```

Les messages sont envoyés à votre ordinateur jusqu'à ce que vous exécutiez une autre commande **console** ou que vous quittiez conman. Pour cesser d'envoyer des messages à votre terminal, entrez la commande conman suivante :

```
console sys
```

Activation de la fonction de fuseau horaire

Les fuseaux horaires sont activés par défaut lors de l'installation du produit.

Lorsque vous procédez à une mise à jour, la fonction fuseau horaire hérite les paramètres de l'installation précédente. Vous pouvez activer le fuseau horaire à l'aide de l'option **enTimeZone** de la commande **optman**, comme suit :

```
optman chg enTimeZone = yes
```

La procédure suivante permet d'implémenter le fonction de fuseau horaire:

1. Chargez Tivoli Workload Scheduler.

   La base de données permet de spécifier les fuseaux horaires pour les postes de travail, mais pas pour les heures de *début* et d'*échéance* dans flots de travaux dans la base de données. La création de plan (JnextPlan) ignore les éventuels fuseaux horaires qui sont présents dans la base de données. Vous ne pourrez pas spécifier de fuseaux horaires dans le plan.
2. Définissez les fuseaux horaires des postes de travail.

   Définissez le fuseau horaire du poste de travail gestionnaire de domaine maître, du gestionnaire de domaine maître de secours et des éventuels agents qui se trouvent dans un fuseau horaire différent du gestionnaire de domaine maître.

Dans la base de données, les zones **Début**, **Dernière heure de début** et **Echéance de fin** n'acceptent pas les fuseaux horaires. Aucun fuseau horaire n'est autorisé dans le plan à ce stade, car **enTimeZone** est défini sur **no**.

3. Une fois que les fuseaux horaires des postes de travail sont définis correctement, activez la fonction de fuseau horaire.

   Tous les utilisateurs peuvent utiliser les fuseaux horaires n'importe où dans la base de données, mais ils devraient attendre la prochaine exécution de JnextPlan pour les utiliser pour **Début**, **Dernière heure de début** et **Echéance de fin**. Lors de la prochaine exécution de JnextPlan, les fuseaux horaires sont transmis au plan et à Dynamic Workload Console, et le système dorsal autorise la spécification de fuseaux horaires n'importe où dans le plan.

4. Commencez à utiliser des fuseaux horaires sur les heures de *début* et les *échéances* lorsque vous en avez besoin.

   Vous pouvez désormais utiliser toutes les références de fuseau horaire dans la base de données et dans le plan avec Dynamic Workload Console et l'interface de ligne de commande.

Configuration de l'utilisation des commandes de rapport

Vous utilisez les commandes de rapport de Tivoli Workload Scheduler pour obtenir des informations résumées ou détaillées à propos de la planification de votre charge de travail. Avant d'utiliser ces commandes, elles doivent cependant être configurées pour votre environnement. Ce processus est décrit dans le chapitre consacré à l'obtention des rapports et des statistiques dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Modification des droits du service jobmon pour Windows

Sur les systèmes d'exploitation Windows, le service Tivoli Workload Scheduler Jobmon s'exécute sous le compte SYSTEM qui dispose du droit **Autoriser le service à interagir avec le bureau**. Vous pouvez supprimer ce droit pour des raisons de sécurité. Toutefois, cette suppression empêche de lancer des travaux interactifs s'exécutant dans une fenêtre sur le bureau de l'utilisateur. Ces travaux seront exécutés, mais ne sont pas accessibles à partir du bureau ou depuis Tivoli Workload Scheduler et n'ont pas accès aux ressources du bureau. En conséquence, ils peuvent s'exécuter en boucle ou s'arrêter de façon anormale en raison du manque de ressources.

Chapitre 3. Configuration du Dynamic Workload Console

Le présent chapitre explique comment configurer Dynamic Workload Console. Il comprend les sections suivantes :

Lancement en contexte avec Dynamic Workload Console

Le présent chapitre explique comment créer une adresse URL pour lancer Dynamic Workload Console et lui faire directement ouvrir les résultats d'une requête spécifique.

Vous pouvez ensuite inclure cette URL dans une application externe, comme par exemple pour surveiller des travaux et des flots de travaux essentiels pour votre activité, et pour les gérer facilement et rapidement. Vous pouvez accéder aux détails d'un travail ou d'un flot de travail sans avoir à créer de requêtes personnalisées ; vous pouvez également surveiller l'état et la santé des postes de travail essentiels dans votre environnement de sorte que, quand une indisponibilité ou un dysfonctionnement affecte la planification des travaux, vous en êtes averti.

Scénarios

Les scénarios principaux ci-après peuvent être identifiés :

- Obtenir le résultat d'une tâche de contrôle sur les :
  - Travaux
  - Travaux critiques
  - Flots de travaux
- Obtenir le résultat d'une tâche de contrôle sur les postes de travail
- Obtenir le résultat d'une tâche sauvegardée.

Pour tous les scénarios, vous devez créer une adresse URL basique comme décrit dans «Création d'une adresse URL basique».

Création d'une adresse URL basique

Pour créer une adresse URL basique, procédez comme suit :

1. Définissez l'adresse URL pour accéder à Dynamic Workload Console:

```
https://{WebUIHostname:adminSecurePort}
/racine_contexte_DASH/xLaunch.do?pageID=com.ibm.tws.
WebUI.External.navigation&showNavArea=false
```

où :

WebUIHostname
Nom d'hôte complet ou adresse IP de l'ordinateur où est installée Dynamic Workload Console.

adminSecurePort
Numéro du port d'écoute port de Dynamic Workload Console.

racine_contexte_DASH
Il s'agit de la racine de contexte Dashboard Application Services Hub définie lors de l'installation. La racine de contexte détermine l'URL d'une application déployée et, par défaut, est identique au répertoire d'application ou à la structure d'archive. Dans ce cas, la valeur par défaut est `ibm/console`.

Exemple

```
https://mypc:29443/ibm/console/xLaunch.do?pageID=com.ibm.tws.WebUI.
External.navigation&showNavArea=false
```

2. Spécifiez l'action que vous voulez exécuter en indiquant le paramètre correspondant :

   &action
   Elle indique l'action que vous voulez effectuer et peut prendre une des valeurs ci-après :
   - BrowseJobs
   - ZBrowseJobs
   - BrowseJobStreams
   - BrowseCriticalJobs
   - BrowseWorkstation
   - InternalTask

3. Indiquez le moteur sur lequel vous souhaitez exécuter la requête en entrant ses paramètres :

   &hostname
   Pour les environnements distribués, il s'agit du nom d'hôte ou de l'adresse TCP/IP de l'ordinateur sur lequel le moteur Tivoli Workload Scheduler est installé. Pour les environnements z/OS, il s'agit du nom d'hôte ou de l'adresse TCP/IP de l'ordinateur sur lequel le connecteur z/OS est installé.

   &port Le numéro de port utilisé pour se connecter à l'ordinateur sur lequel le moteur Tivoli Workload Scheduler ou le connecteur z/OS est installé. Généralement, les numéros de port par défaut sont :

Tableau 24. Numéros de port par défaut

Numéro de port	Moteur
31117	Moteur distribué Tivoli Workload Scheduler
31127	Tivoli Workload Scheduler pour le moteur z/OS avec z/OS connector V.8.3
31217	Tivoli Workload Scheduler pour le moteur z/OS avec connecteur z/OS V.8.5 ou supérieur

Tableau 24. Numéros de port par défaut (suite)

Numéro de port	Moteur
2809	Tivoli Workload Scheduler for z/OS avec connecteur z/OS on z/OS WebSphere Application Server
16312	Tivoli Workload Scheduler for z/OS engine V. 9.1

&server

S'applique uniquement aux systèmes z/OS et est obligatoire. Il s'agit du nom du serveur distant du moteur tel qu'il est spécifié dans le connecteur z/OS.

Exemple

```
&hostname = webuidev&port = 31217&server = C851
```

Exemple d'URL complète :

```
https://mypc:29443/ibm/console/xLaunch.do?pageID=
com.ibm.tws.WebUI.External.navigation&showNavArea=false
&action=BrowseJobs&hostname=webuidev&port=31117
```

Paramètres facultatifs avancés

En fonction de la requête dont vous voulez voir les résultats, vous pouvez renseigner l'adresse URL avec les paramètres suivants.

Surveillance des travaux sur les systèmes répartis

Créez une adresse URL en spécifiant l'action **BrowseJobs**, comme décrit dans «Création d'une adresse URL basique», à la page 97.

Vous pouvez également spécifier un des filtres ci-après :

&workstation
Filtre par le poste de travail sur lequel le travail s'exécute

&jobstream
Filtre par le flot de travaux contenant les travaux.

&job Filtre par le nom du travail.

&schedtime
Filtre par l'heure de planification du travail.

&état Filtre par le statut du travail. Vous pouvez filtrer avec un ou plusieurs statuts. Les valeurs possibles sont les suivantes :

- **W** En attente
- **O** Terminé
- **H** Suspendu
- **R** Prêt
- **E** Erreur
- **U** Indéterminé
- **S** En cours d'exécution
- **C** Annulé
- **B** Bloqué

&colonnes
Spécifiez le nombre de résultats que vous voulez afficher dans votre tableau de résultats. S'il n'est pas spécifié, le nombre de colonnes par défaut de cette requête s'affiche. Les valeurs prises en charge sont :

- **Min** Affiche une ensemble minimum de colonnes
- **All** Affiche toutes les colonnes

Exemple :

```
https://mypc:29043/racine_contexte_DASH/xLaunch.do?pageID=com.ibm.tws.WebUI.
External.navigation&showNavArea=false&action=BrowseJobs
&hostname=webuidev&port=31117
&workstation=my_ws&jobstream=my_js_name&job=my_job_name&status=ESB&columns=ALL
```

où :

racine_contexte_DASH
Il s'agit de la racine de contexte Dashboard Application Services Hub définie lors de l'installation. La racine de contexte détermine l'URL d'une application déployée et, par défaut, est identique au répertoire d'application ou à la structure d'archive. Dans ce cas, la valeur par défaut est `ibm/console`.

Surveillance des travaux sur les systèmes z/OS

Créez une adresse URL en spécifiant l'action **ZBrowseJobs**, comme décrit dans «Création d'une adresse URL basique», à la page 97.

Vous pouvez également spécifier un des filtres ci-après :

&workstation
Filtre par le poste de travail sur lequel le travail s'exécute

&jobstream
Filtre par le flot de travaux contenant les travaux.

&job Filtre par le nom du travail.

&schedtime
Filtre par l'heure de planification du travail.

&colonnes
Spécifiez le nombre de résultats que vous voulez afficher dans votre tableau de résultats. S'il n'est pas spécifié, le nombre de colonnes par défaut de cette requête s'affiche. Les valeurs prises en charge sont :

Min Affiche une ensemble minimum de colonnes

Toutes Affiche toutes les colonnes

Exemple :

```
https://mypc:29043/racine_contexte_DASH/xLaunch.do?pageID=com.ibm.tws.WebUI.
External.navigation&showNavArea=false&action=ZBrowseJobs
&hostname=webuidev&port=31117
&server=C851&workstation=mon_ws&jobstream=my_js_name
&job=my_job_name&schedtime=200812081100
```

où :

racine_contexte_DASH
Il s'agit de la racine de contexte Dashboard Application Services Hub définie lors de l'installation. La racine de contexte détermine l'URL d'une application déployée et, par défaut, est identique au répertoire d'application ou à la structure d'archive. Dans ce cas, la valeur par défaut est `ibm/console`.

Contrôle des travaux critiques

Créez une adresse URL en spécifiant l'action **BrowseCriticalJobs**, comme décrit dans «Création d'une adresse URL basique», à la page 97.

Vous pouvez également spécifier un des filtres ci-après :

&workstation
Filtre par le poste de travail sur lequel le travail s'exécute

&jobstream
Filtre par le flot de travaux contenant les travaux.

&job Filtre par le nom du travail.

&schedtime
Filtre par l'heure de planification du travail.

&colonnes
Spécifiez le nombre de résultats que vous voulez afficher dans votre tableau de résultats. S'il n'est pas spécifié, le nombre de colonnes par défaut de cette requête s'affiche. Les valeurs prises en charge sont :

Min Affiche une ensemble minimum de colonnes

Toutes Affiche toutes les colonnes

Exemple :

```
https://mypc:29043/racine_contexte_DASH/xLaunch.do?pageID=com.ibm.tws.WebUI.
External.navigation&showNavArea=false&action=BrowseCriticalJobs
&hostname=webuidev&port=31117
&workstation=mon_ws&jobstream=my_js_name
&job=my_job_name&server=C851&columns=Min
```

où

&server
est un paramètre utilisé pour z/OS uniquement.

racine_contexte_DASH
Il s'agit de la racine de contexte Dashboard Application Services Hub définie lors de l'installation. La racine de contexte détermine l'URL d'une application déployée et, par défaut, est identique au répertoire d'application ou à la structure d'archive. Dans ce cas, la valeur par défaut est `ibm/console`.

Contrôle des flots de travaux

Créez une adresse URL en spécifiant l'action **BrowseJobStreams**, comme décrit dans «Création d'une adresse URL basique», à la page 97.

Vous pouvez également spécifier un des filtres ci-après :

&workstation
Valide pour le système réparti uniquement. Filtre par le poste de travail sur lequel le flot de travaux s'exécute

&jobstream
Filtre par le nom du flot de travaux .

&colonnes
Spécifiez le nombre de résultats que vous voulez afficher dans votre tableau de résultats. S'il n'est pas spécifié, le nombre de colonnes par défaut de cette requête s'affiche. Les valeurs prises en charge sont :

Min Affiche une ensemble minimum de colonnes

Toutes Affiche toutes les colonnes

Exemple :

```
https://mypc:29043/racine_contexte_DASH/xLaunch.do?pageID=com.ibm.tws.WebUI.
External.navigation&showNavArea=false&action=BrowseJobStreams
&hostname=webuidev&port=31117
&workstation=mon_ws&jobstream=my_js_name
&server=C851&columns=ALL
```

où,

racine_contexte_DASH
: Il s'agit de la racine de contexte Dashboard Application Services Hub définie lors de l'installation. La racine de contexte détermine l'URL d'une application déployée et, par défaut, est identique au répertoire d'application ou à la structure d'archive. Dans ce cas, la valeur par défaut est `ibm/console`.

server Paramètre utilisé pour les systèmes z/OS uniquement.

Contrôle des postes de travail

Créez une adresse URL en spécifiant l'action **BrowseWorkstation**, comme décrit dans «Création d'une adresse URL basique», à la page 97.

Vous pouvez également spécifier un des filtres ci-après :

&workstation
: Filtre par nom de poste de travail.

&colonnes
: Spécifiez le nombre de résultats que vous voulez afficher dans votre tableau de résultats. S'il n'est pas spécifié, le nombre de colonnes par défaut de cette requête s'affiche. Les valeurs prises en charge sont :

  Min Affiche une ensemble minimum de colonnes

  All Affiche toutes les colonnes

Exemple :

```
https://mypc:29043/racine_contexte_DASH/xLaunch.do?pageID=com.ibm.tws.WebUI.
External.navigation&showNavArea=false&action=BrowseWorkstation
&hostname=webuidev&port=31117
&workstation=mon_ws&jobstream=my_js_name
&server=C851&columns=ALL
```

où :

&server
: est un paramètre utilisé pour z/OS uniquement.

racine_contexte_DASH
: Il s'agit de la racine de contexte Dashboard Application Services Hub définie lors de l'installation. La racine de contexte détermine l'URL d'une application déployée et, par défaut, est identique au répertoire d'application ou à la structure d'archive. Dans ce cas, la valeur par défaut est `ibm/console`.

Tâche existante

Créez une adresse URL en spécifiant l'action **InternalTask**, comme décrit dans «Création d'une adresse URL basique», à la page 97.

Vous pouvez sauvegarder cette URL dans les marque-pages de votre navigateur pour pouvoir ouvrir directement les résultats de la tâche créée précédemment en cliquant sur le marque-page.

Pour sauvegarder une adresse URL de tâche, procédez comme suit :

1. Créez une tâche avec la Dynamic Workload Console :

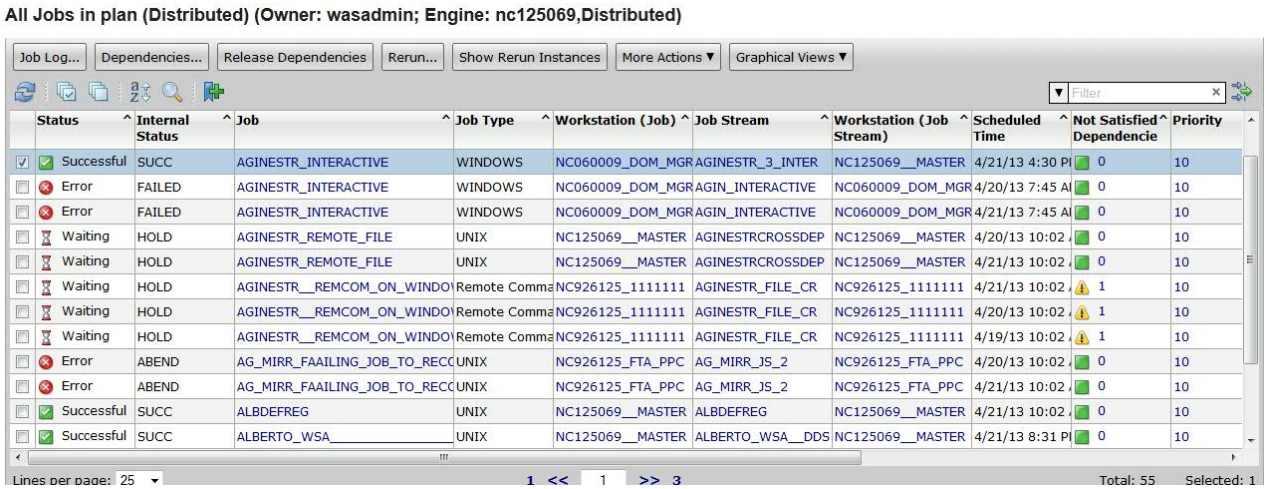
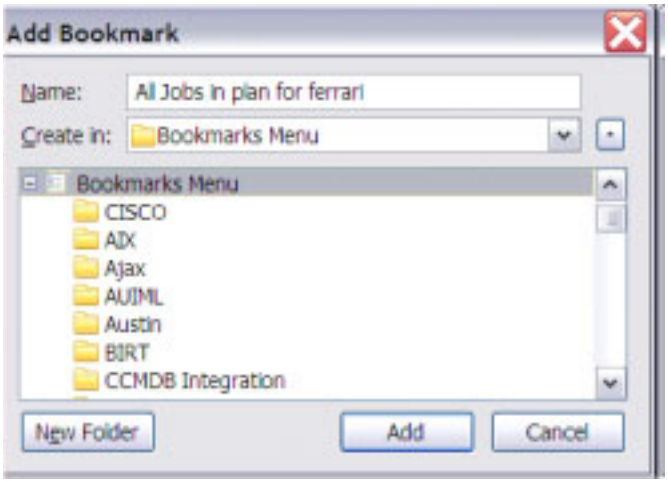


Figure 1. Liste de tâches

2. Dans le panneau affiché, cliquez sur **Icône Ajouter un marque-page**

   pour sauvegarder ce lien dans vos marque-pages.
3. Indiquez un nom pour le nouveau marque-page. Le nom de la tâche est utilisé par défaut. Organisez vos marque-pages selon votre convenance, par exemple, vous pouvez organiser vos tâches sauvegardées dans un dossier différent pour chaque moteur.



Exemple de marque-page sauvegardé :

```
https://cairapc:29043/ibm/console/xLaunch.do?pageID=com.ibm.tws.WebUI.
External.navigation&showNavArea=false
&action=InternalTask&hostname=fferrar4&port=31117
&taskname=All%20Jobs%20in%20plan%20for%20ferrari
```

A partir de ce marque-page, vous pouvez créer une adresse URL manuellement de la façon suivante :

```
https://mypc:29043/racine_contexte_DASH/xLaunch.do?pageID=com.ibm.tws.WebUI.
External.navigation&showNavArea=false&action= InternalTask
&hostname=webuidev&port=31117
&server=C851 &taskname=myTask
```

où :

&server
: est un paramètre utilisé pour z/OS uniquement.

racine_contexte_DASH
: Il s'agit de la racine de contexte Dashboard Application Services Hub définie lors de l'installation. La racine de contexte détermine l'URL d'une application déployée et, par défaut, est identique au répertoire d'application ou à la structure d'archive. Dans ce cas, la valeur par défaut est `ibm/console`.

Configuration de l'accès à Dynamic Workload Console

Dès que l'installation de Dynamic Workload Console est terminée, vous pouvez la lancer en utilisant le lien fourni dans le panneau d'installation final.

Cependant, après l'installation, l'administrateur est le seul utilisateur qui peut se connecter à la console, à l'aide des données d'identification spécifiées dans la console.

Il s'agit de l'utilisateur défini dans le registre de fichier (WIM) WebSphere Application Server.

Un administrateur doit effectuer deux opérations importantes avant que les utilisateurs puissent se connecter et utiliser le Dynamic Workload Console

- «Configuration d'un registre d'utilisateurs», à la page 105
- «Configuration des rôles pour accéder à Dynamic Workload Console», à la page 106

Si WebSphere Application Server est configuré pour utiliser le registre d'utilisateurs du registre de fichier (WIM) (valeur par défaut), les utilisateurs et les groupes doivent être créés via la console d'administration de WebSphere Application Server :

Pour créer et attribuer des rôles, connectez-vous à la console Dashboard Application Services Hub et procédez comme suit :

1. Dans la barre de navigation placée à gauche, cliquez sur l'icône en forme de loupe située en haut de la barre d'outils. Dans la zone de recherche, entrez Console d'administration WebSphere pour ouvrir la console d'administration.
2. Cliquez sur **Lancer la console d'administration WebSphere**.
3. Dans l'arborescence de navigation de la console d'administration, cliquez sur **Utilisateurs et groupes** > **Gestion des utilisateurs** pour créer un nouvel utilisateur dans le registre de fichiers (ne le créez pas sur le système d'exploitation).

Pour plus d'informations à propos de la création et de l'attribution de rôles, reportez-vous à l'aide en ligne de Dashboard Application Services Hub en cliquant sur le **?**" (point d'interrogation) dans l'angle supérieur droit des panneaux.

Si WebSphere Application Server est configuré pour utiliser le registre d'utilisateurs du système d'exploitation local, les utilisateurs et les groupes doivent être créés

dans le système d'exploitation local. Vous pouvez toutefois remplacer le système d'exploitation local par LDAP ou un registre de fichiers et vice versa, ou configurer l'utilisation de plusieurs de ces éléments.

Les utilisateurs définis dans le registre d'utilisateurs peuvent se connecter à Dynamic Workload Console ; ils doivent ensuite être associés à un rôle pour accéder au fonctions de Dynamic Workload Console (voir "«Configuration des rôles pour accéder à Dynamic Workload Console», à la page 106.)

Par défaut, la console Dynamic Workload Console utilise le registre de fichier (WIM) pour les authentifications. Si vous souhaitez passer à une authentification de système d'exploitation local ou de module PAM, suivez la procédure décrite à la section «Configuration de Dynamic Workload Console pour utiliser la méthode d'authentification du système d'exploitation local ou PAM».

Si la console Dynamic Workload Console utilise le module PAM (Pluggable Authentication Module) pour l'authentification sur un système d'exploitation UNIX et que vous souhaitez passer à une authentification de système d'exploitation local, suivez la procédure définie dans «Configuration de Dynamic Workload Console pour utiliser la méthode d'authentification du système d'exploitation local ou PAM».

Remarque : Si au moins deux instances de la console Dynamic Workload Console partagent un même référentiel de base de données pour leurs paramètres, mais qu'elles ne sont pas configurées en mode haute disponibilité, elles doivent toutes disposer du même niveau de groupe de correctifs.

Configuration d'un registre d'utilisateurs

Si WebSphere Application Server est configuré pour utiliser le registre d'utilisateurs LDAP, les utilisateurs et les groupes doivent être créés par l'administrateur système dans la base de données du serveur LDAP choisie.

La configuration des registres d'utilisateurs pour Dynamic Workload Console et tous les autres composants de Tivoli Workload Scheduler est décrite dans Chapitre 5, «Configuration de l'authentification», à la page 199.

Configuration de Dynamic Workload Console pour utiliser la méthode d'authentification du système d'exploitation local ou PAM

Pour remplacer la méthode d'authentification PAM par la méthode d'authentification du système d'exploitation local, procédez comme suit :

1. Connectez-vous à Dynamic Workload Console avec les données d'identification d'administration WebSphere Application Server.
2. Dans la barre de navigation placée à gauche, cliquez sur l'icône en forme de loupe située en haut de la barre d'outils. Dans la zone de recherche, entrez Console d'administration WebSphere pour ouvrir la console d'administration.
3. Cliquez sur **Lancer la console d'administration WebSphere**.
4. Dans l'arborescence de navigation de la console d'administration, cliquez sur **Utilisateurs et groupes** > **Gestion des utilisateurs** pour créer un nouvel utilisateur dans le registre de fichiers (ne le créez pas sur le système d'exploitation).
5. Sauvegardez la configuration de WebSphere Application Server à l'aide de la commande **backupConfig**.

6. Exportez vos propriétés actuelles vers un fichier texte à l'aide de la commande suivante :

   ```
   showSecurityProperties.sh > fichier_texte
   ```

7. Personnalisez les propriétés de sécurité en éditant le fichier comme suit :

   Remarque : Si vous souhaitez utiliser l'authentification PAM, indiquez la propriété suivante dans le fichier `activeUserRegistry=Custom`.

   ```
   ################################################################
   Global Security Panel
   ################################################################
   enabled=true
   enforceJava2Security=false
   useDomainQualifiedUserNames=false
   cacheTimeout=600
   ltpaTimeOut=720
   issuePermissionWarning=false
   activeProtocol=CSI
   useFIPS=false
   activeAuthMechanism=LTPA
   activeUserRegistry=LocalOS

   ################################################################
   Federated Repository Panel
   ################################################################
   PrimaryAdminId=new_user
   UseRegistryServerId=true
   ServerID=new_user
   ServerPassword=new_pwd
   VMMRealm=TWSREALM
   VMMRealmDelimiter=@
   VMMIgnoreCase=true
   ```

8. Arrêtez le serveur à l'aide de l'outil was **stopWas.sh**. Pour arrêter le serveur, utilisez les données d'identification d'administration WebSphere Application Server.
9. Chargez les nouvelles propriétés en entrant la commande suivante :

   ```
   chemin_complet/changeSecurityProperties.sh fichier_texte
   ```

10. Redémarrez le serveur à l'aide de l'outil was **startWas.sh**.

Configuration des rôles pour accéder à Dynamic Workload Console

Lors de l'installation de Dynamic Workload Console, de nouveaux rôles prédéfinis sont créés dans Dashboard Application Services Hub. Ils déterminent quels panneaux de console sont disponibles pour l'utilisateur, et quelles activités cet utilisateur peut effectuer dans Dynamic Workload Console.

Si vous n'attribuez aucun des rôles prédéfinis à l'utilisateur Dashboard Application Services Hub, ce dernier, une fois connecté, ne verra aucune entrée Tivoli Workload Scheduler ou Tivoli Dynamic Workload Broker dans l'arborescence de navigation.

Selon le référentiel de sécurité que vous utilisez, les points suivants s'appliquent :

LocalOS
: Créez l'utilisateur dans le système d'exploitation et attribuez les rôles à cet utilisateur à l'aide de la console Dashboard Application Services Hub.

WIM
: Créez l'utilisateur à l'aide de la console d'administration WebSphere. L'utilisateur est un utilisateur WebSphere Application Server. Attribuez ensuite les rôles à l'aide de la console Dashboard Application Services Hub

comme suit : dans la barre de navigation de la console Dashboard Application Services Hub, cliquez sur l'icône des paramètres ouvrant le menu **Paramètres de la console** et cliquez sur **Rôles**.

Pour créer et attribuer des rôles, connectez-vous à la console Dashboard Application Services Hub et procédez comme suit :

1. Dans la barre de navigation placée à gauche, cliquez sur l'icône en forme de loupe située en haut de la barre d'outils. Dans la zone de recherche, entrez Console d'administration WebSphere pour ouvrir la console d'administration.
2. Cliquez sur **Lancer la console d'administration WebSphere**.
3. Dans l'arborescence de navigation de la console d'administration, cliquez sur **Utilisateurs et groupes** > **Gestion des utilisateurs** pour créer un nouvel utilisateur dans le registre de fichiers (ne le créez pas sur le système d'exploitation).

Pour plus d'informations à propos de la création et de l'attribution de rôles, reportez-vous à l'aide en ligne de Dashboard Application Services Hub en cliquant sur le **?**" (point d'interrogation) dans l'angle supérieur droit des panneaux.

Astuce

Il n'est pas nécessaire d'attribuer un rôle à chaque utilisateur unique. Si le registre d'utilisateurs contient déjà des groupes d'utilisateurs correctement définis pour l'utilisation de la console, vous pouvez attribuer les rôles aux groupes. Si les groupes ne sont pas disponibles dans le registre des utilisateurs, le rôle spécial **tous les utilisateurs autorisés du portail** peut être utilisé pour attribuer des rôles à *tous* les utilisateurs à la fois.

Dans Dashboard Application Services Hub, vous pouvez créer vos propres vues personnalisées afin de permettre aux utilisateurs d'accéder à toutes les pages ou à un sous-ensemble de pages de Tivoli Workload Scheduler pages. Pour ce faire, vous devez disposer du rôle **wasadmin** et procéder comme suit :

1. Créez une nouvelle Vue Dashboard Application Services Hub avec les pages que vous souhaitez rendre disponibles :
   a. Dans la barre de navigation de Dashboard Application Services Hub, cliquez sur l'icône des paramètres ouvrant le menu Paramètres de la console et cliquez sur **Vues**.
   b. Cliquez sur **Nouveau**.
   c. Dans l'écran Vues, cliquez sur **Nouveau** et fournissez un nom pour la nouvelle vue.
   d. Sur la page Vues, développez **Pages dans cette vue** et cliquez sur **Ajouter** pour ajouter des pages à la nouvelle vue.
   e. Sélectionnez les pages que vous voulez rendre disponibles pour cette vue et cliquez sur **Ajouter**
2. Ajoutez la vue créée au rôle **tous les utilisateurs de portail authentifiés** :
   a. Dans la barre de navigation de Dashboard Application Services Hub, cliquez sur l'icône des paramètres ouvrant le menu Paramètres de la console et cliquez sur **Rôles**.
   b. Cliquez sur le rôle **Tous les utilisateurs de portail authentifiés**.
   c. Développez la section **Accéder aux vues**, cliquez sur l'icône **Ajouter** (signe plus), sélectionnez la nouvelle vue que vous voulez ajouter à ce rôle et cliquez sur **Ajouter**.
   d. Développez **Accéder aux vues** et sélectionnez les pages auxquelles les utilisateurs de ce rôle devront avoir accès.

Voici la liste des rôles prédéfinis créés dans Dashboard Application Services Hub pour accéder aux environnements Tivoli Workload Scheduler à l'aide de Dynamic Workload Console :

TWSWEBUIAdministrator
Les utilisateurs dans ce groupe peuvent voir l'intégralité du portefeuille et utiliser toutes les fonctions de Dynamic Workload Console.

Les utilisateurs de ce groupe peuvent également accéder à toutes les fonctions des applications mobiles Catalogue libre-service et Tableaux de bord libre-service, et les utiliser. A partir de l'application mobile Catalogue libre-service, ils peuvent créer et éditer des catalogues, créer et éditer des services, ajouter des services aux catalogues, soumettre des services associés aux flots de travaux et, enfin, partager des catalogues et des services avec d'autres utilisateurs. A partir de l'application mobile Tableaux de bord libre-service, ils peuvent créer et éditer des tableaux de bord pour filtrer des travaux et des postes de travail, afficher un tableau de bord de résultats et effectuer des actions de reprise sur un résultat unique.

TWSWEBUIConfigurator
Les utilisateurs présents dans ce groupe peuvent gérer les connexions au planificateur, les préférences utilisateur et la conception de l'environnement de planification de Dynamic Workload Console.

TWSWEBUIOperator
Les utilisateurs de ce groupe peuvent voir sur Dynamic Workload Console :

- Toutes les tâches de Surveillance
- Les travaux et flots de travaux à soumettre sur demande
- Gestion des préférences utilisateur

TWSWEBUIDeveloper
Les utilisateurs de ce groupe peuvent créer, répertorier et modifier des définitions de charge de travail, des postes de travail et des définitions de règle d'événement dans la base de données Tivoli Workload Scheduler.

TWSWEBUIAnalyst
Les utilisateurs de ce groupe peuvent gérer les rapports et les préférences utilisateur de Dynamic Workload Console.

Les utilisateurs de ce groupe peuvent également accéder aux applications mobiles Catalogue libre-serviceet Tableaux de bord libre-service, mais les actions possibles sont limitées à la soumission de demandes de service (flots de travaux) depuis les applications mobiles Catalogue libre-service et Tableaux de bord libre-service, l'affichage d'un tableau de bord des résultats et l'exécution d'actions de reprise sur ces services.

TWSWEBUIBusinessDeveloper
Les utilisateurs de ce groupe peuvent accéder aux applications mobiles Catalogue libre-service et Tableaux de bord libre-service, et les utiliser. A partir de l'application mobile Catalogue libre-service, ils peuvent créer et éditer des catalogues, créer et éditer des services, ajouter des services aux catalogues, supprimer des services et des catalogues, et soumettre des services associés aux flots de travaux. A partir de l'application mobile Tableaux de bord libre-service, ils peuvent créer et éditer des tableaux de bord pour filtrer des travaux et des postes de travail, afficher un tableau de bord de résultats et effectuer des actions de reprise sur un résultat unique. Pour partager des catalogues et des services avec d'autres utilisateurs, l'utilisateur TWSWEBUIBusinessDeveloper peut les affecter aux rôles

personnalisés dont il dispose mais pas aux rôles prédéfinis. Les utilisateurs disposant des mêmes rôles pourront alors les utiliser. Les utilisateurs avec tous les rôles personnalisés peuvent soumettre des services. Ils peuvent aussi afficher, éditer et supprimer des services, des catalogues et des tableaux de bord. En revanche, les utilisateurs avec un seul ou certains des rôles personnalisés ne peuvent que soumettre des services et afficher des services, des catalogues et des tableaux de bord.

Si un utilisateur avec le rôle administrateur crée des catalogues, des services et des tableaux de bord mais ne leur affecte pas de rôle, les utilisateurs possédant le rôle TWSWEBUIBusinessDeveloper ne peut pas les afficher ou les utiliser.

Remarque : Si un rôle personnalisé est retiré d'un catalogue, d'un service ou d'un tableau de bord, en plus de l'utilisateur TWSWEBUIBusinessDeveloper, les utilisateurs disposant de ce rôle ne pourront plus les voir et les utiliser même s'ils possèdent d'autres rôles personnalisés qui sont actuellement affectés au catalogue ou au service. L'administrateur doit réaffecter le rôle personnalisé au catalogue, au service ou au tableau de bord pour que ces derniers soient de nouveau accessibles à l'utilisateur TWSWEBUIBusinessDeveloper et aux autres utilisateurs disposant de ce rôle personnalisé.

Le tableau suivant contient certaines entrées de la barre de navigation et les activités que vous pouvez effectuer dans Dynamic Workload Console. Les groupes d'utilisateurs ayant accès à chacun de ces éléments sont également indiqués.

Tableau 25. Menu et droits d'accès du groupe

Elément du menu	**Groupes avec des droits d'accès**
Démarrage rapide	TWSWEBUIAdministrator
Toutes les tâches configurées	TWSWEBUIAdministrator TWSWEBUIOperator
Gestion des rapports de charge de travail	TWSWEBUIAdministrator TWSWEBUIAnalyst
Administration -> Conception de la charge de travail	TWSWEBUIAdministrator TWSWEBUIDeveloper
Administration -> Prévision de charge de travail	TWSWEBUIAdministrator TWSWEBUIOperator
Administration -> Soumission de la charge de travail	TWSWEBUIAdministrator TWSWEBUIOperator
Administration -> Surveillance	TWSWEBUIAdministrator TWSWEBUIOperator
Administration -> Conception de la charge de travail	TWSWEBUIAdministrator TWSWEBUIConfigurator
Administration -> Surveillance	TWSWEBUIAdministrator TWSWEBUIOperator
Génération de rapports de charge de travail	TWSWEBUIAdministrator TWSWEBUIAnalyst
Configuration système ->Gestion des moteurs	TWSWEBUIAdministrator TWSWEBUIConfigurator

Tableau 25. Menu et droits d'accès du groupe (suite)

Elément du menu	Groupes avec des droits d'accès
Configuration système -> Gestion des préférences utilisateur	TWSWEBUIAdministrator TWSWEBUIOperator TWSWEBUIConfigurator TWSWEBUIDeveloper TWSWEBUIAnalyst
Configuration système -> Gérer les paramètres	TWSWEBUIAdministrator

L'attribution d'un rôle prédéfini à un utilisateur de Dashboard Application Services Hub permet à cet utilisateur d'accéder aux panneaux de Dynamic Workload Console. L'utilisateur de Tivoli Workload Scheduler spécifié dans la connexion au moteur détermine quelles opérations peuvent être exécutées localement sur le moteur Tivoli Workload Scheduler connecté. Par exemple, si l'utilisateur spécifié dans une connexion au moteur Tivoli Workload Scheduler n'est pas autorisé à exécuter des rapports dans le Tivoli Workload Scheduler *Fichier de sécurité*, même si l'utilisateur de Dashboard Application Services Hub connecté au Dynamic Workload Console peut accéder aux écrans de rapport, il ou elle ne peut pas exécuter d'opérations de rapport sur ce moteur Tivoli Workload Scheduler spécifique. Pour plus d'informations sur la configuration du fichier de sécurité, voir «Configuration du fichier de sécurité», à la page 161.

Les rôles prédéfinis créés dans Dashboard Application Services Hub pour accéder aux environnements Tivoli Dynamic Workload Broker à l'aide de Dynamic Workload Console et les panneaux auxquels ils peuvent accéder sont répertoriés comme suit :

TDWBAdministrator
Tous les panneaux

TDWBOperator
Environnement de planification
Configuration
Définitions, hormis la définition d'un nouveau travail
Surveillance
Préférences

TDWBDeveloper
Configuration
Définitions
Préférences

TDWBConfigurator
Environnement de planification
Configuration
Surveillance, sauf celle des instances de travail
Préférences

Configuration de Dynamic Workload Console en vue d'utiliser une connexion unique

Connexion unique (SSO) est une méthode de contrôle d'accès permettant à un utilisateur de s'authentifier une fois et d'accéder aux ressources de plusieurs applications partageant le même registre d'utilisateurs.

En d'autres termes, SSO vous permet d'exécuter des requêtes sur le plan ou de gérer des définitions d'objets sur la base de données en accédant au moteur sans vous authentifier ; les données d'identification fournies pour vous connecter à Dynamic Workload Console sont automatiquement utilisées.

La même chose est vraie lors de l'utilisation des applications Catalogue libre-service et Tableaux de bord libre-service à partir d'une unité mobile. Si la console Dynamic Workload Console a été configurée pour utiliser SSO, ces applications utilisent alors automatiquement les mêmes identification pour se connecter à Dynamic Workload Console.

Une fois l'installation terminée, vous pouvez configurer Dynamic Workload Console et le moteur Tivoli Workload Scheduler pour l'utilisation de la connexion unique (SSO). Pour ce faire, les applications doivent partager le même registre d'utilisateurs LDAP.

Remarque : Tivoli Workload Scheduler versions antérieures à 8.6 ne prennent pas en charge LDAP dans un référentiel fédéré, tandis que Dynamic Workload Console version 8.6 prend en charge LDAP uniquement dans un registre fédéré. Par conséquent, si vous projetez de configurer Dynamic Workload Console version 8.6 pour la Connexion unique avec Tivoli Workload Scheduler antérieur à la version 8.6, vous devez utiliser l'interface Integrated Solutions Console pour configurer l'authentification, comme décrit dans la section «Configuration de l'authentification à l'aide de WebSphere Administrative Console», à la page 202.

Le protocole LDAP (Lightweight Directory Access Protocol) correspond à un protocole d'application permettant d'interroger et de modifier des services d'annuaire s'exécutant sur TCP/IP (pour plus de détails, voir Chapitre 5, «Configuration de l'authentification», à la page 199).

Si vous avez configuré Dynamic Workload Console pour utiliser Connexion unique avec un moteur, alors, le comportement suivant est appliqué :

Si les données d'identification de l'utilisateur sont spécifiées dans les définitions du moteur de connexion
: Ces données d'identification sont utilisées. Ce comportement concerne également les connexions au moteur partagées et dont les données d'identification sont partagées.

Si les données d'identification de l'utilisateur ne sont pas spécifiées dans le moteur de connexion
: Les données d'identification spécifiées lors de la connexion à Dynamic Workload Console sont utilisées. Ce comportement concerne également les connexions au moteur partagées dont les données d'identification ne sont pas partagées.

clés de type jeton LTPA

Outre le partage du registre d'utilisateurs LDAP, l'instance de WebSphere Application Server prenant en charge Dynamic Workload Console et l'instance prenant en charge le moteur où la Connexion unique est requise doivent être configurées pour utiliser les mêmes clés de type jeton LTPA (Lightweight Third-Party Authentication). Voir «Configuration de LTPA (Lightweight Third-Party Authentication)», à la page 112

Configuration de LTPA (Lightweight Third-Party Authentication)

WebSphere Application Server utilise le mécanisme LTPA (Lightweight Third-Party Authentication) pour propager les données d'identification.

Selon les circonstances, vous allez devoir configurer l'utilisation des mêmes clés de type jeton LTPA entre Dynamic Workload Console et le moteur et/ou désactiver la génération automatique des clés de type jeton LTPA :

Configuration de Connexion unique
Si vous configurez Connexion unique, entre l'une des versions de Dynamic Workload Console et l'un des moteurs, qu'ils soient installés ou pas sur le même système, vous devez configurer les instances de WebSphere Application Server concernées de manière à utiliser les mêmes clés de type jeton LTPA et désactiver leur régénération automatique après expiration, en suivant la procédure décrite ci-dessous :

- «Configuration de l'utilisation des mêmes clés de type jeton LTPA»
- «Désactivation de la génération automatique des clés de type jeton LTPA», à la page 115

Pas de Connexion unique et une seule instance de WebSphere Application Server sur un système
Aucune action à entreprendre.

Configuration de l'utilisation des mêmes clés de type jeton LTPA

Pour utiliser les mêmes clés de type jeton entre plusieurs instances de WebSphere Application Server, vous devez effectuer cette procédure entre Dynamic Workload Console et chaque moteur auquel vous souhaitez vous connecter.

Les clés de type jeton LTPA peuvent être soit exportées depuis Dynamic Workload Console et importées dans le moteur, soit exportées depuis le moteur et importées dans Dynamic Workload Console.

1. Utilisez le script ci-dessous pour exporter les clés de type jeton LTPA du WebSphere Application Server sur lequel réside Dynamic Workload Console et les importer dans l'autre instance de WebSphere Application Server :

   Tivoli Workload Scheduler et Dynamic Workload Console, Version 9.2
   <*TWA_home*>/wastools/manage_ltpa.sh or ...\manage_ltpa.bat

   <*installation_directory_Dynamic_Workload_Console*>/wastools/manage_ltpa.sh ou ...\manage_ltpa.bat, par exemple, /opt/IBM/TWAUI/wastools/manage_ltpa.sh.

   Tivoli Workload Scheduler et Dynamic Workload Console, version 8.5, 8.5.1 et 8.6 <*TWA_home*>/wastools/manage_ltpa.sh or ...\manage_ltpa.bat

   Tivoli Workload Scheduler, Version 8.4
   <*TWA_home*>/wastools/manage_ltpa.sh or ...\manage_ltpa.bat

   Dynamic Workload Console, Version 8.4
   *tdwc_install_dir*_tdwcutils\scripts\manage_ltpa.sh ou ...\manage_ltpa.cmd

   Des copies des scripts manage_ltpa.sh et manage_ltpa.bat figurent également sur chaque image d'installation.

Assurez-vous que l'utilisateur exécutant ce script est autorisé à accéder au profil WebSphere Application Server hébergeant Dynamic Workload Console ou le moteur.

La syntaxe utilisée pour exécuter le script est la suivante :

```
manage_ltpa -operation import|export -profilepath profile_path
            -ltpafile LTPA_file_path -ltpapassword LTPA_file_password
            -user username -password password
            -port port_SOAP -server nom_serveur
```

Où :

–operation
Sélectionnez *export* pour lire les clés de type jeton LTPA à partir du profil et de l'enregistrer dans un fichier. Sélectionnez *import* pour mettre à jour le profil avec les clés_jeton LTPA stockées dans un fichier.

–profilepath
Spécifiez le chemin du profil sur lequel l'application, à savoir Dynamic Workload Console ou Tivoli Workload Scheduler, est installée.

–ltpafile
Spécifiez le nom de chemin complet du fichier contenant les clés de type jeton LTPA, si vous les importez, les exportez ou les chiffrez.

–ltpapassword
Spécifiez le mot de passe de votre choix afin de chiffrer le fichier contenant les clés LTPA lors de leur exportation ou, lors de leur importation, celui qui a été utilisé pour les chiffrer lors de leur exportation. Ce mot de passe n'est utilisé que pour importer et exporter ces clés de type jetons LTPA. Il n'est pas nécessaire qu'il corresponde au mot de passe d'administrateur.

–user L'administrateur du serveur hébergeant Dynamic Workload Console ou le moteur. Dans le cas de Tivoli Workload Scheduler, l'administrateur est, par défaut, le propriétaire de l'instance (*utilisateur_TWS*).

–password
Mot de passe de l'administrateur du serveur défini dans le profil sélectionné.

–port Spécifiez le port SOAP utilisé par le profil. Par défaut, le port SOAP est 28880 pour Dynamic Workload Console installé sur la WebSphere Application Server et 31118 pour Tivoli Workload Scheduler installé sur la WebSphere Application Server.

–server
Spécifiez le nom du serveur du profil sur lequel importer ou exporter les jetons LTPA. Le nom de serveur par défaut varie en fonction de la façon dont il a été installé. Voir tableau 26, à la page 114.

Remarque :

a. La valeur par défaut du serveur et du chemin d'accès peut avoir été modifié après l'installation.
b. Ce mot-clé est obligatoire si le nom de serveur Tivoli Workload Scheduler est différent du nom de serveur Dynamic Workload Console.

Tableau 26. Versions de produit et noms de serveur par défaut

Version du produit	**Version de WebSphere Application Server**	**Nom de serveur par défaut**
Tivoli Workload Scheduler, V9.2:	Le serveur WebSphere Application Server installé dans une instance de Tivoli Workload Automation (sur laquelle tous les composants Tivoli Workload Scheduler sont installés).	server1, accessible à l'emplacement suivant : `<chemin_profil_WAS>/config/cells/TWSNodeCell/nodes/TWSNode/servers/server1/server.xml` où la valeur par défaut du chemin *chemin_profil_WAS* est `<TWA_home>/WAS/TWSprofile`

	Votre version du serveur WebSphere Application Server sur lequel la console Dynamic Workload Console est installée.	server1, accessible à l'emplacement suivant :
		`rép_profil_JazzSM/config/cells/JazzSMNode01Cell/nodes/JazzSMNode01/servers/server1/`
		
		où la valeur par défaut de *rép_profil_JazzSM* est :
		Sur les systèmes d'exploitation Windows
		`C:\Program Files\IBM\JazzSM\profile`
		Sur les systèmes d'exploitation UNIX
		`/opt/IBM/JazzSM/profile`

Tivoli Workload Scheduler version 8.6	La WebSphere Application Server intégré installée dans une instance de Tivoli Workload Automation (sur laquelle tous les composants Tivoli Workload Scheduler, y compris Dynamic Workload Console, sont installés).	server1, accessible à l'emplacement suivant : `<TWA_home>/eWAS/profiles/TIPProfile/config/cells/TIPCell/nodes/TIPNode/servers/server1/`
Tivoli Workload Scheduler version 8.5 et 8.5.1 :	La WebSphere Application Server intégré installée dans une instance de Tivoli Workload Automation (sur laquelle tous les composants Tivoli Workload Scheduler, y compris Dynamic Workload Console, sont installés).	`twaserver<n>`, qui se trouve à l'emplacement suivant : `<TWA_home>/eWAS/profiles/twaprofile/config/cells/DefaultNode/nodes/DefaultNode/servers`
	Votre version externe de WebSphere Application Server, sur lequel Dynamic Workload Console est installée.	`server1`, installé dans le chemin approprié de votre version externe de WebSphere Application Server
Tivoli Workload Scheduler, V8.4 et versions antérieures :	WebSphere Application Server intégré où Tivoli Workload Scheduler est installé.	`server1`, installé dans le chemin suivant : `<racine_TWS>/appserver/profiles/twsprofile/config/cells/DefaultNode/nodes/DefaultNode/servers`
	WebSphere Application Server intégré où Dynamic Workload Console est installé.	`tdwcserver`, installé dans le chemin suivant : `<tdwc_install_dir>/AppServer/profiles/tdwcprofile/servers`
	Votre version externe de WebSphere Application Server, sur lequel Dynamic Workload Console est installée.	`server1`, installé dans le chemin approprié de votre version externe de WebSphere Application Server

2. Arrêtez et redémarrez tous les serveurs impliqués dans cette activité pour l'activer.
3. Si vous configurez une connexion unique, vérifiez que la configuration est correctement définie entre et le moteur en procédant comme suit :
   a. Connectez-vous à Dynamic Workload Console.

b. Créez une connexion au moteur sans spécifier d'ID utilisateur et de mot de passe.
c. Testez la connexion.

L'étape suivante consiste à désactiver la génération automatique des clés de type jeton LTPA (voir «Désactivation de la génération automatique des clés de type jeton LTPA»)

Désactivation de la génération automatique des clés de type jeton LTPA

Désactivez la génération automatique des clés de type jeton LTPA si vous activez la Connexion unique. Vous devez désactiver la génération des clés aux deux extrémités de la communication, c'est-à-dire au niveau de Dynamic Workload Console et au niveau du moteur de Tivoli Workload Scheduler ou de Tivoli Dynamic Workload Broker, selon le cas :

Au niveau de Dynamic Workload Console

1. Connectez-vous à Dynamic Workload Console.
2. Cliquez sur **Paramètres > WebSphere Administrative Console > Lancer WebSphere > Administrative Console**.
3. Dans WebSphere Administrative Console, cliquez sur **SSL certificate and key management**.
4. Cliquez sur le lien relatif aux **groupes de jeux de clés**.
5. Cliquez sur le nom du groupe de jeux de clés affiché dans la liste.
6. Décochez la case de **génération automatique des clés**.
7. Cliquez sur **OK**.
8. Vérifiez dans la liste que la zone relative à la **génération automatique de clés** située sous le groupe de jeux de clés disponibles possède la valeur *false*.

Au niveau de Tivoli Workload Scheduler

L'implémentation du WebSphere Application Server sur le moteur Tivoli Workload Scheduler inclut une version à fonctionnalités limitées du Integrated Solutions Console. Utilisez ce portail pour désactiver la génération automatique de clés_jetons_LTPA, comme suit :

1. Connectez-vous au Integrated Solutions Console à partir d'un navigateur Internet, en utilisant l'URL suivante :

   `http://hostname_TWS:port_(sécurisé)_hôte_admin_WAS/ibm/console`

   Utilisez l'outil **showHostProperties** pour identifier le *port_hôte_admin_WAS* (31123 par défaut) ou *port_sécurisé_hôte_admin_WAS* (31124 par défaut), selon le cas. Pour plus d'informations à propos de cet outil, reportez-vous à «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423.
2. Effectuez la procédure décrite ci-dessus pour désactiver la génération des clés de type jeton de Dynamic Workload Console, en commençant par l'étape 2.

Configuration de Dynamic Workload Console pour l'utilisation de SSL

Le protocole SSL (Secure Sockets Layer) fournit des communications sécurisées entre les processus du serveur distant ou les applications. La sécurité SSL Peut être utilisé en vue d'établir des communications entrant ou sortant d'une application. Pour établir des communications sécurisées, vous devez spécifier un certificat et une configuration SSL pour l'application.

Vous trouverez plus d'informations dans «Scénario : connexion entre Dynamic Workload Console et le Tivoli Workload Scheduler ayant un connecteur distribué», à la page 262.

Personnalisation de Dynamic Workload Console (configuration avancée)

Pour personnaliser le comportement de la console Dynamic Workload Console, vous pouvez éventuellement configurer certains paramètres avancés. Ces paramètres sont spécifiés dans un fichier personnalisable nommé `TdwcGlobalSettings.xml`. Une copie de ce fichier est installée localement à l'emplacement suivant, après l'installation de Dynamic Workload Console :

`<rép_profil_JazzSM>/profile/registry`

Vous trouverez également une copie de ce fichier, sous la forme de modèle, sur le DVD d'installation dans le répertoire `/utilities/TdwcGlobalSettings.xml`.

Modifiez le fichier en remplaçant les valeurs par défaut par des valeurs personnalisées et en activant les sections mises en commentaire, puis enregistrez le fichier dans le répertoire `<JazzSM_profile_dir>/registry`. Voici un exemple du chemin complet du fichier :

Pour Windows :

```
C:\Program Files\IBM\JazzSM\profile\registry\TdwcGlobalSettings.xml
```

Pour UNIX/Linux :

```
/opt/ibm/JazzSM/profile/registry/TdwcGlobalSettings.xml
```

Lorsque vous modifiez ce fichier, arrêtez et redémarrez Dynamic Workload Console afin de rendre les changements effectifs dans votre environnement.

Personnalisation de vos paramètres globaux

Certains paramètres généraux de Dynamic Workload Console peuvent être inclus dans un fichier personnalisable nommé `TdwcGlobalSettings.xml`. Une copie de ce fichier est installée localement à l'emplacement suivant, après l'installation de Dynamic Workload Console :

`<rép_profil_JazzSM>/profile/registry`

Vous trouverez également une copie de ce fichier, sous la forme de modèle, sur le DVD d'installation dans le répertoire `/utilities/TdwcGlobalSettings.xml`.

Modifiez le fichier en remplaçant les valeurs par défaut par des valeurs personnalisées et en activant les sections mises en commentaire, puis enregistrez le fichier dans le répertoire `<JazzSM_profile_dir>/registry`. Le chemin complet du fichier est le suivant :

Pour Windows :

```
C:\Program Files\IBM\JazzSM\profile\registry\TdwcGlobalSettings.xml
```

Pour UNIX/Linux :
/opt/ibm/JazzSM/profile/registry/TdwcGlobalSettings.xml

Les utilisateurs dotés des privilèges d'administrateur peuvent utiliser un fichier de configuration, appelé TdwcGlobalSettings.xml, pour ajouter et modifier certaines informations personnalisables, par exemple :

- Le nombre maximal d'objets présentés dans les vues graphiques.
- Le paramètre permettant d'afficher la vue Plan dans une nouvelle fenêtre.
- Les caractéristiques de configuration permettant d'activer l'alarme de notification des nouvelles et d'être constamment mis à jour sur les informations de produits. Voir Désactivation de la notification d'informations.
- Création de tâches prédéfinies.
- Les adresses URL où vous pouvez mémoriser la documentation personnalisée sur vos travaux ou vos flots de travaux pour leur associer de la documentation personnalisée.
- Le registre d'utilisateurs en cours d'utilisation.
- Le délai d'attente des informations de lecture et d'écriture sur un moteur Tivoli Workload Scheduler for z/OS.
- Le nombre maximal d'objets extraits avec une requête, le nombre maximal de lignes à afficher dans une table et le nombre maximal de requêtes directes à conserver dans l'historique.
- Autoriser ou empêcher les utilisateurs de partager des tâches et des connexions au moteur.
- L'affichage de toutes les dépendances, satisfaites et non satisfaites.
- L'utilisation des fichiers de contrôle pour surveiller les activités dans les applications mobiles Catalogue libre-service et Tableaux de bord libre-service.

Un modèle de ce fichier se trouve sur le DVD d'installation sous /utilities/TdwcGlobalSettings.xml. Vous pouvez le modifier en remplaçant les valeurs par défaut par des valeurs personnalisées et en activant les sections mises en commentaire. Une fois la personnalisation effectuée, vous devez copier le fichier dans le chemin suivant : répertoire *rép_installation*/*rép_profil*//registry, où :

rép_profil
est le répertoire que vous avez défini comme répertoire de votre profil. Par défaut, ce répertoire est JazzSM/profile.

Par exemple, le chemin complet de ce fichier est le suivant :

Pour Windows :
C:\Program Files\IBM\JazzSM\profile\registry\TdwcGlobalSettings.xml

Pour UNIX/Linux :
/opt/ibm/JazzSM/profile/registry/TdwcGlobalSettings.xml

Ce fichier est lu à chaque connexion et toutes les configurations spécifiées dans le fichier sont appliquées immédiatement à l'exception de la propriété **precannedTaskCreation**. Cette propriété est lue uniquement lorsqu'un utilisateur se connecte pour la première fois et est utilisée ensuite à chaque fois que cet utilisateur se reconnecte.

Vous pouvez utiliser n'importe quel texte ou l'éditeur XML pour éditer ce fichier mais assurez-vous que vous l'avez enregistré comme un fichier XML valide.

Ce fichier est subdivisé dans les sections suivantes qui regroupent des propriétés similaires :

Les sections peuvent être également répétées plusieurs fois dans le même fichier et appliquées différemment aux différents rôles utilisateur. Pour n'appliquer une section qu'aux utilisateurs appartenant à un rôle, la section doit être incluse dans le paramètre suivant :

settings role
: Utilisateur pour lequel la configuration suivante doit être appliquée. Valeur par défaut : tous les utilisateurs, sauf indication contraire.

Une seule section de **settings** peut être spécifiée pour chaque rôle. Si un utilisateur possède plusieurs rôles, les paramètres associés au rôle le plus élevé sont pris en compte.

Exemple :

```
<?xml version"1.0"?>
<tdwc>
.
.
<settings>
<graphViews>
<property name="planViewNewWindow" value="true"/>
</graphViews>
</settings>

<settings role="TWSWEBUIOperator">
<graphViews>
<property name="planViewNewWindow" value="false"/>
</graphViews>
</settings>
.
.
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Remplacer les limites de la vue graphique

Cette section contient les paramètres de configuration qui s'appliquent aux vues graphiques du plan, par exemple le nombre maximum d'objets présentés dans chaque vue.

planViewMaxJobstreams
: Le nombre maximal de flots de travaux affichés dans la vue Plan. La valeur par défaut est **1000**.

jobstreamViewLimit
: Le nombre maximal d'objets affichés dans la vue Flot de travaux. La valeur par défaut est **1000**.

impactViewLimit
: Le nombre maximal de flots de travaux affichés dans la vue Impact. La valeur par défaut est **2000**.

preProdPlanViewMaxJobstreams
: Le nombre maximal de flots de travaux affichés dans la vue du plan de préproduction. La valeur par défaut est **1000**.

```
<?xml version"1.0"?>
<tdwc>
.
.
 <settings>
<graphViews>
<property name="planViewMaxJobstreams" value="1000"></property>
<property name="jobstreamViewLimit" value="1000"></property>
<property name="impactViewLimit" value="1000"></property>
<property name="preProdPlanViewMaxJobstreams" value="1000"></property>
</graphViews>
 </settings>.
.
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Vue Plan dans une nouvelle fenêtre

Cette section permet d'éviter que Internet Explorer 7 ne se fige pendant l'utilisation de la vue Plan. Pour résoudre ce problème, définissez la valeur sur **true**.

planViewNewWindow

Définissez-la sur **true** si vous voulez que la vue Plan s'affiche dans une nouvelle fenêtre à chaque lancement. La valeur par défaut est **false**.

```
<?xml version"1.0"?>
<tdwc>
.
.
 <settings>
<graphViews>
<property name="planViewNewWindow" value="true"/>
</graphViews>
.
.
 </settings>
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Désactiver et personnaliser la fonction NewsFeed

Cette section contient les détails de configuration qui doivent être constamment à jour avec les informations produit.

FeedURL

Contient l'URL depuis laquelle vous recevez des informations et mises à jour. La valeur par défaut est : https://www.ibm.com/developerworks/community/wikis/form/anonymous/api/wiki/585f5525-a7f5-48ef-9222-50ad582e85f4/page/e599dd3c-8dc3-4ab6-89fd-33f81a994799/attachment/de677e63-5a9d-46db-a010-18ca38f05812/media/tws.jsonp

FeedType

Chaîne qui identifie le format de mise à jour. La valeur par défaut est **JSONP**.

PollInterval

L'intervalle en secondes entre deux vérifications de mise à jour. La valeur par défaut est **600**.

PollInitialDelay
Délai initial en secondes avant la première tentative de lecture des newsfeeds. Après le chargement initial, l'intervalle d'interrogation est utilisé. La valeur par défaut est **120**.

NewsFeed
Propriété utilisée pour ajouter plus de newsfeeds personnalisés. Indiquez le format et l'adresse du fichier qui contient la communication personnalisée. Les formats pris en charge sont RSS 2.0 et ATOM 1.0. Vous devez écrire la communication au format ATOM 1.0 ou RSS 2.0 et stocker ce fichier sur le serveur HTTP conforme à la *même stratégie d'origine*. Pour des raisons de sécurité du navigateur, cette stratégie permet d'accéder aux informations seulement sur un serveur qui utilise le même protocole, nom d'hôte et numéro de port que celui auquel vous êtes connecté. En option, si vous voulez stocker votre flux personnalisé sur un serveur externe, vous devez configurer un serveur proxy inverse qui mappe l'adresse de serveur externe.

```
<property name="NewsFeed" type="RSS"
value="http://nom_hôte_DWC:numéro_port.com/news.rss" />
```

Remarque : Pour indiquer plusieurs flux, vous devez indiquez plusieurs sections **NewsFeed**.

NewsFeedCategory
Nom des informations personnalisées. Peut être utilisé pour identifier par exemple des messages informatifs, d'avertissement ou d'alerte. Le chemin d'accès à une image peut être également ajouté pour mieux identifier les informations associées à une icône.

Pour ajouter plus d'images de catégorie, indiquez une liste de propriétés appelées **NewsFeedCategory**, par exemple :

```
<property name="NewsFeedCategory" value="infos sur ma société"
icon="http://www.my.company.com/info.png" />
<property name="NewsFeedCategory" value="alerte de ma société"
icon="http://www.my.company.com/alert.png" />
```

Si aucun flux personnalisé n'est indiqué, c'est la valeur par défaut qui est utilisée, c'est-à-dire celle qui récupère les dernières informations de produits à partir des sites de support officiels. Pour désactiver une notification, mettez toute la section en commentaire. Pour désactiver seulement les notifications externes sur les mises à jour des informations du produit, attribuez une chaîne vide comme valeur à la propriété `FeedURL` du flux `JSONP` comme :

```
  <property name="FeedURL" type="JSONP" value="" />
```

Exemple :

```
<?xml version"1.0"?>
<tdwc>
.
.
 <settings>
<NewsFeed>
<property name="NewsFeed" type="RSS"
value="http://www.nom_hôte_DWC:numéro_port.com/my_rss.xml" />
<property name="NewsFeed" type="ATOM" value="http://www.nom_hôte_DWC:numéro_port.com/my_atom.xml" />

<property name="PollInterval" value="600" />
<property name="PollInitialDelay" value="1" />

<property name="FeedURL" type="JSONP" value="" />
```

```
<property name="NewsFeedCategory" value="my company info"
icon="http://www.nom_hôte_DWC:numéro_port.com
/info.png" />
<property name="NewsFeedCategory" value="my company alert"
icon="http://www.nom_hôte_DWC:numéro_port.com
/alert.png" />

</NewsFeed>
 </settings>
.
.
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Désactiver et personnaliser la création de tâches prédéfinies

Cette section définit l'environnement pour lequel des tâches prédéfinies sont créées.

precannedTaskCreation

Certaines tâches prédéfinies sont créées par défaut et sont disponibles lorsque vous vous connectez à la console. Il existe une tâche de surveillance prédéfinie pour chaque objet, à la fois pour les moteurs z/OS et pour les moteurs distribués. La valeur par défaut est **all**. Pour modifier ce paramètre, utilisez l'une des valeurs suivantes :

all Toutes les tâches prédéfinies sont créées. Il s'agit de l'option par défaut.

distributed
Seules les tâches prédéfinies pour les moteurs répartis sont créées

zos Seules les tâches prédéfinies pour les moteurs z/OS sont créées

none Aucune tâche prédéfinie n'est créée.

```
<?xml version"1.0"?>
<tdwc>
.
.
 <settings>
<application>
<property name="precannedTaskCreation" value="all"/>
</application>
 </settings>
.
.
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Ajouter une URL personnalisée à un travail et à des flots de travaux

Cette section présente des adresses URL qui vous permettent de mémoriser de la documentation personnalisée sur vos travaux ou vos flots de travaux. Par défaut, ce paramètre n'est pas indiqué. Pour associer une documentation personnalisée à un travail ou un flot de travaux, utilisez ce paramètre afin de spécifier l'adresse externe qui contient ces informations.

Pour définir une adresse URL sur laquelle la documentation personnalisée pour un travail et un flot de travaux est stockée, supprimez la mise en commentaire des lignes de section, indiquez l'URL requise et affectez éventuellement un nom à l'étiquette UI en spécifiant une valeur pour la propriété customActionLabel. Par défaut, le nom est **Ouvrir la documentation**. Cette étiquette apparaît ensuite dans les menus **Actions supplémentaires** dans Surveiller les travaux et dans les tâches de surveillance des flots de travaux, ainsi que dans les vues graphiques du plan (dans les infobulles, les menus contextuels et les propriétés). Dans cet exemple, sélectionnez **Ouvrir la documentation** pour accéder à la documentation correspondante, ce qui permet d'ouvrir la documentation tout en surveillant votre travail ou flot de travaux dans le plan.

Pour implémenter ce paramètre, affectez les valeurs aux mots clés suivants :

customActionLabel
: Nom de l'action affichée dans les menus, les propriétés d'objet et les infobulles permettant d'accéder à la documentation personnalisée sur vos travaux ou vos flots de travaux. Par défaut, le nom est "Ouvrir la documentation", sauf si vous personnalisez le nom avec ce mot clé.

jobUrlTemplate
: Adresse de la documentation relative à vos travaux. Aucune valeur par défaut n'est proposée.

jobstreamUrlTemplate
: Adresse de la documentation relative à vos flots de travaux. Aucune valeur par défaut n'est proposée.

```
<?xml version"1.0"?>
<tdwc>
.
.
 <settings>
  <twsObjectDoc>
    <property name="jobstreamUrlTemplate"
      value="http://www.yourhost.com/tws/docs/jobstream/${js_name_w}" />
    <property name="jobUrlTemplate"
      value="http://www.yourhost.com/docs/jobs/${job_name_w}" />
    <property name="customActionLabel" value="Your Custom Label Name"/>
  </twsObjectDoc>
 </settings>
.
.
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Ces propriétés doivent être des adresses URL valides contenant une ou plusieurs des variables répertoriées dans le tableau ci-dessous.

Si vous utilisez l'un des caractères spéciaux suivants dans l'adresse URL, vous devez les écrire comme suit :

Tableau 27. Syntaxe des caractères spéciaux

Caractères spéciaux	Ecrivez-les sous la forme...
guillemet (")	\"
apostrophe (')	&apos;
perluète (&)	&amp;

Tableau 27. Syntaxe des caractères spéciaux (suite)

Caractères spéciaux	Ecrivez-les sous la forme...
inférieur à (<)	&lt
supérieur à (>)	&gt
barre oblique inversée (\)	\\

Plusieurs variables peuvent être incluses dans une URL et doivent être spécifiées à l'aide de la syntaxe suivante : ${*variable*}:

Tableau 28. Variables utilisées dans la définition d'URL

Nom	Objet	Description
job_number_w	Travail z/OS	Numéro du travail
job_wkst_w	Travail	Nom du poste de travail sur lequel le travail s'exécute
job_jsname_w	Travail	Nom du flot de travaux qui contient le travail
job_jswkst_w	Travail	Nom du poste de travail sur lequel le flot de travaux s'exécute
job_actualarrival_w	Travail z/OS	Date et heure de début réelles du travail (format de date : AAAA-MM-JJThh:mm:ss)
job_actualend_w	Travail z/OS	Fin réelle du travail (format de date : AAAA-MM-JJThh:mm:ss)
job_starttime_w	Travail	Date et heure de début du travail (format de date : AAAA-MM-JJThh:mm:ss)
job_id_w	Travail	ID du travail
job_returncode_w	Travail	Code retour du travail
js_name_w	Flot de travaux	Nom du flot de travaux qui contient le travail
js_wkst_w	Flot de travaux	Nom du poste de travail sur lequel le flot de travaux s'exécute
js_id_w	Flot de travaux	ID du flot de travaux
js_latest_start_w	Flot de travaux	Dernier démarrage possible du flot de travaux (format de date : AAAA-MM-JJThh:mm:ss)
engine_name_w	Moteur	Nom de la connexion au moteur
engine_host_w	Moteur	Nom d'hôte de la connexion au moteur
engine_port_w	Moteur	Numéro de port de la connexion au moteur
engine_plan_w	Moteur	ID du plan sélectionné

Tableau 28. Variables utilisées dans la définition d'URL (suite)

Nom	Objet	Description
engine_serv_w	Moteur	Nom du serveur distant de la connexion au moteur

Registre d'utilisateurs

Utilisez cette section pour configurer les propriétés associées au registre d'utilisateurs utilisé.

groupIdMap

Cette propriété est associée aux groupes du registre d'utilisateurs et peut être modifiée pour mapper et afficher la valeur indiquée de chaque groupe. La valeur par défaut est le nom usuel du groupe.

Exemples :

```
<?xml version"1.0"?>
<tdwc>
.
.
<settings>
<security>
<property name="groupIdMap" value="cn"></property>
</security>
</settings>
.
.
</tdwc>
```

En conséquence, si vous devez remplacer la valeur par défaut "cn" par "racfid", vous pouvez définir cette propriété comme suit :

```
 <property name="groupIdMap" value="racfid"></property>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Connexions http z/OS

Utilisez cette section pour configurer le délai d'attente de lecteur et d'écriture des informations sur un moteur Tivoli Workload Scheduler pour z/OS. Lorsque vous vous connectez au moteur Tivoli Workload Scheduler for z/OS pour extraire une liste d'objets définis mais que celle-ci n'est pas renvoyée dans le délai spécifié, un message d'erreur apparaît. La valeur est exprimée en millisecondes.

Exemple :

```
<?xml version"1.0"?>
<tdwc>
.
.
 <settings>
<http>
<property name="zosHttpTimeout" value="90000" />
</http>
.
.
 </settings>
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Limiter le nombre d'objets extraits par des requêtes

Si vous êtes connecté au à la version 9.1 des moteurs, ce paramètre est ignoré.

Dans cette section, vous pouvez configurer le nombre de résultats affichés pour les tâches de surveillance, le nombre maximal de lignes à afficher sur chaque page et le nombre de requêtes directes à conserver dans l'historique.

Si vous souhaitez limiter le nombre de résultats générés par vos requêtes, vous pouvez spécifier le nombre maximal d'éléments à extraire à l'aide de la propriété monitorMaxObjectsPM. Le nombre minimum de résultats extraits est 500.

La valeur par défaut est -1 ; toute valeur inférieure à 0 indique qu'il n'existe pas de limite dans le nombre d'objets extraits.

Parce que les données sont extraites par blocs de 250 lignes, la valeur que vous entrez est réglée pour compléter un bloc entier. Par exemple, si vous indiquez une limite de 500, seuls 500 éléments sont extraits, tandis que si vous indiquez une limite de 600, 750 éléments sont extraits.

Pour les tâches à plusieurs moteurs, cette limite s'applique à chaque moteur inclus dans la requête. Par conséquent, si vous spécifiez une limite de 500 résultats et, par exemple, que vous exécutez une tâche de surveillance des travaux pour plusieurs moteurs sur trois moteurs, les résultats produits par votre requête seront inférieurs à 500 *pour chaque moteur*, pour un maximum de 1500 lignes.

Remarque : Ce paramètre ne s'applique pas aux tâches de surveillance critiques.

Pour définir le nombre maximal de lignes à afficher dans une vue Table, configurez la propriété maxRowsToDisplay.

Pour définir le nombre maximal de requêtes directes à conserver dans l'historique, configurez la propriété maxHistoryCount. Ces requêtes sont disponibles dans le menu déroulant de la zone Requête, sur la page Requête directe.

```
<?xml version"1.0"?>
<tdwc>
.
.
 <settings>
  <monitor>
    <property name="monitorMaxObjectsPM" value="2000"></property>
  </monitor>

  <monitor>
    <property name="maxRowsToDisplay" value="25"></property>
  </monitor>

  <monitor>
    <property name="maxHistoryCount" value="100"></property>
  </monitor>
 </settings>
.
.
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Limiter le partage des tâches et des moteurs

Utilisez cette section pour empêcher les utilisateurs de partager des tâches et des moteurs.

Par défaut, il n'existe aucune limite de partage des tâches et des moteurs. Tous les utilisateurs sont autorisés à partager leurs tâches et leurs connexions de moteur. Si vous voulez changer ce comportement, en empêchant les utilisateurs de partager les tâches et les moteurs, définissez cette propriété sur **true**.

La valeur par défaut de la propriété est **false**. Définissez-la sur **true** pour activer la limite :

limitShareTask
: Définissez cette valeur sur true pour éviter que les utilisateurs partagent des tâches.

limitShareEngine
: Définissez cette valeur sur true pour éviter que les utilisateurs partagent des connexions de moteurs.

```
<?xml version"1.0"?>
<tdwc>
.
.
 <settings>
  <security>
   <property name="limitShareTask"     value="false" />
   <property name="limitShareEngine"   value="false" />
  </security>
 </settings>
.
.
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Afficher toutes les dépendances

Cette section détermine s'il faut afficher toutes les dépendances affichées, qu'elles soient satisfaites ou non.

ShowDependencies
: Lorsque vous ouvrez le panneau de dépendances à partir des résultats de Surveiller les travaux et Tâches de surveillance des flots de travaux, seules les dépendances **Non satisfait** s'affichent par défaut. Supprimez la mise en commentaire de cette section et laissez la valeur définie à "**true**" pour afficher toutes les dépendances, qu'elles soient satisfaites ou non. Les valeurs possibles sont les suivantes :

  true Toutes les dépendances sont affichées, qu'elles soient satisfaites ou non.

  false Seules les dépendances non satisfaites sont affichées.

```
<?xml version"1.0"?>
<tdwc>

.
.
 <settings>
  <ShowDependencies>
   <property name = "AlwaysShowAllDependencies"
             value="true"></property>
  </ShowDependencies>
```

```
 </settings>
.
.
 </tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Audit des activités d'application mobile

Cette section définit si les activités réalisées dans les applications Catalogue libre-service et Tableaux de bord libre-service sont consignées ou non dans un fichier journal d'audit.

Pour plus d'informations sur le nom et l'emplacement du fichier journal, consultez les journaux et la section de traces dans le*Guide d'identification et de résolution des problèmes*.

SSAuditing
Cette valeur est définie par défaut sur "**true**", si bien que les opérations effectuées dans les applications Catalogue libre-service et Tableaux de bord libre-service sont consignées dans un fichier journal. Ce fichier journal contient des informations telles que les dates de création, de modification et de suppression, les opérations effectuées sur les applications mobiles ainsi que l'utilisateur effectuant ces opérations. Les valeurs possibles sont les suivantes :

true Les opérations effectuées dans les applications Catalogue libre-service et Tableaux de bord libre-service sont consignées dans un fichier journal d'audit.

false Les opérations effectuées dans les applications Catalogue libre-service et Tableaux de bord libre-service ne sont pas consignées dans un fichier journal d'audit.

SSAuditingLogSize
Taille maximale, en kilooctets, d'un fichier journal. Lorsqu'un fichier journal atteint sa taille maximale, un nouveau fichier journal est créé. Par défaut, la taille maximale d'un fichier journal est de 100 Ko.

SSAuditingLogFiles
Nombre par défaut de fichiers journaux à créer. Lorsque ce nombre est atteint et que le dernier fichier journal arrive à sa taille maximale, le système supprime le fichier journal le plus ancien et en crée un nouveau.

```
<?xml version"1.0"?>
<tdwc>
.
.
 <settings>
 <SSCMAuditing>
            <property name = "SSAuditing"         value="true"></property>
            <property name = "SSAuditingLogSize"  value="100"></property>
            <property name = "SSAuditingLogFiles" value="2"</property>
 </settings>
 .
 .
</tdwc>
```

Voir «Modèle de TdwcGlobalSettings.xml», à la page 128 pour afficher la syntaxe complète du fichier.

Modèle de TdwcGlobalSettings.xml

Ce que suit est un exemple de fichier :

```
 <?xml version="1.0"?>
 <tdwc>
################################################################################
################  PARAMETRES POUR TOUS LES UTILISATEURS                  ######################
################################################################################
<settings>
################################################################################
################  SECTION 1 - REMPLACEMENT DES LIMITES DE VUE GRAPHIQUE#####################
################################################################################
-->
<!--
 Cette section spécifie le nombre maximal d'objets présentés dans chaque vue graphique.
 La valeur par défaut est 1000 pour toutes les propriétés.
  -->
<!--
  <graphViews>
  <property name="planViewMaxJobstreams" value="1000"></property>
<property name="jobstreamViewLimit" value="1000"></property>
<property name="impactViewLimit" value="1000"></property>
</graphViews>

################################################################################
###################  SECTION 2 - VUE PLAN DANS UNE NOUVELLE FENETRE  ###########################
################################################################################
-->
<!--
Cette section permet d'éviter que Internet Explorer 7 ne se fige pendant l'utilisation de la vue Plan.
 Pour résoudre le problème, définissez la valeur sur True. La valeur par défaut est false.
-->
<graphViews>
<property name="planViewNewWindow" value="true"/>
</graphViews> <!--
################################################################################
################  SECTION 3 - DESACTIVATION/ PERSONNALISATION DE LA FONCTION NEWSFEED  ###############
################################################################################
-->
<!--
Cette section permet de remplacer les propriétés concernant la fonction"NewsFeed".
Les valeurs par défaut sont les suivantes :
<NewsFeed>
<property name="FeedURL" value="https://www.ibm.com/developerworks/community/wikis/form/
   anonymous/api/wiki/585f5525-a7f5-48ef-9222-50ad582e85f4/page/e599dd3c-8dc3-4ab6-89fd-
   33f81a994799/attachment/de677e63-5a9d-46db-a010-18ca38f05812/media/tws.jsonp"
<property name="FeedType" value="JSONP" />
<property name="PollInterval" value="3600" />
</NewsFeed>
-->
<!--
Pour désactiver la fonction
-->
<!--
<NewsFeed>
<property name="FeedURL" value="" />
<property name="FeedType" value="JSONP" />
<property name="PollInterval" value="3600" />
</NewsFeed>
-->
<!--
################################################################################
############  SECTION 4 - DESACTIVATION / PERSONNALISATION DE LA CREATION DE TACHES PREDEFINIES ###########
################################################################################
-->
<!--
Pour éviter ou personnaliser la création de tâches prédéfinies à la première connexion.
Les valeurs possibles sont les suivantes :
all             les tâches réparties et z/OS sont créées. Il s'agit de l'option par défaut.
none            aucune tâche n'est créée
distributed     seules les tâches distribuées sont créées
zos             seules les tâches z/OS sont créées
-->
<!--
<application>
<property name="precannedTaskCreation" value="all"/>
  </application>
-->

<!--
################################################################################
############  SECTION 5 - AJOUT D'UNE URL DE DOCUMENTATION PERSONNALISEE AU TRAVAIL/FLOT DE TRAVAUX  ###########
################################################################################
-->
<!--
Cette section présente des adresses URL qui vous permettent de mémoriser de la
documentation personnalisée sur vos travaux ou vos flot de travaux. Par défaut,
ce paramètre n'est pas spécifié. Pour associer une documentation personnalisée à
un travail ou un flot de travaux, utilisez ce paramètre afin de spécifier l'adresse externe
qui contient ces informations. Si vous souhaitez spécifier une adresse URL à ouvrir en tant
que documentation pour un travail et un flot de travaux, supprimez le caractère de commentaire des
   lignes de la section et une nouvelle action, Ouvrir documentation, est insérée dans le menu
Actions supplémentaires dans les tâches de surveillance des travaux et des flots de travaux.
La nouvelle action est liée à l'adresse URL spécifiée.

Vous pouvez personnaliser le modèle d'URL en utilisant des variables. La syntaxe des variables est
                  ${<variable_name>}

Pour obtenir la liste complète des variables, voir la documentation.

-->
<!--
  <twsObjectDoc>
<property name="jobstreamUrlTemplate" value="http://www.yourhost.com/tws/docs/jobstream/${js_name_w}" />
<property name="jobUrlTemplate" value="http://www.yourhost.com/docs/jobs/${job_name_w}"/>
```

```
  <property name="customActionLabel"    value="Custom Action" />
 </twsObjectDoc>
 -->

 <!--
 ###############################################################################
 ############  SECTION 6 - REGISTRE D'UTILISATEURS                  ############
 ###############################################################################
   Dans cette section, vous pouvez configurer certaines propriétés relatives au registre
 d'utilisateurs utilisé. La propriété groupIdMap correspond aux groupes de registre
 d'utilisateurs et elle peut être modifiée pour mapper et afficher la valeur indiquée
 de chaque groupe. Le nom commun du groupe s'affiche par défaut.
 -->
 <!--
 <security>
 <property name="groupIdMap" value="cn"></property>
 </security>
     -->
 <!--
    ###############################################################################
 ############  SECTION 7 - CONNEXIONS HTTP Z/OS                     ############
 ###############################################################################

 Utilisez cette section pour augmenter ou réduire le délai d'expiration de la connexion
 http dans l'environnement Z/OS. Modifiez ce paramètre si vous recevez un délai
 d'expiration de connexion à l'aide des actions/valeurs.

 Le paramètre est exprimé en millisecondes.
 -->
  <!--
 <http>
 <property name="zosHttpTimeout" value="90000" />
 </http>
     -->
 <!--
   ###############################################################################
 #########  SECTION 8 -  LIMITE DU NOMBRE D'OBJETS RENVOYES DANS LES REQUETES ###########
 ###############################################################################
 <!--
 Dans cette section, vous pouvez configurer le nombre de résultats affichés pour les
 tâches de surveillance, le nombre maximal de lignes à afficher sur chaque page
 ainsi que le nombre de requêtes directes à conserver dans l'historique.
 Ce paramètre s'applique à toutes les tâches, sauf à la surveillance des travaux
 critiques et des travaux exécutés sur des moteurs multiples.
  Si vous souhaitez limiter le nombre de résultats générés par vos requêtes,
  vous pouvez spécifier le nombre maximal d'éléments à extraire. La valeur par
  défaut est -1 ; toute valeur inférieure à 0 indique qu'il n'existe pas de limite
  dans le nombre d'objets extraits. Le nombre minimum de résultats extraits est 500.
  Parce que les données sont extraites par blocs de 250 lignes, la valeur que vous
  entrez est réglée pour compléter un bloc entier. Par exemple, si vous indiquez
  une limite de 500, seuls 500 éléments sont extraits, tandis que si vous indiquez
  une limite de 600, 750 éléments sont extraits.
  Pour définir le nombre maximal de lignes à afficher dans une vue Table, configurez
  la propriété maxRowsToDisplay.
  Pour définir le nombre maximal de requêtes directes à conserver dans l'historique,
  configurez la propriété maxHistoryCount. Ces requêtes sont disponibles dans le
  menu déroulant de la zone Requête, sur la page Requête directe.

 <monitor>
    <property name="monitorMaxObjectsPM" value="2000"></property>
  </monitor>

  <monitor>
    <property name="maxRowsToDisplay" value="25"></property>
  </monitor>

  <monitor>
    <property name="maxHistoryCount" value="100"></property>
  </monitor>
 -->


 <!--

   ###############################################################################
 #########  SECTION 9 -  LIMITE DU PARTAGE DES TACHES ET DES MOTEURS         ###########
 ###############################################################################

 Utilisez cette section pour empêcher les utilisateurs de partager des tâches et des moteurs.
  Par défaut, il n'existe aucune limite de partage des tâches et des moteurs.
  Tous les utilisateurs sont autorisés à partager leurs tâches et leurs connexions de moteur.
  Si vous voulez changer ce comportement en empêchant les utilisateurs de partager les
  tâches et les moteurs, définissez cette propriété sur True. La valeur par
  défaut de la propriété est faux, définissez-la sur Vrai pour activer la limite :

 -->
 <!--
 <security>
 <property name="limitShareTask"     value="false" />
 <property name="limitShareEngine"   value="false" />
   </security>
 -->


<!--

###############################################################################
#########  SECTION 10 -  MODIFICATION DU COMPORTEMENT PAR DEFAUT POUR LE PANNEAU DEPENDANCES  ###########
###############################################################################

 Cette section permet de modifier le comportement par défaut de l'interface utilisateur
 lors de l'affichage des dépendances dans le panneau Dépendances. Lorsque vous
 définissez cette valeur à true par défaut, toutes les dépendances sont
 affichées, et pas seulement celles non satisfaites.
-->
<!--
```

```
 <ShowDependencies>
       <property name = "AlwaysShowAllDependencies"
 value="true"></property>
    </ShowDependencies>
-->
    ##############################################################################
    ########  SECTION 11 -  MODIFICATION DU COMPORTEMENT PAR DEFAUT POUR LA FONCTION D'AUDIT SSC ET SSD   ###########
    ##############################################################################

   Cette section permet de modifier le comportement par défaut du contrôle des activités
   réalisées à l'aide des applications de catalogue libre-service et de tableau
   de bord libre-service. La fonction d'audit est activée par défaut.
   Vous pouvez également définir la taille maximale d'un fichier journal avant
   la création d'un nouveau journal, ainsi que le nombre maximal des fichiers journaux conservés.
-->
   <!--  <SSCMAuditing>
            <property name = "SSAuditing"         value="true"></property>
            <property name = "SSAuditingLogSize"  value="100"></property>
            <property name = "SSAuditingLogFiles" value="2"</property>

-->

 </settings>

 <!--
 ##############################################################################
 ######## PARAMETRES POUR TOUS LES utilisateurs TWSWEBUIAdministrators #################
 ##############################################################################
 -->
 <settings role="TWSWEBUIAdministrator">
 <!-- Insérez ici le paramètre à appliquer seulement aux utilisateurs dont
 le rôle est TWSWEBUIAdministrator  -->
 </settings>
 <!--
 ##############################################################################
 #######  PARAMETRES POUR TOUS LES utilisateurs TWSWEBUIOperators ######################
 ##############################################################################
 -->
 <settings role="TWSWEBUIOperator">
 </settings>
 <!--
 ##############################################################################
 #######  PARAMETRES POUR TOUS LES utilisateurs TWSWEBUIConfigurator ###################
 ##############################################################################
 -->
 <settings role="TWSWEBUIConfigurator">
 </settings>
 <!--
 ##############################################################################
 #######  PARAMETRES POUR TOUS LES utilisateurs TWSWEBUIDeveloper    ###################
 ##############################################################################
 -->
 <settings role="TWSWEBUIDeveloper">
 </settings>
 <!--
 ##############################################################################
 #######  PARAMETRES POUR TOUS LES UTILISATEURS TWSWEBUIAnalyst      ###################
 ##############################################################################
 -->
 <settings role="TWSWEBUIAnalyst">
 </settings>

</tdwc>
```

Configuration de la haute disponibilité pour Dynamic Workload Console

Vous pouvez configurer la haute disponibilité pour les noeuds de portail avec des configurations identiques afin de répartir uniformément les sessions utilisateur.

En optimisant la configuration de haute disponibilité sur Dashboard Application Services Hub, il est possible de répondre aux exigences de haute disponibilité pour Dynamic Workload Console. Par conséquent, les rubriques suivantes expliquent comment définir la configuration de haute disponibilité pour Dashboard Application Services Hub et comment la personnaliser pour Dynamic Workload Console.

La haute disponibilité est idéale pour les installations Dashboard Application Services Hub ayant de nombreux utilisateurs. Lorsqu'un noeud échoue, de nouvelles sessions utilisateur sont dirigées vers d'autres noeuds actifs.

Vous pouvez créer une configuration haute disponibilité à partir d'une instance de Dashboard Application Services Hub autonome existante mais vous devez

importer ses données avant de la configurer pour la haute disponibilité. Les données sont ensuite importées vers un des noeuds pour être répliquées vers les autres noeuds.

La charge de travail est distribuée par session et non par requête. Si un noeud échoue, les utilisateurs qui ont une session ouverte avec ce noeud doivent s'y reconnecter pour accéder à Dashboard Application Services Hub. Tout travail n'ayant pas été sauvegardé ne peut pas être récupéré.

Données synchronisées

Une fois la haute disponibilité configurée, les modifications effectuées dans Dynamic Workload Console qui sont stockées dans des référentiels globaux sont synchronisées sur tous les noeuds de la configuration à l'aide d'une base de données commune. Suite aux actions suivantes, les modifications apportées aux référentiels globaux sont utilisées par Dynamic Workload Console. La plupart de ces modifications sont générées par des actions du dossier **Settings** de la fenêtre de navigation de la console.

- Création, restauration, modification ou suppression d'une page.
- Création, restauration, modification ou suppression d'une vue.
- Création, modification ou suppression d'un profil de préférence ou déploiement de profils de préférence à partir de la ligne de commande.
- Copie d'une entité de portlet ou suppression d'une copie de portlet.
- Modification de l'accès à une entité de portlet, une page, une adresse URL externe ou une vue.
- Création, modification ou suppression d'un rôle.
- Modifications des préférences ou des valeurs par défaut de portlet.
- Modifications issues des applications **Utilisateurs et groupes**, incluant l'affectation d'utilisateurs et de groupes à des rôles.

Remarque : Les référentiels globaux ne doivent jamais être mis à jour manuellement.

En mode de fonctionnement normal, dans une configuration haute disponibilité, les mises à jour qui nécessitent une synchronisation sont d'abord validées par rapport à la base de données. Au même moment, le noeud qui soumet la mise à jour pour les référentiels globaux signale la modification à tous les autres noeuds dans la configuration haute disponibilité. Au fur et à mesure que les noeuds reçoivent la notification, ils obtiennent les mises à jour à partir de la base de données et valident la modification dans la configuration locale.

Si la validation des données échoue sur un noeud, quel qu'il soit, un message d'avertissement est consigné dans le fichier journal. Le noeud ne peut pas effectuer ses propres mises à jour dans la base de données. Le redémarrage de l'instance Dashboard Application Services Hub sur le noeud résout la plupart des problèmes de synchronisation. Sinon, supprimez le noeud de la configuration haute disponibilité pour corriger le problème.

Remarque : Si le serveur de base de données redémarre, toutes ses connexions à la configuration haute disponibilité sont perdues. La restauration des connexions et les opérations de mise à jour par les utilisateurs, telles que la modification ou la création de vues ou de pages, peuvent prendre cinq minutes.

Synchronisation manuelle et mode de maintenance

Les mises à jour visant à déployer, redéployer ou supprimer des modules de console ne sont pas synchronisées automatiquement dans la configuration haute disponibilité. Ces modifications doivent être effectuées manuellement sur chaque noeud. Pour les opérations de déploiement et de redéploiement, le package du module de console doit être identique sur chaque noeud.

Lorsqu'une des commandes de déploiement est démarrée sur le premier noeud, le système passe en *mode maintenance* et vous ne pouvez plus apporter de modifications aux référentiels globaux. Une fois les modifications du déploiement terminées sur chaque noeud, le système revient à l'état déverrouillé. Il n'y aucune restriction applicable à l'ordre de déploiement, de suppression ou de redéploiement des modules sur chacun des noeuds.

En mode maintenance, aucune modification affectant les référentiels globaux ne peut être effectuée dans le portail et un message d'erreur est renvoyé. Ensuite, seules les modifications relatives aux préférences du portlet personnel d'un utilisateur sont autorisées dans les référentiels globaux. Toute modification effectuée hors du contrôle du portail, par exemple, la soumission d'un formulaire d'un portlet vers une application distante, est traitée normalement.

De même, les opérations suivantes ne sont pas synchronisées dans la configuration haute disponibilité et doivent être effectuées manuellement sur chaque noeud. Ces mises à jour ne placent pas la configuration haute disponibilité en mode maintenance.

- Déploiement, redéploiement et suppression de connexions et de transformations.
- Modifications de personnalisation apportées à l'interface utilisateur Dynamic Workload Console (par exemple, images personnalisées ou feuilles de style) à l'aide de `consoleProperties.xml`.

Pour éviter que les utilisateurs établissent des sessions avec des noeuds ayant différentes définitions de connexion et de transformations, ou personnalisations d'interface utilisateur, planifiez ces modifications pour qu'elles coïncident avec les déploiements du module de console.

Conditions requises

Les conditions suivantes doivent être remplies pour que la haute disponibilité puisse être activée :

- Si vous créez une configuration haute disponibilité à partir d'une instance autonome de Dynamic Workload Console, vous devez exporter ses données pour pouvoir la configurer pour la haute disponibilité. Lorsque vous avez configuré les noeuds, vous pouvez importer les données sur un des noeuds afin de les répliquer sur les autres noeuds.
- Le protocole LDAP (Lightweight Directory Access Protocol) doit être installé et configuré comme référentiel utilisateur pour chaque noeud de la configuration haute disponibilité. Pour savoir quels serveurs LDAP utiliser, reportez-vous à la liste des logiciels pris en charge pour WebSphere Application Server version 7.0. Pour savoir comment activer le protocole LDAP pour chaque noeud, voir comment configurer les registres d'utilisateurs LDAP.
- Mettez à jour les services WebSphere Application Server avec le nouvel administrateur en indiquant le nouvel ID utilisateur LDAP *utilisateur_WAS* et le

nouveau mot de passe LDAP *password_utilisateur_WAS*. Pour plus d'informations sur la mise à jour des services WebSphere Application Server, voir «updateWasService», à la page 416.

- DB2 version 9.7 doit être installé dans le réseau en vue de synchroniser les référentiels globaux pour la configuration haute disponibilité de Dynamic Workload Console.
- Chaque noeud de la configuration haute disponibilité doit être activé pour utiliser le même protocole LDAP avec la même configuration d'utilisateur et de groupe.
- Tous les noeuds Dynamic Workload Console de la configuration haute disponibilité doivent être installés dans le même nom de cellule. Après l'installation de Dynamic Workload Console sur chaque noeud, utilisez le paramètre **-cellName** dans la commande **manageprofiles**.
- Tous les noeuds Dynamic Workload Console de la configuration haute disponibilité doivent avoir des horloges synchronisées.
- Les versions de WebSphere Application Server et de Dashboard Application Services Hub doivent être au même niveau d'édition, y compris les groupes de correctifs. Les correctifs et les mises à niveau pour l'exécution doivent être appliqués manuellement sur chaque noeud.
- Avant de joindre des noeuds à une configuration haute disponibilité, assurez-vous que chaque noeud utilise le même ID utilisateur de référentiel de fichiers auquel le rôle *iscadmins* a été attribué.

Exportation de données à partir d'un serveur autonome

Vous pouvez exporter des données à partir d'une instance de serveur d'applications existante pour créer un fichier de données qui peut être importé dans une configuration haute disponibilité.

Si vous définissez une configuration haute disponibilité sur des noeuds existants vous devez commencer par exporter toutes les données à partir de l'instance autonome puis les importer une fois la configuration haute disponibilité définie.

Remarque : Si vous joignez le serveur à une configuration haute disponibilité existante, les autres noeuds ne doivent pas contenir de données personnalisées, c'est à dire que chaque noeud doit être une nouvelle installation. Lorsque vous importez des données à partir du serveur autonome il est répliqué sur tous les autres noeuds.

Pour exporter les paramètres à partir de Dynamic Workload Console, suivez la procédure ci-dessous.

1. Connectez-vous à Dynamic Workload Console à l'aide des données d'identification TWSWEBUIAdministrator.
2. Développez le portefeuille en cliquant sur **Tivoli Workload Scheduler** > **Paramètres** > **Gestion des paramètres**.
3. Dans le panneau Gestion des paramètres, cliquez sur **Exporter les paramètres** et sauvegardez le fichier XML dans le répertoire de votre choix.
4. Créez une configuration de haute disponibilité à l'aide du serveur autonome ou joignez-la à une configuration existante.
5. Importez les données précédemment exportées sur tout noeud de la configuration haute disponibilité en procédant comme suit :

   Dans le panneau Gestion des paramètres, cliquez sur **Importer les paramètres** et recherchez le fichier XML contenant les données que vous voulez importer.

Créez une configuration haute disponibilité à l'aide du serveur d'applications autonome ou joignez-la à une configuration existante. Une fois la configuration haute disponibilité terminée vous pouvez importer le fichier de données sur un des noeuds.

Définition d'une configuration haute disponibilité

Vous pouvez configurer une instance Dynamic Workload Console en vue d'utiliser une base de données comme référentiel de fichiers au lieu d'un répertoire local.

Si vous créez une configuration haute disponibilité à partir d'une instance Dynamic Workload Console existante contenant des données personnalisées, vérifiez que vous avez exporté ses données avant de commencer sa configuration pour la haute disponibilité. Une fois la configuration effectuée, vous pouvez importer les données vers un des noeuds de la nouvelle configuration.

Dynamic Workload Console est installé sur une machine à l'aide du nom de cellule désigné pour tous les noeuds de console dans la configuration haute disponibilité. Vous avez installé et configuré un répartiteur réseau (par exemple, IBM HTTP Server), DB2 et LDAP comme décrit dans «Conditions requises», à la page 132.

Remarque : Le serveur Dynamic Workload Console doit être configuré avec DB2 sans connexion SSL. Si vous voulez configurer une connexion SSL, vous pouvez l'activer après avoir activé la configuration haute disponibilité. Pour de plus amples informations, voir «Activation de SSL pour le serveur Dashboard Application Services Hub», à la page 144.

Pour configurer la configuration haute disponibilité, suivez la procédure ci-dessous :

1. Sur la machine où est installé DB2, créez une base de données DB2 (voir Création de base de données).
2. Vérifiez que vous disposez d'un pilote JDBC pour DB2 sur l'ordinateur où Dynamic Workload Console est installé. Le pilote JDBC doit être disponible dans : `rép_install JazzSM/lib/db2`.
3. Dans une invite de commande, accédez au répertoire `rép_install JazzSM/ui/bin/ha` et éditez les paramètres dans `tipha.properties`.

Tableau 29. Propriétés de Dashboard Application Services Hub

Nom de la propriété	Description
DBHost	Nom d'hôte ou adresse IP de la machine sur laquelle la base DB2 est installée. **Exemple :** `tipdb.cn.ibm.com`
DBPort	Numéro de port du serveur DB2. **Exemple :** `50000` (valeur par défaut)
DBName	Nom de la base de données que vous avez créée. **Exemple :** `tipdb`
DBProviderClass	Nom de classe du fournisseur DB2. **Exemple :** `com.ibm.db2.jcc.DB2Driver` (valeur par défaut)
DBProviderName	Nom du fournisseur DB2. **Exemple :** `TIP_Universal_JDBC_Driver` (valeur par défaut)
DBDatasource	Nom JNDI de la source de données. **Exemple :** `jdbc/tipds`
DBDatasourceName	Nom de la source de données utilisée. **Exemple :** `tipds`

Tableau 29. Propriétés de Dashboard Application Services Hub (suite)

Nom de la propriété	Description
DBHelperClassName	Nom de classe DB2 Helper **Exemple :** `com.ibm.websphere.rsadapter.DB2UniversalDataStoreHelper` (default)
DBDsImplClassName	Nom de classe d'implémentation de source de données DB2. **Exemple :** `com.ibm.db2.jcc.DB2ConnectionPoolDataSource` (default)
DBDriverVarName	Nom de variable d'environnement WebSphere pour le chemin d'accès aux classes de pilote JDBC DB2. **Exemple :** `TIP_JDBC_DRIVER_PATH`
DBJDBCDriverPath	Emplacement des bibliothèques de pilote JDBC DB2 (par exemple, `db2jcc.jar`). **Exemple :** `rép_install JazzSM/lib/db2`
DBDriverType	Type de pilote JDBC. **Exemple :** `4` (valeur par défaut)
DBType	Type de base de données **Exemple :** `DB2` (valeur par défaut)
JaasAliaseName	Nom d'alias JAAS utilisé pour stocker le nom utilisateur et le mot de passe de base de données. **Exemple :** `TIPAlias` (valeur par défaut)
JaasAliasDesc	Description du nom d'alias JAAS **Exemple :** `Alias JAAS utilisé pour la configuration haute disponibilité`
LocalHost	Nom d'hôte ou adresse IP de la machine sur laquelle la console est en cours d'exécution. Les options LocalHost et LocalPort identifient de manière unique le noeud dans la configuration haute disponibilité. **Exemple :** `tip01.cn.ibm.com`
LocalPort	Port sécurisé de la console d'administration. Les options LocalHost et LocalPort identifient de manière unique le noeud dans la configuration haute disponibilité. **Exemple :** Lorsque Dynamic Workload Console est installé avec des ports par défaut, la valeur de cette propriété dans `tipha.properties` doit être 16311.
WasRoot	Chemin complet d'accès au système où le serveur d'applications et les images de console ont été extraites pendant l'installation. **Exemple :** `/opt/IBM/WebSphere/AppServer`
ProfileName	Nom de profil indiqué dans la commande **manageprofiles** après l'installation. Si aucun nom de profil n'a été indiqué, la valeur par défaut est utilisée. **Exemple :** `JazzSMProfile` (valeur par défaut)
CellName	Nom de cellule qui a été indiqué dans la commande **manageprofiles** après l'installation. Si aucun nom de cellule n'a été indiqué, la valeur par défaut est utilisée. **Exemple :** `JazzSMNode01Cell` (valeur par défaut)Ce paramètre est facultatif pour une installation de console sur un noeud unique. Cependant, pour la configuration haute disponibilité, il est nécessaire de s'assurer que tous les noeuds utilisent le même nom de cellule.
NodeName	Nom du noeud de serveur d'applications **Exemple :** `JazzSMNode01` (valeur par défaut)

Tableau 29. Propriétés de Dashboard Application Services Hub (suite)

Nom de la propriété	Description
ServerName	Nom d'instance WebSphere Application Server **Exemple :** server1 (valeur par défaut)
IscAppName	Nom d'application d'entreprise Dashboard Application Services HubDashboard Application Services Hub. L'application d'entreprise Dashboard Application Services Hub est installée dans le répertoire suivant : *rép_profil_JazzSM*/installedApps/${CellName}/${IscAppName}.ear **Exemple :** isc (valeur par défaut)
LoggerLevel	Niveau de consignation requis. La valeur par défaut est OFF. **Exemple :** FINER
HAEnabled	Indique qu'une configuration haute disponibilité est activée. **Avertissement :** Ne modifiez pas cette valeur manuellement.
TipHome	Indiques le répertoire de base Dashboard Application Services Hub spécifié pendant l'installation. **Exemple :** rép_install JazzSM/ui
ProfilePath	Indique le répertoire de profil JazzSM spécifié pendant l'installation. **Exemple :** rép_install JazzSM/profile

4. Dans le répertoire *JazzSM_profile_dir*/bin, suivant votre système d'exploitation, entrez une des commandes suivantes :
   - stopServer.bat server1
   - stopServer.sh server1

   Remarque : Sur les systèmes UNIX et Linux, vous êtes invité à fournir un nom utilisateur et un mot de passe d'administrateur.
5. Vérifiez que votre base de données est vide et que le serveur n'est pas démarré. Des problèmes peuvent survenir si vous tentez de définir une configuration haute disponibilité sur une base de données non vide ou sur un serveur actif.
6. Dans une invite de commande, accédez au répertoire rép_install JazzSM/ui/bin/ha et entrez la commande suivante :
   - *JazzSM_profile_dir*\bin\ws_ant.bat -f install.ant configHA -Dusername=*nom_utilisateur_DB2* -Dpassword=*mot_de_passe_DB2* -DWAS_username=*utilisateur_ldap* -DWAS_password =*mot_de_passe_ldap*
   - *JazzSM_profile_dir*/bin/ws_ant.sh -f install.ant configHA -Dusername=*nom_utilisateur_DB2* -Dpassword=*mot_de_passe_DB2* -DWAS_username=*utilisateur_ldap* -DWAS_password =*mot_de_passe_ldap*
7. Dans le répertoire *JazzSM_profile_dir*/bin, suivant votre système d'exploitation, entrez une des commandes suivantes :
   - startServer.bat server1
   - startServer.sh server1

La configuration haute disponibilité est créée et le noeud de console est joint à la configuration comme premier noeud.

Ajoutez (ou joignez) ensuite des noeuds supplémentaires à la configuration.

Jointure d'un noeud à une configuration haute disponibilité

Vous pouvez configurer Dynamic Workload Console en vue de joindre une configuration haute disponibilité.

1. Si vous joignez une instance Dynamic Workload Console autonome à une configuration haute disponibilité, vous devez au préalable exporter toutes ses données. Une fois la jointure à la configuration haute disponibilité effectuée, vous pouvez importer les données précédemment exportées. Les autres noeuds de la configuration ne doivent pas contenir de données personnalisées et doivent être des instances nouvellement installées.
2. Assurez-vous que la configuration haute disponibilité est activée à l'aide de la procédure fournie dans «Définition d'une configuration haute disponibilité», à la page 134.
3. Vérifiez que Dynamic Workload Console est installé sur le noeud à l'aide du même nom de cellule que celui indiqué pour la configuration.
4. Vérifiez que tous les modules de console déployés dans la configuration haute disponibilité sont déjà déployés sur le noeud que vous joignez à la configuration.
5. Déployez toute connexion ou transformation utilisée par les noeuds dans la configuration haute disponibilité.
6. Si la configuration haute disponibilité utilise des modifications de personnalisation du fichier `consoleProperties.xml`, copiez ces modifications et ce fichier au même emplacement sur le noeud que vous joignez à la configuration.
7. Vérifiez que le noeud est configuré pour le même protocole LDAP et avec les mêmes définitions d'utilisateur et de groupe que tous les autres noeuds de la configuration haute disponibilité.

Les paramètres suivants sont utilisés sur l'option `join` lorsqu'un noeud est ajouté :

- **`-Dusername`** - indiquez le nom d'administrateur DB2
- **`-Dpassword`** - indiquez le mot de passe d'administrateur DB2

Pour joindre un noeud nouvellement ajouté, suivez la procédure ci-dessous :

1. Vérifiez que vous disposez d'un pilote JDBC pour DB2 sur l'ordinateur où Dynamic Workload Console est installé. Le pilote JDBC doit être disponible dans : `rép_install JazzSM/lib/db2`.
2. Dans une invite de commande, accédez au répertoire `rép_install JazzSM/ui/bin/ha` et éditez les paramètres dans `tipha.properties`.

Tableau 30. Propriétés de Dashboard Application Services Hub

Nom de la propriété	Description
`DBHost`	Nom d'hôte ou adresse IP de la machine sur laquelle la base DB2 est installée. **Exemple :** `tipdb.cn.ibm.com`
`DBPort`	Numéro de port du serveur DB2. **Exemple :** `50000` (valeur par défaut)
`DBName`	Nom de la base de données que vous avez créée. **Exemple :** `tipdb`
`DBProviderClass`	Nom de classe du fournisseur DB2. **Exemple :** `com.ibm.db2.jcc.DB2Driver` (valeur par défaut)
`DBProviderName`	Nom du fournisseur DB2. **Exemple :** `TIP_Universal_JDBC_Driver` (valeur par défaut)

Tableau 30. Propriétés de Dashboard Application Services Hub (suite)

Nom de la propriété	Description
DBDatasource	Nom JNDI de la source de données. **Exemple :** `jdbc/tipds`
DBDatasourceName	Nom de la source de données utilisée. **Exemple :** `tipds`
DBHelperClassName	Nom de classe DB2 Helper **Exemple :** `com.ibm.websphere.rsadapter.DB2UniversalDataStoreHelper` (default)
DBDsImplClassName	Nom de classe d'implémentation de source de données DB2. **Exemple :** `com.ibm.db2.jcc.DB2ConnectionPoolDataSource` (default)
DBDriverVarName	Nom de variable d'environnement WebSphere pour le chemin d'accès aux classes de pilote JDBC DB2. **Exemple :** `TIP_JDBC_DRIVER_PATH`
DBJDBCDriverPath	Emplacement des bibliothèques de pilote JDBC DB2 (par exemple, `db2jcc.jar`). **Exemple :** `rép_install JazzSM/lib/db2`
DBDriverType	Type de pilote JDBC. **Exemple :** `4` (valeur par défaut)
DBType	Type de base de données **Exemple :** `DB2` (valeur par défaut)
JaasAliaseName	Nom d'alias JAAS utilisé pour stocker le nom utilisateur et le mot de passe de base de données. **Exemple :** `TIPAlias` (valeur par défaut)
JaasAliasDesc	Description du nom d'alias JAAS **Exemple :** `Alias JAAS utilisé pour la configuration haute disponibilité`
LocalHost	Nom d'hôte ou adresse IP de la machine sur laquelle la console est en cours d'exécution. Les options LocalHost et LocalPort identifient de manière unique le noeud dans la configuration haute disponibilité. **Exemple :** `tip01.cn.ibm.com`
LocalPort	Port sécurisé de la console d'administration. Les options LocalHost et LocalPort identifient de manière unique le noeud dans la configuration haute disponibilité. **Exemple :** Lorsque Dynamic Workload Console est installé avec des ports par défaut, la valeur de cette propriété dans `tipha.properties` doit être 16311.
WasRoot	Chemin complet d'accès au système où le serveur d'applications et les images de console ont été extraites pendant l'installation. **Exemple :** `/opt/IBM/WebSphere/AppServer`
ProfileName	Nom de profil indiqué dans la commande **manageprofiles** après l'installation. Si aucun nom de profil n'a été indiqué, la valeur par défaut est utilisée. **Exemple :** `JazzSMProfile` (valeur par défaut)

Tableau 30. Propriétés de Dashboard Application Services Hub (suite)

Nom de la propriété	Description
CellName	Nom de cellule qui a été indiqué dans la commande **manageprofiles** après l'installation. Si aucun nom de cellule n'a été indiqué, la valeur par défaut est utilisée. **Exemple :** JazzSMNode01Cell (valeur par défaut)Ce paramètre est facultatif pour une installation de console sur un noeud unique. Cependant, pour la configuration haute disponibilité, il est nécessaire de s'assurer que tous les noeuds utilisent le même nom de cellule.
NodeName	Nom du noeud de serveur d'applications **Exemple :** JazzSMNode01 (valeur par défaut)
ServerName	Nom d'instance WebSphere Application Server **Exemple :** server1 (valeur par défaut)
IscAppName	Nom d'application d'entreprise Dashboard Application Services HubDashboard Application Services Hub. L'application d'entreprise Dashboard Application Services Hub est installée dans le répertoire suivant : *rép_profil_JazzSM*/installedApps/${CellName}/ ${IscAppName}.ear **Exemple :** isc (valeur par défaut)
LoggerLevel	Niveau de consignation requis. La valeur par défaut est OFF. **Exemple :** FINER
HAEnabled	Indique qu'une configuration haute disponibilité est activée. **Avertissement :** Ne modifiez pas cette valeur manuellement.
TipHome	Indiques le répertoire de base Dashboard Application Services Hub spécifié pendant l'installation. **Exemple :** rép_install JazzSM/ui
ProfilePath	Indique le répertoire de profil JazzSM spécifié pendant l'installation. **Exemple :** rép_install JazzSM/profile

3. Dans le répertoire *JazzSM_profile_dir*/bin, suivant votre système d'exploitation, entrez une des commandes suivantes :
   - stopServer.bat server1
   - stopServer.sh server1

   Remarque : Sur les systèmes UNIX et Linux, vous êtes invité à fournir un nom utilisateur et un mot de passe d'administrateur.
4. Vérifiez que Dashboard Application Services Hub n'est pas en cours d'exécution.
5. Dans une invite de commande, accédez au répertoire *TWA_home*/profiles/ TIPProfile/bin/ha et entrez la commande ci-dessous.
   - ..\ws_ant.bat -f install.ant configHA -Dusername=*nom_utilisateur_DB2* -Dpassword=*mot_de_passe_DB2* -DWAS_username=*utilisateur_ldap* -DWAS_password =*mot_de_passe_ldap*
   - ../ws_ant.sh -f install.ant configHA -Dusername=*nom_utilisateur_DB2* -Dpassword=*mot_de_passe_DB2* -DWAS_username=*utilisateur_ldap* -DWAS_password =*mot_de_passe_ldap*
6. Dans le répertoire *JazzSM_profile_dir*/bin, suivant votre système d'exploitation, entrez une des commandes suivantes :

- `startServer.bat server1`
- `startServer.sh server1`

Le noeud de console est alors joint à la configuration haute disponibilité.

Ajoutez ensuite un autre noeud à la configuration haute disponibilité, ou si vous avez fini d'ajouter des noeuds, activez les relations de confiance entre serveurs pour chaque noeud de la configuration.

En fonction du répartiteur réseau que vous utilisez (par exemple, IBM HTTP Server), vous pourrez avoir d'autres mises à jour à effectuer pour l'acheminement de sessions vers le nouveau noeud. Pour plus d'informations sur votre répartiteur réseau, reportez-vous à la documentation applicable.

Activation d'une relation de confiance de serveur à serveur

Utilisez la procédure ci-dessous pour activer des noeuds afin qu'ils se connectent les uns aux autres et envoient des notifications dans la configuration haute disponibilité.

Pour activer la configuration haute disponibilité entre les noeuds participants, vous devez effectuer les étapes suivantes sur chaque noeud.

1. Dans un éditeur de texte, ouvrez le fichier `ssl.client.props` à partir du répertoire *rép_profil_JazzSM*`/properties`. Le chemin par défaut pour *rép_profil_JazzSM* est `/opt/IBM/JazzSM/profile`.
2. Supprimez la mise en commentaire de la section qui commence par **`com.ibm.ssl.alias=AnotherSSLSettings`** pour qu'elle se présente comme suit :

   ```
   com.ibm.ssl.alias=AnotherSSLSettings
   com.ibm.ssl.protocol=SSL_TLS
   com.ibm.ssl.securityLevel=HIGH
   com.ibm.ssl.trustManager=IbmX509
   com.ibm.ssl.keyManager=IbmX509
   com.ibm.ssl.contextProvider=IBMJSSE2
   com.ibm.ssl.enableSignerExchangePrompt=true
   #com.ibm.ssl.keyStoreClientAlias=default
   #com.ibm.ssl.customTrustManagers=
   #com.ibm.ssl.customKeyManager=
   #com.ibm.ssl.dynamicSelectionInfo=
   #com.ibm.ssl.enabledCipherSuites=
   ```
3. Supprimez la mise en commentaire et modifiez la section qui commence par **`com.ibm.ssl.trustStoreName=AnotherTrustStore`** pour qu'elle se présente comme suit :

   ```
   com.ibm.ssl.trustStoreName=AnotherTrustStore
   com.ibm.ssl.trustStore=${user.root}/etc/trust.p12
   com.ibm.ssl.trustStorePassword=password_magasin_clés_certifiées
   com.ibm.ssl.trustStoreType=PKCS12
   com.ibm.ssl.trustStoreProvider=IBMJCE
   com.ibm.ssl.trustStoreFileBased=true
   com.ibm.ssl.trustStoreReadOnly=false
   ```

   où le mot de passe du magasin de clés certifiées par défaut est `WebAS`.
   Exemple :

   ```
   com.ibm.ssl.trustStore=rép_profil_JazzSM/etc/trust.p12
   com.ibm.ssl.trustStorePassword=WebAS
   com.ibm.ssl.trustStoreType=JKS
   ```

   Remarque : Cet exemple est valide si les certificats Tivoli Workload Scheduler par défaut ont été utilisés. Si vous souhaitez ensuite chiffrer le mot de passe

saisi, exécutez l'outil wastool **encryptProfileProperties script**, tel que décrit dans «Serveur d'applications - chiffrement des fichiers de propriétés du profil», à la page 415.

4. Sauvegardez vos modifications dans `ssl.client.props`.
5. Arrêtez et redémarrez Dashboard Application Services Hub :
   a. Dans le répertoire *JazzSM_profile_dir*/bin, suivant votre système d'exploitation, entrez une des commandes suivantes :
      - `stopServer.bat server1`
      - `stopServer.sh server1`

      Remarque : Sur les systèmes UNIX et Linux, vous êtes invité à fournir un nom utilisateur et un mot de passe d'administrateur.
   b. Dans le répertoire *JazzSM_profile_dir*/bin, suivant votre système d'exploitation, entrez une des commandes suivantes :
      - `startServer.bat server1`
      - `startServer.sh server1`
6. Effectuez toutes ces étapes sur chaque noeud avant de poursuivre avec les étapes restantes.
7. Exécutez la commande suivante sur chaque noeud pour chaque hôte *myremotehost* (c'est-à-dire pour chaque noeud pour lequel vous voulez activer une relation de confiance) dans la configuration haute disponibilité :

   rép_profil_JazzSM\bin\retrieveSigners.bat NodeDefaultTrustStore AnotherTrustStore -host *myremotehost* -port *port_SOAP_distant*

   rép_profil_JazzSM/bin/bin/retrieveSigners.sh NodeDefaultTrustStore AnotherTrustStore -host myremotehost -port *port_SOAP_distant*

   où `myremotehost` est le nom de l'ordinateur avec lequel activer la confiance ; `remote` est le numéro du port de connexion SOAP (16313 est la valeur par défaut). Si vous avez procédé à l'installation avec des ports non définis par défaut, utilisez l'utilitaire showHostProperties pour vérifier le numéro de port SOAP, tel que décrit dans «Modification des propriétés d'un hôte», à la page 420.
8. Arrêtez et redémarrez WebSphere Application Server en entrant les commandes suivantes :
   a. `stopWas.bat -direct -user ldapuser -password ldpapwd` (localisez `stopWas.bat` dans le répertoire `TWA_home\wastools`.)
   b. `startWas.bat -direct -user ldapuser -password ldpapwd` (localisez `startWas.bat` dans le répertoire`TWA_home\wastools`.)

Dans cet exemple, la configuration haute disponibilité comporte deux noeuds Microsoft Windows appelés *myserver1* et *myserver2*. Commande entrée pour *myserver1* :

```
retrieveSigners.bat NodeDefaultTrustStore AnotherTrustStore -host myserver2
 -port 16313
```

Commande entrée pour *myserver2* :

```
retrieveSigners.bat NodeDefaultTrustStore AnotherTrustStore -host myserver1
 -port 16313
```

Puis, entrez l'utilisateur et le mot de passeDynamic Workload Console, lorsque le système vous le demande.

Vérification du bon fonctionnement d'une configuration haute disponibilité

Utilisez les informations fournies dans cette rubrique pour vérifier que votre configuration haute disponibilité Dynamic Workload Console fonctionne correctement après l'ajout de tous les noeuds et l'activation de la relation de confiance entre serveurs.

Cette tâche permet de confirmer que les fonctions suivantes fonctionnent correctement :

- La base de données utilisée pour la configuration haute disponibilité est correctement créée et initialisée.
- Chaque noeud de la configuration utilise la base de données comme référentiel à la place de son propre système de fichiers local.
- La relation de confiance de serveur à serveur est correctement activée entre les noeuds.

Pour vérifier votre configuration haute disponibilité :

1. Vérifiez que chaque instance Dynamic Workload Console sur chaque noeud est en cours d'exécution.
2. Dans un navigateur, connectez-vous à un noeud, créez une vue et sauvegardez vos modifications.
3. Connectez-vous aux noeuds restants et vérifiez que la vue nouvellement créée est disponible dans chacun d'eux.

Configuration de Dynamic Workload Console pour l'utilisation de DB2

Configurez Dynamic Workload Console pour utiliser une base de données comme référentiel de paramètres, pour que toutes les consoles partagent les mêmes paramètres et obtenir ainsi un haut niveau d'évolutivité et de disponibilité.

Vérifiez que vous avez bien configuré Dashboard Application Services Hub (Dynamic Workload Console) pour qu'il fonctionne en mode haute disponibilité et que vous disposez du rôle TWSWEBUIAdministrator.

Afin de configurer Dynamic Workload Console pour l'utilisation et le partage d'une base de données comme référentiel de paramètres, vous devez :

1. Créer une base de données pour Dynamic Workload Console. Vous pouvez également utiliser la base de données créée pour Dashboard Application Services Hub 2.2 mais ce n'est pas recommandé.
2. Configurez éventuellement une connexion SSL entre DB2 et Dynamic Workload Console.
3. Si vous avez configuré une connexion SSL vous devez également l'activer sur Dashboard Application Services Hub. Voir : «Activation de SSL pour le serveur Dashboard Application Services Hub», à la page 144.
4. Définissez la connexion entre la base de données et le serveur Dashboard Application Services Hub. Voir : «Création d'une source de données», à la page 144.
5. Configurez toutes les instances Dynamic Workload Console pour le partage du même référentiel de paramètres. Voir «Partage d'un référentiel de paramètres», à la page 146.

6. Eventuellement, si vous voulez que Dynamic Workload Console accède au référentiel de base de données avec un utilisateur ne disposant pas de privilèges d'administrateur vous devez changer l'utilisateur qui met à jour le référentiel de paramètres sur DB2. Voir «Modification de l'utilisateur Dynamic Workload Console du référentiel de base de données», à la page 147 or «Modification de l'utilisateur Dashboard Application Services Hub du référentiel de base de données», à la page 148.

Pour créer une base de données pour Dynamic Workload Console, procédez comme suit :

1. Ouvrez le centre de contrôle DB2, cliquez avec le bouton droit de la souris sur **All Databases**, et sélectionnez **Create Database** > **Standard**.
2. Dans l'assistant de création de base de données, entrez le nom de la base de données et cliquez sur **Finish** pour accepter toutes les options par défaut.

Configuration de DB2 en mode SSL

Configurez votre serveur DB2 pour le support SSL.

Assurez-vous que vous êtes connecté en tant que propriétaire de l'instance DB2 et avez défini les paramètres de configuration suivants ainsi que la variable de registre DB2COMM.

Utilisez `db2 update dbm cfg` *`parameter_name`* à l'aide de la commande *`parameter_value`* où *parameter_name* est le nom du paramètre à définir et *parameter_value* la valeur du paramètre à définir.

1. Définissez le paramètre de configuration `ssl_svr_keydb` avec le chemin complet du fichier de la base de données de clés. Par exemple, `C:\TWS\installations\tws850cli\TWS\ssl\gskit\TWSClientKeyStore.kdb` où `TWSClientKeyStore.kdb` est le nom complet du fichier de clés qui héberge le certificat DB2 et les certificats sécurisés.

   Remarque : Il doit être reconnu par le certificat WebSphere Application Server JKS. Si `ssl_svr_keydb` a la valeur null (non défini), la prise en charge SSL n'est pas activée.
2. 2. Définissez le paramètre de configuration `ssl_svr_stash` sur le chemin complet du fichier de stockage. Par exemple : `C:\TWS\installations\tws850cli\TWS\ssl\gskit\TWSClientKeyStore.sth`. Si `ssl_svr_stash` a la valeur null (non défini), la prise en charge SSL n'est pas activée.
3. 3. Définissez le paramètre de configuration `ssl_svr_label` sur l'étiquette du certificat numérique du serveur. Si `ssl_sv r_label` n'est pas défini, c'est le certificat par défaut présent dans la base de données de clés qui est utilisé. S'il n'y a pas de certificat par défaut dans cette base de données, SSL n'est pas activée. Par exemple : `client`.
4. 4. Définissez le paramètre de configuration `ssl_svcename` sur le port utilisé par le système de base de données DB2 pour les connexions SSL. Si TCP/IP et SSL sont activés (la variable de registre DB2COMM est définie sur 'TCPIP, SSL'), définissez `ssl_svcename` sur un port différent de celui défini pour `svcename`. Le paramètre de configuration `svcename` définit le port utilisé par le système de base de données de DB2 pour les connexions TCP/IP. Si vous définissez `ssl_svcename` avec le même port que `svcename`, TCP/IP et SSL ne sont ni l'un ni l'autre activés. Si `ssl_svcename` a la valeur null (non défini), la prise en charge SSL n'est pas activée.

   Remarque : Lorsque la variable de registre DB2COMM est définie sur 'TCPIP,SSL', si le support TCPIP n'est pas correctement activé, par exemple, si

le paramètre de configuration svcename est défini sur la valeur NULL, l'erreur SQL5043N est renvoyée et le support SSL n'est pas activé.

5. Ajoutez la valeur SSL à la variable de registre DB2COMM. Par exemple : db2set -i db2inst1 DB2COMM=SSL. Le gestionnaire de base de données peut prendre en charge plusieurs protocoles à la fois. Par exemple, pour activer à la fois les protocoles TCP/IP et SSL, indiquez : db2set -i db2inst1 DB2COMM=SSL,TCPIP où : db2inst1 est le nom d'instance DB2
6. Redémarrez l'instance DB2. Par exemple :

```
db2stop
db2start
```

Activation de SSL pour le serveur Dashboard Application Services Hub

Configuration du serveur Dashboard Application Services Hub pour utilisation de la connexion SSL.

Assurez-vous que la haute disponibilité du serveur Dashboard Application Services Hub sans connexion SSL est configurée.

Pour activer SSL pour le serveur Dashboard Application Services Hub, procédez comme suit :

1. Dans l'interface de ligne de commande, accédez au répertoire {RACINE_TWA}\wastools et exécutez la commande suivante :
   - **Sur les systèmes Windows**

     changeTIPDatasource.bat *nom_source_données* useSsl *port_number*
   - **Sur les systèmes UNIX**

     ./ changeTIPDatasource.sh *nom_source_données* useSsl *port_number*

   où,

   nom_source_données
   Nom JNDI de la source de données utilisée pour la haute disponibilité Dashboard Application Services Hub (indiqué dans tipha.properties par exemple, DBDatasource=jdbc/tipds.

   useSsl Peut être défini sur true ou false. Indiquez **true** pour activer SSL.

   port_number
   Indiquez le numéro de port SSL (même valeur que celle indiquée pour le paramètre ssl_svcename) **Exemple** : ./changeTIPDatasource.sh tipds true 60000.
2. Redémarrez WebSphere Application Server.

Création d'une source de données

Création d'une source de données

1. Modifiez le fichier TDWCDatasource.properties afin d'insérer les valeurs correctes pour les paramètres de connexion à la base de données. Le fichier TDWCDatasource.properties se trouve dans : TWA_home\wastools. Voici un exemple auquel vous pouvez vous référer :

```
################################
# Modèle des propriétés de source de données
################################

# Nom d'hôte du serveur où DB2 est installé
databaseServerName=localhost
```

```
# Port utilisé par DB2
databasePort=50000

# Nom de la base de données à utiliser (doit exister)
databaseName=TDWC

# Si la valeur est true, lorsqu'un fournisseur JDBC ayant le nom fourni est
# localisé dans la configuration, il est supprimé puis recréé.
# Saisissez "true" uniquement la première fois que vous créez la source de données
deleteAndRecreate=false


################################
# Propriétés facultatives
################################

# Utilisez une connexion SSL (mode FIPS). Si la valeur est true, le socket de connexion
# sécurisée est utilisé pour communiquer avec DB2 (la valeur par défaut est false)
#useSslConnection=false

# Nom JNDI à associer à une source de données (à indiquer dans la configuration DWC
#datasourceJndiName=jdbc/TDWC

# Nom du fournisseur JDBC WebSphere à créer
#providerName=tdwcDriver

# Nom de la source de données à créer
#datasourceName=tdwcDatasource

# Nom du noeud WebSphere sur lequel DWC est exécuté.
#nodeName=TIPNode


################################
# Propriétés de pool de connexion
#
# Ces propriétés sont facultatives.
#
# Pour plus d'informations, voir les paramètres de "pool de connexion" dans le centre de
# documentation WAS http://publib.boulder.ibm.com/infocenter/wasinfo/v7r0/index.jsp?topic=/
#com.ibm.webSphere.express.doc/info/exp/ae/udat_conpoolset.html
################################

#connectionTimeout = 180
#maxConnections = 20
#minConnections=1
#reapTime=180
#unusedTimeout=1800
#agedTimeout=0
#purgePolicy=EntirePool
```

2. Dans l'interface de ligne de commande, accédez au répertoire {RACINE_TWA}\wastools et exécutez la commande suivante pour créer une source de données :
   - **Sur les systèmes Windows**

     ```
     installTDWCDataSource.bat TDWCDatasource.properties
     ```

   - **Sur les systèmes UNIX**

     ```
     ./installTDWCDataSource.sh TDWCDatasource.properties
     ```

3. Redémarrez WebSphere Application Server.

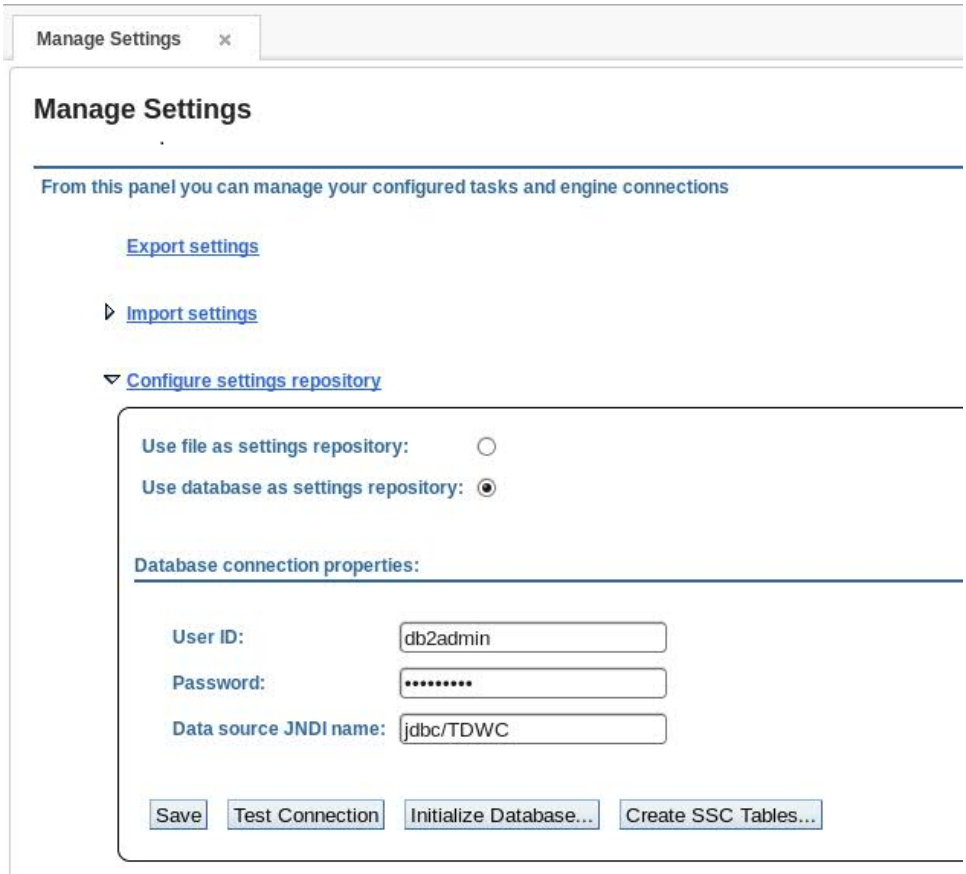
Le fichier TDWCDatasource.properties étant un fichier local vous devez le modifier et exécuter les étapes ci-dessus sur toutes les instances Dynamic Workload Console.

Partage d'un référentiel de paramètres

Cette rubrique explique comment partager un référentiel des paramètres entre plusieurs instances de Dynamic Workload Console.

Pour effectuer cette tâche vous devez disposer du rôle TWSWEBUIAdministrator.

1. Assurez-vous que toutes les instances Dynamic Workload Console qui doivent partager le même référentiel de paramètres utilisent également le même registre d'utilisateurs.
2. Dans Dynamic Workload Console, cliquez sur **Paramètres** > **Gestion des paramètres**.
3. Dans le même panneau, cliquez sur **Configurer le référentiel des paramètres** > **Utiliser la base de données comme référentiel des paramètres** pour indiquer que les paramètres doivent être sauvegardés dans la base de données plutôt que dans un fichier local.
4. Dans la section des **paramètres de base de données** indiquez les accréditations requises pour se connecter à la base de données :



5. Vous pouvez, facultativement, tester la connexion pour vérifier que vous pouvez vous connecter à la base de données.
6. Sauvegardez la nouvelle configuration pour créer le fichier `db.properties` dans le répertoire *<rép_profil_JazzSM>*`/registry` ; par défaut, il s'agit du répertoire `/opt/IBM/JazzSM/profile/registry`.
7. Dans la première console Dynamic Workload Console que vous configurez, cliquez sur **Initialiser la base de données**. Vous pouvez cliquer sur **Initialiser la base de données** pour supprimer et recréer la base de données à tout moment ; dans ce cas, les préférences utilisateur sauvegardées sont perdues.

Remarque : Toutes les instances Dynamic Workload Console qui doivent être synchronisées doivent être commutées sur une configuration DB2. Si vous commutez une console Dynamic Workload Console vous devez également commuter toutes les autres.

Tous les autres paramètres sont sauvegardés dans la base de données, partagés par toutes les instances Dynamic Workload Console et toutes les opérations englobant des paramètres utilisateur sont exécutées sur ce référentiel de paramètres.

Modification de l'utilisateur Dynamic Workload Console du référentiel de base de données

Comment modifier l'utilisateur Dynamic Workload Console qui met à jour le référentiel de paramètres sur DB2.

Pour effectuer cette tâche vous devez disposer du rôle TWSWEBUIAdministrator.

Vous devez faire passer le référentiel de paramètres Dynamic Workload Console d'un fichier local vers un référentiel de base de données, selon la procédure décrite dans la section Modification du référentiel des paramètres.

Seuls les utilisateurs ayant des droits d'accès d'administrateur de base de données sont autorisés à initialiser les tables relatives à Dynamic Workload Console dans la base de données.

Si vous voulez que Dynamic Workload Console accède au référentiel de base de données avec un utilisateur ne disposant pas de privilèges d'administrateur vous devez suivre la procédure ci-dessous :

1. Créez un utilisateur DB2 et octroyez-lui les droits SELECT, INSERT, UPDATE, DELETE sur toutes les tables suivantes, selon le schéma `TDWC` :

   ```
   TDWC_EngineConnection
   TDWC_QueryTask
   TDWC_ReportTask
   TDWC_MEQueryTask
   TDWC_Credential
   TDWC_ConfigurationProperty
   TDWC_Preferenceable
   ```

   Les valeurs ci-dessus correspondent aux droits d'accès par défaut. Cependant, si vous devez restreindre vos règles, vous pouvez octroyer les droits suivants au nouvel utilisateur DB2user :

   ```
   revoke connect,bindadd, createtab, implicit_schema on database from public;
   revoke use of tablespace USERSPACE1     from public;

   grant use of tablespace userspace1 to  user twsdb2;
   grant createtab on database to user  twsdb2;
   grant implicit_schema on database to user twsdb2;
   ```

2. Modifiez l'utilisateur Dynamic Workload Console qui accède à DB2

   a. A partir de la barre d'outils de navigation, cliquez sur  **Configuration système** > **Gérer les paramètres**.

   b. Dans la section **Database Settings** indiquez les accréditations de l'utilisateur nouvellement créé qui doit se connecter à la base de données.

   Remarque : Suite à ce changement d'utilisateur, sans les droits d'administrateur de base de données, les actions suivantes ne seront plus possibles dans le panneau Gérer les paramètres de Dynamic Workload Console :

- **Initialisation de la base de données**
- **Importation de paramètres** avec l'option **Annuler et recréer**.

Modification de l'utilisateur Dashboard Application Services Hub du référentiel de base de données

Comment modifier l'utilisateur Dashboard Application Services Hub qui met à jour le référentiel de paramètres sur DB2.

Vous devez faire passer le référentiel de paramètres Dynamic Workload Console d'un fichier local vers un référentiel de base de données, selon la procédure décrite dans la section Modification du référentiel des paramètres.

Si vous voulez que Dashboard Application Services Hub accède au référentiel de base de données avec un utilisateur ne disposant pas de privilèges d'administrateur vous devez suivre la procédure ci-dessous :

1. Créez un utilisateur DB2 et octroyez-lui les droits CONNECT, CREATETAB, LOAD. Par exemple, `db2 GRANT CONNECT,CREATETAB,LOAD ON DATABASE TO USER db2user2`
2. Sur chaque noeud Dashboard Application Services Hub, configurez le système haute disponibilité en suivant la procédure fournie dans la section «Définition d'une configuration haute disponibilité», à la page 134, et en modifiant les paramètres dans `tipha.properties` suivant les nouvelles informations.

   Par exemple, si le nouveau nom de base de données est `tipdb2` vous devez définir les propriétés suivantes :

   ```
   DBName=tipdb2
   DBDatasource=jdbc/tipds2
   DBDatasourceName=tipds2
   HAEnabled=false
   ```
3. Exécutez le script `ws.ant` en indiquant le nouvel utilisateur DB2.

   Par exemple, `../ws_ant.sh -f install.ant configHA -Dusername=db2user2 -Dpassword=`*`pass`*

Configuration de la haute disponibilité pour plusieurs serveurs Tivoli Workload Scheduler for z/OS

Définition de la configuration haute disponibilité pour des serveurs Tivoli Workload Scheduler for z/OS multiples.

Vous devez avoir configuré la haute disponibilité comme décrit dans la section «Configuration de la haute disponibilité pour Dynamic Workload Console», à la page 130.

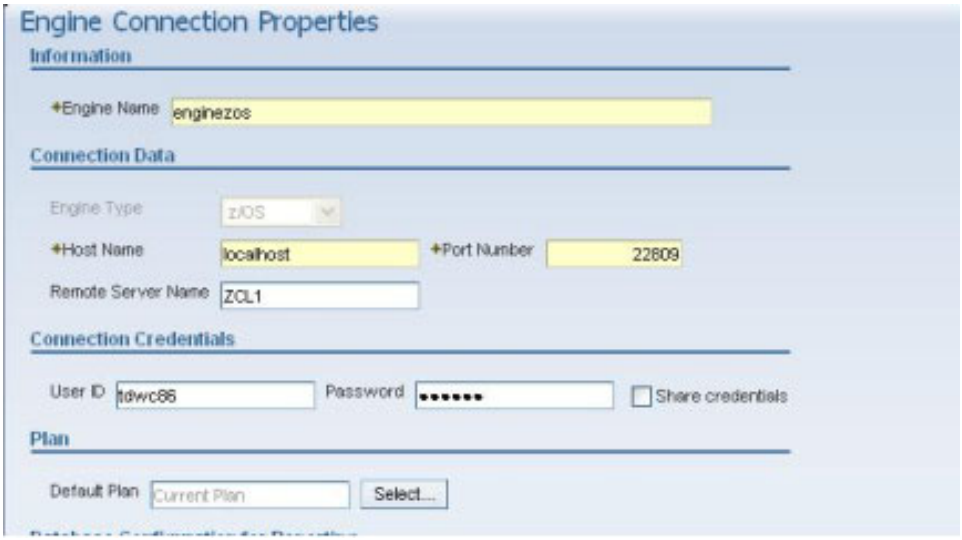
Pour définir la haute disponibilité dans un environnement z/OS et répartir la charge de travail sur plusieurs consoles Dynamic Workload Console et serveurs Tivoli Workload Scheduler for z/OS, suivez la procédure ci-dessous.

1. Installez un connecteur Tivoli Workload Scheduler for z/OS qui se partage WebSphere Application Server avec la Dynamic Workload Console déjà installée.

   Remarque : Pendant l'installation, utilisez les mêmes ports et les mêmes noms d'utilisateur sur chaque noeud.
2. Créez un moteur sur chaque connecteur Tivoli Workload Scheduler for z/OS en utilisant les mêmes noms (par exemple, ZCL1) et noms d'hôte pour tous les

moteurs, mais en indiquant différents numéros de port, afin de définir des connexions pointant sur différents serveurs Tivoli Workload Scheduler for z/OS pour un même contrôleur.

3. Ouvrez le dossier `TWA_home\wastools` et exécutez le script `createZosEngine.sh` (`createZosEnginebat` sous Windows) afin de créer la connexion. Par exemple, `./createZosEngine.sh -name ZCL1 -hostName x.xxx.xxx.xx -portNumber 3446`
4. Si vous voulez vous connecter à plusieurs contrôleurs, répétez cette opération à l'aide d'un nom de moteur différent afin de créer des connexions supplémentaires. Il en résulte qu'un même ensemble de moteurs est défini sur chaque connecteur et tous les moteurs utilisent le même nom et pointent sur le même contrôleur, via un serveur différent.
5. A partir d'une des instances Dynamic Workload Console, créez une connexion de moteur en indiquant le **Nom du serveur distant** défini dans le connecteur Tivoli Workload Scheduler for z/OS (par exemple, ZCL1) et le **Nom d'hôte** en tant que **local host**, pour utiliser le connecteur Tivoli Workload Scheduler for z/OS installé en local avec Dynamic Workload Console. Définissez une connexion de moteur pour chaque nom de moteur défini à l'étape 4.



La configuration haute disponibilité achemine alors les utilisateurs vers un autre serveur Dynamic Workload Console, tandis que chaque console Dynamic Workload Console utilise le connecteur Tivoli Workload Scheduler for z/OS installé en local sur le même serveur WebSphere Application Server et sur un serveur Tivoli Workload Scheduler for z/OS différent.

Gestion du référentiel de paramètres Dynamic Workload Console

Les paramètres utilisateur comme les préférences utilisateur, les tâches enregistrées et les connexions au moteur sont stockés dans le référentiel de paramètres, qui est un fichier local par défaut. Cependant, vous pouvez modifier ce paramètre et utiliser le référentiel des paramètres de base de données pour toutes les opérations de Dynamic Workload Console impliquant des paramètres utilisateur. Cela peut être utile par exemple pour des motifs d'évolutivité ou pour disposer de plusieurs instances Dynamic Workload Console ayant les mêmes paramètres utilisateur.

Pour utiliser une base de données en tant que référentiel de paramètres, vous devez configurer les paramètres de la base de données, comme décrit dans les sections relatives au changement et au partage du référentiel de paramètres dans

Tivoli Workload Scheduler : Guide d'utilisation de Dynamic Workload Console, disponible dans le Centre de documentation du produit : http://publib.boulder.ibm.com/infocenter/tivihelp/v47r1/index.jsp?topic=/com.ibm.tivoli.itws.doc_6/welcome.html/welcome_TWA.html.

Configuration de Dynamic Workload Console pour l'affichage des rapports

Cette rubrique décrit les étapes de configuration à suivre pour afficher les rapports à partir de la console Dynamic Workload Console.

Pour accéder à la base de données contenant les rapports, vous devez remplir les conditions préalables suivantes :

- détenir un ID utilisateur et un mot de passe d'accès à la base de données
- disposer d'une connexion active entre Dynamic Workload Console et la base de données

Suivez la procédure ci-après sur le système sur lequel fonctionne le moteur Tivoli Workload Scheduler :

- «Configuration pour une base de données DB2»
- «Configuration pour une base de données Oracle», à la page 151

Configuration pour une base de données DB2

Lorsque DB2 sur le gestionnaire de domaine maître utilise le pilote DB2 JDBC de type 2, et Dynamic Workload Console à partir duquel vous voulez gérer les rapports est sur un poste de travail différent du gestionnaire de domaine maître, vous devez effectuer quelques étapes de configuration sur le poste de travail de Dynamic Workload Console pour réussir à activer une connexion au moteur Tivoli Workload Scheduler.

1. Installez le client DB2 sur le poste de travail de Dynamic Workload Console.
2. Pour connecter le client DB2 au serveur, exécutez les commandes suivantes dans l'ordre :

   ```
   db2 catalog tcpip node TWS_ND remote nc125139.romelab.it.ibm.com server 50000
   db2 attach to TWS_ND user db2admin using "db2admin"
   db2 catalog db TWS at node TWS_ND
   ```

   où,

   <NOM_NOEUD_TWS>
   Nom du noeud, par exemple TWS_ND.

   <HOTE_TWS>
   Nom d'hôte du poste de travail du serveur DB2.

   <PORT_SRVC_TWS>
   Numéro de port du poste de travail du serveur DB2.

   <UTILISATEUR_ADMIN_TWS>
   Nom de l'utilisateur du serveur DB2.

   <MDP_ADMIN_TWS>
   Mot de passe de l'utilisateur du serveur DB2 .

   <BD_TWS>
   Nom de la base de données Tivoli Workload Scheduler.

   Un exemple peut être :

```
db2 catalog tcpip node TWS_ND remote nc125139.romelab.it.ibm.com server 50000
db2 attach to TWS_ND user db2admin using "db2admin"
db2 catalog db TWS at node TWS_ND
```

3. Arrêtez le serveur Dynamic Workload Console WebSphere Application Server.
4. Modifiez le script setupCmdLine.sh situé dans le chemin <Profil_JAZZSM>/bin/ en ajoutant les lignes suivantes à la fin du script :

```
 LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<REP_BASE_CLIENT_DB2>/lib64; export LD_LIBRARY_PATH
LIBPATH=$LIBPATH:<REP_BASE_CLIENT_DB2>/lib64; export LIBPATH SHLIB_PATH=$SHLIB_PATH:
<REP_BASE_CLIENT_DB2>/lib64; export SHLIB_PATH
```

5. Redémarrez Dynamic Workload Console.

Pour DB2, l'administrateur informatique et/ou celui de Tivoli Workload Scheduler procèdent comme suit :

1. Créer un utilisateur de système d'exploitation et spécifiez un mot de passe.
2. Lancez le script suivant :

```
<rép_base_TWA>/TWS/dbtools/DB2/scripts/dbgrant.bat/.sh
  <ID_utilisateur_à_autoriser>
  <nom_base_de_données>
  [<admin_base_de_données> <mot_de_passe>]
```

   où les variables se présentent comme suit :

   <rép_base_TWA>
   Répertoire de l'instance Tivoli Workload Automation

   <ID_utilisateur_à_autoriser>
   ID de l'utilisateur créé à l'étape 1 et qui va être autorisé à accéder aux rapports

   <nom_base_de_données>
   Nom de la base de données créée lors de l'installation du gestionnaire de domaine maître

   [*<admin_base_de_données>* *<mot_de_passe>*]
   ID utilisateur et mot de passe de l'utilisateur administrateur de base de données. Si vous exécutez cette commande en tant qu'utilisateur administrateur de base de données, vous pouvez ignorer ces paramètres.

3. Connectez-vous à Dynamic Workload Console.
4. Dans le menu Portefeuille, sélectionnez **Gérer les moteurs**. Le panneau de gestion des moteurs s'affiche.
5. Sélectionnez le moteur que vous avez défini ou créez un autre moteur. Le panneau de propriétés de connexion au moteur s'affiche.
6. Dans Configuration de la base de données pour la génération de rapports, procédez comme suit :
   a. Cochez la case **Activer la génération de rapports** pour activer la connexion au moteur sélectionné pour exécuter les rapports.
   b. Dans **ID utilisateur et mot de passe de la base de données**, indiquez l'utilisateur et le mot de passe de base de données permettant d'accéder aux rapports.

Configuration pour une base de données Oracle

Actions effectuées sur le moteur Tivoli Workload Scheduler :

Pour Oracle, l'administrateur informatique et/ou celui de Tivoli Workload Scheduler procèdent comme suit :

1. Utilisez l'**utilisateur TWS Oracle** spécifié pendant l'installation de gestionnaire de domaine maître ou procédez comme suit pour créer un nouvel utilisateur :
   a. Créez un utilisateur de base de données autorisé à accéder à la base de données et spécifiez un mot de passe.
   b. Lancez le script suivant :

      ```
      <TWA_home>/TWS/dbtools/Oracle/scripts/dbgrant.bat/.sh
        <ID_of_user_to_be_granted>
        <database_name>
        <utilisateur_admin_base_données> <password>
      ```

      où les variables se présentent comme suit :

      <TWA_home>
      Répertoire de l'instance Tivoli Workload Automation

      <ID_of_user_to_be_granted>
      ID de l'utilisateur créé à l'étape 1a et qui va être autorisé à accéder aux rapports

      <database_name>
      Nom de la base de données créée lors de l'installation du gestionnaire de domaine maître

      <propriétaire_schéma_base_données> <password>
      ID utilisateur et mot de passe du propriétaire du schéma de base de données.
2. Définition d'une chaîne de connexion valide pour la base de données :
   a. Vérifiez que la propriété suivante est définie dans le fichier `<TWA_home>/WAS/TWSprofile/properties/TWSConfig.properties` pour pointer vers l'URL Oracle JDBC :

      ```
      com.ibm.tws.webui.oracleJdbcURL
      ```

      Par exemple :

      ```
      com.ibm.tws.webui.oracleJdbcURL=jdbc:oracle:thin:@//9.132.235.7:1521/orcl
      ```
   b. Redémarrez le serveur WebSphere Application Server.

Actions effectuées sur la console Dynamic Workload Console :

1. Téléchargez les pilotes JDBC demandés par votre version du serveur Oracle.
2. Copiez les pilotes JDBC dans un répertoire accessible par le serveur WebSphere Application Server utilisé par votre console Dynamic Workload Console.
3. Créez une bibliothèque partagée sur le serveur WebSphere Application Server, en précisant le chemin et le nom de fichier des pilotes JDBC que vous avez copiés, comme décrit dans :

   WebSphere Application Server (systèmes d'exploitation distribués), documentation version 8.0, section sur la Configuration de l'environnement de traitement d'applications> l'Administration des serveurs d'applications> la Gestion des bibliothèques partagées.
4. Associez l'application d'entreprise **isc** à cette bibliothèque partagée, comme décrit dans :

   WebSphere Application Server (systèmes d'exploitation distribués), documentation version 8.0, section sur la Configuration de l'environnement de traitement d'applications> l'Administration des serveurs d'applications> la Gestion des bibliothèques partagées.
5. Connectez-vous à Dynamic Workload Console.

6. Dans la barre de navigation de Dashboard Application Services Hub, sélectionnez **Configuration du système** > **Gestion des moteurs**. Le panneau de gestion des moteurs s'ouvre.
7. Sélectionnez le moteur que vous avez défini ou créez un autre moteur. Le panneau de propriétés de connexion au moteur s'affiche.
8. Dans Configuration de la base de données pour la génération de rapports, procédez comme suit :
   a. Cochez la case **Activer la génération de rapports** pour activer la connexion au moteur sélectionné pour exécuter les rapports.
   b. Dans **ID utilisateur et mot de passe de la base de données**, indiquez l'utilisateur et le mot de passe de base de données permettant d'accéder aux rapports.

Empêchez une connexion à des moteurs Tivoli Workload Scheduler Version 8.3 spécifiques

Exécutez le script suivant du côté Tivoli Workload Scheduler si vous souhaitez désactiver l'établissement des connexions de moteurs à partir de Dynamic Workload Console vers un moteur Tivoli Workload Scheduler Version 8.3

Sous Windows :

```
webui -operation disable
```

Exécutez le script en tant qu'administrateur Tivoli Workload Scheduler, à partir du répertoire *rép_principal_TWS*\wastools:

Sur UNIX

```
./webui.sh -operation disable
```

Exécutez le script en tant que superutilisateur, à partir du répertoire *rép_principal_TWS*/wastools :

Redémarrez le WebSphere Application Server sur le moteur Tivoli Workload Scheduler sur lequel vous exécutez le script.

Chapitre 4. Configuration de l'autorisation des utilisateurs (fichier de sécurité)

Le présent chapitre présente la gestion des autorisation d'accès aux objets de planification octroyées aux utilisateurs de Tivoli Workload Scheduler.

Le présent chapitre comprend les sections suivantes :

Présentation de la gestion de la sécurité

La gestion de la sécurité par Tivoli Workload Scheduler est contrôlée par le fichier de configuration nommé ***fichier de sécurité***. Ce fichier contrôle des activités telles que :

- La liaison des postes de travail
- L'accès aux programmes de ligne de commande et à Dynamic Workload Console.
- L'exécution d'opérations sur les objets de planification dans la base de données ou dans le plan.

Dans le fichier, vous spécifiez pour chaque utilisateur les objets de planification auxquels il peut accéder et les actions qu'il est autorisé à effectuer sur ces objets. Vous pouvez déterminer un accès par type d'objet (par exemple, postes de travail ou ressources) et, au sein d'un type d'objet, par attributs sélectionnés, tels que le nom de l'objet ou du poste de travail dans la définition de l'objet. Vous pouvez utiliser des caractères génériques pour sélectionner des ensembles d'objets apparentés. Les droits d'accès peuvent être octroyés sur une base "inclusive" ou "exclusive", ou une combinaison des deux.

Chaque fois que vous modifiez les autorisations d'accès, vous modifiez le fichier de configuration et le convertissez à un format chiffré (pour améliorer les performances et la sécurité) en remplaçant le fichier précédent. Le système utilise ce *fichier de sécurité* chiffré à partir de cet instant.

Chaque fois qu'un utilisateur exécute des programmes, commandes et interfaces utilisateurTivoli Workload Scheduler, le produit compare le nom de l'utilisateur aux définitions d'utilisateur dans le *fichier de sécurité* pour déterminer si l'utilisateur dispose des droits de modification de ces activités sur les objets de planification spécifiés.

Par défaut, la sécurité des objets de planification est gérée localement sur chaque poste de travail. Cela signifie que l'administrateur système ou l'*utilisateur_TWS* qui a installé le produit sur ce système peut décider quels utilisateurs Tivoli Workload Scheduler définis sur le système peuvent accéder à quelles ressources de planification dans le réseau Tivoli Workload Scheduler et de quelle façon.

Vous pouvez aussi centraliser le contrôle de la gestion des objets sur chaque poste de travail. Cette option peut être configurée en définissant une option globale. Dans ce scénario, vous configurez toutes les autorisations d'utilisateur dans le *fichier de sécurité* sur le gestionnaire de domaine maître. La version chiffrée du fichier est automatiquement distribuée chaque fois que vous exécutez **JnextPlan**, de sorte que tous les postes de travail disposent du fichier au niveau central pour déterminer les autorisations des utilisateurs sur ce poste de travail.

Mise en route

Cette section décrit la mise en route avec la définition des autorisations après l'installation.

Un fichier modèle nommé *TWA_home*/TWS/config/Security.conf est fourni avec le produit. Pendant l'installation, une copie du fichier modèle est installée en tant que *TWA_home*/TWS/Security.conf et une copie opérationnelle compilée est installée en tant que *TWA_home*/TWS/Security.

Cette version du fichier contient une définition complète des accès pour l'utilisateur qui a installé le produit, *utilisateur_TWS*, et l'administrateur système (root sous UNIX ou l'administrateur sous Windows), qui sont les seuls utilisateurs définis et autorisés à se connecter aux interfaces utilisateur et à effectuer toutes les opérations sur toutes les ressources de planification.

Dans le réseau Tivoli Workload Scheduler, à l'aide du fichier de sécurité, vous pouvez faire une distinction entre les utilisateurs **root** locaux et l'utilisateur **root** sur le gestionnaire de domaine maître en autorisant les utilisateurs **root** locaux à effectuer des opérations concernant uniquement leurs postes de travail de connexion et octroyant à l'utilisateur gestionnaire de domaine maître **root** l'autorisation de procéder à des opérations concernant tout poste de travail dans l'ensemble du réseau.

A mesure que vous continuez à utiliser le produit, vous pouvez décider d'ajouter d'autres utilisateurs ayant des rôles différents et de leur accorder l'autorisation d'effectuer des opérations spécifiques sur un ensemble d'objets défini.

Ne modifiez pas le modèle *TWA_home*/TWS/config/Security.conf d'origine, mais suivez les étapes décrites dans «Mise à jour du fichier de sécurité» pour effectuer vos modifications sur la copie opérationnelle du fichier.

Mise à jour du fichier de sécurité

Par défaut, chaque poste de travail d'un réseau Tivoli Workload Scheduler (gestionnaires de domaine, agent tolérant aux pannes et agents standard) possède son propre fichier de sécurité. Vous pouvez entretenir ce fichier sur chaque poste de travail ou, si vous activez la gestion de sécurité centralisée, vous pouvez créer un fichier de sécurité sur le gestionnaire de domaine maître et le copier sur chaque gestionnaire de domaine et sur chaque agent, s'assurant ainsi que tous les utilisateurs de Tivoli Workload Scheduler reçoivent l'autorisation requise dans le fichier (voir «Gestion centralisée de la sécurité», à la page 159). Que vous travailliez sur le poste de travail d'un agent pour un fichier de sécurité individuel ou sur le gestionnaire de domaine maître pour modifier un fichier centralisé, les étapes sont les mêmes ; la seule différence concerne le nombre d'utilisateurs que vous définissez (ceux du système local seulement ou tous les utilisateurs du réseau Tivoli Workload Scheduler).

Ni les processus Tivoli Workload Scheduler, ni l'infrastructure WebSphere Application Server ne doivent être arrêtés ou redémarrés pour mettre à jour le fichier de sécurité. Il suffit de fermer toute interface utilisateur **conman** ouverte avant d'exécuter **makesec**.

Pour modifier le fichier de sécurité, procédez comme suit :

1. Naviguez jusqu'au répertoire *TWA_home*/TWS à partir duquel les commandes **dumpsec** et **makesec** *doivent* être exécutées.
2. Exécutez la commande **dumpsec** pour déchiffrer le fichier de sécurité actuel dans un fichier de configuration éditable. Voir «dumpsec».
3. Modifiez le contenu du fichier de configuration éditable en utilisant la syntaxe décrite dans «Configuration du fichier de sécurité», à la page 161.
4. Fermez toutes les interfaces utilisateur **conman** ouvertes à l'aide de la commande **exit**.
5. Arrêtez tous les connecteurs sur les systèmes exécutant des systèmes d'exploitationWindows.
6. Exécutez la commande **makesec** pour chiffrer le fichier de sécurité et appliquer les modifications. Voir «makesec», à la page 158.
7. Si vous utilisez la sécurité locale, le fichier est immédiatement disponible sur le poste de travail sur lequel il a été mis à jour.

   Si vous utilisez une sécurité centralisée (voir «Gestion centralisée de la sécurité» , à la page 159), vous devez à présent procéder comme suit :

   a. Si vous utilisez un gestionnaire de domaine maître de sauvegarde, copiez le fichier dessus
   b. Distribuez manuellement le fichier centralisé vers tous les agents tolérants aux pannes du réseau (pas les agents standard, étendus ou de courtier) et stockez-le dans le répertoire *TWA_home*/TWS
   c. Exécutez **JnextPlan** pour distribuer le fichier Symphony correspondant au nouveau fichier de sécurité.

   Voir les pages suivantes pour obtenir une description exhaustive de **dumpsec** et de **makesec**.

dumpsec

Ecrit dans un format éditable les informations figurant dans le fichier de sécurité compilé et chiffré. Le fichier de sortie peut être modifié, puis utilisé comme entrée pour la commande **makesec** qui compile et active les paramètres de sécurité modifiés.

Autorisation

Vous devez détenir les droits d'accès ***display*** au fichier de sécurité et les droit d'accès en écriture dans le Répertoire *TWA_home*/TWS depuis lequel la commande *doit* être exécutée.

Syntaxe

dumpsec –v | –u

dumpsec *fichier_sécurité* [**>** *fichier_sortie*]

Commentaires

Si vous ne spécifiez aucun argument, le fichier de sécurité opérationnel est envoyé à stdout. Pour créer une copie éditable du fichier de sécurité, réorientez la sortie de la commande vers un fichier de sortie, en utilisant le symbole de redirection.

Arguments

–v Affiche uniquement les informations sur la version de la commande.

–u Affiche uniquement les informations sur l'utilisation de la commande.

fichier_sécurité
Indique le nom du fichier de sécurité à vider.

[> *fichier_sortie***]**
Spécifie le nom du fichier de sortie. Si cette option est omise, le fichier de sécurité est envoyé vers le stdout.

Exemples

La commande suivante vide le fichier de sécurité opérationnel (*TWA_home*/TWS/Security) dans un fichier nommé **mysec** :

```
dumpsec > mysec
```

La commande suivante permet de vider le fichier de sécurité **sectemp** dans le fichier **stdout** :

```
dumpsec sectemp
```

makesec

Compile les définitions de sécurité et installe le fichier de sécurité. Les changements apportés au fichier de sécurité sont reconnus dès que **makesec** est terminé ou, dans le cas d'une sécurité centralisée, après que **JnextPlan** l'ait distribué.

Remarque : Avant d'exécuter la commande **makesec**, arrêtez **conman** et, sur les systèmes exécutant les systèmes d'exploitation Windows, arrêtez les éventuels connecteurs.

Autorisation

Vous devez détenir les droits d'accès ***modify*** au fichier de sécurité et les droits d'accès en lecture dans le Répertoire *TWA_home*/TWS depuis lequel la commande *doit* être exécutée.

Syntaxe

makesec –v | –u

makesec [**–verify**] *in_file*

Commentaires

La commande **makesec** compile le fichier spécifié et l'installe en tant que fichier de sécurité opérationnel (../*TWA_home*/TWS/Security). Si vous ajoutez l'argument **–verify**, le programme vérifie la syntaxe du fichier sans le compiler ou l'installer.

Arguments

–v Affiche uniquement les informations sur la version de la commande.

–u Affiche uniquement les informations sur l'utilisation de la commande.

–verify
Vérifie la syntaxe des définitions d'utilisateur dans le fichier *in_file*. Par ailleurs, le programme ne compile pas et n'installe pas le fichier au titre de fichier de sécurité.

in_file Précisez le nom d'un ou de plusieurs fichiers renfermant les définitions d'utilisateur. Le programme vérifie automatiquement la syntaxe du fichier de sécurité lors de son installation.

Exemples

Exemple 1 : Modification des définitions du fichier de sécurité - scénario complet

L'exemple suivant montre comment modifier les définitions du fichier de sécurité :

1. Une copie éditable du fichier de sécurité opérationnel est créée dans un fichier nommé `tempsec` à l'aide de la commande **dumpsec** :
   ```
   dumpsec > tempsec
   ```
2. Les définitions d'utilisateur sont modifiées à l'aide d'un éditeur de texte :
   ```
   edit tempsec
   ```
3. Le fichier est alors compilé et installé avec la commande **makesec** :
   ```
   makesec tempsec
   ```

Exemple 2 : Compilation des définitions d'utilisateur à partir de plusieurs fichiers

La commande suivante compile les définitions utilisateur du jeu de fichiers `userdef*` et remplace le fichier de sécurité opérationnel :

```
makesec userdef*
```

Gestion centralisée de la sécurité

Un environnement Tivoli Workload Scheduler où la gestion centralisée de la sécurité est activée est un environnement où tous les postes de travail partagent les mêmes informations contenues dans le fichier de sécurité stocké sur le gestionnaire de domaine maître, et l'administrateur Tivoli Workload Scheduler sur le gestionnaire de domaine maître est la seule personne qui puisse ajouter, modifier et supprimer des entrées du fichier de sécurité valables pour l'ensemble de l'environnement Tivoli Workload Scheduler.

Cela peut être configuré à l'aide de l'option globale ***enCentSec***. Par défaut, la valeur attribuée à l'option ***enCentSec*** est **no**.

Pour définir la gestion de la sécurité centrale, l'administrateur Tivoli Workload Scheduler doit exécuter les étapes suivantes sur le gestionnaire de domaine maître :

1. Utilisez le programme de ligne de commande **optman**, pour définir la valeur attribuée à la propriété globale *enCentSec* sur **yes**. Pour plus d'informations sur la modification des propriétés globales à l'aide de **optman**, voir «Définition des options globales», à la page 7.

2. Enregistrez les informations du fichier de sécurité dans un fichier de configuration éditable à l'aide de la commande **dumpsec**.
3. Définissez les autorisations requises pour tous les utilisateurs de Tivoli Workload Scheduler comme décrit dans «Configuration du fichier de sécurité», à la page 161
4. Fermez toutes les interfaces utilisateur **conman** ouvertes à l'aide de la commande **exit**.
5. Arrêtez tous les connecteurs sur les systèmes exécutant des systèmes d'exploitationWindows.
6. Compilez le fichier de sécurité à l'aide de la commande **makesec**.
7. Si vous utilisez un gestionnaire de domaine maître de sauvegarde, copiez le fichier de sécurité compilé dessus dès que possible.
8. Distribuez le fichier de sécurité compilé à tous les postes de travail de l'environnement et stockez-le dans leurs répertoires *TWA_home*/TWS.
9. Exécutez **JnextPlan** pour mettre à jour les informations de sécurité distribuées à l'aide du fichier Symphony.

   La valeur du total de contrôle (checksum) du fichier de sécurité récemment compilé est chiffrée et chargée dans le fichier Symphony, puis distribuée à tous les postes de travail du réseau Tivoli Workload Scheduler.

   Sur chaque poste de travail, lorsqu'un lien est établi ou lorsqu'un utilisateur se connecte à une interface utilisateur ou tente d'émettre des commandes sur le plan, à l'aide de **conman** ou de Dynamic Workload Console, Tivoli Workload Scheduler compare la valeur checksum figurant dans le fichier de sécurité fourni avec le fichier Symphony avec la valeur checksum dans le fichier de sécurité stockée sur le poste de travail. Si les valeurs sont égales, l'opération est autorisée. Si les valeurs différentes, l'opération échoue et un message de violation de sécurité est émis.

Notes d'utilisation de sécurité centralisées

Dans un réseau bénéficiant d'une gestion centralisée de la sécurité, le programme ne peut pas établir la connexion entre deux postes de travail si l'option ***enCentSec*** est désactivée dans le fichier Symphony de l'un d'entre eux ou si les informations du fichier de sécurité diffèrent.

La seule exception au critère de correspondance du fichier de sécurité introduit par le mécanisme de gestion de sécurité centralisé est qu'un poste de travail doit toujours accepter les connexions entrantes de son gestionnaire de domaine quel que soit le résultat du processus de correspondance du fichier de sécurité.

La sécurité centralisée ne régit pas les transactions Tivoli Workload Scheduler pour lesquelles le fichier Symphony est inutile. Les commandes ne demandant pas l'exécution du fichier Symphony utilisent le fichier de sécurité locale. Par exemple la commande **parms** utilisée pour modifier ou afficher la base de données des paramètres locaux continue à fonctionner selon le fichier de sécurité locale, même si la sécurité centralisée est active et même si le fichier de sécurité locale diffère des règles de sécurité centralisées.

Si le fichier de sécurité d'un poste de travail est supprimé et recréé, le total de contrôle du nouveau fichier de sécurité ne concordera pas avec la valeur du fichier Symphony. En outre, le programme protège le fichier contre toute contrefaçon, grâce à un mécanisme de numéro d'exécution associé au processus de création du fichier Symphony.

Configuration du fichier de sécurité

Dans le fichier de sécurité, vous pouvez spécifier quels sont les objets de planification qu'un utilisateur peut gérer et comment. Vous définissez ces paramètres en écrivant des définitions utilisateur. Une définition d'utilisateur est une association entre un nom et un ensemble d'utilisateurs, les objets auxquels ils peuvent accéder et les opérations qu'ils peuvent réaliser sur les objets spécifiés.

Lors de la définition d'une autorisation utilisateur, tenez compte des points suivants :

- Lorsque les commandes sont émises à partir du programme de ligne de commande **composer**, les autorisations utilisateurs sont vérifiées dans le fichier de sécurité du gestionnaire de domaine maître car les méthodes utilisées pour gérer les entrées dans la base de données sont appelées sur le gestionnaire de domaine maître Par conséquent, l'utilisateur doit être défini :
  - En tant qu'utilisateur système sur le système où le gestionnaire de domaine maître est installé.
  - Dans le fichier de sécurité sur le gestionnaire de domaine maître avec les autorisations requises pour exécuter les commandes autorisées sur les objets spécifiques.
- Lorsque des commandes sont émises à partir du programme de ligne de commande **conman**, l'utilisateur doit être autorisé à exécuter les commandes spécifiques dans le fichier de sécurité, à la fois sur le poste de travail connecté et sur le gestionnaire de domaine maître où la commande s'exécute réellement.

La configuration est décrite dans les sections ci-dessous :

- «Syntaxe du fichier de sécurité»
- «Spécification des attributs d'utilisateur», à la page 163
- «Spécification de types d'objet», à la page 169
- «Spécification des attributs d'objet», à la page 170
- «Définition de l'accès», à la page 175
- «*utilisateur_TWS* - Remarques particulières relatives au fichier de sécurité», à la page 190

Syntaxe du fichier de sécurité

La syntaxe du fichier de sécurité est la suivante :

Fichier de sécurité
Syntaxe

[**#** *commentaires*]

utilisateur *nom_définition attributs_utilisateur*

débuter [***** *commentaires*]

type_objet [*attributs_objet*]. **accès**[**=***mot clé*[**,***mot clé*]...]

[*type_objet* [*attributs_objet*]. **accès**[**=***mot clé*[**,***mot clé*]...]]...

fin | **continuer**

Arguments

[# | *] *commentaire*
Toutes les chaînes de texte comprenant le symbole dièse/astérisque et au moins un espace sont traitées comme des commentaires. Ils ne sont pas copiés dans le fichier de sécurité opérationnel installé par la commande **makesec**.

utilisateur *nom_définition*
Précise le nom de la définition d'utilisateur. Il peut compter 36 caractères alphanumériques maximum et doit commencer par un caractère alphabétique.

attributs_utilisateur
Contient un ou plusieurs attributs identifiant l'utilisateur ou les utilisateurs à qui la définition s'applique. Pour plus d'informations sur la définition des attributs d'utilisateur, voir «Spécification des attributs d'utilisateur», à la page 163.

begin Débute la partie contenant les instructions et les accès aux objets dans la définition de l'utilisateur.

type_objet
Identifie le type d'objet (par exemple : poste de travail, ressource ou invite) auquel l'accès doit être attribué pour l'utilisateur ou les utilisateurs spécifié(s). Tous les types d'objet dont l'utilisateur ou les utilisateurs spécifié(s) ont besoin d'accéder doivent être définis explicitement. Si tel n'est pas le cas, aucun accès ne sera accordé. Pour plus de détails sur la définition des types d'objet, voir «Spécification de types d'objet», à la page 169.

attributs_objet
Contient un ou plusieurs attributs identifiant les objets spécifiques du type d'objet défini auquel le même accès doit être accordé. Si aucun attribut d'objet n'est défini, l'accès est accordé à tous les objets du type d'objet défini. Pour plus de détails sur la définition des attributs d'objet, voir «Spécification des attributs d'objet», à la page 170.

access[=*mot clé***[,***mot clé***]...]**
Décrit les accès aux objets spécifiés accordés aux utilisateurs sélectionnés. Si aucun n'est spécifié (en indiquant simplement le mot clé "access") aucun accès n'est accordé aux objets associés. Si **access=@**, tous les droits d'accès sont accordés aux utilisateurs spécifiés. Pour plus de détails sur la définition des accès, voir «Définition de l'accès», à la page 175.

continuer
Termine la définition de l'utilisateur. Un utilisateur reçoit tous les accès définis pour chaque groupe auquel il appartient, jusqu'à ce que la définition d'un utilisateur avec une instruction `end` soit atteinte. Pour obtenir un exemple d'utilisation du mot clé `continue`, voir «Utilisateurs connectés à plusieurs groupes [mot clé continue]», à la page 195

end Termine la définition de l'utilisateur. Les utilisateurs définis dans la définition d'utilisateur qui se terminent par une instruction `end` ne correspondent à aucune définition d'utilisateur ultérieure.

Caractères génériques

Les caractères génériques acceptés dans la syntaxe d'une définition d'utilisateur sont les suivants :

? Remplace un caractère alphanumérique.

@ Remplace zéro ou plusieurs caractères alphanumériques.

Pour des informations sur les variables fournies avec le produit et utilisables dans les attributs d'objet, voir «Utilisation de variables dans les définitions d'attribut des objets», à la page 174. Voir «Exemple de fichier de sécurité», à la page 191 pour un exemple de la façon d'utiliser les variables.

Spécification des attributs d'utilisateur

Les attributs d'utilisateur définissent *qui* dispose de l'accès qui va être défini par la suite. Ils peuvent identifier un utilisateur, une sélection d'utilisateurs, un groupe d'utilisateurs, une sélection de groupes d'utilisateurs ou tous les utilisateurs. Vous pouvez aussi exclure un ou plusieurs utilisateurs ou groupes spécifiques d'une sélection. Les utilisateurs sont non seulement identifiés par un identifiant de connexion et un nom de groupe, mais peuvent aussi être décrits par le poste de travail à partir duquel ils se connectent. Enfin, vous pouvez associer les critères de sélection, par exemple en sélectionnant tous les utilisateurs dans un groupe nommé qui peuvent accéder à partir d'un ensemble de postes de travail identifié par un caractère générique, mais comprenant un ensemble d'utilisateurs spécifiques identifiés par leur identifiant de connexion.

Syntaxe générale

Vous effectuez cette sélection en spécifiant un ou plusieurs attributs d'utilisateur. Chaque attribut d'utilisateur est spécifié comme suit :

type_attribut_utilisateur=*value*

type_attribut_utilisateur
Peut être *UC* (poste de travail), *group*, ou *logon*

valeur Identifie une *UC* (poste de travail), un *group* ou un *logon* spécifique, ou, en utilisant des caractères génériques, peut identifier un ensemble de ces éléments.

Inclusion ou exclusion

Chaque attribut peut être *inclus* ou *exclu* à partir de la sélection.

Par conséquent, pour chaque *type_attribut*, vos options sont les suivantes :

Include all
Il s'agit de l'option par défaut. Par exemple, pour inclure tous les *groupes*, vous ne devez ajouter aucun attribut d'utilisateur concernant aucun groupe.

Include a selection
Cette option peut être définie comme suit :

- En incluant spécifiquement les utilisateurs que vous voulez sélectionner (individus ou un ou plusieurs ensembles)
- En excluant spécifiquement (à l'aide de la valeur par défaut *inclure tout*) tous les utilisateurs que vous *ne voulez pas* sélectionner
- En incluant spécifiquement un ensemble d'utilisateurs, puis en excluant certains des utilisateurs figurant dans cet ensemble

L'option que vous choisirez est celle qui est la plus simple à spécifier.

Utilisation des symboles d'inclusion ou d'exclusion :

Include

Précédez l'expression de l'attribut d'utilisateur par un signe plus (+). Tous les utilisateurs identifiés par l'expression seront sélectionnés, sauf s'ils sont aussi sélectionnés par une expression *exclude*. Si le premier attribut de votre définition est un *include*, il n'a pas besoin d'être précédé d'un signe (+), car ce signe est implicite.

La valeur par défaut (si vous ne spécifiez aucun attribut d'utilisateur) est l'inclusion de tous les utilisateurs, sur tous les postes de travail, dans tous les troupes ; ainsi, si vous voulez définir, par exemple, tous les utilisateurs à l'exception d'un seul utilisateur nommé, vous fournirez seulement la définition *exclude* pour cet utilisateur.

Exclude

Précédez l'expression de l'attribut d'utilisateur par un tilde (~). Tous les utilisateurs identifiés par l'expression ne sont *jamais* sélectionnés, qu'ils soient ou non identifiés par des expressions *include*.

Expressions de sélection

Vous pouvez utiliser les différents types suivants d'expression de sélection :

Expressions de sélection de base

Inclure seulement un attribut

type_attribut_utilisateur=*value*

Par exemple, pour inclure un identifiant de connexion d'utilisateur nommé et exclure tous les autres utilisateurs :

```
logon=jsmith1
```

Exclure un attribut

~*type_attribut_utilisateur*=*value*

Par exemple, pour exclure un ensemble d'identifiants de connexion identifiés par un caractère générique (ceux qui commencent par la lettre "j"), mais inclure tous les autres :

```
~logon=j@
```

Inclure seulement plusieurs attributs du même type

type_attribut_utilisateur=*value*[,*value*]...

Par exemple, pour inclure trois utilisateurs spécifiques et exclure tous les autres :

```
logon=jsmith1,jbrown1,jjones1
```

Exclure plusieurs attributs du même type

~*type_attribut_utilisateur*=*value*[,*value*]...

Par exemple, pour exclure trois utilisateurs spécifiques et inclure tous les autres :

```
~logon=jsmith1,jbrown1,jjones1
```

Expressions de sélection complexes

Inclure les utilisateurs identifiés par des expressions de sélection différentes

expression_sélection_base[+*expression_sélection_base*]...

Les expressions de sélection peut être du même type d'attribut ou de types d'attribut différents :

Même type d'attribut

Un exemple du même type d'attribut est le suivant, qui

sélectionne tous les groupes commençant par la lettre "j", ainsi que ceux commençant par la lettre "z" :

```
group=j@+group=z@
```

Si la première sélection identifie 200 utilisateurs et la deuxième 300, le nombre total d'utilisateurs sélectionnés est de 500.

Différents types d'attribut

Un exemple d'expressions de sélection de types d'attribut différents est le suivant, qui sélectionne tous les groupes commençant par la lettre "j", ainsi que les identifiants commençant par le chiffre "6" :

```
group=j@+logon=6@
```

Si la première sélection identifie 200 utilisateurs et la deuxième 20, dont 5 figurent également dans le premier groupe, le nombre total d'utilisateurs sélectionnés est 5.

Exclure les utilisateurs identifiés dans une expression de sélection de deux identifiés dans une autre

expression_sélection_base[**~***expression_sélection_base*]...

Même type d'attribut

Les expressions de sélection peuvent être du même type d'attribut, à condition que la deuxième soit un sous-ensemble de la première. Un exemple du même type d'attribut est le suivant, qui sélectionne tous les postes de travail commençant par la lettre "j", ainsi que ceux commençant par la lettre "z" :

```
group=j@~group=jz@
```

Si la première sélection identifie 200 utilisateurs et la deuxième 20, le nombre total d'utilisateurs sélectionnés est de 180. Notez que si la deuxième expression n'était pas un sous-ensemble de la première, la deuxième expression aurait été ignorée.

Différents types d'attribut

Les expressions de sélection de type d'attribut différent n'ont pas à avoir une relation de sous-ensemble ; en voici un exemple, qui sélectionne le groupe "mygroup", mais exclut de la sélection tous les utilisateurs du groupe dont l'identifiant commence par le chiffre "6" :

```
group=mygroup~logon=6@
```

Si la première sélection identifie 200 utilisateurs et la deuxième 20, dont 5 figurent également dans le premier groupe, le nombre total d'utilisateurs sélectionnés est de 195.

Inclusions et exclusions multiples

Vous pouvez lier ensemble autant d'expressions d'inclusion et d'exclusion que nécessaire pour identifier le sous-ensemble précis d'utilisateurs qui ont besoin du même accès. La syntaxe générale est la suivante :

[~]*type_attribut_utilisateur*=*value*[,*value*]...
[{+|~}*type_attribut_utilisateur*=*value*[,*value*]...

Remarque : Si votre *premier* attribut d'utilisateur est *exclude*, *sous* les attributs d'utilisateur de ce type sont sélectionnés *excepté* la *value* indiquée. Ainsi, ~*type_attribut_utilisateur=value* est équivalent à :

type_attribut utilisateur=@~*même_type_attribut_utilisateur=value*

Toutefois, si vous utilisez cette syntaxe, vous ne pouvez pas, et vous n'avez pas besoin, d'ajouter spécifiquement "+*type_attribut_utilisateur*=@", après l'élément refusé, donc vous ne définissez pas :

~*type_attribut_utilisateur=value*+*même_type_attribut_utilisateur=@*

Ordre de la définition des utilisateurs

Vous devez classer les définitions d'utilisateur de la plus spécifique à la moins spécifique. Tivoli Workload Scheduler analyse le fichier de sécurité de haut en bas, chaque ID utilisateur étant testé tour à tour par rapport à chaque définition. Si l'ID utilisateur est satisfait par la définition, il est sélectionné et la correspondance s'interrompt.

Par exemple :

Incorrect :

```
#Première définition d'utilisateur du fichier de sécurité
USER TwsUser
CPU=@+LOGON=utilisateur_TWS
Begin
job name=@ access=modify
End

#Deuxième définition d'utilisateur du fichier de sécurité
USER Twsdomain:TwsUser
CPU=@+LOGON=TWSDomain\\utilisateur_TWS
Begin
job name=@ access=display
End
```

Les définitions ont pour but de déterminer les éléments suivants :

1. Les utilisateurs de tous les postes de travail dont l'identifiant de connexion est "TWS_user" disposent de l'accès "modify" sur tous les travaux
2. Les utilisateurs de tous les postes de travail dont l'identifiant de connexion est "TWSDomain\TWS_user" disposent de l'accès "display" sur tous les travaux

Toutefois, tous les utilisateurs dont l'identifiant de connexion est "TWS_user" satisferont la première règle, quel que soit leur domaine, et bénéficieront d'un accès "modify" sur tous les travaux. Cela est dû au fait que la définition d'un utilisateur sans son domaine est un moyen raccourci de définir cet identifiant d'utilisateur dans *tout* domaine ; c'est l'équivalent de "@\TWS_User". Ainsi, la deuxième règle ne sera jamais satisfaite, pour aucun utilisateur, car la mise en correspondance pour "TWS_user" s'arrête dès qu'une correspondance est obtenue.

Correct

```
#Première définition d'utilisateur du fichier de sécurité
USER Twsdomain:Tws_User
CPU=@+LOGON="TWSDomain\\utilisateur_TWS"
Begin
job name=@ access=display
End

#Deuxième définition d'utilisateur du fichier de sécurité
USER Tws_User
CPU=@+LOGON=utilisateur_TWS
Begin
job name=@ access=modify
End
```

En plaçant en première place la définition la plus spécifique, les deux définitions d'accès aux objets sont appliquées correctement.

Voir «Exemple de fichier de sécurité», à la page 191 pour obtenir un exemple pratique.

Types d'attribut utilisateur - description détaillée

Les *types_attribut_utilisateur* et leurs *valeurs* associées peuvent être les suivants :

UC={*poste_de_travail* | **@**}

Où :

poste_de_travail

Précise le poste de travail auquel l'utilisateur est connecté. Les caractères génériques sont acceptés. En outre, vous pouvez utiliser les variables Tivoli Workload Scheduler suivantes :

$master

Signifie que l'utilisateur est connecté sur le gestionnaire de domaine maître de Tivoli Workload Scheduler.

$manager

Signifie que l'utilisateur est connecté sur le gestionnaire de domaine de Tivoli Workload Scheduler.

$thiscpu

Signifie que l'utilisateur est connecté au poste de travail Tivoli Workload Scheduler sur lequel la vérification de sécurité est en cours d'exécution.

@ Indique que l'utilisateur accède à Tivoli Workload Scheduler par l'intermédiaire de Dynamic Workload Console, ou qu'il est connecté sur un poste de travail Tivoli Workload Scheduler.

group=*groupname*

Spécifie le nom du groupe dont l'utilisateur est membre. Disponible à la fois pour les utilisateurs d'UNIX et de Windows. Les caractères génériques sont acceptés.

logon={*nom d'utilisateur* | **@**}

Où :

nom d'utilisateur

Précise l'ID avec lequel l'utilisateur s'est connecté à un poste de travail Tivoli Workload Scheduler. Les caractères génériques sont acceptés. L'attribut **UC=** doit être défini sur le nom d'un poste de travail spécifique (sans caractère générique) ou sur **@**.

Le format de la valeur nom d'utilisateur peut être l'un des suivants :

nom d'utilisateur
Utilisateur Windows. Par exemple, si vous utilisez la valeur utilisateur1 dans le zone de connexion, le fichier Security présente la ligne suivante :

```
.............
logon=utilisateur1
.............
```

domaine\nom d'utilisateur
L'utilisateur appartient à un domaine Windows. Insérez le caractère d'échappement '\' avant le caractère '\' dans la valeur domaine\nom d'utilisateur. Par exemple, si vous utilisez la valeur MYDOMAIN\user1 dans la zone de connexion, le fichier Security présente la ligne suivante :

```
.............
logon=MYDOMAIN\user1
.............
```

nom d'utilisateur@domaine_internet

L'utilisateur appartient à un domaine Internet. Le nom d'utilisateur est au format UPN (User Principal Name). Le format UPN est le nom d'un utilisateur système dans un format d'adresse électronique. Le nom d'utilisateur est suivi du symbole @, lui-même suivi du nom du domaine Internet auquel l'utilisateur est associé.Insérez le caractère d'échappement '\' avant le caractère '@' dans la valeur nom d'utilisateur@domaine_internet. Par exemple, si vous utilisez la valeur administrateur@bvt.com dans la zone de connexion, le fichier Security présente la ligne suivante :

```
.............
logon=administrateur\@bvt_env.com
.............
```

Pour plus d'informations sur l'utilisation d'un caractère générique avec le format domaine\nom d'utilisateur et nom d'utilisateur@domaine_internet dans le fichier Security, voir «Exemple de fichier de sécurité», à la page 191.

Remarque :

1. Si l'option de configuration de sécurité de WebSphere Application Server **useDomainQualifiedUserNames** est définie à *true*, chaque ID utilisateur défini dans le fichier de sécurité doit avoir le format domaine\username pour utiliser le produit à partir de l'une des propositions suivantes :
   - **composer**
   - **Dynamic Workload Console**
   - **logman**
   - **optman**
   - **planman**

   Pour plus d'informations sur la configuration des paramètres de sécurité WebSphere Application Server, voir «Modification des paramètres de sécurité», à la page 406.

2. Si l'utilisateur est défini sur un système Windows 2003 ou lors d'une mise à niveau du système d'exploitation Windows d'une version ancienne vers l'une de celles mentionnées ci-dessus, pensez à ajouter le droit **Emprunter l'identité d'un client après l'authentification** aux paramètres utilisateur.

@ Spécifie tout utilisateur connecté, quel que soit son nom ou appartenant au groupe des administrateurs Tivoli.

Spécification de types d'objet

Spécifiez un ou plusieurs types d'objet auxquels l'utilisateur ou les utilisateurs correspondant à la définition d'utilisateur associée peuvent accéder. Si vous spécifiez le type d'objet mais pas les attributs, les actions autorisées définies pour l'utilisateur avec le mot clé **access** s'appliquent à tous les objets de ce type définis dans le domaine Tivoli Workload Scheduler. Si un type d'objet appartenant à la liste suivante est omis pour un ou plusieurs utilisateurs, aucun objet de ce type n'est accessible.

Les types d'objet sont les suivants :

action Actions définies dans les règles d'événement de planification

calendars
Agendas de l'utilisateur

cpu Postes de travail, domaines et classes de postes de travail

event Conditions d'événement dans les règles d'événement de planification

eventrule
Définitions de règles d'événement de planification

file Fichier base de données Tivoli Workload Scheduler

job Travaux planifiés et définitions de travail

parameter
Paramètres locaux. Voir la remarque ci-dessous.

prompt
Invites globales

report Les rapports figurant sur Dynamic Workload Console et portant les *noms* suivants :

RUNHIST
Historique d'exécution du travail

RUNSTATS
Statistiques d'exécution du travail

WWS Récapitulatif de la charge de travail du poste de travail

WWR Temps d'exécution de la charge de travail du poste de travail

SQL SQL personnalisé

ACTPROD
Détails de la production réelle (pour les plans en cours et archivés)

PLAPROD
Détails de la production planifiée (pour les plans d'essai et les plans prévisionnels)

L'autorisation d'utilisation de ces rapports est accordée par défaut au *utilisateur_TWS* sur les nouvelles installations.

resource
Ressources de planification

runcygrp
Groupes de cycle d'exécution

schedule
Flots de travaux

userobj
Objets utilisateur

vartable
Tables de variables. Cela comprend l'autorisation d'accès aux définitions des variables dans les tables. Voir la remarque ci-dessous.

wkldappl
Applications de charge de travail

Remarque : A partir de la version 8.5, le type d'objet **parameter** est réservé aux paramètres créés et gérés dans une base de données de paramètres locale avec la commande de l'utilitaire `parms`, alors que l'autorisation d'agir sur les variables globales est gérée grâce au type d'objet **vartable**. C'est la raison pour laquelle, lorsque le fichier de sécurité est migré depuis des versions antérieures vers la version 8.5, une définition de sécurité `vartable` pour la table de variables par défaut est ajoutée pour correspondre à chaque définition `parameter` trouvée, dans le cadre du processus de mise à niveau documenté dans le *Tivoli Workload Scheduler - Guide de planification et d'installation*.

Spécification des attributs d'objet

Indique un ou plusieurs attributs spécifiant un ensemble d'objets auxquels l'utilisateur de la définition utilisateur est autorisé à accéder. Si vous spécifiez le type d'objet, mais pas d'ensembles d'objets, les actions autorisées définies pour l'utilisateur avec le mot clé **access** s'appliquent à tous les objets de ce type définis dans le domaine de Tivoli Workload Scheduler.

Syntaxe générale

Chaque attribut d'objet est spécifié comme suit :

attribut_objet=*value*

attribut_objet
Les attributs d'objet varient selon les objets. Tous les objets peuvent être sélectionnés par *nom*, mais certains, tels que *jobs*, peuvent être sélectionnés par le *workstation* sur lequel il s'exécutent. Voir «Attribut d'objet» pour plus d'informations sur les attributs disponibles pour chaque type d'objet.

valeur Identifie un objet individuel ou, à l'aide de caractères génériques, un ensemble d'objets. Voir «Spécification des attributs d'objet» pour plus d'informations sur les attributs disponibles pour chaque type d'objet.

Attribut d'objet

La section «Spécification des attributs d'objet» répertorie les attributs d'objet qui sont utilisés pour identifier un ensemble spécifique d'objets au sein de tous les objets du même type. Pare exemple, l'accès peut être limité à un ensemble d'objets ressources portant le même nom ou étant définis sur le même poste de travail, ou les deux.

Tableau 31. Types d'attributs d'objet pour chaque type d'objet

Attribut / Objet	nom	cpu	custom	jcl	jcltype	logon	provider	type	host	port
action							✓	✓	✓	✓
calendar	✓									
cpu (poste de travail)		✓						✓		
event			✓				✓	✓		
eventrule	✓									
file	✓									
job	✓	✓		✓	✓	✓				
parameter	✓	✓								
prompt	✓									
report	✓									
resource	✓	✓								
runcygrp	✓									
schedule (flot de travaux)	✓	✓								
userobj	✓	✓				✓				
vartable	✓									
wkldappl	✓									

Inclusion ou exclusion

Chaque attribut peut être *inclus* ou *exclu* de la sélection à l'aide du signe plus (+) et du tilde (~), de la même manière que pour les attributs d'utilisateur.

Expressions de sélection

La syntaxe détaillée et l'utilisation des expressions de sélection pour les objets sont les mêmes que celles utilisées pour sélectionner les utilisateurs :

[~]*attribut_objet*=*value*[,*value*]...[{+ | ~}*attribut_objet*=*value*[,*value*]...

Ordre de la définition des objets

Vous devez classer les définitions d'objet de la plus spécifique à la moins spécifique, de la même manière que pour les attributs d'utilisateur. Par exemple

Incorrect

```
job name=@ access=display
job name=ar@ access=@
```

Dans ce cas, un travail dont le nom commence par "ar" satisferait la première définition, et recevrait donc le droit d'accès à l'affichage (display), mais pas un accès global (all).

Correct

```
job name=ar@ access=@
job name=@ access=display
```

Assurez-vous que vous classez également les définitions d'objet de la plus spécifique à la moins spécifique lorsque vous utilisez le mot clé `Continue`. Avec ce mot clé, vous faites correspondre davantage de définitions d'utilisateur à un même

utilisateur, de sorte que ce dernier reçoit des accès de la part de davantage d'instructions de définition d'utilisateur. Ces accès sont ensuite traités dans l'ordre dans lequel ils sont écrits dans le fichier de sécurité. Pour obtenir un exemple de fichier de sécurité avec le mot clé `Continue`, voir «Utilisateurs connectés à plusieurs groupes [mot clé continue]», à la page 195

Spécification des valeurs d'attribut des objets

La formule suivante décrit les valeurs admises pour chaque type d'attribut d'objet :

name=*name*[**,***nom*]**...**

Affectez un ou plusieurs noms au type d'objet. Les caractères génériques sont acceptés. Si vous indiquez plusieurs noms, utilisez la virgule comme séparateur.

- Les valeurs suivantes s'appliquent au type d'objet **`file`** :

  globalopts
  Permet à l'utilisateur de définir les options globales avec la commande `optman`. Autorise les types d'accès suivants :
  - Accès en consultation pour `optman ls` et `optman show`
  - Accès en modification pour `optman chg`

  prodsked
  Permet à l'utilisateur de créer, d'étendre ou de réinitialiser le plan de production.

  sécurité
  Permet à l'utilisateur de gérer le fichier de sécurité.

  Symphony
  Autorise l'utilisateur à exécuter **stageman** et **JnextPlan**.

  trialsked
  Permet à l'utilisateur de créer des plans d'essai et de prévision ou d'étendre des plans d'essai.

  Remarque : Les utilisateurs disposant d'un accès restreint aux fichiers devraient disposer au moins du droit d'accès suivant pour pouvoir afficher d'autres objets (par ex. agendas et UC).

  ```
  file          name=globalopts  access=display
  ```

- Pour le type d'objet **`event`**, utilisez un ou plusieurs noms de type d'événement répertoriés dans la table de s événements*`TWSObjectsMonitor`* ou dans la table des événements *`FileMonitor`* dans le *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.
- Pour le type d'objet **`action`**, utilisez un ou plusieurs noms de type d'objet répertoriés dans la table *Action types by action provider* du *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.
- Pour le type d'objet **vartable**, vous pouvez utiliser la valeur $DEFAULT pour l'attribut **name** pour indiquer la table de variables par défaut. Cela sélectionne la table définie avec l'attribut `isdefault`.

cpu=*workstation*[**,***workstation*]**...**

Spécifie un ou plusieurs postes de travail, domaines ou noms de classe de poste de travail. Les caractères génériques sont acceptés. Si vous indiquez plusieurs noms, utilisez la virgule comme séparateur. Si vous omettez cet attribut, tous les postes de travail et les domaines définis restent accessibles. Les variables de poste de travail peuvent être utilisées (voir «Utilisation de variables dans les définitions d'attribut des objets», à la page 174).

custom=value[,*value*]**...**

Utilisez cet attribut pour attribuer des droits d'accès aux événements définis dans les plug-ins d'événement. La syntaxe précise de la valeur dépend du plug-in. Par exemple :

- Spécifiez différents droits pour différents utilisateurs en fonction de noms d'événement SAP R/3 lors de la définition de règle d'événement pour les événements SAP R/3.
- Définissez votre propre attribut de sécurité pour vos fournisseurs d'événements personnalisés.
- Spécifiez le type d'événement à surveiller. Chaque événement peut être associé à un fournisseur d'événements.

jcl="*path***" | "***command***" | "***jsdl***"**

Précisez la commande ou le nom du chemin du fichier exécutable d'un objet travail. Vous devez placer la commande ou le chemin entre guillemets (" "). Les caractères génériques sont acceptés. Si vous omettez cet attribut, tous les fichiers de travail et toutes les commandes définis sont éligibles.

Vous pouvez également spécifier une chaîne contenue dans la chaîne de tâches d'une définition JSDL à utiliser pour la mise en correspondance de modèle. Vérifiez que la chaîne commence et se termine par le caractère générique @ et qu'elle est placée entre guillemets comme suit : `"@<ma_chaîne>@"`.

jcltype=[scriptname | docommand]

Indique que l'utilisateur est autorisé à agir sur les définitions de travaux exécutant uniquement des scripts (si défini avec **scriptname**) ou des commandes (si défini avec **docommand**). Utilisez cet attribut facultatif pour limiter les autorisations utilisateur sur les actions portant sur les définitions de travaux de l'un ou l'autre type. Les actions sont octroyées pour les scripts et les commandes lorsque le paramètre **jcltype** est manquant.

Un utilisateur auquel serait refusée l'autorisation de travailler sur les définitions de travaux exécutant une commande ou un script obtient un message d'erreur de sécurité s'il tente d'exécuter une action sur ces définitions.

logon=*username***[,...]**

Spécifie les ID d'utilisateur. Les caractères génériques sont acceptés. Si vous indiquez plusieurs noms, utilisez la virgule comme séparateur. Si vous omettez cette option, tous les ID d'utilisateur sont qualifiés.

L'ID utilisateur peut être un utilisateur de domaine Windows ou un utilisateur de domaine Internet ; il doit être défini dans l'un des formats suivants :

domaine\nom d'utilisateur

L'utilisateur appartient à un domaine Windows. Insérez le caractère d'échappement '\' avant le caractère '\' dans la valeur `domaine\nom d'utilisateur`. Par exemple, si vous utilisez la valeur `MYDOMAIN\user1` dans la zone de connexion, le fichier `Security` présente la ligne suivante :

```
.............
logon=MYDOMAIN\user1
.............
```

nom d'utilisateur@domaine_internet

L'utilisateur appartient à un domaine Internet. Le nom d'utilisateur est au format UPN (User Principal Name). Le format UPN est le nom d'un utilisateur système dans un format d'adresse électronique. Le nom d'utilisateur est suivi du symbole @, lui-même suivi du nom du domaine Internet auquel l'utilisateur est associé.Insérez le caractère d'échappement '\' avant le caractère '@' dans la valeur `nom d'utilisateur@domaine_internet`. Par exemple, si vous utilisez la valeur `administrateur@bvt.com` dans la zone de connexion, le fichier `Security` présente la ligne suivante :

```
.............
logon=administrateur\@bvt_env.com
.............
```

provider=*nom_fournisseur***[,...]**

Pour les types d'objet **action** spécifie le nom du fournisseur d'action.

Pour les types d'objet **event**, indique le nom du fournisseur d'événement.

Les caractères génériques sont acceptés. Si vous indiquez plusieurs noms, utilisez la virgule comme séparateur. Si `provider` n'est pas spécifié, aucun objet défini n'est accessible.

type=*type***[,...]**

Pour les types d'objet **action**, il s'agit de `actionType`.

Pour les types d'objet **event**, il s'agit de `eventType`.

Pour les types d'objet **cpu**, les valeurs admises sont celles qui sont utilisées dans **`composer`** ou dans le Dynamic Workload Console lors de la définition des postes de travail, par exemple `manager`, `broker`, `fta`, `agent`, `s-agent`, `x-agent`, `rem-eng`, `pool` et `d-pool`.

Remarque : La valeur de `master`, utilisée dans **`conman`** est mappée aux attributs de sécurité `manager`.

Les caractères génériques sont acceptés. Si vous indiquez plusieurs noms, utilisez la virgule comme séparateur. Si **type** n'est pas indiqué, tous les objets définis sont accessibles pour les fournisseurs spécifiés (c'est toujours le cas après une installation ou une mise à niveau, car l'attribut de type n'est pas fourni par défaut).

host=*host_name*

Pour les types d'objet **action**, spécifie le nom d'hôte TEC ou SNMP (utilisé pour certains types d'action tels que l'envoi d'événements TEC ou l'envoi SNMP). Si cela ne s'applique pas, cette zone doit être vide.

port=*port_number*

Pour les types d'objet **action**, spécifie le numéro de port TEC ou SNMP (utilisé pour certains types d'action tels que l'envoi d'événements TEC ou l'envoi SNMP). Si cela ne s'applique pas, cette zone doit être vide.

Utilisation de variables dans les définitions d'attribut des objets

Les variables de sécurité suivantes fournies avec le produit peuvent être utilisées dans les attributs d'objet :

Identifiants de poste de travail

$master

gestionnaire de domaine maître de Tivoli Workload Scheduler.

$manager
Gestionnaire de domaine de Tivoli Workload Scheduler.

$thiscpu
Poste de travail sur lequel l'utilisateur exécute la commande ou le programme Tivoli Workload Scheduler

Identificateurs de table de variables

$default
Nom de la table de variables par défaut en cours.

Définition de l'accès

Spécifiez le type d'accès accordé aux utilisateurs sélectionnés pour les objets spécifiés comme suit :

access[=*mot clé*[,*mot clé*]...]

- Pour spécifier qu'aucune action n'est autorisée, utilisez **access=**
- Pour spécifier que toutes les actions sont autorisées, utilisez **access=@**
- Pour spécifier un autre accès, consultez les tables d'accès (par type d'objet) ci-dessous.

Organisation des tables d'accès

Les tables d'accès des types d'objet sont les suivantes :

«Types d'objet - calendar, cpu, eventrule, job, prompt, resource, run cycle group, schedule, userobj, vartable - Utilisation dans composer», à la page 176
La plupart des actions de maintenance de **composer** et de la base de données de l'interface graphique sont communes à la plupart des objets. Elles sont donc répertoriées dans une table de mots clés communs d'accès aux objets.

«Type d'objet - action», à la page 179
Donne les droits d'accès aux objets d'action, qui ne sont pas inclus dans le tableau commun.

«Type d'objet - agenda», à la page 179
Donne les droits d'accès aux agendas, qui sont différents ou viennent en compléments de ceux du tableau commun.

«Type d'objet - cpu», à la page 180
Donne les droits d'accès aux postes de travail (unités centrales), qui sont différents ou viennent en compléments de ceux du tableau commun.

«Type d'objet - event», à la page 181
Donne les droits d'accès aux événements, qui sont différents ou viennent en compléments de ceux du tableau commun.

«Type d'objet - file», à la page 181
Donne les droits d'accès aux fichiers, qui sont différents ou viennent en compléments de ceux du tableau commun.

«Type d'objet - job», à la page 182
Donne les droits d'accès aux travaux, qui sont différents ou viennent en compléments de ceux du tableau commun.

«Type d'objet - parameter», à la page 185
Donne les droits d'accès aux paramètres locaux, qui ne sont pas inclus dans le tableau commun.

«Type d'objet - prompt», à la page 185
Donne les droits d'accès aux invites, qui sont différentes ou viennent en compléments de celles du tableau commun.

«Type d'objet - report», à la page 186
Donne les droits d'accès aux rapports, qui sont différents ou viennent en compléments de ceux du tableau commun.

«Type d'objet - resource», à la page 186
Donne les droits d'accès aux ressources, qui sont différentes ou viennent en compléments de celles du tableau commun.

«Type d'objet - groupe de cycle d'exécution», à la page 187
Donne les droits d'accès aux groupes de cycle d'exécution, qui sont différents ou viennent en compléments de ceux du tableau commun.

«Type d'objet - schedule», à la page 187
Donne les droits d'accès aux flots de travaux (plannings), qui sont différents ou viennent en compléments de ceux du tableau commun.

«Type d'objet - userobj», à la page 188
Donne les droits d'accès à userobj, qui sont différentes ou viennent en compléments de celles du tableau commun.

«Type d'objet - vartable», à la page 189
Donne les droits d'accès aux tables de variables, qui ne sont pas incluses dans le tableau commun.

«Type d'objet - application de charge de travail», à la page 189
Donne les droits d'accès aux applications de charge de travail, qui ne sont pas incluses dans le tableau commun.

Types d'objet - calendar, cpu, eventrule, job, prompt, resource, run cycle group, schedule, userobj, vartable - Utilisation dans composer

Le tableau suivant fournit les mots clés d'accès requis pour utiliser le composeur pour qu'il travaille avec les objets des types suivants :

- calendar
- cpu
- eventrule
- job
- prompt
- resource
- run cycle group
- schedule
- userobj
- vartable

Remarque : A partir de la version 8.5, le mot clé `parameter` est réservé aux paramètres créés et gérés dans une base de données de paramètres locale avec la commande d'utilitaire `parms`. Pour plus d'informations sur `parms`, voir *Guide d'utilisation et de référence*.

Tableau 32. Mots clés d'accès pour les actions du composeur

Activité			**Mots clés d'accès requis**
Composer	add	Ajouter de nouvelles définitions d'objet à la base de données à partir d'un fichier de définitions d'objet. Un accès au déverrouillage est requis pour utiliser l'attribut `;unlock`. Pour *ajouter* des entrées de variable individuelles dans une table, cette dernière doit détenir l'accès *modify*.	add, modify, unlock
	create	Créez un fichier texte de définitions d'objet dans la base de données. L'accès Modify est nécessaire pour utiliser l'attribut `;lock`. Pour les tables de variables, créez des entrées de variables individuelles au sein de la table.	display, modify
	delete	Supprimez les définitions d'objet de la base de données. Pour *supprimer* des entrées de variable individuelles d'une table, cette dernière doit détenir l'accès *modify*.	delete
	display	Affichez les définitions d'objet dans la base de données.	display
	extract	Permet d'extraire un fichier texte de définitions d'objet de la base de données.	display
	list	Répertoriez les définitions d'objet dans la base de données.	Si l'option globale **`enListSecChk`** est définie sur `oui` dans le gestionnaire de domaine maître, alors les mots clés list ou list et display sont requis.
	lock	Verrouillez les définitions d'objet dans la base de données.	modify
	modify	Modifiez les définitions d'objet dans la base de données. Les définitions sont extraites dans un fichier. Une fois le fichier modifié, les définitions sont utilisées pour remplacer les définitions existantes. Pour *modifier* des entrées de variable individuelles dans une table, cette dernière doit détenir l'accès *modify*.	add, modify
	new	Créez des définitions d'objet dans la base de données à partir d'un modèle.	add, modify
	print	Imprimez les définitions d'objet dans la base de données.	display
	rename	Renommez les définitions d'objet dans la base de données. Vous devez ajouter un accès au nouvel objet et supprimer et afficher l'accès à l'ancien objet.	add, delete, display
	replace	Remplacez les définitions d'objet dans la base de données. Un accès au déverrouillage est requis pour utiliser l'attribut `;unlock`.	add, modify, unlock
	unlock	Déverrouillez les définitions d'objet dans la base de données. Pour les tables de variables, le déverrouillage d'une table déverrouille toutes les variables qu'elle contient. Le déverrouillage d'une variable contient la table entière dans laquelle elle est définie.	unlock

Tableau 32. Mots clés d'accès pour les actions du composeur (suite)

Activité			**Mots clés d'accès requis**
Dynamic Workload Console	Créez un objet dans la base de données.	Ajoutez de nouvelles définitions d'objet à la base de données.	add
	Supprimez un objet dans la base de données	Supprimez les définitions d'objet de la base de données. Un accès au déverrouillage est nécessaire pour utiliser l'option `;unlock`.	delete
	Affichez un objet dans la base de données.	Affichez les définitions d'objet dans la base de données.	display
	Répertoriez un objet dans la base de données	Répertoriez les définitions d'objet dans la base de données.	display
	Modifiez un objet dans la base de données.	Modifiez les définitions d'objet dans la base de données. Un accès au déverrouillage est nécessaire pour utiliser l'option `;unlock`.	modify
	Déverrouillez un objet dans la base de données	Déverrouillez des définitions d'objet dans la base de données verrouillées par un autre utilisateur.	unlock
	Procédez aux opérations pour les types de travail avec options avancées, à la fois ceux fournis avec le produit et les types supplémentaires implémentés via les plug-ins personnalisés. Vous pouvez définir et exécuter des opérations sur les types de travail avec options avancées avec Workload Designer.	Procédez aux opérations pour les types de travail avec options avancées dans la base de données.	run
Utilisation de la fonctionnalité assurance de service de charge de travail	Toutes les activités	Pour qu'un utilisateur puisse réaliser des activités d'assurance de service de charge de travail, le *utilisateur_TWS* doit comporter les mots clés d'accès suivants pour tous les objets *cpu*, *job* et *schedule* :	display, modify, list

Exemple : Pour permettre à un utilisateur d'utiliser la liste du composeur, d'afficher et de modifier des actions sur les règles d'événement, spécifiez :

```
eventrule          access=add,display,modify
```

Type d'objet - action

La table suivante indique les mots clés d'accès requis pour les actions :

Tableau 33. Actions - mots clés d'accès

Activité		Mots clés d'accès requis
Dynamic Workload Console	Affichage des instances d'action	display
	Répertorier les instances d'action.	list
Dynamic Workload Console **conman**	Utilisez ces types d'action spécifiques dans les définitions de règle d'événement. • Pour les actions avec le fournisseur `TWSAction` et les types `sbj`, `sbd` ou `sbs`, vous devez définir ce mot clé en combinaison avec le mot clé d'accès `submit` des travaux spécifiques et les flots de travaux spécifiés dans l'action. • Pour les actions avec le fournisseur `TWSAction` et le type `reply`, vous devez définir ce mot clé en combinaison avec le mot clé d'accès `reply` défini pour les invites spécifiques indiquées dans l'action. Le *utilisateur_TWS* du poste de travail exécutant le serveur de traitement d'événement doit disposer de ces autorisations `submit` et `reply` ; dans le cas contraire, le serveur de traitement d'événement ne pourra pas exécuter ce type d'action.	use

Exemple : Pour permettre à un utilisateur d'utiliser Dynamic Workload Console afin de répertorier des instances d'action, spécifiez :

```
action          access=list
```

Type d'objet - agenda

La table suivante fournit les mots clés d'accès supplémentaires requis pour travailler avec les agendas, en plus de ceux décrits dans le tableau 32, à la page 177 :

Tableau 34. Agenda - mots clés d'accès supplémentaires

Activité		Mots clés d'accès requis
Composer Dynamic Workload Console	Utilisez les agendas dans : • les flots de travaux • les cycles d'exécution • les groupes de cycle d'exécution	use

Exemple 1 : Pour permettre à un utilisateur d'utiliser seulement les agendas lorsqu'il travaille avec des flots de travaux dans toute interface, spécifiez :

```
calendar                    access=use
```

Exemple 2 : Pour permettre à un utilisateur d'afficher, de répertorier et d'imprimer des agendas, et de les utiliser en travaillant sur des flots de travaux dans n'importe quelle interface, spécifiez :

```
calendar          access=display,use,list
```

Type d'objet - cpu

Le tableau suivant fournit les mots clés d'accès supplémentaires requis pour utiliser des UC (postes de travail, domaines et classes de postes de travail) autres que ceux décrits dans le tableau 32, à la page 177:

Tableau 35. UC - mots clés d'accès supplémentaires

Activité			**Mots clés d'accès requis**
Conman Dynamic Workload Console	console	Modifier et envoyer des messages à la console Tivoli Workload Scheduler **conman**.	console
	deployconf	Forcez la mise à jour du fichier de configuration de surveillance pour le moteur de surveillance d'événement.	start
	fence	Modifiez la priorité minimale des travaux du poste de travail dans le plan de production.	fence
	limit cpu	Modifiez le nombre maximum de travaux du poste de travail dans le plan de production.	limit
	link	Etablissez les liaisons vers un poste de travail.	link
	resetfta	Génère un fichier Symphony mis à jour et l'envoie à un agent tolérant aux pannes sur lequel le fichier Symphony était corrompu.	resetfta
	showcpus	Affichez les postes de travail, les domaines et les liaisons dans le plan.	list
	shutdown	Arrêter le traitement Tivoli Workload Scheduler.	shutdown
	start	Lancer le traitement Tivoli Workload Scheduler.	start
	startappserver	Démarrez le serveur d'applications.	start
	starteventprocessor	Démarrez le serveur de processeur d'événements.	start
	startmon	Démarrez le moteur de surveillance d'événements.	start
	stop	Arrêter le traitement Tivoli Workload Scheduler.	stop
	stop;progressive	Arrêter le traitement du Tivoli Workload Scheduler progressivement	stop
	stopappserver	Arrêtez le serveur d'applications.	stop
	stopeventprocessor	Arrêtez le serveur de processeur d'événements.	stop
	stopmon	Arrêtez le moteur de surveillance d'événements.	stop
	switcheventprocessor	Permuter le serveur du processeur d'événement à partir du gestionnaire de domaine maître vers le gestionnaire de domaine maître de secours ou vice versa.	start, stop
	switchmgr	Transfère la fonctionnalité du gestionnaire de domaine vers un poste de travail.	start, stop
	unlink	Fermez les liaisons vers le poste de travail.	unlink
Startup	Lancer le traitement Tivoli Workload Scheduler.		start
Utilisation de la fonctionnalité assurance de service de charge de travail	Toutes les activités	Pour qu'un utilisateur puisse réaliser des activités d'assurance de service de charge de travail, l'*utilisateur_TWS* doit posséder les mots clés d'accès suivants :	display, modify, list

Exemple : Pour permettre à un utilisateur d'afficher, de répertorier et d'imprimer les définitions de poste de travail, de classe de poste de travail et de domaine, et d'établir ou annuler les liaisons entre postes de travail, spécifiez :

```
cpu          access=display,link,unlink
```

Type d'objet - event

Le tableau suivant fournit les mots clés d'accès requis pour utiliser des événements :

Tableau 36. Evénéments - mots clés d'accès

Activité		**Mots clés d'accès requis**
Composer Dynamic Workload Console	Utilisez un événement dans une définition de règle d'événement.	use

Exemple : Pour permettre à un utilisateur d'utiliser un événement dans une définition de règle, spécifiez :

```
event          access=use
```

Type d'objet - file

Le tableau suivant fournit les mots clés d'accès requis pour utiliser des fichiers (valide uniquement pour la ligne de commande).

Vous devez spécifier les noms de fichier auxquels s'applique le type d'accès.

Tableau 37. Fichiers - mots clés d'accès

Activité			**Mots clés d'accès requis**
Supprimez des objets de la base de données.			delete
dumpsec	Créez un fichier texte des paramètres figurant dans le fichier de sécurité compilé.		display
JnextPlan	Générez le plan de production.		build
makesec	Compilez le fichier de sécurité à partir d'un fichier texte des paramètres.		modify
optman	ls	Répertoriez toutes les options globales.	display
	show	Affichez les détails d'une option globale.	display
	change	Modifiez les détails d'une option globale.	modify
planman	deploy	Déployez manuellement des règles d'événement.	build
prodsked	Travaillez avec le plan de production.		build
stageman	Reportez les flots de travaux inachevés, archivez l'ancien plan de production et installez le nouveau plan de production.		build

Exemple 1 : Pour permettre à un utilisateur de gérer le fichier de sécurité, spécifiez :

```
file          access=display,modify
```

Exemple 2 : Pour permettre à un utilisateur d'exécuter **JnextPlan**, spécifiez :

```
file          access=build
```

Remarque : L'utilisateur pourra également exécuter **planman deploy**, **prodsked** et **stageman**.

Type d'objet - job

Le tableau suivant fournit les mots clés d'accès supplémentaires requis pour utiliser des travaux, à l'exception de ceux décrits dans le tableau 32, à la page 177 :

Tableau 38. Travaux - mots clés d'accès supplémentaires

Activité			**Mots clés d'accès requis**
Composer Dynamic Workload Console	Utilisez les travaux dans les flots de travaux. De même, si un travail fait office de travail de reprise dans une définition de travail, l'utilisateur doit détenir l'accès "use" à la définition du travail identifiée comme travail de reprise.		use
Conman Dynamic Workload Console	adddep	Ajoutez des dépendances aux travaux du plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	adddep
	altpri	Modifiez la priorité des travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	altpri
	cancel job	Annulez les travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	cancel
	confirm	Confirmez l'exécution des travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	confirm
	deldep job	Supprimez les dépendances des travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	deldep
	display	Affichez les travaux dans le plan.	display
	kill	Arrêtez des travaux en cours d'exécution.	kill
	release job	Libérez les travaux de leurs dépendances dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	release
	reply	Répondez aux invites des travaux dans le plan de production.	reply
	rerun	Relancez les travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	rerun
	showjobs	Affichez des informations sur les travaux du plan de production.	list

Tableau 38. Travaux - mots clés d'accès supplémentaires (suite)

Activité			**Mots clés d'accès requis**
Conman Dynamic Workload Console	submit docommand	Soumettez les commandes en tant que travaux ou travaux de reprise dans le plan de production. Si **submit** identifie également un deuxième travail avec les arguments "ALIAS" ou "RECOVERYJOB", l'utilisateur doit détenir l'accès "submit" à cet autre travail également. Non valide pour les postes de travail dans un environnement de bout en bout.	submit
	submit file	Soumettez les fichiers en tant que travaux ou travaux de reprise dans le plan de production. Si **submit** identifie également un deuxième travail avec les arguments "ALIAS" ou "RECOVERYJOB", l'utilisateur doit détenir l'accès "submit" à cet autre travail également. Non valide pour les postes de travail dans un environnement de bout en bout.	submit
	submit job	Soumettez les travaux ou les travaux de reprise dans le plan de production. Si **submit** identifie également un deuxième travail avec les arguments "ALIAS" ou "RECOVERYJOB", l'utilisateur doit détenir l'accès "submit" à cet autre travail également. Non valide pour les postes de travail dans un environnement de bout en bout.	submit
		Limite l'action de soumission aux travaux définis dans la base de données. Avec ce niveau d'autorisation, un utilisateur ne peut pas soumettre de travaux ad hoc. Utilisez ce mot clé pour autoriser un utilisateur à soumettre uniquement les travaux définis dans la base de données. Utilisez le mot clé **submit** pour autoriser un utilisateur à soumettre à la fois les travaux définis et les travaux ad hoc. Les utilisateurs auxquels seuls les droits **submitdb** ont été octroyés : • Ne parviennent pas à exécuter les commandes **submit docommand** et **submit file** • Peuvent consulter les tâches liées à la soumission de travaux ad hoc sur les interfaces graphiques mais s'ils exécutent ces tâches, ils obtiennent des messages d'erreur indiquant qu'il leur manque le droit d'accès **submit**.	submitdb
	submit sched	Soumettez des flots de travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	submit

Tableau 38. Travaux - mots clés d'accès supplémentaires (suite)

Activité			Mots clés d'accès requis
Dynamic Workload Console	Pour les travaux critiques sur lesquels vous exécutez l'une des actions suivantes : • Affichez la liste des plans • Affichez le chemin d'accès critique • Affichez les prédécesseurs inachevés • Affichez les prédécesseurs achevés	Les prédécesseurs sont répertoriés, même si cette autorisation peut ne pas les concerner. Toutefois, pour exécuter toute autre action sur l'un des prédécesseurs répertoriés, cela nécessite que vous disposiez de l'autorisation appropriée.	list
Utilisation de la fonctionnalité assurance de service de charge de travail	Toutes les activités	Pour qu'un utilisateur puisse réaliser des activités d'assurance de service de charge de travail, l'*utilisateur_TWS* doit posséder les mots clés d'accès suivants :	display, modify, list

Exemple 1 : Pour permettre à un utilisateur de gérer uniquement les dépendances entre travaux, spécifiez :

```
job          access=adddep,deldep
```

Exemple 2 : Pour permettre à un utilisateur de gérer uniquement des travaux critiques, spécifiez :

```
job          access=list,altpri
```

Exemple 3 : L'utilisateur `administrator` détient les droits **`add`** et **`modify`** pour toutes les définitions de travail et il est habilité à créer et modifier les définitions de travail qui exécutent les scripts ou les commandes nécessaires, sans aucune restriction :

```
USER TWSADMIN
CPU=@+LOGON=administrator
BEGIN
JOB CPU=@ ACCESS=ADD,MODIFY,DISPLAY,...
[...]
FIN
```

L'utilisateur `sconnor` détient les mêmes droits pour les travaux qui remplissent la condition **`jcltype=scriptname`**, ce qui signifie qu'il peut créer ou modifier uniquement les définitions de travail qui exécutent des scripts et ne peut en modifier aucune dans un travail exécutant une commande :

```
USER RESTRICTED
CPU=@+LOGON=sconnor
BEGIN
JOB CPU=@+JCLTYPE=SCRIPTNAME ACCESS=ADD,MODIFY,DISPLAY,...
[...]
FIN
```

Exemple 4 : L'utilisateur `administrator` détient les droits **submit** pour tous les travaux ; il est ainsi habilité à soumettre les travaux définis dans la base de données et les travaux ad hoc, sans aucune restriction :

```
USER TWSADMIN
CPU=@+LOGON=administrator
BEGIN
JOB CPU=@ ACCESS=ADD,ADDDEP,...,RERUN,SUBMIT,USE,LIST,UNLOCK
[...]
FIN
```

L'utilisateur `jsmith` détient les droits **submitdb** pour tous les travaux, ce qui lui permet de soumettre tous les travaux définis dans la base de données, mais elle n'est pas habilitée à soumettre des travaux ad hoc :

```
USER RESTRICTED
CPU=@+LOGON=jsmith
BEGIN
JOB CPU=@ ACCESS=ADD,ADDDEP,...,RERUN,SUBMITDB,USE,LIST,UNLOCK
[...]
FIN
```

Type d'objet - parameter

Le tableau suivant fournit les mots clés d'accès requis pour utiliser des paramètres :

Remarque : A partir de la version 8.5, le mot clé `parameter` est réservé aux paramètres créés et gérés dans une base de données de paramètres locale avec la commande d'utilitaire `parms`. Reportez-vous au *Tivoli Workload Scheduler - Guide d'utilisation et de référence* pour plus de détails sur `parms`.

Tableau 39. Paramètres - mots clés d'accès supplémentaires

Activité		Mots clés d'accès requis
parms	Gère les définitions des paramètres locaux.	display

Exemple : Pour permettre à un utilisateur de réaliser toutes les activités sur les paramètres, spécifiez :

```
parameter          access=@
```

Type d'objet - prompt

Le tableau suivant fournit les mots clés d'accès supplémentaires requis pour utiliser des invites, autres que ceux décrits dans le tableau 32, à la page 177:

Tableau 40. Invites - mots clés d'accès supplémentaires

Activité		Mots clés d'accès requis
Composer Dynamic Workload Console	Utilisez les invites pour définir ou soumettre des travaux et des flots de travaux	use

Tableau 40. Invites - mots clés d'accès supplémentaires (suite)

Activité			Mots clés d'accès requis
Conman Dynamic Workload Console	adddep	Utilisez les invites lors de l'ajout de dépendances à des travaux du plan de production Non valide pour les postes de travail dans un environnement de bout en bout.	use
	recall	Affichez des invites qui attendent une réponse.	display
	reply	Répondez à une invite de travail ou de Planificateur de travaux.	reply
	showprompts	Affichez des informations relatives aux invites.	list
	submit docommand	Utilisez les invites pour soumettre des commandes en tant que travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	use
	submit file	Utilisez les invites pour soumettre des fichiers en tant que travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	use
	submit job	Utilisez les invites pour soumettre des travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	use
	submit sched	Utilisez les invites pour soumettre des flots de travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	use

Exemple : Pour permettre à un utilisateur de réaliser toutes les activités sur les invites sauf d'y répondre, spécifiez :

```
prompt          access=use,display,list
```

Type d'objet - report

Le tableau suivant fournit les mots clés d'accès requis pour utiliser des rapports.

Tableau 41. Fichiers - mots clés d'accès

Activité		Mots clés d'accès requis
Dynamic Workload Console	Affiche les rapports sur Dynamic Workload Console.	display

Exemple : Pour permettre à un utilisateur d'afficher des rapports sur Dynamic Workload Console, spécifiez :

```
report          access=display
```

Type d'objet - resource

Le tableau suivant fournit les mots clés d'accès supplémentaires requis pour travailler avec des ressources, autres que ceux décrits dans le tableau 32, à la page 177 :

Tableau 42. Ressources - mots clés d'accès supplémentaires

Activité			Mots clés d'accès requis
Composer Dynamic Workload Console	Utilisez les ressources pour définir ou soumettre des travaux et des flots de travaux		use
Conman Dynamic Workload Console	adddep	Utilisez les ressources pour ajouter des dépendances à des travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	use
	resource	Changez le nombre d'unités d'une ressource sur un poste de travail.	resource
	showresources	Affichez les informations relatives aux ressources.	list
	submit docommand	Utilisez les ressources pour soumettre des commandes en tant que travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	use
	submit file	Utilisez les ressources pour soumettre des fichiers en tant que travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	use
	submit job	Utilisez les ressources pour soumettre des travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	use
	submit sched	Utilisez les ressources pour soumettre des flots de travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	use

Exemple : Pour permettre à un utilisateur d'afficher les informations relatives aux ressources et de modifier les unités d'une ressource sur un poste de travail, mais pas de les utiliser dans d'autres objets ou actions de planification, spécifiez :

```
resource          access=list,resource
```

Type d'objet - groupe de cycle d'exécution

Le tableau suivant fournit les mots clés d'accès requis pour utiliser des groupes de cycle d'exécution :

Tableau 43. Groupes de cycle d'exécution - mots clés d'accès

Activité		Mots clés d'accès requis
Composer Dynamic Workload Console	Utilisez des groupes de cycle d'exécution dans les flots de travaux.	use

Exemple : Pour autoriser un utilisateur à créer et supprimer un groupe de cycle d'exécution, spécifiez :

```
runcygrp          access=add,delete
```

Type d'objet - schedule

Le tableau suivant fournit les mots clés d'accès supplémentaires requis pour utiliser des flots de travaux, autres que ceux décrits dans le tableau 32, à la page 177:

Tableau 44. Flots de travaux - mots clés d'accès supplémentaires

Activité			**Mots clés d'accès requis**
Conman Dynamic Workload Console	adddep	Ajoutez des dépendances à des flux de travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	adddep
	altpri	Modifiez la priorité des flots de travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	altpri
	cancel sched	Annulez les flots de travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	cancel
	deldep sched	Supprimez les dépendances des flots de travaux dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	deldep
	display	Affichez les flots de travaux dans le plan. .	display
	limit sched	Modifiez la limite des travaux s'exécutant simultanément dans un Planificateur de travaux.	limit
	release sched	Libérez les flots de travaux des dépendances dans le plan de production. Non valide pour les postes de travail dans un environnement de bout en bout.	release
	reply	Répondez aux invites des flots de travaux dans le plan de production.	reply
	showschedules	Affichez les informations relatives aux flots de travaux dans le plan de production.	list
	submit sched	Soumettez des flots de travaux dans le plan de production. Si **submit** identifie également un deuxième flot de travaux avec l'argument "ALIAS", l'utilisateur doit détenir les droits d'accès "submit" à cet autre flot de travaux également. Non valide pour les postes de travail dans un environnement de bout en bout.	submit
Utilisation de la fonctionnalité assurance de service de charge de travail	Toutes les activités	Pour qu'un utilisateur puisse réaliser des activités d'assurance de service de charge de travail, l'*utilisateur_TWS* doit posséder les mots clés d'accès suivants :	display, modify, list

Exemple : Pour permettre à un utilisateur de réaliser toutes les actions sur un flot de travail, excepté de le soumettre et de le libérer, spécifiez :

```
schedule          access=adddep,altpri,cancel,deldep,display,limit,reply,list
```

Type d'objet - userobj

Le tableau suivant fournit les mots clés d'accès supplémentaires requis pour utiliser des utilisateurs, autres que ceux décrits dans le tableau 32, à la page 177:

Tableau 45. Utilisateurs - mots clés d'accès supplémentaires

Activité			Mots clés d'accès requis
Composer Dynamic Workload Console	Modélisation des types de travail avec options avancées	Lors de la définition des types de travail avec options avancées, permet au programme de modélisation d'indiquer dans la section des accréditations du travail que les valeurs de `nom utilisateur` et de `mot de passe` requises pour soumettre le travail sont résolues au moment de l'exécution avec celles extraites de la base de données et définies avec les commandes Composer de définition d'utilisateur (`nom utilisateur` et `mot de passe`) ou le panneau Dynamic Workload Console. Notez que sur les agents dynamiques, les définitions d'utilisateur peuvent être utilisées quel que soit le système d'exploitation.	use
Conman Dynamic Workload Console	altpass	Modifiez les mots de passe utilisateur dans le plan.	altpass

Exemple : La définition d'accès ci-dessous permet à un utilisateur d'effectuer les actions suivantes :

- Afficher la liste des informations utilisateur et les modifier, y compris les mots de passe dans la base de données (`display`, `modify` et `altpass`).
- Lorsque vous définissez des types de travail avec options avancées sur des agents dynamiques, indiquez dans la section des accréditations du travail que les valeurs `user name` et `password` requises pour soumettre le travail sont résolues au moment de l'exécution avec celles extraites de la base de données et définies avec la définition d'utilisateur (`use`).

```
userobj          access=display,modify,altpass,use,list
```

Type d'objet - vartable

Le tableau suivant fournit les mots clés d'accès supplémentaires pour utiliser des tables de variables et les variables qu'elles contiennent (cela comprend les variables globales)

Tableau 46. Tables de variables - mots clés d'accès

Activité		Mots clés d'accès requis
Composer Dynamic Workload Console	Utilisez les tables de variables pour les cycles d'exécution, les groupes de cycle d'exécution, les flots de travaux et les postes de travail	use

Exemple : Pour permettre à un utilisateur d'utiliser seulement les tables de variables pour définir d'autres objets de planification, spécifiez :

```
vartable         access=use
```

Type d'objet - application de charge de travail

Le tableau suivant fournit les mots clés d'accès requis pour utiliser des applications de charge de travail :

Tableau 47. Applications de charge de travail - mots clés d'accès

Activité			Mots clés d'accès requis
Dynamic Workload Console	add	Ajouter de nouveaux modèles d'applications de charge de travail dans la base de données. Un accès au déverrouillage est requis pour utiliser l'attribut `;unlock`.	add, unlock
	create	Créer un modèle application de charge de travail dans la base de données. L'accès modify est nécessaire pour utiliser l'attribut `;lock`.	display, modify
	delete	Supprimer un modèle application de charge de travail de la base de données.	delete
	display	Afficher un modèle application de charge de travail.	display
	list	Lister les modèles application de charge de travail que contient la base de données.	display
	lock	Verrouiller les modèles application de charge de travail dans la base de données.	modify
	modify	Modifier les modèles application de charge de travail dans la base de données.	add, modify
	new	Créer un modèle application de charge de travail dans la base de données.	add, modify
	rename	Renommer les modèles application de charge de travail dans la base de données. L'utilisateur doit disposer de l'accès en ajout pour accéder au nouvel objet et d'un accès suppression et affichage pour accéder à l'ancien objet.	add, delete, display
	replace	Remplacer les modèles application de charge de travail dans la base de données. Un accès au déverrouillage est requis pour utiliser l'attribut `;unlock`.	add, modify, unlock
	unlock	Déverrouiller les modèles application de charge de travail dans la base de données.	unlock

Exemple : Pour autoriser un utilisateur à créer et supprimer une application de charge de travail, spécifiez :

```
wkldappl          access=add,delete
```

utilisateur_TWS - Remarques particulières relatives au fichier de sécurité

utilisateur_TWS est un utilisateur particulier dont le fichier de sécurité doit faire l'objet de considérations particulières.

Accès requis pour le *utilisateur_TWS* pour assurance de service de charge de travail Pour qu'un utilisateur puisse réaliser des activités d'assurance de service de charge de travail, les mots clés *display*, *modify* et *list* de *utilisateur_TWS* doivent être attribués aux objets *travail*, *planning* et *UC*.

Nouvel *utilisateur_TWS* dans le fichier de sécurité migré
Si vous modifiez l'*utilisateur_TWS* de votre environnement (dans le cadre d'une mise à niveau parallèle, par exemple), puis que vous migrez le fichier Security (pour conserver vos paramètres), vous devez configurer à l'avance le nouvel *utilisateur_TWS* dans le fichier Security avec tous ses droits d'accès requis, avant de tenter de redémarrer Tivoli Workload Scheduler.

Mettez à jour les définitions pour le domaine Windows *utilisateur_TWS* **dans le fichier de sécurité après avoir effectué la mise à niveau vers la version 9.2**

En raison de la nouvelle prise en charge de l'utilisateur Windows UPN, si certains utilisateurs de domaines Windows sont définis dans les zones logon en tant que `domaine\username`, après avoir effectué une mise à niveau vers la version 9.2, mettez à jour le fichier `Security` avant de démarrer l'instance Tivoli Workload Scheduler. Insérez le caractère d'échappement '\' avant le caractère '\' dans la valeur `domaine\username`. Par exemple, si vous utilisez la valeur `MYDOMAIN\user1` dans la zone logon, après avoir effectué la mise à niveau, dans le fichier `Security` vous devez mettre à jour la ligne de la manière suivante :

```
..............
logon=MYDOMAIN\user1
..............
```

Exemple de fichier de sécurité

Cette section contient un modèle de fichier de sécurité divisé en sections correspondant à chaque classe d'utilisateur distincte.

Notez que les définitions sont classées de la plus spécifique à la moins spécifique. Compte tenu de cet ordre, les utilisateurs *TWS_users* et **root** sont d'abord associés, suivis des utilisateurs du groupe **sys** puis des utilisateurs du groupe **mis**. Quant aux autres utilisateurs, ils sont associés à la dernière définition, qui représente la moins spécifique.

TWS_users et utilisateurs root connectés au gestionnaire de domaine maître

user mastersm cpu=$master + logon=*utilisateur_TWS*,root

```
###########################################################
#     Sample Security File
###########################################################
# APPLIES TO TWS_users AND ROOT USERS LOGGED IN ON THE
# MASTER DOMAIN MANAGER.
user mastersm  cpu=$master + logon=utilisateur_TWS,root
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------      ---------------------
job           cpu=@             access=@
schedule                       access=@
resource                       access=@
prompt                         access=@
file                           access=@
calendar                       access=@
cpu           cpu=@            access=@
parameter     name=@ ~ name=r@ access=@
userobj    cpu=@ + logon=@     access=@
eventrule       name=@           access=add,delete,display,modify,list,unlock
action     provider=@           access=display,submit,use,list
event      provider=@           access=use
report     name=@               access=display
runcygrp      name=@           access=add,delete,display,modify,use,list,unlock
vartable   name=a@,$default      access=add,delete,display,modify,use,list,unlock
wkldappl      name=@            access=add,delete,display,modify,list,unlock
end
```

Cette définition d'utilisateur régit l'accès à l'interface graphique et à l'interface de ligne de commande pour les utilisateurs *TWS_users* et **root** connectés à un gestionnaire de domaine maître. Ces derniers disposent d'un accès illimité à tous

les objets, à l'exception des paramètres dont le nom commence par **r**. Seuls les utilisateurs du groupe **mis** ont accès aux paramètres **r**. Ce sont les seuls utilisateurs qui peuvent générer toutes les sortes de plans et qui peuvent créer, mettre à jour et supprimer les définitions de règles d'événement.

Tous les utilisateurs ont accès à toutes les tables de variables commençant par "a" et à la table par défaut, quel que soit le nom de table de variables par défaut.

TWS_users et utilisateurs root connectés à tout gestionnaire de domaine (autre que le maître)

user testerlondon cpu=$manager + logon=*utilisateur_TWS*,root

```
###########################################################
#     Sample Security File
###########################################################
# APPLIES TO TWS_users AND ROOT USERS LOGGED IN ON ANY
# DOMAIN MANAGER.
user testerlondon  cpu=$manager + logon=utilisateur_TWS,root
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------    ----------------------
job           cpu=@           access=add,delete,display
schedule                      access=add,delete,display
resource                       access=@
prompt                         access=@
file          name=prodsked   access=build, display
file          name=trialsked   access=build,display
calendar                       access=@
cpu           cpu=@            access=@
parameter  name=@ ~ name=v@   access=@
userobj    cpu=@ + logon=@     access=@
eventrule       name=@           access=add,delete,display,modify,list,unlock
action     provider=@            access=display,submit,use,list
event      provider=@            access=use
report     name=@                access=display
runcygrp      name=@            access=add,delete,display,modify,use,list,unlock
vartable   name=a@,$default      access=add,delete,display,modify,use,list,unlock
wkldappl      name=@            access=add,delete,display,modify,list,unlock
end
```

Cette définition d'utilisateur régit l'accès à l'interface graphique et à l'interface de ligne de commande pour les utilisateurs *TWS_users* et **root** connectés à un gestionnaire de domaine autre que le gestionnaire de domaine maître. Ces derniers disposent d'un accès illimité à tous les objets, à l'exception des paramètres dont le nom commence par **v**, et des travaux et flot de travaux pour lesquels ils détiennent un accès limité. Ils peuvent générer tous les types de plans, ainsi que créer, mettre à jour et supprimer des définitions de règles d'événement.

Tous les utilisateurs ont accès à toutes les tables de variables commençant par "a" et à la table par défaut, quel que soit le nom de table de variables par défaut.

TWS_users et utilisateurs root connectés à tout poste de travail autre que tout gestionnaire de domaine

user sm ~CPU=$MANAGER logon=*utilisateur_TWS*,root

```
#########################################################
# APPLIES TO TWS_users AND ROOT USERS LOGGED IN ON ANY
# WORKSTATION OTHER THAN THE MASTER DOMAIN MANAGER.
user sm  logon=utilisateur_TWS,root
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
```

```
# ----------  ------------     ----------------------
job           cpu=$thiscpu     access=@
schedule      cpu=$thiscpu     access=@
resource      cpu=$thiscpu     access=@
prompt                         access=@
calendar                       access=@
cpu           cpu=$thiscpu     access=@
parameter     cpu=$thiscpu
              ~ name=r@        access=@
action     provider=@            access=display,submit,use,list
event      provider=@            access=use
report        name=RUNHIST,RUNSTATS  access=display
runcygrp      name=@            access=add,delete,display,modify,use,list,unlock
file          name=globalopts  access=display
end
```

Cette définition d'utilisateur s'applique aux utilisateurs *TWS_users* et **root** auxquels la définition (1) ne s'applique pas, qui sont ceux qui sont connectés sur tout poste de travail autre que le gestionnaire de domaine maître ou un autre gestionnaire de domaine. Ils disposent d'un accès illimité à tous les objets résidant sur leur poste de travail de connexion. Comme les invites, les fichiers et les agendas sont globaux par nature, ils ne sont pas associés à un poste de travail.

Ils peuvent utiliser les règles d'événement mais ne sont pas autorisés à créer, mettre à jour ou supprimer les définitions de règles d'événement.

Utilisateurs connectés au groupe *sys* sur le gestionnaire de domaine maître

user masterop cpu=$master + group=sys

```
############################################################
# APPLIES TO USERS LOGGED INTO THE SYS GROUP ON THE
# MASTER DOMAIN MANAGER.
user masterop  cpu=$master + group=sys
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------     ----------------------
job           cpu=@
              + logon="TWS_domain\utilisateur_TWS" access=@
job           cpu=@
              + logon=root     access=adddep,altpri,cancel,
                                      confirm,deldep,release,
                                      reply,rerun,submit,use
job           cpu=@
              + logon=@
              ~ logon=root     access=add,adddep,altpri,
                                      cancel,confirm,
                                      deldep,release,reply,
                                      rerun,submit,use
schedule      cpu=$thiscpu     access=@
schedule      cpu=@            access=adddep,altpri,cancel,
                                      deldep,limit,release,
                                      submit
resource                       access=add,display,
                                      resource,use
file          name=globalopts  access=display
file          name=prodsked    access=display
file          name=symphony    access=display
file          name=trialsked   access=build,display
calendar                       access=display,use
cpu           cpu=@            access=@
```

```
parameter     name=@ ~ name=r@ access=@
report        name=RUNHIST,RUNSTATS  access=display
wkldappl      name=@            access=add,delete,display,modify,list,unlock
end
```

Cette définition d'utilisateur concerne les utilisateurs connectés au groupe **sys** sur le gestionnaire de domaine maître. Ils disposent d'un groupe unique de privilèges d'accès. Vous pouvez utiliser plusieurs instructions objet pour octroyer aux utilisateurs un type d'accès particulier selon le groupe d'objet. Par exemple, le programme propose trois instructions :

- La première instruction confère un accès illimité aux travaux qui sont exécutés sur un poste de travail quelconque (@) sous le nom de l'utilisateur (*domaine_TWS\utilisateur_TWS*).
- La deuxième instruction confère un type d'accès particulier aux travaux qui sont exécutés sur un poste de travail quelconque en tant que **root**.
- La troisième instruction confère un type d'accès particulier aux travaux qui sont exécutés sur un poste de travail quelconque. Le programme exclut les travaux exécutés par un utilisateur root.

Ils s'agit des seuls utilisateurs définis sur le gestionnaire de domaine maître, autres que maestro ou root, qui peuvent générer des plans d'essai et prévisionnels.

Utilisateurs connectés au groupe *sys* sur tout poste de travail autre que le gestionnaire de domaine maître

user op ~cpu=$master group=sys

```
############################################################
# APPLIES TO USERS LOGGED INTO THE SYS GROUP ON ANY
# WORKSTATION OTHER THAN THE MASTER DOMAIN MANAGER
user op  group=sys
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------     ----------------------
job           cpu=$thiscpu
              + logon=@        access=@
job           cpu=$thiscpu
              + logon=root     access=adddep,altpri,cancel,
                                      confirm,deldep,release,
                                      reply,rerun,submit,use
job           cpu=$thiscpu
              ~ logon=root     access=adddep,altpri,cancel,
                                      confirm,deldep,release,
                                      reply,rerun,submit,use
schedule      cpu=$thiscpu     access=@
resource                       access=add,display,resource,use
runcygrp      name=@            access=add,delete,display,modify,use,list,unlock
prompt                         access=add,display,reply,use
calendar                       access=use
cpu           cpu=$thiscpu     access=console,fence,limit,
                                      link,start,stop,unlink
parameter     name=@ ~ name=r@ access=@

wkldappl      name=@            access=add,delete,display,modify,list,unlock
end

############################################################
```

Cette définition d'utilisateur s'applique aux utilisateurs du groupe **sys** auxquels la définition (3) ne s'applique pas ; il s'agit de ceux qui sont connectés à tout poste de travail autre que le gestionnaire de domaine maître. Ils disposent d'un ensemble de

privilèges d'accès similaires à ceux de la définition (3), sauf que l'accès est limité aux objets résidant sur le poste de travail de connexion (**$thiscpu**) de l'utilisateur.

Utilisateurs connectés au groupe *mis* sur tout poste de travail

user misusers group=mis

```
############################################################
# APPLIES TO USERS LOGGED INTO THE MIS GROUP ON
# ANY WORKSTATION.
user misusers  cpu=@           group=mis
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------     ----------------------
job           cpu=$thiscpu
              + logon=@        access=@
job           cpu=$thiscpu
              + logon=@
              ~ logon=root     access=submit,use
schedule      cpu=$thiscpu     access=add,submit,
                                      modify,display
cpu           cpu=@ + type=agent,s-agent,fta
                               access=console,fence,limit,
                                      link,start,stop,unlink
parameter     name=r@          access=@
parameter     name=@           access=display
runcygrp      name=@            access=add,delete,display,modify,use,list,unlock
end
############################################################
```

Cette définition d'utilisateur concerne les utilisateurs connectés au groupe **mis** sur n'importe quel poste de travail. Ils disposent d'un ensemble limité de privilèges d'accès aux agents tolérants aux pannes, aux agents standard et aux agents dynamiques. Le programme omet les ressources, les invites, les fichiers, les agendas et les postes de travail, pour interdire tout accès à ces objets. Les utilisateurs disposent d'un accès illimité aux paramètres dont le nom commence par **r**, mais peuvent seulement afficher les autres paramètres.

Utilisateurs connectés à plusieurs groupes [mot clé continue]

Il s'agit d'un exemple de fichier de sécurité dans lequel le mot clé `continue` est utilisé. Avec ce type de fichier de sécurité, les utilisateurs obtiennent tous les accès définis pour chaque groupe auquel ils appartiennent. En conséquence, un utilisateur peut obtenir des autorisations à partir de plusieurs définitions d'utilisateur.

user misusers cpu@ group=mis

```
############################################################
# User misusers USER DEFINITION APPLIES TO USERS LOGGED IN TO
# THE MIS GROUP ON ANY WORKSTATION.
#
# User dbusers USER DEFINITION APPLIES TO USERS LOGGED IN TO
# THE DB GROUP ON ANY WORKSTATION.
#
# User default USER DEFINITION APPLIES TO ALL USERS.
#

user misusers  cpu=@           group=mis
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------     ----------------------
job           cpu=@ + name=mis@
                               access=@
schedule      name=mis@        access=@
```

```
parameter     name=mis@        access=@
continue

user dbusers  cpu=@            group=db
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------     ----------------------
job           cpu=@ + name=db_@
                               access=@
schedule      name=db_@        access=@
parameter     name=db_@        access=@
continue

user default cpu=@ + logon=@
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------     ----------------------
parameter     name=@           access=display
end


##############################################################
```

Les utilisateurs qui appartiennent seulement au groupe *mis* peuvent accéder à tous les objets dont le nom commence par le préfixe *mis*, comme spécifié dans la définition d'utilisateur `user misusers`. En outre, la définition d'utilisateur `user default` leur offre un accès à l'affichage de tous les paramètres.

Les utilisateurs qui appartiennent seulement au groupe *db* peuvent accéder à tous les objets dont le nom commence par le préfixe *db_*, comme spécifié dans la définition d'utilisateur `user dbusers`. En outre, la définition d'utilisateur `user default` leur offre un accès à l'affichage de tous les paramètres.

Les utilisateurs appartenant à la fois aux groupes *mis* et *db* peuvent accéder aux objets dont le nom débute par le préfixe *mis* et aux objets dont le nom commence par le préfixe *db_*, comme spécifié dans les définitions d'utilisateur `user misusers` et `user dbusers`. En outre, la définition d'utilisateur `user default` leur offre un accès à l'affichage de tous les paramètres.

Vous devez classer les définitions de la plus spécifique à la moins spécifique. La définition d'utilisateur `user default` fournit des accès génériques et doit par conséquent être spécifiée à la fin du fichier.

Tous les autres utilisateurs connectés sur n'importe quel poste de travail

user default cpu=@ + logon=@

```
##############################################################
# APPLIES TO ALL OTHER USERS LOGGED IN ON ANY
# WORKSTATION.
user default  cpu=@ + logon=@
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------     ----------------------
job           cpu=@             access=@
schedule                       access=@
resource                       access=@
prompt                         access=@
file                           access=@
calendar                       access=@
cpu           cpu=@            access=@
parameter     name=@ ~ name=r@ access=@
```

```
userobj    cpu=@ + logon=@     access=@
eventrule       name=@           access=add,delete,display,modify,list,unlock
action     provider=@            access=display,submit,use,list
event      provider=@            access=use
report     name=@                access=display
runcygrp      name=@            access=add,delete,display,modify,use,list,unlock
vartable   name=a@,$default      access=add,delete,display,modify,use,list,unlock
wkldappl      name=@            access=add,delete,display,modify,list,unlockend
###########################################################
```

Ils disposent d'un accès illimité à tous les objets, à l'exception des paramètres dont le nom commence par **r**. Ce sont les seuls utilisateurs qui peuvent générer toutes les sortes de plans et qui peuvent créer, mettre à jour et supprimer les définitions de règles d'événement. Tous les utilisateurs ont accès à toutes les tables de variables commençant par "a" et à la table par défaut, quel que soit le nom de table de variables par défaut.

Tous les utilisateurs Windows domain1.com connectés sur n'importe quel poste de travail

user cpu=@ + logon =@\@domain1.com

```
###########################################################
# APPLIES TO ALL OTHER USERS LOGGED IN ON ANY
# WORKSTATION.
user default  cpu=@ + logon=@\@domain1.com
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------     ----------------------
job           cpu=@ + logon =a@\@domain1.com  access=display
job           cpu=@             access=@
schedule                       access=@
resource                       access=@
prompt                         access=@
file                           access=@
calendar                       access=@
cpu           cpu=@            access=@
parameter     name=@ ~ name=r@ access=@
userobj    cpu=@ + logon=@     access=@
eventrule       name=@           access=add,delete,display,modify,list,unlock
action     provider=@            access=display,submit,use,list
event      provider=@            access=use
report     name=@                access=display
runcygrp      name=@            access=add,delete,display,modify,use,list,unlock
vartable      name=g@,$default  access=add,delete,display,modify,use,list,unlock
wkldappl      name=@            access=add,delete,display,modify,list,unlock
end
###########################################################
```

Les utilisateurs Windows de `domain1.com` dont le nom commence par un 'a' peuvent afficher uniquement des travaux et peuvent gérer des paramètres dont le nom ne commence pas par **r**. Tous les autres utilisateurs Windows de `domain1.com` connectés à n'importe quel poste de travail disposent d'un accès illimité à tous les objets, à l'exception des paramètres dont le nom commence par **r**. Il s'agit des seuls utilisateurs qui peuvent générer toutes les sortes de plans et qui peuvent créer, mettre à jour et supprimer les définitions de règles d'événement. Tous les utilisateurs ont accès à toutes les tables de variables commençant par "g" et à la table par défaut, quel que soit le nom de table de variables par défaut.

Tous les utilisateurs Windows MYWINDOM connectés sur n'importe quel poste de travail

user default cpu=@ + logon=MYWINDOM\\@

```
#########################################################
# S'APPLIQUE A TOUS LES UTILISATEURS WINDOWS "MYWINDOM" CONNECTES SUR N'IMPORTE QUEL
#      POSTE DE TRAVAIL.
user default  cpu=@ +  logon=MYWINDOM\\@
begin
#  OBJECT     ATTRIBUTES       ACCESS CAPABILITIES
# ----------  ------------     ---------------------
job           cpu=@             access=@
schedule                       access=@
resource                       access=@
prompt                         access=@
file                           access=@
calendar                       access=@
cpu           cpu=@            access=@
parameter     name=@            access=@
userjob       cpu=@ + logon =MYWINDOM\\r@  access=display
userobj    cpu=@ + logon=@     access=@
eventrule       name=@           access=add,delete,display,modify,list,unlock
action     provider=@            access=display,submit,use,list
event      provider=@            access=use
report     name=@                access=display
runcygrp      name=@            access=add,delete,display,modify,use,list,unlock
vartable      name=g@,$default  access=add,delete,display,modify,use,list,unlock
wkldappl      name=@            access=add,delete,display,modify,list,unlock
end
#########################################################################################################
```

Utilisateurs Windows dans MYWINDOM dont le nom commence par un 'r' ne peuvent afficher que des travaux d'utilisateurs. Tous les autres utilisateurs Windows MYWINDOM connectés à n'importe quel poste de travail se voient accorder un accès non restreint à tous les objets; Ce sont les seuls utilisateurs qui peuvent générer toutes les sortes de plans et qui peuvent créer, mettre à jour et supprimer les définitions de règles d'événement. Tous les utilisateurs ont accès à toutes les tables de variables commençant par "g" et à la table par défaut, quel que soit le nom de table de variables par défaut.

Remarque : A partir de la version 9.2, en raison de la prise en charge des utilisateurs Windows dans le format UPN (User Principal Name), vous devez spécifier les utilisateurs du domaine Windows d'une manière différente dans le fichier Security. Dans le même exemple pour la version précédente, vous obtenez la syntaxe suivante :

```
user default  cpu=@ +  logon=MYWINDOM\@
.....................................
userjob       cpu=@ + logon =MYWINDOM\r@  access=display
```

Chapitre 5. Configuration de l'authentification

La présente section explique comment configurer l'authentification, en utilisant, entre autres, le protocole LDAP (Lightweight Directory Access Protocol) largement utilisé. Elle comprend les rubriques principales ci-après :

Emplacements de configuration de l'authentification

L'authentification doit être configurée pour chaque profil WebSphere Application Server ; suivez le règles suivantes :

Pour l'authentification des utilisateurs de ligne de commande
Pour les utilisateurs de la ligne de commande, du client de ligne de commande et des clients de ligne de commande connectés au gestionnaire de domaine maître à l'aide de HTTP ou HTTPS, la même méthode d'authentification doit être configurée pour les composants suivants :

- Gestionnaire de domaine maître
- Gestionnaire de domaine maître de sauvegarde

Pour l'authentification de Dynamic Workload Console
La console Dynamic Workload Console n'est pas installée sur la même instance que le gestionnaire de domaine maître, l'authentification doit être configurée séparément pour Dynamic Workload Console et le gestionnaire de domaine maître.

Pour l'authentification des utilisateurs du connecteur z/OS
Le connecteur z/OS est toujours installé sur la même instance que la console Dynamic Workload Console. Vous n'avez pas besoin de configurer l'authentification de manière distincte pour celui-ci.

Pour l'authentification des utilisateurs du gestionnaire de domaine dynamique
La même méthode d'authentification doit être configurée pour chaque gestionnaire de domaine dynamique et les gestionnaire de domaine dynamique de secours correspondants. Cette méthode d'authentification ne doit pas être identique à celle utilisée pour le gestionnaire de domaine maître.

Configurations disponibles

Lors de l'installation, tous les composants de Tivoli Workload Scheduler utilisant la WebSphere Application Server sont configurés pour l'authentification en mode VMM (Virtual Member Manager). Cette méthode crée un *Registre d'utilisateurs fédéré*, dans lequel vous pouvez choisir un ou plusieurs systèmes d'authentification parmi les suivants :

- Système d'exploitation local - le système d'authentification par défaut à l'installation sur les systèmes d'exploitation Windows
- Module d'authentification enfichable (PAM, Pluggable Authentication Module) - le système d'authentification par défaut à l'installation sur les systèmes d'exploitation UNIX, Linux et AIX. `Custom` est le système d'authentification par défaut à l'installation sur les systèmes d'exploitation UNIX et Linux, et sur les systèmes d'exploitation AIX, où il se nomme `CustomPAM`, pour le différencier de l'autre système d'authentification pris en charge sous AIX nommé CustomLAM.
- Module d'authentification chargeable (LAM, Loadable Authentication Module) - `CustomLAM` est un mécanisme d'authentification et d'identification réservé aux systèmes d'exploitation AIX.
- LDAP
- Registre de fichiers

Remarque : Sous AIX, le système d'authentification CustomPAM et les systèmes d'authentification CustomLAM et LocalOS s'excluent mutuellement.

Si vous souhaitez utiliser le système d'exploitation local en tant que méthode d'authentification pour Dynamic Workload Console sur les systèmes d'exploitation UNIX, exécutez les opérations décrites dans «Configuration de Dynamic Workload Console pour utiliser la méthode d'authentification du système d'exploitation local ou PAM», à la page 105.

Si vous choisissez d'activer LDAP, vous pouvez utiliser l'un des serveurs suivants, pour lesquels des modèles de configuration sont fournis dans cette documentation :

- IBM Tivoli Directory Server
- Sun Java Director Server
- Microsoft Windows Active Directory
- z/OS Integrated Security Services LDAP Server

Méthodes de configuration de l'authentification

LDAP peut être configuré selon l'une des méthodes suivantes :

Utilisation du WebSphere Administrative Console
Pour chaque profil WebSphere Application Server sur lequel vous souhaitez modifier la configuration d'authentification par défaut, ouvrez la console WebSphere Administrative Console et optez pour la configuration de la sécurité globale. Vous choisissez et configurez le ou les mécanismes d'authentification que vous utilisez dans votre environnement.

Pour une description exhaustive, voir «Configuration de l'authentification à l'aide de WebSphere Administrative Console», à la page 202.

En mode manuel, à l'aide des outils de WebSphere Application Server fournis avec le produit
Pour chaque profil WebSphere Application Server sur lequel vous voulez

modifier la configuration d'authentification par défaut, exécutez un script appelé showSecurityProperties pour créer un modèle contenant la configuration de la sécurité en cours. Vous modifiez ce modèle en ajoutant et complétant les propriétés qui définissent le ou les mécanismes d'authentification que vous utilisez dans votre environnement. Pour finir, vous exécutez un script appelé changeSecurityProperties pour mettre à jour la configuration de sécurité de WebSphere Application Server.

Pour une description exhaustive, voir «Configuration de l'authentification à l'aide des outils WebSphere Application Server», à la page 206.

Scénario de configuration standard

Dans un environnement complexe, vous pouvez envisager d'utiliser le scénario suivant pour configurer le mécanisme d'authentification de votre choix sur votre environnement de planification de charge de travail :

1. Utilisez le script changeSecurityProperties pour configurer le mécanisme sur une instance de WebSphere Application Server, par exemple celle installée avec le gestionnaire de domaine maître.
2. Testez votre connexion à l'aide de l'authentification configurée avec plusieurs ID d'utilisateur.
3. Sur cette instance, exécutez le script showSecurityProperties et enregistrez la sortie pour créer un fichier modèle de texte contenant la configuration. showSecurityProperties extrait *uniquement* les configurations qui ont été créées à l'aide des changeSecurityProperties.
4. Sur chaque système sur lequel vous voulez configurer l'authentification
   - Copiez le fichier modèle texte créé à l'étape précédente.
   - Exécutez showSecurityProperties et enregistrez le fichier de sortie.
   - Fusionnez ce fichier de sortie avec le fichier modèle de configuration.
   - Exécutez changeSecurityProperties pour mettre à jour la configuration WebSphere Application Server.
   - Testez votre connexion à l'aide de l'authentification configurée avec plusieurs ID d'utilisateur.

Règles d'utilisation d'un Registre d'utilisateurs fédéré avec Tivoli Workload Scheduler

Cette section fournit les règles simples que vous devez suivre lors de la configuration de Tivoli Workload Scheduler pour utiliser un Registre d'utilisateurs fédéré:

Aucun ID utilisateur en double
Vous pouvez définir autant de registres d'utilisateurs que vous voulez dans un Registre d'utilisateurs fédéré. Cependant, aucun ID utilisateur ne doit figurer dans plusieurs registres à la fois (cela empêche l'utilisation d'un système d'exploitation local et du module PAM dans un mécanisme d'authentification commun) et aucun ID utilisateur ne doit être présent en double dans un même registre. Ainsi, vous configurez plusieurs registres d'utilisateurs lorsque vous avez des utilisateurs dans différents groupes non inclusifs utilisant différents registres d'utilisateurs et nécessitant l'accès à Tivoli Workload Scheduler.

ID de registre réservés
Les outils de WebSphere Application Server utilisent des ID spécifiques pour reconnaître les registres. Ces ID constituent ainsi des mots clés

réservés que vous ne pouvez pas utiliser pour créer vos propres registres, et ce, quelle que soit la méthode utilisée pour les configurer :

twaLocalOS
Identifie l'adaptateur de pont du registre d'utilisateurs personnalisé, configuré pour les utilisateurs du système d'exploitation local

twaPAM
Identifie l'adaptateur de pont du registre d'utilisateurs personnalisé, configuré pour utiliser le module d'authentification enfichable avec Tivoli Workload Scheduler – il n'est pas disponible sur les systèmes d'exploitation Windows.

twaLDAP
Identifie le pont du registre d'utilisateurs, configuré pour les utilisateurs LDAP

defaultWIMFileBasedRealm
Identifie le registre de fichiers par défaut de la WebSphere Application Server.

Compatibilité
Pour assurer la compatibilité avec les noeuds de votre réseau Tivoli Workload Scheduler d'une version précédente, seuls les composants de Tivoli Workload Scheduler et Dynamic Workload Console de version 8.4 et supérieure peuvent être configurés avec une connexion LDAP.

Configuration de l'authentification à l'aide de WebSphere Administrative Console

WebSphere Administrative Console est l'interface utilisateur d'administration de Dynamic Workload Console et est installé automatiquement avec la WebSphere Application Server.

Si vous prévoyez de configurer Dynamic Workload Console version 8.6 dans Connexion unique et Tivoli Workload Scheduler antérieur à la version 8.6, vous devez utiliser cette interface pour configurer l'authentification ; vous ne pouvez pas utiliser les outils was.

Remarque : Si vous créez le référentiel à l'aide de WebSphere Administrative Console, l'outil was `showSecurityProperties` est susceptible de ne pas afficher les données pour le référentiel.

Utilisez WebSphere Administrative Console pour configurer l'authentification comme suit :

1. **Sauvegardez la configuration**
   Sauvegardez la configuration de WebSphere Application Server avec la commande **backupConfig**.
2. **Accédez à la console WebSphere Administrative Console**
   Pour accéder à WebSphere Administrative Console, utilisez une des adresses URL suivantes avec les données d'identification d'administration WebSphere Application Server :
   `https://<Hostname>:<adminSecurePort>/ibm/console/`

   `http://<Hostname>:<adminPort>/ibm/console/`

Où :

Nom d'hôte
Nom qualifié complet ou adresse IP de l'ordinateur.

adminSecurePort
Si vous vous connectez avec HTTPS, indiquez le port sécurisé d'administration de WebSphere Application Server dont la valeur par défaut est 31124.

adminPort
Si vous vous connectez avec HTTP, indiquez le port d'administration de WebSphere Application Server dont la valeur par défaut est 31123.

Exemple

```
https://mypc:31124/ibm/console/
```

3. **Connectez-vous à la console**

   Connectez-vous à la console à l'aide des données d'identification de WebSphere Application Server. Vous les avez fournies lorsque vous avez installé le composant sur ce système (elles ont pu changer depuis).

4. **Accédez à la section de sécurité**

   Sélectionnez **Sécurité ▸ Sécurité globale**

5. **Configurez les mécanismes d'authentification requis.**

   Dans la section du référentiel des comptes utilisateur, vous voyez que l'option par défaut **Federated repositories** est sélectionnée. Cliquez sur le bouton **Configure** en regard. Utilisez la console WebSphere Administrative Console pour configurer le ou les mécanismes d'authentification. Lors de la modification des lignes dans la table **Repositories in the realm**, la valeur de **InternalFileRepository** correspondant à la colonne Repository Identifier ne doit pas être supprimée.

   Par exemple, cliquez sur **Add Base entry to Realm ...** > **Add Repository...** pour ajouter un nouveau référentiel, tel que LDAP.

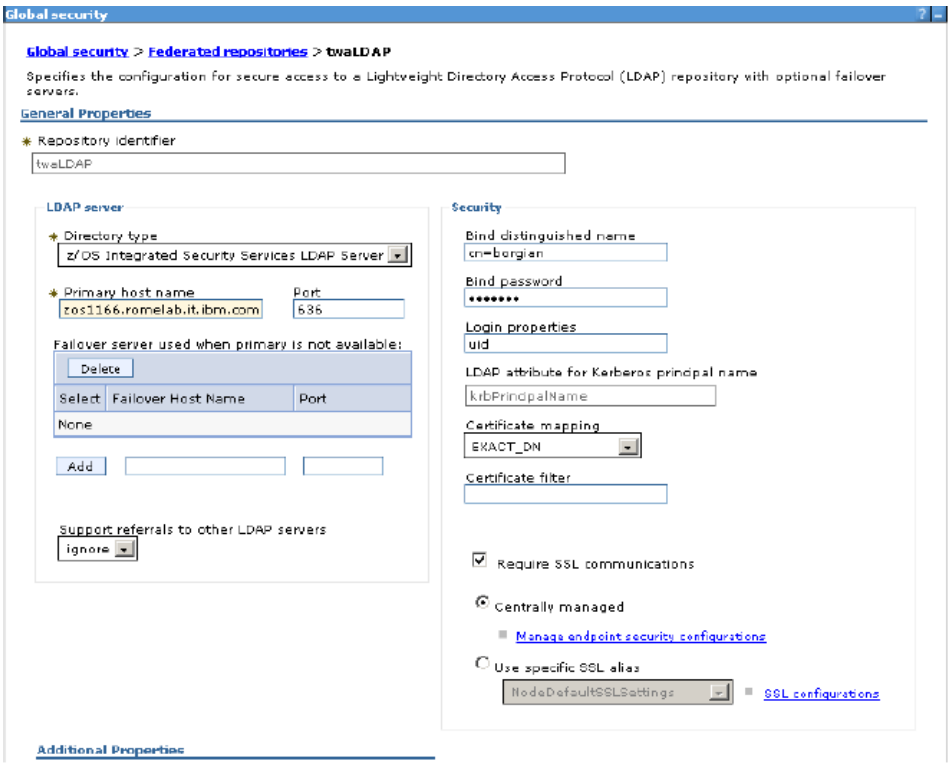
   Remarque : Ne supprimez pas l'entrée `twaPAM` du référentiel avant d'avoir terminé la procédure de configuration.

   Utilisez l'aide contextuelle intégrée pour connaître les informations à fournir dans chaque zone.

   De plus, la sortie de toutes les paires clé/valeur de l'outil **`showSecurityProperties`** sont documentées dans la section «Propriétés de sécurité : référence», à la page 206. Chaque paire clé/valeur correspond à une zone ou à un concept exprimé dans l'interface graphique de la console WebSphere Administrative Console ; les clés sont des mnémoniques pour vous aider à effectuer la correspondance.

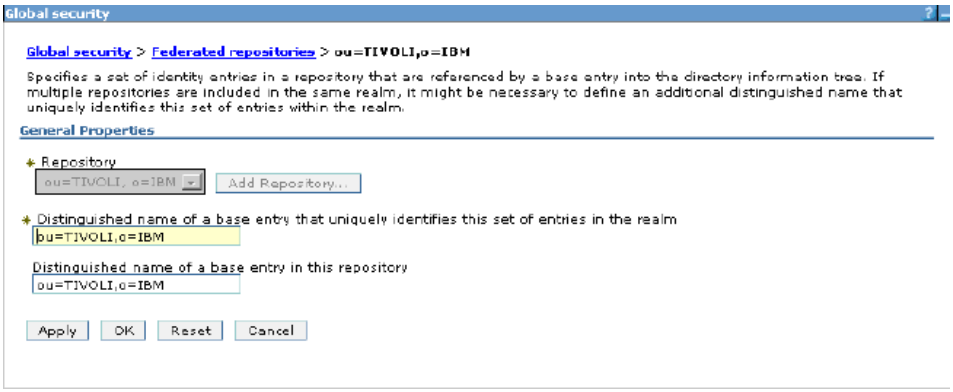
   Remarque : Si vous prévoyez de configurer Dynamic Workload Console version 9.1 dans la Connexion unique avec une version de Tivoli Workload Scheduler antérieure à la version 9.1, dans la fenêtre Global Security, indiquez la même valeur dans les zones **Distinguished name of a base entry...**.

   Le panneau ci-dessous donne un exemple de configuration avec z/OS Integrate Security Service LDAP Server



6. **Sauvegardez la configuration modifiée**

   Cliquez sur **Save** pour sauvegarder la nouvelle configuration.
7. Ensuite, vous pouvez modifier les types d'entité LDAP pour ce référentiel. Dans la section **Additional Properties**, sélectionnez **Supported entity types** > **Group**
8. Dans la zone Entity type, entrez le nom distinctif d'une entrée de base dans le référentiel. Cette entrée détermine l'emplacement par défaut dans le référentiel dans lequel sont placées les entités du type spécifié lors des opérations d'écriture par la gestion des utilisateurs et des groupes.
9. Cliquez sur **Apply** > **Save** pour enregistrer les modifications apportées et revenir dans le panneau précédent.
10. Vous pouvez spécifier les propriétés de **nom distinctif relatif** en entrant les propriétés de nom distinctif relatif (RDN™) pour le type d'entité spécifié. Les valeurs possibles sont `cn` pour **Group**, `uid` ou `cn` pour **PersonAccount** et `o, ou, dc` et `cn` pour **OrgContainer**. Séparez les propriétés de l'entité **OrgContainer** par un point-virgule (;).
11. Cliquez sur **OK** > **Save** our sauvegarder les modifications. Au moment de quitter, vous êtes invité à définir l'entrée de base de ce référentiel, le premier nom, qui est obligatoire, est un nom de votre choix qui identifie de manière unique le référentiel dans la fédération. Le second nom est facultatif et dépend de la configuration du serveur LDAP. Voir l'exemple de panneau de



ci-dessous.

12. **Redémarrez le serveur**

    Arrêtez le serveur d'applications à l'aide de la commande **stopappserver**, comme décrit dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence*. Pour arrêter le serveur, utilisez les données d'identification d'administration WebSphere d'origine.

    Redémarrez le serveur à l'aide de la commande **startappserver**, comme décrit dans le *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

13. Cette étape s'applique uniquement à Dynamic Workload Console. Connectez-vous à Dynamic Workload Console :

    ```
    http://système_console_charge_travail_dynamique:http_port/racine_contexte_DASH
    https://système_console_charge_travail_dynamique:https_port/racine_contexte_DASH
    ```

    où,

    racine_contexte_DASH
    : Il s'agit de la racine de contexte Dashboard Application Services Hub définie lors de l'installation. La racine de contexte détermine l'URL d'une application déployée et, par défaut, est identique au répertoire d'application ou à la structure d'archive. Dans ce cas, la valeur par défaut est `ibm/console`.

    Utilisez les données d'identification WebSphere Application Server (administrateur *d'origine*) et affectez les rôles suivants au nom d'administrateur principal (*nouvel* administrateur).

    - Iscadmins
    - TDWBAdministrator
    - TWSWEBUIAdministrator
    - chartAdministrator

    Pour plus d'informations, voir «Configuration des rôles pour accéder à Dynamic Workload Console», à la page 106.

Vous pouvez maintenant vous connecter à Dynamic Workload Console en tant que *nouvel* administrateur et, facultativement, supprimer l'entrée `twaPAM` du référentiel si vous n'en avez plus besoin.

Configuration de l'authentification à l'aide des outils WebSphere Application Server

Lorsque vous installez un composant de Tivoli Workload Scheduler qui utilise WebSphere Application Server, vous installez également un ensemble d'outils de WebSphere Application Server (également appelé *wastools*). Pour plus d'informations d'ordre général sur ces outils, voir «Utilitaires du serveur d'applications», à la page 424.

Vous pouvez utiliser les outils du serveur d'applications pour configurer seulement les serveurs LDAP suivants :

- Microsoft Active Directory
- Oracle Java System Directory Server
- IBM Tivoli Directory Server
- z/OS Integrated Security Services LDAP Server

Pour les autres serveurs LDAP, suivez la procédure décrite dans «Configuration de l'authentification à l'aide de WebSphere Administrative Console», à la page 202.

Pour plus d'informations sur le schéma de serveur LDAP, voir «Schéma de serveur LDAP», à la page 223.

Pour configurer l'authentification, procédez comme suit :

1. Connectez-vous à Dynamic Workload Console avec les données d'identification d'administration de WebSphere Application Server actuelles.
2. Sauvegardez la configuration de WebSphere Application Server avec la commande **backupConfig**.
3. Pour le serveur LDAP z/OS Integrated Security Services, importez le fichier `cert.arm` dans WebSphere Application Server en exécutant l'outil Java IkeyMan dans `/opt/IBM/<répertoire_installation_JazzSM>/profile/bin/keyman.sh` (UNIX) ou `C:\Program Files\IBM\<répertoire_installation_JazzSM>\profile\bin\ikeyman.bat` (Windows). Les chemins d'accès font référence à l'emplacement par défaut.
4. Exportez vos propriétés de sécurité actuelles vers un fichier texte à l'aide de la commande **showSecurityProperties <fichier_texte>**.
5. Personnalisez les propriétés de sécurité en modifiant *<fichier_texte>*. Voir «Propriétés de sécurité : référence».
6. Arrêtez le serveur à l'aide de la commande **stopappserver**, comme décrit dans le document *Tivoli Workload Scheduler - Guide d'utilisation et de référence*. Pour arrêter le serveur, utilisez les données d'identification d'administration WebSphere d'origine.
7. Chargez les nouvelles propriétés à l'aide de la commande **\<chemin_complet>\changeSecurityProperties <fichier_texte>**.
8. Redémarrez le serveur à l'aide de la commande **startappserver**, comme décrit dans le document*Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Pour des exemples de configurations de serveurs LDAP, voir «Exemples de configuration des serveurs LDAP», à la page 220.

Propriétés de sécurité : référence

Cette section présente les propriétés de sécurité importantes figurant dans le fichier généré par le script showSecurityProperties. Elle comporte plusieurs *panneaux* :

Panneau Sécurité globale

Panneau obligatoire.

```
############################################################
Global Security Panel
############################################################
enabled=true
enforceJava2Security=false
useDomainQualifiedUserNames=false
cacheTimeout=600
ltpaTimeOut=720
issuePermissionWarning=true
activeProtocol=CSI
useFIPS=false
activeAuthMechanism=LTPA
activeUserRegistry=LDAP LocalOS WIM Custom CustomLAM <repository_id>:<repository_base_name>
```

Note aux utilisateurs des versions précédentes de Tivoli Workload Scheduler :
Pratiquement toutes les propriétés demeurent inchangées par rapport aux versions précédentes de Tivoli Workload Scheduler.

Les propriétés suivantes sont nouvelles :

enabled=true | false
: Indique si la sécurité d'application est activée (true) ou non (false). La valeur par défaut est "true".

enforceJava2Security=false
: Indiquez si la sécurité Java 2 est activée (true). Tivoli Workload Scheduler ne prend pas en charge la sécurité Java 2, de sorte que cette option doit être définie sur false (la valeur par défaut).

useDomainQualifiedUserNames=true | false
: Indiquez si les noms d'utilisateur (qualifiés par domaine) doivent être utilisés (true). Si la valeur de cette propriété est true, tous les noms d'utilisateur dans le fichier de sécurité doivent être qualifié avec leurs domaines. La valeur par défaut est false. La modification de cette valeur pendant l'utilisation de Tivoli Workload Scheduler pourrait menacer votre accès au produit ; si vous devez le faire, discutez de la meilleure méthode avec le service de support logiciel IBM.

cacheTimeout=<*secondes*>
: Indique la valeur de dépassement de délai du cache de sécurité exprimée en secondes. Le dépassement de délai du cache de sécurité peut avoir une incidence sur les performances. La valeur de cette propriété indique la fréquence d'actualisation des caches liés à la sécurité. Les informations de sécurité relatives aux beans, droits d'accès et données d'identification sont mises en cache. Lorsque le délai d'attente du cache expire, toutes les informations mises en cache sont invalidées. Les demandes d'informations qui suivent aboutissent à une consultation de base de données. Parfois, l'acquisition de ces informations nécessite l'appel d'une authentification native ou LDAP (Lightweight Directory Access Protocol). Ces deux appels constituent des opérations relativement néfastes en termes de performances. Déterminez le meilleur parti pris pour l'application en consultant les modèles d'utilisation et les besoins du site en matière de sécurité. La valeur par défaut du dépassement de délai d'attente du cache de sécurité est 600 secondes. Si vous disposez d'un petit nombre d'utilisateurs, elle peut être supérieure à cette valeur, ou inférieure si vous disposez d'un grand nombre d'utilisateurs.

ltpaTimeout=<*secondes*>
Indique le dépassement de délai du cache pour les données LTPA. La valeur de dépassement de délai LTPA ne doit pas être inférieure à celle du dépassement de délai du cache de sécurité. La valeur par défaut est 720 secondes.

issuePermissionWarning=true
Indique que lors du déploiement et du démarrage de l'application, l'environnement d'exécution de la sécurité émet un avertissement si des droits d'accès personnalisés ont été octroyés aux applications (true). Les autorisations personnalisées sont des droits définis par les applications de l'utilisateur et non des droits sur les API Java. Les droits sur les API Java sont des droits qui concernent les packages java.* et javax.*. Pour Tivoli Workload Scheduler laissez le paramètre sur "true".

activeProtocol=CSI
Indique le protocole d'authentification actif pour l'invocation RMI (Remote Method Invocation) sur les demandes de protocole RMI IIOP (Internet Inter-ORB Protocol), lorsque la sécurité est activée. Pour Tivoli Workload Scheduler laissez le paramètre sur "CSI".

useFIPS=true | false
Indiquez si le réseau Tivoli Workload Scheduler est conforme aux normes FIPS (true) et utilise GSKit pour SSL, ou s'il n'est pas conforme aux normes FIPS (false) et utilise OpenSSL. La valeur par défaut est false. Pour plus d'informations, voir «Conformité aux normes FIPS», à la page 300.

activeAuthMechanism=LTPA
Indique le mécanisme d'authentification actif. Pour Tivoli Workload Scheduler laissez le paramètre sur "LTPA".

activeUserRegistry=<*liste_séparée_par_des_espaces*>
Indique une liste d'entrées séparées par des espaces utilisée pour identifier les registres à activer. Toutes les entrées répertoriées ici seront activées ensemble dans le Registre d'utilisateurs fédéré VMM. Les valeurs admises sont les suivantes :

- `LocalOS`
- `Custom` (sur les systèmes d'exploitation UNIX et Linux), `CustomPAM` (sur les systèmes d'exploitation AIX)
- `CustomLAM`
- `LDAP`
- `WIM`
- <*ID_REFERENTIEL*>:<*NOM_BASE_DOMAINE_REFERENTIEL*E>

  Utilisez cette option si vous avez configuré un autre référentiel à l'aide de Integrated Solutions Console ou tout autre mécanisme que les outils Tivoli Workload Scheduler WebSphere Application Server, et que vous voulez activer un tel référentiel individuellement ou avec les registres par défaut mentionnés ci-dessus.

  Par exemple, si vous avez créé un référentiel portant l'ID "BluePages" et le nom de base de domaine ibm.com", vous devez spécifier :

  ```
  activeUserRegistry=BluePages:o=ibm.com <other_repository_ids>
  ```

Remarque : Sous AIX, le système d'authentification `CustomPAM` et les valeurs de `CustomLAM` et `LocalOS` s'excluent mutuellement.

Panneau Référentiel fédéré

Panneau obligatoire.

```
############################################################
Federated Repository Panel
############################################################
PrimaryAdminId=
UseRegistryServerId=
ServerID=
ServerPassword=
VMMRealm=TWSREALM
VMMRealmDelimiter=@
VMMIgnoreCase=true
```

Note aux utilisateurs des versions précédentes de Tivoli Workload Scheduler : il s'agit d'un nouveau panneau.

PrimaryAdminId=<*nom*>
: Indique le nom de l'utilisateur doté des privilèges d'administration qui est défini dans le référentiel, par exemple, `adminUser`. Le nom d'utilisateur est utilisé pour la connexion à la console d'administration lorsque la sécurité administrative est activée. WebSphere Application Server nécessite un administrateur distinct de l'identité de l'utilisateur du serveur de sorte que les actions administratives puissent faire l'objet d'audit.

UseRegistryServerId=true | false
: Indique si l'identité du serveur doit être générée automatiquement ou fournie manuellement.
  - Si elle a la valeur true, cette propriété active le serveur d'applications pour générer l'identité du serveur, solution recommandée pour les environnements comprenant uniquement des noeuds WebSphere Application Server 6.1 ou version ultérieure. Les identités de serveur générées automatiquement ne sont pas stockées dans un référentiel d'utilisateurs.
  - Si elle a la valeur false, cette propriété nécessite la spécification d'un utilisateur dans ServerID pour la communication de processus interne.

ServerID=<*nom*>
: Indique une identité d'utilisateur dans le référentiel, utilisée pour la communication de processus interne. Les configurations comprenant également WebSphere Application Server V6.0.x nécessitent une identité d'utilisateur de serveur qui est définie dans le référentiel d'utilisateurs actif.

ServerPassword=<*password*>
: Indique le mot de passe correspondant au nom indiqué dans ServerID.

VMMRealm=<*nom*>
: Indique le nom du domaine. La valeur par défaut est TWSREALM. Vérifiez que cette propriété est configurée correctement pour votre environnement, afin de permettre la communication entre les différents serveurs et d'augmenter la vitesse.

VMMRealmDelimiter=<*valeur*>
: Indique le délimiteur utilisé pour faire la distinction entre l'utilisateur et le domaine lors de la fédération de plusieurs référentiels avec des domaines différents. La valeur par défaut est "@".

VMMIgnoreCase=true | false
: Indique si une vérification d'autorisation insensible à la casse a été effectuée.

- Si la valeur est true, cette propriété indique que le respect de la casse n'est pas une considération à prendre en compte pour l'autorisation. Elle doit être définie sur true lors de l'activation du référentiel LDAP avec IBM Tivoli Directory Server
- Si la valeur est false, la casse de l'ID utilisateur en cours d'authentification sera utilisée pour établir la correspondance avec les ID utilisateur figurant dans le registre.

Panneau LDAP

Complétez ce panneau lors d'une configuration pour un registre d'utilisateurs LDAP.

```
############################################################
LDAP Panel
############################################################
LDAPServerType=IDS
LDAPHostName=
LDAPPort=389
LDAPBaseDN=
LDAPBaseDNEntry=
LDAPBindDN=
LDAPBindPassword=
LDAPloginProperties=
LDAPsearchTimeout=120
LDAPsslEnabled=false
LDAPsslConfig=
LDAPCertificateFilter=
LDAPCertificateMapMode=EXACT_DN
```

Note aux utilisateurs des versions précédentes de Tivoli Workload Scheduler : Ce panneau n'est pas nouveau mais il est très différent par rapport aux versions précédentes.

LDAPServerType=<*nom*>
Indique le type de serveur LDAP auquel vous connecter. Si vous utilisez le serveur IBM Tivoli Directory Server pour z/OS, vous devez spécifier **IDS** . Les valeurs admises sont les suivantes :

IDS IBM Tivoli Directory Server (la valeur LDAP par défaut)

AD Microsoft Windows Active Directory

ZOSDS
z/OS Integrate Security Service LDAP Server

LDAPHostName=<*adresse_IP_ou_nom_d'hôte*>
Indique le nom d'hôte du serveur LDAP principal. Ce nom d'hôte est soit une adresse IP ou un nom DNS (Domain Name Service).

LDAPPort=<*numéro*>
Indique le port du serveur LDAP. La valeur par défaut est 389, ce qui ne correspond pas à une connexion SSL (Secure Sockets Layer). Utilisez le port 636 pour une connexion SSL. Pour certains serveurs LDAP, vous pouvez spécifier un port différent pour une connexion SSL ou non SSL.

LDAPBaseDN=<*nom_distinctif*>
Indique le nom distinctif (DN) LDAP de l'entrée de base au sein du référentiel, qui indique le point de début des recherches LDAP du service d'annuaire. L'entrée et ses descendants sont mappés à la branche identifiée par l'entrée de base "twaLDAP". Si cette zone est laissée à blanc, par défaut, la branche prend la valeur de la racine du référentiel LDAP.

Par exemple, pour un utilisateur avec un nom distinctif (DN) correspondant à `cn=John Doe , ou=Rochester, o=IBM, c=US`, indiquez que le paramètre `LDAPBaseDN` a l'une des options suivantes :

- `ou=Rochester, o=IBM, c=US`
- `o=IBM c=US`
- `c=US`

A des fins d'autorisation, cette zone est sensible à la casse. Cette spécification implique que si un jeton est reçu, par exemple d'une autre cellule ou de Lotus Domino, le DN de base pour le serveur doit correspondre exactement au DN de base de l'autre cellule ou du serveur Lotus Domino. Si le respect de la casse n'est pas une considération à prendre en compte pour les autorisations, activez l'option Ignore case for authorization (Ignorer la casse pour les autorisations).

LDAPBaseDNEntry=<*liste_noms_distinctifs*>
Indique le nom distinctif (DN) LDAP d'une entrée de base qui identifie de manière unique le référentiel externe dans le domaine. Si le domaine comprend plusieurs référentiels, cette zone permet de définir un nom distinctif supplémentaire (DN) qui identifie de manière unique ce jeu d'entrées au sein du domaine. Si vous laissez cette zone vide, la valeur par défaut est alors :

`LDAPBaseDN`
Pour la première personnalisation.

`old value`
Pour la personnalisation suivante. La valeur par défaut est **o=twaLDAP**.

Par exemple, les référentiels LDAP1 et LDAP2 doivent tous les deux utiliser o=ibm, c=us comme entrée de base dans le référentiel. Utilisez le nom distinctif de cette zone pour identifier de manière unique ce jeu d'entrées dans le domaine. Par exemple : o=ibm,c=fr pour LDAP1 et o=ibm2,c=fr pour LDAP2. Le nom distinctif spécifié dans cette zone est mappé au nom distinctif LDAP de l'entrée de base dans le référentiel.

LDAPBindDN=<*nom*>
Indique le nom distinctif (DN) que le serveur d'applications doit utiliser lors de la liaison avec le référentiel LDAP. Si aucun nom n'est spécifié, la liaison du serveur d'applications se fera de manière anonyme. Dans la plupart des cas, `LDAPBindDN` et `LDAPBindPassword` sont nécessaires. En revanche, lorsqu'une liaison anonyme peut satisfaire toutes les fonctions requises, `LDAPBindDN` et `LDAPBindPassword` ne sont pas nécessaires.

LDAPBindPassword=<*password*>
Indique le mot de passe que le serveur d'applications doit utiliser lors de la liaison avec le référentiel LDAP.

LDAPloginProperties=<*liste_jetons_connexion*>
Indique les jetons de connexion à utiliser pour se connecter au serveur d'applications. Cette zone accepte plusieurs jetons de connexion, séparés par un point-virgule (;). Par exemple, `uid;mail`. Toutes les propriétés de connexion font l'objet d'une recherche lors de la connexion. En cas de détection de plusieurs entrées ou d'aucune entrée, une erreur s'affiche. Par exemple, si vous spécifiez les propriétés de connexion sous la forme `uid;mail` et que l'ID de connexion est Bob, le filtre de recherche lance une

recherche sur `uid=Bob` ou `mail=Bob`. Lorsque la recherche renvoie une seule entrée, l'authentification peut se poursuivre. Sinon, une erreur est renvoyée.

Si vous indiquez plusieurs jetons de connexion, l'ordre que vous attribuez à ces jetons est très important, car quel que soit le jeton avec lequel l'utilisateur est authentifié, VMM définit la première propriété comme nom principal. Ce nom principal est ensuite transmis à Tivoli Workload Scheduler. Par exemple, si vous définissez les propriétés de connexion avec `cn;mail`, même si l'utilisateur se connecte avec "mail", le nom principal renvoyé sera "cn" et les vérifications de sécurité de Tivoli Workload Scheduler (dans le fichier de sécurité, par exemple) sont effectuées en utilisant la valeur "cn" pour cet utilisateur.

LDAPsearchTimeout=<*valeur*>
Indique la valeur du délai d'attente de réponse en millisecondes d'un serveur LDAP avant abandon de la demande. La valeur 0 indique qu'il n'existe aucune limite de temps pour la recherche.

LDAPsslEnabled=true|false
Indique si la communication par connexion sécurisée est activée sur le serveur LDAP.

- Si la valeur est true, SSL est activé et les paramètres SSL (Secure Sockets Layer) pour LDAP sont utilisés, s'ils sont fournis.
- Si la valeur est false, SSL n'est pas activé.

LDAPsslConfig=<*alias*>
Indique l'alias de configuration SSL à utiliser pour les communications SSL sortantes LDAP. Cette option remplace la configuration gérée de manière centralisée pour la plateforme JNDI. La valeur par défaut est `"DefaultNode/DefaultSSLSettings"`.

LDAPCertificateFilter=<*spécifications_filtre*>
Indique la propriété de mappage de certificat de filtre utilisée pour le filtre LDAP. Le filtre est employé pour mapper les attributs figurant dans le certificat client aux entrées du référentiel LDAP. Si plusieurs entrées LDAP correspondent à la spécification de filtre en phase d'exécution, l'authentification échoue car le résultat est une correspondance ambiguë. La syntaxe ou la structure de ce filtre est :

<LDAP_attribute>=${<attribut_certificat_client>}

Par exemple, `uid=${SubjectCN}`.

La partie à gauche de la spécification de filtre correspond à un attribut LDAP qui dépend du schéma pour lequel votre serveur LDAP est configuré. La partie à droite du filtre de spécification correspond à l'un des attributs publics figurant dans votre certificat client. La partie à droite doit commencer par le symbole du dollar ($) et une accolade ouvrante ({) et se terminer par une accolade fermante (}). Vous pouvez utiliser l'une des valeurs d'attribut de certificat suivantes à droite de la spécification de filtre (la casse des chaînes est importante) :

- ${UniqueKey}
- ${PublicKey}
- ${PublicKey}
- ${Issuer}
- ${NotAfter}
- ${NotBefore}

- ${SerialNumber}
- ${SigAlgName}
- ${SigAlgOID}
- ${SigAlgParams}
- ${SubjectCN}
- ${Version}

LDAPCertificateMapMode=<*valeur*>
Indique si les certificats X.509 doivent être mappés dans un répertoire LDAP par `EXACT_DN` ou `CERTIFICATE_FILTER`.

Panneau LDAP avancé

Complétez ce panneau lors d'une configuration pour un registre d'utilisateurs LDAP.

```
############################################################
Advanced LDAP Panel
############################################################
LDAPUserEntityType=PersonAccount
LDAPUserObjectClasses=
LDAPUserSearchBases=
LDAPUserSearchFilter=
LDAPUserRDNAttributes=
LDAPGroupEntityType=Group
LDAPGroupObjectClasses=
LDAPGroupSearchBases=
LDAPGroupSearchFilter=
LDAPGroupSearchFilter=
LDAPGroupRDNAttributes=
LDAPOrgContainerEntityType=OrgContainer
LLDAPOrgContainerObjectClasses=
LDAPOrgContainerSearchBases=
LDAPOrgContainerSearchFilter=
LDAPOrgContainerRDNAttributes=
LDAPGroupConfigName=ibm-allGroups
LDAPGroupConfigScope=all
LDAPGroupConfigMemberNames=member;uniqueMember
LDAPGroupConfigMemberClasses=groupOfNames;groupOfUniqueNames
LDAPGroupConfigMemberScopes=direct;direct
LDAPGroupConfigMemberDummies=uid=dummy;
```

Note aux utilisateurs des versions précédentes de Tivoli Workload Scheduler: ce panneau n'est pas nouveau, mais il est différent de celui des versions précédentes.

LDAPUserEntityType=<*value***>**
Indique le nom du type d'entité `PersonAccount` pris en charge par les référentiels de membres. Par défaut, cette valeur est "`PersonAccount`"

LDAPUserObjectClasses=<*liste_types_entité***>**
Indique les classes d'objets mappées au type d'entité `PersonAccount`. Vous pouvez spécifier plusieurs valeurs séparées par un point-virgule (;). Les entrées LDAP contenant une ou plusieurs classes d'objet, appartiennent à ce type d'entité. Vous ne pouvez pas mapper plusieurs types d'entité à la même classe d'objets LDAP.

LDAPUserSearchBases=<*liste_bases_recherche***>**
Indique les bases de recherche utilisées pour rechercher le type d'entité `PersonAccount`. Les bases de recherche spécifiées doivent être des branches de l'entrée de base dans le référentiel.

Par exemple, vous pouvez spécifier les bases de recherche suivantes, où `o=ibm,c=us` correspond à l'entrée de base dans le référentiel :

- `o=ibm,c=us`

- `cn=users,o=ibm,c=us`
- `ou=austin,o=ibm,c=us`

Dans l'exemple précédent, vous ne pouvez pas spécifier les bases de recherche `c=us` ou `o=ibm,c=uk`.

Délimitez les bases de recherche par un point-virgule (;). Par exemple : `ou=austin,o=ibm,c=us;ou=raleigh,o=ibm,c=us`

LDAPUserSearchFilter=<*nom***>**
Indique le filtre de recherche LDAP utilisé pour rechercher le type d'entité `PersonAccount`. Si aucun filtre de recherche n'est spécifié, les classes d'objet et les propriétés de nom distinctif relatif (RDN) sont utilisées pour générer le filtre de recherche.

LDAPUserRDNAttributes=<*liste_attributs_rdn***>**
Spécifie les propriétés de nom distinctif relatif (RDN) qui sont utilisées pour générer le filtre de recherche pour le type d'entité `PersonAccount`. Vous pouvez indiquer plusieurs valeurs délimitées un point-virgule (;). Spécifiez cette valeur sous la forme *<rdn_attribute_name>:<classe_objet>*. *classe_objet* est facultatif ; si vous la spécifiez, cette valeur est utilisée par le type d'entité **PersonAccount** pour mapper le *nom_attribut_rdn* correspondant.

LDAPGroupEntityType=<*nom***>**
Indique le nom du type d'entité Group pris en charge par les référentiels de membres. Par défaut, cette valeur est **Group**.

LDAPGroupObjectClasses=<*liste_classes_objet***>**
Indique les classes d'objets mappées au type d'entité Group. Vous pouvez spécifier plusieurs valeurs séparées par un point-virgule (;). Les entrées LDAP contenant une ou plusieurs classes d'objet, appartiennent à ce type d'entité. Vous ne pouvez pas mapper plusieurs types d'entité à la même classe d'objets LDAP.

LDAPGroupSearchBases=<*liste_bases_recherche***>**
Indique les bases de recherche utilisées pour rechercher le type d'entité Group. Les bases de recherche spécifiées doivent être des branches de l'entrée de base dans le référentiel. Pour plus d'informations, voir "`LDAPUserSearchBases`".

LDAPGroupSearchFilter=<*nom***>**
Indique le filtre de recherche utilisé pour rechercher le type d'entité Group. Si aucun filtre de recherche n'est spécifié, les classes d'objet et les propriétés de nom distinctif relatif (RDN) sont utilisées pour générer le filtre de recherche.

LDAPGroupRDNAttributes=<*liste_attributs_rdn***>**
Spécifie les propriétés de nom distinctif relatif (RDN) utilisées pour générer le filtre de recherche pour le type d'entité **Group**. Vous pouvez indiquer plusieurs valeurs délimitées un point-virgule (;). Spécifiez cette valeur sous la forme *<rdn_attribute_name>:<classe_objet>*. *classe_objet* est facultatif ; si vous spécifiez cette valeur, elle sera utilisée par le type d'entité **Group** pour mapper le *nom_attribut_rdn* correspondant.

LDAPOrgContainerEntityType=<*nom***>**
Indique le nom du type d'entité `OrgContainer` pris en charge par les référentiels de membres. Par défaut, la valeur est "`OrgContainer`".

LDAPOrgContainerObjectClasses=<*liste_classes_objet***>**
Indique les classes d'objets mappées au type d'entité `OrgContainer`. Vous

pouvez spécifier plusieurs valeurs séparées par un point-virgule (;). Les entrées LDAP contenant une ou plusieurs classes d'objet, appartiennent à ce type d'entité. Vous ne pouvez pas mapper plusieurs types d'entité à la même classe d'objets LDAP.

LDAPOrgContainerSearchBases=<*liste_bases_recherche***>**
Indique les bases de recherche utilisées pour rechercher le type d'entité `OrgContainer`. Les bases de recherche spécifiées doivent être des branches de l'entrée de base dans le référentiel. Pour plus d'informations, voir "`LDAPUserSearchBases`".

LDAPOrgContainerSearchFilter=<*nom***>**
Indique le filtre de recherche LDAP utilisé pour rechercher le type d'entité `OrgContainer`. Si aucun filtre de recherche n'est spécifié, les classes d'objet et les propriétés de nom distinctif relatif (RDN) sont utilisées pour générer le filtre de recherche.

LDAPOrgContainerRDNAttributes=<*liste_attributs_rdn***>**
Spécifie les propriétés de nom distinctif relatif (RDN) utilisées pour générer le filtre de recherche pour le type d'entité `OrgContainer`. Vous pouvez indiquer plusieurs valeurs délimitées un point-virgule (;). Spécifiez cette valeur sous la forme *rdn_attribute_name*:<*classe_objet*>. *classe_objet* est facultatif ; si vous spécifiez cette valeur, elle sera utilisée par le type d'entité **OrgContainer** pour mapper le *nom_attribut_rdn* correspondant.

LDAPGroupConfigName=<*value***>**
Indique le nom de l'attribut d'appartenance au groupe. Un seul attribut d'appartenance peut être défini pour chaque référentiel LDAP. Chaque entrée LDAP doit posséder cet attribut pour indiquer les groupes auxquels appartient l'entité.

Par exemple, `memberOf` est le nom de l'attribut d'appartenance utilisé dans Active Directory. `Ibm-allGroups` est le nom de l'attribut d'appartenance utilisé dans IBM Tivoli Directory Server. L'attribut d'appartenance au groupe comportent des valeurs qui référencent les groupes auxquels appartient cette entrée. Si `User` appartient à `Group`, la valeur de l'attribut `memberOf` de `User` doit contenir le nom distinctif de `Group`. Si votre serveur LDAP ne prend pas en charge l'attribut d'appartenance au groupe, ne spécifiez pas cet attribut. Le référentiel LDAP peut rechercher des groupes en effectuant une recherche dans les attributs de membre du groupe, mais les performances risquent de ralentir.

LDAPGroupConfigScope=<*value***>**
Indique la portée de l'attribut d'appartenance au groupe. La valeur par défaut est direct. Pour IBM Tivoli Directory Server, la valeur à spécifier est "`all`". Pour Active Directory la valeur à spécifier est "`direct`". Valeurs admises :

direct L'attribut d'appartenance contient uniquement des groupes directs. Les groupes directs sont les groupes qui contiennent le membre.

Par exemple, si `Groupe1` contient `Groupe2` et que `Groupe2` contient `Utilisateur1`, alors `Groupe2` est un groupe direct d'`Utilisateur1`, mais `Groupe1` n'est pas un groupe direct d'`Utilisateur1`.

nested L'attribut d'appartenance contient des groupes directs et des groupes imbriqués.

all L'attribut d'appartenance contient des groupes directs, des groupes imbriqués et des membres dynamiques.

LDAPGroupConfigMemberNames=<*liste_noms*>
Indique les noms des "attributs de membre" dans LDAP. Vous pouvez spécifier d'autres "attributs de membre" en les séparant par un point-virgule ";".

Par exemple, `member` et `uniqueMember` sont des noms d'attributs de membre d'utilisation courante. L'attribut de membre est utilisé pour stocker les valeurs qui référencent les membres contenus dans le groupe. Par exemple, un type de groupe avec une classe d'objets `groupOfNames` a un attribut de membre nommé `member` ; le type de groupe avec la classe d'objets `groupOfUniqueNames` a un attribut de membre nommé `uniqueMember`.

Un référentiel LDAP prend en charge plusieurs types de groupe si plusieurs attributs de membre et leurs classes d'objet de groupe associées sont spécifiés.

LDAPGroupConfigMemberClasses=groupOfNames;groupOfUniqueNames
Indique la classe d'objets du groupe qui utilise ces attributs de membre. Si cette zone n'est pas définie, cet attribut de membre s'applique à toutes les classes d'objets de groupe. Vous pouvez spécifier d'autres "classes de membre" en les séparant par un point-virgule ";".

Si vous spécifiez plusieurs valeurs dans "`LDAPGroupConfigMemberNames`", vous pouvez spécifier la classe associée au nom de membre spécifique définissant la valeur adéquate à la position adéquate.

Exemple :

```
LDAPGroupConfigMemberNames=member;uniqueMember
LDAPGroupConfigMemberClasses=groupOfNames;groupOfUniqueNames
```

LDAPGroupConfigMemberScopes
Indique la portée de l'attribut de membre. Vous pouvez spécifier d'autres "portées de membre" en les séparant par un point-virgule ";". La valeur par défaut est direct. Si vous spécifiez plusieurs valeurs dans "`LDAPGroupConfigMemberNames`", vous pouvez spécifier la portée associée au nom de membre spécifique définissant la valeur adéquate à la position adéquate. Valeurs admises :

direct L'attribut de membre contient uniquement des membres directs. Les membres directs sont les membres que le groupe contient directement. Par exemple, si `Groupe1` contient `Groupe2` et que `Groupe2` contient `Utilisateur1`, alors `Utilisateur1` est un membre direct de `Groupe2`, mais `Utilisateur1` n'est pas un membre direct de `Groupe1`.

nested L'attribut de membre contient des membres directs et des membres imbriqués.

all L'attribut de membre contient des membres directs, des membres imbriqués et des membres dynamiques.

Exemple :

```
LDAPGroupConfigMemberNames=member;uniqueMember
LDAPGroupConfigMemberScopes=direct;all
```

LDAPGroupConfigMemberDummies=uid=dummy;
Indique que si vous créez un groupe sans spécifier de membre, un membre factice sera complété pour éviter de créer une exception d'attribut obligatoire manquant. La valeur admise est "`uid=dummy`". Si vous spécifiez plusieurs valeurs dans "LDAPGroupConfigMemberNames", vous pouvez

indiquer le membre factice associé au nom de membre spécifique définissant la valeur adéquate à la position adéquate.

Exemple :

```
LDAPGroupConfigMemberNames=member;uniqueMember
LDAPGroupConfigMemberDummies=uid=dummy;
```

Cela signifie que seul "member" aura un membre factice ("dummy member"), alors que uniqueMember" n'aura aucun membre factice activé.

Panneau SSL

Ce panneau est utilisé pour configurer SSL et n'est pas pertinent pour l'authentification d'utilisateur. Toutes les propriétés demeurent inchangées par rapport à la version précédente.

Panneau de données d'authentification J2C

Ce panneau est utilisé pour configurer l'authentification J2C et n'est pas pertinent pour l'authentification d'utilisateur. Toutes les propriétés demeurent inchangées par rapport à la version précédente.

ChangeSecurityProperties - sortie

La sortie du script ChangeSecurityProperties contient des messages pour vous aider à savoir si les modifications de configuration que vous avez apportées ont été acceptées. Ces messages comprennent les messages générés lors de la mise à niveau d'un composant Tivoli Workload Scheduler ou de Dynamic Workload Console à partir d'une version antérieure à la version 8.6.

L'exemple suivant illustre une sortie du script :

```
--------------------------------------------------------------------------------
I: Utilisation du fichier de propriétés : C:/TWS/wastools/FRESHI~1.TXT

I: Configuration de la sécurité globale...
I: LTPA détecté.
I: Délai d'attente LTPA défini à 720 minutes
I: Définition du mécanisme d'authentification avec (cells/DefaultNode|security.xml#LTPA_1)
I: Configuration SSL ...
I: Configuration du registre LocalOS ...
I: Le pont du registre d'utilisateurs twaLocalOS existe déjà
I: Configuration de l'authentification J2C avancée
I: Le registre d'utilisateurs twaLDAP LDAP existe déjà
I: Configuration LDAP ...
I: Configuration LDAP avancée ...
I: Activation du pont de registre d'utilisateurs "LocalOS"
I: Activation du pont de registre d'utilisateurs "LDAP"
I: Le mécanisme d'authentification actif est (cells/DefaultNode|security.xml#LTPA_1)
I: Le registre d'utilisateurs actif est (cells/DefaultNode|security.xml#WIMUserRegistry_1)
   avec les entrées de base :
   o=twaLocalOS
   o=twaLDAP
I: La propriété useRegistryServerId de VMM a la valeur "true"
I: La propriété ignoreCase de VMM a la valeur "true"
I: Le domaine VMM est "TWSREALM"
I: La valeur LDAPServerType du registre d'utilisateurs avec l'ID "twaLDAP" est "IDS"
I: Le mécanisme d'authentification actif est LTPA

I: Validation réussie. Configuration sauvegardée
--------------------------------------------------------------------------------
```

Chaque message commence par une lettre indiquant s'il s'agit d'une information (I), d'un avertissement (W) ou d'une erreur (E).

Remarque :

1. En cas d'erreur, la configuration reste inchangée.
2. Si une propriété n'est pas fournie dans le fichier en entrée, la zone correspondante dans la WebSphere Application Server intégré n'est pas mise à jour.
3. Si une zone de mot de passe est laissée à blanc ou sous la forme "*****", le mot de passe correspondant dans la WebSphere Application Server intégré n'est pas mis à jour.

Fin de la configuration

Après avoir configuré WebSphere Application Server pour l'utilisation d'une nouvelle configuration d'authentification, quelle que soit la méthode employée, vous devez suivre la procédure suivante :

1. Créez les utilisateurs et les groupes

Pour créer des utilisateurs et des groupes après avoir configuré le nouveau registre d'utilisateurs procédez comme suit. Cet exemple utilise Dynamic Workload Console mais vous pouvez également parvenir au même résultat avec **composer**.

1. Connectez-vous à WebSphere Application Server à l'aide de l'ID et du mot de passe administrateur WebSphere Application Server.
2. Attribuez le rôle `TWSWEBUIAdministrator` au nouvel administrateur.
3. Créez de nouveaux utilisateurs et de nouveaux groupes et attribuez-leur des rôles, comme indiqué dans la section «Configuration des rôles pour accéder à Dynamic Workload Console», à la page 106.

2. Mettez à jour le fichier de sécurité de Tivoli Workload Scheduler

Vous devez mettre à jour le fichier de sécurité Tivoli Workload Scheduler pour permettre aux utilisateurs d'accéder aux objets Tivoli Workload Scheduler (voir «Mise à jour du fichier de sécurité», à la page 156). L'exemple suivant présente un fichier de sécurité mis à jour, où l'utilisateur `TEST_LDAP` a été ajouté à la section `USER MAESTRO` :

```
USER MAESTRO
 CPU=@+LOGON=tws83,Administrator,administrator,TEST_LDAP
BEGIN
 USEROBJ CPU=@  ACCESS=ADD,DELETE,DISPLAY,MODIFY,ALTPASS,UNLOCK,LIST
 JOB  CPU=@  ACCESS=ADD,ADDDEP,ALTPRI,CANCEL,CONFIRM,DELDEP,DELETE,DISPLAY,KILL,
                MODIFY,RELEASE,REPLY,RERUN,SUBMIT,USE,LIST,UNLOCK
 SCHEDULE CPU=@ ACCESS=ADD,ADDDEP,ALTPRI,CANCEL,DELDEP,DELETE,
        DISPLAY,LIMIT,MODIFY,RELEASE,REPLY,SUBMIT,LIST,UNLOCK
 RESOURCE CPU=@ ACCESS=ADD,DELETE,DISPLAY,MODIFY,RESOURCE,USE,LIST,UNLOCK
 PROMPT    ACCESS=ADD,DELETE,DISPLAY,MODIFY,REPLY,USE,LIST,UNLOCK
 FILE  NAME=@ ACCESS=CLEAN,DELETE,DISPLAY,MODIFY,UNLOCK
 CPU  CPU=@  ACCESS=ADD,CONSOLE,DELETE,DISPLAY,FENCE,LIMIT,LINK,MODIFY,
        SHUTDOWN,START,STOP,UNLINK,LIST,UNLOCK
 PARAMETER  CPU=@ ACCESS=ADD,DELETE,DISPLAY,MODIFY,UNLOCK,LIST
 CALENDAR    ACCESS=ADD,DELETE,DISPLAY,MODIFY,USE,UNLOCK,LIST
FIN
```

Dans cet exemple, la propriété de sécurité **useDomainQualifiedUserNames** est associée à la valeur **false**, indiquant que l'utilisateur a été spécifié sans le domaine.

3. Mettez à jour les propriétés de WebSphere Application Server associées

Sous Windows et UNIX, après avoir modifié WebSphere Application Server pour utiliser le protocole LDAP, il est important de modifier les propriétés du client SOAP et de valider à nouveau les données d'identification de l'utilisateur des services Windows :

Mettez à jour les propriétés du client SOAP

Pour mettre à jour les propriétés du client SOAP, utilisez le script **updateWAS.sh/.bat** comme suit. (Pour plus d'informations, voir «Serveur d'applications - mise à jour des propriétés SOAP après modification de l'utilisateur WebSphere Application Server ou de son mot de passe», à la page 416).

```
updateWas.sh -user john.smith@domain.com -password zzzz
```

où **-user** et **-password** sont définis sur le nouvel utilisateur logique autorisé à arrêter WebSphere Application Server. L'utilisateur doit être défini dans le registre d'utilisateurs.

Mettez à jour les services Windows

Sous Windows, une foisWebSphere Application Server modifié pour utiliser le protocole LDAP, vous devez mettre à jour ou valider à nouveau les données d'identifications des services Windows à l'aide du script **updateWasService.bat**. (Pour plus d'informations, voir «Serveur d'applications - mise à jour des services Windows après des modifications» , à la page 415).

```
updateWasService -userid tws83 -password zzzz
–wasuser TEST_LDAP –waspassword xxxxxx
```

où **–userid** et **–password** sont définis sur l'ID utilisateur du système d'exploitation et sur le mot de passe de l'utilisateur qui exécute le processus WebSphere Application Server, **–wasuser** et **–waspassword** désignant le nouvel utilisateur logique autorisé à arrêter WebSphere Application Server. L'option **–wasuser** doit être définie dans le registre d'utilisateurs.

4. Propagez les modifications

Pour propager les modifications que vous avez apportées, procédez comme suit :

1. Mettez à jour les zones USERNAME et PASSWORD dans le fichier `useropts` sur chaque client de ligne de commande pointant sur votre poste de travail
2. Mettez à jour les zones USERNAME et PASSWORD dans le fichier `useropts` sur chaque agent tolérant aux pannes de votre environnement qui possède une connexion HTTP/HTTPS définie dans le fichier localopts qui pointe sur votre poste de travail. La connexion HTTP/HTTPS est utilisée pour soumettre un travail prédéfini ou un flot de travaux .
3. Mettez à jour les zones USERNAME et PASSWORD dans les paramètres de connexion au moteur sur chaque poste Dynamic Workload Console connecté.

Remarque : Pour modifier le fichier useropts, modifiez le nom d'utilisateur dans USERNAME et entrez le nouveau mot de passe dans PASSWORD en texte clair et entre guillemets. Le mot de passe sera chiffré lors de votre première connexion.

Exemples de configuration des serveurs LDAP

Reportez-vous également à ce modèle si vous utilisez un serveur LDAP IBM Tivoli Directory Server (ITDS) for z/OS pour accéder aux informations stockées dans RACF. Notez que sur le Integrated Solutions Console, les utilisateurs LDAP ne sont interrogés que par l'attribut userid. Vérifiez qu'une classe auxiliaire de type **eperson** et un attribut uid sont ajoutés à l'ID utilisateur LDAP.

Active Directory

```
################################################################
Global Security Panel
################################################################
enabled=true
enforceJava2Security=false
useDomainQualifiedUserNames=false
cacheTimeout=600
ltpaTimeOut=720
issuePermissionWarning=true
activeProtocol=CSI
useFIPS=false
activeAuthMechanism=LTPA
activeUserRegistry=WIM LocalOS LDAP

################################################################
Federated Repository Panel
################################################################
PrimaryAdminId=tws_admin
UseRegistryServerId=false
ServerID=
ServerPassword=
VMMRealm=myrealm
VMMRealmDelimiter=@
VMMIgnoreCase=true

################################################################
LDAP Panel
################################################################
LDAPServerType=AD
LDAPHostName=nc125088.romelab.it.ibm.com
LDAPPort=389
LDAPBaseDN=dc=test,dc=it
LDAPBaseDNEntry=dc=test,dc=it
LDAPBindDN=CN=ldap bind,DC=test,DC=it
LDAPBindPassword=*****
LDAPloginProperties=uid
LDAPsearchTimeout=120
LDAPsslEnabled=false
LDAPsslConfig=DefaultNode/DefaultSSLSettings
LDAPCertificateFilter=
LDAPCertificateMapMode=EXACT_DN
################################################################
Advanced LDAP Panel
################################################################
LDAPUserEntityType=PersonAccount
LDAPUserObjectClasses=user
LDAPUserSearchBases=
LDAPUserSearchFilter=(objectCategory=user)
LDAPUserRDNAttributes=sAMAccountName:user
LDAPGroupEntityType=Group
LDAPGroupObjectClasses=group
LDAPGroupSearchBases=
LDAPGroupSearchFilter=(objectCategory=group)
LDAPGroupRDNAttributes=cn:group
LDAPOrgContainerEntityType=OrgContainer
LDAPOrgContainerObjectClasses=organization;organizationalUnit
;domain;container
```

```
LDAPOrgContainerSearchBases=
LDAPOrgContainerSearchFilter=
LDAPOrgContainerRDNAttributes=ou:organizationalUnit;cn:container;
dc:domain;o:organization

LDAPGroupConfigName=memberOf
LDAPGroupConfigScope=direct
LDAPGroupConfigMemberNames=member
LDAPGroupConfigMemberClasses=groupOfNames
LDAPGroupConfigMemberScopes=direct
LDAPGroupConfigMemberDummies=
```

IBM Tivoli Directory Server

```
################################################################
Global Security Panel
################################################################
enabled=true
enforceJava2Security=false
useDomainQualifiedUserNames=false
cacheTimeout=600
ltpaTimeOut=720
issuePermissionWarning=true
activeProtocol=CSI
useFIPS=false
activeAuthMechanism=LTPA
activeUserRegistry= WIM LocalOS LDAP

################################################################
Federated Repository Panel
################################################################
PrimaryAdminId=tws_admin
UseRegistryServerId=false
ServerID=
ServerPassword=
VMMRealm=myrealm
VMMRealmDelimiter=@
VMMIgnoreCase=true
################################################################
LDAP Panel
################################################################
LDAPServerType=IDS
LDAPHostName=myhostname
LDAPPort=389
LDAPBaseDN=o=ibm.com
LDAPBindDN=
LDAPBindPassword=
LDAPloginProperties=mail;cn
LDAPsearchTimeout=120000
LDAPsslEnabled=false
LDAPsslConfig=
LDAPCertificateFilter=
LDAPCertificateMapMode=
################################################################
Advanced LDAP Panel
################################################################
LDAPUserEntityType=PersonAccount
LDAPUserObjectClasses=user;ePerson
LDAPUserSearchBases=
LDAPUserSearchFilter=(objectclass=ePerson)
LDAPUserRDNAttributes=mail:ePerson;cn:ePerson
LDAPGroupEntityType=Group
LDAPGroupObjectClasses=group;groupOfUniqueNames
LDAPGroupSearchBases=
LDAPGroupSearchFilter=(&(ou=memberlist)(ou=ibmgroups)
(o=ibm.com)(objectclass=groupOfUniqueNames))
LDAPGroupRDNAttributes=cn:groupOfUniqueNames
LDAPOrgContainerEntityType=OrgContainer
```

```
LDAPOrgContainerObjectClasses=organization;
organizationalUnit;domain;container
LDAPOrgContainerSearchBases=
LDAPOrgContainerSearchFilter=
LDAPOrgContainerRDNAttributes=ou:organizationalUnit;
cn:container;dc:domain;o:organization
LDAPGroupConfigName=ibm-allGroups
LDAPGroupConfigScope=all
LDAPGroupConfigMemberNames=uniqueMember
LDAPGroupConfigMemberClasses=groupOfUniqueNames
LDAPGroupConfigMemberScopes=direct
LDAPGroupConfigMemberDummies=
```

z/OS Integrate Security Service LDAP Server

```
#################################################################
Global Security Panel
#################################################################
enabled=true
enforceJava2Security=false
useDomainQualifiedUserNames=false
cacheTimeout=600
ltpaTimeOut=720
issuePermissionWarning=true
activeProtocol=CSI
useFIPS=false
activeAuthMechanism=LTPA
activeUserRegistry=LocalOS WIM LDAP
#################################################################
Federated Repository Panel
#################################################################
PrimaryAdminId=borgian
UseRegistryServerId=true
ServerID=tws86
ServerPassword=*****
VMMRealm=zos1166.romelab.it.ibm.com:636
VMMRealmDelimiter=@
VMMIgnoreCase=false
#################################################################
LDAP Panel
#################################################################
LDAPServerType=ZOSDS
LDAPHostName=zos1166.romelab.it.ibm.com
LDAPPort=636
LDAPBaseDN=ou=TIVOLI,o=IBM
LDAPBaseDNEntry=ou=TIVOLI,o=IBM
LDAPBindDN=cn=borgian
LDAPBindPassword=*****
LDAPloginProperties=uid
LDAPsearchTimeout=
LDAPsslEnabled=true
LDAPsslConfig=
LDAPCertificateFilter=
LDAPCertificateMapMode=exactdn
#################################################################
Advanced LDAP Panel
#################################################################
LDAPUserEntityType=PersonAccount
LDAPUserObjectClasses=eperson
LDAPUserSearchBases=ou=TIVOLI,o=IBM,
LDAPUserSearchFilter=
LDAPUserRDNAttributes=
LDAPGroupEntityType=Group
LDAPGroupObjectClasses=groupOfNames
LDAPGroupSearchBases=
LDAPGroupSearchFilter=
LDAPGroupRDNAttributes=
LDAPOrgContainerEntityType=OrgContainer
```

```
LDAPOrgContainerObjectClasses=organization;organizationalUnit;
domain;container
LDAPOrgContainerSearchBases=
LDAPOrgContainerSearchFilter=
LDAPOrgContainerRDNAttributes=ou:organizationalUnit;cn:
container;dc:domain;o:organization
LDAPGroupConfigName=
LDAPGroupConfigScope=
LDAPGroupConfigMemberNames=member
LDAPGroupConfigMemberClasses=groupOfNames
LDAPGroupConfigMemberScopes=direct
LDAPGroupConfigMemberDummies=uid=dummy
#################################################################
SSL Panel
#################################################################
alias=DefaultSSLSettings
keyFileName=${RACINE_INSTALLATION_UTILISATEUR}/etc/TWSServerKeyFile.jks
keyFilePassword=*****
keyFileFormat=JKS
trustFileName=${RACINE_INSTALLATION_UTILISATEUR}/etc/TWSServerTrustFile.jks
trustFilePassword=*****
trustFileFormat=JKS
clientAuthentication=false
securityLevel=HIGH
enableCryptoHardwareSupport=false
#################################################################
J2C Authentication Data Panel
#################################################################
j2cAlias=twsj2c
j2cUserid=db2admin
j2cPassword=*****
j2cDescription=TWS authentication data entry for data source
```

Schéma de serveur LDAP

Lors de la définition du schéma sur le serveur LDAP, tenez compte du fait que Dynamic Workload Console 8.6 est basé sur Dashboard Application Services Hub dont les requêtes envoyées au serveur LDAP supposent que l'attribut **uid** est défini pour les utilisateurs. Les utilisateurs LDAP ne sont interrogés que par l'attribut `userid`. Lorsque des utilisateurs sont importés dans LDAP à l'aide d'un fichier LDIF (LDAP Data Interchange Format), une classe auxiliaire de type `eperson` et un attribut `uid` sont ajoutés à l'ID utilisateur LDAP.

Pour plus d'informations, voir la section relative à la configuration d'un référentiel LDAP externe : http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/index.jsp?.

En conséquence, le schéma de serveur LDAP doit contenir l'attribut **uid** et la **classe d'objets** doit être `eperson` (la classe d'objets `person` utilisé par le schéma par défaut ne prend pas en charge un tel attribut). De plus, pour se conformer aux manuel des ID pour z/OS, **useNativeAuth** a été défini sur Tous.

Exemple d'utilisateurs LDAP définis :

```
LDAP Search is started ....
 Host = zos1166.romelab.it.ibm.com
 Port = 636
 Connection Type = SSL
 Timeout = 10 seconds
STEP 1 => Performing LDAP-SSL initialization
LDAP SSL initialization completed
STEP 2 => Connecting to LDAP server using the given credentials...
LDAP bind completed successfully.
```

```
STEP 3 => Searching on the server ...
------------------------------------------------------------------------
Enumerating attributes for DN : cn=John Doe, ou=TIVOLI, o=IBM
cn = John Doe
sn = BORGIAN
objectclass = organizationalperson
objectclass = eperson
objectclass = top
objectclass = person
```

où

```
ou=TIVOLI,o=ibm
ou=TIVOLI
objectclass=top
objectclass=organizationalUnit
description=Tivoli organization
```

Lors de la définition du référentiel LDAP, les attributs `Object Classes` (classes d'objets) et `Search bases` (bases de recherche) ont été adaptés à ce schéma LDAP.

Utilisation du module d'authentification enfichable

Tivoli Workload Scheduler permet d'améliorer le serveur WebSphere Application Server en prenant en charge un mécanisme d'authentification d'utilisateur reposant sur le module d'authentification enfichable.

Cette amélioration permet de disposer d'un mécanisme d'authentification unique capable d'authentifier les utilisateurs, sans tenir compte de l'élément sur lequel sont basées leurs implémentations de registre utilisateur (système d'exploitation local ou LDAP).

Tivoli Workload Scheduler installe automatiquement le module d'extension permettant à WebSphere Application Server d'utiliser l'authentification activée avec le module d'authentification enfichable. Le plug-in utilise le service avec le nom `other`. D'habitude, vous ne devez rien faire pour configurer le module d'authentification enfichable. Cependant, si le niveau de vos autorisations vous empêche d'utiliser `other`, vous devez ajouter le service avec le nom `checkpassword` dans le fichier `/etc/pam.conf`.

L'utilisation du module d'authentification enfichable étend également les fonctions de WebSphere Application Server à la prise en charge de l'authentification dans les environnements HP Trusted Mode.

Tivoli Workload Scheduler est défini par défaut pour utiliser un registre d'utilisateurs du module d'authentification enfichable nommé "custom" ou "customPAM". Si le module d'authentification enfichable n'est pas configuré avec ce registre, WebSphere Application Server regarde dans le registre d'utilisateurs local du gestionnaire de domaine maître.

Utilisation du module d'authentification chargeable

Le module d'authentification chargeable (LAM, Loadable authentication module) assure l'authentification et l'identification sur des systèmes AIX.

Le module LAM est différent du module PAM, qui exécute uniquement l'authentification.

Tivoli Workload Scheduler installe automatiquement le module d'extension qui active WebSphere Application Server pour l'utilisation de l'authentification activée avec le PAM comme système d'authentification sur les systèmes d'exploitation UNIX et Linux.

Le module LAM est utilisé pour l'identification, par exemple, les informations liées aux noms de compte et aux attributs, et/ou pour l'authentification, par exemple le stockage des mots de passe et la vérification. Le sous-système de sécurité AIX envoie les demandes d'authentification et d'identification vers la méthode appropriée à l'aide de deux attributs : **registry** et **SYSTEM**.

L'emplacement où sont définis les utilisateurs et leurs attributs (local, LDAP) est spécifié par l'attribut utilisateur **registry** et la manière dont les utilisateurs sont authentifiés (local, NIS, LDAP, Kerberos) par l'attribut **SYSTEM**.

Tivoli Workload Scheduler utilise un module de registre personnalisé pour intégrer le LAM à WebSphere Application Server. Vous pouvez activer le module LAM sous AIX en définissant la propriété **activeUserRegistry** sur `CustomLAM`, puis en exécutant le script changeSecurityProperties.sh qui indique le fichier de propriétés pour mettre à jour la valeur.

Chapitre 6. Administration du réseau

Le présent chapitre explique comment administrer le réseau Tivoli Workload Scheduler. Il contient les rubriques suivantes :

Présentation du réseau

Un réseau Tivoli Workload Scheduler consiste en un ou plusieurs domaines organisés de façon hiérarchique. Un domaine Tivoli Workload Scheduler est un regroupement logique de postes de travail comportant un gestionnaire de domaine et un certain nombre d'agents.

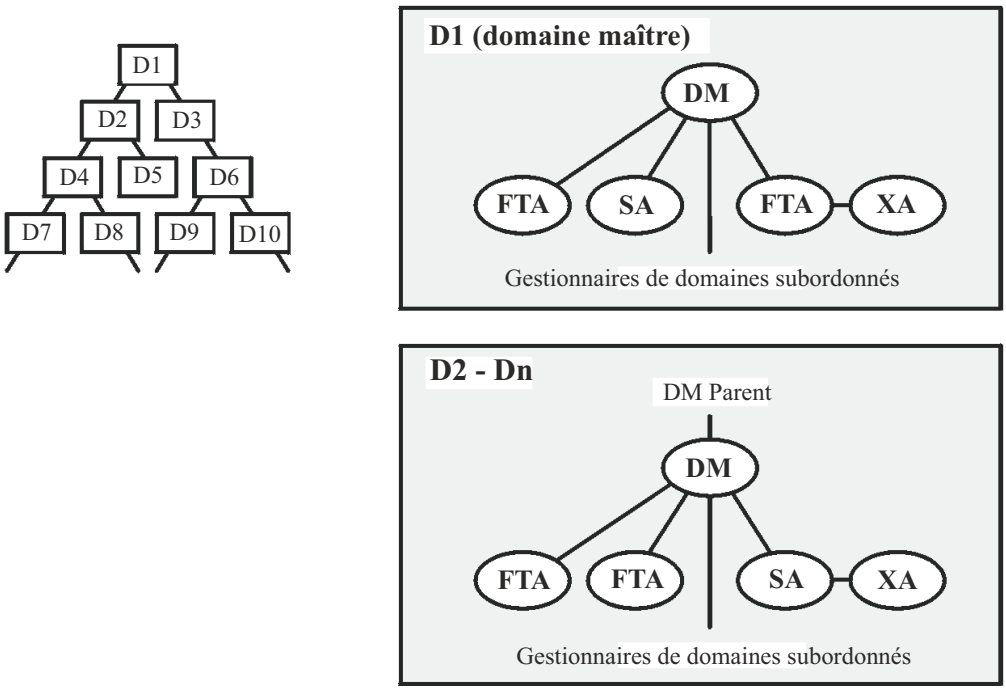


Figure 2. Structure du domaine du réseau Tivoli Workload Scheduler

Définition du réseau

Domaine
Groupe nommé de postes de travail Tivoli Workload Scheduler comportant un ou plusieurs agents et un gestionnaire de domaine. Les domaines ont tous un domaine parent excepté le domaine maître.

Domaine maître
Domaine principal d'un réseau Tivoli Workload Scheduler.

Gestionnaire de domaine maître
Gestionnaire de domaine maître appartenant au domaine principal d'un réseau Tivoli Workload Scheduler. Il contient les fichiers maîtres centralisés utilisés pour documenter les objets de planification. Il crée le fichier de contrôle de production (Symphony) au début de chaque période de production et se charge de toutes les opérations de journalisation et de génération de rapports du réseau. Voir également Gestionnaire de domaine.

gestionnaire de domaine maître de secours
Agent tolérant aux pannes capable d'assumer les responsabilités du gestionnaire de domaine maître.

Domaine parent
Domaine situé directement au-dessus du domaine en cours. À l'exception domaine maître, tous les domaines ont un domaine parent. Toutes les communications entrantes ou sortantes d'un domaine passent par le gestionnaire de domaine parent.

Gestionnaire de domaine
Concentrateur de gestion d'un domaine. Toutes les communications à destination et en provenance des agents d'un domaine passent par le gestionnaire de domaine. Voir également Gestionnaire de domaine maître.

Gestionnaire de domaine de secours
Agent tolérant aux pannes capable d'assumer les responsabilités de son gestionnaire de domaine.

Agent tolérant aux pannes
Poste de travail d'agent capable de résoudre les dépendances locales et de lancer ses travaux en l'absence d'un gestionnaire de domaine.

Agent standard
Poste de travail agent qui ne lance les travaux que sous la direction de son gestionnaire de domaine.

Agent étendu
Poste de travail qui ne lance des travaux que sous la direction de son hôte. Les agents étendus peuvent être utilisés pour servir d'interface entre Tivoli Workload Scheduler et des systèmes et applications non Tivoli Workload Scheduler.

Agent dynamique
Poste de travail qui gère une grande variété de types de travaux, par exemple, des travaux de base de données ou FTP spécifiques, en plus des types de travaux existants. Ce poste de travail est automatiquement créé et enregistré lorsque vous installez l'agent dynamique. Les processus d'installation et d'enregistrement sont effectués automatiquement. Ainsi, lorsque vous affichez l'agent dans Dynamic Workload Console, il apparaît mis à jour par l'agent assistant de ressource Resource Advisor Agent. Vous pouvez regrouper les agents dans des pools et pools dynamiques.

Dans une configuration simple, les agents dynamiques se connectent directement à un gestionnaire de domaine maître ou à un gestionnaire de domaine dynamique. Toutefois, dans des topologies de réseau plus complexes, si la configuration de réseau empêche le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique de communiquer directement avec le agent dynamique, vous pouvez configurer vos agents dynamiques pour utiliser une passerelle locale ou distante.

Hôte Fonction de planification requise par les agents étendus. Elle peut être effectuée par n'importe quel poste de travail Tivoli Workload Scheduler, à l'exception d'un autre agent étendu.

Communications réseau

Dans un réseau Tivoli Workload Scheduler, les agents communiquent avec leurs gestionnaires de domaine, et les gestionnaires de domaine communiquent avec leurs gestionnaires de domaine parents. Il existe deux types de communications :

- Initialisation de la période de début de production (distribution d'un nouveau fichier Symphony)
- Evénements de planification sous la forme de messages de changement d'état au cours de la période de production

Avant le début de chaque nouvelle période de production, le gestionnaire de domaine maître crée un fichier de contrôle de production nommé Symphony. Puis, Tivoli Workload Scheduler redémarre sur le réseau et le gestionnaire de domaine maître envoie une copie du nouveau fichier `Symphony` à chacun de ses agents automatiquement liés et de ses gestionnaires de domaines subordonnés. Les gestionnaires de domaine, à leur tour, envoient des copies de leurs agents et gestionnaires de domaine secondaire liés automatiquement. Les agents et les gestionnaires de domaines qui ne sont pas définis pour se connecter automatiquement sont initialisés avec une copie de `Symphony`, dès qu'une opération de connexion est exécutée dans Tivoli Workload Scheduler.

Une fois que le réseau à démarrer, les messages de planification, comme les démarrages et les arrêts de travaux, sont communiqués par les agents à leur gestionnaire de domaine, et via des gestionnaires de domaine parents au gestionnaire de domaine maître. Le gestionnaire de domaine maître diffuse alors les messages dans l'ensemble de l'arborescence hiérarchique pour mettre à jour les fichiers `Symphony` de tous les gestionnaires de domaine, et ces derniers communiquent les messages à tous les agents tolérants aux pannes de leur domaine fonctionnant en mode *FullStatus*.

Liaisons réseau

Les liaisons permettent aux postes de travail Tivoli Workload Scheduler d'un réseau de communiquer. Les liaisons sont contrôlées par l'indicateur de liaison automatique (Auto Link) et les commandes **link** et **unlink** du gestionnaire de console. Lorsqu'une liaison est ouverte, les messages sont transmis entre les postes de travail. Lorsqu'une liaison est fermée, le poste de travail expéditeur conserve les messages dans un fichier `pobox` local et les envoie au poste de travail de destination dès que la liaison est de nouveau ouverte.

Ceci signifie que lorsque les liaisons sont fermées, les files d'attente de messages se remplissent avec les messages destinés aux postes de travail inaccessibles. Pour

optimiser les performances de Tivoli Workload Scheduler, surveillez la présence de liaisons fermées sur les postes de travail et tentez de les rouvrir dès que possible.

Remarque : Les agents étendus n'ont pas de liaisons. Ils communiquent avec leurs gestionnaires de domaine par l'intermédiaire de leurs hôtes.

Pour qu'une liaison de poste de travail s'ouvre automatiquement, activez l'indicateur de liaison automatique dans la définition du poste de travail. La liaison s'ouvre d'abord au démarrage de Tivoli Workload Scheduler sur le poste de travail du domaine maître. Si le gestionnaire de sous-domaine et les postes de travail ne sont pas initialisés et si leur indicateur de liaison AUTO est activé, le gestionnaire de domaine maître tente d'établir une liaison vers ses subordonnés et commence les processus d'initialisation. Si l'indicateur de liaison automatique est désactivé, le poste de travail ne s'initialise qu'avec l'exécution de la commande **link** à partir du gestionnaire de domaine maître. Une fois le poste de travail initialisé, il démarre automatiquement et envoie une liaison de réponse à son gestionnaire de domaine.

Si vous arrêtez un poste de travail, ses liaisons vers les autres postes sont fermées. Toutefois les liaisons en provenance des autres postes de travail restent ouvertes jusqu'à ce que l'une des situations suivantes se produise :

- Le poste de travail arrêté est redémarré et une commande **link** est émise
- Les processus **mailman** des autres postes de travail expirent et émettent une commande **unlink** à destination du poste de travail

Lorsque la commande **link** est émise et que la connexion est établie, si le gestionnaire de domaine ne reçoit pas de réponse dans les délais impartis, le service `chkhltst` est appelé automatiquement par **mailman**.

Ce service vérifie que la boîte aux lettres du poste de travail peut être lue et recherche les erreurs éventuelles dans l'en-tête de la boîte aux lettres. Les informations qui en résultent sont consignées dans le fichier `TWSMERGE.log` du gestionnaire de domaine, comme suit :

- En cas d'erreur système à l'ouverture de la boîte aux lettres, le message suivant est émis : `AWSBDY126E An error occurred opening the Mailbox.msg file in` *`CPU_NAME`*`.`
- En cas d'erreur à l'ouverture de la boîte aux lettres due à la lecture de cette dernière par **mailman**, le message suivant est émis : `AWSBDY123I The Mailbox.msg file in` *`CPU_NAME`* `is correctly read by Mailman.`
- Si la boîte aux lettres est ouverte correctement mais qu'une erreur s'est produite lors de la lecture de l'en-tête, le message suivant est émis : `AWSBDY125E An error occurred reading the header of the Mailbox.msg file in` *`CPU_NAME`* `(Une erreur s'est produite pendant la lecture de l'en-tête du fichier Mailbox.msg dans (CPU_NAME)).`
- Si la boîte aux lettres est ouverte correctement sans erreur lors de la lecture de l'en-tête, le message suivant est émis : `AWSBDY124W The Mailbox.msg file in` *`CPU_NAME`* `is not read by Mailman (Le fichier Mailbox.msg dans (CPU_NAME) n'est pas lu par Mailman).`

Vous pouvez également lancer ce service manuellement à l'aide de la commande **conman**. Pour plus de détails, voir *IBM Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Pour vous assurer que les communications entre postes de travail sont correctement restaurées, vous pouvez émettre une commande **link** après avoir redémarré un poste de travail.

Utilisation des pare-feu

Pendant la conception d'un réseau Tivoli Workload Scheduler, l'administrateur doit savoir où sont positionnés les pare-feu dans le réseau, quels sont les agents tolérants aux pannes et les gestionnaires de domaine appartenant à un pare-feu particulier et quels sont les points d'entrée dans les pare-feu. Après avoir bien compris toutes ces informations, il doit définir l'attribut **behindfirewall** pour certaines des définitions de poste de travail dans la base de données Tivoli Workload Scheduler. En particulier, si la définition d'un poste de travail est définie avec l'attribut **behindfirewall** sur ON, cela signifie qu'il existe un pare-feu entre ce poste de travail et le gestionnaire de domaine maître Tivoli Workload Scheduler. Dans ce cas, la liaison poste de travail-gestionnaire de domaine est la seule liaison autorisée entre le poste de travail et son gestionnaire de domaine.

Tous les postes de travail Tivoli Workload Scheduler devraient être définis avec l'attribut **behindfirewall** si le lien avec le gestionnaire de domaine correspondant ou avec tout gestionnaire de domaine de la hiérarchie Tivoli Workload Scheduler jusqu'au gestionnaire de domaine maître, se trouve au-delà d'un pare-feu.

Lors du mappage d'un réseau Tivoli Workload Scheduler sur une structure pare-feu existante, il importe peu de savoir quels agents tolérants aux pannes et quels gestionnaires de domaine résident du côté sécurisé ou non sécurisé du pare-feu. Les limites du pare-feu doivent être la seule préoccupation. Par exemple, si le gestionnaire de domaine maître se trouve dans une zone non sécurisée et si certains gestionnaires de domaine se trouvent dans des zones sécurisées (ou inversement), n'entraîne aucune différence. La structure du pare-feu doit toujours être prise en considération à partir du gestionnaire de domaine maître et en suivant la hiérarchie du Tivoli Workload Scheduler, en marquant tous les postes de travail qui sont séparés de leur gestionnaire de domaine par un pare-feu.

Pour tous les postes de travail dont l'attribut **behindfirewall** est défini sur ON, les commandes conman **start** et **stop** sur le poste de travail et les commandes **showjobs** sont envoyés en fonction de la hiérarchie du domaine, au lieu que le gestionnaire de domaine maître ou le gestionnaire de domaine ouvre une connexion directe avec le poste de travail. La sécurité est de ce fait, considérablement améliorée.

Cet attribut fonctionne également pour plusieurs pare-feu. Vous pouvez indiquer qu'un poste de travail d'agent étendu se trouve derrière un pare-feu en réglant l'attribut **behindfirewall** sur ON sur le poste de travail hôte. L'attribut est en lecture seule dans le plan. Pour le modifier, l'administrateur doit le mettre à jour dans la base de données avant de recréer le plan.

Voir le *Tivoli Workload Scheduler - Guide d'utilisation et de référence* pour plus de détails sur la définition de cet attribut.

Configuration des communications des agents dynamiques via une passerelle

Dans certaines topologies de réseau complexes, le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique n'ont pas le droit de communiquer directement avec l'agent dynamique.

Dans une configuration simple, les agents dynamiques se connectent directement au gestionnaire de domaine maître ou au gestionnaire de domaine dynamique. Cependant, dans des topologies plus complexes, si la configuration de réseau empêche le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique de communiquer directement avec l'agent dynamique, par exemple, si les agents sont derrière un pare-feu et doivent communiquer via Internet ou s'ils doivent communiquer avec un processus NAT (Network Address Translation), vous pouvez alors configurer vos agents dynamiques pour utiliser une passerelle locale ou distante (remote).

Vous pouvez configurer vos agents dynamiques pour utiliser une passerelle permettant de communiquer avec le gestionnaire de domaine maître ou avec le gestionnaire de domaine dynamique lorsque vous installez un agent dynamique, ou vous pouvez configurer une passerelle à la suite de l'installation.

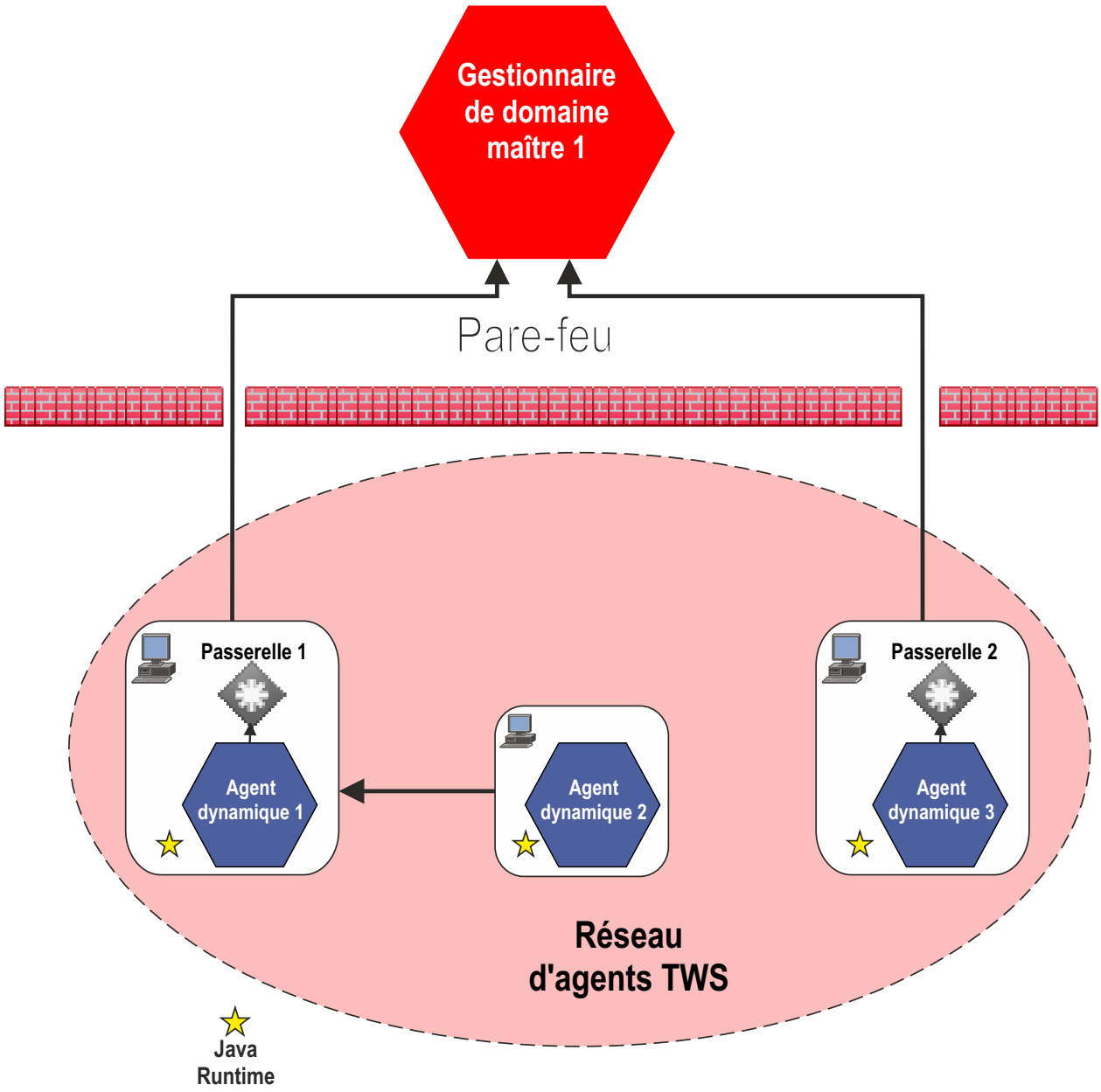
Pour plus d'informations sur les paramètres de passerelle disponibles avec l'installation d'un agent dynamique, voir la rubrique sous "Paramètres d'installation de l'agent" dans le *Guide de planification et d'installation*.

Pour configurer un agent dynamique Tivoli Workload Scheduler version 9.2 ou ultérieur existant pour communiquer avec son gestionnaire de domaine maître ou gestionnaire de domaine dynamique via une passerelle locale, exécutez les étapes de configuration suivantes :

1. Editez le fichier `JobManager.ini` sur le poste de travail de l'agent dynamique dans lequel la passerelle réside. Editez la section [ResourceAdvisorAgent] de sorte que la valeur du paramètre **`ResourceAdvisorURL`** soit `https://$(`*`serveur_tdwb`*`):$(`*`port_tdwb`*`)/ita/JobManagerGW/JobManagerRESTWeb/JobScheduler/resource`, où $(*serveur_tdwb*) et $(*port_tdwb*) correspondent aux nom d'hôte et port du poste de travail de l'agent dynamique avec la passerelle que vous configurez actuellement.
2. Arrêtez et démarrez l'agent dynamique pour implémenter les changements.

Le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique peut maintenant communiquer avec la poste de travail de l'agent dynamique via la passerelle.

Le diagramme suivant présente une topologie de réseau où le gestionnaire de domaine maître communique avec les agents dynamiques situés derrière un pare-feu, par le biais d'une passerelle configurée sur l'un des agents dynamiques.



Voici les paramètres de configuration utilisés dans la topologie de réseau décrite dans la figure:

Tableau 48. Paramètres de configuration

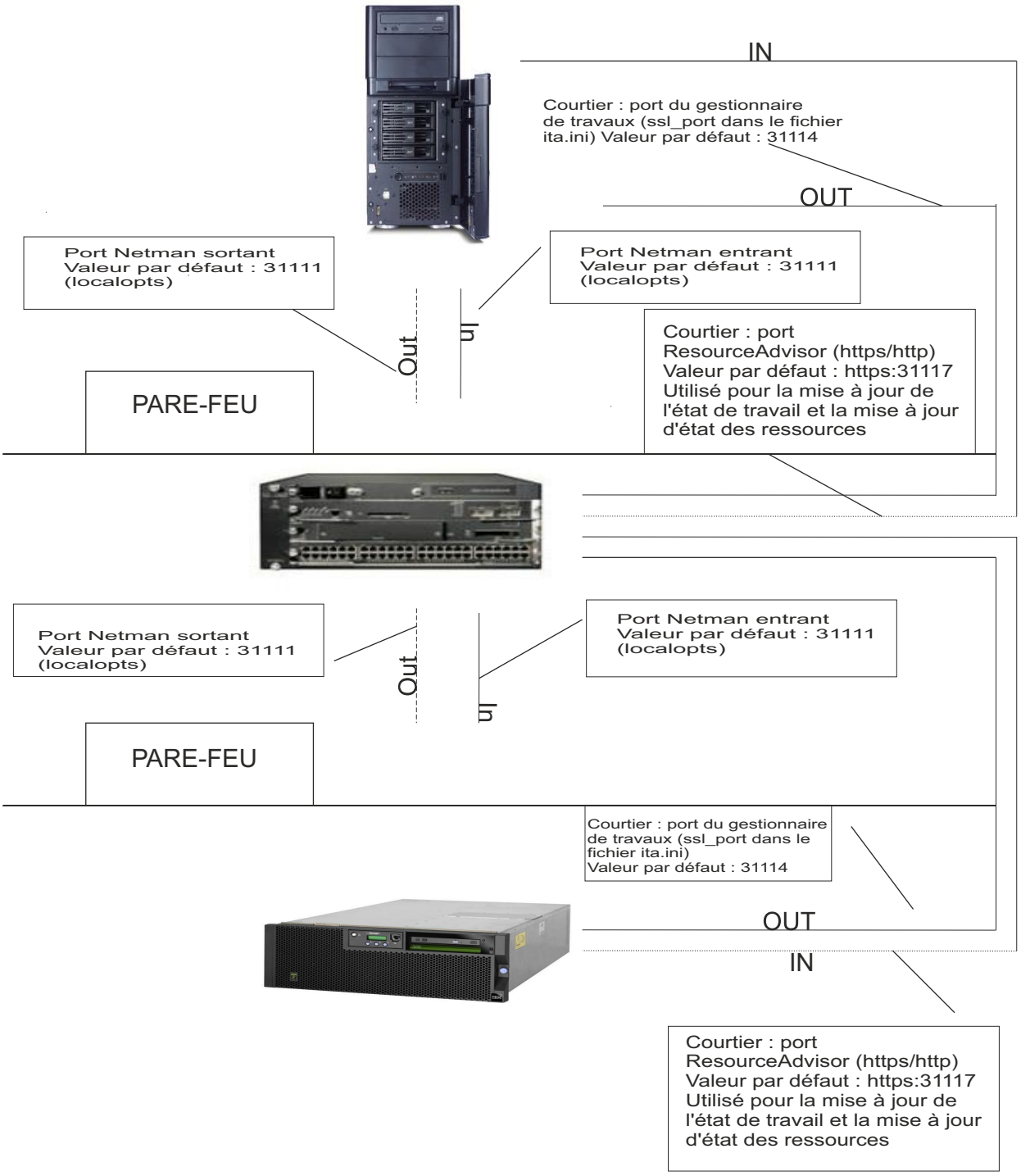
Agent dynamique	Fichier de configuration	Paramètre	Valeur
Agent dynamique 1 - passerelle locale	JobManager.ini	Section `[ResourceAdvisorAgent]` **`ResourceAdvisorUrl`**	`https://$(serveur_tdwb):$(port_tdwb)/ita/JobManagerGW/JobManagerRESTWeb/JobScheduler/resource` où, **$(***serveur_tdwb***)** Nom d'hôte du poste de travail de l'agent dynamique 1. **$(***port_tdwb***)** Numéro de port du poste de travail de l'agent dynamique 1.
Agent dynamique 2 - passerelle distante	JobManager.ini	Section `[ResourceAdvisorAgent]` **`ResourceAdvisorUrl`**	`https://$(serveur_tdwb):$(port_tdwb)/ita/JobManagerGW/JobManagerRESTWeb/JobScheduler/resource` où, **$(***serveur_tdwb***)** Nom d'hôte du poste de travail de l'agent dynamique 2. **$(***port_tdwb***)** Numéro de port du poste de travail de l'agent dynamique 2.
Agent dynamique 3 - passerelle locale	JobManager.ini	Section `[ResourceAdvisorAgent]` **`ResourceAdvisorUrl`**	`https://$(serveur_tdwb):$(port_tdwb)/ita/JobManagerGW/JobManagerRESTWeb/JobScheduler/resource` où, **$(***serveur_tdwb***)** Nom d'hôte du poste de travail de l'agent dynamique 3. **$(***port_tdwb***)** Numéro de port du poste de travail de l'agent dynamique 3.

Pour plus d'informations sur les paramètre des fichiers `JobManager.ini` et `JobManagerGW.ini`, voir «Configuration de l'agent», à la page 52.

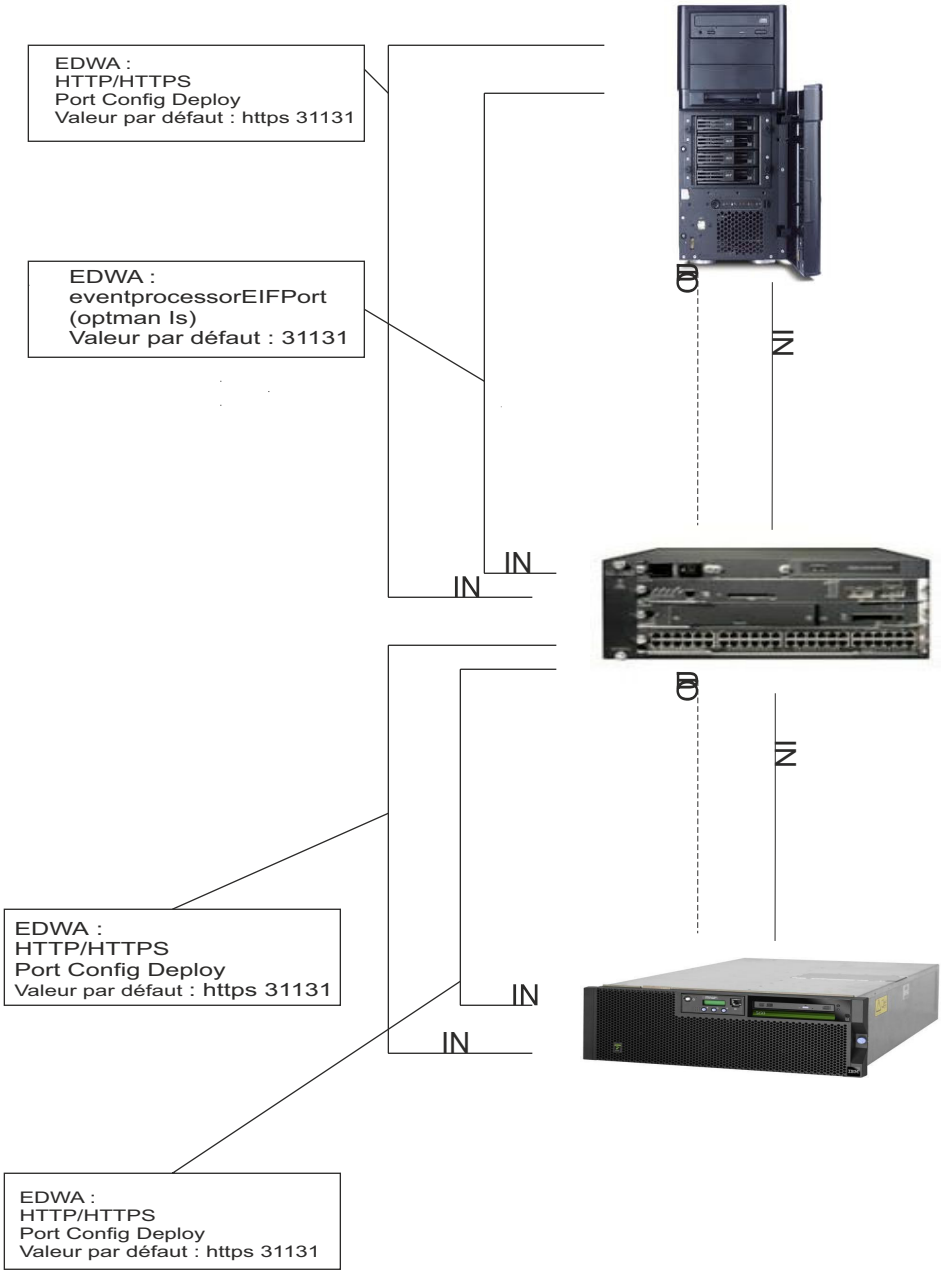
Pour voir un exemple des paramètres d'installation qui doivent être spécifiés pour configurer une passerelle lors de l'installation d'un agent dynamique, consultez la section contenant des exemples d'installation de passerelle d'agent dynamique dans *Guide de planification et d'installation*.

Activation des ports

Lorsque vous installez le gestionnaire de domaine maître dans un réseau Tivoli Workload Scheduler, tous les ports entrants et sortants s'affichent dans la figure ci-dessous :



Si vous activez l'automatisation d'une charge de travail gérée par des événements (EDWA) derrière la fonction de pare-feu, la figure ci-dessous affiche tous les ports entrants et sortants.



Opération réseau

Le processus batchman de chaque gestionnaire de domaine et poste de travail d'agent tolérant aux pannes fonctionne de façon autonome. Il analyse son fichier `Symphony` pour résoudre les dépendances et travaux de lancement. Batchman lance des travaux par le biais du processus jobman. Sur un agent standard, le processus jobman répond aux requêtes de lancement provenant du processus batchman du gestionnaire de domaine.

Le gestionnaire de domaine maître est informé de façon continue des lancements et des arrêts de travaux et est responsable de la diffusion des informations aux gestionnaires de domaines et agents tolérants aux pannes de façon qu'ils puissent résoudre toute éventuelle dépendance entre postes de travail.

Le degré de synchronisation entre les fichiers `Symphony` dépend du paramètre du mode *FullStatus* dans la définition de poste de travail. En supposant que ces modes soient activés, le fichier Symphony d'un agent tolérant aux pannes contient les mêmes informations que celui du gestionnaire de domaine maître (voir la section présentant la gestion des postes de travail dans la base de données dans le *Tivoli Workload Scheduler - Guide d'utilisation et de référence*).

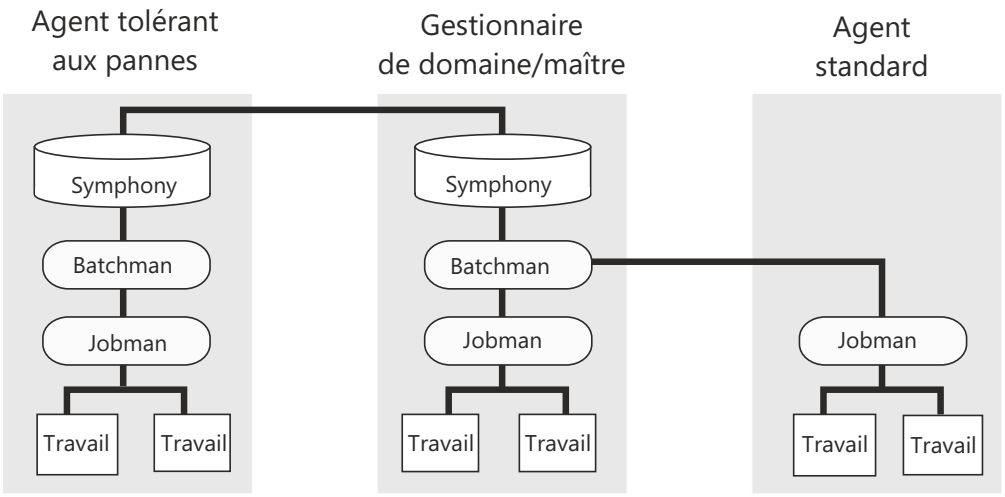


Figure 3. Synchronisation du fichier Symphony

Processus réseau

Netman est démarré par le script **`StartUp`** (commande). Les processus sont créés dans l'ordre suivant : netman, mailman, batchman et jobman. Sur les postes de travail d'agent standard, batchman ne s'exécute pas. Tous les processus, sauf jobman, s'exécutent en tant qu'utilisateur **TWS**. Jobman s'exécute en tant que **root**.

Lorsque l'activité réseau commence, netman reçoit des requêtes de la part de processus mailman distants. Dès réception d'une requête, netman crée un processus d'écriture et lui transmet la connexion. Le programme d'écriture reçoit le message et le transmet au mailman local. Les processus du programme d'écriture (il peut y en avoir plusieurs sur un gestionnaire de domaine) sont démarrés par des demandes de connexion et arrêtés par des demandes de déconnexion (ou lorsque le processus mailman en cours de communication s'arrête).

Les gestionnaires de domaine, y compris le gestionnaire de domaine maître, peuvent communiquer avec un grand nombre d'agents et de gestionnaires de domaine subordonnés. Pour une efficacité accrue, vous pouvez définir les serveurs mailman sur un gestionnaire de domaine pour répartir la charge de communications (voir la section expliquant la gestion des postes de travail dans la base de données dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence*).

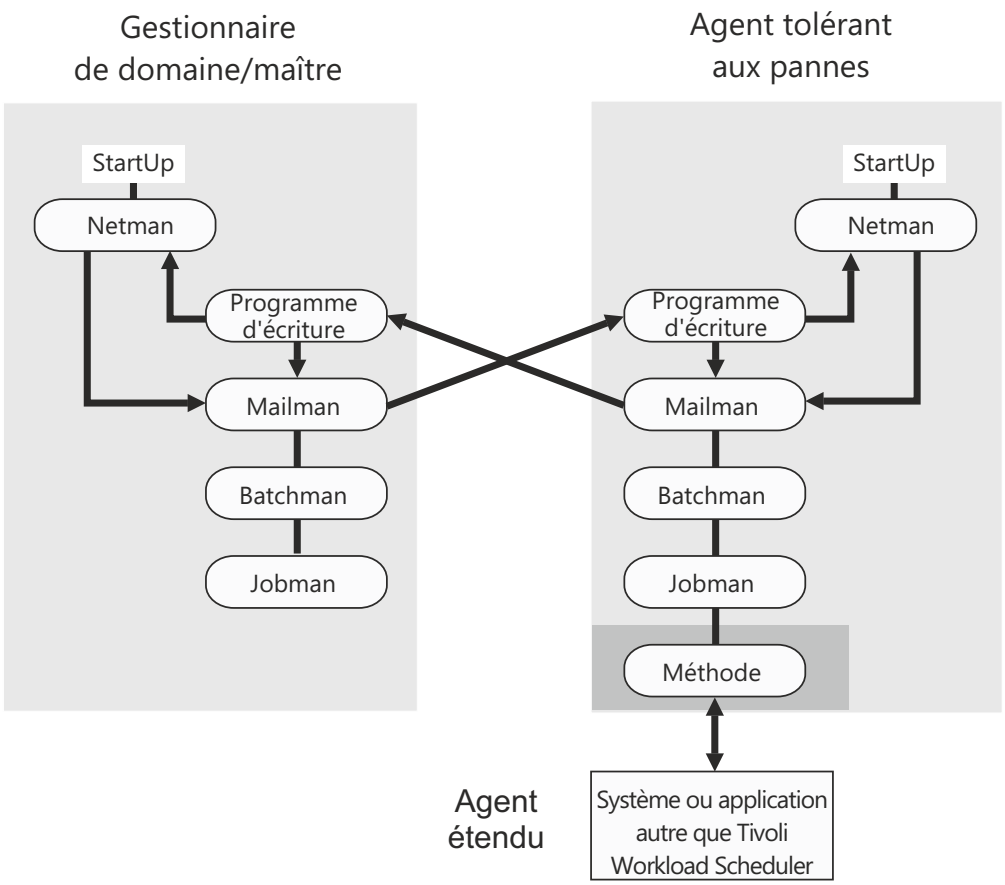


Figure 4. Création de processus dans un gestionnaire de domaine et un agent tolérant aux pannes

La commande **StartUp** s'exécute normalement automatiquement, mais elle peut aussi être exécutée manuellement, comme suit :

StartUp

Démarre le processus **netman** de gestion du réseau Tivoli Workload Scheduler.

Sous Windows, le service **netman** est lancé automatiquement au redémarrage de l'ordinateur. **StartUp** peut être utilisé pour redémarrer le service si celui-ci est arrêté pour une raison quelconque.

Sous UNIX, la commande **StartUp** peut être exécutée automatiquement en l'appelant à partir du fichier `/etc/inittab`, de façon à ce que l'infrastructure WebSphere Application Server et **netman** soient démarrés à chaque fois qu'un ordinateur est redémarré. **StartUp** peut être utilisé pour redémarrer le processus **netman** si celui-ci est arrêté pour une quelconque raison.

Le reste de l'arborescence des processus peut être redémarré avec les commandes

```
conman start
conman startmon
```

Reportez-vous à la documentation relative à **conman** dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence* pour plus d'informations.

Remarque : Si vous lancez la commande StartUp en utilisant un shell distant, le processus netman conserve le shell ouvert sans retourner l'invite. Pour éviter ce problème, modifiez la commande StartUp afin que le processus netman soit appelé à l'arrière-plan, comme suit :

```
# Start netman
/usr/local/TWS851/mae851/TWS/bin/netman&
```

Autorisation

Vous devez avoir un accès ***start*** au poste de travail.

Syntaxe

StartUp [**–v** | **–u**]

Arguments

–v Affiche la version de la commande et quitte l'application.

–u Affiche des informations sur la syntaxe de la commande et quitte l'application.

Exemples

Pour afficher le nom et la version de la commande, exécutez la commande suivante :

```
StartUp -v
```

Pour lancer le processus **netman**, exécutez la commande suivante :

```
StartUp
```

Gestion des processus Tivoli Workload Scheduler

Vous pouvez utiliser l'automatisation EDWA (Event-Driven Workload Automation) pour surveiller le statut des processus réseau et lancer un ensemble prédéfini d'actions au déclenchement d'un ou de plusieurs événements spécifiques. Pour plus d'informations à propos de l'automatisation de charge de travail gérée par les événements, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Vous pouvez surveiller les processus suivants :

- agent
- appservman
- batchman
- jobman
- mailman
- monman
- netman

Le fichier .XML contient la définition d'un exemple de règle d'événement permettant de surveiller le statut des processus spécifiés sur le poste de travail indiqué. Cette règle d'événement appelle le fournisseur d'action MessageLogger pour écrire un message dans un fichier journal figurant dans une base de données d'audit interne. Si la condition décrite dans la règle existe déjà lorsque vous déployez la règle, l'événement associé n'est pas généré. Pour plus d'informations à propos du fournisseur de l'action MessageLogger, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence* :

```
<eventRule name="PROCESSES" ruleType="filter" isDraft="no">
 <eventCondition name="twsProcMonEvt1" eventProvider="TWSApplicationMonitor"
 eventType="TWSProcessMonitor">
   <scope>
    AGENT, BATCHMAN DOWN
   </scope>
   <filteringPredicate>
    <attributeFilter name="ProcessName" operator="eq">
     <value>nom_processus1</value>
    </attributeFilter>
    <attributeFilter name="TWSPath" operator="eq">
     <value>chemin_TWS</value>
    </attributeFilter>
    <attributeFilter name="Workstation" operator="eq">
     <value>workstation_name</value>
    </attributeFilter>
    <attributeFilter name="SampleInterval" operator="eq">
     <value>sample_interval</value>
    </attributeFilter>

   </filteringPredicate>
  </eventCondition>
  <action actionProvider="MessageLogger" actionType="MSGLOG" responseType="onDetection">
   <scope>
    OBJECT=AAAAAAA MESSAGE=TWS PROCESS DOWN: %{TWSPROCMONEVT1.PROCESSNAME}
ON %{TWSPROCMONEVT1.TWSPATH}
   </scope>
   <parameter name="ObjectKey">
    <value>object_key</value>
   </parameter>
   <parameter name="Severity">
    <value>message_severity</value>
   </parameter>
   <parameter name="Message">
    <value>log_message</value>
   </parameter>
  </action>
 </eventRule>
</eventRuleSet>
```

Où :

process_name
: Nom du processus à surveiller. Vous pouvez insérer plusieurs noms de processus, comme suit :

```
<attributeFilter name="ProcessName" operator="eq">
     <value>agent</value>
     <value>batchman</value>
    </attributeFilter>
```

TWS_path
: Répertoire contenant le fichier Symphony et le répertoire `bin`.

workstation_name
: Poste de travail sur lequel est généré l'événement.

sample_interval
: Intervalle, exprimé en secondes, de surveillance de statut de processus.

object_key
: Clé identifiant l'objet auquel se rattache le message.

message_severity
: Gravité du message.

log_message
Message à consigner dans le journal.

Optimisation du réseau

La structure d'un réseau Tivoli Workload Scheduler dépend de la structure du réseau de votre entreprise. La structure des domaines doit refléter la topologie du réseau, de façon à mieux utiliser les canaux de communication disponibles.

Lorsque vous planifiez le réseau Tivoli Workload Scheduler, vous devez tenir compte des éléments suivants :

- Quantités de données
- Connectivité

Quantités de données

La capacité du réseau doit être prévue en fonction de la quantité de données en circulation. Les activités suivantes peuvent générer de très importantes quantités de données :

- Transfert de gros fichiers Symphony.
- Trafic des messages entre le gestionnaire de domaine maître et un agent à *FullStatus*.
- Trafic de messages depuis un gestionnaire de domaine lorsque le domaine a de nombreux agents.
- Utilisation importante de dépendances interréseaux, qui étendent le trafic à l'ensemble du réseau.

Connectivité

Vous devez examiner soigneusement la position sur le réseau des agents les plus critiques. La fiabilité de l'exécution de la charge de travail sur un agent particulier dépend de sa capacité à recevoir un nouveau fichier Symphony au début de la période de production. Si la charge de travail contient de nombreuses dépendances, une connexion fiable au reste du réseau est également requise. Ces facteurs suggèrent que la meilleure solution consiste à placer les agents critiques dans le domaine maître ou à les définir en tant que gestionnaires de domaine immédiatement sous gestionnaire de domaine maître, afin de recevoir leurs fichiers Symphony depuis un ensemble de serveurs mailman dédiés. En outre, il est impératif pour les agents critiques que tout gestionnaire de domaine placé au-dessus d'eux dans l'arborescence soit hébergé sur des systèmes puissants, et dispose d'un système de sauvegarde adapté pour assurer la continuité des opérations en cas d'incident.

Tivoli Workload Scheduler propose deux mécanismes permettant de répondre à une situation réseau particulière : la structure du domaine et les serveurs mailman. La structure de domaine établit une hiérarchie parmi les agents Tivoli Workload Scheduler, et les serveurs mailman sont utilisés pour ajuster les ressources dédiées à la connexion entre deux agents.

Domaine
Utilisez le mécanisme de structure du domaine Tivoli Workload Scheduler pour créer une structure en arborescence pour le réseau, dans laquelle toutes les communications entre deux points utilisent le chemin unique défini par l'arborescence (remontant à l'ancêtre commun et descendant à la cible, par opposition aux communications TCP directes). Par conséquent, la structure du domaine sépare le réseau en parties mieux gérables. Elle

simplifie ainsi le filtrage, la vue générale, les actions et la surveillance. Par contre, elle entraîne un certain retard dans le traitement de la charge de travail. Par exemple, lors de la distribution du fichier Symphony, un agent tolérant aux pannes appartenant au domaine doit attendre que deux étapes de la distribution Symphony soient terminées (du gestionnaire de domaine maître vers le gestionnaire de domaine et du gestionnaire de domaine à l'agent tolérant aux pannes). Ce phénomène s'applique également à tout autre type de communication provenant du gestionnaire de domaine maître.

Ceci a les conséquences suivantes :

- Les activités métier critiques doivent être le plus près possible du gestionnaire de domaine maître
- Le gestionnaire de domaine doit être installé sur le poste de travail le plus puissant possible
- Un gestionnaire de domaine de secours de puissance similaire doit être inclus au réseau
- La liaison réseau entre le gestionnaire de domaine et son gestionnaire de secours doit être le plus rapide possible, pour transmettre toutes les mises à jour reçues de l'arborescence
- Si une intervention est requise directement dans le domaine, accordez aux opérateurs un accès au shell pour utiliser la ligne de commande Tivoli Workload Scheduler, ou installez un connecteur de manière à pouvoir utiliser Dynamic Workload Console.

Serveurs Mailman

Les serveurs Mailman allouent des processus distincts dédiés aux communications avec les autres postes de travail. Le mailman principal est réservé au transfert et aux activités du concentrateur réseau. L'utilisation de serveurs mailman sur le gestionnaire de domaine doit être soigneusement planifiée. Le facteur principal est le nombre de connexions en aval à chaque niveau de l'arborescence. Il s'agit du nombre de serveurs mailman auxquels un mailman principal est connecté, ou du nombre d'agents auxquels un serveur mailman est relié. Le nombre maximum de connexions en aval est d'environ 20 pour Solaris, 50 pour Windows et environ 100 pour les autres postes de travail UNIX, selon leur puissance. En général, le nombre de connexions en aval est d'environ 10 pour Solaris, 15 pour Windows et 20 pour les autres postes de travail sous UNIX. Toutefois, vous devez également tenir compte de la vitesse de la connexion et de la taille des files d'attente, abordées ci-dessous.

Planification de la capacité des files d'attente

Pour prévoir la place pour les files d'attente d'événements, ainsi que les niveaux d'alerte et réactions possibles, vous devez modéliser les flots qui traversent les agents, et en particulier les gestionnaires de domaine.

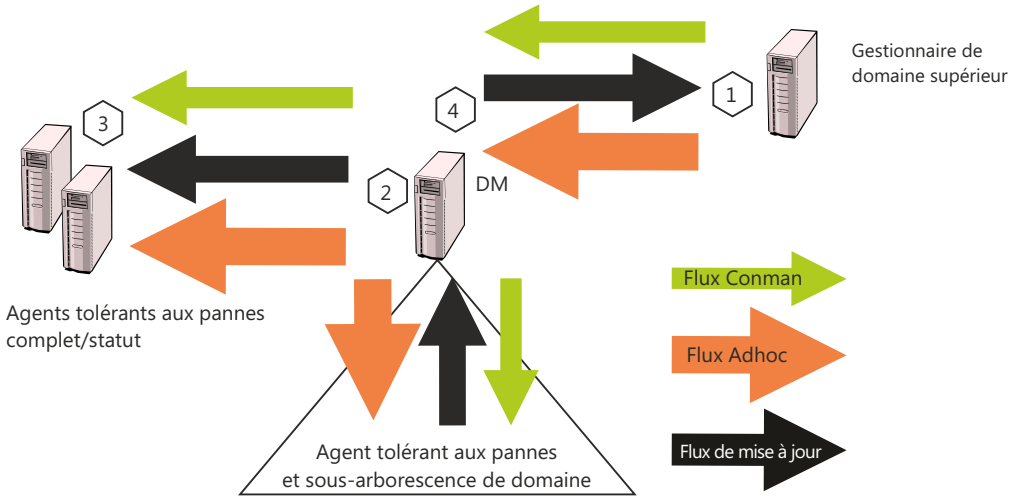


Figure 5. Flots réseau Tivoli Workload Scheduler classiques.

Pour un gestionnaire de domaine classique, le flot principal provient de l'activité de mise à jour rapportée par l'arborescence secondaire, et les soumissions ad hoc émises par le gestionnaire de domaine maître et diffusées sur tout le réseau. Dans ces conditions, les erreurs les plus critiques sont listées par ordre d'importance dans le tableau 49 :

Tableau 49. Erreurs de flux critiques

Flot num.	Emplacement	File d'attente	Risque	Incidence
1	Gestionnaire de domaine principal	dm.msg	La file d'attente se remplit en raison d'un trop grand nombre de postes de travail non connectés ou de l'échec d'un gestionnaire de domaine en aval.	Le gestionnaire de domaine principal échoue et propage l'erreur.
2	Gestionnaire de domaine	fta.msg de *FullStatus*	La file d'attente se remplit car le nombre de postes de travail non connectés dans le domaine est trop élevé ou parce que l'agent tolérant aux pannes *FullStatus* est submergé par le flot.	Le gestionnaire de domaine échoue et favorise les occurrences du cas num. 1.
3	Gestionnaire de domaine et agent tolérant aux pannes à *FullStatus*	Mailbox.msg ou Intercom.msg	La file d'attente se remplit car le *FullStatus* agent tolérant aux pannes est submergé par le flot.	Le *FullStatus* agent tolérant aux pannes échoue et favorise l'occurrence du cas N° 2.
4	Gestionnaire de domaine	tomaster.msg	La file d'attente se remplit en raison d'un trop grand nombre de postes de travail non connectés dans le domaine.	Le gestionnaire de domaine commence à déconnecter l'arborescence secondaire et accumule les messages dans la structure.
5	Agents tolérants aux pannes - uniquement lorsque l'option globale `enSwfaultTol` est définie sur *yes*	deadletter.msg	La file d'attente se remplit en raison d'un trop grand nombre de postes de travail non connectés dans le domaine.	L'agent s'arrête.

Tableau 49. Erreurs de flux critiques (suite)

Flot num.	Emplacement	File d'attente	Risque	Incidence
6	Agents tolérants aux pannes - uniquement lorsque l'option globale `enSwfaultTol` est définie sur *yes*	ftbox.msg	La file d'attente est circulaire. Les messages rentrent dans la file plus vite qu'ils ne sont traités, en raison d'un trop grand nombre de postes de travail non connectés dans le domaine.	Des événements sont perdus.

Remarque :

1. Les flots sont plus importants au niveau du gestionnaire de domaine maître et de tout agent *FullStatus* tolérant aux pannes dans le domaine maître qu'au niveau des gestionnaires de domaines subordonnés ou des *FullStatus* agent tolérant aux pannes.
2. Utilisez la commande **`evtsize -show`** pour surveiller la longueur des files d'attente.
3. La taille du flot de mise à jour est liée à la charge de travail d'une arborescence secondaire particulière, ce qui est inévitable.
4. La taille du flot ad hoc est liée à la taille de la charge de travail supplémentaire en tout point du réseau. Vous pouvez la réduire en planifiant une charge de travail supérieure, même si elle est inactive. Notez que de simples réexécutions (et non des commandes `rerun from`) ne créent pas un flot ad hoc.

La stratégie de planification, d'alerte et de récupération doit tenir compte des points suivants :

- Les fichiers d'attente sont créés avec une taille fixe et les messages sont ajoutés et supprimés de façon cyclique. La capacité d'une file d'attente est atteinte lorsque le flux des messages entrants dépasse le flux sortant pendant une durée suffisante pour utiliser tout l'espace disponible. Par exemple, si des messages sont ajoutés à une file d'attente à une vitesse d'1 Mo par unité de temps, et traités et supprimés à une vitesse de 0,5 Mo par unité de temps, une file d'attente de 10 Mo (valeur par défaut) est la capacité atteinte au bout de 20 unités de temps. Mais si la vitesse du flot entrant diminue jusqu'à égaler celle du flot sortant après 19 unités de temps, la file d'attente n'atteint pas sa capacité maximale.
- Vous pouvez limiter le risque d'échec du gestionnaire de domaine en basculant sur le gestionnaire de domaine de secours. Dans ce cas, le contenu des files d'attente du gestionnaire de domaine est indisponible tant que le gestionnaire de domaine de secours n'est pas démarré. Dans tous les cas, la longueur de la file d'attente du gestionnaire de domaine principal, vers n'importe quel autre gestionnaire de domaine, doit respecter la condition A du tableau 50, à la page 246.
- Vous devez tenir compte à l'avance du risque que les agents tolérants aux pannes et à bascule sur panne ne puissent pas prendre en charge le flot. Les spécifications relatives aux agents tolérants aux pannes et à bascule sur panne doivent être les mêmes que pour le gestionnaire de domaine, de façon à éviter que l'agent ne reçoive une charge supérieure à sa capacité. Surveillez la formation d'une file d'attente au niveau des agents tolérants aux pannes à *FullStatus*, lors des situations normales comme des pics d'activité.
- Une fois le risque num. 2 écarté, la possibilité d'un incident de connexion réseau peut être limitée en dimensionnant correctement la file d'attente d'un gestionnaire de domaine vers les agents tolérants aux pannes à *FullStatus*, en

fonction de la durée moyenne d'indisponibilité réseau et en augmentant la taille de la boîte aux lettres en cas de longue indisponibilité imprévue (voir condition B du tableau 50).

- La même condition s'applique pour éviter le dépassement de capacité de la file d'attente tomaster.msg du gestionnaire de domaine en cas de coupures réseau (voir condition C du tableau 50).

Tableau 50. Conditions de dimensionnement de la file d'attente

A	MaxAlertTime <= size(UpperDM#queueToDM) / averageAdhocFlow
B	MaxNetOutage <= size(DM#queueToFSFTA) / (averageAdhocFlow + averageUpdateFlow)
C	MaxNetOutage <= size(DM#queueToUpperDM) / (averageUpdateFlow)

Gestion des files d'attente de messages Tivoli Workload Scheduler

Vous pouvez utiliser l'automatisation EDWA (Event-Driven Workload Automation) pour surveiller la taille des files d'attente de messages et lancer un ensemble prédéfini d'actions au déclenchement d'un ou de plusieurs événements spécifiques. Pour plus d'informations à propos de l'automatisation de charge de travail gérée par les événements, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Vous pouvez surveiller les files d'attente de messages suivantes :

- appserverbox
- mailbox
- clbox
- intercom
- courier
- monbox
- moncmd
- server
- tomaster
- pobox
- planbox

Le fichier .XML suivant contient la définition d'un exemple de règle d'événement permettant de surveiller la file d'attente de la boîte aux lettres sur le poste de travail indiqué et d'envoyer un e-mail lorsque le pourcentage de remplissage dépasse la valeur spécifiée. Si la condition décrite dans la règle existe déjà lorsque vous déployez la règle, l'événement associé n'est pas généré. Cette règle d'événement appelle le fournisseur d'action MailSender pour envoyer un e-mail aux destinataires que vous spécifiez. Pour plus d'informations à propos du fournisseur de l'action MailSender, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence* :

```
<?xml version="1.0"?>
<eventRuleSet xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://www.ibm.com/xmlns/prod/tws/1.0/event-management/rules"
  xsi:schemaLocation="http://www.ibm.com/xmlns/prod/tws/1.0/event-management/rules
   http://www.ibm.com/xmlns/prod/tws/1.0/event-management/rules/EventRules.xsd">
 <eventRule name="MONITORQUEUE" ruleType="filter" isDraft="no">
  <eventCondition name="twsMesQueEvt1" eventProvider="TWSApplicationMonitor" eventType="TWSMessageQueues">
   <scope>
    MAILBOX FILLED UP 80% ON FTA
   </scope>
```

```
   <filteringPredicate>
    <attributeFilter name="MailboxName" operator="eq">
     <value>mailbox_name</value>
    </attributeFilter>
    <attributeFilter name="FillingPercentage" operator="ge">
     <value>filling_percentage</value>
    </attributeFilter>
    <attributeFilter name="Workstation" operator="eq">
     <value>workstation_name</value>
    </attributeFilter>
    <attributeFilter name="SampleInterval" operator="eq">
     <value>sample_interval</value>
    </attributeFilter>
   </filteringPredicate>
  </eventCondition>
  <action actionProvider="MailSender" actionType="SendMail" responseType="onDetection">
   <scope>
    TWSUSER@TWS : THE MAILBOX ON workstation_name...
   </scope>
   <parameter name="To">
    <value>main_receiver_list</value>
   </parameter>
   <parameter name="Subject">
    <value>mail_subject</value>
   </parameter>
  </action>
 </eventRule>
</eventRuleSet>
```

Où :

mailbox_name
Nom de la boîte aux lettres à surveiller.

filling_percentage
Pourcentage de remplissage. Les opérateurs pris en charge sont les suivants :

ge déclenche la génération d'événement lorsque le pourcentage de remplissage de la boîte aux lettres dépasse la valeur de seuil. Cet événement est généré uniquement la première fois que le pourcentage de remplissage de la boîte aux lettres indiqué est atteint. Si vous redémarrez l'agent SSM et que le pourcentage de remplissage dépasse la valeur de seuil, l'événement est généré une nouvelle fois. Le tableau 51 fournit un exemple dans lequel l'opérateur **ge** est défini avec la valeur 70 %.

Tableau 51. Exemple d'opérateur ge

Nom de la boîte aux lettres	Pourcentage de remplissage	Action
Sample (0)	>= 70 %	événement non généré
Sample (0)	< 70 %	événement non généré
Sample (n-1)	< 70 %	événement non généré
Sample (n)	>= 70 %	événement généré
Sample (n+1)	>= 70 %	événement non généré

le déclenche la génération d'événement lorsque le pourcentage de remplissage de la boîte aux lettres est inférieur à la valeur de seuil. Cet événement est généré uniquement la première fois que le pourcentage de remplissage de la boîte aux lettres indiqué est atteint. Si vous redémarrez l'agent SSM et que le pourcentage de

remplissage est inférieur à la valeur de seuil, l'événement ne sera généré que lorsque le pourcentage de remplissage dépassera la valeur de seuil, puis repassera à nouveau au-dessous de la valeur de seuil. Le tableau 52 fournit un exemple dans lequel l'opérateur **le** est défini avec la valeur 50 % :

Tableau 52. Exemple d'opérateur le

Nom de la boîte aux lettres	Pourcentage de remplissage	Action
Sample (0)	<= 50 %	événement non généré
Sample (0)	> 50 %	événement non généré
Sample (n-1)	> 50 %	événement non généré
Sample (n)	<= 50 %	événement généré
Sample (n+1)	<= 50 %	événement non généré

workstation_name
Poste de travail sur lequel est généré l'événement.

sample_interval
Intervalle, exprimé en secondes, de surveillance du pourcentage de remplissage de la boîte aux lettres.

main_receiver_list
Liste des principaux destinataires.

mail_subject
Sujet du message.

Modification de la taille d'une file d'attente

Utilisez la commande **evtsize** pour redimensionner une file d'attente.

Lorsque vous avez utilisé **evtsize** pour redimensionné une file d'attente, cette dernière conservera cette taille jusqu'à la prochaine utilisation de **evtsize**. Elle ne retrouve sa taille par défaut de 10 Mo que si vous la supprimez, auquel cas Tivoli Workload Scheduler la recrée avec la taille par défaut.

evtsize :

Définit la taille des fichiers message du Tivoli Workload Scheduler. Cette commande est utilisée par l'administrateur de Tivoli Workload Scheduler pour augmenter la taille d'un fichier de messages après réception du message “End of file on events file”ou pour surveiller la taille de la file d'attente de messages contenus dans le fichier message.

Autorisation

Vous devez être un utilisateur **maestro** ou **root** sous UNIX, ou **Administrateur** sous Windows pour exécuter **evtsize**. Arrêtez le moteur IBM Tivoli Workload Scheduler avant d'exécuter cette commande.

Syntaxe

evtsize -V | -U

evtsize *file_name taille*

evtsize -compact *file_name* [*taille*]

evtsize -show *file_name*

Arguments

-V Affiche la version de la commande et quitte l'application.

-U Affiche des informations sur la syntaxe de la commande et quitte l'application.

-compact *file_name* **[***size***]**
Réduit la taille du fichier de messages pour revenir à la taille des messages présents au moment où vous avez exécuté la commande. Vous pouvez éventuellement utiliser ce mot clé pour indiquer également une nouvelle taille de fichier.

-show *file_name*
Affiche la taille de la file d'attente de messages contenue dans le fichier de messages.

file_name
Nom du fichier des événements. Indiquez l'un des arguments suivants :

```
Courier.msg
Intercom.msg
Mailbox.msg
PlanBox.msg
Server.msg
pobox/workstation.msg
```

taille Taille maximale, en octets, du fichier des événements. Lors de la création du fichier par Tivoli Workload Scheduler, la taille maximale est définie sur 10 Mo.

Remarque : La taille du fichier de messages est supérieure ou égale à la taille réelle de la file d'attente de messages qu'il contient et elle augmente progressivement jusqu'à ce que la file d'attente de messages soit vide ; au moment où cela se produit, le fichier de messages est vidé.

Exemples

Pour définir la taille maximale du fichier `Intercom.msg` à 20 Mo, exécutez la commande suivante :

```
evtsize Intercom.msg 20000000
```

Pour définir la taille maximale du fichier `pobox` du poste de travail `chicago` à 15 Mo, exécutez la commande suivante :

```
evtsize pobox\chicago.msg 15000000
```

La commande suivante :

```
evtsize -show Intercom.msg
```

renvoie le résultat suivant :

```
Tivoli Workload Scheduler (UNIX)/EVTSIZE 8.3 (1.2.2.4) Eléments sous licence -
Propriété d'IBM(R)
5698-WSH
(C) Copyright IBM Corp 1998, 2006 All rights reserved.
US Government User Restricted Rights
Use, duplication or disclosure restricted by
GSA ADP Schedule Contract with IBM Corp.
```

```
IBM est une marque d'International Business Machines
Corporation aux Etats-Unis et/ou dans certains autres pays.
AWSDEK703I Taille de la file d'attente courant 240, 10000000 octets maximum (48 en lecture, 288 en é
```

Où :

240 représente la taille de la file d'attente actuelle du fichier `Intercom.msg`

10000000
représente la taille maximale du fichier `Intercom.msg`

48 en lecture
Indique la position du pointeur pour lire les enregistrements

288 en écriture
Indique la position du pointeur pour écrire les enregistrements

Optimisation des serveurs mailman

Une fois les agents distribués sur les serveurs mailman, tous les groupes d'agents connectés au même serveur doivent respecter la condition de liaison.

La condition de liaison met en relation le nombre d'agents connectés à un processus mailman et les paramètres d'optimisation, pour supprimer la liaison au niveau de mailman et du programme d'écriture.

No_agents(i)
Nombre d'agents connectés à un serveur mailman donné *i*

Mm_unlink
Paramètre défini dans le `localopts` du gestionnaire de domaine et de l'agent. Indique le nombre maximal de secondes pendant lesquelles mailman attend avant de supprimer les liaisons avec un poste de travail qui ne répond pas.

Wr_unlink
Paramètre défini dans le `localopts` du gestionnaire de domaine et de l'agent. Nombre de secondes pendant lequel le processus d'écriture attend avant de s'arrêter si aucun message entrant n'est reçu.

Max_down_agents
Nombre maximal probable d'agents indisponibles sans que l'indicateur `ignore` ne soit défini dans la base de données et que l'indicateur `autolink` ne soit activé.

tcp timeout
Paramètre défini dans le `localopts` du gestionnaire de domaine et de l'agent. Spécifiez le nombre maximum de secondes d'attente d'achèvement d'une requête TCP/IP sur un poste de travail connecté qui ne répond pas.

La condition est :

```
Wr_unlink = Mm_unlink > 1.2 * Max_down_agents * tcp timeout
```

Cette condition indique que si le temps écoulé avant la suppression de la liaison est plus court que le temps d'attente probable du processus mailman (attente du délai de connexion de chaque agent arrêté) de sa boucle pour réactiver les connexions, les agents se déconnectent constamment lorsque certains d'entre eux sont arrêtés.

Fichier de configuration Netman

Le fichier de configuration netman est présent sur tous les postes de travail Tivoli Workload Scheduler, pour définir les services fournis par netman. Il est nommé *<TWA_home>*/TWS/network/Netconf. Le fichier NetConf contient des commentaires décrivant chaque service. Les services sont les suivants :

2001 Démarrage d'un processus d'écriture pour gérer les messages entrants provenant d'un mailman distant.

2002 Démarrage d'un processus mailman. Celui-ci démarre à son tour le reste de l'arborescence de processus (batchman, jobman).

2003 arrêt du processus Tivoli Workload Scheduler pour la gestion des messages entrants provenant d'un mailman distant.

2004 Localisation et envoi d'un fichier stdlist au processus Conman demandeur.

2005 Basculement du gestionnaire de domaine dans un domaine.

2006 Téléchargement local de scripts planifiés par un Tivoli Workload Scheduler pour desz/OS gestionnaire de domaine maître.

2007 Requis pour contourner un pare-feu.

2008 Arrêt des postes de travail Tivoli Workload Scheduler de manière hiérarchique

2009 Exécution du script switchmgr pour arrêter et redémarrer un gestionnaire de façon à ce qu'il n'ouvre pas de liaison vers d'autres postes de travail tant qu'il n'a pas reçu l'événement *switchmgr*. Utilisable uniquement lorsque l'option globale enSwfaultTol est définie sur *yes*.

2010 Démarrage de mailman avec le paramètre *demgr*. Utilisé par le service *2009*. Utilisable uniquement lorsque l'option globale enSwfaultTol est définie sur *yes*.

2011 Exécute **monman** en tant que processus enfant (son bin/monman.exe)

2012 Exécute **conman** pour arrêter le moteur de surveillance des événements command bin/conman.exe stopmon).

2013 Exécute **conman** pour basculer les processeurs d'événement (commande bin/conman.exe switchevtproc -this)

2014 Exécute **conman** pour démarrer le traitement d'événement command bin/conman.exe startevtproc -this)

2015 Exécute **conman** pour arrêter le traitement d'événement commande bin/conman.exe stopevtproc -this)

2016 Exécute **conman** pour forcer la mise à jour du fichier de configuration de surveillance pour le moteur de surveillance d'événement command bin/conman.exe deployconf)

2017 Exécute **conman** pour arrêter le traitement d'événement sur un client (client bin/conman.exe synchronizedcmd -stopevtproc)

2018 Exécute **conman** pour vérifier le traitement d'événement sur un client (client bin/conman.exe synchronizedcmd -checkevtproc)

2021 Exécute **conman** pour démarrer appservman

2022 Exécute **conman** pour exécuter la commande secondaire **stopappserver** qui arrête le serveur d'applications.

2023 Exécute **conman** pour exécuter la commande secondaire **startappserver** qui démarre le serveur d'applications

2501 Vérification du statut d'un travail distant.

2502 Démarrage du gestionnaire console –, un service demandé par le côté client de la console distante. Pour plus d'informations, voir *IBM Tivoli Remote Control - Guide d'utilisation*.

2503 Utilisé par le connecteur pour interagir avec l'agent étendu r3batch.

Détermination de la taille de la table Symphony interne

Le service mailman (2002) peut accepter un paramètre facultatif déterminant la taille initiale de la table Symphony interne. Si vous n'indiquez pas ce paramètre, mailman calcule la taille initiale de la table en fonction du nombre d'enregistrements contenus dans le fichier.

Remarque : Mailman permet de développer la table le cas échéant, même si ce paramètre n'est pas fourni.

Dans des circonstances normales, quittez mailman pour prendre la valeur par défaut figurant dans le fichier `NetConf` fourni (32000). En cas de problème de mémoire, vous pouvez affecter une table plus petite. Pour ce faire, vous modifiez le paramètre sur le service *2002* dans le fichier `NetConf`. Le syntaxe de l'entrée est :

```
2002     son     bin/mailman [ -parm <number> ]
```

où <*number*> permet de calculer la taille initiale de la table Symphony en fonction du nombre d'enregistrements présents dans le fichier Symphony.

Si *r* est le nombre d'enregistrements présents dans le fichier Symphony au démarrage de batchman, le tableau 53 présente le calcul de la taille interne Symphony, selon la valeur de <*number*> :

Tableau 53. Calcul de la table Symphony interne

Valeur de <*number*>	**Taille de la table**
0	(4/3r) + 512
n	if n > r, n si n <= r, (4/3r) + 512
-1	65535
-n	if +n => r, n si +n < r, r + 512

Si en période de production vous ajoutez des fichiers, la taille maximale de la table Symphony interne augmente de façon dynamique jusqu'à atteindre le nombre maximal d'enregistrements autorisé dans le fichier Symphony, qui est de 2.000.000.000.

Définition des méthodes d'accès pour des agents

Les méthodes d'accès permettent d'étendre les fonctions de planification des travaux de Tivoli Workload Scheduler à d'autres systèmes et applications. Elles s'exécutent sous :

Agents étendus
Il s'agit de postes de travail logiques liés à une méthode d'accès hébergée par un poste de travail Tivoli Workload Scheduler physique (et non un autre agent étendu). Plusieurs postes de travail d'agent étendu peuvent être hébergés par un même poste de travail Tivoli Workload Scheduler et utiliser la même méthode d'accès. L'agent étendu s'exécute sur des agents tolérants aux pannes définis à l'aide d'une définition de poste de travail Tivoli Workload Scheduler standard, laquelle donne un nom à l'agent étendu et identifie la méthode d'accès. La méthode d'accès est un programme exécuté par le poste de travail hôte chaque fois que Tivoli Workload Scheduler soumet un travail à un système externe.

Des travaux sont définis pour un agent étendu de la même manière que pour les autres postes de travail Tivoli Workload Scheduler, sauf que les attributs des travaux sont dictés par l'application ou le système externe.

Les informations relatives à l'exécution d'un travail sont envoyées à Tivoli Workload Scheduler à partir d'un agent étendu à l'aide du fichier `stdlist` du travail. Un fichier d'options de méthode peut spécifier des connexions de remplacement pour lancer des travaux et vérifier les dépendances de fichier *opens*. Pour plus d'informations, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Un poste de travail physique peut héberger 255 agents étendus au maximum.

agents dynamiques et agents Tivoli Workload Scheduler for z/OS
Ils communiquent avec des systèmes externes pour démarrer le travail et renvoyer son état. Pour exécuter des méthodes d'accès sur des applications externes à l'aide des agents dynamiques, vous définissez un travail de type **méthode d'accès**.

Les méthodes d'accès sont disponibles sur les systèmes et applications suivants.
- Oracle E-Business Suite
- SAP R/3
- z/OS
- Méthodes personnalisées
- unixssh
- unixrsh
- UNIX local (agents tolérants aux pannes uniquement)

Les méthodes d'accès UNIX, inclus avec Tivoli Workload Scheduler, sont décrits dans le fichier «Méthodes d'accès UNIX».

Si vous travaillez avec des agents dynamiques, pour obtenir des informations sur la définition des postes de travail Tivoli Workload Scheduler, voir la section qui explique comment définir des postes de travail dans la base de données dans le manuel *Tivoli Workload Scheduler - Guide d'utilisation et de référence*. Pour plus d'informations sur la rédaction des méthodes d'accès, voir la section relative à l'interface de méthode d'accès dans le manuel *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Méthodes d'accès UNIX

Tivoli Workload Scheduler inclut deux types de méthodes d'accès UNIX, les méthodes d'accès UNIX locales et les méthodes d'accès UNIX distantes.

Les méthodes d'accès UNIX locales s'exécutent sur des agents étendus. Utilisez la méthode d'accès UNIX locale pour activer un poste de travailUNIX unique afin qu'il fonctionne comme deux postes de travail Tivoli Workload Scheduler, tous deux étant en mesure d'exécuter des travaux planifiés Tivoli Workload Scheduler.

La méthode d'accès UNIX distante s'exécute sur des agents étendus et des agents dynamiques.

Sur les agents étendus
Utilisez la méthode d'accès UNIX à distance pour désigner un poste de travail UNIX distant afin qu'il exécute les travaux planifiés de Tivoli Workload Scheduler sans que Tivoli Workload Scheduler ne soit installé dessus.

Sur les agents dynamiques
Définissez un travail de type **xajob** qui s'exécute sur des agents dynamiques. L'agent dynamique communique avec le système externe pour démarrer le travail et renvoyer son état.

Méthode d'accès UNIX locale exécutée sur des agents tolérants aux pannes uniquement

La méthode d'accès UNIX locale peut être utilisée pour définir plusieurs postes de travail Tivoli Workload Scheduler sur un même poste de travail : le poste de travail hôte et un ou plusieurs agents étendus. Lorsque Tivoli Workload Scheduler envoie un travail à un agent étendu UNIX local, la méthode d'accès **unixlocl** est appelée par l'hôte pour exécuter le travail. La méthode commence par exécuter le script de configuration standard sur le poste de travail hôte (*<TWA_home>*/TWS/jobmanrc). Si l'utilisateur de connexion du travail est autorisé à utiliser un script de configuration local et que le script existe en tant que $HOME/TWS/.jobmanrc, le script de configuration local est également exécuté. Le travail lui-même est ensuite exécuté par le script de configuration standard ou local. Si aucun de ces deux scripts n'existe, la méthode démarre le travail.

Le lancement des scripts de configuration jobmanrc et .jobmanrc est paramétrable dans le script de méthode. La méthode exécute les scripts de configuration par défaut, s'ils existent. Pour désactiver cette fonction, vous devez mettre en commentaire plusieurs lignes du script de méthode. Pour plus d'informations, reportez-vous au fichier de script *<TWA_home>*/TWS/methods/unixlocl sur l'hôte de l'agent étendu.

Méthode d'accès UNIX distante

La méthode d'accès UNIX à distance peut être utilisée pour désigner un poste de travail non Tivoli Workload Scheduler pour qu'il exécute les travaux planifiés par Tivoli Workload Scheduler. Vous pouvez utiliser unixrsh ou unixssh :

Méthode d'accès unixrsh
Lorsque Tivoli Workload Scheduler envoie un travail à un agent étendu UNIX distant, la méthode d'accès, unixrsh, crée un répertoire /tmp/maestro sur le poste de travail non Tivoli Workload Scheduler. Elle transfère ensuite un script d'encapsuleur dans le répertoire et l'exécute. L'encapsuleur exécute ensuite le travail planifié. L'encapsuleur n'est créé qu'une seule fois, à moins qu'il ne soit supprimé, déplacé ou obsolète.

Pour exécuter des travaux à l'aide de la méthode d'accès unixrsh, les utilisateurs de connexion du travail doivent disposer des droits d'accès adéquats au poste de travail non-Tivoli Workload Scheduler UNIX. Pour offrir un accès approprié, un fichier .rhost, /etc/host.equiv ou un fichier équivalent doit être défini sur le poste de travail. Sur des agents étendus, si

vous devez aussi vérifier les dépendances de fichier *opens*, l'accès *root* doit également être autorisé. Pour obtenir de l'aide, contactez votre administrateur système. Pour plus d'informations à propos de la méthode d'accès, consultez le fichier de script <*TWA_home*>/TWS/methods/unixrsh sur l'hôte d'un agent étendu.

Méthode d'accès unixssh

La méthode d'accès unixssh fonctionne comme unixrsh mais utilise un shell éloigné sécurisé pour se connecter à l'hôte éloigné. Les fichiers utilisés par cette méthode sont les suivants :

```
methods/unixssh
methods/unixssh.wrp
```

La méthode unixssh utilise la clé *ssh*. Vous pouvez le générer avec n'importe quel outil compatible avec le shell distant sécurisé.

Remarque : La phrase passe doit être vide.
Le scénario suivant explique comment configurer la méthode :

Vous avez installé Tivoli Workload Scheduler, un agent tolérant aux pannes ou un agent dynamique avec l'*utilisateur_TWS* : twsuser. Vous voulez exécuter un shell distant sur l'hôte distant "REMOTE_HOST" avec l'utilisateur "guest". La procédure est la suivante :

1. Créez la clé publique et la clé privée pour l'utilisateur twsuser ; exemple avec utilisation de rsa :
   a. Connectez-vous en tant que twsuser
   b. Exécutez
      ```
      ssh-keygen -t rsa
      ```
   c. Lorsque l'outil demande la phrase passe, appuyez sur Entrée (laissez la phrase passe vide). Les clés sont enregistrées de la façon suivante :

Clé	Emplacement	Commentaire
Publique	<*TWA_home*>/TWS/.ssh/id_rsa.pub	
Privée	<*TWA_home*>/TWS/.ssh/id_rsa	N'envoyez pas ce fichier !

   Remarque : Chaque outil conserve la clé dans un emplacement qui lui est propre.
2. Sur l'hôte distant, effectuez les actions suivantes :
   a. Exécutez une commande Telnet.
   b. Connectez-vous en tant que "guest".
   c. Accédez au répertoire .ssh du répertoire principal de l'utilisateur ou créez-le s'il n'existe pas (les droits d'accès aux répertoire appropriés doivent être définis, par exemple, 700 pour le répertoire et 600 pour son contenu).
   d. Ajoutez la clé *publique* créée à l'étape 1 au fichier authorized_keys (créez le fichier s'il n'existe pas), à l'aide de la commande :
      ```
      cat id_rsa.pub >> authorized_keys
      ```
3. Sur l'agent tolérant aux pannes ou l'agent dynamique, faites "connaître" l'hôte distant avant de permettre aux processus Tivoli Workload Scheduler d'utiliser la connexion. Pour ce faire, utilisez l'une des deux méthodes suivantes :

- Ouvrez une session en tant que `twsuser` et connectez-vous à l'hôte à l'aide de la commande suivante :

  ```
  ssh -l guest <hostname_distant> ls
  ```

  Une invite s'affiche, indiquant que l'hôte est inconnu et demandant le droit d'y accéder. Accordez le droit et l'hôte s'ajoute à la liste des hôtes connus.
- Vous pouvez aussi utiliser la documentation ssh pour ajouter l'hôte distant au fichier des hôtes connus.

Gestion de la production pour les agents étendus

En général, les travaux exécutés sur les agents étendus ont le même comportement que les autres travaux Tivoli Workload Scheduler. Tivoli Workload Scheduler suit le statut d'un travail et enregistre la sortie dans les fichiers `stdlist` du travail. Ces fichiers sont stockés sur le poste de travail de l'agent étendu *host*. Pour plus d'informations à propos de la gestion des travaux, reportez-vous à la section qui décrit les tâches du plan Tivoli Workload Scheduler dans le *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Echec lors du lancement de travaux sur des agents étendus et des agents dynamiques

Si la méthode d'accès n'est pas dans le répertoire adéquat de l'hôte de l'agent étendu, sur l'agent dynamique, ou si la méthode n'est pas accessible par Tivoli Workload Scheduler, le lancement des travaux échoue ou une dépendance de fichier n'est pas vérifiée. Pour un travail, la connexion des travaux Tivoli Workload Scheduler ou la connexion spécifiée dans le fichier des options de la méthode doit disposer de droits de lecture et d'exécution relatifs à la méthode d'accès. Lorsque vous vérifiez un fichier pour satisfaire une dépendance *opens*, la connexion s'effectue en tant que root à moins qu'une autre connexion ne soit spécifiée dans le fichier des options de méthode. Pour plus d'informations sur les options de méthode, voir *Tivoli Workload Scheduler : Guide d'utilisation et de référence*.

Validation d'adresse IP

Lorsqu'une connexion TCP/IP est établie, netman lit le nom de noeud du demandeur et l'adresse IP depuis le socket. L'adresse IP et le nom d'hôte servent à rechercher un poste de travail Tivoli Workload Scheduler connu dans le fichier `Symphony`, avec les résultats possibles suivants :

- Si une correspondance d'adresse IP est trouvée, la validation réussit.
- Si une correspondance de nom d'hôte est trouvée, la validation réussit.
- Si aucune correspondance n'est trouvée dans le fichier `Symphony` ou si l'adresse IP retournée ne correspond pas à celle qui est lue à partir du socket, la validation échoue.

L'option locale, `nm ipvalidate`, détermine l'action à entreprendre si la validation IP échoue. Si l'option est définie sur `full`, l'échec de la validation entraîne la fermeture de la connexion par Tivoli Workload Scheduler et la génération d'un message d'erreur. Si l'option est définie sur `none` (valeur par défaut), Tivoli Workload Scheduler autorise toutes les connexions, mais génère un message d'avertissement en cas d'échec de vérification de validation.

Prise en charge du protocole IP version 6

Tivoli Workload Scheduler prend en charge le protocole IP version 6 (IPv6) en plus du protocole IPv4. Pour aider les clients à passer d'un environnement IPv4 à un

environnement IPv6 complet, Tivoli Workload Scheduler prend en charge l'utilisation des protocoles IP en double pile. En d'autres termes, le produit est conçu pour communiquer simultanément, via les protocole IPv4 et IPv6, avec d'autres applications utilisant IPv4 ou IPv6.

Dans ce but, les fonctions spécifiques d'IPv4 `gethostbyname` et `gethostbyaddr` ont été remplacées par la nouvelle interface de programme d'application `getaddrinfo` qui rend le mécanisme client-serveur entièrement indépendant du protocole.

La fonction `getaddrinfo` gère la translation nom-adresse et service-port et renvoie des structures `sockaddr` au lieu d'une liste d'adresses. Ces structures `sockaddr` peuvent alors être utilisées directement par les fonctions de connecteur. De cette façon, `getaddrinfo` masque toutes les dépendances de protocole dans la fonction de bibliothèque, là où elles sont stockées. L'application ne traite qu'avec les structures d'adresse de connexion qui sont spécifiées par `getaddrinfo`.

Configuration du système d'exploitation (UNIX uniquement)

La validation IP dépend de l'appel système `getaddrinfo()` pour rechercher toutes les adresses IP pour un hôte. Le comportement de cette routine varie en fonction de la configuration du système. Lorsque `getaddrinfo()` utilise le fichier `/etc/hosts`, il retourne la première entrée correspondante. Si la connexion est démarrée sur une adresse qui s'affiche après la première entrée correspondante, la validation IP échoue. Pour résoudre l'incident, placez l'entrée utilisée pour démarrer la connexion avant toute autre entrée correspondante du fichier `/etc/hosts`. Si `getaddrinfo()` utilise le serveur de noms "named" ou le serveur Network Information Service et que `getaddrinfo()` échoue, contactez votre administrateur système pour obtenir de l'aide.

Messages de validation de l'adresse IP

Voici une liste de messages de validation IP. Si l'option locale `nm ipvalidate` est définie sur `none` (par défaut), les erreurs apparaissent sous forme d'avertissements.

Voir après la liste de conditions pour connaître la signification des variables :

- Le nom du poste de travail Tivoli Workload Scheduler est introuvable dans le fichier Symphony

```
Validation de l'adresse IP non effectuée pour la requête :
Le service <num> for <program> sur <workstation>(<operating_system_type>).
Connexion reçue de l'adresse IP :
<c_ipaddr>. UC MAESTRO <workstation> introuvable dans le
fichier Symphony.
```

- Echec de l'appel à `getaddrinfo()` :

```
Validation de l'adresse IP non effectuée pour la requête :
Service num pour <program> sur l'UC (<operating_system_type>).
Connexion reçue de l'adresse IP :
<c_ipaddr>. getaddrinfo() a échoué, impossible de
récupérer l'adresse IP du noeud à connecter : <noeud>.
```

- Les adresses IP retournées par `getaddrinfo()` ne correspondent pas à l'adresse IP du poste de travail de connexion :

```
Validation de l'adresse IP non effectuée pour la requête :
Le service <num> for <program> sur <workstation>(<operating_system_type>).
Connexion reçue de l'adresse IP :
<c_ipaddr>. Adresses IP connues du système pour le nom de
noeud noeud : <k_ipaddr>.
```

- L'adresse IP spécifiée dans la définition du poste de travail Tivoli Workload Scheduler indiqué dans le paquet de demande de service ne correspond pas à l'adresse IP du poste de travail de connexion :

```
Validation de l'adresse IP non effectuée pour la requête :
Le service <num> for <program> sur <workstation>(<operating_system_type>).
Connexion reçue de l'adresse IP :
<c_ipaddr>. Adresses IP connues par TWS pour l'UC
<k_ipaddr>.
```

- Quel que soit l'état de `nm ipvalidate`, le message d'information suivant s'affiche lorsque la validation IP ne peut pas s'effectuer parce que le fichier `Symphony` n'existe pas ou qu'une erreur s'est produite lorsqu'il a été lu :

```
Validation de l'adresse IP non effectuée pour la
requête : Service <num> pour <program> sur
<workstation>(<operating_system_type>). Connexion reçue de l'adresse
IP : <c_ipaddr>. Impossible d'ouvrir ou de lire
fichier Symphony. Demande de service acceptée.
```

Où :

<num>
Numéro de service (2001-**writer**, 2002-**mailman**...)

<programme>
Programme demandant le service

<workstation>
Tivoli Workload Schedulernom du poste de travail de connexion

<operating_system_type>
Système d'exploitation du poste de travail de connexion

<noeud>
Nom d'hôte ou adresse IP du poste de travail de connexion

<c_ipaddr>
Adresse IP du poste de travail de connexion

<k_ipaddr>
Adresse IP connue du poste de travail de connexion

La validation de l'adresse IP réussit toujours en l'absence d'un fichier `Symphony`. Dans le cadre de communications depuis un gestionnaire de domaine vers un agent, elle aboutit toujours car il n'existe pas encore de fichier `Symphony`. Toutefois, si l'agent détient un fichier `Symphony` provenant d'une exécution précédente, il est possible que la demande initiale de connexion échoue si ce fichier `Symphony` n'inclut pas le nom du gestionnaire de domaine.

Impact des modifications réseau

Toute modification apportée au réseau peut avoir des répercussions sur Tivoli Workload Scheduler. Les postes de travail peuvent être identifiés dans Tivoli Workload Scheduler par le nom d'hôte ou l'adresse IP. Toute modification apportée aux noms d'hôtes ou adresses IP de postes de travail spécifiques doit également être effectuée dans la base de données Tivoli Workload Scheduler. Toutefois, gardez en mémoire que si ces postes de travail sont impliqués dans des travaux planifiés dans le fichier Symphony, ces travaux recherchent l'ancienne identité des postes de travail.

Les modifications apportées aux noms d'hôte ou aux adresses IP de postes de travail spécifiques peuvent être activées immédiatement en exécutant **JnextPlan -for 0000**. Un nouveau plan de production est créé (contenant les adresses IP et les noms d'hôte mis à jour), mais l'intervalle de temps du plan n'est pas prolongé.

Par conséquent, prévoyez vos modifications réseau en tenant compte des planifications de travaux. Pour les modifications majeures, il est conseillé d'interrompre les activités Tivoli Workload Scheduler jusqu'à ce que les changements réseau soient terminés et répercutés dans la base de données Tivoli Workload Scheduler.

Les modifications réseau ont également un impact spécifique sur les paramètres de connexion utilisés par le serveur d'applications et le client de ligne de commande :

Serveur d'applications
Si vous modifiez le réseau, vous devez changer les paramètres de communication indiqués dans les fichiers de configuration du serveur d'applications. La procédure est décrite dans l'annexe des utilitaires fournis avec WebSphere Application Server dans le manuel *Tivoli Workload Scheduler - Guide de planification et d'installation*.

Client de ligne de commande
Lorsque vous vous connectez à partir du client de ligne de commande, vous indiquez un ensemble de paramètres de connexion. Pour ce faire, utilisez une des méthodes suivantes :

Depuis le fichier localopts
La méthode par défaut consiste à personnaliser les paramètres de connexion du fichier `localopts` lors de l'installation du client de ligne de commande.

Depuis le fichier useropts
Il est possible qu'un fichier `useropts` ait été créé pour l'utilisateur en question, contenant une version des paramètres de connexion personnalisés pour lui.

Sur la ligne de commande, individuellement
Lorsque vous appelez l'un des programmes de ligne de commande, vous pouvez inclure les paramètres en tant qu'arguments de la commande. Ils sont alors prioritaires sur les valeurs des fichiers `localopts` ou `useropts`.

Sur la ligne de commande, dans un fichier
Lorsque vous appelez l'un des programmes de ligne de commande, vous pouvez inclure les paramètres dans un fichier dont le nom est identifié par l'argument **-file** de la commande. Ils sont alors prioritaires sur les valeurs des fichiers `localopts` ou `useropts`.

Modifiez la méthode que vous utilisez pour intégrer les nouvelles informations sur la connexion réseau.

Chapitre 7. Définition de la sécurité des connexions

Tivoli Workload Scheduler fournit des paramètres de sécurité de connexion par défaut lors de l'installation. Vous pouvez personnaliser votre sécurité de connexion dans votre environnement Tivoli Workload Scheduler.

Présentation de la sécurité des connexions

Tivoli Workload Scheduler fournit un mécanisme de connexion sécurisée, authentifiée et cryptée pour des communications basées sur le protocole SSL (Secure Sockets Layer), qui est automatiquement installé avec Tivoli Workload Scheduler.

Tivoli Workload Scheduler fournit également des certificats par défaut pour gérer le protocole SSL qui est basé sur méthodologie de clé privée et publique.

Si vous ne personnalisez pas la communication SSL avec vos certificats, pour communiquer en mode SSL, Tivoli Workload Scheduler utilise alors les certificats par défaut qui sont stockés dans les répertoires par défaut. Toutefois, dans un environnement de production, nous vous recommandons de personnaliser la communication SSL avec vos propres certificats comme indiqué dans les scénarios suivants.

Vous pouvez personnaliser une communication SSL avec vos certificats conformément à vos exigences en matière de sécurité.

Vous pouvez être confronté aux scénarios suivants :

- «Scénario : connexion entre Dynamic Workload Console et le Tivoli Workload Scheduler ayant un connecteur distribué», à la page 262.
- «Scénario : connexion entre l'agents dynamiques et le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique», à la page 282.
- «Scénario : communication SSL sur le réseau Tivoli Workload Scheduler», à la page 286.
- «Scénario : HTTPS pour les clients de ligne de commande», à la page 296.

Tivoli Workload Scheduler utilise les types de magasins suivants :

magasin de clés de confiance
Dans le contexte de la sécurité, il s'agit d'un objet d'archivage (fichier ou carte cryptographique matérielle) où sont stockées ses clés publiques sous forme de certificats de confiance, à des fins d'authentification lors de transactions Internet. Dans certaines applications, ces certificats de confiance sont déplacés dans le fichier de clés de l'application à stocker avec les clés privées.

magasin de clés
Dans le contexte de la sécurité, il s'agit d'un fichier ou d'une carte cryptographique physique dans lequel les identités et les clés privées servant à l'authentification et au chiffrement sont enregistrées. Certains magasins de clés contiennent aussi des clés publiques ou dignes de confiance.

Scénario : connexion entre Dynamic Workload Console et le Tivoli Workload Scheduler ayant un connecteur distribué

Dynamic Workload Console se connecte en mode SSL au composant Tivoli Workload Scheduler ayant un connecteur distribué, à l'aide de certificats par défaut. Vous pourriez configurer la console Dynamic Workload Console pour se connecter en mode SSL à l'aide de vos certificats.

Vous pouvez avoir une communication SSL entre la console Dynamic Workload Console et l'un des composants suivants :

- Gestionnaire de domaine maître
- Gestionnaire de domaine maître de sauvegarde
- Gestionnaire de domaine dynamique
- Gestionnaire de domaine dynamique de sauvegarde.
- Agent avec connecteur distribué.

Lorsque vous personnalisez les paramètres de Dynamic Workload Console, assurez-vous que les clés possèdent le même mot de passe que le magasin de clés où elles sont enregistrées. Le mot de passe du magasin de clé de Dynamic Workload Console doit être le même que celui du client Dynamic Workload Console du serveur Tivoli Workload Scheduler.

Remarque : Lorsque vous configurez la console Dynamic Workload Console pour se connecter à différents agents avec connecteur distribué, le fichier de clés certifiées de la console Dynamic Workload Console doit avoir un certificat pour chaque connecteur, pour activer une connexion SSL.

Présentation

Pour plus d'informations sur la connexion SSL entre la console Dynamic Workload Console et des composants qui ont un connecteur distribué, voir «Présentation».

Connexion SSL à l'aide des certificats par défaut

Pour plus d'informations sur la connexion par défaut SSL, voir «Connexion SSL à l'aide de certificats par défaut», à la page 264.

Connexion SSL à l'aide de vos certificats

Pour plus d'informations sur comment créer et activer vos certificats SSL, voir «Connexion SSL à l'aide de vos certificats», à la page 267.

Présentation

Présentation de la connexion SSL de Dynamic Workload Console

Pour implémenter la communication d'invocation RMI sur protocole IIOP sur SSL (RMI/IIOP over SSL) entre la console Dynamic Workload Console et la communication interne SOAP du gestionnaire de domaine maître, du gestionnaire de domaine maître de sauvegarde, du gestionnaire de domaine dynamique et du gestionnaire de domaine dynamique de sauvegarde ou l'agent avec connecteur distribué, utilisez les fonctions de sécurité du serveur et du client de WebSphere Application Server.

Le paradigme de sécurité SSL implémenté dans WebSphere Application Server requiert deux magasins de clés sur les clients et le serveur : le magasin de clés contenant la clé privée et le fichier de clés certifiées contenant les certificats des contreparties sécurisés.

La figure 6 illustre les clés du serveur et du client et indique où elles doivent être exportées pour la console Dynamic Workload Console :

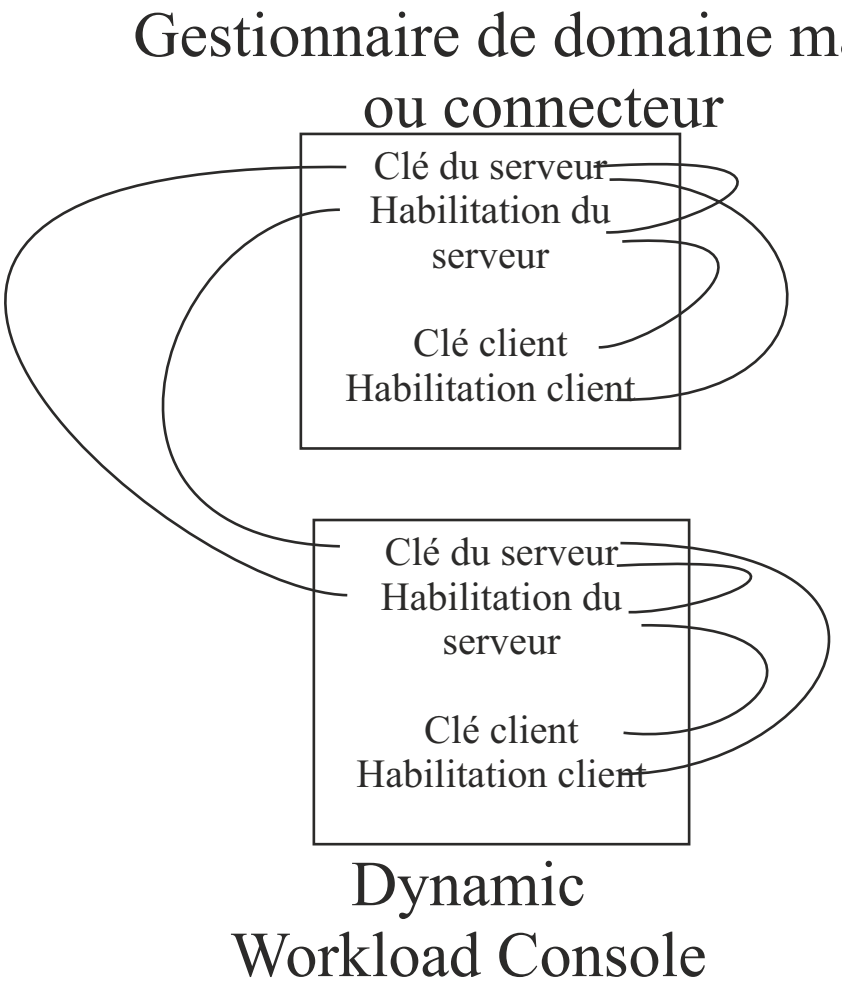


Figure 6. Clés du serveur et du client SSL

Le diagramme présente la console Dynamic Workload Console des clés et les composants qui disposent d'un connecteur distribué doivent être extraits et distribués pour activer une communication SSL.L'interface de la Dynamic Workload Console utilise les certificats par défaut installés dans les magasins de clés par défaut pour communiquer avec l'agent avec connecteur distribué. Vous pouvez configurer la console Dynamic Workload Console pour une connexion en mode SSL avec un agent avec connecteur distribué, à l'aide de vos certificats qui répondent à vos paramètres de sécurité requis.

Outre la création de clés, vous pouvez également personnaliser le nom, l'emplacement et le mot de passe du magasin de clés et du fichier de clés certifiées. Pour plus d'informations sur les possibilités, voir tableau 54, à la page 264.

Tableau 54. Changements autorisés dans le magasin de clés et le fichier de clés certifiées de Tivoli Workload Scheduler

Fichier	Nom	Chemin	Mot de passe	Nouvelle clé
Magasin de clés du serveur TWS	✓	✓	✓	✓
Fichier de clés certifiées du serveur TWS	✓	✓	✓	✓
Magasin de clés du client TWS				✓
Fichier de clés certifiées du client TWS				✓
Magasin de clés du client TDWC				✓
Fichier de clés certifiées du client TDWC				✓

Lorsque vous personnalisez les paramètres de la console Dynamic Workload Console, vérifiez que les clés ont le même mot de passe que le magasin de clés dans lequel elles ont été enregistrées. Le mot de passe du magasin de clés de Dynamic Workload Console doit être le même que celui du client Dynamic Workload Console et du serveur Tivoli Workload Scheduler.

Remarque : Lorsque vous configurez la console Dynamic Workload Console pour se connecter à différents agents avec connecteur distribué, le fichier de clés certifiées de la console Dynamic Workload Console doit avoir un certificat pour chaque connecteur, pour activer une connexion SSL.

Connexion SSL à l'aide de certificats par défaut

La connexion SSL entre Dynamic Workload Console et le gestionnaire de domaine maître, le gestionnaire de domaine maître de sauvegarde, le gestionnaire de domaine dynamique, le gestionnaire de domaine dynamique de sauvegarde ou l'agent avec connecteur distribué est activée à l'aide de certificats par défaut.

Vous obtenez l'environnement suivant :

Dynamic Workload Console est installé sur le poste de travail *DWC-WKS* **:**

- Dynamic Workload Console est installé dans le répertoire `<REP_INST_DWC>`.
- Le serveur WebSphere Application Server imbriqué est installé dans le répertoire `<REP_INST_DWC>\eWAS`.

Le gestionnaire de domaine maître, le gestionnaire de domaine maître de sauvegarde, le gestionnaire de domaine dynamique, le gestionnaire de domaine dynamique de sauvegarde ou l'agent avec connecteur distribué sont installés sur le poste de travail *TWS-WKS* **:**

- L'agent avec connecteur distribué est installé dans le répertoire `<REP_INST_TWS>`.
- Le serveur WebSphere Application Server imbriqué est installé dans le répertoire `<REP_INST_TWS>\eWAS`.

Par défaut, la connexion SSL entre Dynamic Workload Console et le composant avec connecteur distribué est activé à l'aide de certificats par défaut. Le mot de

passe par défaut associé à chaque fichier de clés par défaut correspond à `default`. La connexion SSL dispose des certificats suivants :

Sur les poste de travail Dynamic Workload Console :

Magasin de clés de confiance

Sur les système d'exploitation Windows :

- `<REP_INST_DWC>\eWAS\profiles\TIPProfile\etc\TWSServerTrustFile.jks`
- `<REP_INST_DWC>\eWAS\profiles\TIPProfile\etc\TWSClientTrustFile.jks`

Sur les systèmes d'exploitation UNIX :

- `<REP_INST_DWC>/eWAS/profiles/TIPProfile/etc/TWSServerTrustFile.jks`
- `<REP_INST_DWC>/eWAS/profiles/TIPProfile/etc/TWSClientTrustFile.jks`

Magasin de clés

Sur les système d'exploitation Windows :

- `<REP_INST_DWC>\eWAS\profiles\TIPProfile\etc\TWSServerKeyFile.jks`
- `<REP_INST_DWC>\eWAS\profiles\TIPProfile\etc\TWSClientKeyFile.jks`

Sur les systèmes d'exploitation UNIX :

- `<REP_INST_DWC>/eWAS/profiles/TIPProfile/etc/TWSServerKeyFile.jks`
- `<REP_INST_DWC>/eWAS/profiles/TIPProfile/etc/TWSClientKeyFile.jks`

où *<REP_INST_DWC>* est le répertoire d'installation de Dynamic Workload Console.

Sur les postes de travail du gestionnaire de domaine maître, gestionnaire de domaine maître de sauvegarde, du gestionnaire de domaine dynamique, du gestionnaire de domaine dynamique de sauvegarde ou de l'agent avec connecteur distribué :

Magasin de clés de confiance

Sur les système d'exploitation Windows :

- `<REP_INST_TWS>\eWAS\profiles\TIPProfile\etc\TWSServerTrustFile.jks`
- `<REP_INST_TWS>\eWAS\profiles\TIPProfile\etc\TWSClientTrustFile.jks`

Sur les systèmes d'exploitation UNIX :

- `<REP_INST_TWS>/eWAS/profiles/TIPProfile/etc/TWSServerTrustFile.jks`
- `<REP_INST_TWS>/eWAS/profiles/TIPProfile/etc/TWSClientTrustFile.jks`

Magasin de clés

Sur les système d'exploitation Windows :

- <REP_INST_TWS>\eWAS\profiles\TIPProfile\etc\TWSServerKeyFile.jks
- <REP_INST_TWS>\eWAS\profiles\TIPProfile\etc\TWSClientKeyFile.jks

Sur les systèmes d'exploitation UNIX :

- <REP_INST_TWS>/eWAS/profiles/TIPProfile/etc/TWSServerKeyFile.jks
- <REP_INST_TWS>/eWAS/profiles/TIPProfile/etc/TWSClientKeyFile.jks

où *<REP_INST_TWS>* est le répertoire d'installation de Tivoli Workload Scheduler.

Pour plus d'informations sur les fichiers de configuration SSL hébergeant les informations du fichier de clés certifiées et du fichier de clé, voir «Localisation des fichiers de clés».

Remarque : Les certificats par défaut ne sont pas utilisés pour l'authentification du client Dynamic Workload Console obtenue à l'aide d'un ID utilisateur et d'un mot de passe.

Localisation des fichiers de clés

Pour localiser les fichiers de clés, exécutez l'utilitaire `showSecurityProperties`, décrit dans la section suivante : «Propriétés de sécurité : référence», à la page 206. Pour modifier le nom, l'emplacement, le mot de passe des fichiers de clés et de clés certifiées du serveur Tivoli Workload Scheduler, vous devez modifier les fichiers de configuration qui les décrivent.

Fichiers de clés client pour tous les composants

Les fichiers de clés client pour Tivoli Workload Scheduler master sont décrits dans le fichier : *TWA_home*`/WAS/TWSprofile/properties/ssl.client.props`. Les fichiers de clés client pour Dynamic Workload Console sont décrits dans le fichier :`rēp_profil_JazzSM/properties/ssl.client.props` .

Voici un exemple de fichier :

```
# KeyStore information
com.ibm.ssl.keyStoreName=ClientDefaultKeyStore
com.ibm.ssl.keyStore=/opt/ibm/TWA0/WAS/TWSprofile/etc/
                                                TWSClientKeyFile.jks
com.ibm.ssl.keyStorePassword={xor}Ozo5PiozKw\=\=
com.ibm.ssl.keyStoreType=JKS
com.ibm.ssl.keyStoreProvider=IBMJCE
com.ibm.ssl.keyStoreFileBased=true

# TrustStore information
com.ibm.ssl.trustStoreName=ClientDefaultTrustStore
com.ibm.ssl.trustStore=/opt/ibm/TWA0/WAS/TWSprofile/etc/
                                                TWSClientTrustFile.jks
com.ibm.ssl.trustStorePassword={xor}Ozo5PiozKw\=\=
com.ibm.ssl.trustStoreType=JKS
com.ibm.ssl.trustStoreProvider=IBMJCE
com.ibm.ssl.trustStoreFileBased=true
```

Pour modifier les noms, chemins ou mots de passe du fichier de clés du serveur, modifiez les fichiers de configuration à l'aide du script **changeSecurityProperties**

situé dans le répertoire *TWA_home*/TWS/wastool. Pour connaître la procédure à appliquer, voir «Modification des paramètres de sécurité», à la page 406. Voici un exemple de données en entrée :

```
############################################################
SSL Panel
############################################################
alias=DefaultSSLSettings
keyFileName=${RACINE_INSTALLATION_UTILISATEUR}/etc/TWSServerKeyFile.jks
keyFilePassword=*****
keyFileFormat=JKS
trustFileName=${RACINE_INSTALLATION_UTILISATEUR}/etc/TWSServerTrustFile.jks
trustFilePassword=*****
trustFileFormat=JKS
clientAuthentication=false
securityLevel=HIGH
enableCryptoHardwareSupport=false
```

Important : Les certificats pour Dynamic Workload Console ont été modifiés et expireront après une année. Pour renouveler les certificats, suivez la procédure décrite dans la documentation suivante : section à propos de la sécurité et du renouvellement d'un certificat SSL pour WebSphere Application Server. .

Le tableau suivant indique l'ancien et le nouveau nom et chemin d'accès à Tivoli Workload Scheduler et les certificats Dynamic Workload Console.

Tableau 55. Fichiers de clés et fichiers de clés certifiées

Magasin	Certificat précédent	Chemin d'accès actuel au certificat
Fichier de clés de licence serveur TWS	TWSServerKeyFile.jks	/opt/ibm/TWA0/WAS/TWSprofile/etc/TWSServerKeyFile.jks
Fichier de clés certifiées du serveur TWS	TWSServerTrustFile.jks	/opt/ibm/TWA0/WAS/TWSprofile/etc/TWSServerTrustFile.jks
Fichier de clés du client TWS	TWSClientKeyFile.jks	/opt/ibm/TWA0/WAS/TWSprofile/etc/TWSClientKeyFile.jks
Fichier de clés certifiées du client TWS	TWSClientTrustFile.jks	/opt/ibm/TWA0/WAS/TWSprofile/etc/TWSClientTrustFile.jks
Magasin de clés du serveur DWC	TWSServerKeyStore.jks	*rép_profil_JazzSM*/config/cells/JazzSMNode01Cell/nodes/JazzSMNode01/key.p12
Fichier de clés certifiées du serveur DWC	TWSServerTrustStore.jks	*rép_profil_JazzSM*/config/cells/JazzSMNode01Cell/nodes/JazzSMNode01/trust.p12
Fichier de clés client TWS	TWSClientKeyStore.jks	*rép_profil_JazzSM*/etc/key.p12
Fichier de clés certifiées du client DWC	TWSClientTrustStore.jks	*rép_profil_JazzSM*/etc/trust.p12

Connexion SSL à l'aide de vos certificats

Vous pouvez configurer Dynamic Workload Console pour une connexion en mode SSL avec un gestionnaire de domaine maître, gestionnaire de domaine maître de

sauvegarde, gestionnaire de domaine dynamique, gestionnaire de domaine dynamique de sauvegarde ou un agent avec connecteur distribué à l'aide de vos certificats.

Vous obtenez l'environnement suivant :

Dynamic Workload Console est installé sur le poste de travail *DWC-WKS* **:**

- Dynamic Workload Console est installé dans le répertoire `<REP_INST_DWC>`.
- Le serveur WebSphere Application Server imbriqué est installé dans le répertoire `<REP_INST_DWC>\eWAS`.

Le gestionnaire de domaine maître, gestionnaire de domaine maître de sauvegarde, gestionnaire de domaine dynamique, gestionnaire de domaine dynamique de sauvegarde ou l'agent avec connecteur distribué est installé sur le poste de travail *TWS-WKS* **:**

- Le gestionnaire de domaine maître, gestionnaire de domaine maître de sauvegarde, gestionnaire de domaine dynamique, gestionnaire de domaine dynamique de sauvegarde ou l'agent avec connecteur distribué est installé dans le répertoire `<REP_INST_TWS>`.
- Le serveur WebSphere Application Server imbriqué est installé dans le répertoire `<REP_INST_TWS>\eWas`.

  Remarque : Le gestionnaire de domaine maître, gestionnaire de domaine maître de sauvegarde, gestionnaire de domaine dynamique, gestionnaire de domaine dynamique de sauvegarde ou l'agent avec connecteur distribué est appelé *agent* avec connecteur distribué. De la même manière, le mot clé utilisé au cours de la création des clés s'intitule *agent*. Lorsque vous exécutez la procédure, vous pouvez insérer un nom qui spécifie l'agent pour lequel vous exécutez la procédure, à savoir *Maître* pour le gestionnaire de domaine maître ou *ddm* pour le gestionnaire de domaine dynamique.

Comme décrit dans figure 6, à la page 263, dans WebSphere Application Server, vous devez créer les bases de données de clés suivantes :

Sur l'agent avec instance de connecteur distribué :

- *Agent* Server key
- *Agent* Server trust
- *Agent* Client key
- *Agent* Client trust

Sur l'instance Dynamic Workload Console :

- *DWC* Server key
- *DWC* Server trust
- *DWC* Client key
- *DWC* Client trust

Puis, exportez et importez mutuellement les clés et activez WebSphere Application Server pour fonctionner avec les nouveaux certificats.

Procédure rapide :

1. Créez la base de données *Agent Server key*, exécutez 1, à la page 269.
2. Créez la base de données *Agent Server trust*, exécutez 2, à la page 270.

3. Créez la base de données *Agent Client Key*, exécutez 3, à la page 271.
4. Créez la base de données *Agent Client Trust*, exécutez 4, à la page 272.
5. Créez la base de données *DWC Server Key*, exécutez 5, à la page 272.
6. Créez la base de données *DWC Server Trust*, exécutez 6, à la page 274.
7. Créez la base de données *DWC Client Key*, exécutez 7, à la page 274.
8. Créez la base de données *DWC Client Trust*, exécutez 8, à la page 275.
9. Importez les certificats signés dans `AgentServerTrust`, exécutez 9, à la page 276.
10. Importez les certificats signés dans `AgentClientTrust`, exécutez 10, à la page 276.
11. Importez les certificats signés dans `DWCServerTrust`, exécutez 11, à la page 277.
12. Importez les certificats signés dans `DWCClientTrust`, exécutez 12, à la page 278.
13. Configurez les nouveaux fichiers de clés serveur dans l'agent Tivoli Workload Scheduler, exécutez 13, à la page 278.
14. Configurez les nouveaux fichiers client dans l'agent Tivoli Workload Scheduler, exécutez 14, à la page 279.
15. Configurez les nouveaux fichiers de clés serveur dans Dynamic Workload Console, exécutez 15, à la page 280.
16. Configurez les nouveaux fichiers client dans Dynamic Workload Console, exécutez 16, à la page 281.

Exécutez les étapes suivantes :

1. **Créez la base de données *Agent Server key*** :
   a. Ouvrez une session en tant qu'administrateur sur les systèmes d'exploitation Windows ou comme superutilisateur sur les systèmes d'exploitation UNIX et Linux, sur la machine dans laquelle vous avez installé l'agent *TWS-AGENT*.
   b. Exécutez la commande `<REP_INST_TWS>\eWas\java\jre\bin\ikeyman` ou utilisez la commande `ikeyman` fournie par une instance Java de votre machine.
   c. Sur le panneau IBM Key Management, cliquez sur **Key Database File** > **Nouveau**.
   d. Sur le panneau Nouveau, entrez les informations suivantes :

      Type de base de données de clés
      Sélectionnez la valeur type `JKS`.

      Nom de fichier
      Insérez la valeur `Agent Server key` : `ServerAgentKey.jks`

      Emplacement
      Insérez le nom de répertoire *<REP_CERTS_TWS>* dans lequel vous voulez enregistrer le fichier `ServerAgentKey.jks`.

   e. Cliquez sur **OK**.
   f. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.
   g. Cliquez sur **OK**.
   h. Dans la section Key database information du panneau IBM Key Management, vous pouvez voir le `<REP_CERTS_TWS>\ServerAgentKey.jks` qui vient d'être créé. Dans la liste déroulante, sélectionnez **Certificats personnels**, puis cliquez sur **New Self-Signed...**.

i. Créez le certificat autosigné, en entrant au moins l'une des informations suivantes sur le panneau Create New Self-Signed Certificate :

Intitulé de clé
Insérez la valeur TWSAgentServer.

Version
Insérez la valeur X509 V3.

Taille de clé
Insérez la valeur 2048.

Algorithme de signature
Insérez la valeur SHA2WithRSA.

Nom usuel
Insérez la valeur AgentServer.

Période de validité
Insérez la valeur 365.

j. Cliquez sur **OK**. *twsagentsserver* apparaît dans la liste **Certificats personnels**.

k. Pour créer le certificat certAgentServer.arm, sélectionnez *twsagentsserver* dans la liste des Certificats personnels, puis cliquez sur **Extract certificate**.

l. Sur le panneau Nouveau, entrez les informations suivantes :

Type de données :
Sélectionnez Base64-encoded ASCII data (Données ASCII codées en Base64).

Nom du fichier certificat :
Insérez la valeur certAgentServer.arm.

Emplacement
Insérez le nom de répertoire <*REP_CERTS_TWS*> dans lequel vous voulez enregistrer le fichier certAgentServer.arm.

m. Cliquez sur **OK**.

2. **Créez la base de données *Agent Server trust*** :

a. Ouvrez une session en tant qu'administrateur sur les systèmes d'exploitation Windows ou comme superutilisateur sur les systèmes d'exploitation UNIX et Linux, sur la machine dans laquelle vous avez installé l'agent *TWS-AGENT*.

b. Exécutez la commande <REP_INST_TWS>\eWas\java\jre\bin\ikeyman ou utilisez la commande ikeyman fournie par une instance Java de votre machine.

c. Sur le panneau IBM Key Management, cliquez sur **Key Database File** > **Nouveau**.

d. Sur le panneau Nouveau, entrez les informations suivantes :

Type de base de données de clés
Sélectionnez la valeur type JKS.

Nom de fichier
Insérez la valeur *Agent Server trust* : ServerAgentTrust.jks

Emplacement
Insérez le nom de répertoire <*REP_CERTS_TWS*> dans lequel vous voulez enregistrer le fichier ServerAgentTrust.jks.

e. Cliquez sur **OK**.

f. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.

g. Cliquez sur **OK**.

h. Dans la liste déroulante, sélectionnez **Certificat de signataire**, puis cliquez sur **Ajouter** pour ajouter le fichier `certAgentServer.arm` créé dans 1l, à la page 270.

i. Entrez le libellé `AgentServerTrust` pour le certificat `certAgentServer.arm`. *AgentServerTrust* apparaît dans la liste **Certificats de signataire**..

j. Cliquez sur **OK**.

3. **Créez la base de données *Agent Client Key*** :

a. Ouvrez une session en tant qu'administrateur sur les systèmes d'exploitation Windows ou comme superutilisateur sur les systèmes d'exploitation UNIX et Linux, sur la machine dans laquelle vous avez installé l'agent Tivoli Workload Scheduler.

b. Exécutez la commande `<REP_INST_TWS>\eWas\java\jre\bin\ikeyman` ou utilisez la commande `ikeyman` fournie par une instance Java de votre machine.

c. Sur le panneau IBM Key Management, cliquez sur **Key Database File** > **Nouveau**.

d. Sur le panneau Nouveau, entrez les informations suivantes :

Type de base de données de clés
Sélectionnez la valeur type `JKS`.

Nom de fichier
Insérez la valeur *Agent Client Key* : `ClientAgentKey.jks`

Emplacement
Insérez le nom de répertoire *<REP_CERTS_TWS>* dans lequel vous voulez enregistrer le fichier `ClientAgentKey.jks`.

e. Cliquez sur **OK**.

f. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.

g. Cliquez sur **OK**.

h. Dans la section Key database information du panneau IBM Key Management, vous pouvez voir le `<REP_CERTS_TWS>\ClientAgentKey.jks` qui vient d'être créé. Dans la liste déroulante, sélectionnez **Certificats personnels**, puis cliquez sur **New Self-Signed...**.

i. Créez le certificat autosigné, en entrant au moins l'une des informations suivantes sur le panneau Create New Self-Signed Certificate :

Intitulé de clé
Insérez la valeur `TWSAgentClient`.

Version
Insérez la valeur `X509 V3`.

Taille de clé
Insérez la valeur `2048`.

Algorithme de signature
Insérez la valeur `SHA2WithRSA`.

Nom usuel
Insérez la valeur `AgentClient`.

Période de validité
Insérez la valeur 365.

j. Cliquez sur **OK**. *twsagentsClient* apparaît dans la liste **Certificats personnels**.

k. Pour créer le certificat certAgentClient.arm, sélectionnez *twsagentsClient* dans la liste des Certificats personnels, puis cliquez sur **Extract certificate**.

l. Sur le panneau Nouveau, entrez les informations suivantes :

Type de données :
Sélectionnez Base64-encoded ASCII data (Données ASCII codées en Base64).

Nom du fichier certificat :
Insérez la valeur certAgentClient.arm.

Emplacement
Insérez le nom de répertoire <*REP_CERTS_TWS*> dans lequel vous voulez enregistrer le fichier certAgentClient.arm.

m. Cliquez sur **OK**.

4. **Créez la base de données *Agent Client Trust*** :

a. Ouvrez une session en tant qu'administrateur sur les systèmes d'exploitation Windows ou comme superutilisateur sur les systèmes d'exploitation UNIX et Linux, sur la machine dans laquelle vous avez installé l'agent *TWS-AGENT*.

b. Exécutez la commande <REP_INST_TWS>\eWas\java\jre\bin\ikeyman ou utilisez la commande ikeyman fournie par une instance Java de votre machine.

c. Sur le panneau IBM Key Management, cliquez sur **Key Database File** > **Nouveau**.

d. Sur le panneau Nouveau, entrez les informations suivantes :

Type de base de données de clés
Sélectionnez la valeur type JKS.

Nom de fichier
Insérez la valeur *Agent Client Trust* : ClientAgentTrust.jks

Emplacement
Insérez le nom de répertoire <*REP_CERTS_TWS*> dans lequel vous voulez enregistrer le fichier ClientAgentTrust.jks.

e. Cliquez sur **OK**.

f. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.

g. Cliquez sur **OK**.

h. Dans la liste déroulante, sélectionnez **Certificats de signataire**, puis cliquez sur **Ajouter** pour ajouter le fichier certAgentClient.arm créé dans 3l.

i. Entrez le libellé *ClientAgentTrust* pour le certificat certAgentClient.arm. *ClientAgentTrust* apparaît dans la liste **Certificats de signataire**.

j. Cliquez sur **OK**.

5. **Créez la base de données *DWC Server Key*** :

a. Ouvrez une session en tant qu'administrateur sur les systèmes d'exploitation Windows ou comme superutilisateur sur les systèmes d'exploitation UNIX et Linux, sur la machine dans laquelle vous avez installé *DWC* Dynamic Workload Console.

b. Exécutez la commande <REP_INST_DWC>\eWas\java\jre\bin\ikeyman ou utilisez la commande ikeyman fournie par une instance Java de votre machine.

c. Sur le panneau IBM Key Management, cliquez sur **Key Database File** > **Nouveau**.

d. Sur le panneau Nouveau, entrez les informations suivantes :

Type de base de données de clés
Sélectionnez la valeur type JKS.

Nom de fichier
Insérez la valeur *DWC Server Key* : ServerDWCKey.jks

Emplacement
Insérez le nom de répertoire <*REP_CERTS_DWC*> dans lequel vous voulez enregistrer le fichier ServerDWCKey.jks.

e. Cliquez sur **OK**.

f. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.

g. Cliquez sur **OK**.

h. Dans la section Key database information du panneau IBM Key Management, vous pouvez voir le <REP_CERTS_TWS>\ServerDWCKey.jks qui vient d'être créé. Dans la liste déroulante, sélectionnez **Certificats personnels**, puis cliquez sur **New Self-Signed...**.

i. Créez le certificat autosigné, en entrant au moins l'une des informations suivantes sur le panneau Create New Self-Signed Certificate :

Intitulé de clé
Insérez la valeur TWSDWCServer.

Version
Insérez la valeur X509 V3.

Taille de clé
Insérez la valeur 2048.

Algorithme de signature
Insérez la valeur SHA2WithRSA.

Nom usuel
Insérez la valeur DWCServer.

Période de validité
Insérez la valeur 365.

j. Cliquez sur **OK**. *twsDWCserver* apparaît dans la liste **Certificats personnels**.

k. Pour créer le certificat certDWCServer.arm, sélectionnez *twsDWCsserver* dans la liste des Certificats personnels, puis cliquez sur **Extract certificate**.

l. Sur le panneau Nouveau, entrez les informations suivantes :

Type de données :
Sélectionnez Base64-encoded ASCII data (Données ASCII codées en Base64).

Nom du fichier certificat :
Insérez la valeur certDWCServer.arm.

Emplacement
Insérez le nom de répertoire <*REP_CERTS_DWC*> dans lequel vous voulez enregistrer le fichier certDWCServer.arm.

m. Cliquez sur **OK**.

6. **Créez la base de données *DWC Server Trust*** :
   a. Ouvrez une session en tant qu'administrateur sur les systèmes d'exploitation Windows ou comme superutilisateur sur les systèmes d'exploitation UNIX et Linux, sur la machine dans laquelle vous avez installé *DWC* Dynamic Workload Console.
   b. Exécutez la commande `<REP_INST_DWC>\eWas\java\jre\bin\ikeyman` ou utilisez la commande `ikeyman` fournie par une instance Java de votre machine.
   c. Sur le panneau IBM Key Management, cliquez sur **Key Database File** > **Nouveau**.
   d. Sur le panneau Nouveau, entrez les informations suivantes :

      Type de base de données de clés
      Sélectionnez la valeur type `JKS`.

      Nom de fichier
      Insérez la valeur *DWC Server Trust* : `ServerDWCTrust.jks`

      Emplacement
      Insérez le nom de répertoire *<REP_CERTS_DWC>* dans lequel vous voulez enregistrer le fichier `ServerDWCTrust.jks`.

   e. Cliquez sur **OK**.
   f. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.
   g. Cliquez sur **OK**.
   h. Dans la liste déroulante, sélectionnez **Certificats de signataire**, puis cliquez sur **Ajouter** pour ajouter le fichier `certDWCServer.arm` créé dans 5l, à la page 273.
   i. Entrez le libellé *DWCServerTrust* pour le certificat `certDWCServer.arm`. `certDWCServer` apparaît dans la liste **Certificats de signataire**.
   j. Cliquez sur **OK**.

7. **Créez la base de données *DWC Client Key*** :
   a. Ouvrez une session en tant qu'administrateur sur les systèmes d'exploitation Windows ou comme superutilisateur sur les systèmes d'exploitation UNIX et Linux, sur la machine dans laquelle vous avez installé *DWC* Dynamic Workload Console.
   b. Exécutez la commande `<REP_INST_DWC>\eWas\java\jre\bin\ikeyman` ou utilisez la commande `ikeyman` fournie par une instance Java de votre machine.
   c. Sur le panneau IBM Key Management, cliquez sur **Key Database File** > **Nouveau**.
   d. Sur le panneau Nouveau, entrez les informations suivantes :

      Type de base de données de clés
      Sélectionnez la valeur type `JKS`.

      Nom de fichier
      Insérez la valeur *DWC Client Key* : `ClientDWCKey.jks`

      Emplacement
      Insérez le nom de répertoire *<REP_CERTS_DWC>* dans lequel vous voulez enregistrer le fichier `ClientDWCKey.jks`.

   e. Cliquez sur **OK**.

f. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.

g. Cliquez sur **OK**.

h. Dans la section Key database information du panneau IBM Key Management, vous pouvez voir le `<REP_CERTS_DWC>\ClientDWCKey.jks` qui vient d'être créé. Dans la liste déroulante, sélectionnez **Certificats personnels**, puis cliquez sur **New Self-Signed...**.

i. Créez le certificat autosigné, en entrant au moins l'une des informations suivantes sur le panneau Create New Self-Signed Certificate :

Intitulé de clé
Insérez la valeur `TWSDWCClient`.

Version
Insérez la valeur `X509 V3`.

Taille de clé
Insérez la valeur `2048`.

Algorithme de signature
Insérez la valeur `SHA2WithRSA`.

Nom usuel
Insérez la valeur `DWCClient`.

Période de validité
Insérez la valeur `365`.

j. Cliquez sur **OK**. *twsDWCClient* apparaît dans la liste **Certificats personnels**.

k. Pour créer le certificat `certDWCClient.arm`, sélectionnez *twsDWCClient* dans la liste des Certificats personnels, puis cliquez sur **Extract certificate**.

l. Sur le panneau Nouveau, entrez les informations suivantes :

Type de données :
Sélectionnez `Base64-encoded ASCII data` (Données ASCII codées en Base64).

Nom du fichier certificat :
Insérez la valeur `certDWCClient.arm`.

Emplacement
Insérez le nom de répertoire *<REP_CERTS_TWS>* dans lequel vous voulez enregistrer le fichier `certDWCClient.arm`.

m. Cliquez sur **OK**.

8. **Créez la base de données *DWC Client Trust*** :

a. Ouvrez une session en tant qu'administrateur sur les systèmes d'exploitation Windows ou comme superutilisateur sur les systèmes d'exploitation UNIX et Linux, sur la machine dans laquelle vous avez installé *DWC* Dynamic Workload Console.

b. Exécutez la commande `<REP_INST_DWC>\eWas\java\jre\bin\ikeyman` ou utilisez la commande `ikeyman` fournie par une instance Java de votre machine.

c. Sur le panneau IBM Key Management, cliquez sur **Key Database File** > **Nouveau**.

d. Sur le panneau Nouveau, entrez les informations suivantes :

Type de base de données de clés
Sélectionnez la valeur type `JKS`.

Nom de fichier
Insérez la valeur *DWC Client Trust* : ClientDWCTrust.jks

Emplacement
Insérez le nom de répertoire <*REP_CERTS_DWC*> dans lequel vous voulez enregistrer le fichier ClientDWCTrust.jks.

e. Cliquez sur **OK**.

f. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.

g. Cliquez sur **OK**.

h. Dans la liste déroulante, sélectionnez **Certificats de signataire**, puis cliquez sur **Ajouter** pour ajouter le fichier certDWCClient.arm créé dans 7l, à la page 275.

i. Entrez le libellé *DWCClientTrust* pour le certificat certDWCClient.arm. *DWCClientTrust* apparaît dans la liste **Certificats de signataire**.

j. Cliquez sur **OK**.

9. **Importez les certificats <REP_CERTS_TWS>\certAgentClient.arm, <REP_CERTS_TWS>\certAgentServer.arm et <REP_CERTS_DWC>\certDWCServer.arm dans les certificats signés AgentServerTrust comme décrit dans figure 6, à la page 263**:

a. Copiez le certificat <REP_CERTS_DWC>\certDWCServer.arm du poste de travail *DWC-WKS* vers le poste de travail *TWS-WKS* dans le répertoire <REP_CERTS_TWS>.

b. Cliquez sur **Open** (Ouvrir) pour ouvrir les certificats signés AgentServerTrust créés dans 2i, à la page 271.

c. Sur le panneau Open (Ouvrir), entrez les informations suivantes :

Type de base de données de clés
Sélectionnez la valeur type JKS.

Nom de fichier
Entrez <REP_CERTS_TWS>\ServerAgentTrust.jks

Emplacement
Insérez le nom de répertoire <*REP_CERTS_TWS*>.

d. Cliquez sur **OK**.

e. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.

f. Cliquez sur **OK**.

g. Sélectionnez les certificats signés AgentServerTrust créés dans 2i, à la page 271.

h. Cliquez sur **Ajouter** pour ajouter le <REP_CERTS_TWS>\certAgentClient.arm créé dans 3l, à la page 272.

i. Cliquez sur **OK**.

j. Cliquez sur **Ajouter** pour ajouter le <REP_CERTS_TWS>\certAgentServer.arm créé dans 1l, à la page 270.

k. Cliquez sur **OK**.

l. Cliquez sur **Ajouter** pour ajouter le <REP_CERTS_DWC>\certDWCServer.arm créé dans 5l, à la page 273.

m. Cliquez sur **OK**.

10. **Importez le certificat <REP_CERTS_TWS>\certAgentServer.arm dans les certificats signés AgentClientTrust comme décrit figure 6, à la page 263**, en procédant comme suit :

a. Cliquez sur **Open** (Ouvrir) pour ouvrir les certificats signés `AgentClientTrust` créés dans 4i, à la page 272.
b. Sur le panneau Open (Ouvrir), entrez les informations suivantes :

   Type de base de données de clés
   Sélectionnez la valeur type `JKS`.

   Nom de fichier
   Entrez `<REP_CERTS_TWS>\ClientAgentTrust.jks`

   Emplacement
   Insérez le nom de répertoire *<REP_CERTS_TWS>*.
c. Cliquez sur **OK**.
d. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.
e. Cliquez sur **OK**.
f. Dans la liste déroulante, sélectionnez **Certificats signés**.
g. Sélectionnez les certificats signés `AgentClientTrust` créés dans 4i, à la page 272.
h. Cliquez sur **Ajouter** pour ajouter le `<REP_CERTS_TWS>\certAgentServer.arm` créé dans 1l, à la page 270.
i. Cliquez sur **OK**.

11. **Importez les certificats <REP_CERTS_TWS>\certAgentServer.arm, <REP_CERTS_DWC>\certDWCServer.arm et <REP_CERTS_DWC>\certDWCClient.arm dans les certificats signés DWCServerTrust comme décrit dans figure 6, à la page 263**, en procédant comme suit :
    a. Copiez le certificat `<REP_CERTS_TWS>\certAgentServer.arm` du poste de travail *TWS-WKS* vers le poste de travail *DWC-WKS* dans le répertoire `<REP_CERTS_DWC>`.
    b. Cliquez sur **Open** (Ouvrir) pour ouvrir les certificats signés `DWCServerTrust` créés dans 6i, à la page 274.
    c. Sur le panneau Open (Ouvrir), entrez les informations suivantes :

       Type de base de données de clés
       Sélectionnez la valeur type `JKS`.

       Nom de fichier
       Entrez `<REP_CERTS_DWC>\ServerDWCTrust.jks`

       Emplacement
       Insérez le nom de répertoire *<REP_CERTS_DWC>*.
    d. Cliquez sur **OK**.
    e. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.
    f. Cliquez sur **OK**.
    g. Dans la liste déroulantes, sélectionnez **Certificats de signataire**.
    h. Sélectionnez les certificats signés `DWCServerTrust` créés dans 6i, à la page 274.
    i. Cliquez sur **Ajouter** pour ajouter le `<REP_CERTS_DWC>\certAgentServer.arm` créé dans 1l, à la page 270.
    j. Cliquez sur **OK**.
    k. Sélectionnez les certificats signés `DWCServerKey` créés dans 5l, à la page 273.
    l. Cliquez sur **Ajouter** pour ajouter le `<REP_CERTS_DWC>\certDWCServer.arm` créé dans 5l, à la page 273.

m. Cliquez sur **OK**.

n. Cliquez sur **Ajouter** pour ajouter le <REP_CERTS_DWC>\certDWCClient.arm créé dans 7l, à la page 275.

o. Cliquez sur **OK**.

12. **Importez le certificat <REP_CERTS_DWC>\certDWCServer.arm dans les certificats signés DWCClientTrust** comme décrit dans figure 6, à la page 263, en procédant comme suit :

    a. Cliquez sur **Open** (Ouvrir) pour ouvrir les certificats signés DWCClientTrust créés dans 8i, à la page 276.

    b. Sur le panneau Open (Ouvrir), entrez les informations suivantes :

    Type de base de données de clés
    Sélectionnez la valeur type JKS.

    Nom de fichier
    Entrez <REP_CERTS_DWC>\ClientDWCTrust.jks

    Emplacement
    Insérez le nom de répertoire *<REP_CERTS_DWC>*.

    c. Cliquez sur **OK**.

    d. Sur le panneau Password prompt insérez le mot de passe et confirmez-le. Par exemple, *passw0rd*.

    e. Cliquez sur **OK**.

    f. Dans la liste déroulantes, sélectionnez **Certificats de signataire**.

    g. Sélectionnez les certificats signés DWCClientTrust créés dans 8i, à la page 276.

    h. Cliquez sur **Ajouter** pour ajouter le <REP_CERTS_DWC>\certDWCServer.arm créé dans 5l, à la page 273.

    i. Cliquez sur **OK**.

13. **Configurez les nouveaux fichiers de clé serveur dans l'agent Tivoli Workload Scheduler avec un connecteur distribué** :

    a. Arrêtez WebSphere Application Server sur l'agent Tivoli Workload Scheduler avec un connecteur distribué. Pour plus d'informations sur cet utilitaire, voir *Guide d'administration> Tâches administratives > Tâches du serveur d'applications*.

    b. Exécutez le script suivant :

    Sur les système d'exploitation Windows :
    ```
    showSecurityProperties.bat > My_Security.prop
    ```

    Systèmes d'exploitation UNIX et Linux :
    ```
    showSecurityProperties.sh > My_Security.prop
    ```

    c. Dans le fichier My_Security.prop, section SSL Panel, insérez le nom keyFileName que vous avez créé dans 1d, à la page 269 et le nom trustFileName que vous avez créé dans 2d, à la page 270:

    Remarque : Utilisez / pour les système d'exploitation Windows et UNIX.

    ```
    ############################################################
    SSL Panel
    ############################################################
    alias=NodeDefaultSSLSettings
    keyFileName=
    <REP_CERTS_TWS>/ServerAgentKey.jks
    keyFilePassword=*****
    keyFileFormat=JKS
    trustFileName=
    ```

```
<REP_CERTS_TWS>/ServerAgentTrust.jks
trustFilePassword=*****
trustFileFormat=JKS
clientAuthentication=false
securityLevel=HIGH
enableCryptoHardwareSupport=false
```

Remarque :

- Sur les système d'exploitation UNIX et Windows, utilisez / dans le chemin d'accès `keyfilename` et `trustFilename`.
- Chiffrez le mot de passe à l'aide de l'utilitaire **encryptProfileProperties**. Pour plus d'informations sur cet utilitaire, voir *Guide d'administration> Tâches administratives > Tâches du serveur d'applications > et le chiffrement des fichiers de propriétés de profils pour plus de détails sur comment chiffrer des propriétés de profils*

d. Modifiez les propriétés de sécurité en exécutant le script suivant :

Sur les système d'exploitation Windows :

```
changeSecurityProperties.bat My_Security.prop
```

Systèmes d'exploitation UNIX et Linux :

```
changeSecurityProperties.sh My_Security.prop
```

14. **Configurez les nouveaux fichiers client dans l'agent Tivoli Workload Scheduler avec un connecteur distribué** :

a. Recherchez le fichier suivant :

Sur les système d'exploitation Windows :

```
<REP_INST_TWS>\eWas\profiles\TIPProfile\properties\
ssl.client.props
```

Systèmes d'exploitation UNIX et Linux :

```
<REP_INST_TWS>/eWas/profiles/TIPProfile/properties/
ssl.client.props
```

b. Dans le fichier `ssl.client.props`, modifiez les sections KeyStore information et TrustStore information, en insérant les valeurs suivantes :

```
# KeyStore information
com.ibm.ssl.keyStoreName=ClientDefaultKeyStore
com.ibm.ssl.keyStore=<REP_CERTS_TWS>/ClientAgentKey.jks
com.ibm.ssl.keyStorePassword=passw0rd
com.ibm.ssl.keyStoreType=JKS
com.ibm.ssl.keyStoreProvider=IBMJCE
com.ibm.ssl.keyStoreFileBased=true

# TrustStore information
com.ibm.ssl.trustStoreName=ClientDefaultTrustStore
com.ibm.ssl.trustStore=<REP_CERTS_TWS>/ClientAgentTrust.jks
com.ibm.ssl.trustStorePassword=passw0rd
com.ibm.ssl.trustStoreType=JKS
com.ibm.ssl.trustStoreProvider=IBMJCE
com.ibm.ssl.trustStoreFileBased=true
com.ibm.ssl.trustStoreReadOnly=false
```

où

com.ibm.ssl.keyStore
Insérez le fichier `<REP_CERTS_TWS>/ClientAgentKey.jks` que vous avez généré dans 3d, à la page 271.

com.ibm.ssl.keyStorePassword
Insérez la valeur de mot de passe que vous avez utilisée dans 3f, à la page 271.

com.ibm.ssl.trustStore
Insérez le fichier <REP_CERTS_TWS>/ClientAgentTrust.jks que vous avez généré dans 4d, à la page 272.

com.ibm.ssl.trustStorePassword
Insérez la valeur de mot de passe que vous avez utilisée dans 4f, à la page 272.

Remarque :

- Sur les système d'exploitation UNIX et Windows, utilisez / dans le chemin d'accès keyfilename et trustFilename.
- Chiffrez le mot de passe à l'aide de l'utilitaire **encryptProfileProperties**. Pour plus d'informations sur cet utilitaire, voir *Guide d'administration> Tâches administratives > Tâches du serveur d'applications > et le chiffrement des fichiers de propriétés de profils pour plus de détails sur comment chiffrer des propriétés de profils*

c. Démarrez WebSphere Application Server sur l'agent Tivoli Workload Scheduler avec connecteur distribué. Pour plus d'informations sur cet utilitaire, voir *Guide d'administration> Tâches administratives > Tâches du serveur d'applications*.

15. **Configurez les nouveaux fichiers de clés serveur dans Dynamic Workload Console** :

a. Arrêtez WebSphere Application Server sur Dynamic Workload Console. Pour plus d'informations sur cet utilitaire, voir *Guide d'administration> Tâches administratives > Tâches du serveur d'applications*.

b. Exécutez le script suivant :

Sur les système d'exploitation Windows :
```
showSecurityProperties.bat > My_Security.prop
```

Systèmes d'exploitation UNIX et Linux :
```
showSecurityProperties.sh > My_Security.prop
```

c. Dans le fichier My_Security.prop, section SSL Panel, insérez le nom keyFileName que vous avez créé dans 5d, à la page 273 et le nom trustFileName que vous avez créé dans 6d, à la page 274:

```
#############################################################
SSL Panel
#############################################################
alias=NodeDefaultSSLSettings
keyFileName=
<REP_CERTS_TWS>/ServerDWCKey.jks
keyFilePassword=passw0rd
keyFileFormat=JKS
trustFileName=
<REP_CERTS_TWS>/ServerDWCTrust.jks
trustFilePassword=passw0rd
trustFileFormat=JKS
clientAuthentication=false
securityLevel=HIGH
enableCryptoHardwareSupport=false
```

Remarque :

- Sur les système d'exploitation UNIX et Windows, utilisez / dans le chemin d'accès keyfilename et trustFilename.
- Chiffrez le mot de passe à l'aide de l'utilitaire **encryptProfileProperties**. Pour plus d'informations sur cet utilitaire, voir *Guide d'administration> Tâches administratives > Tâches du serveur*

d'applications > et le chiffrement des fichiers de propriétés de profils pour plus de détails sur comment chiffrer des propriétés de profils

d. Modifiez les propriétés de sécurité en exécutant le script suivant :

Sur les système d'exploitation Windows :

```
changeSecurityProperties.bat My_Security.prop
```

Systèmes d'exploitation UNIX et Linux :

```
changeSecurityProperties.sh My_Security.prop
```

16. **Configurez les nouveaux fichiers client dans Dynamic Workload Console**:

a. Recherchez le fichier suivant :

Sur les système d'exploitation Windows :

```
<REP_INST_DWC>\eWas\profiles\TIPProfile\properties\
ssl.client.props
```

Systèmes d'exploitation UNIX et Linux :

```
<REP_INST_DWC>/eWas/profiles/TIPProfile/properties/
ssl.client.props
```

b. Dans le fichier `ssl.client.props`, modifiez les sections KeyStore information et TrustStore information, en insérant les valeurs suivantes :

```
# KeyStore information
com.ibm.ssl.keyStoreName=ClientDefaultKeyStore
com.ibm.ssl.keyStore=<REP_CERTS_DWC>/ClientDWCKey.jks
com.ibm.ssl.keyStorePassword=passw0rd
com.ibm.ssl.keyStoreType=JKS
com.ibm.ssl.keyStoreProvider=IBMJCE
com.ibm.ssl.keyStoreFileBased=true

# TrustStore information
com.ibm.ssl.trustStoreName=ClientDefaultTrustStore
com.ibm.ssl.trustStore=<REP_CERTS_DWC>/ClientDWCTrust.jks
com.ibm.ssl.trustStorePassword=passw0rd
com.ibm.ssl.trustStoreType=JKS
com.ibm.ssl.trustStoreProvider=IBMJCE
com.ibm.ssl.trustStoreFileBased=true
com.ibm.ssl.trustStoreReadOnly=false
```

où

com.ibm.ssl.keyStore
Insérez le fichier `<REP_CERTS_DWC>/ClientDWCKey.jks` que vous avez généré dans 7d, à la page 274.

com.ibm.ssl.keyStorePassword
Insérez la valeur de mot de passe que vous avez utilisée dans 7f, à la page 275.

com.ibm.ssl.trustStore
Insérez le fichier `<REP_CERTS_DWC>/ClientDWCTrust.jks` que vous avez généré dans 8d, à la page 275.

com.ibm.ssl.trustStorePassword
Insérez la valeur de mot de passe que vous avez utilisée dans 8f, à la page 276.

Remarque :

- Sur les système d'exploitation UNIX et Windows, utilisez / dans le chemin d'accès `keyfilename` et `trustFilename`.
- Chiffrez le mot de passe à l'aide de l'utilitaire **`encryptProfileProperties`**. Pour plus d'informations sur cet utilitaire,

voir *Guide d'administration> Tâches administratives > Tâches du serveur d'applications > et le chiffrement des fichiers de propriétés de profils pour plus de détails sur comment chiffrer des propriétés de profils*

c. Démarrez WebSphere Application Server sur Dynamic Workload Console. Pour plus d'informations sur cet utilitaire, voir *Guide d'administration> Tâches administratives > Tâches du serveur d'applications*.

Scénario : connexion entre l'agents dynamiques et le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique

Les certificats par défaut fournis au cours du processus d'installation de Tivoli Workload Scheduler assurent une connexion sécurisée entre les composants suivants :

Gestionnaire de domaine maître et gestionnaire de domaine dynamique ou gestionnaire de domaine dynamique de sauvegarde :

La connexion au serveur du courtier installé avec le gestionnaire de domaine dynamique nécessite l'utilisation de certificats à partir d'une autorité de certification pour permettre l'authentification. En outre, les gestionnaires de domaine maîtres et les gestionnaires de domaine maîtres de sauvegarde qui communiquent avec le gestionnaire de domaine dynamique ou sa sauvegarde doivent être définis sur le serveur du courtier associé pour garantir une autorisation par rôle.

L'gestionnaire de domaine dynamique communique avec tous les composants suivants :

- Gestionnaire de domaine maître
- gestionnaire de domaine maître de sauvegarde (le cas échéant)
- gestionnaire de domaine dynamique de sauvegarde (le cas échéant)

Gestionnaire de domaine maître ou gestionnaire de domaine dynamique et agents dynamiques :

L'agent dynamique communique avec tous les composants suivants :

- Gestionnaire de domaine maître
- gestionnaire de domaine maître de sauvegarde (le cas échéant)
- Gestionnaire de domaine dynamique (le cas échéant)
- gestionnaire de domaine dynamique de sauvegarde (le cas échéant)

Par défaut, la communication entre des agents dynamiques et un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique auprès duquel ils sont enregistrés s'effectue en protocole HTTPS. Ce type de communication utilise les certificats par défaut du produit.

La ligne de commande ResourceCLI et le serveur de courtier sont installés sur le gestionnaire de domaine maître

Vous pouvez activer la communication entre la ligne de commande **ResourceCLI** installée sur le gestionnaire de domaine dynamique et le serveur de courtier installé sur le gestionnaire de domaine maître à l'aide de certificats par défaut ou de vos propres certificats. Voir «Personnalisation de la connexion SSL entre un gestionnaire de domaine maître et une ligne de commande de ressource», à la page 285.

Personnalisation de la connexion SSL entre un gestionnaire de domaine maître et un gestionnaire de domaine dynamique ou son gestionnaire de secours à l'aide de vos certificats

Personnalisation de la connexion SSL entre un gestionnaire de domaine maître et un gestionnaire de domaine dynamique ou son gestionnaire de secours à l'aide de vos certificats.

La connexion au serveur du courtier installé avec le gestionnaire de domaine dynamique nécessite l'utilisation de certificats à partir d'une autorité de certification pour permettre l'authentification. En outre, les gestionnaires de domaine maîtres et les gestionnaires de domaine maîtres de secours qui communiquent avec le gestionnaire de domaine dynamique ou le gestionnaires de domaine dynamique de sauvegarde doivent être définis sur le serveur du courtier associé pour garantir l'autorisation en fonction de rôles.

Les exemples présentés dans cette section se réfèrent à un gestionnaire de domaine dynamique qui communique avec un gestionnaire de domaine maître, mais la même configuration s'applique lorsque le gestionnaire de domaine dynamique communique avec l'un des composants suivants :

- Gestionnaire de domaine maître
- gestionnaire de domaine maître de sauvegarde (le cas échéant)
- gestionnaire de domaine dynamique de sauvegarde (le cas échéant)

Si vous utilisez les certificats par défaut installés avec le produit, la communication entre tous les composants est automatiquement assurée.

Lorsque vous installez Tivoli Workload Scheduler, les certificats par défaut fournis garantissent une authentification correcte et une autorisation en fonction de rôles entre les composants. La valeur par défaut du certificat est `Server` sur le gestionnaire de domaine maître.

Si vous prévoyez d'utiliser vos certificats au lieu des certificats par défaut, suivez la procédure décrite dans la section ci-dessous pour activer la communication entre des composants.

Par exemple, la procédure suivante active la communication entre un gestionnaire de domaine maître et un gestionnaire de domaine dynamique.

Procédure

Procédure pour activer la communication entre un gestionnaire de domaine maître et un gestionnaire de domaine dynamique :

1. Modifiez le certificat sur le gestionnaire de domaine maître. Par exemple, cette procédure suppose que le nom commun présent dans le certificat sur le gestionnaire de domaine maître est `mdm1`.
2. Déployez le certificat sur le gestionnaire de domaine maître et le gestionnaire de domaine dynamique, comme indiqué à la section Chapitre 7, «Définition de la sécurité des connexions», à la page 261.
3. Modifiez la liste des noms communs sur le gestionnaire de domaine dynamique, comme suit :
   a. Accédez au répertoire *TWA_home*/TDWB/config.
   b. Ouvrez le fichier `BrokerWorkstation.properties`.

c. Dans l'option `Broker.AuthorizedCNs`, définissez le nom commun du gestionnaire de domaine maître autorisé. Dans cet exemple

```
Broker.AuthorizedCNs=mdm1
```

Si vous voulez activer la communication avec plusieurs gestionnaires de domaine maîtres, séparez chaque valeur par un point-virgule (;). Par exemple, vous pouvez définir la liste suivante :

```
Broker.AuthorizedCNs=mdm;mdm1;mdm2
```

Cette liste garantit que tous les gestionnaires de domaine maîtres portant ces noms communs peuvent se connecter au gestionnaire de domaine dynamique.

4. Arrêtez, puis redémarrez le gestionnaire de domaine dynamique comme suit pour que la modification prenne effet :
   a. Utilisez l'outil was **stopBrokerApplication.sh** sous UNIX et Linux ou **stopBrokerApplication.bat** sous Windows :

```
stopBrokerApplication -user username -password password [-port portnumber]
```

où *username* et *password* correspondent aux données d'identification utilisées lors de l'installation. Le paramètre *port_number* est facultatif. S'il n'est pas spécifié, la valeur par défaut est utilisée.

   b. Utilisez l'outil was **startBrokerApplication.sh** sous UNIX et Linux ou **startBrokerApplication.bat** sous Windows :

```
startBrokerApplication -user username -password password [-port portnumber]
```

où *username* et *password* correspondent aux données d'identification utilisées lors de l'installation. Le paramètre *portnumber* est facultatif. S'il n'est pas spécifié, la valeur par défaut est utilisée.

Pour plus d'informations, voir «Fichier BrokerWorkstation.properties», à la page 75 et «Démarrage, arrêt et affichage du statut de Dynamic Workload Broker», à la page 407.

Personnalisation de la connexion SSL entre des agents dynamiques et un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique à l'aide de vos certificats

Personnalisation de la connexion SSL entre un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique et des agents dynamiques connectés à ce gestionnaire à l'aide de vos certificats.

Par défaut, la communication entre des agents dynamiques et un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique auprès duquel ils sont enregistrés s'effectue en protocole HTTPS. Ce type de communication utilise les certificats par défaut du produit. Si vous voulez utiliser vos propres certificats personnalisés pour cette communication parce que vous avez personnalisé les certificats du gestionnaire de domaine maître ou du gestionnaire de domaine dynamique, vous devez personnaliser la configuration et les certificats d'agent. Pour activer la communication entre des agents dynamiques et un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique, procédez comme suit :

1. Générez un fichier de clés CMS `.kdb`. Ce fichier doit contenir une clé privée sécurisée par le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique auprès duquel l'agent est enregistré agent, ainsi que la clé publique du gestionnaire de domaine maître ou du gestionnaire de domaine dynamique de sorte que l'agent puisse leur faire confiance.

2. Enregistrez le mot de passe du fichier de clés dans un fichier de dissimulation de même nom que le fichier généré à l'étape 1, à la page 284 et avec l'extension `.sth`.
3. Ouvrez le fichier de configuration d'agent `ita.ini` et définissez les valeurs propres à de votre environnement pour les propriétés suivantes :

   ```
   cert_label=<étiquette_clé_privée_agent>
   key_db_name=<file_name>
   key_repository_dir=<répertoire>
   ```

   Où :

   étiquette_clé_privée_agent
   Est l'étique de la clé privée de l'agent que vous voulez utiliser pour la communication. La valeur par défaut est **client**.

   file_name
   Spécifie le nom du fichier sans l'extension. La valeur par défaut est **TWSClientKeyStore**.

   répertoire
   Spécifie le répertoire contenant les fichiers générés à l'étape 1, à la page 284 and in Step 2. Le chemin d'accès par défaut est `/opt/IBM/TWA/TWS/ITA/cpa/ita/cert`.
4. Arrêtez l'agent IBM i en exécutant la commande suivante :

   ```
   ShutDownLwa
   ```
5. Démarrez l'agent IBM i exécutant la commande suivante :

   ```
   StartUpLwa
   ```

Personnalisation de la connexion SSL entre un gestionnaire de domaine maître et une ligne de commande de ressource

Personnalisation de la connexion SSL entre un gestionnaire de domaine maître et la ligne commande de ressource.

La communication entre la ligne de commande de ressource et le gestionnaire de domaine maître s'effectue par défaut en protocole HTTP. Si vous voulez communiquer en protocole HTTPS, vous pouvez utiliser les certificats par défaut ou vos propres certificate.

Pour utiliser les certificats par défaut, procédez comme suit :

1. Trouvez la valeur définie pour le port WebSphere Application Server du gestionnaire de domaine maître dans la propriété **httpsPort**, en exécutant l'outil was **showHostProperties**. La valeur par défaut est **31116**.
2. Ouvrez le fichier `TWS/TDWB_CLI/config/CLIConfig.properties`.
3. Définissez la propriété **ITDWBServerSecurePort** sur cette valeur. Par exemple, si vous avez utilisez le port **31116**, entrez :

   ```
   ITDWBServerSecurePort=31116
   ```
4. Définissez la propriété **use_secure_connection** sur **true**, en entant

   ```
   use_secure_connection=true
   ```

Ainsi, vous utilisez les certificats par défaut fournis avec le produit et stockés dans le fichier de clés et le fichier de clés certifiées suivants sur le agent sur lequel vous utilisez la ligne de commande de ressource :

```
keyStore=TWS_inst_dir/TWS/TDWB_CLI/certs/TWSClientKeyFile.jks
trustStore=TWS_inst_dir/TWS/TDWB_CLI/certs/TWSClientTrustFile.jks
```

Si vous voulez utiliser vos propres certificats, exécutez les étapes 1 à 4, puis procédez comme suit :

- Remplacez les certificats par défaut présent sur présents sur l'agent par les certificats personnalisés présents sur le gestionnaire de domaine maître. Les certificats gestionnaire de domaine maître se trouvent dans le répertoire *<chemin_profil_WAS>*/etc où la valeur par défaut pour *<chemin_profil_WAS>* est *<TWA_home>*/WAS/TWSprofile. Vérifiez que les propriétés keyStore et trustStore de l'agent pointent sur les certificats appropriés. Par exemple, si vous avez enregistré le gestionnaire de domaine maître dans le répertoire tmp du fichier de clés (keyStore) et le fichier de clés certifiées (trustStore) de l'agent, entrez :

  ```
  keyStore=tmp/TWS/TDWB_CLI/certs/TWSClientKeyFile.jks
  trustStore=tmp/TWS/TDWB_CLI/certs/TWSClientTrustFile.jks
  ```

Scénario : communication SSL sur le réseau Tivoli Workload Scheduler

Vous pouvez activer la connexion SSL à l'aide de OpenSSL Toolkit pour les composants suivants :

- Gestionnaire de domaine maître et ses gestionnaires de domaine
- Gestionnaire de domaine maître et agents tolérants aux pannes et dans le domaine maître
- Gestionnaire de domaine maître et gestionnaire de domaine maître de sauvegarde
- Gestionnaire de domaine et agents tolérants aux pannes appartenant à ce domaine

Les certificats par défaut se trouvent dans le répertoire <INSTALL_DIR>\TWS\ssl\OpenSSL.

Utilisation de SSL pour netman et conman

Tivoli Workload Scheduler propose un mécanisme de connexion sécurisé, authentifié et chiffré pour établir des communications dans la topologie réseau. Ce mécanisme est basé sur le protocole Secure Sockets Layer (SSL) et utilise le kit d'outils OpenSSL, lequel est automatiquement installé avec Tivoli Workload Scheduler.

Le protocole SSL repose sur une méthodologie de clé privée et publique. SSL propose les méthodes d'authentification suivantes :

Sécurisation des autorités de certification uniquement
Deux postes de travail se font confiance si chacun reçoit de l'autre un certificat signé ou sécurisé, c'est-à-dire si le certificat figure dans la liste des autorités de certification sécurisées sur chaque poste de travail. Avec ce niveau d'authentification, les postes de travail n'effectuent pas de contrôles supplémentaires sur le contenu des certificats, tel que le nom distinctif. N'importe quel certificat signé ou sécurisé peut être utilisé pour établir une session SSL. Pour une définition de l'option **caonly** utilisée par le mot clé **ssl auth mode**, voir «Définition des options locales», à la page 30.

Vérifier si le nom distinctif correspond à une chaîne définie
Deux postes de travail se font confiance si, après avoir reçu un certificat sécurisé ou signé, chacun d'entre eux effectue un contrôle supplémentaire en extrayant le nom distinctif du certificat et en le comparant à une chaîne

qui a été définie dans son fichier d'options locales. Pour obtenir une définition de l'option **string**, voir «Définition des options locales», à la page 30.

Vérifier si le nom distinctif correspond au nom du poste de travail
Deux postes de travail se font confiance si, après avoir reçu un certificat signé ou sécurisé, chacun d'entre eux effectue un contrôle supplémentaire en extrayant le nom distinctif du certificat et en le comparant au nom du poste de travail qui a envoyé le certificat. Pour obtenir une définition de l'option **cpu**, voir «Définition des options locales», à la page 30.

Pour fournir la sécurité SSL à un gestionnaire de domaine attaché à z/OS dans une connexion de bout en bout, configurez le SSL du système de services cryptographiques OS/390 dans le code Tivoli Workload Scheduler qui s'exécute dans le shell OS/390 USS UNIX dans l'espace d'adresse du serveur Tivoli Workload Scheduler for z/OS. Consultez la documentation z/OS relative à Tivoli Workload Scheduler.

Lors de la configuration du protocole SSL, les possibilités suivantes sont à votre disposition :

Utilisation du même certificat pour l'ensemble du réseau
Si les postes de travail sont configurés avec la sécurisation des autorités de certification uniquement, ils acceptent les connexions avec tout autre poste de travail qui transmet un certificat signé ou sécurisé. Pour faire appliquer l'authentification, vous pouvez définir, dans le fichier `localopts` de chaque poste de travail, un nom ou une liste de noms qui doit correspondre au contenu de la zone du nom distinctif du certificat.

Utiliser un certificat pour chaque domaine
Installez les clés privées et les certificats signés de chaque domaine du réseau. Configurez ensuite chaque poste de travail pour accepter une connexion uniquement avec les partenaires qui disposent d'une chaîne spécifique dans la zone DN (nom distinctif) de leur certificat, au sein du fichier `localopts` de chaque poste de travail.

Utiliser un certificat pour chaque poste de travail
Installez une clé différente et un certificat signé sur chaque poste de travail, et ajoutez une liste des autorités de certification sécurisées contenant l'autorité de certification qui a signé le certificat. Configurez ensuite chaque poste de travail pour accepter une connexion uniquement avec les partenaires dont le nom de poste de travail indiqué dans le fichier `Symphony` est enregistré dans la zone DN du certificat.

Configuration de clés privées et de certificats

Pour utiliser l'authentification SSL sur un poste de travail, vous devez créer et installer les éléments suivants :

- Clé privée et certificat correspondant permettant d'identifier le poste de travail dans une session SSL
- Liste des autorités de certification pouvant être sécurisées par le poste de travail

Utilisez l'utilitaire de ligne de commande **openssl** pour :

- Créer un fichier contenant des octets générés pseudo-aléatoires (TWS.rnd). Certains systèmes d'exploitation requièrent le fichier pour assurer un fonctionnement correct de la fonction SSL.
- Créer une clé privée.

- Sauvegarder dans un fichier le mot de passe que vous avez utilisé pour créer la clé.
- Créer une requête de signature de certificat.
- Envoyer cette requête à une autorité de certification qui se chargera de la signer ou :
  - Créer votre propre autorité de certification.
  - Créer un certificat d'autorité de certification autosigné (structure X.509) avec la clé de zone RSA de votre propre autorité de certification.
  - Utiliser votre propre autorité de certification pour signer et créer des certificats réels.

Cette procédure permet de générer les fichiers suivants que vous allez installer sur le ou les postes de travail :

- Fichier de clés privées (par exemple, TWS.key). Ce fichier doit être protégé, de manière à ne pas être volé pour utiliser l'identité du poste de travail. Vous devriez l'enregistrer dans un répertoire permettant un accès en lecture à l'utilisateur TWS du poste de travail, par exemple *TWA_home*/TWS/ssl/TWS.key.
- Fichier de certificat correspondant (par exemple, TWS.crt). Vous devriez l'enregistrer dans un répertoire qui accorde également l'accès en lecture à l'utilisateur TWS du poste de travail, tel que *TWA_home*/TWS/ssl/TWS.crt.
- Fichier contenant une séquence d'octets pseudo-aléatoire. Vous pouvez l'enregistrer dans tout répertoire accordant l'accès en lecture à l'utilisateur TWS du poste de travail, tel que *TWA_home*/TWS/ssl/TWS.rnd.

En outre, vous devez créer les fichiers suivants :

- Fichier contenant le mot de passe permettant de chiffrer la clé privée. Vous devriez l'enregistrer dans un répertoire qui accorde l'accès en lecture à l'utilisateur de TWS du poste de travail, tel que *TWA_home*/TWS/ssl/TWS.sth.
- Fichier de hiérarchie de certificats. Il contient la concaténation des certificats d'autorités de certification codés selon la norme PEM, qui forment la hiérarchie du certificat du poste de travail. Cette action commence par la délivrance du certificat d'autorité de certification du poste de travail et peut aller jusqu'au certificat d'autorité de certification de niveau racine. Ce fichier constitue tout simplement la concaténation des différents fichiers de certificat d'autorités de certification codés selon la norme PEM, généralement par ordre de hiérarchie de certificats.
- Fichier d'autorité de certification sécurisé. Il contient les certificats d'autorités de certification sécurisés à utiliser pendant l'authentification. Les CA figurant dans ce fichier sont également utilisées pour créer la liste des CA client acceptables communiquées au client lorsque le côté serveur de la connexion demande un certificat de client. Ce fichier constitue tout simplement la concaténation des différents fichiers de certificat d'autorités de certification codés selon la norme PEM, par ordre de préférence.

Création de votre propre autorité de certification

Si vous avez l'intention d'utiliser l'authentification SSL dans les limites de votre entreprise et non pour le commerce avec l'extérieur sur Internet, il peut être plus simple de créer votre propre autorité de certification (CA) pour faire confiance à toutes vos installations Tivoli Workload Scheduler. Pour ce faire, procédez comme suit.

Remarque : Dans les étapes suivantes, les noms des fichiers créés au cours du processus TWS et TWSca sont des exemples. Vous pouvez utiliser vos propres noms, mais conservez les mêmes extensions de fichier.

1. Choisissez un poste de travail comme votre installation CA root.
2. Saisissez la commande suivante à partir du répertoire SSL pour initialiser le générateur de nombre pseudo-aléatoire. Sinon, les commandes que vous taperez par la suite risquent de ne pas fonctionner.
   - Sous UNIX :

     ```
     $ openssl rand -out TWS.rnd -rand ./openssl 8192
     ```
   - Sous Windows :

     ```
     $ openssl rand -out TWS.rnd -rand ./openssl.exe 8192
     ```
3. Entrez la commande suivante pour créer la clé privée de l'autorité de certification (CA) :

   ```
   $ openssl genrsa -out TWSca.key 2048
   ```
4. Entrez la commande suivante pour créer un certificat de l'autorité de certification (CA) autosigné (structure X.509) :

   ```
   $ openssl req -new -x509 -days 365 -key TWSca.key -out TWSca.crt -config ./openssl.cnf
   ```

Vous disposez maintenant d'une autorité de certification que vous pouvez utiliser pour accorder votre confiance à toutes vos installations. Si vous le souhaitez, vous pouvez créer plusieurs autorités de certification.

Création de clés privées et de certificats

La procédure ci-dessous permet de créer une clé et un certificat. Vous pouvez décider d'utiliser une paire clé et certificat pour l'ensemble du réseau, une pour chaque domaine ou une pour chaque poste de travail. Pour la procédure ci-dessous, nous considérons que vous allez créer une paire clé et certificat pour chaque poste de travail. Par conséquent, le nom générique *workstation_name* a été attribué aux fichiers de sortie créés au cours du processus.

Sur chaque poste de travail, créez une clé privée et un certificat :

1. Entrez la commande suivante à partir du répertoire SSL pour initialiser le générateur de nombre pseudo-aléatoire. Sinon, les commandes que vous taperez par la suite risquent de ne pas fonctionner.
   - Sur les systèmes d'exploitation Windows :

     ```
     $ openssl rand -out workstationname.rnd -rand ./openssl.exe 8192
     ```
   - Sur les systèmes d'exploitation UNIX and Linux :

     ```
     $ openssl rand -out workstationname.rnd -rand ./openssl 8192
     ```
2. Tapez la commande suivante pour créer la clé privée (cet exemple illustre le chiffrage DES triple) :

   ```
   $ openssl genrsa -des3 -out workstationname.key 2048
   ```

   Ensuite, enregistrez le mot de passe qui a été demandé pour chiffrer la clé dans un fichier nommé *workstation_name*.pwd.

   Remarque : Vérifiez que le fichier *workstation_name*.pwd contient uniquement les caractères du mot de passe. Par exemple, si vous avez indiqué le mot de passe*maestro*, votre fichier *workstation_name*.pwd ne doit contenir aucun caractère CR ou LF à la fin (il doit contenir 7 octets).
3. Créez un fichier de dissimulation. Vous pouvez choisir de créer un fichier de dissimulation ou chiffrer votre fichier de mot de passe :

Fichier de dissimulation

Tapez la commande suivante pour enregistrer votre mot de passe en le codant en base 64 dans le fichier de dissimulation approprié :

```
$ openssl base64 -in workstation_name.pwd -out workstation_name.sth
```

Vous pouvez alors supprimer le fichier *workstation_name*.pwd.

Fichier de mot de passe chiffré

Exécutez la commande suivante pour enregistrer votre mot de passe chiffré avec un code en base 64 :

```
$ conman crypt workstation_name.pwd
```

Exemple : si vous disposez du fichier *workstation_name*.pwd qui contient la chaîne *secreat*, mot de passe que vous avez défini, après avoir exécuté la commande `$ conman crypt workstation_name.pwd`, votre fichier *workstation_name*.pwd contient la chaîne *{3DES}poh56FeTy+=/jhtf2djur* qui est le mot de passe chiffré.

4. Tapez la commande suivante pour créer une requête de signature de certificat :

   ```
   $ openssl req -new -key workstation_name.key -out workstation_name.csr
      -config ./openssl.cnf
   ```

   Le programme vous invite à préciser certaines valeurs (nom de l'entreprise, de l'utilisateur, etc.). Pour une compatibilité future, vous pouvez indiquer le nom du poste de travail comme nom distinctif.

5. Envoyez le fichier *workstation_name*.csr à votre autorité de certification afin d'obtenir le certificat correspondant à cette clé privée.

   A l'aide de sa clé privée (TWSca.key) et de son certificat (TWSca.crt), l'autorité de certification va signer la requête de signature de certificat (*workstation_name*.csr) et créer un certificat privé (*workstation_name*.crt) via la commande suivante :

   ```
   $ openssl x509 -req -CA TWSca.crt -CAkey TWSca.key -days 365
      -in workstation_name.csr   -out workstation_name.crt -CAcreateserial
   ```

6. Distribuez au poste de travail le nouveau certificat *workstation_name*.crt et le certificat public d'autorité de certification TWSca.crt.

Le tableau ci-dessous récapitule les fichiers créés au cours du processus qui doivent être définis comme les valeurs des options locales du poste de travail.

Tableau 56. Fichiers des options locales

Option locale	Fichier
SSL key	*workstation_name*.key
SSL certificate	*workstation_name*.crt
SSL key pwd	*workstation_name*.sth
SSL ca certificate	TWSca.crt
SSL random seed	*workstation_name*.rnd

Configuration des attributs SSL

Utilisez la ligne de commande **composer** ou Dynamic Workload Console pour mettre à jour la définition du poste de travail dans la base de données. Pour plus d'informations, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Configurez les attributs suivants :

secureaddr

Définit le port utilisé pour détecter les connections SSL entrantes. Cette valeur doit correspondre à celle définie dans l'option locale **nm SSL port** du poste de travail. Elle doit être différente de l'option locale **nm port** qui définit le port utilisé pour les communications normales. Si **securitylevel** est indiqué mais si cet attribut est manquant, 31113 est utilisé comme valeur par défaut.

securitylevel

Indique le type d'authentification SSL pour le poste de travail. Cet attribut doit être défini sur l'une des valeurs suivantes :

activée
Le poste de travail utilise l'authentification uniquement si le poste de travail de son gestionnaire de domaine ou un autre agent tolérant aux pannes situé en-dessous dans la hiérarchie des domaines l'exige.

sur Le poste de travail utilise l'authentification SSL lorsqu'il se connecte à son gestionnaire de domaine. Le gestionnaire de domaine l'utilise lorsqu'il se connecte à son gestionnaire de domaine parent. L'agent tolérant aux pannes refuse toute connexion entrante autre que SSL à partir de son gestionnaire de domaine.

force Le poste de travail utilise l'authentification SSL pour toutes ses connexions et accepte les connexions à la fois du gestionnaire de domaine parent et du gestionnaire de domaine subordonné. Il refuse toutes les connexions entrantes non SSL.

Si cet attribut fait défaut, le poste de travail n'est pas configuré pour les connexions SSL. Dans ce cas, toutes les valeurs de **secureaddr** sont ignorées. Vous devez également définir l'option locale **nm ssl port** sur 0 pour vous assurer que ce port n'est pas ouvert par netman. Le tableau suivant décrit le type de communication utilisé pour chaque type de configuration **securitylevel**.

Tableau 57. Type de communication en fonction de la valeur securitylevel

Agent tolérant aux pannes (gestionnaire de domaine)	Gestionnaire de romaine (gestionnaire de domaine parent)	Type de connexion
-	-	TCP/IP
Activé	-	TCP/IP
Actif	-	Pas de connexion
Force	-	Pas de connexion
-	Actif	TCP/IP
Activé	Actif	TCP/IP
Actif	Actif	SSL
Force	Actif	SSL
-	Activé	TCP/IP
Activé	Activé	TCP/IP
Actif	Activé	SSL
Force	Activé	SSL
-	Force	Pas de connexion
Activé	Force	SSL

Tableau 57. Type de communication en fonction de la valeur securitylevel (suite)

Agent tolérant aux pannes (gestionnaire de domaine)	Gestionnaire de romaine (gestionnaire de domaine parent)	Type de connexion
Actif	Force	SSL
Force	Force	SSL

L'exemple suivant illustre une définition de poste de travail, qui comprend les attributs de sécurité suivants :

```
cpuname MYWIN
os WNT
node apollo
tcpaddr 30112
secureaddr 32222
for maestro
autolink off
fullstatus on
securitylevel on
end
```

Configuration du protocole de connexion SSL pour le réseau

Pour configurer SSL dans votre réseau, procédez comme suit :

1. Créez un répertoire SSL sous le répertoire *TWA_home*/TWS. Par défaut, le chemin *TWA_home*/TWS/ssl est enregistré dans le fichier localopts. Si vous créez un répertoire avec un nom différent de ssl dans le répertoire *TWA_home*/TWS, mettez à jour le fichier localopts en conséquence.
2. Copiez openssl.cnf et openssl.exe dans le répertoire SSL.
3. Créez le nombre de clés privées, de certificats et d'autorités de certification sécurisées que vous envisagez d'utiliser dans votre réseau.
4. Sur chaque poste de travail qui utilisera l'authentification SSL :
   - Mettez à jour sa définition dans la base de données Tivoli Workload Scheduler avec les attributs SSL.
   - Ajoutez les options locales SSL dans le fichier localopts.

Même si vous n'êtes pas obligé de suivre une séquence spécifique, ces tâches doivent toutes être effectuées pour activer la prise en charge de SSL.

Dans Tivoli Workload Scheduler, la prise en charge de SSL est disponible uniquement pour les agents tolérants aux pannes (notamment le gestionnaire de domaine maître et les gestionnaires de domaine), mais pas pour les agents étendus. Si vous voulez utiliser l'authentification SSL pour un poste de travail exécutant un agent étendu, vous devez indiquer ce paramètre dans la définition du poste de travail hôte de l'agent étendu.

Configuration de la sécurité SSL totale

La présente section explique comment implémenter une sécurité SSL totale lors de l'utilisation d'une connexion SSL pour les communications via le réseau par **netman** et **conman**. Elle contient les rubriques suivantes :

- «Présentation», à la page 293
- «Configuration de la sécurité SSL complète», à la page 293
- «Migration d'un réseau vers la sécurité complète des connexions SSL», à la page 294
- «Configuration de la prise en charge SSL pour les dépendances inter-réseau», à la page 295

Remarque : La fonction complète de la sécurité SSL ne s'applique pas à la communication entre les agents dynamiques et le poste de travail courtier défini pour gestionnaire de domaine maître ou le gestionnaire de domaine dynamique auquel les agents dynamiques sont connectés.

Présentation : Cette fonction permet de définir un degré supérieur de sécurité des connexions SSL sur les réseaux Tivoli Workload Scheduler par rapport au niveau de sécurité SSL déjà disponible.

Si vous avez besoin d'un niveau de protection SSL plus complet, cette amélioration fournit de nouvelles options de configuration permettant de configurer la sécurité avancée des connexions.

Si vous n'avez pas besoin d'une sécurité SSL supérieure à celle que Tivoli Workload Scheduler offrait avant la publication de cette fonction, vous pouvez utiliser les paramètres standard documentés ci-dessus dans ce chapitre.

L'ensemble des améliorations de la sécurité SSL : La prise en charge totale de la sécurité SSL fournit les améliorations suivantes :

- Les ports TCP qui pourraient poser des problèmes de sécurité ne restent plus ouverts.
- Les données qui sont transférées, y compris les en-têtes et les bas de page des communications, sont désormais *totalement* chiffrées.

Compatibilité entre les niveaux de prise en charge SSL : Les niveaux de prise en charge partiel et complet de SSL s'excluent mutuellement. En d'autres termes, ils ne peuvent pas être configurés simultanément et ne peuvent pas être activés en même temps. Si vous activez la prise en charge SSL totale pour un réseau Tivoli Workload Scheduler, toute tentative de connexion par des agents qui ne sont pas configurés pour le SSL complet sera rejetée par les agents sur lesquels la prise en charge SSL complète est activée. Vice versa, les agents configurés pour la prise en charge SSL complète ne peuvent pas communiquer avec le reste d'un réseau configuré pour une prise en charge SSL partielle.

Configuration de la sécurité SSL complète :
Pour définir une sécurité SSL complète pour votre réseau, vous devez, *en plus de toutes les étapes décrites ci-dessus dans Chapitre 7, «Définition de la sécurité des connexions», à la page 261)*, configurer les options suivantes :

enSSLFullConnection (ou sf)
Utilisez `optman` sur le gestionnaire de domaine maître pour définir cette option globale sur `Yes` afin d'activer la prise en charge SSL complète pour le réseau.

nm SSL full port
Modifiez le fichier `localopts` sur chaque agent du réseau (y compris le gestionnaire de domaine maître) pour définir cette option locale sur le numéro de port utilisé pour écouter les connexions SSL entrantes. Notez les éléments suivants :

- Ce numéro de port doit être défini également pour le paramètre `SECUREADDR` dans la définition du poste de travail de l'agent.
- Dans une configuration de sécurité SSL complète, l'option locale `nm SSL port` doit être définie avec la valeur zéro.
- Vous devez arrêter **netman** (**conman shut;wait**), puis le redémarrer (**StartUp**) après avoir apporté les modifications dans `localopts`.

- Vérifiez que la valeur *enabled* a été attribuée au paramètre `securitylevel` de la définition de chaque poste de travail utilisant SSL.

Outre la valeur modifiée de `secureaddr`, aucun autre changement n'est requis dans les définitions de poste de travail pour configurer cette fonction.

Migration d'un réseau vers la sécurité complète des connexions SSL :
Exécutez les opérations suivantes pour faire migrer votre environnement de production Tivoli Workload Scheduler version 8.3 vers la prise en charge de la sécurité complète des connexions SSL. Le scénario suppose que le réseau fonctionne déjà en SSL partiel ; en d'autres termes, que le maître et tous les agents comprennent :

- L'attribut `securitylevel` défini sur `enabled`, `on` ou `force` dans leur définition de poste de travail. Sur le maître, cette option est définie sur `enabled`.
- L'option locale `nm port` ou `nm SSL port` configurée et le numéro de port défini comme la valeur de l'attribut `secureaddr` dans leur définition de poste de travail.
- Des clés privées et certificats de groupe ou individuels.

Procédez comme suit :

1. Mettre à niveau tous les agents. Il s'agit de mettre à niveau localement chaque agent du réseau (y compris le gestionnaire de domaine maître). Vous pouvez effectuer cette opération sur plusieurs jours. Sur le maître et sur chaque agent :
   a. Installez le correctif contenant la fonction de prise en charge SSL complète.
   b. Ajoutez l'option locale `nm SSL full port` et définissez-la sur un numéro de port.

   A ce stade, le réseau fonctionne toujours sur la sécurité partielle des connexions SSL.
2. Activez la prise en charge SSL complète dans le réseau. Effectuez cette opération en une seule fois. Pour ce faire :
   a. Vérifiez qu'aucun pare-feu ne bloque la connexion entre les agents et leur gestionnaire de domaine (et éventuellement, le gestionnaire de domaine maître).
   b. Dans la définition de poste de travail du maître et de chaque agent, définissez la valeur de l'attribut `secureaddr` sur le numéro de port que vous avez configuré pour l'option locale `nm SSL full port`.
   c. Utilisez Optman pour définir l'option globale `enSSLFullConnection` sur `yes` dans la base de données.
   d. Exécutez la commande `JnextPlan -for 0000` pour rendre ces paramètres opérationnels.

   A ce stade, le réseau fonctionne sur une sécurité complète des connexions SSL. Tout agent restant sur la sécurité SSL ne peut plus communiquer avec le reste du réseau à sécurité SSL complète.

   Les postes de travail mis à niveau ont toujours les anciens ports SSL et TCP ouverts en mode écoute. L'objectif de l'étape finale est de les fermer.
3. Désactivez les anciens ports SSL et TCP sur le maître et sur chaque agent. Vous pouvez effectuer cette opération sur plusieurs jours. Pour ce faire, modifiez le fichier d'options locales de chaque poste de travail comme suit :
   - Sur les postes de travail dont l'attribut `securitylevel` est défini sur `enabled` ou sur `on`, définissez l'option locale `nm SSL port` sur 0.
   - Sur les postes de travail dont l'attribut `securitylevel` est défini sur `force`, définissez les options locales `nm port` et `nm SSL port` sur 0.

A ce stade, tous les agents fonctionnent avec les nouvelles connexions SSL et tous les agents définis sur `securitylevel=force` écoutent seulement sur le nouveau port SSL complet. Désormais :

- Aucun octet n'est envoyé en clair.
- Aucun service actif n'est laissé en clair.
- Aucun port TCP ne reste en mode d'écoute sur les agents avec `securitylevel=force`.

Configuration de la prise en charge SSL pour les dépendances inter-réseau : L'agent réseau qui résout les dépendances inter-réseau requiert une configuration particulière pour la prise en charge SSL complète.

Pour activer un agent réseau pour la prise en charge SSL complète :

1. Configurez l'agent hôte et l'agent tolérant aux pannes distant pour la prise en charge SSL complète.
2. Sur l'agent hôte tolérant aux pannes, copiez ou déplacez le fichier `netmth.opts` du répertoire *TWA_home*`/TWS/config` vers le répertoire *TWA_home*`/TWS/methods` et ajoutez (et configurez) les options suivantes :

   SSL remote CPU
   Nom du poste de travail de l'agent tolérant aux pannes ou du maître distant.

   SSL remote full port
   Numéro de port défini pour la prise en charge SSL complète sur l'agent tolérant aux pannes ou le maître distant.

   Options locales qui spécifient la clé privée et le certificat sur l'agent hôte tolérant aux pannes
   Elles sont documentées dans la rubrique «Définition des options locales», à la page 30).

   Notez que si l'agent hôte tolérant aux pannes héberge plusieurs agents de réseau, le répertoire *TWA_home*`/TWS/methods` contient un fichier `netmth.opts` pour chaque agent de réseau défini. Dans ce cas, le nom complet de chaque fichier `netmth.opts` doit devenir :

   network-agent-name`_netmth.opts`

   Si le répertoire *TWA_home*`/TWS/methods` contient à la fois les fichiers *nom-agent-réseau*`_netmth.opts` et `netmth.opts`, seul *nom-agent-réseau*`_netmth.opts` est utilisé. Si plusieurs agents sont définis et que le répertoire contient seulement `netmth.opts`, ce fichier est utilisé pour tous les agents réseau.

L'exemple suivant ajoute la prise en charge complète par SSL à l'exemple décrit dans *Exemple de définition d'agent de réseau* dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence* :

- Il s'agit de la définition de poste de travail pour l'agent réseau `NETAGT` :

```
CPUNAME NETAGT
 DESCRIPTION "AGENT RESEAU"
 OS OTHER
 NODE MASTERA.ROME.TIVOLI.COM
 TCPADDR 31117
 FOR maestro
  HOST MASTERB
  ACCESS NETMTH
END
```

- Ce sont les options de sécurité SSL complète dans le fichier `netmeth.opts` de `NETAGT` :

```
################################################
# Remote cpu parameters
################################################

SSL remote full port = 31119
SSL remote CPU = MASTERA

################################################
# Configuration Certificate
################################################

SSL key                 ="C:\TWS\installations\SSL\XA.key"
SSL certificate         ="C:\TWS\installations\SSL\XA.crt"
SSL CA certificate      ="C:\TWS\installations\SSL\VeriSte.crt"
SSL key pwd             ="C:\TWS\installations\SSL\XA.sth"
SSL certificate chain   ="C:\TWS\installations\SSL\TWSCertificateChain.crt"
SSL random seed         ="C:\TWS\installations\SSL\random_file.rnd"
SSL auth mode           =cpu
SSL auth string = tws
```

  Remarque : Les options du certificat de configuration SSL doivent faire référence à la clé privée et au certificat définis sur l'agent hôte tolérant aux pannes.
- Il s'agit de la définition de poste de travail pour `MASTERA` (le poste de travail distant) :

```
CPUNAME MASTERA
  OS WNT
  NODE 9.168.68.55 TCPADDR 31117
  SECUREADDR 31119
  DOMAIN NTWKA
  FOR MAESTRO
    TYPE MANAGER
    AUTOLINK ON
    BEHINDFIREWALL OFF
    SECURITYLEVEL enabled
    FULLSTATUS ON
    SERVER H
END
```

Scénario : HTTPS pour les clients de ligne de commande

Vous pouvez être confronté à l'un des scénarios suivants :

- Connexion SSL entre les utilitaires de ligne de commande (**composer** et **conman**) sur le gestionnaire de domaine maître et le connecteur installé dans le gestionnaire de domaine maître.
- Connexion SSL entre le client de ligne de commande distant installé sur un poste de travail et le poste de travail gestionnaire de domaine maître distant.

Connexion SSL à l'aide des certificats par défaut

Pour plus d'informations sur la connexion par défaut SSL, voir «Configuration de SSL à l'aide du certificat prédéfini», à la page 297.

Connexion SSL à l'aide de vos certificats

Pour plus d'informations sur comment créer et activer vos certificats SSL, voir «Utilisation d'un certificat personnalisé», à la page 298.

Personnalisation de la connexion SSL pour un client de ligne de commande

Les clients de ligne de commande de Tivoli Workload Scheduler entrent en contact avec le connecteur via le protocole HTTP ou HTTPS. Le type de connexion par défaut est HTTPS. Si l'utilitaire de ligne de commande établit une connexion via un serveur proxy, utilisez le protocole HTTP car HTTPS n'est pas pris en charge dans ce type de configuration.

Vous configurez le protocole de connexion comme indiqué dans la section «Configuration de l'authentification pour l'accès au client de ligne de commande», à la page 91. Si vous n'avez pas déjà utilisé SSL pour l'accès au client de ligne de commande, il vous faudra au moins modifier les paramètres suivants :

proxy Indiquez l'adresse IP ou le nom du serveur proxy.

proxy port
Indiquez le port d'écoute du serveur proxy.

protocole
Indiquez le type de protocole : HTTP ou HTTPS.

port Indiquez le port requis par le protocole que vous avez défini dans l'option **protocol**. La valeur par défaut est 31115 pour HTTP et 31116 pour HTTPS.

Le protocole de connexion HTTPS offre en outre les fonctions de sécurité suivantes :

- Chiffrement des données entre l'utilitaire de ligne de commande et le connecteur
- Authentification serveur facultative par validation des certificats du serveur

Vous pouvez activer l'authentification serveur facultative en effectuant l'une des opérations suivantes :

- «Configuration de SSL à l'aide du certificat prédéfini»
- «Configuration de plusieurs instances de communication SSL», à la page 298
- «Utilisation d'un certificat personnalisé», à la page 298

Configuration de SSL à l'aide du certificat prédéfini

Pour personnaliser la connexion SSL du client de ligne de commande en utilisant le certificat prédéfini, procédez comme suit :

1. Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver**.
2. Extrayez le certificat du magasin de clés `TWA_home/WAS/TWSprofile/etc/TWSServerKeyFile.jks` :

```
TWA_home/WAS//java/jre/bin/keytool -export
         -alias server
         -rfc
        -file server.crt
         -keystore TWA_home/WAS/TWSprofile/etc/TWSServerKeyFile.jks
        -storepass default
```

3. Copiez le certificat .crt (`server.crt` dans l'exemple précédent) sur chaque poste de travail où est installé un client de ligne de commande, en copiant le certificat dans le chemin défini à l'option de client de ligne de commande **cli ssl server certificate** (voir étape suivante).
4. Définissez les options du client de ligne de commande **cli ssl server auth** et **cli ssl server certificate** dans le fichier `localopts`. Pour plus de détails sur la définition de ces options, voir «Définition des options locales», à la page 30.

5. Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver**.

Configuration de plusieurs instances de communication SSL

Pour personnaliser la connexion SSL du client de ligne de commande afin d'établir plusieurs connexions avec WebSphere Application Server, procédez comme suit :

1. Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver**.
2. Extrayez un certificat du fichier de clés `TWSServerKeyFile.jks`.

   ```
   keytool -export
           -alias tws
           -rfc
           -file server.crt
           -keystore ServerKeyFile.jks
           -storepass default


   .
   ```

3. Extrayez le numéro de hachage de chaque certificat exporté :

   ```
   openssl x509
           -hash
           -noout
           -in keyname
   ```

4. Renommez chaque fichier de certificat avec la clé exportée.
5. Copiez les certificats renommés sur chaque poste de travail où un client de ligne de commande est installé.
6. Définissez les options du client de ligne de commande **`cli ssl server auth`** et **`cli ssl trusted dir`** dans le fichier `localopts`. Voir «Définition des options locales», à la page 30.
7. Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver**.

Utilisation d'un certificat personnalisé

Pour personnaliser le certificat SSL et le magasin de clés, procédez comme suit :

1. Créez une clé de zone RSA et extrayez la clé du magasin de clés serveur TWSServerKeyFile.jks.
2. Importer le certificat au format PEM :

   ```
   keytool -import
           -alias tws
           -file server.crt
           -trustcacerts
           -noprompt
           -keystore TWSClientTrustFile.jks
           -storepass default
   ```

3. Effectuez les étapes décrites dans «Configuration de SSL à l'aide du certificat prédéfini», à la page 297.

Remarque : Lorsque vous personnalisez les certificats de plusieurs instances, effectuez ces opérations pour chaque instance.

Utilisation du protocole SSL pour l'automatisation de charge de travail gérée par événement (EDWA) derrière des pare-feu

Cette fonction permet d'exécuter le gestionnaire de domaine comme proxy inverse pour les protocoles HTTP (HyperText Transfer Protocol) et EIP (Event Integration Facility), afin d'envoyer le trafic vers le processeur d'événements. Une option, activée à l'aide du programme de ligne de commande **optman**, permet de définir si les postes de travail qui se trouvent derrière un pare-feu doivent se connecter au gestionnaire de domaine au lieu du processeur d'événements, de sorte que le nouveau proxy sur le gestionnaire de domaine redirige son trafic vers le processeur d'événements.

Restriction : Cette configuration n'est pas prise en charge si le poste de travail d'agent et un agent dynamique.

Le trafic entrant est réacheminé comme suit :

- Si un agent se trouve derrière un pare-feu, le trafic est envoyé au gestionnaire de domaine sur l'agent. Si un agent ne se trouve pas derrière un pare-feu, le trafic est envoyé directement au processeur d'événements.
- Si des gestionnaires de domaine ont des noeuds enfants situés derrière un pare-feu, le trafic est réacheminé vers le processeur d'événements.
- Les gestionnaires de domaine principal envoient toujours le trafic vers le processeur d'événements en cours.
- Les gestionnaires de domaine de niveau inférieur réacheminent le trafic vers les gestionnaires de domaine de niveau supérieur s'ils se trouvent derrière un pare-feu, ou vers le processeur d'événements s'ils ne se trouvent pas derrière un pare-feu.

Pour utiliser cette fonction, procédez comme suit :

1. Activez la fonction en affectant à l'option **optman** la valeur `yes`. La valeur par défaut est `no` :

   ```
   enEventDrivenWorkloadAutomationProxy | pr = {yes|no}
   ```

2. Dans la définition de poste de travail dans la base de données de l'agent définissez l'attribut **behindfirewall** sur ON.
3. Configurez OpenSSL ou GSKit sur le gestionnaire de domaine.

Pour plus d'informations sur la définition de l'attribut **behindfirewall**, voir le manuel *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Configuration de Tivoli Workload Scheduler pour l'utilisation de LDAP

Pour utiliser LDAP configuré dans SSL avec Tivoli Workload Scheduler, procédez comme suit :

1. Importez la clé publique LDAP dans le magasin de clés certifiées du serveur Tivoli Workload Scheduler, en la stockant dans le fichier `TWSServerTrustFile.jks` situé dans `<chemin_profil_WAS>/etc`, où la valeur par défaut du chemin `<chemin_profil_WAS>` est `<TWA_home/WAS/TWSprofile`.
2. Importez la clé publique LDAP dans le magasin de clés certifiées du client Tivoli Workload Scheduler, en la stockant dans le fichier `TWSClientTrustFile.jks` situé dans `<chemin_profil_WAS>/etc`.

Conformité aux normes FIPS

Cette section présente la conformité aux normes FIPS. Elle comprend les sous-sections suivantes :

Présentation générale des normes FIPS

Les normes FIPS sont des normes et instructions émises par le National Institute of Standards and Technology pour les systèmes informatique du gouvernement fédéral. Les normes FIPS sont développées pour répondre aux besoins en normes du gouvernement fédéral, comme par exemple en matière de sécurité ou d'interopérabilité, et que des normes ou solutions acceptables n'existent pas déjà dans l'Industrie. Les agences gouvernementales et les institutions financières utilisent ces normes pour garantir la conformité des produits aux exigences de sécurité spécifiées.

Tivoli Workload Automation utilise des modules cryptographiques conformes aux normes FIPS-140-2. Les certificats utilisés en interne sont chiffrés à l'aide des algorithmes de cryptographie agréés par les normes FIPS. Les modules agréés par les normes FIPS peuvent, en option, être utilisés pour la transmission de données.

Pour satisfaire aux exigences de la norme FIPS 140-2, vous devez utiliser les bibliothèques dynamiques d'exécution IBM Global Security Kit (GSKit) version 7d au lieu de OpenSSL. GSKit utilise IBM Crypto for C version 1.4.5 qui est la norme FIPS 140-2 de niveau 1 certifiée par le numéro de certificat 755. Voir http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401val2007.htm. IBM Java JSSE FIPS 140-2 Cryptographic est un autre module utilisé par Tivoli Workload Automation. Il possède le numéro de certificat 409.

Si vous utilisez actuellement le protocole SSL pour les connexions sécurisées sur le réseau, vous devez utiliser GSKit pour les connexions sécurisées au lieu de OpenSSL Toolkit afin de vous conformer aux normes FIPS. GSKit est automatiquement installé avec Tivoli Workload Scheduler. Il se base sur les bibliothèques dynamiques et propose plusieurs utilitaires de gestion des certificats.

Pour respecter les normes FIPS, tous les composants de Tivoli Workload Automation doivent être conformes aux normes FIPS. Vous devez utiliser Dynamic Workload Console ou la ligne de commande Tivoli Workload Scheduler comme interface avec Tivoli Workload Scheduler. Vous devez également utiliser DB2 comme base de données Tivoli Workload Scheduler.

Si la conformité aux normes FIPS ne concerne pas votre organisation, vous pouvez continuer à utiliser le protocole SSL pour les connexions sécurisées sur votre réseau.

Des composants Tivoli Workload Automation non conformes aux normes FIPS ne peuvent pas communiquer avec des composants Tivoli Workload Automation conformes à ces normes.

Pour configurer votre réseau conformément aux normes FIPS, suivez les procédures décrites dans les sections ci-après :

- Pour créer des certificats FIPS, voir «Utilisation des certificats FIPS».
- Pour configurer le protocole SSL conformément aux normes FIPS, voir «Configuration du protocole SSL conformément aux normes FIPS», à la page 305.
- Pour configurer votre base de données DB2 conformément aux normes FIPS, voir «Configuration de DB2 pour FIPS», à la page 308.

Utilisation des certificats FIPS

Pour vérifier que votre réseau est conforme aux normes FIPS, procédez comme suit pour créer des certificats FIPS :

- Si vous ne possédez pas de certificats SSL, voir «Utilisation de nouveaux certificats FIPS».
- Si vous possédez des certificats SSL mais que vous basculez sur GSKit, voir «Basculement de OpenSSL à GSKit», à la page 302.

Si vous utilisez les certificats FIPS, vous devez utiliser les paramètres SSL pour communiquer sur le réseau. Lors de l'installation ou la mise à niveau vers Tivoli Workload Scheduler version 8.5.1, notez que les certificats SSL par défaut se trouvent dans les répertoires suivants :

install_dir_TWS\TWS\ssl\GSKit
install_dir_TWS\TWS\ssl\OpenSSL

Utilisation de nouveaux certificats FIPS

Créez des certificats FIPS pour les communications entre les postes de travail à l'aide de l'option –fips dans l'utilitaire de la ligne de commande GSKit. Vous pouvez créer des certificats FIPS des deux manières suivantes :

- En utilisant les certificats FIPS par défaut présents sur chaque agent Tivoli Workload Scheduler du réseau. Remarque : les certificats FIPS par défaut ne sont pas sécurisés.
- En créant vos propres certificats FIPS sécurisés. Voir «Création de vos propres certificats FIPS».

Création de vos propres certificats FIPS : Utilisez l'utilitaire de ligne de commande **gsk7capicmd** pour :

- Créer votre propre autorité de certification.
- Créer un certificat d'autorité de certification autosigné (x.509 structure) pour votre autorité de certification.
- Exporter le certificat d'autorité de certification au format PEM.

Création de votre propre autorité de certification : Créez l'autorité de certification sur n'importe quel poste de travail de votre réseau. Effectuez une seule fois les étapes ci-après pour créer une autorité de certification qui sera utilisée chaque fois qu'un nouveau certificat doit être créé et signé.

1. Entrez la commande ci-après pour créer la base de données de clés CMS “ca.kdb” avec le mot de passe “password00” qui expire au bout de 1000 jours.

   ```
   gsk7capicmd -keydb -create -db ca.kdb -pw password00 -stash -expire 1000 -fips
   ```

2. Entrez la commande ci-après pour créer le certificat autosigné avec l'étiquette “CA certificate” à l'aide du nom distinctif “CN=CA certificate,O=IBM,OU=TWS,C=IT”. Le certificat expire au bout de 1000 jours.

   ```
   gsk7capicmd -cert -create  -db ca.kdb -pw password00 -label "CA certificate"
        -size 2048 -expire 1000 -dn "CN=CA certificate,O=IBM,OU=TWS,C=IT"
   ```

3. Entrez la commande ci-après pour extraire le certificat d'autorité de certification dans un fichier externe “ca.crt”. L'étiquette se rapporte au certificat correspondant.

   ```
   gsk7capicmd -cert -extract -db ca.kdb -pw password00 -label "CA certificate"
         -target CA.crt
   ```

Le fichier contiendra le certificat public de l'autorité de certification.

Création d'un certificat pour l'agent Tivoli Workload Scheduler : Procédez comme suit pour créer des certificats signés par une autorité de certification sécurisée, commune et locale sur chaque agent Tivoli Workload Scheduler de votre réseau.

1. Entrez la commande ci-après pour créer une base de données de clés CMS par défaut client.kdb” avec le mot de passe “password02” qui expire au bout de 1000 jours. Le mot de passe est également stocké dans le fichier stash“client.sth”.

   ```
   gsk7capicmd -keydb -create -db client.kdb -pw password02
       -stash -expire 1000 -fips
   ```

2. Entrez la commande suivante pour ajouter le certificat d'autorité de certification en tant que certificat sécurisé dans la base de données clé CMS. L'étiquette “CA certificate client” est utilisée comme référence à ce certificat.

   ```
   gsk7capicmd -cert -add -db client.kdb -pw password02
       -label "CA certificate client" -trust enable -file CA.crt
       -format ascii -fips
   ```

3. Entrez la commande suivante pour créer la demande de certificat client basé sur une clé 2048 bits, avec le libellé **"Client TWS85 Certificate"** et le nom distinctif **"CN=Client TWS85,O=IBM,OU=TWS,C=IT"**. La demande de certificat “client.csr” est créée et la clé privée est créée dans la base de données client.kdb.

   ```
   gsk7capicmd -certreq -create -db client.kdb -pw password02
       -label "Client TWS85 Certificate" -size 2048 -file client.csr
       –dn "CN=Client  TWS85,O=IBM,OU=TWS,C=IT" -fips
   ```

4. Entrez la commande suivante pour que le CA signe la demande de certificat du client et génère un nouveau fichier signé “client.crt”.

   ```
   gsk7capicmd -cert -sign -db ca.kdb -pw password00 -label "CA certificate"
        -target client.crt -expire 365 -file client.csr -fips
   ```

5. Entrez la commande suivante pour importer le certificat signé “client.crt” dans la base de données de clés CMS “client.kdb”.

   ```
   gsk7capicmd -cert -receive -db client.kdb -pw password02 -file client.crt -fips
   ```

Vous pouvez répéter les étapes ci-dessus pour tous les agents ou utiliser le même certificat pour tous les agents, selon vos stratégies de sécurité et les configurations localopts de Tivoli Workload Automation.

Basculement de OpenSSL à GSKit

Cette section explique comment faire migrer vos certificats OpenSSL vers des certificats GSKit.

Les formats de certificat pouvant être migrés vers le format GSKit (**KDB**) sont les suivants :

- **PEM** : Utilisé par OpenSSL

- **JKS** : Utilisé par Java et WebSphere Application Server
- **PKCS12** : Utilisé par les applications Microsoft et Internet Explorer

Pour faire migrer les certificats, vous pouvez utiliser un ou plusieurs des outils listés ci-après :

- **gsk8capicmd** : Ligne de commande native fournie par GSKit
- **openssl** : Ligne de commande native fournie par OpenSSL
- **ikeyman** : Interface graphique facultative fournie par GSKit
- **keytool** : Interface graphique facultative fournie par Java Virtual Machine (JVM)

Remarque : Sauvegardez vos certificats d'origine avant de les faire migrer au format GSKit.

Pour faire migrer vos certificats, procédez comme suit :

1. «Configuration de l'environnement de l'outil»
2. «Migration des certificats»

Configuration de l'environnement de l'outil : Cette section décrit les commandes que vous devez exécuter pour configurer gsk8capicmd et openssl.

Configuration de gsk8capicmd :

gsk8capicmd sur 32 bits

```
set PATH=C:\Program Files\IBM\TWA\TWS\GSkit32\8\lib; C:\Program
Files\IBM\TWA\TWS\GSkit32\8\bin;%PATH%
```

gsk8capicmd_64 sur 64 bits

```
set PATH=C:\Program Files\IBM\TWA\TWS\GSkit64\8\lib64; C:\Program
Files\IBM\TWA\TWS\GSkit64\8\bin;%PATH%
```

Configuration de openssl :

UNIX `tws_env.sh`

Windows

```
tws_env.cmd
```

Migration des certificats : Cette section présente les commandes que vous devez exécuter pour faire migrer les certificats au format KDB, conforme aux normes FIPS.

Remarque : le format PEM ne peut pas être directement converti au format KDB ; Vous devez d'abord le convertir au format PKCS12 puis au format KDB.

La liste ci-après répertorie les commandes que vous devez exécuter pour convertir d'un format à un autre :

Format JKS vers le format KDB

gsk7cmd -keydb -convert -db TWSClientKeyFile.jks -pw default -old_format jks -new_format cms

gsk7cmd -keydb -convert -db TWSClientTrustFile.kdb -pw default -old_format cms -new_format jks

Format PKCS12 vers le format KDB

gsk7capicmd -cert -export -target TWSClientKeyFile_new.kdb -db TWSClientKeyFileP12.P12 -fips -target_type cms -type pkcs12

Format PKCS12 vers le format PEM
openssl pkcs12 -in TWSClientKeyFileP12.P12 -out TWSClientKeyFile.pem

Format PEM vers le format PKCS12
openssl pkcs12 -export -in TWSClientKeyFile.pem -out cred.p12

Format KDB vers le format PKCS12
gsk7capicmd -cert -export -db TWSClientKeyFile.kdb -target TWSClientKeyFileP12.P12 -fips -target_type pkcs12 -type cms

Conversion des certificats PEM en certificats CMS : Cette section décrit la procédure de conversion des certificats PEM (OpenSSL) en certificats CMS (GSKit). Les exemples de cette section utilisent les fichiers d'entrée et de sortie ci-après.

Fichiers en entrée

Fichier de certificat personnel : *CPU1.crt*
Clé personnelle du fichier de certificat : *CPU1.key*
Certificat du fichier d'autorité de certification : *TWSca.crt*
Fichier stash : *CPU1.sth*

Fichiers de sortie

Fichier de base de données de clés : *TWS.kdb*
Fichier stash : *TWS.sth*
Etiquette de votre certificat : *CPU1*

Pour faire migrer des certificats OpenSSL vers des certificats GSKit, procédez comme suit :

1. Fusionnez les clés privées et publiques dans un nouveau fichier temporaire nommé **all.pem** en exécutant les commandes suivantes :

   UNIX cat CPU2.crt CPU2.key > all.pem

   Windows
   tapez CPU1.crt CPU1.key > all.pem
2. Si vous ne connaissez pas le mot de passe, extrayez-le du fichier stash en exécutant `openssl base64 -d -in CPU1.sth`.
3. Sélectionnez un mot de passe pour la nouvelle base de données de clés. Vous pouvez réutiliser l'ancien mot de passe.
4. Sélectionnez une étiquette pour votre certificat personnel et votre clé personnelle (dans cet exemple, CPU1) et créez la base de données PKCS12 contenant les étiquettes. Utilisez le nom, CPU1, comme étiquette de la nouvelle base de données de clés. Pour créer la base de données PKCS12, exécutez la commande suivante :

   ```
   openssl pkcs12 -export -in all.pem -out TWS.p12 -name CPU1  -passin pass:
           password1 -passout pass:password2
   ```

   où *password_1* est le mot de passe extrait du fichier stash et *password_2* est le nouveau mot de passe permettant de gérer la nouvelle base de données de clés.
5. Convertissez la base de données PKCS12 de TWS.p12 vers la base de données CMS, TWS.kdb en exécutant la commande suivante :

   ```
   gsk7capicmd -cert -import -target TWS.kdb -db TWS.p12 -target_type cms
            -type pkcs12 -label CPU1 -target_pw "password_2" -pw "password_3"
   ```

   où *password2* est l'ancien mot de passe que vous avez extrait du fichier stash, CPU1.sth et *password3* est le nouveau mot de passe.

6. Choisissez une étiquette pour votre autorité de certification contenue dans TWSca.crt. Dans cet exemple, il s'agit de *TWSca*.
7. Ajoutez le certificat de l'autorité de certification dans votre fichier TWS.kdb en exécutant la commande :

   ```
   gsk7capicmd -cert -add -db TWS.kdb -label TWSca -trust -file TWSca.crt
           -format ascii -pw "password"
   ```

8. Supprimez tous les fichiers .pem.

Configuration du protocole SSL conformément aux normes FIPS

Pour configurer le protocole SSL conformément aux normes FIPS, procédez comme suit :

- Définissez les paramètres localopts. Voir «Définition des paramètres localopts pour les normes FIPS».
- Configurez WebSphere Application Server. Voir «Configuration de WebSphere Application Server pour les normes FIPS», à la page 306.
- Configurez le port de la fonction d'intégration d'événements Tivoli. Voir «Configuration du port de la fonction d'intégration d'événements Tivoli», à la page 307.

Remarque :

Si vous utilisez Dynamic Workload Broker pour la planification dynamique sur votre réseau, notez que le poste de travail de type **BROKER** ne prend pas en charge le protocole SSL. Tous les postes de travail Tivoli Workload Scheduler doivent communiquer avec le poste de travail de type **BROKER** à l'aide du protocole TCP/IP.

Définition des paramètres localopts pour les normes FIPS

Pour configurer votre environnement conformément aux normes FIPS, définissez l'option locale suivante sur chaque agent Tivoli Workload Scheduler du réseau.

SSL Fips enabled (FIPS SSL activées) = *oui*

L'exemple suivant s'applique à un agent Windows. Définissez les options locales suivantes pour le moteur :

fichier de clés de sécurité SSL = *"<TWA_home>\TWS\ssl\GSKit\TWSClientKeyStore.kdb"*
SSL certificate keystore label = *"client"*
SSL keystore pwd = *"<TWA_home>\TWS\ssl\GSKit\TWSClientKeyStore.sth"*

où *<rép_princ_TWA>* est le répertoire d'installation de l'instance de Tivoli Workload Automation sur lequel l'agent est installé.

Définissez les options locales suivantes pour l'interface CLI :

CLI SSL keystore file (Fichier de clés SSL d'interface CLI) =
"<TWA_home>\TWS\ssl\GSKit\TWSClientKeyStore.kdb"
CLI SSL certificate keystore label = *"client"*
CLI SSL keystore pwd =
"<TWA_home>\TWS\ssl\GSKit\TWSClientKeyStore.sth"

où *<rép_princ_TWA>* est le répertoire d'installation de l'instance de Tivoli Workload Automation sur lequel l'agent est installé.

Remarque : Sur les postes de travail Windows, l'utilisateur **SYSTEM**, doit posséder des droits en lecture pour lire les certificats FIPS GSKi.

Configuration de WebSphere Application Server pour les normes FIPS

Pour vous conformer aux normes FIPS, vous devez configurer WebSphere Application Server pour Tivoli Workload Scheduler.

Cette section explique comment :

- Configurer WebSphere Application Server pour Tivoli Workload Scheduler. Voir «Configuration de WebSphere Application Server pour Tivoli Workload Scheduler».
- Configurer le port de la fonction d'intégration d'événements Tivoli. Voir «Configuration du port de la fonction d'intégration d'événements Tivoli», à la page 307.

Configuration de WebSphere Application Server pour Tivoli Workload Scheduler : Pour configurer WebSphere Application Server pour la conformité aux normes FIPS, procédez comme suit :

1. Dans l'interface d'administration de WebSphere Application Server, cliquez sur **Sécurité > Gestion des clés et des certificats SSL**. Sélectionnez **Use the United States Federal Information Processing Standard (FIPS) algorithms (Utiliser les algorithmes FIPS des Etats-Unis** et cliquez sur **Appliquer**. Vous pouvez également utiliser des outils WAS et exécuter **changeSecurityProperties** pour modifier le paramètre suivant :

   ```
   useFIPS=true
   ```

2. Dans le fichier **profile_root/properties/ssl.client.props**, Définissez les paramètres suivants :
   - **com.ibm.security.useFIPS**=*true*
   - **com.ibm.ssl.protocol**=*SSL_TLS*
3. Si un de vos clients d'administration utilise un connecteur SOAP, ajoutez la ligne suivante au fichier **profile_root/properties/soap.client.props** :

   ```
   com.ibm.ssl.contextProvider=IBMJSSEFIPS
   ```

4. Modifiez le fichier java.security SDK situé dans le répertoire `WASHOME/java/jre/lib/security` pour insérer le fournisseur **IBMJCEFIPS**(**com.ibm.crypto.fips.provider.IBMJCEFIPS**). **IBMJCEFIPS** doit précéder le fournisseur **IBMJCE** dans la liste des fournisseurs.

   Voici un exemple de fichier java.security SDK modifié :

   ```
   security.provider.1=com.ibm.crypto.fips.provider.IBMJCEFIPS
   security.provider.2=com.ibm.crypto.provider.IBMJCE
   security.provider.3=com.ibm.jsse.IBMJSSEProvider
   security.provider.4=com.ibm.jsse2.IBMJSSEProvider2
   security.provider.5=com.ibm.security.jgss.IBMJGSSProvider
   security.provider.6=com.ibm.security.cert.IBMCertPath
   security.provider.7=com.ibm.crypto.pkcs11.provider.IBMPKCS11
   security.provider.8=com.ibm.security.cmskeystore.CMSProvider
   security.provider.9=com.ibm.security.jgss.mech.spnego.IBMSPNEGO
   ```

   Voici un exemple de fichier java.security modifié si vous utilisez le kit de développement Oracle Java SE :

   ```
   security.provider.1=sun.security.provider.Sun
   security.provider.2=com.ibm.crypto.fips.provider.IBMJCEFIPS
   security.provider.3=com.ibm.crypto.provider.IBMJCE
   security.provider.4=com.ibm.jsse.IBMJSSEProvider
   security.provider.5=com.ibm.jsse2.IBMJSSEProvider2
   security.provider.6=com.ibm.security.jgss.IBMJGSSProvider
   ```

```
security.provider.7=com.ibm.security.cert.IBMCertPath
security.provider.8=com.ibm.i5os.jsse.JSSEProvider
#security.provider.8=com.ibm.crypto.pkcs11.provider.IBMPKCS11
security.provider.9=com.ibm.security.jgss.mech.spnego.IBMSPNEGO
```

5. Redémarrez le serveur WebSphere Application Server.

Remarque : Pour plus d'informations sur WebSphere Application Server et les normes FIPS, voir la documentation WebSphere Application Server.

Suppression de la configuration du fournisseur FIPS : Pour supprimer la configuration du fournisseur FIPS, inversez les modifications que vous avez effectuées dans «Configuration de WebSphere Application Server pour les normes FIPS», à la page 306. Après avoir inversé les modifications, vérifiez que vous avez effectué les modifications suivantes dans les fichiers ssl.client.props, soap.client.props et java.security :

- Dans le fichier ssl.client.props, modifiez la valeur de com.ibm.security.useFIPS sur false.
- Dans le fichier java.security, modifiez le fournisseur FIPS pour un fournisseur non FIPS.
- Si vous utilisez le fichier java.security SDK , modifiez le premier fournisseur pour un fournisseur non FIPS comme dans l'exemple suivant :

  ```
  #security.provider.1=com.ibm.crypto.fips.provider.IBMJCEFIPS
  security.provider.1=com.ibm.crypto.provider.IBMJCE
  security.provider.2=com.ibm.jsse.IBMJSSEProvider
  security.provider.3=com.ibm.jsse2.IBMJSSEProvider2
  security.provider.4=com.ibm.security.jgss.IBMJGSSProvider
  security.provider.5=com.ibm.security.cert.IBMCertPath
  #security.provider.6=com.ibm.crypto.pkcs11.provider.IBMPKCS11
  ```

- Si vous utilisez le fichier java.security Oracle JDK, remplacez le troisième fournisseur par un fournisseur non FIPS comme dans l'exemple suivant :

  ```
  security.provider.1=sun.security.provider.Sun
  security.provider.2=com.ibm.security.jgss.IBMJGSSProvider
  #security.provider.3=com.ibm.crypto.fips.provider.IBMJCEFIPS
  security.provider.3=com.ibm.crypto.provider.IBMJCE
  security.provider.4=com.ibm.jsse.IBMJSSEProvider
  security.provider.5=com.ibm.jsse2.IBMJSSEProvider2
  security.provider.6=com.ibm.security.cert.IBMCertPath
  #security.provider.7=com.ibm.crypto.pkcs11.provider.IBMPKCS11
  ```

- Cette étape s'applique uniquement si vous avez ajouté les paramètres d'usine du socket JSSE par défaut au fichier java.security SDK comme indiqué dans la section «Configuration de DB2 pour FIPS», à la page 308. Si vous les avez ajoutés, supprimez les paramètres suivants :

  ```
  ssl.SocketFactory.provider=com.ibm.jsse2.SSLSocketFactoryImpl
  ssl.ServerSocketFactory.provider=com.ibm.jsse2.SSLServerSocketFactoryImpl
  ```

Configuration du port de la fonction d'intégration d'événements Tivoli

Le port de la fonction d'intégration d'événements Tivoli pour SSL, **eventProcessorEIFSSLPort**, est utilisé pour la gestion des événements. Pour que le port de la fonction d'intégration d'événements Tivoli communique en mode FIPS, vous devez d'abord configurer WebSphere Application Server pour les normes FIPS. Voir «Configuration de WebSphere Application Server pour Tivoli Workload Scheduler», à la page 306.

Pour configurer le port de la fonction d'intégration d'événements Tivoli, procédez comme suit :

1.

Définissez l'option globale du port à l'aide de optman. Définissez le port comme suit :

`eventProcessorEIFSSLPort / ef = portnumber`

où *portnumber* est le numéro de port de tout port disponible sur votre réseau.

2. Pour mettre à jour le fichier Symphony, exécutez **JnextPlan -for 0000**.
3. Redémarrez EventProcessor à l'aide des commandes **conman stopevtp** et **conman startevtp**.
4. Redémarrez le moteur de surveillance Tivoli Workload Scheduler avec les commandes conman, **stopmon** et **startmon**.

Configuration de DB2 pour FIPS

Pour configurer DB2 conformément aux normes FIPS, effectuez les procédures suivantes sur la version de DB2 prise en charge que vous utilisez:

- «Configuration de DB2»
- «Configuration de la connexion de DB2 à Tivoli Workload Scheduler», à la page 311

Remarque : Si vous voulez créer vos propres certificats DB2, reportez-vous à la documentation DB2.

Configuration de DB2

Pour configurer les versions prises en charge de DB2 pour la conformité FIPS, procédez comme suit :

Remarque : Pour plus d'informations sur les versions prises en charge de DB2, référez-vous au Document relatif à la configuration requise à l'adresse http://www-01.ibm.com/support/docview.wss?rs=672&uid=swg27041009.

1. Vérifiez que le chemin vers les bibliothèques GSKit est compris dans les variables d'environnement correspondantes. Les noms des variables d'environnement varient en fonction du système d'exploitation, comme suit :

   UNIX et Linux
   Variables d'environnement LIBPATH, LD_LIBRARY_PATH ou SHLIB_PATH. Indiquez ces informations dans le fichier .profile du propriétaire de l'instance DB2.

   Windows
   Variable d'environnement PATH. Par exemple, `c:\Program Files\IBM\gsk8\lib`.

   GSKit est automatiquement inclus à l'emplacement d'installation du système de base de données DB2.

   Sur les systèmes Windows 32 bits
   Les bibliothèques GSKit figurent dans C:\Program Files\IBM\GSK8\lib. Dans ce cas, la variable PATH du système doit inclure C:\Program Files\IBM\GSK8\lib.

   Sur les systèmes d'exploitation Windows 64 bits
   Les bibliothèques GSKit 64 bits figurent dans C:\Program Files\IBM\GSK8\lib64 et les bibliothèques GSKit 32 bits figurent dans C:\Program Files (x86)\IBM\GSK8\lib.

   Sur les systèmes d'exploitation UNIX et Linux,
   les bibliothèques GSKit figurent dans sqllib/lib. C'est pourquoi les variables d'environnement LIBPATH, SHLIB_PATH ou LD_LIBRARY_PATH doivent inclure sqllib/lib.

Sur les systèmes d'exploitation autres que Windows
Le gestionnaire de base de données DB2 installe GSKit en local, et pour une instance donnée, les bibliothèques GSKit doivent figurer dans sqllib/lib ou sqllib/lib64.

2. Pour configurer votre serveur DB2 pour la prise en charge SSL, connectez-vous en tant que propriétaire de l'instance DB2, puis définissez les paramètres de configuration suivants, ainsi que la variable de registre **DB2COMM**. Utilisez la commande **db2 update dbm cfg** *parameter_name* **using** *parameter_value*, où

   parameter_name
   Correspond au nom du paramètre à définir.

   parameter_value
   Correspond à la valeur du paramètre à définir.

   a. Définissez le paramètre de configuration **ssl_svr_keydb** avec le chemin complet du fichier de la base de données de clés. Par exemple :
   ```
   C:\TWS\installations\tws850cli\TWS\ssl\gskit\TWSClientKeyStore.kdb
   ```

   Où :

   TWSClientKeyStore.kdb
   Correspond au nom qualifié complet du fichier de clés qui héberge le certificat DB2 et les certificats sécurisés, par exemple les certificats du serveur WebSphere Application Server auquel se connecter. Ce fichier de clés peut être le même que celui spécifié dans les paramètres localopts. Voir «Définition des paramètres localopts pour les normes FIPS», à la page 305. Remarque : il doit être reconnu par le certificat WebSphere Application Server JKS.

   Si **ssl_svr_keydb** a la valeur null (non défini), la prise en charge SSL n'est pas activée.

   b. Définissez le paramètre de configuration **ssl_svr_stash** avec le chemin complet du fichier stash. Par exemple :
   ```
   C:\TWS\installations\tws850cli\TWS\ssl\gskit\TWSClientKeyStore.sth
   ```

   Si **ssl_svr_stash** a la valeur null (non défini), la prise en charge SSL n'est pas activée.

   c. Définissez le paramètre de configuration **ssl_svr_label** avec l'étiquette du certificat numérique du serveur. Si le paramètre **ssl_svr_label** n'est pas défini, c'est le certificat par défaut de la base de données de clés qui est utilisé. S'il n'y a pas de certificat par défaut dans cette base de données, SSL n'est pas activée. Par exemple :
   ```
   "client"
   ```

   d. Définissez le paramètre de configuration **ssl_svcename** avec la valeur du port utilisé par le système de base de données DB2 pour écouter les connexions SSL. Si les protocoles TCP/IP et SSL sont activés (variable de registre **DB2COMM** définie avec 'TCPIP, SSL'), définissez le paramètre **ssl_svcename** avec un port différent de celui sur lequel est défini **svcename**. Le paramètre de configuration **svcename** définit le port utilisé par le système de base de données DB2 pour écouter les connexions TCP/IP. Si vous définissez **ssl_svcename** avec le même port que **svcename**, TCP/IP et SSL ne sont ni l'un ni l'autre activés. Si **ssl_svcename** a la valeur null (non défini), la prise en charge SSL n'est pas activée.

   Remarque :

1) Dans les environnements de reprise à haut niveau de disponibilité après incident (HADR), ne définissez pas **hadr_local_svc** sur le système de base de données principal ou de secours avec la même valeur que vous avez indiquée pour **ssl_svcename**. Ne définissez pas non plus **hadr_local_svc** avec la même valeur que **svcename** ou **svcename** plus un.
2) Lorsque la variable de registre **DB2COMM** est définie avec 'TCPIP,SSL', si la prise en charge TCPIP n'est pas activée correctement, par exemple suite à la définition du paramètre de configuration **svcename** avec la valeur null, l'erreur SQL5043N est renvoyée et la prise en charge SSL n'est pas activée.

e. (Facultatif) Pour spécifier les algorithmes de cryptographie pouvant être utilisés par le serveur, définissez le paramètre de configuration **ssl_cipherspecs**. Si vous laissez le paramètre **ssl_cipherspecs** avec la valeur null (non défini), cela permet à GSKit de sélectionner l'algorithme de cryptographie le plus fort disponible, pris en charge à la fois sur le client et le serveur.

f. Ajoutez la valeur SSL à la variable de registre **DB2COMM**. Par exemple :

```
db2set -i db2inst1 DB2COMM=SSL
```

Le gestionnaire de base de données peut prendre en charge plusieurs protocoles à la fois. Par exemple, pour activer les protocoles de communication TCP/IP et SSL :

```
db2set -i db2inst1 DB2COMM=SSL,TCPIP
```

Où :

db2inst1
Correspond au nom de l'instance DB2.

Remarque : Pendant l'installation d'un gestionnaire de domaine maître Tivoli Workload Scheduler ou d'un gestionnaire de domaine maître de sauvegarde, vous devez activer le port TCPIP DB2. DB2 peut simultanément prendre en charge les protocoles de communication TCP/IP et SSL. L'administrateur DB2 peut définir le port TCPIP à l'aide de la commande **db2set DB2COMM=TCPIP, SSL**. Utilisez cette commande si vous installez un gestionnaire de domaine maître Tivoli Workload Scheduler ou un gestionnaire de domaine maître de sauvegarde et que vous possédez déjà une instance de DB2 conforme aux normes FIPS. Après l'installation, vous pouvez choisir de réinitialiser DB2COMM avec le protocole SSL uniquement.

g. Redémarrez l'instance DB2. Par exemple :

```
db2stop
db2starts
```

3. Insérez les paramètres usine du socket JSSE par défaut suivants dans le fichier java.security de l'instance Tivoli Workload Scheduler WebSphere Application Server :

```
ssl.SocketFactory.provider=com.ibm.jsse2.SSLSocketFactoryImpl
ssl.ServerSocketFactory.provider=com.ibm.jsse2.SSLServerSocketFactoryImpl
```

4. Redémarrez DB2.

Remarque : Il n'est pas nécessaire de mettre à jour la JVM de DB2. Vous avez déjà mis à jour la JVM de WebSphere Application Server dans la procédure décrite dans «Configuration de WebSphere Application Server pour les normes FIPS», à la page 306.

Pour plus d'informations sur la configuration de DB2 conformément aux normes FIPS, consultez la documentation DB2 qui décrit comment configurer la prise en charge SSL dans une instance DB2.

Configuration de la connexion de DB2 à Tivoli Workload Scheduler

Après avoir configuré DB2, vous devez configurer Tivoli Workload Scheduler pour communiquer avec les nouveaux paramètres de DB2. Procédez comme suit :

1. Modifiez les propriétés de la source de données DB2 dans les Wastool en exécutant **showDataSourceProperties** et **changeDataSourceProperties** pour y inclure les paramètres suivants :

   ```
   DB2Type4PortNumber=nnnnn
   DB2Type4SslConnection=true
   ```

   où *nnnnn* est le numéro de port SSL DB2.
2. Redémarrez le serveur WebSphere Application Server.

Utilisation de Dynamic Workload Console et des normes FIPS

Pour vérifier que vous vous connectez à Dynamic Workload Console en utilisant les normes FIPS, procédez comme suit :

1. Activez comme suit le protocole TLS dans votre navigateur :
   - Pour activer TLS dans Internet Explorer, ouvrez le navigateur et cliquez sur **Outils > Options Internet**. Dans l'onglet Avancé, sélectionnez **TLS 1.0**.
   - Pour activer TLS dans Mozilla Firefox, ouvrez le navigateur et cliquez sur **Outils >Options >Avancé**. Dans l'onglet Chiffrement, sélectionnez **Utiliser TLS 1.0**.
   - Pour activer TLS dans les autres navigateurs Internet, consultez la documentation produit du navigateur concerné.
2. En fonction de votre configuration, effectuez l'une des procédures suivantes :

   Dynamic Workload Console sur WebSphere Application Server :

   a. Vérifiez que WebSphere Application Server est conforme aux normes FIPS. Voir «Configuration de WebSphere Application Server pour les normes FIPS», à la page 306.

   Dynamic Workload Console avec un référentiel de paramètres DB2 :

   a. Vérifiez que DB2 est conforme aux normes FIPS. Voir «Configuration de DB2 pour FIPS», à la page 308.

   b. Pour garantir la connexion SSL requise entre DB2 et Dynamic Workload Console, procédez comme suit :

      1) Accédez au répertoire wastools de Tivoli Workload Scheduler et modifiez les propriétés TDWCDataSource de sorte à inclure les paramètres suivants :

         ```
         useSslConnection=true
         deleteAndRecreate=true
         databasePort=nnnnn
         ```

         où *nnnnn* est le numéro de port SSL DB2.

      2) Exécutez **installTDWCDataSource** en entrant les commandes suivantes :

         Systèmes d'exploitation UNIX et Linux

         InstallTDWCDataSource.sh TDWCDataSource.properties

Systèmes d'exploitation Windows
InstallTDWCDataSource.bat TDWCDataSource.properties

c. Redémarrez le serveur WebSphere Application Server.

3. Si vous utilisez Dynamic workload broker configurez une connexion sécurisée comme suit :
   a. Dans Dynamic Workload Console, allez dans Tivoli Dynamic Workload Broker et développez le menu **Configuration**.
   b. Cliquez sur **Connexions au serveur**.
   c. Dans l'écran Connexions au serveur, sélectionnez **Utiliser une connexion sécurisée**.
   d. Cliquez sur **OK**.
4. Si vous voulez configurer une connexion SSL à un moteur Tivoli Workload Scheduler for z/OS, lancez l'un des utilitaires suivants, en attribuant la valeur **true** au paramètre `useSSL` :

   createZosEngine
   Utilisez cet utilitaire si vous n'avez pas encore créé de connexion au moteur Tivoli Workload Scheduler for z/OS.
   - Sur les systèmes d'exploitation Windows, lancez `\wastools\createZosEngine.bat`
   - Sur les systèmes d'exploitation UNIX et Linux, lancez `/wastools/createZosEngine.sh`

   updateZosEngine
   Utilisez cet utilitaire si vous avez déjà créé de connexion au moteur Tivoli Workload Scheduler for z/OS.
   - Sur les systèmes d'exploitation Windows, lancez `\wastools\updateZosEngine.bat`
   - Sur les systèmes d'exploitation UNIX et Linux, lancez `/wastools/updateZosEngine.sh`

Remarque : Pour activer les communications entre Dynamic Workload Console et DB2, configurez les propriétés système Java dans Dynamic Workload Console pour utiliser le fichier de clés certifiées. Pour cela, définissez les propriétés système Java suivantes :

```
javax.net.ssl.trustStore
javax.net.ssl.trustStorePassword
```

Pour de plus amples informations, consultez la documentation DB2.

Configuration de Dynamic Workload Broker pour FIPS

Si vous utilisez le composant Dynamic Workload Broker sur votre réseau, vous devez définir les configurations suivantes :

- Configurez le fichier ita.ini de chaque agent qui communiquera avec le composant Dynamic Workload Broker. Vérifiez que le port ssl_port est défini et définissez fips_enable = 1.
- Si vous utilisez Dynamic Workload Console configurez une connexion sécurisée comme suit :
  1. Dans Dynamic Workload Console, allez dans Dynamic Workload Broker et développez le menu **Configuration**.
  2. Cliquez sur **Connexions au serveur**.

3. Dans l'écran Connexions au serveur, sélectionnez **Utiliser une connexion sécurisée**.
4. Cliquez sur **OK**.

Configuration des rapports de traitement par lots pour FIPS

Pour configurer les rapports par lots pour la conformité FIPS, procédez comme suit :

- Importez le certificat FIPS à partir du serveur de base de données dans un fichier de clés certifiées Java sur le client. Utilisez l'utilitaire keytool Java pour importer le certificat dans le fichier de clés certifiées.
- Modifiez le fichier SDK java.security situé dans le répertoire `INSTALL_DIR/java/jre/lib/security` pour insérer le fournisseur **IBMJCEFIPS** (**com.ibm.crypto.fips.provider.IBMJCEFIPS**). **IBMJCEFIPS** doit précéder le fournisseur **IBMJCE** dans la liste des fournisseurs.

  Voici un exemple de fichier java.security SDK modifié :

  ```
  security.provider.1=com.ibm.crypto.fips.provider.IBMJCEFIPS
  security.provider.2=com.ibm.crypto.provider.IBMJCE
  security.provider.3=com.ibm.jsse.IBMJSSEProvider
  security.provider.4=com.ibm.jsse2.IBMJSSEProvider2
  security.provider.5=com.ibm.security.jgss.IBMJGSSProvider
  security.provider.6=com.ibm.security.cert.IBMCertPath
  security.provider.7=com.ibm.crypto.pkcs11.provider.IBMPKCS11
  security.provider.8=com.ibm.security.cmskeystore.CMSProvider
  security.provider.9=com.ibm.security.jgss.mech.spnego.IBMSPNEGO
  ```

  Voici un exemple de fichier de sécurité java modifié si vous utilisez le Kit de développement Oracle Java SE :

  ```
  security.provider.1=sun.security.provider.Sun
  security.provider.2=com.ibm.crypto.fips.provider.IBMJCEFIPS
  security.provider.3=com.ibm.crypto.provider.IBMJCE
  security.provider.4=com.ibm.jsse.IBMJSSEProvider
  security.provider.5=com.ibm.jsse2.IBMJSSEProvider2
  security.provider.6=com.ibm.security.jgss.IBMJGSSProvider
  security.provider.7=com.ibm.security.cert.IBMCertPath
  security.provider.8=com.ibm.i5os.jsse.JSSEProvider
  #security.provider.8=com.ibm.crypto.pkcs11.provider.IBMPKCS11
  security.provider.9=com.ibm.security.jgss.mech.spnego.IBMSPNEGO
  ```

- Vérifiez que le paramètre `keystore.type` est identique à la valeur spécifiée pour le `type` de `keystore` dans le fichier `config.file`. La valeur par défaut est `JKS`.

Configuration de LDAP pour FIPS

Pour vous conformer aux normes FIPS si vous utilisez un serveur LDAP, avant de configurer LDAP, éditez le fichier security.xml. Modifiez la valeur suivante :

```
"com.ibm.ssl.contextProvider" value="IBMJSSEFIPS"
```

Recherche de la version de GSKit sur les agents s'exécutant sur les systèmes d'exploitation UNIX et Linux

Pour savoir quelle version de GSKit s'exécute sur votre agent, accédez au chemin suivant en fonction de la version de GSKit et soumettez la commande appropriée :

GSKit 32 bits

Chemin

```
/usr/Tivoli/TWS/GSKit32/8/bin
```

Commande

gsk8ver

GSKit 64 bits

Chemin
/usr/Tivoli/TWS/GSKit64/8/bin

Commande
gsk8ver_64

Sur UNIX et Linux, vous pouvez éventuellement exécuter le script ita_props.sh pour définir l'environnement sur /usr/Tivoli/TWS/GSKit32/8/bin ou /usr/Tivoli/TWS/GSKit64/8/bin, de manière à pouvoir exécuter cette commande directement sans avoir à spécifier le chemin relatif.

Chapitre 8. Maintenance des données

Le présent chapitre explique comment procéder à la maintenance de votre base de données Tivoli Workload Scheduler et des autres fichiers de données. La base de données est hébergée sur l'infrastructure DB2 ou Oracle RDBMS, selon le choix effectué à l'installation. Consultez la documentation de DB2 ou d'Oracle pour obtenir des instructions générales sur la maintenance de la base de données. Le présent chapitre décrit les activités de maintenance spécifiques à Tivoli Workload Scheduler.

Il comprend les sections suivantes :

Maintenance de la base de données

Cette section traite des sujets suivants :

- Sauvegarde et restauration de fichiers dans les bases de données Tivoli Workload Scheduler. Voir «Sauvegarde et restauration».
- Vérification de la mise à jour d'un gestionnaire de domaine maître à la dernière version disponible. Voir «Utilisation d'un gestionnaire de domaine maître de secours avec une base de données de sauvegarde», à la page 316.
- Préservation du niveau de performance des bases de données Tivoli Workload Scheduler. Voir «Réorganisation de la base de données», à la page 317.

Sauvegarde et restauration

Pour réduire les interruptions pendant une reprise après incident, sauvegardez fréquemment vos fichiers de données maîtres vers le stockage hors ligne ou vers un gestionnaire de domaine maître de secours.

Sauvegarde de la base de données sur un support de stockage autonome

Sauvegardez régulièrement la base de données sur un support de stockage autonome. Suivez les instructions figurant dans la documentation DB2 ou la documentation Oracle.

Le produit Tivoli Workload Scheduler est accompagné d'un utilitaire qui permet d'effectuer des sauvegardes : Il est appelé **tws_inst_pull_info**. Il s'agit principalement d'un outil de collecte d'informations de Tivoli Workload Scheduler, destinées au service de support logiciel IBM en cas d'incident. Cependant, il s'agit également d'un outil de sauvegarde. Il sauvegarde la base de données (DB2 uniquement), les fichiers de configuration et les fichiers journaux.

Cet outil est décrit dans *Tivoli Workload Scheduler - Guide d'identification des problèmes* et fournit des détails complets sur les fichiers qui sont sauvegardés, la manière d'exécuter une sauvegarde et de procéder à une restauration à partir d'une sauvegarde.

Utilisation d'un gestionnaire de domaine maître de secours avec une base de données de sauvegarde

Configurez un gestionnaire de domaine maître de secours qui accède à une base de données différente de celle du gestionnaire de domaine maître, puis demandez à votre administrateur de base données de configurer une version miroir de la base de données du gestionnaire de domaine maître dans la base de données du gestionnaire de domaine maître de secours. De cette façon, votre gestionnaire de domaine maître de secours reçoit une copie de chaque message relatif au traitement (ce qui est défini par l'attribut *FullStatus* sur le gestionnaire de domaine maître de secours) et il a accès à la base de données en miroir. La fréquence de la fonction miroir doit être suffisamment élevée pour correspondre à la fréquence de modification de la base de données.

Pour plus d'informations sur l'utilisation d'un gestionnaire de domaine maître de secours, voir «Modification d'un gestionnaire de domaine ou d'un gestionnaire de domaine dynamique», à la page 373.

Sauvegarde des fichiers de configuration

Les fichiers de configuration utilisés par Tivoli Workload Scheduler se trouvent aux emplacements suivants :

<*TWA_home*>**/TWS**
: Pour le fichier d'options de l'utilisateur, `useropts`.

<*TWA_home*>**/TWS/*.***
: Pour les fichiers `localopts`, `Sfinal`, `Security` et `*.msg`.

<*TWA_home*>**/TWS/mozart/*.***
: Ce répertoire contient les fichiers suivants :

  `runmsgno`
  : Il attribue un numéro unique aux invites. Sur le gestionnaire de domaine maître, ce fichier ne peut pas être édité manuellement. Sur d'autres postes de travail, il ne peut être modifié que dans les circonstances décrites dans *Tivoli Workload Scheduler - Guide de dépannage*. Il n'est pas nécessaire de sauvegarder ce fichier.

  `globalopts`
  : Il permet de stocker une copie de trois propriétés globales stockées dans la base de données. Dans les versions de Tivoli Workload Scheduler antérieures à la version 8.3, il s'agissait d'un fichier modifiable qui contenait les options globales. Il n'est plus utilisé dans ce but. Il peut être édité uniquement dans les conditions indiquées à la section «Modification d'un gestionnaire de domaine maître», à la page 378. Ce fichier doit être sauvegardé après une modification.

<*TWA_home*>**/WAS/TWSprofile/properties**
: Pour le fichier de configuration du serveur d'applications, `TWSConfig.properties`

<*TWA_home*>**/WAS/TWSprofile/config**
: Il contient les autres fichiers de configuration de la WebSphere Application Server. Ne les sauvegardez pas manuellement. La section «Serveur

d'applications - sauvegarde et restauration des fichiers de configuration», à la page 417 décrit un utilitaire qui permet de les sauvegarder.

<*TWA_home*>**/TWS/schedForecast**
Pour les fichiers du plan prévisionnel.

<*TWA_home*>**/TWS/schedlog**
Pour les fichiers de plan archivés.

<*TWA_home*>**/TWS/schedTrial**
Pour les fichiers de plan d'essai.

Aucune liste détaillée de tous les fichiers n'est fournie, car les fichiers sont trop nombreux. Sauvegardez tous les fichiers dans ces répertoires.

Remarque : L'outil **tws_inst_pull_info** (décrit dans *Tivoli Workload Scheduler - Guide d'identification des problèmes*) est fourni pour envoyer des informations au support, mais peut également servir à exécuter une sauvegarde d'une base de données DB2 et de certains fichiers de configuration.

Sauvegarde des fichiers journaux

Exécutez des sauvegardes régulières hors ligne de tous les fichiers journaux, en les identifiant à partir des informations fournies dans la section consacrée aux fichiers journaux et aux fichiers de trace dans *Tivoli Workload Scheduler - Guide d'identification des problèmes*.

Si vous utilisez l'outil **tws_inst_pull_info** pour la sauvegarde (voir la documentation dans le même guide), vous n'avez pas à sauvegarder ces fichiers séparément.

Réorganisation de la base de données

La base de données nécessite une maintenance régulière, dont voici le détail :

DB2 La base de données DB2 a été configurée pour assurer sa propre maintenance, de sorte que les interventions de l'utilisateur sont rarement nécessaires. DB2 vérifie régulièrement la base de données à l'aide d'une routine interne, DB2 détermine quand cette opération doit être effectuée à l'aide d'une politique par défaut. Le cas échéant, il est possible de modifier cette règle ou de la désactiver afin que DB2 n'effectue pas de maintenance automatique interne. A l'aide des informations statistiques obtenues par DB2 via cette routine, DB2 ajuste ses paramètres de traitement interne pour optimiser ses performances.

Vous pouvez également exécuter cette routine manuellement si vous trouvez que les performances de DB2 ont baissé ou si vous craignez qu'elles ne soient diminuées suite à l'ajout d'un grand nombre de données. La routine est intégrée à un outil appelé **dbrunstats**, qui peut être exécuté pour améliorer les performances pendant le traitement des données par DB2, sans entraîner d'interruption.

Il est également possible de réorganiser la base de données physiquement et logiquement, à l'aide du script **dbreorg**. Celui-ci recrée l'*espace table* à l'aide de ses algorithmes internes afin de déterminer la meilleure organisation physique et logique des tables et index sur le disque. Cette opération peut prendre un certain temps, pendant lequel Tivoli Workload Scheduler doit être arrêté, mais vous disposez ensuite d'une base de données bien réorganisée, incluant les dernières modifications importantes.

L'utilisation de ces outils est décrite dans «Tâches d'administration - DB2», à la page 325.

Ces outils sont des implémentations des fonctions DB2 standard. Si vous êtes un utilisateur expérimenté de DB2, vous pouvez obtenir les mêmes résultats avec les fonctions standard de DB2. Pour plus de détails, allez sur le centre de documentation de DB2, version 9.5, à l'adresse : http://publib.boulder.ibm.com/infocenter/db2luw/v9r5//index.jsp.

Oracle Pour les bases de données Oracle, consultez la documentation sur la maintenance des bases de données Oracle.

Par défaut, Oracle 10g dispose d'une procédure planifiée interne pour collecter les statistiques de base de données : si le planning par défaut n'est pas modifié, Oracle 10g optimise automatiquement ses performances en exécutant cette procédure quotidiennement. Oracle 9i ne dispose pas du même planning par défaut, mais pourrait être configurée pour ce faire.

Gestion du système de fichiers

Certains systèmes de fichiers et répertoires nécessitent une maintenance périodique. Vous trouverez de plus amples détails dans les sections suivantes :

- «Eviter les systèmes de fichiers complets»
- «Fichiers journaux et fichiers archivés», à la page 321
- «Fichiers temporaires», à la page 325
- «Gestion de la taille des fichiers de file d'attente des messages d'événement», à la page 325

Eviter les systèmes de fichiers complets

La tâche de maintenance qui est peut-être la plus importante est celle qui consiste à contrôler régulièrement le ou les systèmes de fichiers où Tivoli Workload Scheduler est installé, en particulier sur le gestionnaire de domaine maître.

Tivoli Workload Scheduler dispose d'un certain nombre de fichiers dont le volume peut augmenter, soit en raison d'une utilisation extensive (fichier Symphony, par exemple), soit en cas de problème au niveau du réseau (fichiers de messages, par exemple). S'il n'est pas possible d'augmenter la taille du fichier Symphony en particulier, afin qu'il puisse contenir tous les enregistrements nécessaires, il risque d'être endommagé. Si cela se produit sur un agent tolérant aux pannes ou sur un gestionnaire de domaine autre que le gestionnaire de domaine maître, il existe une procédure de reprise (voir le *Tivoli Workload Scheduler - Guide de dépannage*). Si le fichier Symphony est endommagé sur le gestionnaire de domaine maître, vous n'avez d'autre possibilité que de redémarrer Tivoli Workload Scheduler (vous perdez alors la charge de travail du plan en cours).

Il est également *très important* de contrôler l'espace disponible dans le système de fichiers du gestionnaire de domaine maître sur lequel le fichier Symphony est généré afin qu'il y ait toujours suffisamment d'espace pour autoriser une augmentation de volume du fichier Symphony d'une part, pour palier à d'éventuels pics de charges de travail, et des fichiers de messages d'autre part, en cas d'incident au niveau du réseau. Lorsque vous maîtriserez mieux votre charge de travail et votre réseau, vous apprendrez à définir les limites acceptables pour l'espace disque disponible.

Il est possible de donner une estimation à l'avance de la taille approximative du fichier Symphony. Il contient des éléments concernant à la fois le plan (voir

tableau 58) et la base de données (voir tableau 59). Faites une estimation du nombre d'éléments dans chaque catégorie, multipliez ce chiffre par la taille indiquée en octets et additionnez-les pour obtenir la taille approximative du fichier Symphony :

Tableau 58. Algorithme de calcul de la taille approximative des données du plan du fichier Symphony

Données du plan en cours du fichier Symphony	**Octets par instance**
Par instance de Planificateur de travaux :	512
Par instance de travail :	512
Par chaîne de travail "docommand" > 40 octets :	Longueur de la chaîne "docommand"
Par invite ad hoc :	512
Par dépendance de fichier :	512
Par invite de reprise :	512
Par travail de reprise :	512

Tableau 59. Algorithme de calcul de la taille approximative des données de la base de données du fichier Symphony

Données du fichier Symphony à partir de la base de données (sur le gestionnaire de domaine maître)	**Octets par instance**
Par poste de travail :	512
Par ressource :	512
Par utilisateur :	256
Par invite :	512
Si l'option globale `ignoreCalendars` est définie sur *off*, par calendrier :	512

Si vous estimez que l'espace disque devient trop restreint et que vous ne pouvez pas l'augmenter de façon dynamique, vous devez créer un gestionnaire de domaine maître de secours avec beaucoup plus d'espace dans sont système de fichiers, puis utiliser la commande **`switchmgr`** pour que le gestionnaire de secours devienne votre nouveau gestionnaire de domaine. Vous trouverez des instructions à ce propos pour tout type de gestionnaire de domaine dans «Modification d'un gestionnaire de domaine ou d'un gestionnaire de domaine dynamique», à la page 373, et plus particulièrement pour un gestionnaire de domaine maître, dans «Modification d'un gestionnaire de domaine maître», à la page 378.

Surveillance de l'espace disque utilisé par Tivoli Workload Scheduler

Vous pouvez utiliser l'automatisation EDWA (Event-Driven Workload Automation) pour surveiller l'espace disque utilisé par Tivoli Workload Scheduler et lancer un ensemble prédéfini d'actions au déclenchement d'un ou de plusieurs événements spécifiques. Vous pouvez également l'utiliser pour surveiller l'espace disque utilisé, vérifier qu'il est suffisant pour la génération des fichiers Symphony et des fichiers journaux, et permettre le fonctionnement normal du produit. Pour plus d'informations à propos de l'automatisation de charge de travail gérée par les événements, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Le fichier .XML suivant contient la définition d'un modèle de règle d'événement pour surveiller le taux de remplissage du disque en pourcentage. Cette règle

d'événement appelle le fournisseur d'action MessageLogger pour écrire un message dans un fichier journal figurant dans une base de données d'audit interne. Si la condition décrite dans la règle existe déjà lorsque vous déployez la règle, l'événement associé n'est pas généré. Pour plus d'informations à propos du fournisseur d'action MessageLogger, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence* :

```
<?xml version="1.0"?>
<eventRuleSet xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://www.ibm.com/xmlns/prod/tws/1.0/event-management/rules"
  xsi:schemaLocation="http://www.ibm.com/xmlns/prod/tws/1.0/event-management/rules
   http://www.ibm.com/xmlns/prod/tws/1.0/event-management/rules/EventRules.xsd">
 <eventRule name="FILESYSTEMFULL" ruleType="filter" isDraft="yes">
  <eventCondition name="twsDiskMonEvt1" eventProvider="TWSApplicationMonitor" eventType="TWSDiskMonitor">
   <scope>
    * Disk is filling up
   </scope>
   <filteringPredicate>
    <attributeFilter name="FillingPercentage" operator="ge">
     <value>filling_percentage</value>
    </attributeFilter>
    <attributeFilter name="Workstation" operator="eq">
     <value>workstation_name</value>
    </attributeFilter>
    <attributeFilter name="SampleInterval" operator="eq">
     <value>sample_interval</value>
    </attributeFilter>
    <attributeFilter name="MountPoint" operator="eq">
     <value>mount_point</value>
    </attributeFilter>

   </filteringPredicate>
  </eventCondition>
  <action actionProvider="MessageLogger" actionType="MSGLOG" responseType="onDetection">
   <scope>
    OBJECT=ADWDAD MESSAGE=Disk is filling up
   </scope>
   <parameter name="ObjectKey">
    <value>object_key</value>
   </parameter>
   <parameter name="Severity">
    <value>message_severity</value>
   </parameter>
   <parameter name="Message">
    <value>log_message</value>
   </parameter>
  </action>
 </eventRule>
```

Où :

filling_percentage
: Pourcentage de remplissage. Les opérateurs pris en charge sont les suivants :

  ge
  : déclenche la génération d'événement lorsque le pourcentage de remplissage du disque dépasse la valeur de seuil. Cet événement est généré uniquement la première fois que le pourcentage de remplissage de disque indiqué est atteint. Si vous redémarrez l'agent SSM et que le pourcentage de remplissage dépasse la valeur de seuil, l'événement est généré une nouvelle fois. Le tableau 60, à la page 321 fournit un exemple dans lequel l'opérateur **ge** est défini avec la valeur 70 %.

Tableau 60. Exemple d'opérateur ge

Nom de la boîte aux lettres	Pourcentage de remplissage	Action
Sample (0)	>= 70 %	événement non généré
Sample (0)	< 70 %	événement non généré
Sample (n-1)	< 70 %	événement non généré
Sample (n)	>= 70 %	événement généré
Sample (n+1)	>= 70 %	événement non généré

le déclenche la génération d'événement lorsque le pourcentage de remplissage du disque est inférieur à la valeur de seuil. Cet événement est généré uniquement la première fois que le pourcentage de remplissage de disque indiqué est atteint. Si vous redémarrez l'agent SSM et que le pourcentage de remplissage est inférieur à la valeur de seuil, l'événement ne sera généré que lorsque le pourcentage de remplissage dépassera la valeur de seuil, puis repassera à nouveau au-dessous de la valeur de seuil. Le tableau 61 fournit un exemple dans lequel l'opérateur **le** est défini avec la valeur 50 % :

Tableau 61. Exemple d'opérateur le

Nom de la boîte aux lettres	Pourcentage de remplissage	Action
Sample (0)	<= 50 %	événement non généré
Sample (0)	> 50 %	événement non généré
Sample (n-1)	> 50 %	événement non généré
Sample (n)	<= 50 %	événement généré
Sample (n+1)	<= 50 %	événement non généré

workstation_name
Poste de travail sur lequel est généré l'événement.

sample_interval
Intervalle, exprimé en secondes, de surveillance du pourcentage de remplissage du disque.

mount_point
Est le point de montage du fichier sur lequel Tivoli Workload Scheduler est installé. par exemple : "C:" sur les systèmes Windows ou "/" sur les systèmes UNIX.

object_key
Clé identifiant l'objet auquel se rattache le message.

message_severity
Gravité du message.

log_message
Message à consigner dans le journal.

Fichiers journaux et fichiers archivés

Plusieurs types d'activités de Tivoli Workload Scheduler génèrent des fichiers journaux. D'autres activités génèrent des fichiers qui sont archivés après utilisation. Le tableau 62, à la page 322 en donne le détail :

Tableau 62. Maintenance des fichiers journaux et des fichiers de trace

Activité	Description	Emplacement	Méthode de maintenance
agent tolérant aux pannes	Chaque processus Tivoli Workload Scheduler consigne ses activités dans des fichiers de messages journaux et des fichiers de messages de trace :		
	Messages de journal Ces messages sont destinés à être utilisés par vous directement. Ils fournissent des informations, des messages d'erreur et des avertissements relatifs aux processus.	**netman** <*TWA_home*>/TWS/stdlist/logs/<*yyyymmdd*>_NETMAN.log **Autres processus** <*TWA_home*>/TWS/stdlist/logs/<*yyyymmdd*>_TWSMERGE.log Il s'agit de la situation par défaut. Vous pouvez définir une option dans le fichier `localopts` pour créer des fichiers journaux distincts pour les principaux processus.	**rmstdlist**
	Messages de fonction de trace Il s'agit de messages générés lors de problèmes que vous ne pouvez probablement pas résoudre sans l'aide du service de support logiciel IBM.	**netman** <*TWA_home*>/TWS/stdlist/traces/<*yyyymmdd*>_NETMAN.log **Autres processus** <*TWA_home*>/TWS/stdlist/traces/<yyyymmdd>_TWSMERGE.log Il s'agit de la situation par défaut. Vous pouvez définir une option dans le fichier `localopts` pour créer des fichiers de trace distincts pour les principaux processus.	
Gestion des travaux du gestionnaire de domaine maître	Le processus du gestionnaire de travaux du gestionnaire de domaine maître procède à l'archivage du fichier `Symphony` de la période précédente.	<*TWA_home*>/TWS/schedlog	Manuel
Travail	Chaque travail exécuté sous le contrôle de Tivoli Workload Scheduler génère un fichier de sortie. Ces fichiers sont archivés.	<*TWA_home*>/TWS/stdlist/<*date*> où <*date*> est au format `yyyy.mm.dd`	**rmstdlist**

Tableau 62. Maintenance des fichiers journaux et des fichiers de trace (suite)

Activité	Description	Emplacement	Méthode de maintenance
Agent dynamique	**Messages de journal**	**Windows** <*TWA_home*>\TWS\stdlist\JM\JobManager_message.log **UNIX** <*TWA_home*e>/TWS/stdlist/JM/JobManager_message.log	Un nettoyage régulier est opéré par le biais de la configuration de plusieurs paramètres. Voir «Maintenance régulière», à la page 66.
	Messages de fonction de trace	**Windows** • <*TWA_home*>\TWS\stdlist\JM\ITA_trace.log • <*TWA_home*>\TWS\stdlist\JM\JobManager_trace.log • <*TWA_home*>\TWS\JavaExt\logs\javaExecutor0.log **UNIX** • <*TWA_home*>/TWS/stdlist/JM/ITA_trace.log • <*TWA_home*>/TWS/stdlist/JM/JobManager_trace.log • <*TWA_home*>/TWS/JavaExt/logs/javaExecutor0.log	
	Les travaux avec options avancées	**Windows** <*TWA_home*>\TWS\stdlist\JM\<*date*> **UNIX** <*TWA_home*>/TWS/stdlist/JM/<*date*> où <*date*> est au format yyyy.mm.dd	
Création de plans prévisionnels et de plans d'essai	La création de plan prévisionnels et de plans d'essai génère un fichier journal.	**Essai** <*TWA_home*>/TWS/schedTrial/*.log **Autres processus** <*TWA_home*>/TWS/schedForecast/*.log	Manuel
Audit	La fonction d'audit génère des fichiers journaux.	<*TWA_home*>/TWS/audit	Manuel

Tableau 62. Maintenance des fichiers journaux et des fichiers de trace (suite)

Activité	Description	Emplacement	Méthode de maintenance
UDB DB2	DB2 consigne ses activités.	Des informations concernant l'emplacement et la méthode de visualisation des fichiers journaux de DB2 sont fournies dans la documentation de DB2. Accédez au Centre de documentation pour DB2 (voir *Tivoli Workload Automation : Publications* pour obtenir le lien). Le fichier principal à vérifier est le fichier `db2diag.log` ; il s'agit du principal fichier de diagnostic de DB2 qui, sans intervention, se développe à l'infini sans réutiliser l'espace perdu. Cela ne s'applique pas cependant aux fichiers journaux de la base de données utilisés par Tivoli Workload Scheduler, lesquels sont configurés pour une réutilisation circulaire de l'espace disque, de sorte que leur taille ne dépasse pas une valeur maximale.	Voir la documentation DB2.
base de données Oracle	Oracle consigne ses activités.	Consultez la documentation Oracle.	Consultez la documentation Oracle.

WebSphere Application Server	Le serveur d'applications génère des fichiers journaux.	*<chemin_profil_WAS>*`/logs`
		
		où la valeur par défaut du chemin *chemin_profil_WAS* est *<TWA_home>*`/WAS/TWSprofile`
		
		Dans Dynamic Workload Console :
		rép_profil_JazzSM`/logs`
		
		où la valeur par défaut de *rép_profil_JazzSM* est :
		
		Sur les systèmes d'exploitation Windows
		`C:\Program Files\IBM\JazzSM\profile`
		
		Sur les systèmes d'exploitation UNIX
		`/opt/IBM/JazzSM/profile`
			Manuel

Avent de surveillance SSM Netcool (non pris en charge sur les systèmes IBM i)	L'agent génère des fichiers journaux.	*<TWA_home>*`/ssm/Log/`
		`ssmagent.log`
		`traps.log`
			Manuel

Autre	D'autres types d'activités génèrent également des fichiers de trace et des fichiers journaux.	*<TWA_home>*`/TWS/methods`	Manuel

Pour contrôler l'augmentation du volume de ces répertoires, le plus simple est de définir le nombre de jours pendant lesquels ces fichiers sont nécessaires, puis de programmer un travail Tivoli Workload Scheduler pour supprimer tous les fichiers antérieurs au nombre de jours indiqué. Utilisez la commande **rmstdlist** pour les

fichiers journaux de processus et de travaux, et utilisez une routine de vérification manuelle et de suppression par date pour les autres. Assurez-vous qu'aucun processus n'utilise ces fichiers lorsque vous procédez à ces activités.

Reportez-vous au *Tivoli Workload Scheduler : Guide d'utilisation et de référence* pour plus de détails sur la commande **rmstdlist**.

Remarque : Pour un même scénario, il se peut que la commande **rmstdlist** ne produise pas les mêmes résultats sur des plateformes différentes. Ceci est dû au fait que sur les plateformes UNIX, la commande utilise l'option *–mtime* de la commande **find**, qui est interprétée différemment selon les plateformes UNIX.

Fichiers temporaires

Le Tivoli Workload Scheduler gestionnaire de domaine maître utilise des fichiers temporaires, situés dans *<TWA_home>*/TWS/tmp ou /tmp et nommés TWS*<XXXX>*, lors de la compilation de nouvelles bases de contrôle de production. Une fois la compilation terminée, ces fichiers sont supprimés.

Ce répertoire contient également les fichiers d'installation et les fichiers journaux de Tivoli Workload Scheduler.

Gestion de la taille des fichiers de file d'attente des messages d'événement

Cette publication contient les informations suivantes en ce qui concerne la gestion de la taille des fichiers de file d'attente de messages d'événement :

- Voir «Planification de la capacité des files d'attente», à la page 243 pour plus d'informations sur la planification d'espace pour les files d'attente des messages d'événement (et sur l'utilisation de **evtsize** pour redimensionner les files d'attente).
- Voir «Gestion du processeur d'événements», à la page 407 pour plus d'informations sur la gestion de la file d'attente d'événements EIF
- Voir «Espace disque», à la page 454 pour plus d'informations sur l'impact de la tolérance accrue aux pannes sur les files d'attente de messages.
- Voir «Répartition de la charge de travail», à la page 451 pour plus d'informations sur la manière d'éviter les goulets d'étranglement dans la file d'attente Mailbox.msg.

Tâches d'administration - DB2

Cette section explique comment réaliser certaines tâches d'administration spécifiques sur DB2, comme suit :

- «Modification des mots de passe DB2»
- «Localisation des outils DB2», à la page 326
- «Droits d'utilisateur pour l'exécution des outils DB2», à la page 326
- «Administration de la fonction de maintenance de DB2», à la page 326
- «Réorganisation de la base de données DB2», à la page 329
- «Surveillance de la mémoire de liste des verrous», à la page 330

Modification des mots de passe DB2

Pour modifier les mots de passe utilisés par DB2 autres que le mot de passe *<utilisateur_TWS>* ou les mots de passe des ID utilisateur utilisés par Tivoli Workload Scheduler pour accéder à la base de données (voir «Modification des

mots de passe Tivoli Workload Scheduler clés», à la page 382) suivez les instructions de la documentation DB2 ; elles n'entraînent pas un impact direct sur Tivoli Workload Scheduler.

Localisation des outils DB2

Tivoli Workload Scheduler est fourni avec quelques outils qui vous permettent d'effectuer les tâches d'administration suivantes pour DB2 :

- Exécutez le programme de statistiques DB2 afin d'optimiser les performances de DB2 (dbrunstats). Voir «Exécution manuelle de la maintenance de DB2», à la page 328 pour obtenir une description complète de l'utilisation de cet outil.
- Réorganisez la base de données (dbreorg). Voir «Réorganisation de la base de données DB2», à la page 329 pour obtenir une description complète de l'utilisation de cet outil.

Ces outils se trouvent dans le répertoire suivant :

`<TWA_home>/TWS/dbtools/db2/scripts`

Remarque : Certains des outils figurant dans ce répertoire sont destinés à l'utilisation de l'assistance logicielle IBM :

dbcatalog
dbsetup

N'exécutez pas ces scripts. Cela risquerait d'endommager ou d'écraser les données présentes dans votre base de données.

Droits d'utilisateur pour l'exécution des outils DB2

Les outils DB2 doivent être exécutés par un utilisateur doté des droits suivants :

- Droits administrateur DB2 – l'utilisateur doit être défini dans DB2 en tant qu'Administrateur DB2
- Accès complet (777) au répertoire d'installation Tivoli Workload Scheduler

Administration de la fonction de maintenance de DB2

Au moment de l'installation, la fonction de maintenance automatique de DB2 est activée, ce qui signifie que DB2 vérifie régulièrement si de nouvelles statistiques de base de données doivent être collectées afin de pouvoir procéder à la maintenance, en réglant les paramètres de façon à optimiser les performances.

Cette section décrit l'administration de la maintenance automatique : modification de la manière et du moment de son exécution, désactivation et réactivation de la maintenance, et exécution manuelle de la maintenance. Reportez-vous aux rubriques suivantes :

- «Modification de la règle de maintenance automatique de DB2»
- «Désactivation de la maintenance automatique», à la page 327
- «Activation de la maintenance automatique», à la page 327
- «Exécution manuelle de la maintenance de DB2», à la page 328

Modification de la règle de maintenance automatique de DB2

Pour savoir quand et comment les statistiques utilisées par la maintenance automatique doivent être collectées, DB2 utilise une politique par défaut qui peut être personnalisée. La procédure est la suivante :

1. Cliquez avec le bouton droit de la souris sur la base de données dans le Centre de contrôle DB2 et sélectionnez l'option de **configuration automatique de la maintenance** dans le menu.
2. Suivez les instructions de l'assistant : modifiez tous les paramètres de la règle par défaut susceptibles d'améliorer la façon dont DB2 choisit le moment d'exécuter la maintenance automatique.

Désactivation de la maintenance automatique

Si vous voulez contrôler uniquement manuellement la base de données, désactivez la maintenance automatique de la façon suivante :

1. Vérifiez que l'utilisateur qui va exécuter la procédure dispose des droits appropriés (voir «Droits d'utilisateur pour l'exécution des outils DB2», à la page 326)
2. Sur l'ordinateur sur lequel se trouve le serveur DB2, ouvrez un interpréteur de commandes DB2, comme suit :

   UNIX Procédez comme suit :

   a. Lancez la commande **su - db2inst1** ou accédez au sous-répertoire `sqllib` du répertoire racine du propriétaire de l'instance DB2 (par défaut, *db2inst1*)
   b. Exécutez la commande **. ./db2profile**

   Windows

   Dans le menu **Démarrer**, sélectionnez **Programmes → IBM DB2 → Outils ligne de commande → Fenêtre Commande**
3. Pour vérifier l'initialisation correcte de l'interpréteur de commandes, lancez la commande **db2** et assurez-vous qu'elle est reconnue.
4. Lancez la commande **quit** pour quitter le mode DB2 Processeur.
5. Emettez la commande suivante :

   ```
   db2 UPDATE DB CFG FOR <database_name> USING AUTO_MAINT OFF
   ```

   où *<nom_base_données>* est le nom de la base de données Tivoli Workload Scheduler (le nom installé par défaut est *TWS* ; fournissez cette valeur sauf si vous l'avez modifiée).
6. Pour rendre ces changements effectifs, désactivez et réactivez tous les clients DB2 ou redémarrez le DB2 instance (à l'aide de **db2stop** et de **db2start**).

Activation de la maintenance automatique

Pour activer à nouveau la maintenance automatique, procédez comme suit :

1. Vérifiez que l'utilisateur qui va exécuter la procédure dispose des droits appropriés (voir «Droits d'utilisateur pour l'exécution des outils DB2», à la page 326)
2. Sur l'ordinateur sur lequel se trouve le serveur DB2, ouvrez un interpréteur de commandes DB2, comme suit :

   UNIX Procédez comme suit :

   a. Lancez la commande **su - db2inst1** ou accédez au sous-répertoire `sqllib` du répertoire racine du propriétaire de l'instance DB2 (par défaut, *db2inst1*)
   b. Exécutez la commande **. ./db2profile**

   Windows

   Dans le menu **Démarrer**, sélectionnez **Programmes → IBM DB2 → Outils ligne de commande → Fenêtre Commande**

3. Pour vérifier l'initialisation correcte de l'interpréteur de commandes, lancez la commande **db2** et assurez-vous qu'elle est reconnue.
4. Lancez la commande **quit** pour quitter le mode DB2 Processeur.
5. Emettez la commande suivante :

   db2 UPDATE DB CFG FOR <database_name> USING AUTO_MAINT ON

   où <*nom_base_données*> est le nom de la base de données Tivoli Workload Scheduler (le nom installé par défaut est *TWS* ; fournissez cette valeur sauf si vous l'avez modifiée).
6. Pour rendre ces changements effectifs, désactivez et réactivez tous les clients DB2 ou redémarrez le DB2 instance (à l'aide de **db2stop** et de **db2start**).

Exécution manuelle de la maintenance de DB2

Cette section décrit l'exécution du processus de maintenance DB2 à la demande, au lieu d'attendre que DB2 le fasse en fonction de sa politique de maintenance automatique. Ce processus est exécuté grâce à l'outil **dbrunstats**, qu'il est possible d'exécuter à volonté sans arrêter DB2 ni interrompre son fonctionnement.

Pour exécuter cet outil, suivez la procédure suivante :

1. Localisez les outils DB2 : voir «Localisation des outils DB2», à la page 326.
2. Vérifiez que l'utilisateur qui va exécuter la procédure dispose des droits appropriés (voir «Droits d'utilisateur pour l'exécution des outils DB2», à la page 326)
3. Ouvrez un shell DB2, comme suit :

   UNIX Procédez comme suit :

   a. Lancez la commande **su - db2inst1** ou accédez au sous-répertoire `sqllib` du répertoire racine du propriétaire de l'instance DB2 (par défaut, *db2inst1*)
   b. Exécutez la commande **. ./db2profile**

   Windows

   Dans le menu **Démarrer**, sélectionnez **Programmes** → **IBM DB2** → **Outils ligne de commande** → **Fenêtre Commande**
4. Pour vérifier l'initialisation correcte de l'interpréteur de commandes, lancez la commande **db2** et assurez-vous qu'elle est reconnue.
5. Lancez la commande **quit** pour quitter le mode DB2 Processeur.
6. Depuis l'intérieur du shell, accédez au répertoire <*TWA_home*>`/TWS/dbtools/db2/scripts`
7. Exécutez le script :

   UNIX **dbrunstats.sh database [user [password]]**

   Windows

   dbrunstats database [user [password]]

   Où :

   database

   Nom de la base de données :

   - Si l'exécution s'effectue à partir de l'ordinateur sur lequel le serveur DB2 est installé, le nom par défaut installé est *TWS*. Indiquez cette valeur à moins que vous ne l'ayez modifiée.
   - Si l'exécution s'effectue à partir de l'ordinateur sur lequel le client DB2 est installé, le nom par défaut installé est *BD_TWS*. Indiquez cette valeur à moins que vous ne l'ayez modifiée.

utilisateur
Utilisateur d'administration de DB2. En cas d'omission, c'est le nom de l'utilisateur qui exécute la commande qui est utilisé.

password
Mot de passe de l'utilisateur d'administration de DB2. En cas d'omission, il sera demandé de façon interactive.

Le script est exécuté, envoyant plusieurs messages indiquant sa progression et sont aboutissement. A la fin (cette exécution prenant assez peu de temps), les paramètres de performances de la base de données sont réinitialisés de façon à optimiser les performances.

Réorganisation de la base de données DB2

Cette outil permet à la base de données de réorganiser physiquement les index et tables de données afin d'optimiser l'utilisation de l'espace disque et de faciliter l'accès aux données. Ce processus prend un certain temps et nécessite une sauvegarde de la base de données et l'arrêt de Tivoli Workload Scheduler, mais une fois terminé, vous disposez d'une base de données entièrement réorganisée.

Pour réorganiser la base de données, procédez comme suit :

1. Sauvegardez la base de données Tivoli Workload Scheduler en suivant la méthode décrite à la section «Sauvegarde de la base de données sur un support de stockage autonome», à la page 315.
2. Arrêtez tous les processus Tivoli Workload Scheduler. Pour plus d'informations, voir la section «Suppression des liaisons et arrêt de Tivoli Workload Scheduler», à la page 392.
3. Vérifiez que l'utilisateur qui va exécuter la procédure dispose des droits appropriés (voir «Droits d'utilisateur pour l'exécution des outils DB2», à la page 326)
4. Ouvrez un shell DB2, comme suit :

   UNIX Procédez comme suit :

   a. Lancez la commande **su - db2inst1** ou accédez au sous-répertoire `sqllib` du répertoire racine du propriétaire de l'instance DB2 (par défaut, *db2inst1*)
   b. Exécutez la commande **. ./db2profile**

   Windows
   Dans le menu **Démarrer**, sélectionnez **Programmes → IBM DB2 → Outils ligne de commande → Fenêtre Commande**
5. Pour vérifier l'initialisation correcte de l'interpréteur de commandes, lancez la commande **db2** et assurez-vous qu'elle est reconnue.
6. Lancez la commande **quit** pour quitter le mode DB2 Processeur.
7. Depuis l'intérieur du shell, accédez au répertoire `<TWA_home>/TWS/dbtools/db2/scripts`
8. Exécutez le script :

   UNIX **dbreorg.sh base_de_données [utilisateur [password]]**

   Windows
   dbreorg base_de_données [utilisateur [password]]

   Où :

   database
   Nom de la base de données :

- Si l'exécution s'effectue à partir de l'ordinateur sur lequel le serveur DB2 est installé, le nom par défaut installé est *TWS*. Indiquez cette valeur à moins que vous ne l'ayez modifiée.
- Si l'exécution s'effectue à partir de l'ordinateur sur lequel le client DB2 est installé, le nom par défaut installé est *BD_TWS*. Indiquez cette valeur à moins que vous ne l'ayez modifiée.

utilisateur
Utilisateur d'administration de DB2. En cas d'omission, c'est le nom de l'utilisateur qui exécute la commande qui est utilisé.

password
Mot de passe de l'utilisateur d'administration de DB2. En cas d'omission, il sera demandé de façon interactive.

Le script est exécuté, envoyant plusieurs messages indiquant sa progression et sont aboutissement.

9. Redémarrez Tivoli Workload Scheduler.

Surveillance de la mémoire de liste des verrous

Si la mémoire allouée par DB2 à cette liste de verrous est presque saturée, DB2 peut être forcé dans le cadre d'une *escalade de verrous*", où il commence à verrouiller l'ensemble des tables au lieu de lignes individuelles de la table, ce qui augmente les risques de blocage.

Cela se produit particulièrement en cas de longues transactions, comme la création ou l'extension d'un plan (de production, d'essai ou de prévision).

Pour éviter cet incident, définissez la notification automatique dans le Centre de santé DB2, de manière à pouvoir être informé de l'occurrence d'un incident dans la liste de verrous.

Toutefois, si vous pensez que la situation de blocage est en cours, suivez la procédure ci-dessous pour le vérifier :

1. WebSphere Application Server étant actif, connectez-vous en tant qu'administrateur DB2 au serveur DB2, par exemple.
   su - db2inst1
2. Exécutez la commande suivante pour déterminer l'emplacement de la base de données Tivoli Workload Scheduler :
   db2 list active databases
   La sortie peut se présenter comme suit :
   ```
   Nom de la base de données                  = TWS
   Applications connectées = 2
   Chemin de la base de données                  = /home/db2inst1/db2inst1/NODE0000/SQL00002/
   ```
3. Exécutez :
   cd <Database path>/db2event/db2detaildeadlock
4. Connectez-vous à la base de données Tivoli Workload Scheduler, par exemple :
   db2 connect to TWS
5. Videz le moniteur d'événements qui recherche les blocages (actif par défaut) grâce à la commande suivante :
   db2 flush event monitor db2detaildeadlock
6. Déconnectez-vous de la base de données avec :
   db2 terminate

7. Obtenez la sortie du moniteur d'événements avec :

   db2evmon -path . > deadlock.out

   Le fichier `deadlock.out` contient désormais l'historique complet des blocages depuis l'opération de vidage précédente.

8. Pour savoir si des blocages ont eu lieu et à quel moment, exécutez la commande suivante :

   grep "Deadlock detection time" deadlock.out

   La sortie peut se présenter comme suit :

   ```
   Heure de détection du blocage : 11/07/2008 13:02:10.494600
   Heure de détection du blocage : 11/07/2008 14:55:52.369623
   ```

9. Cependant, l'occurrence d'un blocage ne signifie pas nécessairement que la mémoire de la liste de verrous est inadaptée. Pour le savoir, vous devez établir une relation avec l'escalade de verrous. Pour savoir si des problèmes liés à l'escalade de verrous se produit avant les blocages, exécutez la commande suivante :

   grep "Requesting lock as part of escalation: TRUE" deadlock.out

   La sortie peut se présenter comme suit :

   ```
   Requesting lock as part of escalation: TRUE
   Requesting lock as part of escalation: TRUE
   ```

   Si une escalade de verrous s'est produite en liaison avec des blocages, il peut s'avérer judicieux de modifier les valeurs des paramètres ci-dessous.

   LOCKLIST
   Permet de configurer, en pages de 4 Ko, la quantité de mémoire allouée à la gestion du verrouillage

   MAXLOCKS
   Permet de configurer le pourcentage de mémoire qu'une seule transaction peut utiliser et au-dessus de laquelle DB2 escalade, même si la mémoire n'est pas complètement saturée

10. Pour déterminer les valeurs appliquées à la base de données Tivoli Workload Scheduler, procédez comme suit :

    db2 get db cfg for TWS | grep LOCK

    La sortie peut se présenter comme suit :

    ```
    Max storage for lock list (4KB)             (LOCKLIST) = 8192
    Percent. of lock lists per application      (MAXLOCKS) = 60
    Lock timeout (sec)                        (LOCKTIMEOUT) = 180
    ```

    L'exemple illustre la sortie classique pour la base de données Tivoli Workload Scheduler si aucune modification n'a été apportée à ces valeurs :

    - "8192" = 4Ko x 8192 pages = 32 Mo de mémoire
    - "60" = 60 % – pourcentage de mémoire qu'une seule transaction peut occuper avant de déclencher une escalade
    - "180" = 3 minutes de délai d'attente pendant la période au cours de laquelle une transaction peut attendre d'obtenir un verrou

11. L'action la plus simple consiste à doubler la quantité de mémoire à 64 Mo, ce que vous pouvez faire à l'aide de la commande suivante :

    db2 update db cfg for TWS using LOCKLIST 16384 immediate

12. Vous pouvez aussi définir DB2 pour qu'il modifie automatiquement les paramètres LOCKLIST et MAXLOCKS en fonction de la quantité d'escalades survenus et de la mémoire système disponible. Ce réglage automatique est un processus lent. Mais il permet d'adapter la base de données en fonction des

besoins des données et de la configuration système disponible. Pour ce faire, il suffit d'attribuer la valeur AUTOMATIC à ces paramètres, comme suit :

```
db2 update db cfg for TWS using LOCKLIST AUTOMATIC immediate
```

DB2 répond par des messages vous indiquant que MAXLOCKS a également été défini sur la valeur AUTOMATIC.

```
SQL5146W "MAXLOCKS" must be set to "AUTOMATIC" when "LOCKLIST" is
"AUTOMATIC".
"MAXLOCKS" has been set to "AUTOMATIC"
```

Remarque : Cette fonction de réglage automatique est uniquement disponible à partir de la version 9.1 de DB2.

Tâches d'administration - Oracle

Cette section explique comment réaliser certaines tâches d'administration spécifiques relatives à la base de données Oracle.

- «Modification du mot de passe d'accès Oracle»
- «Localisation des outils Oracle»
- «Maintenance de la base de données Oracle», à la page 333
- «Obtention d'informations sur les bases de données Tivoli Workload Scheduler installées sur une instance Oracle», à la page 333
- «Droits utilisateur pour l'exécution des outils Oracle», à la page 333
- «Modification du nom d'hôte, du port ou du nom de base de données Oracle», à la page 400

Modification du mot de passe d'accès Oracle

Cette opération fait partie du processus de modification du mot de passe d'un gestionnaire de domaine maître ou d'un gestionnaire de domaine maître de secours. Voir «Modification des mots de passe Tivoli Workload Scheduler clés», à la page 382.

Localisation des outils Oracle

Tivoli Workload Scheduler est fourni avec quelques outils qui vous permettent d'effectuer les tâches d'administration suivantes pour Oracle :

- Octroyer les droits d'utilisateur pour les vues Dynamic Workload Console (`dbgrant`). Pour plus d'informations, voir l'aide en ligne Dynamic Workload Console.
- Migration de DB2 vers Oracle ou vice versa (`prepareSQLScripts`, `createdb_root_ora`, `updateSetupCmdLine`). Pour plus d'informations, voir la section «Migration des données de DB2 vers Oracle et *vice versa*», à la page 333.

Localisez ces outils dans le répertoire suivant :

`<TWA_home>/TWS/dbtools/oracle/scripts`

Remarque : Le répertoire contient également certains scripts qui sont réservés à l'utilisation de l'assistance logicielle IBM :

dbmigrate
dbpartition
dbsetup

dbupgrade
launchdb_root_ora
_migratedb_root_ora

N'exécutez pas ces scripts. Cela risquerait d'endommager ou d'écraser les données présentes dans votre base de données.

Maintenance de la base de données Oracle

Comme DB2, Oracle dispose d'une routine qui assure une maintenance périodique de la base de données et qu'il est également possible d'exécuter manuellement. Pour appeler l'outil correspondant, procédez comme suit :

```
dbms_stats.gather_schema_stats<schema_owner>
```

Pour savoir exactement quand et comment l'exécuter, consultez la documentation Oracle.

Obtention d'informations sur les bases de données Tivoli Workload Scheduler installées sur une instance Oracle

Pour déterminer quelles bases de données Tivoli Workload Scheduler sont installées sur une instance Oracle, procédez comme suit :

```
su - oracle (UNIX uniquement)
sqlplus system/<system_password>@<service_name>
SQL> select * from all_tws_schemas;
```

La sortie doit être du type suivant :

```
SCHEMA_NAME
------------------------------
MDL
mdm85utilisateur_TWS
```

Remarque :

1. Plusieurs instances de Tivoli Workload Scheduler peuvent être partagées par une instance d'Oracle, à l'aide de schémas différents.
2. Dans Oracle, les concepts de "schéma" et "utilisateur" sont identiques : par conséquent, supprimer un schéma Oracle signifie supprimer un utilisateur Oracle, en procédant comme suit :

   ```
   SQL> drop user MDL cascade;
   ```

Droits utilisateur pour l'exécution des outils Oracle

Les outils Oracle doivent être exécutés par un utilisateur doté des droits suivants :

- Droits administrateur Oracle – l'utilisateur doit être défini dans Oracle en tant qu'Administrateur Oracle
- Accès complet (777) au répertoire d'installation Tivoli Workload Scheduler

Migration des données de DB2 vers Oracle et *vice versa*

Cette section s'applique aux gestionnaires de domaine maître Tivoli Workload Scheduler et à sa sauvegarde. Elle présente la procédure de migration des données Tivoli Workload Scheduler d'un système de gestion de base de données relationnelle (SGBD relationnel) vers un autre.

Vous disposez de deux méthodes pour effectuer la migration. Elles permettent toutes les deux de migrer vos données depuis DB2 vers Oracle ou vice versa.

Migration parallèle des données
La migration s'effectue entre deux instances de Tivoli Workload Scheduler, l'une utilisant DB2 et l'autre Oracle

Reconfiguration
La base de données est migrée depuis un mécanisme de support RDBMS vers un autre, et une instance Tivoli Workload Scheduler est reconfigurée pour pointer vers une base de données différente sans installer d'autre instance.

Remarque : Aucune de ces procédures ne fait migrer les informations suivantes depuis la base de données source :

- Le plan de préproduction
- L'historique des exécutions et des statistiques des travaux
- L'état des instances de règle d'événements d'exécution. Cela signifie que toute règle d'événement complexe, où une partie de la règle a été satisfaite avant la migration de la base de données, est générée après la migration en tant que nouvelles règles. Même si les conditions suivantes de la règle d'événement sont satisfaites, l'enregistrement indiquant que la première partie de la règle a été satisfaite n'est plus disponible, de sorte que la règle ne sera jamais complètement satisfaite.

Migration parallèle des données de DB2 vers Oracle

Les étapes suivantes permettent de faire migrer toutes les définitions d'objet de planification et les options globales depuis la base de données DB2 d'une instance de la version Tivoli Workload Scheduler 9.2 vers la base de données Oracle d'une autre instance.

1. Installez une autre instance d'un Tivoli Workload Scheduler version .9.2 gestionnaire de domaine maître et faites-la pointer vers une base de données Oracle en définissant `MDM_ORACLE` comme nom de poste de travail de gestionnaire de domaine maître. Le processus d'installation définit automatiquement les postes de travail suivants dans la base de données Oracle :
   - `MDM_ORACLE_DWB` est le nom du poste de travail du courtier.
   - `MDM_ORACLE_1` est le nom du poste de travail de l'agent.
2. Utilisez **composer** ou Dynamic Workload Console pour définir cette instance en tant qu'agent tolérant aux pannes dans la base de données du gestionnaire de domaine maître en cours qui pointe vers DB2. Le nom du poste de travail de l'agent agent tolérant aux pannes doit être `MDM_ORACLE`, comme vous l'avez spécifié pendant le processus d'installation.
3. Sur le gestionnaire de domaine maître qui pointe vers DB2 exécutez la commande ou le script `dataexport` pour exporter toutes les définitions d'objets de planification et les options globales à partir de DB2. Recherchez ce fichier dans le sous-répertoire `bin` du répertoire de base Tivoli Workload Scheduler.

   Exécutez `dataexport` à partir d'une invite de commande Windows ou UNIX, comme suit :

   ```
   dataexport <source_dir> <export_dir>
   ```

   Où :

   source_dir
   Le répertoire d'installation de l'instance de Tivoli Workload Scheduler version 8.5 qui pointe vers la base de données DB2.

export_dir
Le répertoire dans lequel les fichiers d'exportation sont créés.

Par exemple :

```
dataexport.cmd F:\TWS92\twsDB2user F:\TWS92\export
```

Les définitions d'objet et les options globales sont récupérées de la base de données DB2 et placées dans le répertoire `\TWS92\export`.

4. Vérifiez que les fichiers suivants ont été créés dans `export_dir` :
   - calendars.def
   - jobs.def

     Remarque : La longueur d'enregistrement prise en charge par DB2 est de 4 095 caractères, mais elle passe à 4 000 caractères avec Oracle. Lorsque vous migrez vos définitions de travail vers Oracle, les scripts de travaux ou les commandes dont la longueur dépasse 4 000 caractères ne sont pas migrés. Dans ce cas, l'utilitaire d'importation des données remplace la définition de travail par une définition de travail factice et définit la priorité du travail à 0, garantissant ainsi que les successeurs ne sont pas exécutés.
   - `globalOpts.def`
   - `erules.def`
   - `parms.def`
   - `prompts.def`
   - `resources.def`
   - `scheds.def`
   - `topology.def`
   - `users.def` (inclut les mots de passe d'utilisateur chiffrés)
   - `vartables.def`
   - `rcgroups.def`
5. Sur le gestionnaire de domaine maître qui pointe vers DB2, procédez comme suit :
   a. Assurez-vous que l'option de réacheminement est définie à `ALL`. Exécutez :

      ```
      optman chg cf=ALL
      ```
   b. Ajoutez la nouvelle instance (que vous avez installée à l'étape 1, à la page 334 et que vous avez momentanément définie comme agent tolérant aux pannes) dans le plan actuel. Pour ce faire, exécutez :

      ```
      JnextPlan -for 0000
      ```
   c. Contrôlez que la nouvelle instance est liée en exécutant :

      ```
      conman sc
      ```
6. Ouvrez le fichier `export_dir\topology.def` et remplacez les valeurs suivantes :
   - `MDM_DB2` nom du poste de travail du gestionnaire de domaine maître dans la base de données DB2 ayant la valeur `MDM_ORACLE` nom du poste de travail du gestionnaire de domaine maître dans la base de données Oracle.
   - `MDM_DB2_DWB` nom du poste de travail du courtier dans la base de données DB2 ayant la valeur `MDM_ORACLE_DWB` nom du poste de travail du courtier dans la base de données Oracle.
   - `MDM_DB2_1` nom du poste de travail de l'agent dans la base de données DB2 ayant la valeur `MDM_ORACLE_1` nom du poste de travail de l'agent dans la base de données Oracle.
7. Sur la nouvelle instance, exécutez le script ou la commande `dataimport` pour importer toutes les définitions d'objets de planification et les options globales vers la base de données Oracle. Recherchez ce fichier dans le sous-répertoire `bin` du répertoire de base Tivoli Workload Scheduler.

Exécutez dataimport à partir d'une invite de commande Windows ou UNIX, comme suit :

```
dataimport <source_dir> <export_dir>
```

Où :

source_dir
Le répertoire d'installation de la nouvelle instance de Tivoli Workload Scheduler version 9.2 pointant vers la base de données Oracle.

export_dir
Le répertoire à partir duquel les fichiers d'exportation sont lus. Ce répertoire est le même répertoire export_dir spécifié pour dataexport.

Par exemple :

```
dataimport.cmd F:\TWS92\twsORACLEuser F:\TWS92\export
```

Les définitions d'objet et les options globales sont récupérées du répertoire F:\TWS85\export et stockées dans la base de données Oracle.

8. Sur le gestionnaire de domaine maître qui pointe vers DB2 exécutez la commande conman switchmgr pour faire pointer l'instance de Tivoli Workload Scheduler vers Oracle en tant que gestionnaire de domaine maître actif. Pour plus d'informations sur cette commande, voir *Guide d'utilisation et de référence*.

Vous avez maintenant terminé la procédure de migration des données.

Migration parallèle des données d'Oracle vers DB2

Les étapes suivantes permettent de faire migrer toutes les définitions d'objets de planification et les options globales de la base de données Oracle d'une instance de Tivoli Workload Scheduler version 9.2 DB2 vers une autre instance (récemment installée ou mise à jour vers la version 9.2).

1. Installez une nouvelle instance de Tivoli Workload Scheduler version 9.2 ou effectuez une mise à niveau d'une instance existante vers la version 9.2 en la faisant pointer vers une base de données DB2 en définissant MDM_DB2 comme nom pour le poste de travail du gestionnaire de domaine maître. Le processus d'installation définit automatiquement les postes de travail suivants dans la base de données DB2 :
   - MDM_DB2_DWB est le nom du poste de travail du courtier.
   - MDM_DB2_1 est le nom du poste de travail de l'agent.
2. Utilisez composer ou Dynamic Workload Console pour définir cette instance en tant qu'agent tolérant aux pannes dans la base de données du gestionnaire de domaine maître qui pointe vers Oracle. Le nom du poste de travail de l'agent agent tolérant aux pannes doit être MDM_DB2, comme vous l'avez spécifié pendant le processus d'installation.
3. Sur le gestionnaire de domaine maître actuel pointant vers la base de données Oracle, exécutez la commande ou le script dataexport pour exporter toutes les définitions d'objets de planification et les options globales. Recherchez ce fichier dans le sous-répertoire bin du répertoire de base Tivoli Workload Scheduler.

   Exécutez dataexport à partir d'une invite de commande Windows ou UNIX, comme suit :

   ```
   dataexport <source_dir> <export_dir>
   ```

   Où :

source_dir
Le répertoire d'installation de l'instance de Tivoli Workload Scheduler qui pointe vers la base de données Oracle.

export_dir
Le répertoire dans lequel les fichiers d'exportation sont créés.

Par exemple :

```
dataexport.cmd F:\TWS92\twsORACLEuser F:\TWS92\export
```

Les définitions d'objet et les options globales sont récupérées de la base de données Oracle et placées dans le répertoire `F:\TWS92\export`.

4. Vérifiez que les fichiers suivants ont été créés dans `export_dir` :
   - calendars.def
   - erules.def
   - jobs.def
   - globalOpts.def
   - parms.def
   - prompts.def
   - resources.def
   - scheds.def
   - topology.def
   - users.def (inclut les mots de passe d'utilisateur chiffrés)
   - vartables.def
   - rcgroups.def
5. Sur le gestionnaire de domaine maître actuel qui pointe vers Oracle, procédez comme suit :
   a. Assurez-vous que l'option de réacheminement est définie à `ALL`. Exécutez :
      ```
      optman chg cf=ALL
      ```
   b. Ajoutez la nouvelle instance (que vous avez installée à l'étape 1, à la page 336 et que vous avez momentanément définie comme agent tolérant aux pannes) dans le plan actuel. Pour ce faire, exécutez :
      ```
      JnextPlan -for 0000
      ```
   c. Contrôlez que la nouvelle instance est liée en exécutant :
      ```
      conman sc
      ```
6. Ouvrez le fichier `export_dir\topology.def` et remplacez les valeurs suivantes :
   - `MDM_ORACLE` nom du poste de travail du gestionnaire de domaine maître dans la base de données Oracle ayant la valeur `MDM_DB2` nom du poste de travail du gestionnaire de domaine maître dans la base de données DB2.
   - `MDM_ORACLE_DWB` nom du poste de travail du courtier dans la base de données Oracle ayant la valeur `MDM_DB2_DWB` nom du poste de travail du courtier dans la base de données DB2.
   - `MDM_ORACLE_1` nom du poste de travail de l'agent dans la base de données Oracle ayant la valeur `MDM_DB2_1` nom du poste de travail de l'agent dans la base de données DB2.
7. Sur la nouvelle instance, exécutez le script ou la commande `dataimport` pour importer toutes les définitions d'objets de planification et les options globales vers DB2. Recherchez ce fichier dans le sous-répertoire `bin` du répertoire de base Tivoli Workload Scheduler.

   Exécutez `dataimport` à partir d'une invite de commande Windows ou UNIX, comme suit :

   ```
   dataimport <source_dir> <export_dir>
   ```

   Où :

source_dir
Le répertoire d'installation de l'instance de Tivoli Workload Scheduler qui pointe vers la base de données DB2.

export_dir
Le répertoire à partir duquel les fichiers d'exportation sont lus. Ce répertoire est le même répertoire `export_dir` spécifié pour `dataexport`.

Par exemple :

```
dataimport.cmd F:\TWS92\twsDB2user F:\TWS92\export
```

Les définitions d'objet et les options globales sont récupérées du répertoire `F:\TWS92\export` et stockées dans DB2.

8. Sur le gestionnaire de domaine maître qui pointe vers Oracle, exécutez la commande `conman switchmgr` pour que l'instance de Tivoli Workload Scheduler pointe vers DB2 en tant que gestionnaire de domaine maître actif. Pour plus d'informations sur la commande `switchmgr`, voir *Guide d'utilisation et de référence*.

Vous avez maintenant terminé la procédure de migration des données.

Reconfiguration de DB2 vers Oracle

Suivez la procédure ci-dessous pour faire migrer toutes les définitions d'objets de planification et les options globales de la base de données DB2 d'un gestionnaire de domaine maître Tivoli Workload Scheduler version 9.2 et pour les faire pointer vers une base de données Oracle.

1. Pour exporter toutes les options globales et définitions d'objets de planification de DB2, à partir d'une invite de commande Windows ou UNIX, exécutez la commande `dataexport` ou le script situé dans le répertoire `<TWA_home>\bin` :

   ```
   dataexport <source_dir> <export_dir>
   ```

   Où :

   source_dir
   Le répertoire d'installation de Tivoli Workload Scheduler version 9.2.

   export_dir
   Le répertoire dans lequel les fichiers d'exportation sont créés.

   Par exemple :

   ```
   dataexport.cmd F:\TWS92\tws92user F:\TWS92\tws92user\export
   ```

   Les définitions d'objet et les options globales sont récupérées dans la base de données DB2 et stockées dans le répertoire `F:\TWS92\tws92user\export`.

2. Vérifiez que les fichiers suivants ont été créés dans `export_dir` :
   - calendars.def
   - erules.def
   - jobs.def
   - globalOpts.def
   - parms.def
   - prompts.def
   - resources.def
   - scheds.def
   - topology.def
   - users.def (inclut les mots de passe d'utilisateur chiffrés)

- vartables.def
- rcgroups.def

3. Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver** voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413).
4. Pour personnaliser les scripts SQL avec les paramètres obligatoires pour créer le schéma Tivoli Workload Scheduler dans la base de données Oracle, exécutez le fichier prepareSQLScripts.bat (.sh) situé dans le répertoire *TWA_home*/TWS/dbtools/oracle/scripts comme suit :
   - A partir d'un interpréteur de commandes UNIX :

     ```
     prepareSQLScripts
        -dbRoot <dbRoot>
        -dbName <dbName>
        -twsDbUser <twsDbUser>
        -twsDbPassword <twsDbUser_password>
        [-tempDir <tempDir>]
        [-dataTablespace <dataTablespace_name>]
        [-logTablespace <logTablespace_name>]
        [–tempTablespace <tempTablespace_name>]
        [-companyName <companyName>]
        [-masterDmName <masterDmName>]
        [-eifPort <eifPort>]
     ```

   - A partir d'une invite de commande Windows :

     ```
     cmd /K prepareSQLScripts
        -dbRoot <dbRoot>
        -dbName <dbName>
        -twsDbUser <twsDbUser>
        -twsDbPassword <twsDbUser_password>
        [-tempDir <tempDir>]
        [-dataTablespace <dataTablespace_name>]
        [-logTablespace <logTablespace_name>]
        [–tempTablespace <tempTablespace_name>]
        [-companyName <companyName>]
        [-masterDmName <masterDmName>]
        [-eifPort <eifPort>]
     ```

   Où :

   dbRoot Chemin d'installation du logiciel système de gestion de base de données relationnelle, à savoir répertoire *de base Oracle*.

   dbName
   Le nom de la base de données.

   twsDbUser
   Utilisateur de base de données pour Tivoli Workload Scheduler.

   twsDbPassword
   Mot de passe de l'utilisateur.

   tempDir
   Le répertoire de stockage des fichiers temporaires créés par ce processus. Sous Windows, la valeur par défaut est *<unité>*\Documents and Settings*<utilisateur_en_cours>*\Local Settings\Temp.

   dataTablespace
   Le nom de l'espace de table pour les données Tivoli Workload Scheduler. La valeur par défaut est USERS. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

logTablespace
Nom de l'espace de table pour le journal Tivoli Workload Scheduler. La valeur par défaut est `USERS`. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

tempTablespace
Le nom de l'espace de table pour les données temporaires. Par défaut, il s'agit de `TEMP`. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

companyName
Le nom de votre société. Par défaut, il s'agit de `MYCOMPANY`. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

masterDmName
Le nom du domaine maître. Par défaut, il s'agit de `MASTERDM`. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

eifPort Le port EIF. La valeur par défaut est `31123`. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez cette valeur.

Par exemple :

```
cmd /K prepareSQLScripts.bat -dbRoot D:\Oracle
                      -dbName TWS
                      -twsDbUser tws85User
                      -twsDbPassword mypassw0rd
```

Les scripts SQL sont personnalisés pour créer un schéma nommé `tws92user` sur la base de données `TWS`. Les noms des espaces de table sont les valeurs par défaut : `USERS` et `TEMP`.

5. Exécutez `createdb_root.bat (.sh)` pour créer la base de données et le schéma sur la base des spécifications de l'étape précédente. Ce fichier se trouve dans le répertoire *tempDir*`/TWA/tws92/scripts`, où *tempDir* est le paramètre que vous avez spécifié à l'étape 4, à la page 339.

   Exécutez `createdb_root` comme suit :

   - A partir d'un interpréteur de commandes UNIX

     ```
     createdb_root
        <netService>
        <oracleAdmin>
        <oracleAdminPassword>
        <twsDbUser>
        <twsDbPassword>
        <isBackupManager>
        <isPartitioned>
     ```

   - A partir d'une invite de commande Windows

     ```
     cmd /K createdb_root
        <netService>
        <oracleAdmin>
        <oracleAdminPassword>
        <twsDbUser>
        <twsDbPassword>
        <isBackupManager>
        <isPartitioned>
     ```

   Où :

   netService
   Le nom du service réseau requis pour la connexion à la base de données Oracle.

oracleAdmin
L'ID d'utilisateur de l'administrateur de base de données Oracle.

oracleAdminPassword
Le mot de passe de `oracleAdmin`.

twsDbUser
Le propriétaire du schéma Tivoli Workload Scheduler que vous avez également spécifié à l'étape 4, à la page 339.

twsDbPassword
Le mot de passe pour `twsDbUser` que vous avez également spécifié à l'étape 4, à la page 339.

isBackupManager
Spécifiez `TRUE` si vous faites migrer un gestionnaire de domaine maître de sauvegarde. Sinon, spécifiez `FALSE`.

isPartitioned
Spécifiez `TRUE` si la fonction `Partitioning` d'Oracle est activée pour la base de données. Sinon, spécifiez `FALSE`.

Par exemple :

```
createdb_root TWS SYSTEM passw1rd tws85user passw0rd FALSE TRUE
```

crée un schéma de base de données nommé `tws85user` sur la base de données `TWS`.

En cas d'erreur, si vous devez exécuter à nouveau cette étape après avoir réparé l'erreur, vous devez en premier lieu vous connecter à la base de données en tant qu'administrateur, puis abandonner l'utilisateur *twsDbUser* (`tws92user` dans l'exemple).

6. Modifiez les propriétés de la source de données en passant de DB2 à Oracle en utilisant la commande ou le script `changeDataSource.bat (.sh)` pour commuter la source de données dans WebSphere Application Server de DB2 vers Oracle.

   Pour plus de détails sur l'utilisation de cette commande, voir «Modification des propriétés de source de données», à la page 395.

   a. Effacez les valeurs des propriétés suivantes :

      ```
      DB2Type4JndiName
      DB2Type4DatabaseName
      DB2Type4ServerName
      DB2Type4PortNumber
      ```

      Remarque : Pour la propriété `DB2Type4JndiName`, remplacez la valeur par un caractère ou une chaîne qui va l'annuler.

   b. Définissez ces propriétés sur les valeurs suivantes :

      ```
      OracleType2JndiName=jdbc/twsdb
      OracleType2DatabaseName=le nom de l'instance d'Oracle
      OracleType2PortNumber=numéro de port d'écoute Oracle
      ```

   c. Définissez le chemin d'accès au pilote JDBC pour la base de données Oracle sur `ORACLE_JDBC_DRIVER_PATH=`*`racine_Oracle`*`/jdbc/lib` et le type d'instance et le nom Oracle sur `ORACLETYPE2URL=JDBC:ORACLE:OCI:@`*`nom_instance`*

7. Utilisez la commande ou le script `changeSecurityProperties.bat (.sh)` pour modifier les paramètres de sécurité suivants (voir «Modification des paramètres de sécurité», à la page 406 pour plus de détails sur l'utilisation de cette commande) :

j2cUserid
Indiquez la valeur que vous avez utilisée pour `twsDbUser` dans `prepareSQLScripts`.

j2cPassword
Saisissez le mot de passe pour `twsDbUser`.

Remarque : Après avoir mis à jour ces deux valeurs, assurez-vous que vous effacez aussi toutes les autres lignes du fichier de propriétés de sécurité avant de l'exporter à nouveau à l'aide de la commande `changeSecurityProperties`. A défaut, tous les mots de passe figurant dans le fichier sont enregistrés en tant que chaînes d'astérisques (*).

8. Modifiez le fichier `TWSConfig.properties` situé dans le répertoire `<rép_profil_WAS>/properties`, où *rép_profil_WAS* correspond au chemin du profil WebSphere Application Server que vous avez spécifié au moment de l'installation. Le chemin par défaut est : `<TWA_home>/WAS/TWSprofile`. Retirez les commentaires des lignes suivantes et modifiez-les comme indiqué :

   ```
   com.ibm.tws.dao.rdbms.rdbmsName = Oracle
   com.ibm.tws.dao.rdbms.modelSchema = <twsDbUser>
   com.ibm.tws.dao.rdbms.eventRuleSchema=<twsDbUser>
   com.ibm.tws.dao.rdbms.logSchema=<twsDbUser>
   ```

   où `twsDbUser` est le propriétaire du schéma Tivoli Workload Scheduler que vous avez également spécifié lors des étapes précédentes.

9. Pour UNIX uniquement : pour définir les chemins d'accès de la nouvelle base de données dans le profil WebSphere Application Server, exécutez la commande `updateSetupCmdLine.sh` ou le script situé dans le répertoire `<TWA_home>/TWS/dbtools/oracle/scripts` comme suit :

   ```
   updateSetupCmdLine.sh -installRoot <TWA_home> -dbRoot <DB_home>
   ```

   Où :

   –installRoot *<TWA_home>*
   Le répertoire d'installation Tivoli Workload Scheduler.

   –dbRoot *<racine_BD>*
   Le répertoire d'installation de la base de données.

10. Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413).
11. Copiez manuellement la définition de gestionnaire de domaine maître à partir du fichier `topology.def` que vous avez exporté à l'étape 1, à la page 338 (il se trouve dans `export_dir`) et utilisez `composer new` pour l'ajouter à la base de données Oracle.
12. Pour importer toutes les options globales et définitions d'objets de planification dans la base de données Oracle, à partir d'une invite de commande UNIX ou Windows, exécutez le fichier `dataimport` situé dans `<TWA_home>\bin` comme suit :

    ```
    dataimport <source_dir> <export_dir>
    ```

    Où :

    source_dir
    Répertoire d'installation de Tivoli Workload Scheduler.

export_dir
Le répertoire à partir duquel les fichiers d'exportation sont lus. Ce répertoire est le même répertoire `export_dir` spécifié pour `dataexport`.

Par exemple :

```
dataimport.cmd F:\TWS92\tws92user F:\TWS92\tws92user\export
```

Les définitions d'objet et les options globales sont récupérées dans le répertoire `F:\TWS92\tws92user\export` et stockées dans la nouvelle base de données Oracle.

13. Exécutez la commande suivante pour définir l'option de réacheminement à ALL :

    ```
    optman chg cf=ALL
    ```

14. Mettez le fichier Symphony à jour en créant un plan avec une période d'extension de 0 qui commence à la fin du plan actuel :

    ```
    JnextPlan -from début -for 0000
    ```

    où *début* est la date et l'heure auxquelles le plan en cours se termine.

Vous avez maintenant terminé la procédure de reconfiguration.

Pour migrer un gestionnaire de domaine maître de sauvegarde, procédez comme suit :

- Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413).
- Exécutez les étapes 6, à la page 341, 7, à la page 341, 9, à la page 342
- Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413).
- Définissez le paramètre `isBackupManager` de la commande `createdb_root` (script) sur `TRUE`.

Reconfiguration d'Oracle vers DB2

Les étapes suivantes permettent de migrer toute définition d'objet de planification et option globale de la base de données Oracle d'un Tivoli Workload Scheduler version 9.1 gestionnaire de domaine maître et de les faire pointer vers une base de données DB2.

1. Exécutez le script ou la commande `dataexport` pour exporter toutes les définitions d'objets de planification et les options globales depuis Oracle. Ce fichier se trouve dans le sous-répertoire `bin` du répertoire de base de Tivoli Workload Scheduler version 9.1.

   Exécutez `dataexport` à partir d'une invite de commande Windows ou UNIX, comme suit :

   ```
   dataexport <source_dir> <export_dir>
   ```

   Où :

   source_dir
   Le répertoire d'installation de Tivoli Workload Scheduler version 9.1.

   export_dir
   Le répertoire dans lequel les fichiers d'exportation sont créés.

Par exemple :

```
dataexport.cmd F:\TWS91\tws91user F:\TWS91\tws91user\export
```

Les définitions d'objet et les options globales sont récupérées dans la base de données Oracle et placées dans le répertoire `F:\TWS91\tws91user\export`.

2. Vérifiez que les fichiers suivants ont été créés dans `export_dir` :
   - calendars.def
   - erules.def
   - jobs.def
   - globalOpts.def
   - parms.def
   - prompts.def
   - resources.def
   - scheds.def
   - topology.def
   - users.def (inclut les mots de passe d'utilisateur chiffrés)
   - vartables.def
   - rcgroups.def
3. Arrêtez WebSphere Application Server comme indiqué dans «Serveur d'applications - démarrage et arrêt», à la page 410.
4. Exécutez `prepareSQLScripts.bat (.sh)` pour personnaliser les scripts SQL avec les paramètres requis pour créer la base de données Tivoli Workload Scheduler dans DB2. Recherchez ce fichier dans le répertoire *TWA_home*`/TWS/dbtools/db2/scripts`.

   Exécutez `prepareSQLScripts` à partir d'une invite de commande Windows ou UNIX, comme suit :

   - A partir d'un shell UNIX exécutez :

     ```
     prepareSQLScripts
        -dbRoot <dbRoot>
        -dbName <dbName>
        -dbLocalAdmin <dbLocalAdmin>
        -twsDbUser <twsDbUser>
        [-tempDir <tempDir>]
        [-dataTablespace <dataTablespace_name>]
        [-dataTablespacePath <dataTablespacePath>]
        [-logTablespace <logTablespace_name>]
        [-logTablespacePath <logTablespacePath>]
        [–tempTablespace <tempTablespace_name>]
        [–userTempTablespace <userTempTablespace_name>]
        [-companyName <companyName>]
        [-masterDmName <masterDmName>]
        [-eifPort <eifPort>]
     ```

   - A partir d'une invite de commande Windows, exécutez :

     ```
     cmd /K prepareSQLScripts
        -dbRoot <dbRoot>
        -dbName <dbName>
        -dbLocalAdmin <dbLocalAdmin>
        -twsDbUser <twsDbUser>
        [-tempDir <tempDir>]
        [-dataTablespace <dataTablespace_name>]
        [-dataTablespacePath <dataTablespacePath>]
        [-logTablespace <logTablespace_name>]
        [-logTablespacePath <logTablespacePath>]
        [–tempTablespace <tempTablespace_name>]
        [–userTempTablespace <userTempTablespace_name>]
        [-companyName <companyName>]
        [-masterDmName <masterDmName>]
        [-eifPort <eifPort>]
     ```

Où :

dbRoot
Le chemin d'accès du répertoire où le logiciel RDBMS est installé.

dbName
Le nom de la base de données.

dbLocalAdmin
L'ID d'utilisateur de l'administrateur de base de données locale.

twsDbUser
Utilisateur de base de données pour Tivoli Workload Scheduler.

tempDir
Le répertoire où sont placés les fichiers temporaires créés par ce processus. Sur Windows, la valeur par défaut est<*drive*>\Documents and Settings\<*current_user*>\Local Settings\Temp.

dataTablespace
Le nom de l'espace de table pour les données Tivoli Workload Scheduler. Par défaut, il s'agit de TWSDATA. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

dataTablespacePath
Chemin d'accès de l'espace de table pour les données Tivoli Workload Scheduler.

logTablespace
Nom de l'espace de table pour le journal Tivoli Workload Scheduler. La valeur par défaut est TWSLOG. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

logTablespacePath
Chemin d'accès de l'espace de table pour les fichiers journaux Tivoli Workload Scheduler.

tempTablespace
Le nom de l'espace de table pour les données temporaires. Par défaut, il s'agit de TEMP. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

userTempTablespace
Le nom de l'espace de table pour les données d'utilisateur temporaires. La valeur par défaut est USERTEMP. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

companyName
Le nom de votre société. Par défaut, il s'agit de MYCOMPANY. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

masterDmName
Le nom du domaine maître. Par défaut, il s'agit de MASTERDM. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

eifPort
Le port EIF. La valeur par défaut est 31123. Si vous avez indiqué une valeur différente au moment de l'installation, saisissez à nouveau cette valeur.

Par exemple :

```
cmd /K prepareSQLScripts.bat -dbRoot D:\DB2
                      -dbName TWS
                      -dbLocalAdmin db2admin
                      -twsDbUser tws85User
```

Les scripts SQL sont personnalisés pour créer une base de données nommée TWS dans DB2 pour l'utilisateur tws85user. Les noms des espaces de table sont les valeurs par défaut : TWSDATA, TWSLOG, TEMP et USERTEMP.

5. Exécutez createdb_root.bat (.sh) pour créer la base de données sur la base des spécifications de l'étape précédente. Ce fichier se trouve dans le répertoire *tempDir*/TWA/tws91/scripts, où *tempDir* est le paramètre que vous avez spécifié à l'étape 4, à la page 344.

   Exécutez createdb_root comme suit :

   - A partir d'un shell UNIX, exécutez :

     ```
     createdb_root
        <dbName>
        <isClientInstallation>
        <dbNodeName>
        <hostName>
        <srvPortNumber>
        <db2Admin>
        <db2AdminPwd>
        <instanceName>
        <isBackupManager>
     ```

   - A partir d'une invite de commande Windows, exécutez :

     ```
     cmd /K createdb_root
        <dbName>
        <isClientInstallation>
        <dbNodeName>
        <hostName>
        <srvPortNumber>
        <db2Admin>
        <db2AdminPwd>
        <instanceName>
        <isBackupManager>
     ```

   Où :

   dbName
   Le nom de la base de données DB2. La longueur maximale est de 5 caractères.

   isClientInstallation
   La valeur est :
   - TRUE si la base de données est un client DB2.
   - FALSE si la base de données est un serveur DB2.

   dbNodeName
   Le nom du noeud DB2.

   hostName
   Le nom d'hôte de l'ordinateur sur lequel DB2 doit être installé.

   srvPortNumber
   Le numéro de port TCP/IP utilisé pour communiquer avec le serveur DB2. La valeur par défaut est 50000.

   db2Admin
   L'ID d'utilisateur de l'administrateur de DB2.

   db2AdminPwd
   Le mot de passe pour db2Admin.

instanceName
Le nom de l'instance de serveur DB2.

isBackupManager
Spécifiez TRUE si vous faites migrer un gestionnaire de domaine maître de sauvegarde. Sinon, spécifiez FALSE.

Par exemple :

```
createdb_root TWS FALSE TWS_ND myhost 50000 db2admin passw1rd DB2 FALSE
```

crée une base de données appelée TWS sur une instance de serveur DB2 nommée DB2.

6. Utilisez la commande ou le script changeDataSource.bat (.sh) pour basculer la source de données figurant dans WebSphere Application Server d'Oracle vers DB2.

   Pour plus de détails sur l'utilisation de cette commande, voir «Modification des propriétés de source de données», à la page 395.

   a. Effacez les propriétés suivantes :

      ```
      OracleType2JndiName
      OracleType2DatabaseName
      OracleType2ServerName
      OracleType2PortNumber
      ```

   b. Définissez les propriétés suivantes :

      ```
      DB2Type4JndiName
      DB2Type4DatabaseName
      DB2Type4ServerName
      DB2Type4PortNumber
      ```

   c. Définissez le chemin d'accès au pilote JDBC pour le DB2 à la fois dans DB2_JDBC_DRIVER_PATH et dans DB2UNIVERSAL_JDBC_DRIVER_PATH (le chemin d'accès est le même pour les deux propriétés).

7. Attribuez un nouveau nom à la propriété ...JndiName du système de gestion de base de données relationnelle pour lequel vous optez.
8. Définissez à jdbc/twsdb la propriété ...JndiName du nouveau SGBD relationnel.
   - Vérifiez la définition des propriétés suivantes :
     – Pour DB2 :

       ```
       DB2Type4JndiName
       DB2Type4DatabaseName
       DB2Type4ServerName
       DB2Type4PortNumber
       ```

9. Exécutez le script ou la commande changeSecurityProperties.bat (.sh) pour modifier les paramètres de sécurité suivants :

   j2cUserid
   Indiquez la valeur que vous avez utilisée pour twsDbUser dans prepareSQLScripts.

   j2cPassword
   Indiquez le mot de passe pour twsDbUser.

   Remarque : Après avoir mis à jour ces deux valeurs, assurez-vous que vous effacez aussi toutes les autres lignes du fichier de propriétés de sécurité avant de l'exporter à nouveau à l'aide de la commande changeSecurityProperties. A défaut, tous les mots de passe figurant dans le fichier sont enregistrés en tant que chaînes d'astérisques (*).

Pour plus d'informations, voir «Modification des paramètres de sécurité», à la page 406.

10. Modifiez le fichier `TWSConfig.properties` situé dans le répertoire `<chemin_profil_WAS>/properties`, où *chemin_profil_WAS* correspond au chemin du profil WebSphere Application Server que vous avez spécifié au moment de l'installation. Le chemin par défaut est : *TWA_home*`/WAS/TWSprofile`. Commentez (marquez) les quatre lignes suivantes :

    ```
    com.ibm.tws.dao.rdbms.rdbmsName = Oracle
    com.ibm.tws.dao.rdbms.modelSchema = <twsDbUser>
    com.ibm.tws.dao.rdbms.eventRuleSchema
    com.ibm.tws.dao.rdbms.logSchema
    ```

    où `twsDbUser` est le propriétaire du schéma Oracle Tivoli Workload Scheduler.

11. Pour UNIX uniquement : exécutez la commande ou le script `updateSetupCmdLine.sh` pour définir les chemins d'accès de la nouvelle base de données. Recherchez ce script dans le répertoire `<`*TWA_home*`>/TWS/dbtools/db2/scripts`. La syntaxe est la suivante :

    ```
    updateSetupCmdLine.sh -installRoot <TWA_home> -dbRoot <DB_home>
    ```

    Où :

    –installRoot <*TWA_home*>
    Le répertoire d'installation Tivoli Workload Scheduler.

    –dbRoot <*racine_BD*>
    Le répertoire d'installation de la base de données.

12. Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)

13. Copiez manuellement la définition du .gestionnaire de domaine maître à partir du fichier `topology.def` que vous avez exporté à l'étape 1, à la page 343 (il se trouve dans le répertoire `export_dir`) et utilisez `composer new` pour l'ajouter à la base de données DB2.

14. Exécutez `dataimport` pour importer toutes les options globales et définitions d'objets de planification dans DB2. Recherchez ce fichier dans le sous-répertoire `bin` du répertoire de base Tivoli Workload Scheduler.

    Exécutez `dataimport` à partir d'une invite de commande Windows ou UNIX, comme suit :

    ```
    dataimport source_dir export_dir
    ```

    Où :

    source_dir
    Répertoire d'installation de Tivoli Workload Scheduler.

    export_dir
    Le répertoire à partir duquel les fichiers d'exportation sont lus. Ce répertoire est le même répertoire `export_dir` spécifié pour `dataexport`.

    Par exemple :

    ```
    dataimport.cmd F:\TWS91\tws91user F:\TWS91\tws91user\export
    ```

    Les définitions d'objet et les options globales sont récupérées dans le répertoire `F:\TWS91\tws91user\export` et stockées dans la nouvelle base de données DB2.

15. Exécutez la commande suivante pour définir l'option de réacheminement à ALL :

    ```
    optman chg cf=ALL
    ```

16. Mettez le fichier Symphony à jour en créant un plan avec une période d'extension de 0 qui commence à la fin du plan actuel :

    ```
    JnextPlan -from début -for 0000
    ```

    où *début* est la date et l'heure auxquelles le plan en cours se termine.

Vous avez maintenant terminé la procédure de reconfiguration.

Pour migrer un gestionnaire de domaine maître de sauvegarde, procédez comme suit :

- Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)
- Effectuez les étapes 6, à la page 347 à 11, à la page 348.
- Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)
- Définissez le paramètre `isBackupManager` de la commande `createdb_root` (script) sur `TRUE`.

Mise à niveau de votre base de données

Si vous voulez mettre à niveau votre base de données, modifiez le propriétaire de l'instance ou déplacez-le vers un autre hôte, la procédure de mise à niveau de votre base de données, en modifiant le propriétaire de l'instance, ou déplacez-la, comme suit :

1. Si vous modifiez DB2, vérifiez le *répertoire des noeuds* et le *répertoire de bases de données* et prenez note de la configuration en cours. Pour ce faire, émettez les commandes suivantes sur la ligne de commande DB2

   ```
   db2 list node directory show detail

   db2 list database directory
   ```

   où l'attribut `show detail` est indiqué pour donner des informations exhaustives dans le répertoire.

   Prenez note des détails qui s'affichent.

2. Arrêtez le serveur d'applications à l'aide de la commande

   ```
   stopWas -direct -user <user> -password <password>
   ```

3. Procédez à la mise à niveau, à la modification du propriétaire de l'instance ou au réadressage de la base de données en suivant les instructions de votre fournisseur de base de données.
4. Si vous avez modifié l'hôte de base de données, le port ou le nom de base de données, vous devez mettre à jour les propriétés de la source de données du serveur d'applications (voir «Modification du nom d'hôte, du port ou du nom d'une base de données», à la page 393).
5. Si vous avez modifié les autorisations d'accès à la base de données, vous devez mettre à jour les propriétés de sécurité du serveur d'applications (voir «Modification des paramètres de sécurité», à la page 406).
6. Reconfigurez la base de données pour Tivoli Workload Scheduler, comme suit :

   DB2

a. Vérifiez le *répertoire des noeuds* et le *répertoire de base de données*, comme vous l'avez fait à l'étape 1, à la page 349
b. Le cas échéant, modifiez les données affichées par ces commandes de sorte qu'elles correspondent à celles que vous avez notées à l'étape 1, à la page 349. Si vous n'êtes pas sûr quant à la manière de procéder, demandez de l'aide au service de support IBM.

Oracle Vérifiez le programme d'écoute Oracle et assurez-vous que le nom du service est correctement mentionné.

7. Redémarrez la base de données.
8. Redémarrez le serveur d'applications à l'aide de la commande suivante :

```
startWas -direct -user <utilisateur> -password <password>
```

Utilitaires d'audit

Décrit les utilitaires d'audit permettant de suivre les modifications dans les bases de données et le plan, ainsi que ceux qui suivent les modifications apportées aux objets participant à la planification dynamique de charge de travail.

Les analyses rétrospectives sont utiles pour vérifier l'application et l'efficacité des contrôles informatiques, ainsi que pour les analyses de responsabilité, de vulnérabilité et de risque. Les organisations informatiques peuvent aussi utiliser l'aufit des activités critiques liées à la sécurité pour faciliter les enquêtes consécutives aux incidents de sécurité. Lorsqu'un incident de sécurité se produit, les analyses rétrospectives permettent d'analyser l'historique des activités (qui a fait quoi, quand, où et comment) survenues avant l'incident de sécurité, de manière à pouvoir prendre les mesures correctives appropriées. C'est pourquoi les analyses rétrospectives doivent être archivées et rester accessibles pendant des années.

Les journaux d'audit sont créés au format XML et peuvent être affichés à l'aide d'un éditeur de texte standard ou analysés par des utilitaires tiers.

Vous pouvez aussi afficher les journaux d'audit en utilisant l'outil Log and Trace Analyzer (LTA), un composant d'IBM Autonomic Computing Toolkit. D'une manière générale, l'utilitaire Log and Trace Analyzer est utilisé pour importer et corréler les différents journaux générés par des produits différents. L'utilitaire Log and Trace Analyzer peut être très utile pour corréler les journaux d'audit avec d'autres journaux provenant de sources différentes, telles que des bases de données (DB2, Oracle), WebSphere Application Server et le système d'exploitation. Reportez-vous à la section Engine Log Analyzer dans *Tivoli Workload Scheduler : Guide d'identification et de résolution des problèmes* pour plus de détails.

Deux outils distincts d'analyse rétrospective sont fournis.

- Suivi des modifications de base de données et de plan - voir «Audit de base de données et de plan»
- Suivi des modifications aux objets de planification pour prendre en charge la planification dynamique de charge de travail - voir «Audit de la planification dynamique de charge de travail», à la page 357

Audit de base de données et de plan

Une option d'audit est disponible pour suivre les changements apportés à la base de données et au plan. Par défaut, elle est désactivée. Elle est décrite dans les sections suivantes :

- «Fonctionnement de l'audit», à la page 351

Fonctionnement de l'audit

Le stockage des enregistrements d'audit varie selon que vous gérez ou non des analyses rétrospectives pour la base de données ou le plan. Plusieurs options s'offrent à vous :

Audit de base de données
Vous pouvez suivre les modifications apportées à la base de données dans un fichier, dans la base de données elle-même ou dans les deux. Toutes les modifications de l'utilisateur sont consignées, y compris la définition actuelle de chaque objet de base de données modifié. Si un objet est ouvert et enregistré, l'action est consignée, même si aucune modification n'a été apportée.

Audit de plan
Vous pouvez suivre les modifications apportées au plan dans un fichier. Toutes les modifications apportées au plan par l'utilisateur sont consignées. Les actions sont consignées, qu'elles soient réussies ou non.

Chaque journal d'audit fournit des informations concernant une journée, de 00:00:00 UTC à 23:59:59 UTC, quel que soit le fuseau horaire du poste de travail utilisé, mais le fichier journal n'est créé que lorsqu'une action est exécutée ou que le WebSphere Application Server est démarré.

Les fichiers, appelés yyyymmdd, sont créés dans les répertoires suivants :

<*TWA_home*>/TWS/audit/plan

<*TWA_home*>/TWS/audit/database

Les entrées d'audit sont consignées dans un fichier texte à plat sur des postes de travail individuels dans le réseau Tivoli Workload Scheduler. Cela minimise le risque d'échec de l'audit dû à des problèmes réseau. D'une manière générale, les formats de journal sont les mêmes pour le plan et la base de données. Les journaux se composent d'une partie en-tête qui est la même pour tous les enregistrements, d'un ID d'action et d'une section de données qui varie en fonction du type d'action. Toutes les données sont conservées en texte clair et formatées pour être lisibles et modifiables à partir d'un éditeur de texte tel que **vi** ou **notepad**.

Remarque : Pour les commandes **modify**, deux entrées sont consignées dans le journal pour les ressources, les calendriers, les paramètres et les invites. La commande **modify** est affichée dans le journal en tant que combinaison des commandes **delete** et **add**.

Activation de la fonction d'audit

L'option d'audit est activée par la définition des deux entrées suivantes dans les options globales, à l'aide de **optman** :

```
enPlanAudit = 0|1
enDbAudit = 0|1
```

Une valeur de *1* (un) active l'audit et une valeur de *0* (zéro) désactive l'audit. Par défaut, l'audit est désactivé à l'installation du produit.

Pour initier l'audit de base de données, vous devez arrêter entièrement Tivoli Workload Scheduler. Lorsque vous redémarrez Tivoli Workload Scheduler, le journal d'audit de la base de données est lancé. L'audit de plan entre en vigueur lorsque JnextPlan est exécuté.

Format d'en-tête du journal d'audit

Chaque fichier journal commence par un enregistrement d'en-tête qui contient des informations relatives à la date et à l'heure de création du journal et précisent s'il s'agit d'un journal de plan ou de base de données.

Les zones d'en-tête d'enregistrement sont séparées par des barres verticales (|), comme suit :

```
EN-TÊTE|<date_GMT>|<heure_GMT>|<date_locale>|<heure_locale>|<type_objet>| >
<workstation>|<user_ID>|<version>| <level>
```

Type de journal
HEADER

Date GMT
La date GMT de création du fichier journal.

Heure GMT
L'heure GMT de création du fichier journal.

Date locale
La date locale de création du fichier journal. La date locale est définie par l'option de fuseau horaire du poste de travail.

Heure locale
L'heure locale de création du fichier journal. L'heure locale est définie par l'option de fuseau horaire du poste de travail.

Type d'objet
DATABASE pour un fichier journal de base de données et PLAN pour un fichier journal de plan.

Nom du poste de travail
Le nom du poste de travail Tivoli Workload Scheduler pour lequel ce fichier a été créé. Chaque poste de travail du réseau Tivoli Workload Scheduler crée son propre fichier journal.

ID utilisateur
ID utilisateur Tivoli Workload Scheduler qui a créé le fichier journal.

Version
La version du fichier.

Niveau
Le niveau de journalisation.

Format du corps de journal d'audit

Les formats de journaux d'audit sont généralement les mêmes pour le plan et la base de données. Le journal comprend un en-tête, un ID d'action et des sections de données qui varient selon le type d'action. Les données sont en texte en clair et les différents éléments de données sont séparés par des barres verticales (|).

Les entrées de fichier journal se présentent dans le format suivant :

```
<type_journal>|<date_GMT>|<heure_GMT>|<date_locale>|<heure_locale>|<type_objet>| >
<type_action>|<poste_travail>|<ID_utilisateur>|<nom_objet>|<zones_données_action>
```

Les fichiers journaux contiennent les informations suivantes :

type_journal
Affiche une valeur de huit caractères indiquant la source de l'enregistrement de journal. Les types de journaux suivants sont pris en charge :

CONMAN
texte de commande **conman**

DATABASE
Action de base de données

HEADER
L'en-tête du fichier journal

MAKESEC
exécution de **makesec**

PARMS
Texte de commande de paramètre

PLAN Action de plan

RELEASE
texte de commande d'**édition**

STAGEMAN
exécution de **stageman**

date_GMT
Affiche la date GMT à laquelle l'action a été réalisée. Le format est *yyyymmdd* où *yyyy* correspond à l'année, *mm* au mois et *dd* au jour.

Heure_GMT
Affiche l'heure GMT à laquelle l'action a été réalisée. Le format est *hhmmss* où *hh* correspond à l'heure, *mm* aux minutes et *ss* aux secondes.

date_locale
Affiche la date locale à laquelle l'action a été réalisée. La date locale est définie par l'option de fuseau horaire du poste de travail. Le format est *yyyymmdd* où *yyyy* correspond à l'année, *mm* au mois et *dd* au jour.

heure_locale
Affiche l'heure locale à laquelle l'action a été réalisée. L'heure locale est définie par l'option de fuseau horaire du poste de travail. Le format est *hhmmss* où *hh* correspond à l'heure, *mm* aux minutes et *ss* aux secondes.

type_objet
Affiche le type de l'objet affecté par une action, parmi les suivants :

DATABASE
Définition de base de données (pour l'en-tête seulement)

DBCAL
Définition du calendrier de base de données

DBDOMAIN
Définition du domaine de base de données

DBJBSTRM
Définition de base de données Planificateur de travaux

DBJOB
Définition du travail de base de données

DBPARM
Définition du paramètre de base de données

DBPROMPT
Définition de l'invite de base de données

DBRES
Définition de la ressource de base de données

DBSEC
Sécurité de base de données

DBUSER
Définition de l'utilisateur de la base de données

DBVARTAB
Définition de la table de variables de la base de données

DBWKCLS
Définition de la classe de poste de travail de base de données

DBWKSTN
Définition du poste de travail de base de données

PLAN Plan (pour l'en-tête seulement)

PLDOMAIN
Domaine du plan

PLFILE
Fichier de plan

PLJBSTRM
Plan Planificateur de travaux

PLJOB
Travail du plan

PLPROMPT
Invite du plan

PLRES
Ressource du plan

PLWKSTN
Poste de travail du plan

type_action
Affiche l'action réalisée sur l'objet. Les valeurs appropriées pour cette zone dépendent de l'action réalisée.

Pour le plan, le <action_type> peut être ADD, DELETE, MODIFY ou INSTALL.

Pour la base de données, les actions ADD, DELETE et MODIFY sont enregistrées pour le poste de travail, les classes de poste de travail, les domaines, les utilisateurs, les travaux, flots de travaux, les calendriers, les ressources et les paramètres dans la base de données.

La zone <action_type> enregistre également l'installation d'un nouveau fichier de sécurité. Lors de l'exécution de **makesec**, Tivoli Workload Scheduler l'enregistre en tant qu'action INSTALL pour un objet de définition de sécurité.

Les actions LIST et DISPLAY correspondant aux objets ne sont pas journalisées.

Pour les paramètres, la ligne de commande est journalisée avec ses arguments.

poste_de_travail
Affiche le poste de travail Tivoli Workload Scheduler à partir duquel l'utilisateur exécute l'action.

ID_utilisateur
Affiche l'utilisateur connecté qui a exécuté l'action en question. Sur les systèmes d'exploitation Windows, si l'utilisateur qui a installé WebSphere Application Server était un utilisateur du domaine, our les types de journal **stageman** et **conman**, cette zone contient l'ID utilisateur entier *domain\utilisateur*.

nom_objet
Affiche le nom qualifié complet de l'objet. Le format de cette zone dépend du type d'objet, comme illustré ici.

DATABASE
Indisponible

DBCAL
<*calendrier*>

DBDOMAIN
<*domaine*>

DBJBSTRM
<*poste_de_travail*>#<*flot_travaux*>

DBJOB
<*poste_de_travail*>#<*travail*>

DBPARM
<*poste_de_travail*>#<*paramètre*>

DBPROMPT
<*invite*>

DBRES
<*poste_de_travail*>#<*ressource*>

DBSEC
Indisponible

DBUSER
[<*poste_de_travail*>#]<*utilisateur*>

DBVARTAB
<*table_variables*>

DBWKCLS
<*classe_poste_travail*>

DBWKSTN
<*poste_de_travail*>

PLAN Indisponible

PLDOMAIN
<domaine>

PLFILE
<poste_de_travail>#*<poste_de_travail>*(*<qualificateur>*)

PLJBSTRM
<poste_de_travail>#*<instance_flot_travaux>*

PLJOB
<poste_de_travail>#*<instance_flot_travaux>*.*<travail>*

PLPROMPT
[*<poste_de_travail>*#]*<invite>*

PLRES
<poste_de_travail>#*<ressource>*

PLWKSTN
<poste_de_travail>

zones_données_action
Affiche les zones de données spécifiques à une action. Le format de ces données dépend de la zone <type_action>.

Exemples d'entrées de journal d'audit :
Voici un exemple de journal d'audit de base de données :

```
HEADER  |20080207|084124|20080207|094124|DATABASE|      |WK1|         | | |Version=A1.0| Level=1

DATABASE|20080207|084124|20080207|094124|DBRES   |ADD   |WK1|operator1| |res=WK1#RESOURCE    |

DATABASE|20080207|100524|20080207|110524|DBWKSTN |MODIFY|WK1|operator1| |ws=TIVOLI10         |

DATABASE|20080207|100525|20080207|110525|DBWKSTN |MODIFY|WK1|operator1| |ws=ASLUTRI1         |

DATABASE|20080207|100525|20080207|110525|DBWKSTN |MODIFY|WK1|operator1| |ws=WK1               |

DATABASE|20080207|100526|20080207|110526|DBDOMAIN|MODIFY|WK1|operator1| |dom=MASTERDM        |

DATABASE|20080207|100610|20080207|110610|DBWKSTN |MODIFY|WK1|operator1| |ws=TIVOLI10         |

DATABASE|20080207|100610|20080207|110610|DBWKSTN |MODIFY|WK1|operator1| |ws=ASLUTRI1         |

DATABASE|20080207|100611|20080207|110611|DBWKSTN |MODIFY|WK1|operator1| |ws=WK1               |

DATABASE|20080207|100611|20080207|110611|DBWKSTN |ADD   |WK1|operator1| |ws=WK2               |

DATABASE|20080207|100612|20080207|110612|DBDOMAIN|MODIFY|WK1|operator1| |dom=MASTERDM        |
```

Voici un exemple de journal d'audit de plan :

```
HEADER  |20080207|100758|20080207|110758|PLAN    |       |WK1|admin| |     |Version=A1.0|Level=1

STAGEMAN|20080207|100758|20080207|110758|PLAN    |INSTALL|WK1|admin| |C:\IBM\TWS\oper1\Symphony|
                   AWSBHV030I The new Symphony file is installed.

STAGEMAN|20080207|100758|20080207|110758|PLAN    |INSTALL|WK1|admin| |C:\IBM\TWS\oper1\Sinfonia|
                 AWSBHV036I Multi-workstation Symphony file copied to C:\IBM\TWS\oper1\Sinfonia
```

```
STAGEMAN|20080207|100758|20080207|110758|ADITLEVL|MODIFY |WK1|admin| |                          |
                    AWSBHV077I Audit level changing from 0 to 1.

CONMAN  |20080207|100800|20080207|110800|PLWKSTN |MODIFY |   |admin| |WK1                       |
                    continue & start

CONMAN  |20080207|100941|20080207|110941|PLWKSTN |MODIFY |   |admin| |SLUTRI1                   |
                    limit cpu=slutri1;10

PLAN    |20080207|101018|20080207|111018|PLWKSTN |MODIFY |WK1|oper1| |WK1                       |
                    limit cpu=SLUTRI1;20

PLAN    |20080207|101028|20080207|111028|PLDOMAIN|MODIFY |WK1|oper1| |ECCOLO                    |
                    reply ECCOLO;yes
```

Une commande **ResetPlan** exécutée sur le plan de production courant est stockée dans le fichier journal d'audit comme suit :

```
STAGEMAN|20080207|100758|20080207|110758|PLAN|DELETE|WK1|admin|
     |/home/WK1/schedlog/M200803140127|
                    AWSBHV025I The old Symphony file renamed /home/WK1/schedlog/M200803140127
```

Audit de la planification dynamique de charge de travail

Description

Lorsque vous sélectionnez la fonctionnalité de planification dynamique de charge de travail au moment de l'installation, la fonction d'audit est automatiquement installée. Par défaut, la fonction d'audit est désactivée.

Les événements pouvant faire l'objet d'un audit sont les suivants :

JobDefinitionAuditEvent
: Assure un suivi des opérations réalisées sur les définitions de travail

JobLogAuditEvent
: Assure un suivi des opérations réalisées sur les journaux de travail

JobAuditEvent
: Assure un suivi des opérations réalisées sur les travaux

ResourceAuditEvent
: Assure un suivi des opérations réalisées sur les ressources

RelationshipAuditEvent
: Assure un suivi des opérations réalisées sur les relations entre les ressources

RecoveryActionAuditEvent
: Assure un suivi des opérations réalisées sur les actions de reprise

HistoryDataAuditEvent
: Assure un suivi des opérations réalisées sur les données historiques.

Pour configurer l'audit des événements, activez la fonction d'audit et modifiez éventuellement les valeurs par défaut dans le fichier de configuration pour définir les types d'événement à auditer. Le fichier de configuration se trouve à l'emplacement suivant :

TWA_home\TDWB\config\audit.properties

Configuration de l'audit

Configurez une ou plusieurs des propriétés du fichier `audit.properties` pour activer et configurer l'audit :

audit.enabled
Spécifie si la fonction d'audit est activée ou désactivée. La valeur par défaut est false. Les valeurs prises en charge sont les suivantes :

false La fonction d'audit n'est pas activée.

true La fonction d'audit est activée.

onSecurityEnabled
La fonction d'audit est activée si la sécurité globale est activée sur le WebSphere Application Server.

audit.consumer.file.auditFilePrefix
Spécifie le préfixe du fichier journal d'audit. Le nom de fichier est défini à l'aide du préfixe du fichier, suivi du suffixe _audit*N*.log, où *N* est un nombre progressif. Si vous voulez que la date et l'heure de création du fichier soient spécifiées dans le préfixe du fichier, utilisez le format suivant : 'tdwb_'yyyy-MM-dd. Par exemple, l'utilisation du préfixe par défaut 'tdwb_'yyyy-MM-dd génère la famille de fichiers tdwb_2010-12-20_audit*N*.log. Notez que le texte entre apostrophes (') n'est pas traité par le programme et demeure inchangé. Ce format créer un fichier différent pour chaque jour au cours duquel la fonction d'audit est activée. De plus, le changement du préfixe en 'tdwb_'yyyy-MM génère la famille de fichiers tdwb_2010-12_audit*N*.log. Ce format crée un fichier différent pour chaque mois au cours duquel la fonction d'audit est activée.

Vous pouvez modifier ce format à votre convenance pour créer des fichiers sur une base hebdomadaire, mensuelle ou annuelle, selon vos exigences de contrôle. Selon le format d'heure et de date choisi, la taille et le nombre maximum de fichiers journaux varient. La taille et le nombre maximum de fichiers journaux sont définis à l'aide des propriétés **audit.consumer.file.maxFileSize** et **audit.consumer.file.maxAuditFiles** respectivement. Utilisez ces trois paramètres pour contrôler la taille des journaux d'audit stockés. Par exemple, si vous utilisez les valeurs par défaut de ces paramètres, vous aurez chaque jour au maximum 10 Mo x 100 fichiers. Une fois la taille maximale atteinte, le premier fichier créé est écrasé. Si vous voulez consacrer moins d'espace au stockage des journaux d'audit, vous pouvez choisir de modifier le nombre maximum de fichiers ou de ne générer des fichiers que chaque mois, en spécifiant le format de la propriété `audit.consumer.file.auditFilePrefix` comme 'tdwb_'yyyy-MM.

audit.consumer.file.auditFileLocation
Spécifie l'emplacement de création des fichiers journaux. L'emplacement par défaut est `/audit`.

audit.consumer.file.maxFileSize
Spécifie la taille maximale en octets des fichiers journaux. Lorsqu'un fichier atteint la taille maximale, un nouveau fichier journal est créé. La valeur par défaut est 10000000 octets (10 Mo). Il s'agit aussi de la valeur la plus élevée prise en charge.

audit.consumer.file.maxAuditFiles
Spécifie le nombre maximum de fichiers dotés d'un préfixe spécifique. Lorsque tous les fichiers atteignent la taille maximale et que le nombre

total de fichiers est dépassé, le fichier le plus ancien possédant un préfixe spécifique est écrasé. La valeur par défaut est 100 fichiers. Il s'agit aussi de la valeur la plus élevée prise en charge.

Configuration des événements d'audit dynamiques

La table suivante répertorie les actions et propriétés prises en charge pour chaque événement avec les valeurs par défaut liées. Vous pouvez configurer ces valeurs dans le fichier `audit.properties`.

Tableau 63. Propriétés d'événement pouvant faire l'objet d'un audit

Evénement	Action	Propriété	Valeur par défaut
JobDefinitionAuditEvent	create	audit.tdwb.JobDefinitionAuditEvent.create.enabled	true
	delete	audit.tdwb.JobDefinitionAuditEvent.delete.enabled	true
	get	audit.tdwb.JobDefinitionAuditEvent.get.enabled	true
	query	audit.tdwb.JobDefinitionAuditEvent.query.enabled	false
	update	audit.tdwb.JobDefinitionAuditEvent.update.enabled	true
JobLogAuditEvent	get	audit.tdwb.JobLogAuditEvent.get.enabled	true
JobAuditEvent	cancel	audit.tdwb.JobAuditEvent.cancel.enabled	true
	get	audit.tdwb.JobAuditEvent.get.enabled	true
	query	audit.tdwb.JobAuditEvent.query.enabled	false
	submit	audit.tdwb.JobAuditEvent.submit.enabled	true
ResourceAuditEvent	create	audit.tdwb.ResourceAuditEvent.create.enabled	true
	delete	audit.tdwb.ResourceAuditEvent.delete.enabled	true
	query	audit.tdwb.ResourceAuditEvent.query.enabled	false
	resume	audit.tdwb.ResourceAuditEvent.resume.enabled	true
	suspend	audit.tdwb.ResourceAuditEvent.suspend.enabled	true
	update	audit.tdwb.ResourceAuditEvent.update.enabled	true
RelationshipAuditEvent	create	audit.tdwb.RelationshipAuditEvent.create.enabled	true
	delete	audit.tdwb.RelationshipAuditEvent.delete.enabled	true
	query	audit.tdwb.RelationshipAuditEvent.query.enabled	false
RecoveryActionAuditEvent	invoke	audit.tdwb.RecoveryActionAuditEvent.invoke.enabled	true
HistoryDataAuditEvent	move	audit.tdwb.HistoryDataAuditEvent.move.enabled	true

Par défaut, l'audit est désactivé pour les actions de requête, alors que toutes les autres actions sont activées. Si la fonction d'audit est désactivée, toutes les propriétés sont ignorées.

Spécifications de fichier journal

Les éléments utilisés dans les fichiers journaux d'audit sont des extensions du schéma Common Base Event (CBE). Les types et les éléments répertoriés ci-dessous sont disponibles dans les fichiers journaux d'audit. Les types d'action pris en charge pour chaque élément sont répertoriés dans le tableau 63.

Action

Représente l'action entreprise. Chaque événement auditable prend en charge un ensemble différent d'actions possibles. Voir tableau 63. Le type d'action contient l'élément suivant :

Tableau 64. Eléments dans le type d'action

Nom de l'élément	Description de l'élément	Toujours retourné dans la sortie
Action	Le type d'action entrepris sur l'objet Dynamic Workload Broker.	Oui

ObjectInfoList

Représente une liste d'objets Dynamic Workload Broker. Le type `ObjectInfoList` contient les éléments suivants :

Tableau 65. Eléments dans le type ObjectInfoList

Nom de l'élément	Description de l'élément	Toujours retourné dans la sortie
objectInfo	La classe de l'objet participant à l'action	Oui

ObjectInfo

Représente les informations relatives à un objet Dynamic Workload Broker dans un type `objectInfoList` ou dans un autre élément `objectInfo`. Le type `ObjectInfo` contient les éléments suivants :

Tableau 66. Eléments dans le type ObjectInfo

Nom de l'élément	Description de l'élément	Toujours retourné dans la sortie
objectClass	La classe de l'objet participant à l'action.	Oui
objectName	Le nom de l'objet Dynamic Workload Broker.	Seulement s'il est disponible
objectNamespace	L'espace de nom de l'objet Dynamic Workload Broker.	Seulement s'il est disponible
objectType	Le type de l'objet Dynamic Workload Broker.	Seulement s'il est disponible
objectAlias	L'alias de l'objet Dynamic Workload Broker.	Seulement s'il est disponible
objectIdentifier	L'identificateur unique de l'objet Dynamic Workload Broker.	Seulement s'il est disponible
objectRole	Le rôle de l'objet Dynamic Workload Broker, le cas échéant. Par exemple, une ressource peut avoir un rôle de source ou de cible dans une relation.	Seulement s'il est disponible

Tableau 66. Eléments dans le type ObjectInfo (suite)

Nom de l'élément	Description de l'élément	Toujours retourné dans la sortie
objectSubmitterType	Le type du composant qui a soumis l'opération. Le composant est l'un des suivants : • Tivoli Dynamic Workload Broker Console • Ligne de commande • Dynamic workload brokerworkstation • Utilitaire tiers	Seulement s'il est disponible
objectInfo	Un objet `objectInfo` enfant. Par exemple, une relation est toujours liées à deux ressources.	Seulement s'il est disponible

Outcome

Définit le résultat d'un événement de sécurité. Le type Outcome contient les éléments suivants :

Tableau 67. Eléments dans le type Outcome

Nom de l'élément	Description de l'élément	Toujours retourné dans la sortie
result	Le statut de l'événement. Cette informations peut être utilisée lors du filtrage des informations dans le fichier journal.	Oui
failureReason	Informations supplémentaires sur le résultat de l'opération.	Oui, si l'opération a échoué.

UserInfoList

Représente une liste d'éléments `userInfo`, représentant chacun la liste des utilisateurs dans la chaîne de délégation. Le type `UserInfoList` contient les éléments suivants :

Tableau 68. Eléments dans le type UserInfoList

Nom de l'élément	Description de l'élément	Toujours retourné dans la sortie
objectInfo	Un ensemble d'informations relatives à chaque utilisateur dans la chaîne de délégation. Le premier élément `userInfo` identifie l'utilisateur qui s'est authentifié en premier. Le dernier élément `userInfo` identifie l'utilisateur dont les données d'identification sont utilisées par l'action entreprise.	Oui

UserInfo

Représente les informations relatives à un utilisateur. Les éléments de ce type retournent ds informations concernant l'utilisateur participant à l'opération en cours d'audit. Le type `UserInfo` contient les éléments suivants :

Tableau 69. Eléments dans le type UserInfo

Nom de l'élément	Description de l'élément	Toujours retourné dans la sortie
UserInfo	Le nom d'utilisateur fourni à Dynamic Workload Broker pour l'authentification.	Oui

Comment exécuter des requêtes sur des fichiers journaux

Les fichiers journaux peuvent être très longs et détaillés. Lorsque vous affichez vos fichiers journaux à l'aide de Log and Trace Analyzer, vous pouvez appliquer une ou plusieurs requêtes pour filtrer les informations dans le fichier et accélérer les recherches. Vous pouvez utiliser les requêtes suivantes pour filtrer seulement les informations pertinentes ou vous pouvez créer vos propres requêtes en fonction de vos exigences. Les requêtes suivantes sont rédigées dans le langage de requête XPath.

- Pour filtrer tous les événements générés par un utilisateur spécifique :

  /CommonBaseEvent [extendedDataElements/children[@name='*userInfo*' and values='*username*']]
- Pour filtrer tous les événements liés à une classe d'objets spécifique :

  /CommonBaseEvent [extendedDataElements//children[@name='objectClass' and values='Resource]]
- Pour filtrer tous les événements liés à un objet spécifique :

  //CommonBaseEvent [extendedDataElements//children[@name='objectName' and values='myresource']/../children[@name='objectClass' and values='Resource']]
- Pour filtrer tous les événements liés à une action spécifique :

  /CommonBaseEvent [extendedDataElements[@name='action' and values='uninstall']]
- Pour filtrer tous les événements avec résultat SUCCESSFUL :

  /CommonBaseEvent [extendedDataElements/children[@name='result' and values='SUCCESSFUL']]

La requête suivante retourne toutes les actions de création :

```
/CommonBaseEvent[ extendedDataElements[@name = 'action' and values = 'create']]
```

Vous pouvez exporter cette requête dans un fichier XML comme suit :

```
<?xml version="1.0" encoding="UTF-8"?><cbeviewer_configuration>
<logParserSets>
   <logParserSet  description="Parser for CBE log"
                  id="com.ibm.cbeviewer.parsers.cbeLogParserSet"
                  label="Common Base Event log"
                  parentId="com.ibm.cbeviewer.parsers.jdLogParserSet"/>
   <logParserSet  description="Parser for CEI Server"
                  id="com.ibm.cbeviewer.parsers.ceiLogParserSet"
                  label="Common Event Infrastructure server"
                  parentId="com.ibm.cbeviewer.parsers.jdLogParserSet"/>
   <logParserSet description="Other parsers"
                  id="com.ibm.cbeviewer.parsers.otherParsersLogParserSet"
                  label="Other parsers"/>
```

```
</logParserSets>
<recent_expressions>
     <xpath name="All Create Events">
     /CommonBaseEvent[ extendedDataElements[@name = 'action' and values = 'create']]
     </xpath>
</recent_expressions></cbeviewer_configuration>
```

Voici un bref exemple de fichier journal :

```
<CommonBaseEvent
      creationTime="2007-06-06T14:26:23.311Z"
      extensionName="TDWB_JOB_AUDIT_EVENT"
      globalInstanceId="CEFC6DD156CA54D902A1DC1439E6EC4ED0"
      sequenceNumber="1"
      version="1.0.1">
   <extendedDataElements
         name="userInfoList"
         type="noValue">
      <children
            name="userInfo"
            type="string">
         <values>UNAUTHENTICATED</values>
      </children>
   </extendedDataElements>
   <extendedDataElements
         name="action"
         type="string">
      <values>submit</values>
   </extendedDataElements>
   <extendedDataElements
         name="outcome"
         type="noValue">
      <children
            name="result"
            type="string">
         <values>SUCCESSFUL</values>
      </children>
   </extendedDataElements>
</CommonBaseEvent>
```

Exemples

Les exemples suivants décrivent une utilisation standard de la fonction d'audit.

Dans l'exemple suivant, l'utilisateur `root` récupère avec succès la définition d'un travail nommé **MyTestJob** à l'aide de la commande `jobstore`.

```
<CommonBaseEvent
      creationTime="2007-06-21T16:05:19.455Z"
      extensionName="TDWB_JOB_AUDIT_EVENT"
      globalInstanceId="CE8F5E102AE3419AF7A1DC201135463A40"
      sequenceNumber="188"
      version="1.0.1">
   <extendedDataElements
         name="userInfoList"
         type="noValue">
      <children
            name="userInfo"
            type="string">
         <values>root</values>
      </children>
   </extendedDataElements>
   <extendedDataElements
         name="action"
         type="string">
      <values>get</values>
   </extendedDataElements>
   <extendedDataElements
         name="outcome"
```

```
      type="noValue">
   <children
         name="result"
         type="string">
      <values>SUCCESSFUL</values>
   </children>
</extendedDataElements>
<extendedDataElements
      name="objectInfoList"
      type="noValue">
   <children
         name="objectInfo"
         type="noValue">
      <children
            name="objectClass"
            type="string">
         <values>Job</values>
      </children>
      <children
            name="objectName"
            type="string">
               <values>MyTestJob</values>
      </children>
      <children
            name="objectIdentifier"
            type="string">
         <values>3ebf6d62-0b83-3270-9b83-83c393e9cbca</values>
      </children>
      <children
            name="objectSubmitterType"
            type="string">
         <values>TDWB CLI</values>
      </children>
   </children>
</extendedDataElements>
<extendedDataElements
      name="CommonBaseEventLogRecord:sequenceNumber"
      type="long">
    <values>80808</values>
</extendedDataElements>
<extendedDataElements
      name="CommonBaseEventLogRecord:threadID"
      type="int">
   <values>280</values>
</extendedDataElements>
<sourceComponentId
      application="JobManagement"
      component="None"
      componentIdType="Application"
      location="tdws08"
      locationType="Hostname"
      subComponent="None"
      threadId="Default : 84"
      componentType="http://www.ibm.com/namespace/autonomic/Tivoli_componentTypes"/>
<situation
      categoryName="ReportSituation">
   <situationType
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:type="ReportSituation"
         reasoningScope="INTERNAL"
         reportCategory="SECURITY"/>
</situation>
</CommonBaseEvent>
```

Dans l'exemple suivant, l'utilisateur `testuser` tente de supprimer une instance de travail nommée **MySecondJob** à l'aide de la ligne de commande appropriée. L'opération échoue car le travail a été soumis par un autre utilisateur. La suppression de travaux soumis par d'autres utilisateurs requiert les droits `Opérateur` ou `Administrateur`. Pour plus d'informations sur les droits d'accès, voir

IBM Tivoli Workload Scheduler - Planification dynamique de la charge de travail ou *IBM Tivoli Workload Scheduler - Guide d'administration*.

```
<CommonBaseEvent
      creationTime="2007-06-21T16:05:32.746Z"
      extensionName="TDWB_JOB_AUDIT_EVENT"
      globalInstanceId="CE8F5E102AE3419AF7A1DC20113D32BB20"
      sequenceNumber="189"
      version="1.0.1">
   <extendedDataElements
         name="userInfoList"
         type="noValue">
      <children
            name="userInfo"
            type="string">
         <values>testuser</values>
      </children>
   </extendedDataElements>
   <extendedDataElements
         name="action"
         type="string">
      <values>cancel</values>
   </extendedDataElements>
   <extendedDataElements
         name="outcome"
         type="noValue">
      <children
            name="result"
            type="string">
         <values>UNSUCCESSFUL</values>
      </children>
      <children
            name="failureReason"
            type="string">
         <values>userNotAuthorized</values>
      </children>
   </extendedDataElements>
   <extendedDataElements
         name="objectInfoList"
         type="noValue">
      <children
            name="objectInfo"
            type="noValue">
         <children
               name="objectClass"
               type="string">
            <values>Job</values>
         </children>
         <children
               name="objectName"
               type="string">
            <values>MySecondJob</values>
         </children>
         <children
               name="objectIdentifier"
               type="string">
            <values>a05732c8-c008-3103-afd1-84b567d78de7</values>
         </children>
         <children
               name="objectSubmitterType"
               type="string">
            <values>TDWB CLI</values>
         </children>
      </children>
   </extendedDataElements>
   <extendedDataElements
         name="CommonBaseEventLogRecord:sequenceNumber"
         type="long">
      <values>80964</values>
   </extendedDataElements>
   <extendedDataElements
```

```
        name="CommonBaseEventLogRecord:threadID"
        type="int">
      <values>292</values>
   </extendedDataElements>
   <sourceComponentId
        application="JobManagement"
        component="None"
        componentIdType="Application"
        location="tdws08"
        locationType="Hostname"
        subComponent="None"
        threadId="Default : 91"
        componentType="http://www.ibm.com/namespace/autonomic/Tivoli_componentTypes"/>
   <situation
        categoryName="ReportSituation">
      <situationType
           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
           xsi:type="ReportSituation"
           reasoningScope="INTERNAL"
           reportCategory="SECURITY"/>
   </situation>
</CommonBaseEvent>
```

Suivi des modifications de base de données à l'aide des rapports d'audit

Pour assurer le suivi en continu des modifications ayant une incidence sur les objets stockés dans la base de données, vous pouvez utiliser les rapports d'audit suivants, pouvant être exécutés en mode batch à l'aide de l'interface de ligne de commande :

Rapport d'audit général
Ce rapport fournit des informations sur les objets ayant été modifiés dans la base de données. Plus spécifiquement, il détaille l'auteur des modifications, les objets concernés et de quand datent les modifications.

Rapport Détails
Ce rapport fournit plus de détails sur les modifications mises en oeuvre. Il indique l'auteur des modifications, les objets concernés, quand ont été apportées les modifications et ce qui a changé. Plus spécifiquement, il présente la définition des objets avant et après leur modification.

Exemple de scénario métier

L'administrateur d'une compagnie d'assurance a besoin de suivre toutes les modifications ayant une incidence sur les contrats d'assurance, les termes et conditions applicables à tous les clients enregistrés dans la base de données de la compagnie. Pour ce faire, l'administrateur exécute à intervalles réguliers les rapports d'audit généraux et détaillés.

Pour satisfaire cette demande, il crée un rapport général d'audit qui fournit des détails sur les objets TWS ayant été modifiés dans la base de données, l'auteur et la date des modifications. Puis, pour obtenir plus de détails sur les modifications apportées, il crée également un rapport des détails d'audit.

Pour accomplir cette tâche, il procède comme suit :

1. Il personnalise les fichiers de propriétés associés aux rapports d'audit, indiquant le format et le contenu de la sortie du rapport.
2. Il planifie les travaux afin d'obtenir les rapports :
   a. Le premier travail génère un audit à sauvegarder en local.
   b. Le second travail exécute un rapport détaillé au cours de la nuit pour extraire davantage de détails sur les modifications spécifiques mises en

oeuvre. La sortie du rapport est envoyée par e-mail à l'analyste. Les informations collectées sont utilisées pour tenir au courant toutes les succursales de la compagnie de toutes les modifications et des nouveautés.

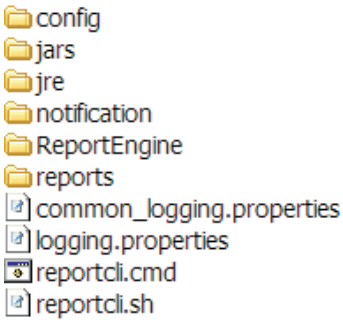
3. L'administrateur ajoute ces deux travaux dans un flot de travaux dont l'exécution est planifiée toutes les semaines et qui génère le plan.

Configuration de la génération de rapports de la ligne de commande

Avant d'exécuter ces rapports, vous devez effectuer quelques étapes de configuration :

1. Le logiciel nécessaire pour exécuter ces rapports se trouve dans un package nommé `TWSBatchReportCli` inclus dans l'image d'installation Tivoli Workload Scheduler, dans le répertoire `TWSBatchReportCli`. Si vous avez l'intention de les exécuter dans un travail planifié, extrayez le fichier de package sur l'un des systèmes d'exploitation listés dans le Document relatif à la configuration requise à l'adresse http://www-01.ibm.com/support/docview.wss?rs=672&uid=swg24036758.

   Après avoir extrait le package, vous obtenez la structure de fichier suivante :

   Dans la mesure où l'utilitaire natif UNIX tar ne prend pas en charge les noms de fichier longs, si vous extrayez les fichiers sur des systèmes AIX, Solaris ou HP-UX, assurez-vous que la toute dernière version GNU de tar (gtar) est installée pour extraire les fichiers avec succès.

   Remarque :

   a. Vérifiez bien que vous exécutez les commandes suivantes dans le répertoire d'extraction des fichiers :

      Sur UNIX

      ```
      chmod -R +x *
      chown -R username *
      ```

      Sur Windows

      Vérifiez que Tivoli Workload Scheduler est installé.

      ```
      setown -u username *
      ```

      Où *username* est l'utilisateur Tivoli Workload Scheduler qui exécutera les rapports.

   b. Si vous envisagez de planifier les travaux pour exécuter ces rapports, le système sur lequel vous avez extrait le package doit être accessible en tant que système de fichier réseau à partir d'un agent tolérant aux pannes défini dans l'environnement de planification local.

2. Téléchargez les pilotes JDBC demandés par votre version du serveur Oracle.

3. Copiez les pilotes JDBC dans les répertoires *report_cli_installation_dir*\jars et *report_cli_installation_dir*\ReportEngine\plugins\org.eclipse.birt.report.data.oda.jdbc_4.2.1.v20120820\drivers. Le rapport de la ligne de commande découvre automatiquement les deux fichiers JAR.
4. Configurez le fichier modèle .\config\common.properties en indiquant les informations nécessaires pour :
   a. Se connecter à la base de données dans laquelle sont stockées les données d'historique.
   b. Définir le format de date et d'heure, y compris le fuseau horaire. Le fichier .\config\timezone.txt contient une liste de fuseaux horaires pris en charge par Tivoli Workload Scheduler et des informations sur leur mode de configuration. Les noms de fuseaux horaires sont sensibles à la casse.
   c. Diffusez la sortie du rapport sur l'URL indiquée dans la zone **ContextRootUrl**. Voici un exemple de paramètres de configuration :

```
#####################################################################
# HTTP Server information
#####################################################################

 #Specify the context root where the report will be available
 #To leverage this possibility it needs to specify in the report output dir
 #the directory that is referred by your HTTP Server with this context root

 ContextRootUrl=http://myserver/reportoutput
```

   Dans ce cas vérifiez que le *répertoire_rapport_sortie* indiqué lors de l'exécution de la commande reports pointe vers le même répertoire indiqué dans la zone **ContextRootUrl**.
   d. Envoyez la sortie du rapport par courriel. Voici un exemple de paramètres de configuration :

```
#####################################################################
# Email Server configuration
#####################################################################
 PARAM_SendReportByEmail=true

 #SMTP server
 mail.smtp.host=myhost.mydomain.com
 #IMAP provider
 mail.imap.socketFactory.fallback=false
 mail.imap.port=993
 mail.imap.socketFactory.port=993
 #POP3 provider
 mail.pop3.socketFactory.fallback=false
 mail.pop3.port=995
 mail.pop3.socketFactory.port=995

 #####################################################################
 # Email properties
 #####################################################################
 PARAM_EmailFrom=user1@your_company.com
 PARAM_EmailTo=user2@your_company.com,user3@your_company.com
 PARAM_EmailCC=user4@your_company.com
 PARAM_EmailBCC=user5@your_company.com
 PARAM_EmailSubject=Test d'envoi de rapport par e-mail
 PARAM_EmailBody=Voici le rapport en pièce jointe
```

   Une explication de toutes les zones personnalisables est incluse dans le fichier modèle.

Exécution de rapports d'audit à partir de la ligne de commande

Pour exécuter un rapport d'audit sur la base de données, vous devez d'abord activer la fonction d'audit et configurer les options d'audit décrites à la section «Options globales - Description détaillée», à la page 14.

Le répertoire `\reports\templates` contient un exemple de fichier modèle pour chaque type de rapport.

Avant d'exécuter l'un de ces rapports, veillez à personnaliser le fichier modèle correspondant, soit `ad.properties`, soit`ag.properties`.

Dans ce fichier, nommé *`nom_rapport`*`.properties`, vous pouvez spécifier :

- Les informations à afficher dans l'en-tête du rapport.
- Comment filtrer les informations pour afficher le résultat escompté.
- Le format et le contenu de la sortie du rapport.

Pour plus d'informations sur les paramètres spécifiques, voir l'explication fournie dans le fichier modèle en regard de chaque zone.

Après avoir configuré l'environnement comme indiqué dans «Configuration de la génération de rapports de la ligne de commande», à la page 367, ainsi que le fichier modèle, utilisez la syntaxe suivante pour exécuter le rapport :

reportcli -p *report_name.property*
[**-o** *rép_rapport_sortie*]
[**-r** *nom_sortie_rapport*]
[**-k** key=*value*]
[**-k** key=*value*]
.....

Où :

-p *report_name.property*
Indique le chemin d'accès au fichier modèle du rapport.

-o *rép_rapport_sortie*
Indique le répertoire de sortie des résultats du rapport.

-r *nom_sortie_rapport*
Indique le nom de la sortie du rapport.

-k key=*value*
Indique la valeur d'un paramètre. Cette valeur remplace la valeur correspondante, si elle est définie, dans le fichier `common.properties` ou dans le fichier *`nom_rapport`*`.properties`.

Exemples :

1. Dans cet exemple, la commande `reportcli.cmd` est exécutée avec le paramètre par défaut :

   ```
   reportcli.cmd -p D:\ReportCLI\TWSReportCli\reports\templates\ag.properties
   -r audit1
   ```

2. Dans cet exemple, la commande `reportcli.cmd` est exécutée à l'aide du paramètre `-k` pour remplacer les valeurs définies pour **PARAM_DateFormat** dans le fichier `.\config\common.properties` :

   ```
   reportcli.cmd -p D:\ReportCLI\TWSReportCli\reports\templates\ag.properties
   -r audit2 -k PARAM_DateFormat=short
   ```

3. Dans cet exemple, la commande `reportcli.cmd` est exécutée à l'aide du paramètre `-k` pour remplacer le format indiqué pour la sortie du rapport dans le fichier `.properties` :

```
./reportcli.sh -p /TWSReportCli/REPCLI/reports/templates/ag.properties
-r audit3 -k REPORT_OUTPUT_FORMAT=html -k OutputView=charts
```

Remarque : Si le rapport est exécuté via un travail Tivoli Workload Scheduler, le résultat de la commande s'affiche dans la sortie du rapport.

Chapitre 9. Tâches d'administration

Le présent chapitre explique comment réaliser certaines tâches d'administration spécifiques sur Tivoli Workload Scheduler, comme suit :

Les tâches

«Modification d'un gestionnaire de domaine ou d'un gestionnaire de domaine dynamique», à la page 373
Modifiez un gestionnaire de domaine ou un gestionnaire de domaine dynamique, soit en cas de défaillance de l'ordinateur sur lequel il est installé, soit dans le cadre d'une activité de remplacement planifiée.

«Modification d'un gestionnaire de domaine maître», à la page 378
Modifiez un gestionnaire de domaine maître, soit en cas de défaillance de l'ordinateur sur lequel il est installé, soit dans le cadre d'une activité de remplacement planifiée.

«Modification des mots de passe Tivoli Workload Scheduler clés», à la page 382
Modifiez le mot de passe du utilisateur_TWS, ou de tout autre utilisateur disposant d'un rôle d'infrastructure dans Tivoli Workload Scheduler.

«Suppression des liaisons et arrêt de Tivoli Workload Scheduler», à la page 392
La procédure correcte pour délier le gestionnaire de domaine maître de ses agents et arrêter le traitement du maître.

«Modification du nom d'hôte, du port ou du nom d'une base de données», à la page 393
Si vous devez modifier l'hôte, le port ou le nom de la base de données, effectuez ce changement dans le serveur d'applications, où la configuration de la source de données est entretenue.

«Modification du nom d'hôte ou de l'adresse IP du poste de travail», à la page 400
Modifiez le nom d'hôte ou l'adresse IP d'un poste de travail.

«Modification des paramètres de sécurité», à la page 406
Si vous devez mettre à jour les propriétés qui définissent votre connexion SSL ou votre mécanisme d'authentification, vous devez effectuer ces modifications dans la WebSphere Application Server intégré

«Gestion du processeur d'événements», à la page 407
Si vous utilisez l'automatisation de la charge de travail gérée par les événements, vous devrez procéder à la maintenance périodique du processeur d'événement.

«Démarrage, arrêt et affichage du statut de Dynamic Workload Broker», à la page 407
Procédure de démarrage ou d'arrêt de Dynamic Workload Broker.

Tâches du serveur d'applications
Les tâches suivantes devront peut-être être réalisées sur le serveur d'applications :

Modification des propriétés du serveur d'applications

Plusieurs des tâches ci-dessus exigent d'effectuer une procédure commune grâce à laquelle vous pouvez :

1. Exécuter un utilitaire qui affiche un ensemble des propriétés en cours du serveur d'applications et les enregistre dans un fichier
2. Editer le fichier pour modifier les propriétés
3. Exécuter une autre procédure pour mettre à jour le serveur d'applications avec les propriétés modifiées

Cette procédure est décrite en détail à la section «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423

Informations de référence sur les utilitaires du serveur d'applications

Des informations de référence sur les utilitaires du serveur d'applications sont également fournies dans «Utilitaires du serveur d'applications», à la page 424. Pour plus d'informations, voir *IBM Redbooks : WebSphere Application Server V6 - Manuel de gestion et de configuration du système*.

Modification d'un gestionnaire de domaine ou d'un gestionnaire de domaine dynamique

Il vous faudra peut-être modifier un gestionnaire de domaine ou un gestionnaire de domaine dynamique pour pouvoir l'exécuter sur un autre poste de travail ou vous serez forcé de le faire en raison de problèmes de liaison au niveau du réseau ou de l'échec du poste de travail du gestionnaire de domaine ou du gestionnaire de domaine dynamique. Cette section et ses sous-sections, expliquent comment préparer et utiliser un gestionnaire de domaine ou un gestionnaire de domaine dynamique de secours. Cependant, si le gestionnaire de domaine à modifier est un gestionnaire de domaine maître, certaines procédures complémentaires spécifiques doivent être réalisées (voir «Modification d'un gestionnaire de domaine maître», à la page 378).

Une exécution sans gestionnaire de domaine produit les effets suivants :

- Les agents et les gestionnaire de domaine subordonnés ne peuvent pas résoudre les dépendances entre postes de travail car les enregistrements d'activité diffusés par le gestionnaire de domaine maître ne sont pas reçus.
- Le flux ascendant d'événements est interrompu, Cela affecte les événements qui rapportent le statut des travaux, le flots de travaux et les dépendances définies sur les postes de travail dans la hiérarchie du réseau Tivoli Workload Scheduler sous le gestionnaire de domaine ayant échoué.
- Les agents standard hébergés par le gestionnaire de domaine ayant échoué ne peuvent procéder à aucun traitement puisqu'ils dépendent du gestionnaire de domaine pour tout ce qui concerne la planification et le lancement de travaux.

Si le problème ne doit pas durer longtemps, vous pouvez attendre qu'il soit résolu et Tivoli Workload Scheduler se rétablira par lui-même, comme décrit dans *Tivoli Workload Scheduler - Guide d'identification des problèmes* à la section consacrée aux problèmes de liaison. Si vous ne savez pas combien de temps il va durer, ou si vous voulez restaurer le fonctionnement normal de l'agent, vous devez passer à un dispositif de secours, comme indiqué dans les sections qui suivent.

Choix d'un gestionnaire de domaine ou d'un gestionnaire de domaine dynamique de secours

La reprise sera d'autant plus facile que vous serez préparé aux incidents de réseau. Définissez un gestionnaire de domaine de secours ou un gestionnaire de domaine

dynamique de secours pour chaque gestionnaire de domaine ou gestionnaire de domaine dynamique respectif de votre réseau, pour être sûr de pouvoir répondre aux pics de charges de planification de travaux de Tivoli Workload Scheduler. Sélectionnez comme gestionnaire de domaine ou gestionnaire de domaine dynamique de secours n'importe quel agent tolérant aux pannes du domaine.

Définition d'un gestionnaire de domaine de sauvegarde

Vérifiez que le mode *FullStatus* est sélectionné dans la définition du poste de travail du gestionnaire de domaine ou du gestionnaire de domaine dynamique de secours.

Assurez-vous également que l'heure du gestionnaire de domaine de secours est synchronisée avec celle du gestionnaire de domaine. Le moyen le plus sûr consiste à utiliser un serveur Network Time Protocol Server pour contrôler l'heure sur les deux systèmes, avec le même intervalle entre les répétitions.

Sécurité réseau

La sécurité réseau est assurée à l'aide de la validation de l'adresse IP. Par conséquent, la connexion des postes de travail (option *autolink* ou **link**) risque d'échouer si un agent a un ancien fichier `Symphony` qui ne contient pas le nouveau gestionnaire de domaine. En cas d'échec d'une connexion, supprimez l'ancien fichier `Symphony` de l'agent et tentez à nouveau de procéder à la connexion.

Changement d'un gestionnaire de domaine

Utilisez l'une des procédures suivantes en cas de perte momentanée d'un gestionnaire de domaine.

Utilisation de la ligne de commande

Reportez-vous à la procédure décrite sous la commande **switchmgr** dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Utilisation du Dynamic Workload Console

1. Lancez Dynamic Workload Console
2. Connectez-vous au moteur du gestionnaire de domaine en cours
3. A partir de la barre de navigation, cliquez sur **Statut et état de santé du système** > **Surveillance de l'environnement** > **Surveillance des postes de travail**.
4. Sélectionnez une tâche pour surveiller les postes de travail.
5. Sélectionnez le poste de travail que vous voulez utiliser comme gestionnaire de domaine
6. A partir de la table contenant la liste des postes de travail, sélectionnez un poste de travail et cliquez sur **Plus d'actions** > **Devenir gestionnaire de domaine maître**.
7. Cliquez sur **OK**.

Les gestionnaires de domaine restent commutés jusqu'à une nouvelle opération de changement de gestionnaire de domaine ou jusqu'à l'exécution de **JnextPlan**. Pour revenir au gestionnaire de domaine d'origine sans exécuter **JnextPlan**, répétez cette procédure.

Procédure complète de basculement d'un gestionnaire de domaine

Cette section récapitule les étapes requises pour remplacer un gestionnaire de domaine en cours d'exécution par sa sauvegarde et pour achever la procédure par la restauration du gestionnaire de domaine d'origine dans sa fonction. Procédez comme suit pour vous assurer qu'aucun problème de chevauchement ne survienne avec des versions obsolètes du fichier Symphony. Les étapes sont documentées pour quatre scénarios :

Indisponibilité planifiée
Le gestionnaire de domaine est remplacé par sa sauvegarde pour des travaux de maintenance planifiés (par exemple, une mise à niveau du système d'exploitation).

Indisponibilité non planifiée
Le gestionnaire de domaine est remplacé par sa sauvegarde en raison d'un dysfonctionnement ou d'une défaillance inattendue.

Court terme
Le gestionnaire de domaine est prévu pour être remis en service avant la prochaine période de nouvelle production (exécution du travail JnextPlan).

Long terme
Il n'est pas prévu de remettre le gestionnaire de domaine en service avant la prochaine période de nouvelle production (exécution du travail JnextPlan).

Tableau 70. Procédure complète de basculement d'un gestionnaire de domaine en cas d'indisponibilité planifiée

Indisponibilité planifiée	
Court terme	**Long terme**
1. Basculez le gestionnaire de domaine vers le poste de travail de sauvegarde. Utilisez la commande `conman switchmgr` ou Dynamic Workload Console. Pour plus d'informations, voir la commande `switchmgr` dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence*. Vérifiez que les boîtes de message pour le gestionnaire de domaine faisant l'objet de la maintenance sont suffisamment grandes pour ne pas être remplies avant sa restauration. Augmentez leur taille si nécessaire.	1. Basculez le gestionnaire de domaine vers le poste de travail de sauvegarde. Utilisez la commande `conman switchmgr` ou Dynamic Workload Console. Pour plus d'informations, voir la commande `switchmgr` dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence*. Vérifiez que les boîtes de message pour le gestionnaire de domaine faisant l'objet de la maintenance sont suffisamment grandes pour ne pas être remplies avant sa restauration. Augmentez leur taille si nécessaire.
2. Arrêtez le traitement de Tivoli Workload Scheduler sur le gestionnaire de domaine faisant l'objet de la maintenance.	2. Arrêtez le traitement de Tivoli Workload Scheduler sur le gestionnaire de domaine d'origine faisant l'objet de la maintenance.
3. Dans la base de données de Tivoli Workload Scheduler, attribuez le rôle de gestionnaire de domaine au poste de travail de sauvegarde.	3. Dans la base de données de Tivoli Workload Scheduler, attribuez le rôle de gestionnaire de domaine au poste de travail de sauvegarde.
	4. Définissez le poste de travail qui exécute le gestionnaire de domaine d'origine à `ignore`, à l'aide de la commande `conman cpuname` ou de la console Dynamic Workload Console.

Tableau 70. Procédure complète de basculement d'un gestionnaire de domaine en cas d'indisponibilité planifiée (suite)

Indisponibilité planifiée	
Lorsque vous êtes prêt à restaurer l'appartenance du domaine au gestionnaire de domaine d'origine :	Lorsque vous êtes prêt à restaurer l'appartenance du domaine au gestionnaire de domaine d'origine :
4. Réaffectez l'appartenance du domaine au gestionnaire de domaine d'origine dans la base de données Tivoli Workload Scheduler.	5. Supprimez l'indicateur `ignore` du poste de travail qui exécute le gestionnaire de domaine d'origine.
5. Basculez le poste de travail de sauvegarde vers le gestionnaire de domaine en utilisant l'une des méthodes mentionnées à l'étape 1.	6. Réaffectez l'appartenance du domaine au gestionnaire de domaine d'origine dans la base de données Tivoli Workload Scheduler.
6. Liez le gestionnaire de domaine à partir du maître pour télécharger une version actualisée du fichier Symphony.	Eventuellement, retirez du gestionnaire de domaine d'origine, la commande `conman start` de la procédure d'initialisation et supprimez les copies existantes des fichiers Symphony, Sinfonia et de boîte de message. Nous recommandons d'exécuter cette étape pour éviter que des éléments symphony obsolètes présents dans l'ordinateur ne se déclenchent automatiquement au premier démarrage. Vous pourrez rajouter la commande `conman start` ultérieurement.
	7. Basculez le poste de travail de sauvegarde vers le gestionnaire de domaine en utilisant l'une des méthodes mentionnées à l'étape 1.
	8. Liez le gestionnaire de domaine à partir du maître pour télécharger une version actualisée du fichier Symphony.

Tableau 71. Procédure complète de basculement d'un gestionnaire de domaine après une indisponibilité non planifiée.

Indisponibilité non planifiée	
Court terme	**Long terme**
1. Basculez le gestionnaire de domaine vers le poste de travail de sauvegarde. Utilisez la commande `conman switchmgr` ou Dynamic Workload Console. Pour plus d'informations, voir la commande `switchmgr` dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.	1. Basculez le gestionnaire de domaine vers le poste de travail de sauvegarde. Utilisez la commande `conman switchmgr` ou Dynamic Workload Console. Pour plus d'informations, voir la commande `switchmgr` dans *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.
Vérifiez que les boîtes de message du gestionnaire de domaine en défaut sont suffisamment grandes pour ne pas être remplies avant sa restauration. Augmentez leur taille si nécessaire.	Vérifiez que les boîtes de message du gestionnaire de domaine en défaut sont suffisamment grandes pour ne pas être remplies avant sa restauration. Augmentez leur taille si nécessaire.

Tableau 71. Procédure complète de basculement d'un gestionnaire de domaine après une indisponibilité non planifiée. (suite)

Indisponibilité non planifiée	
3. Dans la base de données de Tivoli Workload Scheduler, attribuez le rôle de gestionnaire de domaine au poste de travail de sauvegarde.	3. Dans la base de données de Tivoli Workload Scheduler, attribuez le rôle de gestionnaire de domaine au poste de travail de sauvegarde.
	4. Définissez le poste de travail qui exécute le gestionnaire de domaine en défaut à `ignore`, à l'aide de la commande `conman cpuname` ou de la console Dynamic Workload Console.
Lorsque vous êtes prêt à restaurer l'appartenance du domaine au gestionnaire de domaine d'origine :	Lorsque vous êtes prêt à restaurer l'appartenance du domaine au gestionnaire de domaine d'origine :
4. Réaffectez l'appartenance du domaine au gestionnaire de domaine d'origine dans la base de données Tivoli Workload Scheduler. Eventuellement, retirez du gestionnaire de domaine d'origine, la commande `conman start` de la procédure d'initialisation et supprimez les copies existantes des fichiers Symphony, Sinfonia et de boîte de message. Nous recommandons d'exécuter cette étape pour éviter que des éléments symphony obsolètes présents dans l'ordinateur ne se déclenchent automatiquement au premier démarrage. Vous pourrez rajouter la commande `conman start` ultérieurement.	5. Supprimez l'indicateur `ignore` du poste de travail qui exécute le gestionnaire de domaine d'origine.
5. Basculez le poste de travail de sauvegarde vers le gestionnaire de domaine en utilisant l'une des méthodes mentionnées à l'étape 1.	6. Réaffectez l'appartenance du domaine au gestionnaire de domaine d'origine dans la base de données Tivoli Workload Scheduler. Eventuellement, retirez du gestionnaire de domaine d'origine, la commande `conman start` de la procédure d'initialisation et supprimez les copies existantes des fichiers Symphony, Sinfonia et de boîte de message. Nous recommandons d'exécuter cette étape pour éviter que des éléments symphony obsolètes présents dans l'ordinateur ne se déclenchent automatiquement au premier démarrage. Vous pourrez rajouter la commande `conman start` ultérieurement.
6. Liez le gestionnaire de domaine à partir du maître pour télécharger une version actualisée du fichier Symphony.	7. Basculez le poste de travail de sauvegarde vers le gestionnaire de domaine en utilisant l'une des méthodes mentionnées à l'étape 1.
	8. Liez le gestionnaire de domaine à partir du maître pour télécharger une version actualisée du fichier Symphony.

Permutation d'un gestionnaire de domaine dynamique

Utilisez la procédure décrite dans «Permutation d'un gestionnaire de domaine maître ou d'un gestionnaire de domaine dynamique», à la page 381 en cas de perte d'un gestionnaire de domaine dynamique pendant une brève période. Les

informations de configuration définies dans Dynamic Workload Broker qui est installé avec le gestionnaire de domaine dynamique font l'objet d'une sauvegarde automatique dans la base de données Tivoli Workload Scheduler. Lorsque vous passez au gestionnaire de domaine dynamique de secours, ces informations sont automatiquement appliquées au gestionnaire de domaine dynamique de secours.

Modification d'un gestionnaire de domaine maître

En cas de perte d'un gestionnaire de domaine maître ou si vous envisagez de le changer, toutes les remarques de la section «Modification d'un gestionnaire de domaine ou d'un gestionnaire de domaine dynamique», à la page 373 sont valables, mais notez également les points suivants :

Choix d'un poste de travail pour la sauvegarde de gestionnaire de domaine maître

Dans la mesure où vous devez transférer les fichiers entre le gestionnaire de domaine maître et son serveur de secours, les postes de travail doivent disposer de systèmes d'exploitation compatibles. N'associez pas des postes de travail UNIX et Windows, et sous UNIX, n'associez pas des postes de travail de format big-endian (HP-UX, Solaris et AIX) et des postes de travail au format little endian (la plupart des systèmes d'exploitation reposant sur Intel, y compris Windows et Linux).

Pour connaître la configuration requise pour un gestionnaire de domaine maître de secours, voir le manuel *Tivoli Workload Scheduler - Guide de planification et d'installation*.

Promotion d'un agent vers le gestionnaire de domaine maître de secours

Le processus normal consiste à installer un gestionnaire de domaine maître de sauvegarde lorsque vous configurez votre réseau de planification. Toutefois, si vous ne l'avez pas fait et que vous décidez par la suite que vous avez besoin d'un gestionnaire de domaine maître de sauvegarde, deux solutions s'offrent à vous :

- Installer un gestionnaire de domaine maître de sauvegarde sur un système ne se trouvant pas actuellement dans le réseau de planification de la charge de travail. Suivez les instructions qui figurent dans *Tivoli Workload Scheduler - Guide de planification et d'installation*
- Promouvoir un agent vers le gestionnaire de domaine maître de secours. Cette option prend du temps et nécessite l'interruption des activités de planification de la charge de travail, mais si vous voulez l'utiliser, suivez la procédure décrite dans cette section.

Vous ne pouvez *pas* promouvoir un agent gestionnaire de domaine maître de secours à l'aide d'une commande ou d'une procédure qui autorise la continuité des activités de planification de la charge de travail.

Si vous avez besoin de modifier un poste de travail agent pour le promouvoir gestionnaire de domaine maître de secours, vous devez interrompre les activités de planification de la charge de travail. La procédure est la suivante :

1. Vérifiez que la configuration du poste de travail correspond à celle requise pour un gestionnaire de domaine maître de secours
2. Si c'est le cas, arrêtez toutes les opérations de planification de la charge de travail en cours sur le poste et désactivez-les
3. Désinstallez l'agent, en suivant les instructions qui figurent dans *Tivoli Workload Scheduler - Guide de planification et d'installation*

4. Installez le gestionnaire de domaine maître de secours sur le système où l'agent a été installé, en suivant les instructions du manuel *Tivoli Workload Scheduler - Guide de planification et d'installation*
5. Assurez-vous que l'entrée de la base de données correspondant au poste de travail est correcte pour un gestionnaire de domaine maître de secours (voir le *Tivoli Workload Scheduler - Guide d'utilisation et de référence* pour plus d'informations sur la définition de poste de travail
6. Définissez et démarrez toutes les opérations de planification de la charge de travail dont vous avez besoin sur le poste de travail.

Définition d'un gestionnaire de domaine maître de sauvegarde

Assurez-vous que sur le gestionnaire de domaine maître comme sur le gestionnaire de domaine maître de secours, l'option *FullStatus* est activée dans la définition du poste de travail. C'est important si vous avez besoin d'utiliser une récupération à long terme, où le gestionnaire de domaine maître de sauvegarde génère un fichier `Symphony` (exécute JnextPlan). Si l'option *FullStatus* n'est pas activée, l'ancien gestionnaire de domaine maître apparaît comme un agent classique tolérant aux pannes après la première occurrence de JnextPlan. Pendant les opérations normales, le travail JnextPlan active automatiquement l'indicateur *FullStatus* pour le gestionnaire de domaine maître, s'il n'est pas encore activé. Lorsque le nouveau gestionnaire de domaine maître exécute JnextPlan, il ne reconnaît pas l'ancien gestionnaire de domaine maître en tant que gestionnaire de domaine maître de sauvegarde sauf si l'indicateur est activé. L'ancien gestionnaire de domaine maître ne dispose pas d'un fichier `Symphony` précis au moment de la commutation inverse.

Assurez-vous également que l'heure du gestionnaire de domaine maître de secours est synchronisée avec celle du gestionnaire de domaine maître. Le moyen le plus sûr pour ce faire consiste à utiliser un serveur Network Time Protocol Server pour contrôler l'heure sur les deux systèmes, avec le même intervalle entre les répétitions.

Copie de fichiers à utiliser sur le gestionnaire de domaine maître de secours

Pour sauvegarder les fichiers importants du gestionnaire de domaine maître dans le gestionnaire de domaine maître de secours, procédez comme suit :

1. Copiez le fichier `Security` figurant sur le gestionnaire de domaine maître dans le répertoire *<TWA_home>*`/TWS` du gestionnaire de domaine maître de secours. Ajoutez un suffixe au fichier afin qu'il n'écrase pas le fichier de sécurité du gestionnaire de domaine maître' de secours, par exemple `Security_from_MDM`.
2. Copiez tous les fichiers dans le répertoire *<TWA_home>*`/TWS/mozart`.
3. Copiez le fichier `localopts` (voir «Définition des options locales», à la page 30 pour connaître son emplacement). Ajoutez un suffixe au fichier afin qu'il n'écrase pas le fichier gestionnaire de domaine maître de sauvegarde' `localopts` ; par exemple, `localopts_from_MDM`.

Cette procédure doit être effectuée lors de chaque création ou lorsque des modifications importantes ont été apportées à un objet quelconque. Cette procédure peut également être appliquée à un script.

Outre ces fichiers nécessaires, vous devez également copier les éléments suivants :

- Tous les scripts que vous avez écrits.
- Les fichiers Symphony archivés, pour référence.
- Les fichiers journaux, pour référence.

Remarque : Une autre solution consiste à placer tous les fichiers indiqués ci-dessus sur un système de fichiers montable distinct qui puisse être aisément démonté du gestionnaire de domaine maître et monté sur le gestionnaire de domaine maître de secours en cas de besoin. Vous souhaiterez sans doute sauvegarder ces fichiers sur un autre système de fichiers montable afin d'éviter toute perte de données.

Pour éviter la perte de messages provoquée par une erreur du gestionnaire de domaine maître, vous pouvez utiliser la fonction de gestionnaire de permutation tolérante aux pannes.

Permutation d'un gestionnaire de domaine maître

Utilisez la procédure décrite dans «Changement d'un gestionnaire de domaine», à la page 374 en cas de perte d'un gestionnaire de domaine maître pendant une brève période.

Le gestionnaire de domaine maître est utilisé jusqu'à une nouvelle opération de changement de gestionnaire de domaine. Pour revenir au gestionnaire de domaine maître d'origine, répétez cette procédure avant la fin de la période de production suivante, sauf si vous ne pensez pas que le gestionnaire de domaine maître sera disponible pendant la prochaine période de production (travail final du Planificateur de travaux et du JnextPlan). Dans ce cas, suivez la procédure de la section suivante.

Perte de longue durée ou changement définitif de gestionnaire de domaine maître

Utilisez la procédure suivante pour permuter vers le serveur de secours si le fonctionnement du gestionnaire de domaine maître d'origine ne doit pas être rétabli avant la prochaine période de nouvelle production (travail final de Planificateur de travaux et JnextPlan). Pour UNIX, utilisez des barres obliques dans les noms de chemin.

1. Utilisez la fonction **stop** de conman pour arrêter Tivoli Workload Scheduler sur le gestionnaire de domaine maître et le gestionnaire de secours correspondant.
2. Si vous avez copié le fichier `Security` du gestionnaire de domaine maître sur le gestionnaire de domaine maître de secours *en ajoutant un suffixe*, supprimez maintenant le fichier `Security` du gestionnaire de domaine maître de secours et renommez le fichier `Security` avec suffixe simplement `Security`.
3. Si vous avez copié le fichier `localopts` à partir du gestionnaire de domaine maître vers le gestionnaire de domaine maître de sauvegarde *avec un suffixe*, fusionnez le fichier `localopts` du gestionnaire de domaine maître de secours avec le fichier `localopts` du gestionnaire de domaine maître. Examinez chaque propriété tour à tour pour déterminer quelle version vous souhaitez conserver sur ce qui deviendra votre nouveau gestionnaire de domaine maître. Par exemple, la propriété *thiscpu* doit être celle du gestionnaire de domaine maître de sauvegarde, mais les options de contrôle de l'exécution des processus peuvent être empruntées au gestionnaire de domaine maître.
4. Sur le gestionnaire de domaine maître de secours, annulez l'option *final* Planificateur de travaux dans le fichier Symphony (elle fait référence au JnextPlan de la prochaine période de production sur l'ancien gestionnaire de domaine maître).
5. Sur le gestionnaire de domaine maître de secours, utilisez le composeur pour modifier tout flots de travaux important qui s'exécute sur le gestionnaire de

domaine maître, en particulier l'option *final* Planificateur de travaux. Pour chacun d'eux, modifiez le nom du poste de travail pour attribuer le nom du poste de travail de secours.

6. Modifiez la définition du poste de travail du gestionnaire de domaine maître pour le faire passer de *gestionnaire* à *agent tolérant aux pannes*.
7. Modifiez la définition du poste de travail du gestionnaire de domaine maître pour le faire passer de *agent tolérant aux pannes* à *gestionnaire*.

   Remarque : Ces deux étapes doivent être effectuées dans l'ordre indiqué, car le système ne permet pas d'avoir deux *gestionnaires* en même temps.
8. Sur le gestionnaire de domaine maître de secours, modifiez le fichier `<TWA_home>/TWS/mozart/globalopts` et remplacez l'option *master* par le nom du poste de travail gestionnaire de domaine maître de secours (ceci est utilisé principalement pour la génération de rapports).
9. Utilisez la fonction **switchmgr** de conman pour passer au gestionnaire de domaine maître de secours. Voir «Changement d'un gestionnaire de domaine», à la page 374.
10. Soumettez une nouvelle option *final* Planificateur de travaux au nouveau gestionnaire de domaine maître (l'ancien gestionnaire de domaine maître de secours).
11. Exécutez **JnextPlan -for 0000** sur le nouveau gestionnaire de domaine maître pour générer le nouveau fichier `Symphony`.
12. N'oubliez pas de vous connecter au gestionnaire de domaine maître de secours en ouvrant Dynamic Workload Console, en définissant d'abord un nouveau moteur pour y accéder.
13. Si l'ancien gestionnaire de domaine maître a échoué ou s'il est remplacé, vous pouvez supprimer sa définition de poste de travail et le retirer du réseau.

Permutation d'un gestionnaire de domaine maître ou d'un gestionnaire de domaine dynamique

La permutation d'un gestionnaire de domaine maître ou d'un gestionnaire de domaine dynamique affecte l'exécution du serveur Dynamic Workload Broker.

L'installation d'un gestionnaire de domaine maître ou d'un gestionnaire de domaine dynamique et de ses postes de travail de secours inclut également l'installation d'un serveur Dynamic Workload Broker.

Avant de permuter votre gestionnaire de domaine maître ou votre gestionnaire de domaine dynamique sur un poste de travail de secours, vous devez arrêter le serveur Dynamic Workload Broker. Une fois la commutation terminée, vous devez démarrer le serveur Dynamic Workload Broker sur le nouveau gestionnaire de domaine maître ou gestionnaire de domaine dynamique. Ce processus n'est pas automatique et vous devez veiller à éviter de vous retrouver avec deux serveurs actifs en même temps.

Si vous devez commuter le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique car le système exécutant le poste de travail courant a échoué, vérifiez également que le serveur Dynamic Workload Broker est arrêté. Dans ce cas, vous devez uniquement démarrer le nouveau serveur Dynamic Workload Broker après avoir commuté le maître.

Si vous permutez le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique pour tout autre raison qu'une panne système et que vous basculez sur

un poste de travail de secours alors que le système exécutant le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique en cours est actif, vous risquez de vous retrouver avec deux serveurs actifs en même temps. Pour éviter cela, arrêtez le serveur Dynamic Workload Broker courant avant d'effectuer la commutation, et démarrez la nouvelle instance lorsque le poste de travail de secours prend le relais.

Voici la procédure à suivre chaque fois que vous permutez le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique si vous exécutez la planification dynamique dans votre réseau :

1. Si le serveur Dynamic Workload Broker de votre gestionnaire de domaine maître ou gestionnaire de domaine dynamique actif est toujours en cours d'exécution, arrêtez-le. Utilisez wastool **stopBrokerApplication.sh** sous UNIX et Linux ou **stopBrokerApplication.bat** sous Windows, comme suit :

   ```
   stopBrokerApplication -user username -password password [-port portnumber]
   ```

   où *username* et *password* correspondent aux données d'identification utilisées lors de l'installation. Le paramètre *portnumber* est facultatif. S'il n'est pas spécifié, la valeur par défaut est utilisée.
2. Basculez le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique sur un poste de travail de secours. Utilisez la commande `conman switchmgr` ou Dynamic Workload Console. Pour plus d'informations, voir la section relative à la commande `switchmgr` dans le document *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.
3. Démarrez le serveur Dynamic Workload Broker en cours d'exécution avec le nouveau poste de travail. Utilisez wastool **startBrokerApplication.sh** sous UNIX et Linux ou **startBrokerApplication.bat** sous Windows, comme suit :

   ```
   startBrokerApplication -user username -password password [-port portnumber]
   ```

   où *username* et *password* correspondent aux données d'identification utilisées lors de l'installation. Le paramètre *portnumber* est facultatif. S'il n'est pas spécifié, la valeur par défaut est utilisée.
4. Liez le serveur Dynamic Workload Broker en cours d'exécution avec le nouveau poste de travail en exécutant la commande suivante :

   ```
   conman link
   workstationname_broker
   ```

Modification des mots de passe Tivoli Workload Scheduler clés

Lorsque vous modifiez les mots de passe des utilisateurs clés dans votre environnement Tivoli Workload Scheduler, vous devez effectuer plusieurs opérations, en fonction du mot de passe d'utilisateur modifié, du type de système d'exploitation sur lequel il est déployé et du type de noeud Tivoli Workload Scheduler sur lequel le mot de passe est modifié. Vous pouvez effectuer ces opérations manuellement, ou vous pouvez utiliser le script **changePassword** décrit dans «Utilisation du script changePassword», à la page 390 pour effectuer automatiquement les opérations requises.

Si vous décidez de procéder manuellement, les pages suivantes décrivent la marche à suivre si le mot de passe de l'un des utilisateurs suivants change :

propriétaire d'instance Tivoli Workload Scheduler
: Le composant <*TWS_user*> (the instance owner) of a Tivoli Workload Scheduler (sur Windows uniquement).

Utilisateur de WebSphere Application Server
L'utilisateur de WebSphere Application Server (tel qu'identifié par les outils WebSphere Application Server) qui authentifie le <*TWS_user*> en cours d'utilisation par les composants Tivoli Workload Scheduler.

Utilisateur de la base de données (J2C) d'un composant Tivoli Workload Scheduler :

DB2 Si vous utilisez une base de données DB2, il s'agit de l'ID utilisateur qui permet d'accéder à DB2.

Remarque : Il diffère selon que c'est le serveur ou le client qui est installé :

Serveur DB2 installé
L'utilisateur d'administration de DB2 (local) est utilisé.

Client DB2 installé
L'utilisateur DB2 de Tivoli Workload Scheduler sur le serveur distant est utilisé.

Oracle Si vous utilisez une base de données Oracle, le propriétaire du schéma Oracle est utilisé.

Remarque : Le propriétaire du schéma Oracle n'est pas un ID du système d'exploitation : même si sa valeur est identique à celle d'un ID du système d'exploitation qui se trouve sur le même ordinateur, ils sont complètement distincts et leurs mots de passe sont modifiés séparément.

Utilisateur Streamlogon
L'utilisateur streamlogon de n'importe quel travail exécuté dans l'environnement Tivoli Workload Scheduler (travaux exécutés sous Windows uniquement)

Pour tous les autres utilisateurs de Tivoli Workload Scheduler, aucune action n'est nécessaire en cas de modification du mot de passe.

Si vous utilisez le script **changePassword**, le mot de passe est modifié et les opérations correspondantes sont effectuées automatiquement. Pour obtenir des informations détaillées sur le script, voir «Utilisation du script changePassword», à la page 390. Si vous décidez de procéder manuellement, consultez tableau 72 pour déterminer si un changement de mot de passe nécessite des actions à entreprendre pour un rôle sur les différents composants Tivoli Workload Scheduler. Recherchez le rôle et le composant, puis déterminer dans la cellule correspondante du tableau où les modifications doivent être apportées :

- Si la cellule contient un "✓", effectuez la modification sur le système sur lequel s'exécute le composant indiqué
- Si la cellule contient "MDM", effectuez la modification sur le gestionnaire de domaine maître auquel appartient le composant

Tableau 72. Le mot de passe doit-il être modifié, et à quel endroit

Rôle	MDM	BKM	FTA	FTA + CONN
Propriétaire de l'instance Tivoli Workload Scheduler (Windows)	✓	✓	✓	✓
WebSphere Application Serverutilisateur	✓	✓		✓

Tableau 72. Le mot de passe doit-il être modifié, et à quel endroit (suite)

Rôle	MDM	BKM	FTA	FTA + CONN
Utilisateur de la base de données	✓	✓		
Utilisateur streamlogon (Windows)	✓	✓	MDM	MDM

Par exemple, si vous êtes le utilisateur_TWS (propriétaire de l'instance) d'un agent tolérant aux pannes, vous devez implémenter la modification de mot de passe sur le système sur lequel réside l'agent tolérant aux pannes. Mais si vous êtes également l'utilisateur streamlogon des travaux qui s'exécutent sur ce système, les modifications requises pour le nouveau mot de passe doivent être appliquées au niveau du gestionnaire de domaine maître auquel appartient l'agent tolérant aux pannes.

Si vous n'êtes pas certain de votre rôle utilisateur, consultez la section «Détermination du rôle de l'utilisateur dont le mot de passe a été modifié».

Une fois déterminé votre rôle utilisateur, consultez la section «Détermination des actions à effectuer», à la page 385 pour déterminer si vous devez entreprendre des actions et, si c'est le cas, à quel l'endroit.

Détermination du rôle de l'utilisateur dont le mot de passe a été modifié

Suivez la procédure ci-dessous pour déterminer le ou les rôle(s) de l'utilisateur dont le mot de passe a été modifié.

Avertissement : Un utilisateur peut jouer plusieurs rôles, auquel cas vous devez suivre plusieurs procédures de modification du mot de passe.

1. Vérifiez si l'utilisateur est le propriétaire de l'instance Tivoli Workload Scheduler :

Windows
Vérifiez si l'utilisateur dont le mot de passe doit être modifié est celui qui est propriétaire du service *Tivoli Workload Scheduler pour <utilisateur_TWS>*.

UNIX Exécutez la commande suivante :

```
ps -ef | grep netman
```

Si l'utilisateur dont le mot de passe a été modifié correspond à l'ID d'utilisateur révélé à cette étape, l'utilisateurs est le *propriétaire de l'instance Tivoli Workload Scheduler*.

2. Vérifiez si l'utilisateur est celui de WebSphere Application Server et/ou celui de la base de données :

1. Connectez-vous à l'ordinateur sur lequel Tivoli Workload Scheduler est installé en tant que l'utilisateur suivant :

   UNIX root

   Windows
   Tout utilisateur du groupe *Administrators*.
2. Accédez au répertoire : *<TWA_home>*/wastools
3. Dans ce répertoire, exécutez le script suivant :

UNIX `showSecurityProperties.sh > <fichier_sortie.txt>`

Windows
`showSecurityProperties.bat > <fichier_sortie.txt>`

Remarque : Il se peut que cette commande affiche un message du serveur d'applications (WASX7357I:) dans le fichier de sortie. Vous pouvez ignorer ce message.

4. Ouvrez <*fichier_sortie.txt*> dans un éditeur de texte.
5. Exécutez le script `showSecurityProperties` afin de vérifier l'ID de serveur associé à la valeur de la clé `activeUserRegistry`. Si l'utilisateur dont le mot de passe a été modifié correspond à la valeur de l'ID de serveur indiqué dans le Panneau Référentiel fédéré, l'utilisateurs est le l'utilisateur *WebSphere Application Server*.
6. Vérifiez la valeur de la clé `j2cUserid` . Si l'utilisateur dont le mot de passe a été modifié correspond à cette clé, il s'agit de l'*utilisateur de base de données*.

   Remarque : Si l'utilisateur est le propriétaire du schéma Oracle, le mot de passe doit également être modifié dans Oracle (voir la documentation Oracle).

3. Vérifiez si l'utilisateur est un utilisateur streamlogon
A l'aide de **composer** ou de Dynamic Workload Console, vérifiez si l'utilisateur est identifié comme un utilisateur. Si c'est le cas, il s'agit d'un *utilisateur streamlogon*.

Une fois identifiés les rôles de chaque utilisateur, voir le tableau 72, à la page 383 pour déterminer si la modification de mot de passe doit être implémentée et à quel endroit il doit l'être, puis consultez la section «Détermination des actions à effectuer».

Détermination des actions à effectuer

Consultez le tableau 73 pour définir les actions à effectuer pour modifier un mot de passe :

Tableau 73. Actions à effectuer pour modifier un mot de passe

	Propriétaire de l'instance TWS (Windows uniquement)	WebSphere Application Serverutilisateur	Utilisateur de la base de données	Utilisateur streamlogon (Windows uniquement)
«Action 1 - modification du mot de passe de l'ID utilisateur WebSphere Application Server», à la page 386		✔ (1, à la page 386)		
«Action 2 - modification du mot de passe utilisé par les clients de ligne de commande pour accéder au gestionnaire de domaine maître», à la page 387		✔		
«Action 3 - modification du mot de passe utilisé par les systèmes de l'agent tolérant aux pannes pour accéder au gestionnaire de domaine maître (pour conman)», à la page 388		✔		

Tableau 73. Actions à effectuer pour modifier un mot de passe (suite)

	Propriétaire de l'instance TWS (Windows uniquement)	WebSphere Application Serverutilisateur	Utilisateur de la base de données	Utilisateur streamlogon (Windows uniquement)
«Action 4 - mise à jour des paramètres de connexion du moteur dans les interfaces graphiques», à la page 388		✓		
«Action 5 - modification du mot de passe de l'ID utilisateur j2c», à la page 388			✓ (1)	
«Action 6 - mise à jour des propriétés SOAP», à la page 389		✓		
La procédure suivante s'applique uniquement aux mots de passe des utilisateurs qui travaillent sous Windows				
«Action 7 - Windows - mise à jour des services Windows», à la page 389	✓	✓		
«Action 8 - modification de la définition de l'utilisateur Tivoli Workload Scheduler», à la page 390				✓

Remarque :

1. Si l'utilisateur est à la fois l'utilisateur de WebSphere Application Server *et* l'utilisateur de la base de données, les modifications apportées par l'exécution de **changeSecurityProperties** peuvent être réalisées comme une action, modifiant les deux mots de passe avec la même valeur.

Action 1 - modification du mot de passe de l'ID utilisateur WebSphere Application Server

Utilisez **changeSecurityProperties** pour modifier le mot de passe de l'ID utilisateur WebSphere Application Server.

Pour cette procédure, vous devez créer un fichier texte contenant les propriétés de sécurité en cours, éditer le fichier, arrêter le serveur d'applications, exécuter l'utilitaire et redémarrer le serveur d'applications.

Remarque : Le fichier texte doit déjà avoir été créé lors de la détermination de votre rôle (voir «Détermination du rôle de l'utilisateur dont le mot de passe a été modifié», à la page 384).

Pour savoir comment procéder, consultez les informations suivantes :

- La section «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423 fournit une description générale de la procédure de modification des propriétés du WebSphere Application Server
- La section «Modification des paramètres de sécurité», à la page 406 fournit des informations de référence sur cet utilitaire
- Lors de l'édition du fichier texte des propriétés de sécurité en cours, recherchez `LocalOSServerpassword` ou `LDAPPassword`, selon le type d'authentification que vous utilisez (voir le «Détermination du rôle de l'utilisateur dont le mot de passe a été modifié», à la page 384) et attribuez la nouvelle valeur au mot de passe, en texte en clair.

Remarque :

1. Si l'utilisateur est à la fois celui de WebSphere Application Server *et* de la base de données, vous pouvez modifier leurs propriétés dans la même action. Voir «Action 5 - modification du mot de passe de l'ID utilisateur j2c», à la page 388. pour plus de détails sur la propriété à modifier.
2. Il se peut que l'utilitaire **changeSecurityProperties** affiche un message provenant du serveur d'applications (WASX7357I:). Vous pouvez ignorer ce message.
3. Lorsque vous fournissez un mot de passe dans un fichier texte pour **changeSecurityProperties**, il existe un léger risque de sécurité. Lorsque vous saisissez un mot de passe dans le fichier, le mot de passe est entré en clair (non chiffré). Après avoir exécuté la commande **changeSecurityProperties**, le mot de passe reste en clair dans le fichier texte que vous avez modifié ; toutefois, si vous exécutez **showSecurityProperties**, la sortie du mot de passe est chiffrée. Ainsi, le risque de sécurité potentiel est limité à la période qui s'écoule entre le moment où vous avez saisi le mot de passe dans le fichier texte et celui où vous avez manuellement supprimé le fichier texte après avoir exécuté **changeSecurityProperties**.

   Avertissement : si vous voulez par la suite modifier d'*autres* paramètres *sans* modifier d'autres mots de passe, vous devez procéder de l'une des façons suivantes avant d'exécuter **changeSecurityProperties** :

   - Reformulez les mots de passe en clair
   - Commentez les propriétés de mot de passe
   - Supprimez les propriétés de mot de passe

   Il s'agit d'éviter que la ligne composée d'astérisques ne s'applique comme mot de passe.

Action 2 - modification du mot de passe utilisé par les clients de ligne de commande pour accéder au gestionnaire de domaine maître

Si vous avez modifié le mot de passe de l'utilisateur WebSphere Application Server qu'utilisent les clients de ligne de commande pour se connecter au gestionnaire de domaine maître, les paramètres de connexion doivent être mis à jour.

Procédez comme suit :

1. Identifiez tous les systèmes dotés d'une connexion distante au client de ligne de commande définie avec le gestionnaire de domaine maître
2. Sur ces postes de travail, ouvrez les fichiers d'options utilisateur (un par utilisateur). Le nom de fichier par défaut est *<User_home>*`/.TWS/useropts`, mais si vous disposez de plusieurs instances de Tivoli Workload Scheduler sur un système, vous avez peut-être implémenté des fichiers d'options d'utilisateur distincts pour effectuer des connexions distinctes, auquel cas vous pouvez consulter la clé *useropts* dans le fichier `localopts` sur chaque instance afin de déterminer le nom du fichier `useropts` spécifique à cette instance.
3. Pour chaque fichier, recherchez la clé de mot de passe (chiffrée) et remplacez sa valeur par celle du nouveau mot de passe en texte en clair, entre guillemets. Le mot de passe est enregistré en clair, mais il sera chiffré lors de la première connexion de l'ID utilisateur.
4. Sauvegardez les fichiers.
5. Vérifiez si le fichier suivant existe : *<Root_home>*`/.TWS/useropts`. Si c'est le cas, modifiez le mot de passe de la même manière.

Action 3 - modification du mot de passe utilisé par les systèmes de l'agent tolérant aux pannes pour accéder au gestionnaire de domaine maître (pour conman)

Si vous avez modifié le mot de passe de l'utilisateur WebSphere Application Server que les agents tolérants aux pannes utilisent avec une connexion HTTP ou HTTPS définie dans les options locales pointant vers le gestionnaire de domaine maître, les paramètres de connexion doivent être mis à jour.

Procédez comme suit :

1. Identifiez tous les agents tolérants aux pannes avec une connexion HTTP ou HTTPS définie dans les options locales pointant vers le gestionnaire de domaine maître.
2. Sur ces postes de travail, ouvrez le fichier d'options de l'utilisateur *<Root_home>*/.TWS/useropts
3. Recherchez la clé du mot de passe (chiffrée) et attribuez-lui comme nouvelle valeur le nouveau mot de passe en texte en clair, entre guillemets. Le mot de passe est enregistré en clair, mais il sera chiffré lors de la première connexion de l'ID utilisateur.
4. Enregistrez le fichier.

Action 4 - mise à jour des paramètres de connexion du moteur dans les interfaces graphiques

Si vous avez modifié le mot de passe de l'utilisateur WebSphere Application Server que Dynamic Workload Console utilise pour se connecter au moteur distribué, les paramètres de connexion du moteur doivent être mis à jour comme suit :

1. Sur chaque instance de Dynamic Workload Console, localisez la page que vous utilisez pour modifier les paramètres de connexion du moteur distribué
2. Modifiez le mot de passe et soumettez la page.

Action 5 - modification du mot de passe de l'ID utilisateur j2c

Utilisez **changeSecurityProperties** pour modifier le mot de passe de l'ID utilisateur de la base de données j2c.

Pour cette procédure, vous devez créer un fichier texte contenant les propriétés de sécurité en cours, modifier le fichier, arrêter le serveur d'applications, exécuter l'utilitaire et redémarrer le serveur d'applications.

Remarque : Vous aurez sans doute créé le fichier texte lors de la détermination de votre rôle (voir «Détermination du rôle de l'utilisateur dont le mot de passe a été modifié», à la page 384).

Pour plus d'informations, consultez les sections suivantes :

- La section «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423 fournit une description générale de la procédure de modification des propriétés de WebSphere Application Server.
- La section «Modification des paramètres de sécurité», à la page 406 fournit des informations de référence sur l'utilitaire **changeSecurityProperties**.
- Lors de l'édition du fichier texte des propriétés de sécurité en cours, recherchez le fichierj2cPassword et attribuez la nouvelle valeur au mot de passe, en texte clair.

Remarque :

1. Si l'utilisateur est à la fois celui de WebSphere Application Server *et* de la base de données, vous pouvez modifier leurs propriétés dans la même action. Pour plus de détails sur les propriétés à modifier, voir «Action 1 - modification du mot de passe de l'ID utilisateur WebSphere Application Server», à la page 386.
2. L'utilitaire **changeSecurityProperties** affichera sans doute un message provenant du serveur d'applications (WASX7357I). Vous pouvez ignorer ce message.
3. Lorsque vous fournissez un mot de passe dans un fichier texte pour **changeSecurityProperties**, il existe un léger risque de sécurité. Lorsque vous saisissez un mot de passe dans le fichier, le mot de passe est entré en clair (non chiffré). Après avoir exécuté la commande **changeSecurityProperties**, le mot de passe reste en clair dans le fichier texte que vous avez modifié ; toutefois, si vous exécutez **showSecurityProperties**, la sortie du mot de passe est chiffrée. Ainsi, le risque de sécurité potentiel est limité à la période qui s'écoule entre le moment où vous avez saisi le mot de passe dans le fichier texte et celui où vous avez manuellement supprimé le fichier texte après avoir exécuté **changeSecurityProperties**.

   Avertissement : Si vous voulez modifier d'*autres* paramètres *sans* modifier de mots de passe, vous devez effectuer l'une des actions suivantes avant d'exécuter **changeSecurityProperties**. Il s'agit d'empêcher que la ligne composée d'astérisques ne s'applique comme mot de passe :

   - Reformulez les mots de passe en clair.
   - Mettez en commentaire les propriétés de mot de passe.
   - Supprimez les propriétés de mot de passe.

Action 6 - mise à jour des propriétés SOAP

Après la modification du mot de passe de l'utilisateur d'administration WebSphere Application Server, il est important de modifier les propriétés du client SOAP à l'aide du script **updateWas.sh/.bat** (pour plus de détails, voir «Serveur d'applications - mise à jour des propriétés SOAP après modification de l'utilisateur WebSphere Application Server ou de son mot de passe», à la page 416). Par exemple :

```
updateWas.sh -user john.smith@domain.com -password zzzz
```

où les options **user** et **password** sont l'utilisateur autorisé à arrêter WebSphere Application Server.

Arrêtez et redémarrez WebSphere Application Server à l'aide des commandes **stopappserver** et **startappserver** pour que les modifications prennent effet.

Action 7 - Windows - mise à jour des services Windows

Sur Windows, le compte <*utilisateur_TWS*> est utilisé pour lancer les services suivants :

- Tivoli Token Service pour <*utilisateur_TWS*>
- Tivoli Workload Scheduler pour <*utilisateur_TWS*>
- IBM WebSphere Application Server V7.0 - <*utilisateur_TWS*>.

Pour que ces services puissent démarrer lors de la prochaine relance de l'ordinateur, vous devez mettre à jour le mot de passe dans leurs propriétés. Pour ce faire, procédez comme suit :

1. Arrêtez tous les processus Tivoli Workload Scheduler. Pour plus d'informations, voir «Suppression des liaisons et arrêt de Tivoli Workload Scheduler», à la page 392.
2. Recherchez le script **updateWasService.bat** dans le répertoire <*TWA_home*>/wastools.
3. Exécutez **updateWasService.bat**, comme décrit dans «Serveur d'applications - mise à jour des services Windows après des modifications», à la page 415, en indiquant le nouveau mot de passe comme <*password_utilisateur_WAS*>.
4. Redémarrez tous les processus Tivoli Workload Scheduler à l'aide de la commande **StartUp**.

Action 8 - modification de la définition de l'utilisateur Tivoli Workload Scheduler

Si l'ID utilisateur est utilisé dans Tivoli Workload Scheduler pour exécuter des travaux, procédez comme suit :

1. Exécutez la commande **composer modify user**. Les détails de l'utilisateur sélectionné sont inscrits dans un fichier temporaire qui s'ouvre alors.
2. Editez la zone du mot de passe de telle sorte qu'elle contienne la nouvelle valeur du mot de passe entre guillemets (").
3. Enregistrez le fichier. Son contenu est ajouté à la base de données.
4. Pour que la modification soit effective immédiatement dans le plan en cours, lancez la commande **conman altpass**.

Pour obtenir la syntaxe complète de ces commandes, voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence*.

Utilisation du script changePassword

Utilisez le script **changePassword** du répertoire <*TWA_home*>/wastools pour modifier les mots de passe de l'un des utilisateurs suivants :

- Propriétaire de l'instance de Tivoli Workload Scheduler (<*utilisateur_TWS*>)
- WebSphere Application Serverutilisateur
- Utilisateur de la base de données (J2C) pour Oracle ou DB2
- Utilisateur streamlogon (Windows uniquement)

Si nécessaire, le script exécute les modifications nécessaires sur le fichier *useropts*, puis arrête et redémarre WebSphere Application Server. Vous pouvez exécuter ce script à partir du gestionnaire de domaine maître ou de l'agent Tivoli Workload Scheduler. Le script détermine le rôle des utilisateurs pour lesquels le mot de passe doit être modifié, et effectue les vérifications et les actions de la procédure manuelle indiquée dans les actions 1 à 8. Exécutez le script comme suit :

UNIX

```
changePassword.sh  -user <USERID>
                      -password <PASSWORD>
                      [-wasuser <WASUSER>]
                      [-waspassword <WASPASSWORD>]
                      [-usroptshome <HOMEDIR>]
```

Où les arguments sont les suivants :

–user <*USERID*>
Cet argument est obligatoire. Indiquez l'utilisateur dont vous modifiez le mot de passe.

–password <*PASSWORD*>
Cet argument est obligatoire. Indiquez le nouveau mot de passe de l'utilisateur.

–wasuser <*WASUSER*>
Cet argument est facultatif. Spécifiez l'utilisateur WebSphere Application Server. Par défaut, la valeur <*USERID*> est utilisée.

–waspassword <*WASPASSWORD*>
Cet argument est facultatif. Spécifiez le mot de passe utilisateur WebSphere Application Server. Par défaut, la valeur <*PASSWORD*> est utilisée. Les valeurs <*WASUSER*> et <*WASPASSWORD*> sont ignorées si le WebSphere Application Server n'est pas présent sur cette instance ou si le script s'exécute pour une instance de Tivoli Workload Scheduler.

–usroptshome<*HOMEDIR*>
Cet argument est facultatif. Le script recherche le fichier USEROPTS dans le répertoire *TWA_home*/*.TWS*WebSphere Application Server. Cet argument est ignoré si le script n'est pas en cours d'exécution pour une instance Tivoli Workload Scheduler.

Windows

```
changePassword.bat  -user <USERID>
                      -password <PASSWORD>
                      [-srvuser <SRVUSERID>]
                      [-srvpassword <SRVPASSWORD>]
                      [-wasuser <WASUSERID>]
                      [-waspassword <WASPASSWORD>]
                      [-usroptshome <HOMEDIR>]
                      [-streamlogonws <WS>]
                      [-streamlogondm <DOMAIN>]
```

Où les arguments sont les suivants :

–user <*USERID*>
Cet argument est obligatoire. Indiquez l'utilisateur dont vous modifiez le mot de passe.

–password <*PASSWORD*>
Cet argument est obligatoire. Indiquez le nouveau mot de passe de l'utilisateur.

–srvuser <*SRVUSERID*>
Cet argument est facultatif. Indiquez l'utilisateur du service Windows. Si vous exécutez le script pour une instance Tivoli Workload Scheduler, la valeur spécifiée ici est identique à l'utilisateur Tivoli Workload Scheduler. Par défaut, la valeur <*USERID*> est utilisée.

–srvpassword <*SRVPASSWORD*>
Cet argument est facultatif. Le mot de passe de l'utilisateur du service Windows. Par défaut, la valeur <*PASSWORD*> est utilisée.

–wasuser <*WASUSER*>
Cet argument est facultatif. Spécifiez l'utilisateur WebSphere Application Server. Par défaut, la valeur <*USERID*> est utilisée.

–waspassword <*WASPASSWORD*>
Cet argument est facultatif. Spécifiez le mot de passe utilisateur WebSphere Application Server. Par défaut, la valeur <*PASSWORD*> est utilisée. Les valeurs <*WASUSER*> et <*WASPASSWORD*> sont ignorées si le WebSphere

Application Server n'est pas présent sur cette instance ou si le script s'exécute pour une instance de Tivoli Workload Scheduler.

–usroptshome<*HOMEDIR*>
Cet argument est facultatif. Le script recherche le fichier `USEROPTS` dans le répertoire *`TWA_home/.TWS`*WebSphere Application Server. par défaut, le répertoire principal de l'utilisateur exécutant le script est utilisé.

–streamlogonws<*WS*>
Cet argument est facultatif (Windows uniquement). Le script met à jour la définition de l'utilisateur pour l'utilisateur de `<WS>#<DOMAIN>/<USER>` dans la base de données Tivoli Workload Scheduler. Par défaut, la définition de l'utilisateur pour `<USER>` est mise à jour. La mise à jour est effectuée uniquement si l'outil est en cours d'exécution sur le gestionnaire de domaine maître dans un environnement Windows.

–streamlogondm<*DOMAIN*>
Cet argument est facultatif. Spécifiez le domaine de l'utilisateur pour `<USER>`.

Suppression des liaisons et arrêt de Tivoli Workload Scheduler

Avant d'effectuer une mise à niveau ou une désinstallation, d'installer un groupe de correctifs ou de procéder à des activités de maintenance, vérifiez que tous les processus et services Tivoli Workload Scheduler sont arrêtés. Procédez comme suit :

1. Si des travaux sont en cours d'exécution sur le poste de travail, attendez qu'ils s'achèvent. Pour identifier ceux qui ne sont pas terminés, recherchez ceux dont l'état est *exec*. Lorsqu'aucun travail n'est dans cet état et que vous disposez d'assez de temps pour que tous les événements soient distribués sur le réseau, vous pouvez continuer la procédure.
2. Si le poste de travail que vous souhaitez arrêter n'est pas le gestionnaire de domaine maître, supprimez les liens du poste de travail en émettant la commande suivante à partir de la ligne de commande du gestionnaire de domaine maître :

   ```
   conman "unlink workstationname;noask"
   ```

3. Tous les processus Tivoli Workload Scheduler du poste de travail doivent ensuite être arrêtés manuellement. A partir de la ligne de commande, lorsque vous êtes connecté en tant qu'<*utilisateur_TWS*>, utilisez la commande suivante :

   ```
   conman "stop;wait"
   ```

4. A partir de la ligne de commande, arrêtez le processus netman de la manière suivante :

   UNIX Exécutez :

   ```
   conman "shut"
   ```

   Remarque : N'utilisez pas la commande UNIX **`kill`** pour arrêter les processus Tivoli Workload Scheduler.

   Windows
   Exécutez la commande `shutdown.cmd` à partir du répertoire racine Tivoli Workload Scheduler.

5. Si le poste de travail se trouve à la version V8.4 ou supérieure, arrêtez l'agent SSM, comme suit :

- Sous Windows, arrêtez le service Windows : Tivoli Workload Scheduler SSM Agent (pour *<utilisateur_TWS>*).
- Sous UNIX, exécutez **stopmon**.

6. Si vous mettez à jour un agent, supprimez (démontez) les répertoires montés sur tous les répertoires NFS du gestionnaire de domaine maître.
7. Si vous mettez à niveau une version comportant le connecteur, arrêtez le connecteur.
8. Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)

Pour vérifier si des services et des processus sont encore en cours d'exécution, suivez la procédure ci-dessous :

UNIX Saisissez la commande :

```
ps -u <utilisateur_TWS>
```

Vérifiez que les processus suivants ne sont pas en cours d'exécution : netman, mailman, batchman, writer, jobman, JOBMAN, stageman, logman, planman, monman, ssmagent.bin et appservman.

Windows

Exécutez **Task Manager**, puis vérifiez que les processus suivants ne sont pas en cours d'exécution : netman, mailman, batchman, writer, jobman, stageman, JOBMON, tokensrv, batchup, logman, planman, monman, ssmagent et appservman.

Vérifiez également qu'aucun programme système n'accède au répertoire ou aux sous-répertoires, y compris l'invite de commande et l'explorateur Windows.

Modification du nom d'hôte, du port ou du nom d'une base de données

Pour modifier le nom d'hôte, le port ou le nom d'une base de données, la procédure diffère selon que la base de données se trouve sur DB2 ou Oracle :

- «Modification du nom d'hôte, du port ou du nom de base de données DB2»
- «Modification du nom d'hôte, du port ou du nom de base de données Oracle», à la page 400

Modification du nom d'hôte, du port ou du nom de base de données DB2

Si vous devez modifier le nom d'hôte DB2, le port ou le nom de la base de données, vous pouvez faire appel à l'utilitaire **changeDataSourceProperties** pour refléter ces modifications au niveau du serveur d'applications sur le gestionnaire de domaine maître.

Au moment de l'installation de Tivoli Workload Scheduler, le nom de base de données par défaut utilisé pour sa création était *TWS* (que vous avez peut-être modifié). Vous avez également fourni le nom du port et le nom d'hôte du serveur DB2. Pour modifier ces détails, procédez comme suit :

1. Arrêtez DB2 et Tivoli Workload Scheduler

2. Utilisez les fonctions de DB2 (voir la documentation DB2 pour plus de détails) ou celles du système d'exploitation pour modifier le nom, le port ou le nom d'hôte de la base de données.
3. Modifiez la configuration du serveur d'applications Tivoli Workload Scheduler de façon à ce qu'il pointe directement sur la configuration modifiée de DB2.

   Pour cette procédure, vous devez arrêter le serveur d'applications, créer un fichier texte contenant les propriétés de source de données en cours, éditer le fichier, exécuter l'utilitaire et redémarrer le serveur d'applications. Pour savoir comment procéder, consultez les informations suivantes :

   - La section «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423 fournit une description générale de la procédure de modification des propriétés du WebSphere Application Server
   - La section «Modification des propriétés de source de données», à la page 395 dresse la liste de toutes les propriétés de source de données et fournit d'autres informations de référence sur l'utilitaire
   - Lors de l'édition du fichier texte des propriétés de source de données en cours, procédez comme suit :

     a. Editez le fichier texte et localisez les propriétés suivantes :

     ```
     ################################################################
     # DB2 Type4 Resource Properties
     ################################################################
     DB2Type4DatabaseName=TWS
     DB2Type4ServerName=localhost
     DB2Type4PortNumber=50083
     ```

     b. Définissez les entrées suivantes :

     DB2Type4DatabaseName
     Nouveau nom de la base de données Tivoli Workload Scheduler.

     DB2Type4ServerName
     Nouveau nom d'hôte du serveur DB2.

     DB2Type4PortNumber
     Nouveau port du serveur DB2.

     Lorsque vous modifiez un port de serveur DB2, vous devez aussi modifier la configuration du noeud sur lequel le Tivoli Workload Scheduler a été catalogué :

     – Si vous travaillez avec un client DB2, ouvrez une session de ligne de commande et connectez-vous en tant qu'administrateur DB2, puis exécutez les commandes suivantes :

     ```
     DB2 CLIENT
     db2 uncatalog node <TWSDBNAME>_ND
     db2 catalog tcpip node <TWSDBNAME>_ND remote <HOSTNAME>
     server <NEWPORT>
     ```

     – Si vous travaillez avec un serveur DB2, ouvrez une session de ligne de commande et connectez-vous en tant qu'administrateur DB2, puis exécutez les commandes suivantes :

     ```
     DB2 SERVER
     db2 uncatalog node LBNODE
     db2 catalog tcpip node LBNODE remote 127.0.0.1 server <NEWPORT>
     ```

     Ne modifiez aucune autre propriété.

   Remarque : Il se peut que l'utilitaire affiche un message provenant du serveur d'applications (WASX7357I:). Vous pouvez ignorer ce message.
4. Démarrez DB2 et Tivoli Workload Scheduler.

Ce script peut également être utilisé pour modifier d'autres propriétés de source de données. Mais, auquel cas, Tivoli Workload Scheduler risque de ne pas fonctionner correctement. Pour toute autre modification, il est conseillé de suivre exclusivement les instructions du service de support logiciel IBM, afin de corriger certains problèmes spécifiques. Par exemple, pour résoudre des problèmes relatifs au pilote JBDC, voir «Résolution de problèmes relatifs au pilote JDBC», à la page 398.

Modification des propriétés de source de données

Vous exécutez le script **changeDataSourceProperties** sur le gestionnaire de domaine maître pour modifier les propriétés de source de données du RDBMS utilisé avec le gestionnaire de domaine maître. Vous devez mettre à jour les propriétés de source de données dans les cas suivants :

- Vous migrez vos données depuis une base de données Oracle vers DB2 à l'aide de la méthode de reconfiguration.
- Vous changez le nom de la base de données, le serveur, l'hôte ou le port.
- Vous changez le chemin d'accès au pilote JDBC du SGBD relationnel.

La procédure d'exécution du script est décrite en détails dans «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423, mais en résumé, vous effectuez les opérations suivantes :

- Exécutez **showDataSourceProperties.sh (.bat) > my_file_name** pour obtenir les propriétés actuelles
- Modifiez *mon_fichier*
- Exécutez **changeDataSourceProperties.sh (.bat) my_file_name**

Remarque : *my_file_name* doit être le chemin d'accès qualifié complet du fichier.

L'utilitaire de modification appelle l'utilitaire **wsadmin** en exécutant **ChangeDataSourceProperties.jacl** avec le fichier de propriétés comme entrée.

Seules les `resources.xml` de WebSphere Application Server sont affectées par ce script. Le chemin complet du fichier est :

```
<chemin_profil_WAS>/config/cells/TWSNodeCell/nodes/
  TWSNode/servers/server1/resources.xml
```

où la valeur par défaut de *<chemin_profil_WAS>* est *<rép_base_TWA>*/WAS/TWSprofile.

Voici une liste des propriétés modifiables à l'aide de cet utilitaire. Seules certaines options indiquées sont actuellement utilisées par Tivoli Workload Scheduler.

```
################################################################
JDBC Path Variables
################################################################
ORACLE_JDBC_DRIVER_PATH=
DB2_JDBC_DRIVER_PATH=c:/ibm/sqllib/java
DB2UNIVERSAL_JDBC_DRIVER_PATH=c:/ibm/sqllib/java


################################################################
DB2 Type2 Resource Properties
################################################################
DB2Type2JndiName=
DB2Type2Description=
DB2Type2ConnectionAttribute=cursorhold=0
DB2Type2EnableMultithreadedAccessDetection=false
DB2Type2Reauthentication=false
DB2Type2JmsOnePhaseOptimization=false
```

```
DB2Type2DatabaseName=TWSZ_DB
DB2Type2PreTestSQLString=SELECT 1 FROM SYSIBM.SYSDUMMY1


#############################################################
DB2 Type4 Resource Properties
#############################################################
DB2Type4JndiName=jdbc/twsdb
DB2Type4DatabaseName=TWSZ
DB2Type4DriverType=4
DB2Type4ServerName=myhost.mydomain.com
DB2Type4PortNumber=50000
DB2Type4SslConnection=false
DB2Type4Description=
DB2Type4TraceLevel=
DB2Type4TraceFile=
DB2Type4FullyMaterializeLobData=true
DB2Type4ResultSetHoldability=2
DB2Type4CurrentPackageSet=
DB2Type4ReadOnly=false
DB2Type4DeferPrepares=true
DB2Type4CurrentSchema=
DB2Type4CliSchema=
DB2Type4RetrieveMessagesFromServerOnGetMessage=true
DB2Type4ClientAccountingInformation=
DB2Type4ClientApplicationInformation=
DB2Type4ClientUser=
DB2Type4ClientWorkstation=
DB2Type4CurrentPackagePath=
DB2Type4CurrentSQLID=
DB2Type4KerberosServerPrincipal=
DB2Type4LoginTimeout=0
DB2Type4SecurityMechanism=
DB2Type4TraceFileAppend=false
DB2Type4CurrentFunctionPath=
DB2Type4CursorSensitivity=
DB2Type4KeepDynamic=
DB2Type4CurrentLockTimeout=
DB2Type4EnableMultithreadedAccessDetection=false
DB2Type4Reauthentication=false
DB2Type4JmsOnePhaseOptimization=false
DB2Type4PreTestSQLString=SELECT 1 FROM SYSIBM.SYSDUMMY1
DB2Type4DbFailOverEnabled=false
DB2Type4ConnRetriesDuringDBFailover=100
DB2Type4ConnRetryIntervalDuringDBFailover=3000
# DB2Type4IsolationLevel peut correspondre à :
#       CURSOR_STABILITY ou READ_STABILITY
DB2Type4IsolationLevel=CURSOR_STABILITY


#############################################################
Oracle Type2 Resource Properties
#############################################################
OracleType2JndiName=
OracleType2DriverType=
OracleType2URL=jdbc:oracle:oci:@ORCL
OracleType2DatabaseName=
OracleType2ServerName=
OracleType2PortNumber=1521
OracleType2OracleLogFileSizeLimit=0
OracleType2OracleLogFileCount=1
OracleType2OracleLogFileName=
OracleType2OracleLogTraceLevel=INFO
OracleType2OracleLogFormat=SimpleFormat
OracleType2OracleLogPackageName=oracle.jdbc.driver
OracleType2TNSEntryName=
OracleType2NetworkProtocol=
```

```
OracleType2DataSourceName=
OracleType2LoginTimeout=
OracleType2Description=
OracleType2EnableMultithreadedAccessDetection=false
OracleType2Reauthentication=false
OracleType2JmsOnePhaseOptimization=false
OracleType2PreTestSQLString=SELECT 1 FROM DUAL
OracleType2DbFailOverEnabled=false
OracleType2ConnRetriesDuringDBFailover=100
OracleType2ConnRetryIntervalDuringDBFailover=3000


#############################################################
Oracle Type4 Resource Properties
#############################################################
OracleType4JndiName=
OracleType4DriverType=
OracleType4URL=jdbc:oracle:thin:@//localhost:1521/ORCL
OracleType4DatabaseName=
OracleType4ServerName=
OracleType4PortNumber=1521
OracleType4OracleLogFileSizeLimit=0
OracleType4OracleLogFileCount=1
OracleType4OracleLogFileName=
OracleType4OracleLogTraceLevel=INFO
OracleType4OracleLogFormat=SimpleFormat
OracleType4OracleLogPackageName=oracle.jdbc.driver
OracleType4TNSEntryName=
OracleType4NetworkProtocol=
OracleType4DataSourceName=
OracleType4LoginTimeout=
OracleType4Description=
OracleType4EnableMultithreadedAccessDetection=false
OracleType4Reauthentication=false
OracleType4JmsOnePhaseOptimization=false
OracleType4PreTestSQLString=SELECT 1 FROM DUAL
OracleType4DbFailOverEnabled=false
OracleType4ConnRetriesDuringDBFailover=100
OracleType4ConnRetryIntervalDuringDBFailover=3000
```

Lorsque vous modifiez les propriétés de la source de données, appliquez les règles suivantes :

- Si une propriété n'est pas fournie dans le fichier de propriétés, la valeur en cours n'est pas modifiée
- Si une propriété est fournie avec une valeur non vide, la valeur actuelle est mise à jour.
- Si une propriété est fournie avec une valeur vide, le paramètre est défini sur vide si la propriété est classée en tant qu'effaçable ou laissée telle quelle dans le cas contraire.
- Utilisez toujours des sources de données de type 4 pour DB2 et de type 2 pour Oracle.
- Définissez la variable appropriée du chemin d'accès au pilote JDBC pour le SGBD relationnel de votre choix.
  - Pour DB2, le pilote JDBC se trouve dans le sous-dossier `java` du répertoire `sqllib`. Par exemple :

    ```
    DB2_JDBC_DRIVER_PATH=c:/program files/ibm/sqllib/java
    ```

    ou

    ```
    DB2UNIVERSAL_JDBC_DRIVER_PATH=c:/program files/ibm/sqllib/java
    ```
  - Pour Oracle, il se trouve dans le sous-dossier `jdbc/lib` du répertoire de base Oracle. Par exemple :

```
ORACLE_JDBC_DRIVER_PATH=C:/Oracle/product/10.2.0/db_1/jdbc/lib
```

- Assurez-vous que le nom JNDI de la source de données est toujours défini sur jdbc/twsdb dans la propriété ...JndiName du système de gestion de base de données relationnelle que vous utilisez. Si vous modifiez le système de gestion de base de données relationnelle, procédez comme suit :
  1. Attribuez un nouveau nom à la propriété ...JndiName du système de gestion de base de données relationnelle pour lequel vous optez.
  2. Définissez à jdbc/twsdb la propriété ...JndiName du nouveau système de gestion de base de données relationnelle.
- Vérifiez la définition des propriétés suivantes :
  – Pour DB2 :

    ```
    DB2Type4JndiName
    DB2Type4DatabaseName
    DB2Type4ServerName
    DB2Type4PortNumber
    ```

  – Pour Oracle :

    ```
    OracleType2JndiName
    OracleType2DatabaseName
    OracleType2ServerName
    OracleType2PortNumber
    ```

Affichage des propriétés en cours de la source de données : Pour afficher les propriétés en cours, utilisez l'utilitaire suivant :

UNIX **showDataSourceProperties.sh**

Windows
showDataSourceProperties.bat

Résolution de problèmes relatifs au pilote JDBC

Tivoli Workload Scheduler est fourni avec le pilote JDBC de type 4 pour DB2 et de type 2 pour Oracle. Néanmoins, chacun peut utiliser le type de pilote de l'autre, si nécessaire. Il se peut que le service de support logiciel IBM vous demande d'utiliser ce pilote. La présente section explique la procédure à suivre.

Avertissement : Cette procédure ne doit être exécutée que sous le contrôle du service de support logiciel IBM.

Pour changer de pilote, vous devez modifier les propriétés de la source de données en suivant la procédure décrite à la section «Modification du nom d'hôte, du port ou du nom d'une base de données», à la page 393. Toutefois, les paramètres que vous modifiez sont différents. Voici un exemple de paramètres de type 4 et de type 2 pour DB2 :

Paramètres du pilote JDBC de type 4

```
############################################################
# DB2 Type4 Resource Properties
############################################################
DB2Type4JndiName=jdbc/twsdb
DB2Type4DatabaseName=TWSZ
DB2Type4DriverType=4
DB2Type4ServerName=myhost.mydomain.com
DB2Type4PortNumber=50000
DB2Type4SslConnection=false
DB2Type4Description=
DB2Type4TraceLevel=
DB2Type4TraceFile=
DB2Type4FullyMaterializeLobData=true
DB2Type4ResultSetHoldability=2
DB2Type4CurrentPackageSet=
```

```
DB2Type4ReadOnly=false
DB2Type4DeferPrepares=true
DB2Type4CurrentSchema=
DB2Type4CliSchema=
DB2Type4RetrieveMessagesFromServerOnGetMessage=true
DB2Type4ClientAccountingInformation=
DB2Type4ClientApplicationInformation=
DB2Type4ClientUser=
DB2Type4ClientWorkstation=
DB2Type4CurrentPackagePath=
DB2Type4CurrentSQLID=
DB2Type4KerberosServerPrincipal=
DB2Type4LoginTimeout=0
DB2Type4SecurityMechanism=
DB2Type4TraceFileAppend=false
DB2Type4CurrentFunctionPath=
DB2Type4CursorSensitivity=
DB2Type4KeepDynamic=
DB2Type4CurrentLockTimeout=
DB2Type4EnableMultithreadedAccessDetection=false
DB2Type4Reauthentication=false
DB2Type4JmsOnePhaseOptimization=false
DB2Type4PreTestSQLString=SELECT 1 FROM SYSIBM.SYSDUMMY1
DB2Type4DbFailOverEnabled=false
DB2Type4ConnRetriesDuringDBFailover=100
DB2Type4ConnRetryIntervalDuringDBFailover=3000
# DB2Type4IsolationLevel peut correspondre à :
#       CURSOR_STABILITY ou READ_STABILITY
DB2Type4IsolationLevel=CURSOR_STABILITY
```

Paramètres du pilote JDBC de type 2

```
################################################################
# DB2 Type2 Resource Properties
################################################################
DB2Type2JndiName=
DB2Type2Description=
DB2Type2ConnectionAttribute=cursorhold=0
DB2Type2EnableMultithreadedAccessDetection=false
DB2Type2Reauthentication=false
DB2Type2JmsOnePhaseOptimization=false
DB2Type2DatabaseName=TWSZ_DB
DB2Type2PreTestSQLString=SELECT 1 FROM SYSIBM.SYSDUMMY1
```

Changement de pilote ou modification du nom JNDI : Le nom JNDI de la source de données doit être unique. Dans les exemples qui précèdent, le nom JNDI du pilote de type 4 est défini sur la valeur correcte. Pour changer de pilote, modifiez les paramètres de telle sorte que les valeurs soient inversées, comme suit :

```
Exemple 1 : valeurs par défaut pour le nom JNDI :

#DB2Type4JndiName=jdbc/twsdb

...

#DB2Type2JndiName=jdbc/twsdb2



Exemple 2 : valeurs inversées pour le nom JNDI :

#DB2Type4JndiName=jdbc/twsdb2

...

#DB2Type2JndiName=jdbc/twsdb
```

Pour attribuer une autre valeur aux noms du pilote, voir :

```
Exemple 3 : autres valeurs pour le nom JNDI :

#DB2Type4JndiName=jdbc/twsdb_test4

...

#DB2Type2JndiName=jdbc/twsdb_test2
```

Modification du nom d'hôte, du port ou du nom de base de données Oracle

Si vous devez modifier le nom d'hôte Oracle, le port ou le nom de la base de données, vous pouvez normalement gérer ce changement au sein d'Oracle. puisque WebSphere Application Server pointe vers le service Oracle dans lequel ces éléments sont définis. Pour savoir comment les modifier, consultez la documentation Oracle.

Cependant, les propriétés que vous modifiez sont parfois définies dans *<chemin_profil_WAS>*/properties/TWSConfig.properties, où *chemin_profil_WAS* correspond au chemin du profil WebSphere Application Server que vous avez défini au moment de l'installation. Le chemin par défaut est *TWA_home*/WAS/TWSprofile. Dans ce cas, vous devez vous assurer qu'elles sont également modifiées à cet endroit. Les propriétés en question sont les suivantes :

```
com.ibm.tws.dao.rdbms.rdbmsName = ORACLE
com.ibm.tws.dao.rdbms.modelSchema = <TWS_Oracle_User>
com.ibm.tws.dao.rdbms.eventRuleSchema = <TWS_Oracle_User>
com.ibm.tws.dao.rdbms.logSchema = <TWS_Oracle_User>
```

Modification du nom d'hôte ou de l'adresse IP du poste de travail

Lorsque vous modifiez le nom d'hôte et/ou l'adresse IP sur les postes de travail de votre environnement Tivoli Workload Scheduler, pour qu'il fonctionne correctement, vous devez reporter les valeurs modifiées sur :

- WebSphere Application Server si les composants suivants ont modifié le nom d'hôte et/ou l'adresse IP :
  - Gestionnaire de domaine maître
  - gestionnaire de domaine maître de secours
  - Connecteur ou connecteur z/OS
  - Dynamic Workload Console

  Pour plus d'informations, voir «Report des modifications dans le fichier de configuration WebSphere Application Server», à la page 401.
- Les composants suivants en cas de modification du nom d'hôte et/ou de l'adresse IP du poste de travail sur lequel vous avez installé le SGBD relationnel :
  - Gestionnaire de domaine maître
  - gestionnaire de domaine maître de secours
  - gestionnaire de domaine dynamique
  - gestionnaire de domaine dynamique de secours

  Pour plus d'informations, voir «Report de la valeur de nom d'hôte ou d'adresse IP modifiée du poste de travail sur lequel vous avez installé le SGBD relationnel», à la page 402.
- Les définitions du poste de travail si vous avez installé les composants suivants :
  - Gestionnaire de domaine maître
  - gestionnaire de domaine maître de secours

– gestionnaire de domaine dynamique
– gestionnaire de domaine dynamique de secours
– Agent tolérant aux pannes et agent standard
– Gestionnaire de domaine

Pour plus d'informations, voir «Report de la valeur de nom d'hôte ou d'adresse IP modifiée dans la définition du poste de travail», à la page 403.

Report des modifications dans le fichier de configuration WebSphere Application Server

Si les composants suivants ont modifié le nom d'hôte ou l'adresse IP, vous devez reporter la valeur modifiée dans le fichier de configuration WebSphere Application Server, comme suit :

- Gestionnaire de domaine maître
- gestionnaire de domaine maître de secours
- Connecteur ou connecteur z/OS
- Dynamic Workload Console

1. Arrêtez le serveur WebSphere Application Server.
2. Obtenez la valeur modifiée du nom d'hôte et/ou de l'adresse IP.
3. Exécutez l'outil **showHostProperties** en redirigeant la sortie vers un fichier texte afin d'obtenir les propriétés en cours.
4. Ouvrez le fichier et accédez au panneau de configuration de l'hôte.

   Voici un exemple de section à consulter :

```
################################################################
# Host Configuration Panel
################################################################
# Old Hostname
oldHostname=myoldhost.romelab.ibm.it.com
# New Hostname
newHostname=mynewhost.romelab.ibm.it.com.....

################################################################
Ports Configuration Panel
################################################################
bootPort=41117
bootHost=myhost.mydomain.com
soapPort=41118
soapHost=myhost.mydomain.com
httpPort=41115
httpHost=*
httpsPort=41116
httpsHost=*
adminPort=41123
adminHost=*
adminSecurePort=41124
adminSecureHost=*
sasPort=41119
sasHost=myhost.mydomain.com
csiServerAuthPort=41120
csiServerAuthHost=myhost.mydomain.com
csiMuthualAuthPort=41121
csiMuthualAuthHost=myhost.mydomain.com
orbPort=41122
orbHost=myhost.mydomain.com
```

5. Vérifiez que les valeurs des propriétés répertoriées ci-dessous ont été remplacées par les valeurs réelles :
   - Old Hostname
   - New Hostname
   - Noms d'hôte des propriétés de port spécifiques

Si ces valeurs sont différentes des valeurs de nom d'hôte et d'adresse IP réelles, passez à l'étape 6. Si ces valeurs n'ont pas changé, ignorez le étapes ci-dessous.

6. Modifiez les valeurs des propriétés en exécutant l'outil **changeHostProperties**. Pour de plus amples informations, voir «Serveur d'applications - modification du nom d'hôte ou des ports TCP/IP», à la page 419.
7. Redémarrez WebSphere Application Server si vous n'avez pas besoin d'apporter des modifications au SGBD relationnel. Pour modifier le SGBD relationnel, voir «Report de la valeur de nom d'hôte ou d'adresse IP modifiée du poste de travail sur lequel vous avez installé le SGBD relationnel».
8. Propagez les modifications aux interfaces comme suit :

   Adresse des modifications du gestionnaire de domaine maître

   - Sur chaque agent tolérant aux pannes, agent dynamique, et agent standard que vous avez configuré pour une connexion à la ligne de commande **comman**, mettez à jour le paramètre **host** présent dans la section "Attributs des connexions CLI" dans le fichier `localopts`. En principe, le paramètre **host** est défini dans le fichier `localopts` des postes de travail que vous utilisez pour soumettre les travaux prédéfinis et les flots de travaux (commandes sbj et sbs).
   - Sur chaque client de ligne de commande, mettez à jour le paramètre **host** présent dans la section "Attributs des connexions CLI" dans le fichier `localopts`.
   - Sur Dynamic Workload Console mettez à jour les connexions au moteur.

   Adresse des modifications de Dynamic Workload Console

   Communiquez la nouvelle adresse Web à tous les utilisateurs.

Report de la valeur de nom d'hôte ou d'adresse IP modifiée du poste de travail sur lequel vous avez installé le SGBD relationnel

Si vous avez modifié le nom d'hôte ou l'adresse IP du poste de travail sur lequel vous avez installé le SGBD relationnel, contactez l'administrateur de base de données pour reconfigurer votre SGBD relationnel afin qu'il utilise le nouveau nom d'hôte ou la nouvelle adresse IP. Si vous utilisez DB2, voir la procédure décrite dans le document https://www-304.ibm.com/support/docview.wss?uid=swg21258834.

Propagez les modifications sur les composants suivants, comme suit :

- Gestionnaire de domaine maître
- gestionnaire de domaine maître de secours
- gestionnaire de domaine dynamique
- gestionnaire de domaine dynamique de secours

1. Arrêtez le serveur WebSphere Application Server.
2. Exécutez l'outil **showDataSourceProperties** en redirigeant la sortie vers un fichier texte afin d'obtenir les propriétés en cours.
3. Ouvrez le fichier et identifiez la section de la base de données où la propriété *databasetype*`Type`*n*`JndiName` est égale à **jdbc/twsdb**.

   Où *databasetype* correspond à la base de données que vous utilisez, par exemple DB2, et *n* peut avoir la valeur 2 ou 4.

   Voici un exemple de section à consulter si vous utilisez DB2 :

```
############################################################
DB2 Type4 Resource Properties
############################################################
DB2Type4JndiName=jdbc/twsdb
DB2Type4DatabaseName=TWSZ
DB2Type4DriverType=4
DB2Type4ServerName=myhost.mydomain.com
.....
```

4. Vérifiez la valeur de la propriété *databasetype*Type*n*ServerName. Si cette valeur est modifiée, passez à l'étape 5. Si cette valeur n'a pas changé, ignorez les étapes ci-dessous.
5. Modifiez la propriété *databasetype*Type*n*ServerName=*value* en exécutant **changeDataSourceProperties**. Pour de plus amples informations, voir «Modification du nom d'hôte, du port ou du nom d'une base de données», à la page 393.
6. Pour utiliser :

   les rapports Dynamic Workload Console
   Mettez à jour les connexions de base de données.

   Rapports de ligne de commande
   Mettez à jour la section suivante du fichier <*report_home*>\config\common.properties

   ```
   ######################################################################
   # DATABASE PROPERTIES
   ######################################################################
   # Indiquez le nom d'hôte ou l'adresse TCP/IP de la base de données,
   # son numéro de port et son nom.
   DatabaseHostname=<hostname>
   DatabasePort=50000
   DatabaseName=TWS
   ......................
   ```

   Ou <*rép_principal_rapport*> est le répertoire dans lequel vous extrayez le package.
7. Redémarrez le serveur WebSphere Application Server.

Report de la valeur de nom d'hôte ou d'adresse IP modifiée dans la définition du poste de travail

Exécutez cette procédure si vous avez modifié le nom d'hôte ou l'adresse IP dans les composants suivants :

- Gestionnaire de domaine maître
- gestionnaire de domaine maître de secours
- Agent tolérant aux pannes et agent standard
- Gestionnaire de domaine

Pour modifier le nom d'hôte ou l'adresse IP dans la définition du poste de travail, procédez comme suit :

1. Utilisez **composer** ou Dynamic Workload Console pour vérifier la définition du poste de travail stockée dans la base de données pour l'instance Tivoli Workload Scheduler installée sur le poste de travail sur lequel l'adresse IP ou le nom d'hôte ont été modifiés.
2. Vérifiez que l'attribut **node** contient le nouveau nom d'hôte ou la nouvelle adresse IP. Si cette valeur est modifiée, passez à l'étape 3. Si cette valeur n'a pas changé, ignorez les étapes ci-dessous.
3. Remplacez la valeur du paramètre **node** par la nouvelle valeur.

4. Actualisez la nouvelle définition du poste de travail dans le plan. Faites-le immédiatement si vous modifiez le nom d'hôte ou l'adresse IP d'un gestionnaire de domaine maître ou d'un gestionnaire de domaine. Si vous modifiez ces valeurs sur un poste de travail qui n'est pas un gestionnaire de domaine maître ou un gestionnaire de domaine, vous pouvez attendre la prochaine génération de plan planifiée pour actualiser la définition du poste de travail dans le fichier Symphony. Le cas échéant, durant ce jour de production, vous ne pourrez pas exécuter de travaux sur ce poste de travail. Pour générer le plan, procédez comme suit :
   a. Exécutez la commande **optman ls** et notez bien la valeur réelle du paramètre **enCarryForward**.
   b. Si cette valeur n'est pas définie avec **all**, exécutez
      ```
      optman chg cf=ALL
      ```
      pour lui donner la valeur **all**
   c. Ajoutez la nouvelle définition du poste de travail dans le plan, en exécutant la commande suivante :
      ```
      JnextPlan –for 0000
      ```
   d. Réattribuez la valeur d'origine au paramètre **enCarryForward**.

Report de la valeur de nom d'hôte ou d'adresse IP modifiée sur le serveur Dynamic Workload Broker

Le serveur Dynamic Workload Broker est un composant installé par Tivoli Workload Scheduler lorsque vous installez les composants suivants :
- Gestionnaire de domaine maître
- gestionnaire de domaine maître de secours
- gestionnaire de domaine dynamique
- gestionnaire de domaine dynamique de secours

Si vous avez modifié le nom d'hôte ou l'adresse IP sur le serveur Dynamic Workload Broker, ou si vous en avez installé un nouveau, exécutez la procédure décrite dans «Report des modifications dans le fichier de configuration WebSphere Application Server», à la page 401.

Si vous avez modifié le nom d'hôte ou l'adresse IP sur un gestionnaire de domaine maître ou un gestionnaire de domaine maître de secours et que vous avez exécuté la procédure «Report des modifications dans le fichier de configuration WebSphere Application Server», à la page 401, ignorez cette section.

Si vous avez modifié le nom d'hôte ou l'adresse IP sur le gestionnaire de domaine dynamique ou le gestionnaire de domaine dynamique de secours, vous n'avez pas besoin de modifier la définition du poste de travail du courtier (entrez **broker**), car la valeur de l'attribut **node** est définie avec la valeur *localhost* ^pour permettre la permutation entre le serveur Dynamic Workload Broker et son homologue de secours.

Après avoir exécuté la procédure, propagez les modifications sur l'agent dynamique et mettez à jour la propriété **ResourceAdvisorURL** dans le fichier `JobManager.ini` sur chaque agent connecté à ce serveur Dynamic Workload Broker, en procédant comme suit :

1. Exécutez la commande suivante pour arrêter l'agent :
   ```
   ShutDownLwa
   ```

2. Editez le fichier `JobManager.ini` et modifiez le nom d'hôte ou l'adresse IP dans la propriété **ResourceAdvisorURL**.
3. Exécutez la commande suivante pour démarrer l'agent :

   ```
   StartUpLwa
   ```

Procédez aux modifications suivantes :

1. Ouvrez le fichier `JobDispatcherConfig.properties` et modifiez la valeur de la propriété **JDURL=https://*host_name*** en fonction du nouveau nom d'hôte ou de la nouvelle adresse IP.
2. Ouvrez le fichier `CliConfig.properties` et modifiez la valeur de la propriété **ITDWBServerHost=/*host_name*** en fonction du nouveau nom d'hôte ou de la nouvelle adresse IP.
3. Ouvrez le fichier `ResourceAdvisorConfig.properties` et modifiez la valeur de la propriété **ResourceAdvisorURL=https://*host_name*** en fonction du nouveau nom d'hôte ou de la nouvelle adresse IP.
4. Depuis le répertoire *<TWA_home>*`/TDWB/bin`, exécutez la commande suivante :

   Sur les système d'exploitation Windows :

   ```
   exportserverdata.bat
   ```

   Systèmes d'exploitation UNIX et Linux :

   ```
   exportserverdata.sh
   ```

   Cette commande extrait une liste d'URI (Uniform Resource Identifier) de toutes les instances du courtier de charge de travail dynamique de la base de données Tivoli Workload Scheduler et les copie dans un fichier temporaire. Par défaut, la liste des URI est enregistrée dans le fichier `server.properties`, situé dans le répertoire actuel.
5. Modifiez toutes les entrées contenant le nom d'hôte précédent pour refléter le nouveau nom d'hôte.
6. Replacez le fichier dans la base de données, en exécutant la commande suivante :

   Sur les système d'exploitation Windows :

   ```
   importserverdata.bat
   ```

   Systèmes d'exploitation UNIX et Linux :

   ```
   importserverdata.sh
   ```
7. Arrêtez Dynamic Workload Broker en exécutant la commande suivante :

   ```
   StopBrokerApplication
   ```
8. Démarrez Dynamic Workload Broker en exécutant la commande suivante :

   ```
   StartBrokerApplication
   ```

Report de la valeur de nom d'hôte ou d'adresse IP modifiée de l'agent dynamique

Si vous avez modifié le nom d'hôte ou l'adresse IP sur le poste de travail où vous avez installé l'agent dynamique, les modifications sont automatiquement reportées en arrêtant puis en redémarrant l'agent avec les commandes suivantes :

1. Pour arrêter l'agent :

   ```
   ShutDownLwa
   ```
2. Pour démarrer l'agent :

   ```
   StartUpLwa
   ```

Remarque : Ne modifiez pas manuellement la valeur du paramètre **node** dans la définition du poste de travail de l'agent dynamique.

Modification des paramètres de sécurité

Cette section explique comment modifier les paramètres de sécurité de Tivoli Workload Scheduler.

Utilisez le script **changeSecurityProperties** situé dans *TWA_home/TWS/wastool* pour modifier plusieurs paramètres de sécurité sur le serveur d'applications. Pour les paramètres liés à la couche Secure Sockets Layer, voir Chapitre 7, «Définition de la sécurité des connexions», à la page 261. Pour les paramètres associés aux mots de passe des utilisateurs de l'accès à la base de données, voir «Modification des mots de passe Tivoli Workload Scheduler clés», à la page 382. Vous pouvez également modifier d'autres paramètres, tels que le registre d'utilisateurs actifs ou l'ID et le mot de passe du système d'exploitation local.

Pour cette procédure, vous devez arrêter le serveur d'applications, créer un fichier texte contenant les propriétés de sécurité en cours, éditer le fichier, exécuter l'utilitaire et redémarrer le serveur d'applications.

- Pour plus d'informations sur la procédure de modification des propriétés WebSphere Application Server, voir «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423.
- Pour déterminer les propriétés à modifier, consultez les rubriques suivantes :
  - Chapitre 5, «Configuration de l'authentification», à la page 199, pour plus d'informations sur les propriétés à changer afin de modifier la configuration de votre registre d'utilisateurs pour l'authentification d'utilisateur.
  - «Scénario : connexion entre Dynamic Workload Console et le Tivoli Workload Scheduler ayant un connecteur distribué», à la page 262, pour plus d'informations sur les propriétés à changer afin de configurer la communication SSL entre différentes interfaces et le moteur Tivoli Workload Scheduler.
  - «Migration des données de DB2 vers Oracle et *vice versa*», à la page 333, pour plus d'informations sur les propriétés à modifier lors de la migration de votre base de données d'une plateforme de base de données à une autre.
  - «Modification des mots de passe Tivoli Workload Scheduler clés», à la page 382, pour plus d'informations sur l'utilisation des propriétés afin de déterminer la procédure requise pour changer les mots de passe de clés.
- Pour modifier le fichier texte des propriétés de sécurité actuelles, procédez comme suit :
  1. Editez le fichier texte et localisez les propriétés que vous devez modifier.
  2. Apportez toutes les modifications nécessaires aux propriétés.

     Ne modifiez aucune autre propriété.

     Remarque :

     1. Il se peut que l'utilitaire affiche un message provenant du serveur d'applications (WASX7357I:). Vous pouvez ignorer ce message.
     2. Lorsque vous fournissez un mot de passe dans un fichier texte pour **changeSecurityProperties**, il existe un léger risque de sécurité. Lorsque vous saisissez un mot de passe dans le fichier, le mot de passe est entré en clair (non chiffré). Après l'exécution de **changeSecurityProperties**, le mot de passe reste en clair dans le fichier texte que vous avez modifié ; toutefois, si vous exécutez **showSecurityProperties**, la sortie du mot de passe est chiffrée et se

présente sous la forme d'une ligne d'astérisques. Ainsi, le risque de sécurité potentiel est limité à la période qui s'écoule entre le moment où vous avez saisi le mot de passe dans le fichier texte et celui où vous avez manuellement supprimé le fichier texte après avoir exécuté **changeSecurityProperties**.

Avertissement : Si vous voulez modifier des paramètres *autres* qu'un mot de passe *sans* modifier de mot de passe, effectuez l'une des actions suivantes avant d'exécuter **changeSecurityProperties**. Cette manipulation est obligatoire pour éviter que la ligne composée d'astérisques ne s'applique comme mot de passe :

- Reformulez les mots de passe en clair.
- Mettez en commentaire les propriétés de mot de passe.
- Supprimez les propriétés de mot de passe.

Gestion du processeur d'événements

La seule maintenance requise sur le processeur d'événements est la gestion de la file d'attente EIF, `cache.dat`. La file d'attente d'événements est circulaire, les événements étant ajoutés à la fin et supprimés au début. S'il n'y a pas assez de place à la fin de la file d'attente pour écrire un événement, il est écrit au début en écrasant un événement du début de la file d'attente.

Pour augmenter la capacité de la file d'attente d'un processeur d'événements, procédez comme suit :

1. Sur le poste de travail qui exécute le processeur d'événements, localisez le fichier

   `<chemin_profil_WAS>/temp/TWS/EIFListener/eif.templ`

   où *chemin_profil_WAS* correspond au chemin du profil WebSphere Application Server que vous avez spécifié au moment de l'installation. Le chemin par défaut est `TWA_home/WAS/TWSprofile`.
2. Editez le fichier et localisez le mot clé

   `BufEvtMaxSize`
3. Augmentez la valeur de ce mot de passe en fonction de vos besoins.
4. Arrêtez et redémarrez la WebSphere Application Server à l'aide des commandes **conman stopappserver** et **conman startappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413).

Démarrage, arrêt et affichage du statut de Dynamic Workload Broker

Pour démarrer ou arrêter Dynamic Workload Broker, utilisez la commande **startBrokerApplication** ou **stopBrokerApplication** sur le gestionnaire de domaine maître actif. Ces commandes sont traitées de façon asynchrone ; la commande **brokerApplicationStatus** vous permet donc de vérifier le statut de Dynamic Workload Broker après un arrêt ou un démarrage. Vérifiez que WebSphere Application Server s'exécute, puis procédez comme suit :

Démarrage de Dynamic Workload Broker

Utilisez **startBrokerApplication.sh** sous UNIX et Linux ou **startBrokerApplication.bat** sous Windows, comme suit :

startBrokerApplication -user *username* -password *password* [-port *portnumber*]

où *username* et *password* correspondent aux données d'identification utilisées lors de l'installation. Le paramètre *portnumber* est facultatif ; cette

valeur est définie dans la propriété `Broker.Workstation.Port` sous `/opt/IBM/TWA92_SVT/TDWB/config/BrokerWorkstation.properties`. La valeur par défaut est 31 114.

Arrêt de Dynamic Workload Broker

1. Utilisez **stopBrokerApplication.sh** sous UNIX et Linux ou **stopBrokerApplication.bat** sous Windows, comme suit :

   stopBrokerApplication -user *username* -password *password* [-port *portnumber*]

   où *username* et *password* correspondent aux données d'identification utilisées lors de l'installation. Le paramètre *portnumber* est facultatif. S'il n'est pas défini, le numéro de port par défaut est utilisé.
2. Exécutez la commande **link**. Si vous n'exécutez pas cette commande, le serveur Dynamic Workload Broker est automatiquement lié dix minutes après l'opération d'arrêt.

Affichage du statut de Dynamic Workload Broker

Utilisez **brokerApplicationStatus.sh** sous UNIX et Linux ou **brokerApplicationStatus.bat** sous Windows, comme suit :

brokerApplicationStatus -user *username* -password *password* [-port *portnumber*]

où *username* et *password* correspondent aux données d'identification utilisées lors de l'installation. Le paramètre *portnumber* est facultatif. S'il n'est pas défini, le numéro de port par défaut est utilisé.

Initialisation automatique des instances Tivoli Workload Scheduler

Sur les systèmes UNIX, vous pouvez vous assurer que vos instances de Tivoli Workload Scheduler sont automatiquement initialisées au démarrage du système d'exploitation en ajoutant un service Tivoli Workload Scheduler au processus init de votre système d'exploitation. Pour cela, vous pouvez utiliser l'exemple de script de démarrage `twa_initd` figurant dans le répertoire *TWA_home*/config et l'ajouter au niveau d'exécution approprié après l'avoir personnalisé selon les besoins.

Procédez comme suit :

1. Copiez, renommez et éditez le script `twa_initd` en fonction de vos besoins. Fournissez les informations suivantes, en fonction du système d'exploitation que vous utilisez :

   Required-Start
   Sur les systèmes Linux, spécifiez les services de précondition

   Default-Start
   Sur les systèmes Linux, spécifiez les niveaux d'exécution (runlevels) requis. Par exemple runlevels 2, 3 et 5.

   RACINE_TWS
   Sur tous les systèmes UNIX pris en charge, spécifiez le chemin complet qualifié de l'instance de Tivoli Workload Scheduler que vous souhaitez démarrer.
2. Sauvegardez le fichier édité dans le dossier approprié en fonction du système, comme suit :

   Systèmes d'exploitation Linux pris en charge
   Sauvegardez le script dans le dossier /etc/init.d et enregistrer le service à l'aide de la commande **insserv -v *script_name***.

Systèmes d'exploitation AIX pris en charge
Copiez le script dans le répertoire rc*runlevel*.d approprié. Renommez le script en fonction de la définition du script de niveau d'exécution. Par exemple, S*numéro_séquencenom_service*, comme par exemple S10tws860ma.

Systèmes d'exploitation Solaris pris en charge
Sauvegardez le script dans le dossier /etc/init.d et associez-le à un fichier approprié dans le dossier rc*runlevel*.d. Par exemple, S*numéro_séquencenom_service*, comme par exemple S10tws860ma.

Systèmes d'exploitation HP-UX pris en charge
Sauvegardez le script dans le dossier /sbin/init.d et associez-le à un fichier approprié dans le dossier rc*runlevel*.d. Par exemple, S*numéro_séquencenom_service*, comme par exemple S10tws860ma.

Pour plus d'informations sur les commandes **inittab, init.d, insserv** et **init**, voir la documentation de référence de votre système d'exploitation.

L'exemple suivant présente un script personnalisé :

```
#!/bin/sh
################################################################
# Eléments sous licence - Propriété d'IBM
# Restricted Materials of IBM
# 5698-WSH
# (C) Copyright IBM Corp. 1998, 2011 All Rights Reserved.
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
################################################################

### BEGIN INIT INFO
# Provides:       tws860ma
# Required-Start: network
# Default-Start:  2 3 5
# Description:    TWS service
### END INIT INFO

# Specify the fully qualified path name of the TWS Installation to start
#
TWS_HOME=/opt/IBM/TWA/TWS

TWS_START_SCRIPT=${TWS_HOME}/StartUp

if [ -f ${TWS_START_SCRIPT} ]
then
  case "$1" in
    start)
      echo -n "Starting Tivoli Workload Scheduler instance"
      ${TWS_START_SCRIPT}
      exit $?
    ;;
  *)
    echo "Usage: $0 {start}"
    exit 1
    ;;
  esac
else
  exit 5
fi
```

Tâches du serveur d'applications

Il vous faudra peut-être effectuer les tâches de serveur d'applications suivantes :

Serveur d'applications - démarrage et arrêt

Utilisez les commandes **startappserver** et **stopappserver** ou les commandes équivalentes de Dynamic Workload Console pour démarrer ou arrêter la WebSphere Application Server (voir *Tivoli Workload Scheduler - Guide d'utilisation et de référence* pour une description de ces commandes.)

Ces commandes permettent également d'arrêter **appservman**, le service qui surveille et éventuellement redémarre le serveur d'applications.

Si vous ne voulez pas arrêter **appservman**, vous pouvez émettre **startWas** ou **stopWas**, en indiquant l'argument **–direct**.

La syntaxe complète de **startWas** et **stopWas** est la suivante :

UNIX

Démarrage du serveur d'applications

```
startWas.sh  [-direct]
```

Arrêt du serveur d'applications

```
stopWas.sh  [-direct]
          -user <user_ID>
           -password <password>
```

Remarque : La syntaxe ci-dessus permettant d'arrêter WebSphere Application Server peut s'appliquer à un gestionnaire de domaine maître WebSphere Application Server. Si votre serveur WebSphere Application Server provient de Dynamic Workload Console, vous devez utiliser la syntaxe suivante :

```
stopWas.sh -direct
         -user <user_ID>
          -password <password>
```

où l'argument **-direct** est obligatoire.

Windows

Démarrage du serveur d'applications

```
startWas.bat [-direct]
             [-service <nom_service>]
             [-options <paramètres>]
```

Arrêt du serveur d'applications

```
stopWas.bat [-direct]
            [-service <nom_service>]
            [-wasHome <installation_directory>]
            [-options <paramètres>]
```

où les arguments sont les suivants :

–direct

Démarre ou arrête de façon facultative le serveur d'applications, sans démarrer ou arrêter la surveillance du serveur d'applications **appservman**.

Par exemple, vous pouvez utiliser cet argument après avoir modifié certains paramètres de configuration. En arrêtant WebSphere Application Server sans mettre fin à **appservmann**, ce dernier redémarre immédiatement WebSphere Application Server, à l'aide des nouvelles propriétés de configuration. Cet argument est obligatoire sous UNIX lorsque les composants produit ne sont pas intégrés.

–options <*paramètres*>
Fournit des paramètres en option aux commandes WebSphere Application Server **startServer** ou **stopServer**. Pour plus de détails, voir la documentation WebSphere Application Server.

–password <*password*>
Définit le mot de passe à utiliser lors de l'arrêt du serveur d'applications sous UNIX.

–service <*nom_service*>
Définit le nom du service WebSphere Application Server, s'il ne s'agit pas de la valeur par défaut de `IBM WebSphere Application Server V6 - <utilisateur_TWS>`

–user <*user_ID*>
Définit l'ID d'utilisateur à utiliser pour arrêter le serveur d'applications sous UNIX.

–wasHome <*installation_directory*>
Définit le répertoire d'installation de WebSphere Application Server, s'il ne s'agit pas de la valeur par défaut.

Serveur d'applications - redémarrage automatique après incident

En cas d'incident sur le serveur d'applications, un service est proposé pour surveiller son état et le démarrer automatiquement. Nommé **appservman**, ce service est activé et contrôlé par les options locales à l'ordinateur sur lequel le serveur d'applications est exécuté.

Les sections suivantes décrivent le service, son fonctionnement et la façon dont il est contrôlé :

Appservman - Fonctionnement

Appservman est un service qui démarre, arrête et surveille le serveur d'applications. Il peut également le redémarrer en cas d'incident. **Appservman** peut être contrôlé depuis les noeuds qui exécutent le serveur d'applications, mais également depuis tout autre noeud qui exécute **conman**.

Il est lancé en tant que service par **netman** lors du démarrage de Tivoli Workload Scheduler, et il démarre ensuite le serveur d'applications. **Netman** le lance également lorsque la commande **conman startappserver** est exécutée.

Appservman est arrêté à la fermeture de Tivoli Workload Scheduler. **Netman** arrête également le serveur d'applications et **appservman** lorsque vous utilisez la commande **conman stopappserver** ou, sous Windows uniquement, lorsque vous exécutez la commande **Shutdown –appsrv**.

Lorsqu'il est exécuté, **appservman** surveille la disponibilité du serveur d'applications, et envoie des événements qui décrivent l'état du serveur

d'applications. Si la fonction de redémarrage automatique est activée et que le serveur d'applications échoue, le service détermine à partir de la règle de redémarrage indiquée dans les options `localopts` si le serveur d'applications doit être redémarré ou non. Si la règle l'autorise, le service redémarre le serveur d'applications et envoie des événements pour signaler ses actions.

Par défaut, les utilitaires **startWas** et **stopWas** de WebSphere Application Server démarrent et arrêtent le serveur d'applications et **appservman** à l'aide des commandes **startappserver** et **stopappserver**. Toutefois, ces utilitaires peuvent être configurés pour démarrer et arrêter le serveur d'applications sans arrêter **appservman** à l'aide des utilitaires **startWas** et **stopWas** paramétrés avec l'option **–direct**.

Contrôle de appservman

Appservman est contrôlé par les options locales suivantes (dans le fichier `localopts`) :

Appserver auto restart
Détermine si la fonction de redémarrage automatique est activée.

La valeur par défaut est *yes*. Pour désactiver l'option, définissez-la sur *no*.

Appserver check interval
Détermine la fréquence à laquelle le service vérifie le statut du serveur d'applications. Vous ne devez pas définir cette valeur sur une durée inférieure au temps généralement requis pour démarrer le serveur d'applications sur l'ordinateur.

La valeur par défaut est 5 minutes.

Appserver min restart time
Détermine le délai minimum requis entre deux incidents du serveur d'applications pour que le redémarrage automatique fonctionne. Cette option empêche **appservman** de redémarrer immédiatement le serveur d'applications s'il échoue au lancement initial ou lors d'un redémarrage.

La valeur par défaut est 10 minutes.

Appserver max restarts
Détermine le nombre maximum de redémarrages automatiques du serveur d'applications par **appservman** au cours de l'intervalle que vous avez défini (Appserver count reset interval).

La valeur par défaut est de 5 redémarrages.

Appserver count reset interval
Détermine l'intervalle au cours duquel le nombre maximum de redémarrages est effectué (Appserver max restarts).

La valeur par défaut est 24 heures.

Appserver service name
Cette option n'est utilisée que sous Windows. Elle est générée de la façon suivante :

`IBMWAS61Service - <utilisateur_TWS>`

Utilisation des options : Les paramètres par défaut constituent un bon point de départ. Suivez les indications ci-dessous si vous n'êtes pas satisfait de la disponibilité du serveur d'applications :

- Si le serveur d'applications ne redémarre pas après un incident, vérifiez les points suivants :

– *Appserver auto restart* est défini sur *yes*.
– *Appserver check interval* n'est pas défini sur une valeur trop élevée. Si cette valeur est définie sur *50* minutes au lieu des *5* minutes par défaut, le serveur d'applications peut attendre 45 minutes avant d'être redémarré suite à un incident survenu très tôt.
– *Appserver min restart time* est suffisant pour permettre au serveur d'applications de redémarrer complètement. Si, lorsque le serveur vérifie l'état du serveur d'applications, il trouve que celui-ci est toujours en train de démarrer, il arrive qu'il ne soit pas capable de faire la distinction entre un état de démarrage et un état d'échec, qu'il l'indique comme ayant échoué et tente de le redémarrer. Le résultat est alors le même. Ce phénomène se reproduit jusqu'à ce que *Appserver max restarts* soit dépassé. Si vous êtes dans ce cas, augmentez *Appserver min restart time*.

- Si le serveur d'applications échoue rarement mais ne redémarre pas suite à plusieurs échecs, définissez l'option *Appserver max restarts* sur une valeur supérieure, ou *Appserver count reset interval* sur une valeur inférieure, ou les deux. Dans ce cas, il peut être intéressant d'étudier le schéma des incidents et d'adapter ces options de façon à obtenir la disponibilité souhaitée.

Démarrage et arrêt du serveur d'applications et de appservman

Si vous devez arrêter et redémarrer le serveur d'applications, par exemple pour effectuer une modification dans la configuration du serveur d'applications, utilisez les commandes suivantes :

stopappserver[*domain***!]***workstation* **[;wait]**
Cette commande arrête le serveur d'applications et **appservman**. Vous pouvez arrêter le serveur d'applications sur un poste de travail distant. Le paramètre **;wait** en option indique à **conman** d'interrompre le traitement jusqu'à ce que la commande indique que le serveur d'applications et le serveur ont été arrêtés.

startappserver[*domain***!]***workstation* **[;wait]**
Cette commande démarre le serveur d'applications et **appservman**. Vous pouvez démarrer le serveur d'applications sur un poste de travail distant. Le paramètre **;wait** en option indique à **conman** d'interrompre le traitement jusqu'à ce que la commande indique que le serveur d'applications et le serveur sont en fonctionnement.

Pour arrêter et démarrer le serveur d'applications sans arrêter **appservman**, voir «Serveur d'applications - démarrage et arrêt», à la page 410.

Configuration de l'utilisateur et du mot de passe pour exécuter la commande conman stopappserver

Lorsque vous exécutez la commande **conman stopappserver**, le processus `appserverman` commence par vérifier si WebSphere Application Server parvient à extraire les données d'identification de l'utilisateur (nom d'utilisateur et mot de passe) à partir du fichier `soap.client.props` figurant dans le profil WebSphere Application Server. Si la vérification est négative, `appserverman` lit ces données à partir du fichier `useropts` de l'utilisateur et exécute le script `stopServer.sh (bat)` pour les transférer sur WebSphere Application Server.

Pour pouvoir exécuter la commande **conman stopappserver**, vous devez suivre l'une des deux procédures de personnalisation suivantes afin de fournir les données d'identification de l'utilisateur à WebSphere Application Server :

- Personnalisez les propriétés de nom d'utilisateur (com.ibm.SOAP.loginUserid) et de mot de passe (com.ibm.SOAP.loginPassword) du fichier soap.client.props figurant à l'emplacement suivant :

chemin_profil_WAS/properties (maître et agents de version 9.1 et ultérieurs)

où *chemin_profil_WAS* correspond au chemin du profil WebSphere Application Server que vous avez spécifié au moment de l'installation. La valeur par défaut de ce chemin est : *TWA_home*/WAS/TWSprofile.

Vous devez aussi :

1. Définir la propriété com.ibm.SOAP.securityEnabled avec la valeur true dans le même fichier pour activer la sécurité du client SOAP
2. Exécuter le script encryptProfileProperties.sh pour chiffrer le mot de passe. Reportez-vous au Tivoli Workload Scheduler *Guide d'administration* pour plus d'informations sur cet outil de serveur d'applications.

- Personnaliser la section Attributes for conman (CLI in version 8.4) connections (Attributs des connexions conman (CLI dans la version 8.4)) dans le fichier localopts en spécifiant les détails du connecteur ou du gestionnaire de domaine maître.

  Vous devez aussi :

  1. Créer (ou personnaliser s'il existe déjà) le fichier useropts manuellement, en ajoutant les attributs USERNAME et PASSWORD de l'utilisateur qui exécutera la commande **stopappserver**. Vérifiez que le nom du fichier useropts est entré dans la clé USEROPTS dans la section Attributes for conman (CLI) connections. Pour plus de détails, voir *Guide d'administration*.
  2. Chiffrer le mot de passe dans le fichier useropts en exécutant simplement la commande **conman**.

Surveillance de l'état du serveur d'applications

Pour voir à tout moment l'état en cours du serveur d'applications, consultez la zone STATE dans les détails du poste de travail.

Cette zone contient une chaîne de caractères apportant des informations sur le statut des objets et processus du poste de travail. L'état du serveur d'applications est un indicateur à un caractère dans cette chaîne, avec l'une des valeurs suivantes si le serveur d'applications est installé :

[A|R]

Où :

A Le serveur WebSphere Application Server est en cours d'exécution.

R Le serveur WebSphere Application Server est en cours de redémarrage.

Si le serveur d'applications est en panne ou s'il n'a pas été installé, aucune de ces valeurs n'est présente dans l'entrée STATE.

Obtention d'informations sur les incidents du serveur d'applications

Appservman n'explique pas pourquoi le serveur d'applications a échoué. Pour obtenir cette information, consultez les fichiers journaux du serveur d'applications (voir *Tivoli Workload Scheduler - Guide d'identification et de résolution des problèmes*).

Evénements créés par appservman

Appservman envoie un événement nommé *ApplicationServerStatusChanged* depuis le fournisseur TWSObjectsMonitor vers le processus configuré de surveillance des événements, à chaque fois que l'état du serveur d'applications change.

Serveur d'applications - chiffrement des fichiers de propriétés du profil

Utilisez le script **encryptProfileProperties** pour chiffrer les mots de passe des fichiers de propriétés du serveur WebSphere Application Server suivants :

- <*chemin_profil_WAS*>/properties/soap.client.props
- <*chemin_profil_WAS*>/properties/sas.client.props
- <*chemin_profil_WAS*>/properties/sas.stdclient.properties
- <*chemin_profil_WAS*>/properties/sas.tools.properties

où *chemin_profil_WAS* correspond au chemin du profil WebSphere Application Server que vous avez spécifié en installant l'un des composants suivants : gestionnaire de domaine maître, gestionnaire de domaine maître de sauvegarde, gestionnaire de domaine dynamique, gestionnaires de domaine dynamique de sauvegarde. Le chemin par défaut est : *TWA_home*/WAS/TWSprofile.

La création de fichiers de clés SSL est un exemple de cas où vous pouvez utiliser la fonction de chiffrement. Vous entrez les mots de passe dans les fichiers de clés, puis vous les chiffrez à l'aide du script **encryptProfileProperties**. Voir *Tivoli Workload Scheduler - Guide de planification et d'installation*.

Le script utilise **PropFilePasswordEncoder.bat**. Redémarrez le serveur pour appliquer les modifications.

La commande de chiffrement des propriétés utilise la syntaxe suivante : «Serveur d'applications - chiffrement des fichiers de propriétés du profil»

```
encryptProfileProperties.bat (.sh)
```

Serveur d'applications - mise à jour des services Windows après des modifications

Si vous avez modifié l'un des éléments suivants, vous devez mettre à jour le service Windows qui exécute le serveur d'applications :

- L'ID utilisateur et le mot de passe de l'utilisateur du système d'exploitation local qui exécute le processus du serveur d'applications
- Le répertoire d'installation de la WebSphere Application Server
- Le répertoire dans lequel le profil du serveur d'applications Tivoli Workload Scheduler est enregistré

Pour mettre à jour le service, exécutez la commande **updateWasService** à partir du répertoire <*TWA_home*>/wastools.

Remarque : Cette commande permet également de modifier le mode de démarrage du serveur d'applications.

Au moment de l'exécution, le script appelle **WASService.exe**.

updateWasService
Format

updateWasService –userid <*utilisateur_TWS*> –password <*password_utilisateur_TWS*>
[–wasuser <utilisateur_WAS> –waspassword <password_utilisateur_WAS>]
[–startType {automatic | manual | disabled}]
[–wasHome <installation_directory_WebSphere>]
[–profilePath <répertoire_profil_serveur>]

Paramètres

–userid <*utilisateur_TWS*> –password <*password_utilisateur_TWS*>
Fournissez l'<*utilisateur_TWS*> et son mot de passe.

[–wasuser <utilisateur_WAS> –waspassword <password_utilisateur_WAS>]
L'utilisateur que le système d'exploitation local utilise pour exécuter le processus d'application st défini par défaut sur l'<*utilisateur_TWS*>. Si vous voulez le changer et attribuer un utilisateur et un mot de passe différents, spécifiez ce paramètre.

Remarque : En raison d'un problème connu avec cet utilitaire, lorsque vous modifiez le mot de passe, vous devez d'abord utiliser la fonction Windows de modification du mot de passe, telle que décrite dans «Action 7 - Windows - mise à jour des services Windows», à la page 389, puis exécutez cet utilitaire, en fournissant le nouveau mot de passe que vous venez de définir en tant que <*password_utilisateur_WAS*>.

[–startType {automatic | manual | disabled}]
Le serveur d'applications démarre automatiquement par défaut, au démarrage de l'ordinateur. Si vous voulez qu'il en soit autrement, spécifiez ce paramètre.

[–wasHome <installation_directory_WebSphere>]
Si vous avez modifié le nom du répertoire d'installation de la WebSphere Application Server, spécifiez ce paramètre en indiquant le nouveau nom.

[–profilePath <répertoire_profil_serveur>]
Si vous avez modifié le nom du répertoire dans lequel le profil du serveur d'applications Tivoli Workload Scheduler est enregistré, spécifiez ce paramètre en indiquant le nouveau nom.

Serveur d'applications - mise à jour des propriétés SOAP après modification de l'utilisateur WebSphere Application Server ou de son mot de passe

Si vous avez modifié l'ID utilisateur ou le mot de passe de l'utilisateur d'administration WebSphere Application Server pour Tivoli Workload Scheduler ou Dynamic Workload Console, vous devez également mettre à jour les propriétés du client SOAP.

Pour mettre à jour les propriétés, exécutez la commande **updateWas.sh/.bat** à partir du répertoire <*TWA_home*>/wastools.

Après l'utilisation de cette commande, vous devez redémarrer le serveur d'applications.

updateWas.sh
Format

updateWas.sh –user <*nouvel_utilisateur_admin_WAS*> -password <*password*>

Paramètres

–user <*nouvel_utilisateur_admin_WAS*> **–password** <*password*>
Indiquez l'utilisateur et le mot de passe du nouvel utilisateur d'administration WebSphere Application Server que vous voulez configurer comme justificatifs dans les propriétés du client SOAP.

Serveur d'applications - sauvegarde et restauration des fichiers de configuration

Le serveur d'applications dispose de fichiers de configuration, qui doivent être sauvegardés après toute modification. Utilisez le script **backupConfig** dans le répertoire <*TWA_home*> /wastools pour Tivoli Workload Scheduler ou dans <*TDWC_INSTALL_PATH*> /wastools pour Dynamic Workload Console.

Le cas échéant, les fichiers peuvent être restaurés à l'aide du script **restoreConfig** qui se trouve dans le même répertoire.

Il n'est pas nécessaire d'arrêter le serveur d'applications pour procéder à la sauvegarde, par contre vous devez l'arrêter et le redémarrer si vous avez restauré les fichiers d'une sauvegarde précédente.

Pour plus d'informations, voir *IBM Redbooks : WebSphere Application Server - Manuel de gestion et de configuration du système V6*.

Syntaxe de la commande de sauvegarde

La commande de sauvegarde utilise la syntaxe suivante :

```
backupConfig.bat [fichier_sauvegarde]
                 [-nostop]
                 [-quiet]
                 [-logfile file_name]
                 [-replacelog]
                 [-trace]
                 [-username user_ID]
                 [-password password]
                 [-profileName profil]
                 [-help]
```

où,

fichier_sauvegarde **de Dynamic Workload Console**
est le fichier (nom complet et chemin) utilisé pour sauvegarder la configuration du profil JazzSM. Si *fichier_sauvegarde* n'est pas spécifié, la sortie est signalée par défaut dans un fichier compressé comme suit : *chemin_installation_DynamicWorkloadConsole*/TDWC/backup/WebSphereConfig_backup.zip

fichier_sauvegarde **pour Tivoli Workload Scheduler**
est le fichier (nom complet et chemin) utilisé pour sauvegarder la configuration du profil JazzSM. Si *fichier_sauvegarde* n'est pas spécifié, la commande ne s'exécute pas.

Voici un exemple d'exécution du fichier **backupconfig.bat** :

```
C:\Program Files\ibm\TWA0\wastools>backupConfig.bat
ADMU0116I: Les informations sur les outils sont consignées dans le fichier
           C:\Program Files\ibm\TWA0\WAS\TWSprofile\logs\
           backupConfig.log
ADMU0128I: Démarrage de l'outil avec le profil twsprofile
ADMU5001I: Sauvegarde du répertoire de configuration dans le fichier
           C:\Program Files\ibm\TWA0\WAS\TWSprofile\config to file
           C:\Program Files\ibm\TWA0\wastools\WebSphereConfig_2005-12-12.zip
ADMU0505I: Serveurs trouvés dans la configuration :
ADMU0506I: Nom du serveur : server1
ADMU2010I: Arrêt de tous les processus serveurs pour le noeud DefaultNode
ADMU0512I: Le serveur server1 ne peut pas être joint. Il semble être arrêté.
.................................................................
ADMU5002I: 137 fichiers sauvegardés avec succès.
```

Syntaxe de la commande de restauration

La commande de restauration utilise la syntaxe suivante à partir du répertoire *profile_root*/bin :

```
restoreConfig.bat [fichier_sauvegarde]
                  [-location emplacement_restauration]
                  [-quiet]
                  [-nowait]
                  [-logfile file_name]
                  [-replacelog]
                  [-trace]
                  [-username user_ID]
                  [-password password]
                  [-profileName profil]
                  [-help]
```

où,

fichier_sauvegarde **de Dynamic Workload Console**
est le fichier (nom complet et chemin) utilisé pour restaurer la configuration du profil JazzSM. Si *fichier_sauvegarde* n'est pas spécifié, le fichier de sauvegarde nommé *DynamicWorkloadConsole_installpath*/TDWC/backup/WebSphereConfig_backup.zip est utilisé s'il existe.

fichier_sauvegarde **pour Tivoli Workload Scheduler**
est le fichier (nom complet et chemin) utilisé pour restaurer la configuration du profil JazzSM. Si *fichier_sauvegarde* n'est pas spécifié, la commande ne s'exécute pas.

Voici un exemple d'exécution de la commande **restoreConfig.bat** :

```
C:\Program Files\ibm\TWA0\wastools>restoreConfig.bat WebSphereConfig_2005-12-11.zip
ADMU0116I: Les informations sur les outils sont consignées dans le fichier
           C:\Program Files\ibm\TWA0\WAS\TWSprofile\logs\
           restoreConfig.log
ADMU0128I: Démarrage de l'outil avec le profil twsprofile
ADMU0505I: Serveurs trouvés dans la configuration :
ADMU0506I: Nom du serveur : server1
ADMU2010I: Arrêt de tous les processus serveurs pour le noeud DefaultNode
ADMU0512I: Le serveur server1 ne peut pas être joint. Il semble être arrêté.
ADMU5502I: Le répertoire C:\Program Files\ibm\TWA0\WAS\TWSprofile\config
           existe déjà ; celui-ci est renommé en
           C:\Program Files\ibm\TWA0\WAS\TWSprofile\config.old
ADMU5504I: L'emplacement de restauration a été renommé avec succès
ADMU5505I: Restauration du fichier WebSphereConfig_2005-12-11.zip à l'emplacement
           C:\Program Files\ibm\TWA0\WAS\TWSprofile\config
.................................................................
ADMU5506I: 127 fichiers ont été restaurés avec succès
ADMU6001I: Début de la préparation de l'application -
ADMU6009I: Traitement terminé.
```

Serveur d'applications - modification du nom d'hôte ou des ports TCP/IP

Pour modifier le nom d'hôte de l'ordinateur sur lequel le serveur d'applications est installé ou les ports TCP/IP qu'il utilise, exécutez le script **changeHostProperties**.

Pour cette procédure, vous devez arrêter le serveur d'applications, créer un fichier texte contenant les propriétés hôtes en cours, éditer le fichier, exécuter l'utilitaire et redémarrer le serveur d'applications. Pour savoir comment procéder, consultez les informations suivantes :

- La section «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423 fournit une description générale de la procédure de modification des propriétés du WebSphere Application Server
- La section «Modification des propriétés d'un hôte», à la page 420 dresse la liste de toutes les propriétés hôtes et fournit d'autres informations de référence sur l'utilitaire
- Lorsque vous éditez le fichier texte des propriétés hôtes en cours, procédez comme suit :
  1. Editez le fichier texte et localisez les propriétés suivantes :

```
################################################################
# Host Configuration Panel
################################################################

# Old Hostname
oldHostname=myoldhost.mydomain.com

# New Hostname
newHostname=myhost.mydomain.com

################################################################
Ports Configuration Panel
################################################################
bootPort=41117
bootHost=myhost.mydomain.com
soapPort=41118
soapHost=myhost.mydomain.com
httpPort=41115
httpHost=*
httpsPort=41116
httpsHost=*
adminPort=41123
adminHost=*
adminSecurePort=41124
adminSecureHost=*
sasPort=41119
sasHost=myhost.mydomain.com
csiServerAuthPort=41120
csiServerAuthHost=myhost.mydomain.com
csiMuthualAuthPort=41121
csiMuthualAuthHost=myhost.mydomain.com
orbPort=41122
orbHost=myhost.mydomain.com
```

     Les règles de modification de ces valeurs sont les suivantes :

     – Pour modifier le nom d'hôte, indiquez l'ancien (`oldHostname`) et le nouveau (`newHostname`). Vérifiez également que la valeur de `bootHost` et de `csiServerAuthHost` est correctement définie (normalement sur le nouveau nom d'hôte).
     – Si vous modifiez le nom d'hôte, ce sont les paramètres de l'ancien port hôte qui sont utilisés, sauf si vous les modifiez expressément.
     – Si vous n'indiquez aucun numéro de port, il reste inchangé.

Ne modifiez aucune autre propriété.

Remarque : Il se peut que l'utilitaire affiche un message provenant du serveur d'applications (WASX7357I:). Vous pouvez ignorer ce message.

Modification des propriétés d'un hôte

Vous pouvez utiliser le script **changeHostProperties** pour modifier le nom d'hôte du poste de travail dans les fichiers de configuration WebSphere Application Server ou les ports TCP/IP utilisés par WebSphere Application Server. Pour modifier ou désactiver les ports TCP/IP, voir «Désactivation des ports TCP/IP», à la page 421.

La procédure d'exécution du script est décrite en détails dans «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423, mais en résumé, vous effectuez les opérations suivantes :

- Exécutez **showHostProperties.sh (.bat) > my_file_name** pour obtenir les propriétés actuelles
- Modifiez *mon_fichier*
- Exécutez **changeHostProperties.sh (.bat) my_file_name**

Remarque : *my_file_name* doit être le chemin d'accès qualifié complet du fichier.

L'utilitaire de modification appelle l'utilitaire **wsadmin** en exécutant **ChangeHostProperties.jacl** avec le fichier de propriétés comme entrée.

Seul le fichier de configuration WebSphere Application Server `serverindex.xml` est concerné par ce script. Le chemin du fichier correspond à :

chemin_profil_WAS`/config/cells/TWSNodeCell/nodes/TWSNode/servers/server1/server.xml`

où la valeur par défaut du chemin <*chemin_profil_WAS*> est `<TWA_home>/WAS/TWSprofile` pour le gestionnaire de domaine maître, et <*rép_profil_JazzSM*> pour le connecteur z/OS et Dynamic Workload Console, où la valeur par défaut de <*rép_profil_JazzSM*> est `/opt/IBM/JazzSM/profile`.

Voici une liste des propriétés modifiables à l'aide de cet utilitaire :

```
#############################################################
# Host Configuration Panel
#############################################################

# Old Hostname
oldHostname=myoldhost.romelab.ibm.it.com

# New Hostname
newHostname=mynewhost.romelab.ibm.it.com

#############################################################
Ports Configuration Panel
#############################################################
bootPort=41117
bootHost=myhost.mydomain.com
soapPort=41118
soapHost=myhost.mydomain.com
httpPort=41115
httpHost=*
httpsPort=41116
httpsHost=*
adminPort=41123
adminHost=*
adminSecurePort=41124
```

```
adminSecureHost=*
sasPort=41119
sasHost=myhost.mydomain.com
csiServerAuthPort=41120
csiServerAuthHost=myhost.mydomain.com
csiMuthualAuthPort=41121
csiMuthualAuthHost=myhost.mydomain.com
orbPort=41122
orbHost=myhost.mydomain.com
```

Toutes les propriétés du fichier de propriétés sont facultatives. Lorsque vous modifiez le fichier de propriétés, vous devez prendre en compte les éléments suivants :

- Lorsque vous définissez les paramètres **oldHostname** et **newHostname** (ces paramètres doivent être indiqués ensemble), la propriété host de chaque port est mise à jour si elle n'a pas été indiquée dans le fichier de propriétés et que la valeur en cours correspond à **oldHostname**.
- Les paramètres du port sont mis à jour uniquement s'ils sont indiqués dans le fichier de propriétés.
- Un paramètre d'hôte propre à un port, tel que **httpHost**, n'est pas mis à jour s'il n'est pas indiqué, sauf lorsque son paramètre en cours correspond à **oldHostname**.
- Le système considère qu'un paramètre vide n'est pas défini.

Désactivation des ports TCP/IP : A l'aide du script **changeHostProperties**, vous pouvez également désactiver certains ports TCP/IP en définissant la valeur des propriétés correspondantes suivantes sur **false**. Si vous utilisez une communication SSL sur votre réseau, le fait de désactiver les ports de la console d'administration et du HTTP non sécurisé garantit que seules les communications chiffrées sont établies sur votre réseau. Ces ports sont tous activés par défaut. Pour les désactiver, utilisez les propriétés suivantes :

httpEnabled
Pour désactiver le port `httpPort`.

httpsEnabled
Pour désactiver le port `httpsPort`.

adminEnabled
Pour désactiver le port `adminPort`.

adminSecureEnabled
Pour désactiver le port `adminSecurePort`.

Affichage des propriétés de l'hôte en cours : Pour afficher les propriétés en cours, utilisez l'utilitaire suivant :

UNIX **showHostProperties.sh**

Windows
showHostProperties.bat

Serveur d'applications - modification des propriétés de trace

Vous pouvez utiliser le script **changeTraceProperties** pour modifier les propriétés de trace du serveur WebSphere Application Server.

Le script appelle l'utilitaire **wsadmin** en exécutant **ChangeServerTracing.jacl** avec le fichier de propriétés dont le modèle est `TracingProps.properties`.

Reportez-vous au document *Tivoli Workload Scheduler - Guide d'identification des problèmes* pour plus de détails sur la modification des paramètres de trace.

Le fichier de propriétés définit les modes de trace suivants :

```
wsmm_odr=com.ibm.ws.xd.comm.*=all:com.ibm.wsmm.grm.Controller=
all:com.ibm.ws.xd.work*
=all:com.ibm.ws.xd.arfm.*=all:com.ibm.wsmm.policing.*
=all:com.ibm.wsmm.xdglue.*
=all:com.ibm.ws.odc.ODCTreeImpl$Save=all
wsmm_node=com.ibm.ws.xd.comm.*=all:com.ibm.ws.xd.placement*
=all:com.ibm.ws.xd.arfm.*=all
reset=*=info
owsmm007=com.ibm.wsmm.policing.*=all
dcs=DCS=finest:RMM=finest
ham=hamanageditem=all
tcpdcs=DCS=finest:RMM=finest:com.ibm.ws.tcp.channel.*=finest
tcp=com.ibm.ws.tcp.channel.*=finest
vizcache=com.ibm.ws.xd.visualizationengine.cacheservice.cacheimpl.*=all
runtime=com.ibm.ws.console.xdruntime.*=all
proxy=com.ibm.ws.console.proxy.*=all
placement=com.ibm.ws.xd.placement*=all=enabled
charting=com.ibm.ws.console.chart.*=all
dwlm=com.ibm.ws.dwlm.*=all
operationalpolicy=com.ibm.ws.xd.operationalpolicymonitor.*=all
wsmm_na=*=info:com.ibm.ws.xd.comm.*=all:com.ibm.ws.xd.placement*
=all:com.ibm.ws.xd.workprofiler.*=all:com.ibm.ws.xd.arfm.*=
all:com.ibm.ws.dwlm.*
=all:com.ibm.ws.xd.hmm.*=all:com.ibm.ws.xd.admin.utils.*
=all:com.ibm.ws.clustersensor.impl.*
=all:com.ibm.ws.xd.placement.memory.profiler.impl.*=off
wsmm_o=*=info:com.ibm.ws.xd.comm.*=all:com.ibm.wsmm.grm.Controller=
all:com.ibm.ws.xd.workprofiler.*
=all:com.ibm.ws.xd.arfm.*=all:com.ibm.wsmm.policing.*=
all:com.ibm.wsmm.xdglue.*
=all:com.ibm.ws.dwlm.*=all:com.ibm.ws.dwlm.client.*=off
dmgr=com.ibm.ws.odc.*=
all:com.ibm.ws.xd.visualizationengine.cacheservice.cacheimpl.
DeploymentTargetCache*
=all
grid=grid.capacityplacement=all
webcontainer=com.ibm.ws.webcontainer.*=all:com.ibm.ws.http.*=all
odc=com.ibm.ws.odc.*=all:com.ibm.ws.dwlm.client.*=
all:com.ibm.ws.xd.dwlm.client.*
=all:com.ibm.ws.proxy.*=all
wssec_all=com.ibm.ws.security.*=all
wssec_tws_all=com.ibm.ws.security.*=all:com.ibm.tws.*=all
tws_all=com.ibm.tws.*=all
tws_alldefault=com.ibm.tws.*=error=enabled
tws_db=com.ibm.tws.dao.model.*=all:com.ibm.tws.dao.rdbms.*=all
tws_planner=com.ibm.tws.planner.*=all:com.tivoli.icalendar.*=
all:com.ibm.tws.runcycles.*
=all:com.ibm.tws.conn.planner.*=all:com.ibm.tws.cli.planner.*=all
tws_cli=com.ibm.tws.cli.*=all:com.ibm.tws.objects.*=all
tws_utils=com.ibm.tws.util.*=all
tws_conn=com.ibm.tws.conn.*=all:com.ibm.tws.objects.*=
all:com.ibm.tws.updatemanager.*
=all:com.ibm.tws.dao.plan.*=all
tws_secjni=com.ibm.tws.audit.*=all:com.ibm.tws.security.*=all
active_correlation=com.ibm.correlation.*=all
tws_jni=TWSJNI=all
tws_all_jni=com.ibm.tws.*=all:TWSJNI=all
tws_all_act=com.ibm.tws.*=all:com.ibm.correlation.*=all
tws_broker_all=com.ibm.scheduling.*=all:TWSAgent=all
tws_broker_rest=com.ibm.scheduling.jobmanager.rest.*=all
tws_engine_broker_all=com.ibm.tws.*=all:com.ibm.scheduling.*=
all:TWSAgent=all
```

```
tws_bridge=TWSAgent=all
tws_db_transactions=com.ibm.tws.planner.currentplan.PlannerEngine=
all:com.ibm.tws.dao.rdbms.util.DatabaseTransaction=all
```

Vous pouvez définir d'autres modes de trace et mettre à jour le fichier `TracingProps.properties`, ou créer un fichier de propriétés.

Deux paramètres ne doivent pas être modifiés : le nom de serveur (**server1** par défaut) et le noeud (**DefaultNode** par défaut).

Outils WebSphere Application Server - Références

Serveur d'applications - utilisation des utilitaires qui modifient les propriétés

La présente section décrit une procédure commune que nous vous conseillons de suivre lorsque vous utilisez les utilitaires suivants :

- changeDataSourceProperties
- changeHostProperties
- changeSecurityProperties

Pour éviter de modifier par inadvertance une valeur de configuration, suivez une procédure permettant de créer un fichier contenant les propriétés en cours, modifiez ce fichier en indiquant les valeurs requises, et appliquez les modifications. Les détails se présentent comme suit :

1. Connectez-vous à l'ordinateur sur lequel Tivoli Workload Scheduler est installé en tant que l'utilisateur suivant :

   UNIX root

   Windows
   Tout utilisateur du groupe *Administrators*.
2. Accédez au répertoire : <*TWA_home*>/wastools
3. Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)
4. Dans ce répertoire, exécutez le script suivant afin de créer un fichier contenant les propriétés en cours :

   UNIX **show<property_type>Properties.sh > mon_fichier**

   Windows
   show<type_propriété>Properties.bat > mon_fichier

   où <*type_propriété*> est l'un des éléments suivants :
   - DataSource
   - Hôte
   - Sécurité
5. Editez le fichier `mon_fichier` avec un éditeur de texte. Vérifiez le début du fichier. Il se peut que la commande ait écrit un message du serveur d'applications (WASX7357I:) au début du fichier. Supprimez ce message.
6. Modifiez la valeur des paramètres de configuration en fonction de vos besoins. Vous n'êtes pas obligé d'indiquer tous les paramètres du fichier.
7. Enregistrez le fichier *mon_fichier*.
8. Exécutez le script :

Windows

```
change<type_propriété>Properties.bat mon_fichier
```

UNIX

```
change<type_propriété>Properties.sh mon_fichier
```

où <*type_propriété*> est le même que celui utilisé à l'étape 4, à la page 423, et *mon_fichier* est le *chemin d'accès qualifié complet* du fichier contenant les nouveaux paramètres.

Les propriétés sont mises à jour en fonction des règles indiquées dans les descriptions de chaque type de propriété.

9. Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)
10. Vérifiez que la modification a été implémentée dans Tivoli Workload Scheduler.

Familiarisation avec les modèles

Comme indiqué dans la présentation de ces utilitaires, un fichier de modèle des propriétés est fourni avec le produit pour chacun de ces utilitaires. Toutefois, ce fichier de modèle ne contient pas de valeurs de configuration créées par le processus d'installation du produit ou toute utilisation précédente des utilitaires de configuration. Si vous décidez d'utiliser le modèle au lieu de créer le document de propriétés comme indiqué à l'étape 4, à la page 423, vous devez vous assurer que les valeurs saisies dans le fichier pour chaque paramètre sont celles que le serveur d'applications doit utiliser.

Utilitaires du serveur d'applications

Cette section fournit des informations de référence sur les utilitaires fournis avec la WebSphere Application Server qui prennent en charge Tivoli Workload Scheduler.

Ces utilitaires sont un ensemble de scripts utilisant des fichiers de commandes par lots Windows, des scripts de shell UNIX et Linux, et des procédures WebSphere Jacl. Les scripts exécutent les utilitaires de WebSphere Application Server permettant de procéder à la reconfiguration. La plupart d'entre eux chargent les paramètres de configuration à partir d'un fichier de propriétés. Des modèles de fichier sont également disponibles.

Tivoli Workload Scheduler installe les scripts Windows suivants :

- backupConfig.bat
- brokerApplicationStatus.bat
- changeBrokerSecurityProperties.bat
- changeDataSourceProperties.bat
- changeHostProperties.bat
- changePassword.bat
- changeSecurityProperties.bat
- changeTraceProperties.bat
- encryptProfileProperties.bat
- InstallOracledataSource.bat
- InstallTDWCdataSource.bat
- manage_ltpa.bat
- modifyThreadPool.bat
- restoreConfig.bat

- setEnv.bat
- showBrokerSecurityProperties.bat
- showDataSourceProperties.bat
- showHostProperties.bat
- showSecurityProperties.bat
- startBrokerApplication.bat
- startWas.bat
- stopBrokerApplication.bat
- stopWas.bat
- updateWas.bat
- updateWasService.bat

Tivoli Workload Scheduler installe les scripts UNIX et Linux suivants :

- backupConfig.sh
- brokerApplicationStatus.sh
- changeBrokerSecurityProperties.bat
- changeDataSourceProperties.sh
- changeHostProperties.sh
- changePassword.sh
- changeSecurityProperties.sh
- changeTraceProperties.sh
- createCustomRegistryforPAM.sh
- encryptProfileProperties.sh
- InstallOracledataSource.sh
- InstallTDWCdataSource.sh
- manage_ltpa.sh
- modifyThreadPool.sh
- restoreConfig.sh
- setEnv.sh
- showBrokerSecurityProperties.sh
- showDataSourceProperties.sh
- showHostProperties.sh
- showSecurityProperties.sh
- startBrokerApplication.sh
- startWas.sh
- stopBrokerApplication.sh
- stopWas.sh
- manage_ltpa.sh
- modifyThreadPool.sh
- InstallOracledataSource.sh
- updateWas.sh
- wasstart.sh

Les modèles suivants sont installés pour les systèmes d'exploitation Windows et UNIX :

- BrokerSecurityProps.properties

- DataSourceProps.properties
- HostConfigProps.properties
- SecurityProps_FULL.properties
- SecurityProps_TEMPLATE.properties
- TDWCDatasource.properties
- TracingProps.properties

Pour plus d'informations sur le fonctionnement des modèles, voir «Serveur d'applications - utilisation des utilitaires qui modifient les propriétés», à la page 423.

Chapitre 10. Administration d'un environnement dynamique IBM i

Présentation de la manière d'administrer l'environnement dynamique Tivoli Workload Scheduler IBM i.

Avant de planifier des travaux avec des options avancées sur des agents IBM i, vous devez configurer les agents.

Configuration de l'agent sur des systèmes IBM i

Présentation de la manière de configurer l'agent sur des systèmes IBM i.

Les paramètres de configuration de l'agent sont contenus dans le fichier `JobManager.ini` (pour obtenir le chemin d'accès à ce fichier, voir «Emplacement de l'installation des produits et des composants», à la page 1). Le fichier est composé de plusieurs sections. Chaque nom de section est placé entre crochets et chaque section comporte une séquence d'instructions `variable = value`.

Vous pouvez personnaliser les propriétés des éléments suivants :

- Propriétés de journal
- Propriétés de trace lorsque l'agent est arrêté. Vous pouvez également personnaliser des traces lorsque l'agent est exécuté à l'aide de la procédure décrite dans la section «Configuration des propriétés de trace lorsque l'agent est en cours d'exécution», à la page 57.
- Exécuteur de travail natif
- Exécuteur de travail Java
- Agent d'assistant de ressources
- Scanner du système

Sur les systèmes IBM i, les messages de journal sont consignés dans le fichier suivant :

```
<TWA_home>/TWS/stdlist/JM/JObManager_message.log
```

Sur les systèmes IBM i, les messages de trace sont consignés dans les fichiers suivants :

```
<TWA_home>/TWS/stdlist/JM/ITA_trace.log
<TWA_home>/TWS/stdlist/JM/JobManager_trace.log
<TWA_home>/TWS/stdlist/JM/javaExecutor0.log
```

Vous ne pouvez pas personnaliser toutes les propriétés du fichier `JobManager.ini`. Pour obtenir la liste des propriétés configurables, consultez les sections suivantes :

- «Configuration des propriétés des messages de journal [JobManager.Logging.cclog]», à la page 55.
- «Configuration des propriétés de trace lorsque l'agent est arrêté [JobManager.Logging.cclog]», à la page 56.
- «Configuration des propriétés communes des lanceurs de tâches [Launchers]», à la page 60.
- «Configuration des propriétés du lanceur de travaux natif [NativeJobLauncher]», à la page 61.

- «Configuration des propriétés du lanceur de travaux Java [JavaJobLauncher]», à la page 63.
- «Configuration des propriétés de l'agent assistant de ressources [ResourceAdvisorAgent]», à la page 63.
- «Configuration des propriétés du scanner du système [SystemScanner]», à la page 65

Configuration des propriétés des messages de journal [JobManager.Logging.cclog]

Pour configurer les journaux, éditez la section [JobManager.Logging.cclog] dans le fichier `JobManager.ini`. Cette procédure nécessite l'arrêt et le redémarrage de l'agent Tivoli Workload Scheduler

La section qui contient les propriétés de journal est appelée :
`[JobManager.Logging.cclog]`

Vous pouvez modifier les propriétés suivantes :

JobManager.loggerhd.fileName
Nom du fichier où sont consignés les messages.

JobManager.loggerhd.maxFileBytes
Taille maximale que peut atteindre le fichier journal. La valeur par défaut est de 1024000 octets.

JobManager.loggerhd.maxFiles
Nombre maximum de fichiers journaux stockés. La valeur par défaut est 3.

JobManager.loggerhd.fileEncoding
Par défaut, les fichiers journaux de l'agent ne sont pas codés au format UTF-8. Si vous souhaitez produire le journal dans un autre format, ajoutez cette propriété et spécifiez la page de code requise.

JobManager.loggerfl.level
Quantité d'informations à fournir dans les journaux. La plage de valeurs se situe entre 4000 et 7000. Les valeurs plus faibles correspondent à des journaux plus détaillés. La valeur par défaut est 3000.

JobManager.ffdc.maxDiskSpace
Dépassement de cet espace disque maximum, les fichiers journaux collectés par le mécanisme de capture de données à la première défaillance sont supprimés, en commençant par les fichiers les plus anciens.

JobManager.ffdc.baseDir
Répertoire dans lequel sont copiés des fichiers journaux et de trace collectés par l'outil de capture de données à la première défaillance. Le répertoire par défaut est `<TWA_home>\TWS\stdlist\JM\JOBMANAGER-FFDC`.

JobManager.ffdc.filesToCopy
Fichiers journaux et de trace (JobManager_message.log et JobManager_trace.log) collectés par l'outil de capture de données à la première défaillance se trouvant dans `<TWA_home>\TWS\stdlist\JM`. Par exemple, `JobManager.ffdc.filesToCopy = "/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JobManager_message.log" "/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JobManager_trace.log"`

Lorsqu'un message est consigné (JobManager.ffdc.triggerFilter = JobManager.msgIdFilter) avec un ID en corrélation avec le modèle "AWSITA*E" (JobManager.msgIdFilter.msgIds = AWSITA*E), lequel

correspond à tous les messages d'erreur, les fichiers journaux et de trace (JobManager.ffdc.filesToCopy = "/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JobManager_message.log" "/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JobManager_trace.log") sont copiés (JobManager.ffdc.className = ccg_ffdc_filecopy_handler) to the directory JOBMANAGER-FFDC (JobManager.ffdc.baseDir = /opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/JOBMANAGER-FFDC). Si les fichiers copiés dépassent 10 Mo (JobManager.ffdc.maxDiskSpace = 10000000), les fichiers les plus anciens sont alors supprimés les premiers (JobManager.ffdc.quotaPolicy = QUOTA_AUTODELETE).

Configuration des propriétés de trace lorsque l'agent est arrêté [JobManager.Logging.cclog]

Comment configurer les propriétés de trace lorsque l'agent est arrêté.

Pour configurer les propriétés de trace lorsque l'agent est arrêté, éditez la section [JobManager.Logging] dans le fichier `JobManager.ini`, puis redémarrez agent Tivoli Workload Scheduler.

La section qui contient les propriétés de trace est appelée :

```
[JobManager.Logging.cclog]
```

Vous pouvez modifier les propriétés suivantes :

JobManager.trhd.fileName
Nom du fichier de trace.

JobManager.trhd.maxFileBytes
Taille maximale que peut atteindre le fichier de trace. La valeur par défaut est de 1024000 octets.

JobManager.trhd.maxFiles
Nombre maximum de fichiers de trace stockés. La valeur par défaut est 3.

JobManager.trfl.level
Détermine le type des messages trace consignés. Modifiez cette valeur pour choisir de consigner vous-même, ou sur demande du service de support logiciel IBM. Les valeurs valides sont les suivantes :

DEBUG_MAX
Traçage maximum. Chaque message de trace du code est écrit dans les journaux de trace.

INFO Les messages *informatifs*, *d'avertissement*, *d'erreur* et *critiques* sont écrits dans la trace. Valeur par défaut.

WARNING
Les messages d'*avertissement*, d'*erreur* et *critiques* sont écrits dans la trace.

ERROR
Tous les messages de trace *error* et *critical* sont écrits dans la trace.

CRITICAL
Seules les messages provoquant l'arrêt de l'agent sont écrits dans la trace.

La trace de sortie (JobManager_trace.log) est au format XML.

Configuration des propriétés de trace lorsque l'agent est en cours d'exécution

Utilisez la commande **twstrace** pour définir la trace sur l'agent lorsqu'il est en cours d'exécution.

A l'aide de la commande **twstrace** vous pouvez effectuer les actions suivantes sur l'agent lorsque ce dernier est en cours d'exécution :

- «Affichage de la syntaxe de commande et vérification de la version», à la page 57.
- «Activation ou désactivation de la trace», à la page 58.
- Définissez les traces sur un niveau spécifique, indiquez le nombre de fichiers de trace que vous voulez créer et la taille minimale de chacun d'eux. Voir «Définition des informations de trace», à la page 58.
- «Affichage des informations de trace», à la page 58.
- Collectez les fichiers de trace, les fichiers message et les fichiers de configuration dans un fichier compressé. Voir «Collecte des informations de trace», à la page 59.

Vous pouvez également configurer les traces lorsque l'agent n'est pas en cours d'exécution en modifiant la section [JobManager.Logging] du fichier `JobManager.ini`, comme indiqué dans la section Configuration de l'agent. Cette procédure nécessite l'arrêt et le redémarrage de l'agent.

Commande twstrace

Utilisez la commande **twstrace** pour configurer des traces et collecter des fichiers journaux, de traces et de configuration (ita.ini et jobManager.ini) pour les agents. Vous collectez toutes les informations dans un fichier compressé lorsque la commande est en cours d'exécution, sans l'arrêter ni la redémarrer.

Affichage de la syntaxe de commande et vérification de la version

Pour afficher la syntaxe et les options de la commande, utilisez la syntaxe ci-dessous.

Syntaxe

twstrace -u | -v

Paramètres

-u Affiche la syntaxe de la commande.

-v Affiche la version de la commande.

Activation ou désactivation de la trace

Pour définir le niveau de trace maximal ou le désactiver, utilisez la syntaxe ci-dessous.

Syntaxe

twstrace -enable | -disable

Paramètres

-enable
Définit le niveau de trace maximal. Le niveau maximal est **3000**.

-disable
Désactive les traces sur l'agent.

Définition des informations de trace

Pour définir un niveau de trace spécifique, indiquez le nombre de fichiers de trace que vous voulez créer ainsi que la taille maximale des fichiers de trace en utilisant la syntaxe ci-dessous.

Syntaxe

twstrace [**-level** <*numéro_niveau*>] [**-maxFiles** <*nombre_fichiers*>] [**-maxFileBytes** <*nombre_octets*>]

Paramètres

-level <*numéro_niveau*>
Définissez le niveau de trace en indiquant une valeur comprise entre 1000 et 3000.

-maxFiles <*nombre_fichiers*>
Indiquez le nombre de fichiers de trace que vous voulez créer.

-maxFileBytes <*nombre_octets*>
Définissez la taille maximale en octets que peuvent atteindre les fichiers de trace. La valeur par défaut est **1024000** octets.

Affichage des informations de trace

Pour afficher le niveau de trace courant, le nombre de fichiers de trace et leur taille maximale, utilisez la syntaxe ci-dessous.

Syntaxe

twstrace -level | -maxFiles | -maxFileBytes

Paramètres

-level
Affiche le niveau de trace que vous définissez.

-maxFiles
Affiche le nombre de fichiers de trace que vous créez.

-maxFileBytes
Affiche la taille maximale que vous définissez pour chaque fichier de trace

Exemple

L'exemple montre les informations que vous recevez lorsque vous exécutez la commande suivante :

```
twstrace -level -maxFiles -maxFileBytes

AWSITA176I The trace properties are: level="1000",
max files="3", file size="1024000".
```

Collecte des informations de trace

Pour collecter les fichiers de trace, les fichiers message et les fichiers de configuration dans un fichier compressé, utilisez la syntaxe suivante.

Syntaxe

twstrace -getLogs [**-zipFile** <*nom_fichier_compressé*>] [**-host** <*nom_hôte*>] [**-protocol** {*http | https* } [**-port** <*numéro_port*>][**-iniFile** <*nom_fichier_ini*>]

Paramètres

-zipFile <*nom_fichier_compressé*>
Indiquez le nom du fichier compressé contenant toutes les informations, à savoir, les fichiers journaux, de trace et de configuration (ita.ini et jobManager.ini) pour l'agent. Le fichier par défaut est **logs.zip**.

-host <*nom_hôte*>
Indiquez le nom d'hôte ou l'adresse IP de l'agent pour lequel vous voulez collecter la trace. La valeur par défaut est **localhost**.

-protocol *http|https*
Indiquez le protocole de l'agent pour lequel vous collectez la trace. La valeur par défaut est le protocole indiqué dans le fichier **.ini** de l'agent.

-port <*numéro_port*>
Indiquez le port de l'agent. Par défaut il s'agit du numéro de port de l'agent pour lequel vous exécutez la ligne de commande.

-iniFile <*nom_fichier_ini*>
Indiquez le nom du fichier **.ini** contenant la configuration SSL de l'agent pour lequel vous voulez collecter les traces. Si vous collectez les traces pour un agent distant pour lequel vous avez personnalisé les certificats de sécurité, vous devez importer le certificat sur l'agent local et indiquer le nom du fichier **.ini** contenant ces informations. Pour cela, procédez comme suit :

1. Extrayez le certificat du magasin de clés de l'agent distant.
2. Importez le certificat dans un magasin de clés d'agent local. Vous pouvez créer un magasin de clés ad hoc dont le nom doit être **TWSClientKeyStore.kdb**.
3. Créez un fichier **.ini** dans lequel vous indiquez :
   - **0** dans la propriété **tcp_port** comme suit :
     `tcp_port=0`
   - Le port de l'agent distant dans la propriété **ssl_port** est le suivant :
     `ssl_port=<ssl_port>`
   - Le chemin d'accès au magasin de clés créé à l'étape 2, à la page 59 dans la propriété **key_repository_path** est le suivant :
     `key_repository_path=<chemin_magasin_clés_agent_local>`

Configuration des propriétés communes des lanceurs de tâches [Launchers]

Dans le fichier `JobManager.ini`, la section contenant les propriétés communes aux différents lanceurs de tâches (ou exécuteurs) est appelée :

`[Launchers]`

Vous pouvez modifier les propriétés suivantes :

BaseDir
Chemin d'installation de l'agent Tivoli Workload Scheduler.

CommandHandlerMinThreads
La valeur par défaut est **20**.

CommandHandlerMaxThreads
La valeur par défaut est **100**.

CpaHeartBeatTimeSeconds
Intervalle d'interrogation (en secondes) permettant de vérifier que la processus d'**agent** est toujours actif et en cours d'exécution. S'il est inactif, le produit arrête également le processus **JobManager**. La valeur par défaut est **30**.

DirectoryPermissions
Droits d'accès attribués à l'agent pour la création de répertoires lors de l'exécution de travaux. La valeur par défaut est **0755**. Les valeurs prises en charge sont des entrées au format UNIX en notation hexadécimale.

ExecutorsMaxThreads
La valeur par défaut est **400**.

ExecutorsMinThreads
La valeur par défaut est **38**.

FilePermissions
Droits d'accès attribués à l'agent pour la création de fichiers lors de l'exécution de travaux. La valeur par défaut est **0755**. Les valeurs prises en charge sont des entrées au format UNIX en notation hexadécimale.

MaxAge
Délai de conservation en jours des journaux de travaux (à l'emplacement *TWA_home*/TWS/stdlidst/JM) avant suppression. La valeur par défaut est **30**. La plage des valeurs possibles commence à 1 jour.

NotifierMaxThreads
La valeur par défaut est **5**.

NotifierMinThreads
La valeur par défaut est **3**.

SpoolDir
Chemin du dossier contenant le jobstore et les sorties. La valeur par défaut est :

valeur de BaseDir/stdlidst/JM

StackSizeBytes
Taille de la pile du système d'exploitation (en octets). La valeur par défaut est **DEFAULT**, ce qui signifie que l'**agent** utilise la valeur par défaut pour le système d'exploitation.

Configuration des propriétés du lanceur de travaux natif [NativeJobLauncher]

Dans le fichier JobManager.ini, la section contenant les propriétés du lanceur de travaux natif est appelée :

```
[NativeJobLauncher]
```

Vous pouvez modifier les propriétés suivantes :

AllowRoot
S'applique uniquement aux systèmes UNIX. Indique si le superutilisateur peut exécuter des travaux sur l'agent. Ce paramètre peut être défini sur `true` ou `false`. La valeur par défaut est `true`.

CheckExec
Si elle est définie sur `true`, avant de lancer le travail, l'agent vérifie la disponibilité et les droits d'exécution du fichier binaire. La valeur par défaut est `true`.

JobUnspecifiedInteractive
S'applique uniquement aux systèmes d'exploitation Windows. Spécifie si les travaux natifs sont lancés en mode interactif. Ce paramètre peut être défini sur `true` ou `false`. La valeur par défaut est `false`.

KeepCommandTraces
Définissez cette propriété sur `true` afin d'enregistrer les traces de l'appel de méthode pour les actions effectuées sur une définition de travail, par exemple, lors de la sélection d'une liste de réquisition. Ces fichiers sont enregistrés dans le chemin `/opt/IBM/TWA_<utilisateur_TWS>/TWS/stdlist/JM/r3batch_cmd_exec`. Le paramètre par défaut est `false`.

KeepJobCommandTraces
Définissez cette propriété sur `true` pour enregistrer les traces de l'appel de méthode pour les actions effectuées sur une instance de travail, par exemple, afficher une liste de spool. Ces fichiers sont enregistrés dans le fichier .zip de l'instance de travail. Le paramètre par défaut est `true`.

LoadProfile
Spécifie si le profil utilisateur va être chargé. Ce paramètre peut être défini sur `true` ou `false`. La valeur par défaut est `true`.

PortMax
La plage maximum des numéros de port utilisés par le lanceur de tâches pour communiquer avec le gestionnaire de travaux. La valeur par défaut est 0 ; cette valeur indique au système d'exploitation d'attribuer le port automatiquement.

PortMin
La plage minimum des numéros de port utilisés par le lanceur de tâches pour communiquer avec le gestionnaire de travaux. La valeur par défaut est 0 ; cette valeur indique au système d'exploitation d'attribuer le port automatiquement.

PromotedNice
Utilisée dans l'assurance de service de charge de travail. Cette propriété n'est pas prise en charge sur l'Agent for z/OS.

Pour les systèmes d'exploitation UNIX et Linux uniquement, attribue la valeur prioritaire à un travail critique qui soit être promu afin que le système d'exploitation le traite avant les autres. S'applique aux travaux critiques ou à leurs prédécesseurs qui doivent être promus afin qu'ils puissent commencer à l'heure critique locale.

Les valeurs limites varient en fonction des plateformes, mais généralement, les valeurs inférieures correspondent aux niveaux de haute priorité et vice versa. La valeur par défaut est -1.

Sachez que :

- Le processus de promotion n'est efficace qu'avec des valeurs négatives. Si vous définissez une valeur positive, le système l'exécute avec la valeur par défaut -1.
- ne valeur hors plage (par exemple -200) incite le système d'exploitation à promouvoir automatiquement les travaux dont la valeur nice attribuée est la plus basse.
- L'utilisation abusive du mécanisme de promotion (c'est-à-dire la définition d'un nombre excessif de travaux comme critiques et la définition de la valeur de priorité la plus élevée ici) risque de surcharger le système d'exploitation, entraînant un impact négatif sur les performances générales du poste de travail.

PromotedPriority
Utilisée dans l'assurance de service de charge de travail. Cette propriété n'est pas prise en charge sur l'Agent for z/OS.

Pour les systèmes d'exploitationWindows uniquement, cette valeur indique la priorité selon laquelle le système d'exploitation traite un travail critique lorsqu'il est promu. S'applique aux travaux critiques ou à leurs prédécesseurs qui doivent être promus afin qu'ils puissent commencer à l'heure critique locale. Les valeurs valides sont les suivantes :

- `High`
- `AboveNormal` (valeur par défaut)
- `Normal`
- `BelowNormal`
- `Low` ou `Idle`

Si vous définissez une valeur de priorité inférieure à celle qui peut être attribuée aux travaux non critiques, aucun avertissement n'est envoyé.

RequireUserName

Lorsqu'il est défini sur `true`, vous devez ajouter le nom d'utilisateur dans la définition de travail JSDL.

Lorsqu'il est défini sur `false`, il s'exécute avec le nom d'utilisateur utilisé par le gestionnaire de travaux, à savoir :

- *utilisateur_TWS* sur les systèmes UNIX et Linux
- Le compte de système local sur les systèmes Windows

La valeur par défaut est `false`.

ScriptSuffix
Suffixe à utiliser lors de la création des fichiers script. Il s'agit de :

`.cmd` Pour Windows

`.sh` Pour UNIX

VerboseTracing
Active la fonction de trace prolixe. Ce paramètre est défini par défaut sur `true`.

Configuration des propriétés du lanceur de travaux Java [JavaJobLauncher]

Dans le fichier `JobManager.ini`, la section contenant les propriétés du lanceur de travaux Java est nommée :

```
[JavaJobLauncher]
```

Vous pouvez modifier les propriétés suivantes :

JVMDir
Chemin d'accès à la machine virtuelle utilisée pour démarrer les types de travail avec options avancées. Vous pouvez modifier le chemin en le remplaçant par un autre compatible avec la machine virtuelle Java.

JVMOptions
Les options à fournir à la machine virtuelle Java pour lancer les types de travail avec options avancées. Les mot clés pris en charge pour établir une connexion sécurisée sont :

- htttps.proxyHost
- https.proxyPort

Les mots clés pris en charge pour établir une connexion non sécurisée sont :

- Dhttp.proxyHost
- Dhttp.proxyPort

Par exemple, pour définir des types de travail avec options avancées, en fonction du protocole http JVM par défaut, sur le serveur proxy non authentifié appelé avec le nom myproxyserver.mycompany.com, définissez l'option suivante :

```
JVMOptions = -Dhttp.proxyHost=myproxyserver.mycompany.com
 -Dhttp.proxyPort=80
```

Configuration des propriétés de l'agent assistant de ressources [ResourceAdvisorAgent]

Dans les fichiers `JobManager.ini` et `JobManagerGW.ini`, la section contenant les propriétés de l'agent assistant de ressources s'intitule :

```
[ResourceAdvisorAgent]
```

Vous pouvez modifier les propriétés suivantes :

BackupResourceAdvisorUrls
La liste des adresses URL retournées par le maître Tivoli Workload Scheduler dans un environnement réparti ou par le gestionnaire de domaine dynamique dans un environnement z/OS ou réparti. L'agent utilise cette liste pour se connecter au maître ou au gestionnaire de domaine dynamique.

CPUScannerPeriodSeconds
L'intervalle de temps où l'agent d'assistant de ressources collecte les informations de ressources concernant l'unité centrale locale. La valeur par défaut est toutes les 10 secondes.

FullyQualifiedHostname
Nom d'hôte qualifié complet de l'agent. Il est configuré automatiquement lors de l'installation et utilisé pour la connexion au maître dans un environnement réparti ou au gestionnaire de domaine dynamique dans un environnement z/OS ou réparti. Modifiez-le uniquement si le nom d'hôte est modifié après l'installation.

NotifyToResourceAdvisorPeriodSeconds
L'intervalle de temps où l'agent d'assistant de ressources transfert les informations des ressources collectées à l'assistant de ressources. La valeur par défaut (et maximum) est toute les 180 secondes.

ResourceAdvisorUrl

JobManager.ini

L'adresse URL du maître dans un environnement réparti ou de gestionnaire de domaine dynamique dans un environnement z/OS ou réparti qui héberge l'agent. Cette URL est utilisée jusqu'à ce que le serveur réponde en renvoyant la liste de ses URL. La valeur est `https://$(tdwb_server):$(tdwb_port)/JobManagerRESTWeb/JobScheduler/resource`, où :

$(*tdwb_server***)**
Nom d'hôte qualifié complet du maître dans un environnement réparti, ou du gestionnaire de domaine dynamique dans un environnement z/OS ou réparti.

$(*tdwb_port***)**
Numéro de port du maître dans un environnement réparti, ou du gestionnaire de domaine dynamique dans un environnement z/OS ou réparti.

Il est automatiquement configuré au moment de l'installation. Modifiez cette valeur uniquement si le nom d'hôte ou le nom de port est modifié après l'installation, ou si vous n'utilisez pas de connexion sécurisée (adresse définie sur `http`). Si vous définissez le numéro de port sur zéro, l'agent d'assistant de ressources ne démarre pas. Le port est défini sur zéro si, au moment de l'installation, vous avez spécifié que vous n'utiliseriez pas le maître dans un environnement réparti ou le gestionnaire de domaine dynamique dans un environnement z/OS ou réparti.

Dans un environnement distribué, si **-gateway** est défini à `local` ou `remote`, il s'agit alors de l'adresse URL du poste de travail de l'agent dynamique dans lequel la passerelle réside et par lequel les agents dynamiques communiquent. La valeur est `https://$(tdwb_server):$(tdwb_port)/ita/JobManagerGW/JobManagerRESTWeb/JobScheduler/resource`, où :

$(*tdwb_server***)**
Nom d'hôte qualifié complet du poste de travail agent dynamique où la passerelle réside et par lequel l'agent dynamique communique avec Dynamic Workload Broker.

$(*tdwb_port***)**
Numéro de port du poste de travail agent dynamique où la passerelle réside et par lequel l'agent dynamique communique avec Dynamic Workload Broker.

JobManagerGW.ini

Dans un environnement distribué, si **-gateway** est défini à `local`, **ResourceAdvisorUrl** est alors l'adresse URL du maître ou du gestionnaire de domaine dynamique. La valeur est `https://$(tdwb_server):$(tdwb_port)/JobManagerRESTWeb/JobScheduler/resource`, où :

$(*tdwb_server***)**
Nom d'hôte qualifié complet du maître ou du gestionnaire de domaine dynamique.

$(*tdwb_port*)
Numéro de port du maître ou du gestionnaire de domaine dynamique.

ScannerPeriodSeconds
L'intervalle de temps où l'agent d'assistant de ressources collecte les informations sur toutes les ressources du système local autres que les ressources de l'unité centrale. La valeur par défaut est toutes les 120 secondes.

L'agent assistant de ressource analyse les ressources de la machine par intermittence (le système informatique, le système d'exploitation, les systèmes de fichiers et les réseaux) et envoie périodiquement une mise à jour de leur état au maître ou au gestionnaire de domaine dynamique dans un environnement z/OS ou réparti.

L'unité centrale est analysée toutes les `CPUScannerPeriodSeconds` secondes, alors que les autres ressources le sont toutes les `ScannerPeriodSeconds` secondes. Dès qu'une des analyses présente une modification significative de l'état d'une ressource, les ressources sont synchronisées avec le maître dans un environnement réparti ou avec le gestionnaire de domaine dynamique dans un environnement z/OS ou réparti. La politique appliquée par l'agent pour signifier si l'attribut d'une ressource a changé de manière significative est la suivante :

- Une ressource a été ajoutée ou supprimée
- La valeur d'un attribut chaîne a changé
- Une valeur de l'unité centrale a changé de plus de `DeltaForCPU`
- Une valeur du système de fichiers a changé de plus de `DeltaForDiskMB` mégaoctets
- Une valeur de la mémoire a changé de plus de `DeltaForMemoryMB` mégaoctets

S'il n'y a aucune modification significative, les ressources sont synchronisées avec le maître Tivoli Workload Scheduler dans un environnement réparti ou avec le gestionnaire de domaine dynamique dans un environnement z/OS ou réparti, à la fréquence en secondes indiquée dans `NotifyToResourceAdvisorPeriodSeconds`.

Configuration des propriétés du scanner du système [SystemScanner]

Dans le fichier `JobManager.ini`, la section contenant les propriétés du scanner du système est appelée :

```
[SystemScanner]
```

Vous pouvez modifier les propriétés suivantes :

CPUSamples
Le nombre d'exemples utilisés pour calculer l'utilisation moyenne de l'unité centrale. La valeur par défaut est 3.

DeltaForCPU
La modification de l'utilisation de l'unité centrale est considérée comme importante lorsqu'elle excède ce pourcentage (par exemple, la valeur de DeltaForCPU est 20 si l'utilisation de l'unité centrale passe de 10 à 30%). La valeur par défaut est 20%.

DeltaForDiskMB
La modification de l'utilisation des ressources du système de fichiers est considérée comme importante lorsqu'elle excède cette valeur. La valeur par défaut est 100 Mo.

DeltaForMemoryMB
La modification de l'utilisation de la mémoire système est considérée comme importante lorsqu'elle excède cette valeur. La valeur par défaut est 100 Mo.

Configuration pour planifier des types de travail avec options avancées

Parallèlement à la définition des types de travail avec options avancées avec Dynamic Workload Console ou la commande **composer**, vous pouvez utiliser les fichiers de configuration associés. Les options que vous définissez dans le fichier de configuration s'appliquent à tous les types de travail avec options avancées du même type. Vous pouvez substituer ces options lorsque vous définissez le travail à l'aide de Dynamic Workload Console ou de la commande **composer**.

Les fichiers de configuration sont disponibles sur chaque agent dynamique dans *TWA_home*/TWS/JavaExt/cfg pour les types de travail avec options avancées suivants :

Tableau 74. Fichiers de configuration pour les types de travail avec options avancées

Type de travail	Nom de fichier	Mot clé
• Type de travail de base de données • Travail MSSQL	DatabaseJobExecutor.properties	Utilisez le mot clé `jdbcDriversPath` pour spécifier le chemin d'accès aux pilotes JDBC. Définissez le mot de passe de sorte qu'il pointe vers le répertoire des fichiers jar JDBC, par exemple : `jdbcDriversPath=c:\\mydir\\jars\\jdbc` Les fichiers jar JDBC doivent se trouver dans le répertoire spécifié ou dans ses sous-répertoires. Assurez-vous que vous disposez des autorisations de liste pour le répertoire et ses sous-répertoires. **Remarque :** Pour la base de données MSSQL, utilisez la version 4 des pilotes JDBC.
Type de travail Java	JavaJobExecutor.properties	Utilisez le mot clé `jarPath` pour spécifier le chemin d'accès au répertoire dans lequel les fichiers jar sont stockés. Sont compris tous les fichiers jar stockés dans le répertoire spécifié et tous les sous-répertoires.
Type de travail J2EE	J2EEJobExecutorConfig.properties	Pour plus d'informations sur le type de travail J2EE, voir Configuration pour planifier des travaux J2EE.

Personnalisation de la connexion SSL entre des agents IBM i et un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique à l'aide de vos propres certificats

Personnalisation de la connexion SSL entre un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique et des agents IBM i connecté à ce gestionnaire à l'aide de vos propres certificats.

Par défaut, la communication entre des agents IBM i et un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique auprès duquel ils sont enregistrés utilise le protocole HTTPS.

La communication SSL utilise les certificats par défaut fournis par Tivoli Workload Scheduler.

Si vous voulez utiliser vos propres certificats personnalisés pour cette communication parce que vous avez personnalisé les certificats du gestionnaire de domaine maître ou du gestionnaire de domaine dynamique, vous devez personnaliser les certificats d'agent ainsi que le fichier de configuration d'agent.

Pour activer la communication entre un gestionnaire de domaine maître ou un gestionnaire de domaine dynamique et un agent IBM i, vous devez d'abord créer vos propres certificats pour l'agent IBM i, puis certifier les certificats d'agent dans le fichier de clés du gestionnaire de domaine maître ou du gestionnaire de domaine dynamique.

Procédez comme suit :

1. Connectez-vous, en tant qu'administrateur sous Windows ou en tant que superutilisateur sous UNIX et Linux, sur la machine où vous avez installé une instance Tivoli Workload Scheduler contenant l'utilitaire **openssl**, par exemple, le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique.
2. Accédez au répertoire *<REP_INSTALL_TWS>*/TWS/ssl, où *<REP_INSTALL_TWS>* est le répertoire d'installation de Tivoli Workload Scheduler, et copiez-y les fichiers suivants :
   - *<REP_INSTALL_TWS>*/TWS/bin/openssl(.exe)
   - *<REP_INSTALL_TWS>*/TWS/bin/openssl.cnf
3. Générez un fichier aléatoire pour l'agent IBM i en exécutant la commande suivante :

   ```
   openssl rand
   -out <suffixe>.rnd
   -rand ./openssl 8192
   ```

   où *<suffixe>* est un mot générique. Par exemple, vous pouvez utiliser le nom du poste de travail de l'agent IBM i de manière à trouver facilement les fichiers générés pour ce poste de travail.
4. Générez la clé privée *<suffixe>*.key en exécutant la commande suivante :

   ```
   openssl genrsa -des3
   -out <suffixe>.key 2048
   ```

   et enregistrez le mot de passe entré à la commande précédente dans le fichier *<suffixe>*.pwd.

   Remarque : Veillez à noter ce mot de passe car vous sera nécessaire dans les étapes qui suivent.

5. Générez le fichier PEM ita_prv<*suffixe*>.pem contenant la clé privée de l'agent, en renommant <*suffixe*>.key en ita_prv<*suffixe*>.pem.
6. Enregistrez le mot de passe de la clé privée de l'agent dans un fichier de dissimulation <*suffixe*>.sth en exécutant la commande suivante :

   ```
   openssl base64
   -in <suffixe>.pwd
   -out <suffixe>.sth
   ```

7. Générez la demande de signature de certificat <*suffixe*>.csr en exécutant la commande suivante :

   ```
   openssl req -new
   -key <suffixe>.key
   -out <suffixe>.csr
   -config ./openssl.cnf
   ```

8. Générez le certificat <*suffixe*>.crt contenant la clé privée <*suffixe*>.key en exécutant la commande suivante :

   ```
   openssl x509 -req
   -CA TWSca.crt
   -CAkey TWSca.key
   -days 365
   -in <suffixe>.csr
   -out <suffixe>.crt
   -CAcreateserial
   ```

9. Générez le fichier PEM <*suffixe*>.pem contenant le certificat de clé privée de l'agent en effectuant une copie du certificat <*suffixe*>.crt, puis nommez le fichier copié <*suffixe*>.pem.
10. Générez le fichier PEM ita_pub<*suffixe*>.pem contenant le certificat de clé privée de l'agent en effectuant une copie du certificat <*suffixe*>.crt, puis nommez le fichier copié ita_pub<*suffixe*>.pem.
11. Créez une copie du fichier ita_pub<*suffixe*>.pem créé à l'étape 10, puis nommez le fichier copié ita_cert<*suffixe*>.pem.
12. Sur la machine du gestionnaire de domaine maître ou du gestionnaire de domaine dynamique auquel l'agent IBM i doit être connecté, générez le certificat server.pem en exécutant la commande :

    ```
    keytool -export -rfc
    -alias server
    -file <REP_INSTALL_TWS>/TWS/ssl/server.pem
    -keypass <password>
    -keystore <REP_INSTALL_TWS>/TWA/WAS/TWSprofile/etc/TWSServerKeyFile.jks
    -storepass default
    ```

    où <password> est la valeur entrée à l'étape 4, à la page 440.
13. Générez le fichier ita_ca_cert<*suffixe*>.pem, qui est la concaténation des fichiers ita_pub<*suffixe*>.pem et server.pem, en effectuant les opérations suivantes :
    a. Créez une copie du fichier ita_pub<*suffixe*>.pem, puis nommez la copie ita_ca_cert<*suffixe*>.pem.
    b. Editez le fichier ita_ca_cert<*suffixe*>.pem.
    c. Ajoutez à la fin du fichier ita_ca_cert<*suffixe*>.pem le contenu du fichier server.pem.
    d. Enregistrez la version finale du fichier ita_ca_cert<*suffixe*>.pem.

    Remarque : Le fichier ita_ca_cert<*suffixe*>.pem contient les certificats de l'agent IBM i et le gestionnaire de domaine maître ou le gestionnaire de domaine dynamique auquel l'agent est connecté.

14. Connectez-vous en tant qu'utilisateur <*TWS_IBMi_USER*> sur la machine de l'agent IBM i et recherchez le répertoire <*REPINSTALL_TWS_IBMI*>/TWS/ITA/cpa/ita/cert/ où <*REPINSTALL_TWS_IBMI*> est le répertoire d'installation de l'agent Tivoli Workload Scheduler IBM i pour l'utilisateur <*TWS_IBMi_USER*>.
15. A partir du répertoire <*REP_INSTALL_TWS*>/TWS/ssl de la machine sur laquelle vous avez généré les fichiers PEM, copiez les fichiers suivants dans le répertoire <*REPINSTALL_TWS_IBMI*>/TWS/ITA/cpa/ita/cert/ du répertoire d'installation de l'agent IBM i :
    - ita_prv<*suffixe*>.pem.
    - ita_pub<*suffixe*>.pem.
    - ita_cert<*suffixe*>.pem.
    - ita_ca_cert<*suffixe*>.pem.
    - <*suffixe*>.sth.
    - <*suffixe*>.rnd.

    Remarque : Vérifiez que les fichiers copiés sont la propriété de <*TWS_IBMi_USER*>.
16. Sur la machine où vous avez installé l'agent IBM i, ouvrez le fichier d'agent de configuration ita.ini et définissez les valeurs appropriées à votre environnement dans les propriétés suivantes :

    fichier_mot_passe=<chemin_complet_fichier_dissimulation>
    fichier_aléatoire=<chemin_complet_fichier_aléatoire>
    étiquette_certificat=<étiquette_clé_privée_agent>
    nom_bd_clés=<*suffixe*>
    rép_référentiel_clés=<répertoire_ita_*<*suffixe*>.pem>

    Où :

    <chemin_complet_fichier_dissimulation>
    Chemin d'accès complet au fichier de dissimulation <*suffixe*>.sth qui contient le mot de passe de clé privée d'agent. Il s'agit du fichier créé à l'étape 6, à la page 441. La valeur par défaut est <*REPINSTALL_TWS_IBMI*>/TWS/ITA/cpa/ita/cert/password.sth.

    <chemin_complet_fichier_aléatoire>
    Chemin d'accès complet au fichier aléatoire <*suffixe*>.rnd. Il s'agit du fichier créé à l'étape 3, à la page 440. La valeur par défaut est <*REPINSTALL_TWS_IBMI*>/TWS/ITA/cpa/ita/cert/TWS.rnd.

    <étiquette_clé_privée_agent>
    Etiquette de la clé privée de l'agent.

    <*suffixe*>
    Suffixe utilisé dans les noms de tous les fichiers que vous avez générés. La valeur par défaut est **tws**.

    <répertoire_ita_*<*suffixe*>.pem>
    Répertoire qui contient les fichiers .pem générés suivants :

    Magasin de clés de confiance
    ita_ca_cert<*suffixe*>.pem généré à l'étape 13, à la page 441.

    Magasin de clés
    - ita_prv<*suffixe*>.pem généré à l'étape 5, à la page 441.
    - ita_pub<*suffixe*>.pem généré à l'étape 10, à la page 441.
    - ita_cert<*suffixe*>.pem généré à l'étape 11, à la page 441.

Le répertoire par défaut est <*REPINSTALL_TWS_IBMI*>/TWS/ITA/cpa/ita/cert.

17. Arrêtez l'agent IBM i en exécutant la commande suivante :

    ```
    ShutDownLwa
    ```

18. Démarrez l'agent IBM i exécutant la commande suivante :

    ```
    StartUpLwa
    ```

19. Sur la machine du gestionnaire de domaine maître ou du gestionnaire de domaine dynamique auquel l'agent IBM i doit être connecté, certifiez le certificat d'agent IBM i, <*REP_INSTALL_TWS*>/TWS/ssl/<*suffixe*>.pem, généré à l'étape 9, à la page 441, dans le fichier de clés, à l'aide des commandes suivantes :

    ```
    keytool -import -trustcacerts
    -alias <suffixe>
    -file <REP_INSTALL_TWS>/TWS/ssl/<suffixe>.pem
    -keypass <password>
    -keystore <REP_INSTALL_TWS>/TWA/WAS/TWSprofiles/etc/
        TWSServerTrustFile.jks
    -storepass default
    ```

    où <*REP_INSTALL_TWS*> est le répertoire d'installation du gestionnaire de domaine maître ou du gestionnaire de domaine dynamique et <mot_passe> la valeur entrée à l'étape 4, à la page 440.

Vous obtenez l'environnement suivant :

- Agent IBM i installé dans le répertoire opt/ibm/TWS de la machine nc117031 pour l'utilisateur twsuserIBMi.
- Gestionnaire de domaine maître installé dans le répertoire opt/IBM/TWA92 de la machine nc060201.

Pour créer les certificats d'agent IBM i pour la connexion au gestionnaire de domaine maître, procédez comme suit :

1. Connectez-vous en tant que superutilisateur à la machine nc060201 sur laquelle vous avez installé le gestionnaire de domaine maître.
2. Accédez au répertoire opt/IBM/TWA92/TWS/ssl et copiez-y les fichiers suivants :
   - opt/IBM/TWA92/TWS/bin/openssl
   - opt/IBM/TWA92/TWS/bin/openssl.cnf
3. Générez le fichier aléatoire nc117031.rnd dans le répertoire opt/IBM/TWA92/TWS/ssl en exécutant la commande suivante :

   ```
   openssl rand
   -out nc117031.rnd
   -rand ./openssl 8192
   ```

4. Générez la clé privée nc117031.key dans le répertoire opt/IBM/TWA92/TWS/ssl en exécutant la commande suivante :

   ```
   openssl genrsa -des3
   -out nc117031.key 2048
   ```

   et enregistrez le mot de passe maestro00 entré dans le fichier nc117031.pwd au format texte dans le répertoire opt/IBM/TWA92/TWS/ssl .
5. Créez une copie du fichier nc117031.key dans le répertoire opt/IBM/TWA892/TWS/ssl et nommez-le ita_prvnc117031.pem.
6. Enregistrez le mot de passe maestro00 dans un fichier de dissimulation nc117031.sth du répertoire opt/IBM/TWA92/TWS/ssl en exécutant la commande suivante :

```
openssl base64
-in nc117031.pwd
-out nc117031.sth
```

7. Générez la demande de signature de certificat nc117031.csr dans le répertoire opt/IBM/TWA92/TWS/ssl en exécutant la commande suivante :

```
openssl req -new
-key nc117031.key
-out nc117031.csr
-config ./openssl.cnf
```

8. Générez le certificat nc117031.crt dans le répertoire opt/IBM/TWA92/TWS/ssl qui contient la clé privée nc117031.key en exécutant la commande suivante :

```
openssl x509 -req
-CA TWSca.crt
-CAkey TWSca.key
-days 365
-in nc117031.csr
-out nc117031.crt
-CAcreateserial
```

9. Créez une copie du certificat nc117031.crt dans le répertoire opt/IBM/TWA92/TWS/ssl et nommez-le nc117031.pem.
10. Créez une copie du certificat nc117031.crt dans le répertoire opt/IBM/TWA92/TWS/ssl et nommez-le ita_pubnc117031.pem.
11. Créez une copie du fichier ita_pubnc117031.pem dans le répertoire opt/IBM/TWA92/TWS/ssl et nommez-le ita_certnc117031.pem.
12. Sur la machine nc060201, générez le certificat server.pem dans le répertoire opt/IBM/TWA92/TWS/ssl en exécutant la commande suivante :

```
keytool -export -rfc
-alias server
-file opt/IBM/TWA/TWS/ssl/server.pem
-keypass maestro00
-keystore opt/IBM/TWA/WAS/TWSprofile/etc/TWSServerKeyFile.jks
-storepass default
```

13. Générez dans le répertoire opt/IBM/TWA86/TWS/ssl le fichier ita_ca_certnc117031.pem, qui est la concaténation des fichiers ita_pubnc117031.pem et server.pem, en effectuant les opérations suivantes :
    a. Créez une copie du fichier ita_pubnc117031.pem dans le répertoire opt/IBM/TWA/TWS/ssl et nommez-le ita_ca_certnc117031.pem.
    b. Editez le fichier ita_ca_certnc117031.pem.
    c. Ajoutez à la fin du fichier ita_ca_certnc117031.pem le contenu du fichier server.pem.
    d. Enregistrez la version finale du fichier ita_ca_certnc117031.pem.
14. Connectez-vous en tant qu'utilisateur twsuserIBMi à la machine nc117031 et recherchez le répertoire opt/ibm/TWS/ITA/cpa/ita/cert/.
15. A partir du répertoire opt/IBM/TWA/TWS/ssl de la machine nc060201 sur laquelle vous avez généré les fichiers PEM, copiez les fichiers suivants dans le répertoire opt/ibm/TWS/ITA/cpa/ita/cert/ :
    - ita_prvnc117031.pem.
    - ita_pubnc117031.pem.
    - ita_certnc117031.pem.
    - ita_ca_certnc117031.pem.
    - nc117031.sth.
    - nc117031.rnd.

    Vérifiez que tous les fichiers sont la propriété de twsuserIBMi.

16. Sur la machine nc117031, ouvrez le fichier d'agent de configuration ita.ini et définissez les valeurs suivantes pour les propriétés répertoriées :

    fichier_mot_passe=opt/ibm/TWS/ITA/cpa/ita/cert/nc117031.sth
    fichier_aléatoire=opt/ibm/TWS/ITA/cpa/ita/cert/nc117031.rnd
    étiquette_certificat=nc117031
    nom_bd_clés=nc117031
    rép_référentiel_clés=opt/ibm/TWS/ITA/cpa/ita/cert/*nc117031.pem>

17. Arrêtez l'agent IBM i en exécutant la commande suivante :

    ```
    ShutDownLwa
    ```

18. Démarrez l'agent IBM i exécutant la commande suivante :

    ```
    StartUpLwa
    ```

19. Sur la machine nc060201, certifiez le certificat d'agent opt/IBM/TWA92/TWS/ssl/nc117031.pem en effectuant les opérations suivantes :

    ```
    keytool -import -trustcacerts
    -alias nc117031
    -file opt/IBM/TWA/TWS/ssl/ssl/nc117031.pem
    -keypass maestro00
    -keystore opt/IBM/TWA/WAS/TWSprofile/etc/TWSServerTrustFile.jks
    -storepass default
    ```

Chapitre 11. Performances

Le présent chapitre apporte des informations sur les éléments qui agissent sur les performances. Utilisez ces informations pour éviter tout incident et faciliter la résolution de ceux qui se sont produits.

Il contient les rubriques suivantes :

Trafic réseau

Une description complète de la façon dont un réseau Tivoli Workload Scheduler est structuré et de la façon dont les différents noeuds communiquent est proposée au début du Chapitre 6, «Administration du réseau», à la page 227. Consultez en particulier la section «Optimisation du réseau», à la page 242, qui explique comment concevoir et utiliser votre réseau Tivoli Workload Scheduler de façon à optimiser les performances.

Fonction de trace

Les performances d'un poste de travail dépendent du niveau de traçage à effectuer. Le manuel *Tivoli Workload Scheduler - Guide d'identification des problèmes* contient un chapitre de présentation des outils de diagnostic disponibles intégrant une section relative à l'utilitaire de traçage en cours Tivoli Workload Scheduler, qui montre

comment fonctionne l'utilitaire et qui explique comment le personnaliser afin d'améliorer les performances du poste de travail.

Les activités de traçage de WebSphere Application Server peuvent également avoir des répercussions sur les performances.

Journalisation

Les performances d'un poste de travail dépendent de la manière dont le mécanisme de journalisation de Tivoli Workload Scheduler utilise la mémoire. Les paramètres par défaut appliqués à cette version assurent des performances optimales. Toutefois, ces valeurs par défaut sont différentes de celles des versions précédentes, et si vous rencontrez des problèmes de performances vous devez vous assurer que ces paramètres n'ont pas été remplacés par les valeurs précédentes. Dans le chapitre consacré aux outils de diagnostic de *Tivoli Workload Scheduler - Guide d'identification des problèmes*, une section évoque CCLog et, outre la discussion relative à la personnalisation de CCLog, décrit le contrôle des valeurs de traitement par défaut de CCLog.

Maintenance de la base de données

Maintenir une bonne organisation de la base de données est important pour optimiser les performances. Pour plus d'informations, voir «Réorganisation de la base de données», à la page 317.

Dimensionnement du fichier Symphony

Pour calculer la taille du fichier Symphony et comprendre son impact sur les performances, voir «Eviter les systèmes de fichiers complets», à la page 318.

Optimisation d'un gestionnaire de domaine UNIX pour la gestion d'un grand nombre de agent tolérant aux panness

Les performances des gestionnaires de domaine sous UNIX peuvent diminuer s'ils servent un nombre important d'agent tolérant aux pannes. Pour limiter les pertes de performances, vous pouvez modifier les paramètres du noyau. Les paramètres diffèrent en fonction du système d'exploitation et vous devrez en tester plusieurs pour obtenir les performances optimales.

Voici un exemple de paramètres de noyau pour HP-UX permettant de gérer approximativement 200 agents tolérants aux pannes :

```
max_thread_proc=256
nprocess=1800
maxusers=120
maxuprc=1700
nflocks=500
maxfiles=1024
```

Optimisation du traitement des travaux sur un poste de travail

Cette section explique le réglage des options sélectionnées dans le fichier Tivoli Workload Scheduler `localopts` pour améliorer les performances de Tivoli Workload Scheduler. Ces options contrôlent la période entre les instances successives d'une activité. Le tableau 75, à la page 449 indique les activités à

optimiser, l'option correspondante à définir dans le fichier `localopts` et décrit l'impact de la valeur modifiée sur les performances.

Tableau 75. Options d'optimisation du traitement des travaux sur un poste de travail

Activité	Option	Impact sur les performances
batchman analyse régulièrement le fichier `Symphony` pour rechercher les travaux prêts à être traités.	***bm look***	Dans tous ces cas, un temps plus court est synonyme d'analyses plus fréquentes, de consommation plus importante des ressources de l'unité centrale et de répercussions sur les autres processus en cours. Cependant, cela signifie également que les délais d'attente de toutes les activités sont réduits au minimum. Si le débit est important et si le poste de travail dispose d'une grande quantité de mémoire, essayez de réduire ces délais. Une période plus longue entre les activités successives implique une attente plus longue entre les travaux et donc une exécution plus longue. Toutefois, des analyses moins fréquentes permettent de préserver une quantité de mémoire plus importante pour les travaux, puisque les activités de surveillance en consomment moins. Examinez les implications des diverses options. Si votre objectif est d'exécuter les travaux le plus rapidement possible mais que la vitesse à laquelle les informations sur les travaux terminés sont distribuées n'a pas d'importance, vous pouvez réduire les délais d'attente pour *bm look* et *jm read* et les augmenter pour d'autres paramètres. Sinon, pour accélérer le traitement général des travaux (depuis le lancement initial du travail jusqu'à la mise à jour avec état d'achèvement) vous pouvez modifier *bm look*, *jm look* et *mm read*.
jobman accède au fichier `Courier.msg` pour voir si des travaux doivent être lancés.	***jm read***	
Après avoir lancé un travail, **jobman** vérifie régulièrement l'état d'achèvement des travaux.	***jm look***	
mailman recherche régulièrement les travaux terminés dans `Mailbox.msg`.	***mm read***	
batchman vérifie régulièrement la présence de travaux terminés dans `Intercom.msg`, de façon à pouvoir mettre à jour le fichier `Symphony`.	***bm read***	

Si vous décidez de régler ces paramètres, procédez comme suit :

- Testez le résultat dans un système de test avant d'apporter des modifications à votre environnement de production. Pour obtenir des résultats valables, l'environnement de test doit avoir les mêmes caractéristiques que l'environnement de production.
- Modifiez les paramètres nécessaires. Mieux vaut les modifier un par un et constater les changements de performances plutôt que de les changer tous à la fois.
- Faites une copie de secours du fichier `localopts` pour être sûr de pouvoir revenir aux options par défaut si nécessaire.

Arrêtez et démarrez l'agent pour activer les modifications appliquées au fichier `localopts`.

Optimisation de la base de données

Pour apprendre à optimiser la base de données, consultez la documentation produit adéquate :

DB2 Allez sur http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/index.jsp, et sélectionnez **Best practices**.

Oracle Voir le manuel *Performance Tuning Guide* dans la documentation Oracle.

Optimisation de la réplication du fichier Symphony dans la base de données

Optimisation des paramètres de configuration de la base de données DB2 afin d'améliorer les performances lorsque le fichier Symphony est répliqué dans la base de données.

Dans un environnement Tivoli Workload Scheduler où la soumission de plus de 200 000 travaux est planifiée, plusieurs paramètres de configuration de la base de données DB2 peuvent être optimisés afin d'améliorer les performances lorsque le plan Symphony est répliqué dans la base de données Tivoli Workload Scheduler.

Les valeurs suggérées pour un plan de plus de 200 000 travaux sont les suivantes :

```
LOGBUFSZ = 2150
DBHEAP = AUTOMATIC (ou supérieur à LOGBUFSZ)

LOGFILSIZ = 10000
LOGPRIMARY = 80
LOGSECOND = 40
```

De plus, augmentez la taille du paramètre **TWS_PLN_BUFFPOOL** à 182000 à l'aide de la commande **ALTER BUFFERPOOL**.

Avant de modifier l'une de ces valeurs, reportez-vous aux information relatives à l'optimisation d'une base de données DB2 dans la documentation produit appropriée, à l'adresse suivante http://pic.dhe.ibm.com/infocenter/db2luw/v10r1/index.jsp.

Optimisation de la WebSphere Application Server

Pour apprendre à optimiser la WebSphere Application Server, consultez la documentation adéquate.

Accédez à http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/index.jsp, sélectionnez **WebSphere Application Server (systèmes d'exploitation distribués), Version 8.5**, puis **Optimisation des performances**.

Taille de segment Java inadéquate

La valeur par défaut de taille maximale du segment de Java peut être trop petite pour vos besoins. Si vous avez des raisons de soupçonner les performances de la WebSphere Application Server, augmentez la taille du segment comme décrit dans «Augmentation de la taille de segment du serveur d'applications», à la page 456.

Nombre trop élevé de soumissions manuelles de travaux

Tivoli Workload Scheduler est conçu pour une efficacité maximale lorsqu'il gère des travaux soumis à l'aide d'un plan programmé. Par conséquent, il est moins adapté au traitement des travaux soumis manuellement. De ce fait, les performances peuvent être améliorées en réduisant le nombre de travaux soumis manuellement.

Nombre trop élevé de contrôles de dépendance de fichier

Chaque contrôle de dépendance de fichier a un impact sur les performances. Si vous concevez un plan qui contrôle constamment les dépendances de fichier, vous réduisez les performances du poste de travail sur lequel ces travaux sont exécutés.

Si plusieurs fichiers “opens” sont utilisés en tant que dépendance, utilisez l'option “–a” (and). Par exemple, pour vérifier l'existence des trois répertoires principaux `/tom`, `/dick` et `/harry`, exécutez la commande suivante avant de lancer `myjob` :

```
job2 opens "/users" (-d %p/tom -a -d %p/dick -a -d %p/harry)
```

Ceci a pour effet de vérifier simultanément la présence de ces trois répertoires, au lieu de les examiner un par un.

D'autres facteurs ayant un impact sur les performances lors de l'évaluation des dépendances de fichier sont les paramètres **bm check** du fichier `localopts`. Ces éléments sont documentés dans *Tivoli Workload Scheduler - Guide de planification et d'installation*, dans le chapitre consacré à la personnalisation.

Répartition de la charge de travail

Quels que soient les travaux à planifier, essayez de les répartir tout au long de la période de production de façon qu'il n'y ait pas de concentration à un moment donné. Essayez également d'éviter de planifier des activités au moment où le trafic utilisateur normal est important sur le réseau, par exemple le matin lorsque les utilisateurs commencent à travailler et ouvrent leurs e-mails.

Si vous ne prenez pas cette précaution, vous risquez de créer un goulet d'étranglement au niveau de la file d'attente Mailbox.msg, susceptible de retarder la mise à jour du fichier Symphony, ce qui crée à son tour des retards dans la disponibilité des statuts de travaux pour **conman** ou Dynamic Workload Console.

Amélioration des performances de traitement des travaux

Le traitement et la surveillance des travaux d'un poste de travail sont contrôlés principalement par plusieurs paramètres du fichier `localopts` et par les options générales gérées par **optman**. Ces paramètres sont décrits dans *Tivoli Workload Scheduler - Guide de planification et d'installation*.

En cas de baisse de performances lors du traitement et de la surveillance des travaux, contactez le service de support logiciel IBM pour obtenir des conseils sur la façon d'optimiser ces paramètres dans votre environnement particulier pour améliorer les performances.

Mise en cache de la boîte aux lettres - avantages et inconvénients

Mailman utilise le paramètre *mm cache mailbox* du fichier `localopts` pour savoir s'il doit mettre en cache les messages de la boîte aux lettres. Cette section explique les avantages et les inconvénients de l'activation et de la désactivation de ce paramètre.

Définition du paramètre *mm cache mailbox* **sur** *no*

Ceci signifie que mailman doit effectuer une action de lecture individuelle pour chaque message avant de le traiter, puis une action de suppression individuelle une fois le message traité. L'activité d'E-S pour le traitement individuel de ces messages est proportionnelle à la quantité de données

lues. Ce facteur diminue les performances. D'un autre côté, le traitement est simple puisque chaque message est lu, traité puis supprimé de la boîte aux lettres. Tout incident système entraîne la relecture d'au plus un seul message et évite toute perte de donnée.

Définition du paramètre *mm cache mailbox* **sur** *yes* **(valeur par défaut)**
Ceci signifie que mailman lit un bloc de messages dans la mémoire cache, traite la totalité des messages puis les supprime tous de la boîte aux lettres. L'avantage en terme de délais d'E-S est évident. La lecture et la suppression d'un ensemble séquentiel de messages en une seule action est plus efficace au niveau des E-S, que de lire et supprimer les messages un par un. Les performances sont ainsi améliorées.

Toutefois, en cas d'incident avec mailman ou le système d'exploitation, le cache est perdu. Au redémarrage, mailman relit l'ensemble des messages précédemment mis en cache, dont certains ont peut-être déjà été traités. Par exemple, si mailman lit un bloc de 32 messages en cache et en a traité 30 au moment où l'incident survient, il relit ces 32 enregistrements lors de son redémarrage, et doit traiter 30 doublons avant de pouvoir reprendre où il s'était arrêté.

La plupart des événements gèrent les modifications de l'état des travaux, et ces événements peuvent être répétés sans créer d'incident. Le mécanisme des événements critiques est alors capable de gérer les autres. Toutefois, les performances diminuent pendant ce processus de reprise et si les mécanismes intégrés ne sont pas capables de gérer les messages en double, une erreur plus grave peut se produire, pouvant entraîner au pire la perte partielle ou totale du contenu de la boîte aux lettres.

Le nombre de messages lus en une action est configurable à l'aide du paramètre *mm cache size*. Par défaut, ce paramètre est défini sur 32 messages, et le maximum sur 512. Le fait de définir ce paramètre sur une valeur supérieure à la valeur par défaut augmente les performances lorsque le fonctionnement est normal, mais diminue les performances en cas d'incident, pour les raisons indiquées ci-dessus. De plus, la mémoire cache supplémentaire signifie que la mémoire requise par le moteur Tivoli Workload Scheduler augmente aussi. Si un poste de travail dispose d'une quantité de mémoire limitée ou exécute des applications très gourmandes, il peut s'avérer négatif d'augmenter la mémoire cache de la boîte aux lettres, car le système d'exploitation devra peut-être démarrer la pagination de la mémoire cache.

En conclusion, la valeur par défaut optimise les performances. Vous ne devez le définir sur *no* que si vous commencez à perdre des événements.

Définition du paramètre de niveau de synchronisation

Cette section décrit l'impact des différents réglages du paramètre *synch level* dans le fichier `localopts`. Le paramètre *synch level* n'a d'impact que sur les environnements UNIX.

L'activité d'E-S effectuée par le moteur du Tivoli Workload Scheduler dans le cadre de la gestion des plans, des flots de travaux, et des travaux consiste à lire et à écrire dans le fichier `Symphony` et les fichiers d'événement (`Mailbox.msg`, `Intercom.msg` et `Courier.msg`). Lorsque Tivoli Workload Scheduler écrit dans ces fichiers, il doit effectuer plus qu'une simple opération *write*. Par exemple, lorsqu'il écrit dans le fichier `Mailbox.msg`, il effectue les actions décrites par le pseudo-code suivant :

```
TWS_write_event_lock {
     Lock Mailbox to write
}

TWS_write_event_update {
     Check Available Space
     Write Header
     Write Record
     Update Write Pointer
     Unlock Mailbox
}
```

Chaque action exige un ou plusieurs accès en écriture au disque. Ces actions sont réalisées de la façon suivante avec les différentes options de niveau de synchronisation :

synch level = high
: Chaque opération d'écriture sur les fichiers d'événements est immédiatement écrite physiquement sur le disque. L'impact sur les performances est important en raison d'une forte dépendances avec les E-S.

synch level = medium
: Chaque événement d'écriture est considéré comme une seule opération. Par exemple, alors que `TWS_write_event_lock` contient une seule action, `TWS_write_event_update` en comprend cinq. Avec `synch level` défini sur *medium*, les cinq actions de cet événement d'écriture sont réalisées en un seul accès au disque, réduisant ainsi de façon notable les E-S.

synch level = low (valeur par défaut)
: Le système d'exploitation décide de la façon et du moment de la synchronisation des données sur le disque. L'impact de cette option est plus difficile à évaluer, car les règles sont différentes pour chaque système d'évaluation et système de fichiers.

Impact du gestionnaire de basculement tolérant aux pannes sur les performances

Cette section décrit l'impact de l'activation du gestionnaire de basculement tolérant aux pannes sur les performances de l'architecture générale et du système individuel. Vous pouvez activer le gestionnaire de basculement tolérant aux pannes en définissant l'option globale `enSwfaultTol` sur *yes*. Dans ce cas, le gestionnaire de domaine maître distribue les messages à tous les agent tolérant aux pannes avec *FullStatus* défini sur *yes*. Cette option n'a pas de fonctions dynamiques et n'est pas conçue pour fonctionner avec des agents de courtier.

L'activation de cette option a un impact sur les éléments suivants :

- Trafic réseau
- Espace disque

Remarque : La fonction gestionnaire de basculement tolérant aux pannes d'est disponible que si tous les postes de travail du domaine sont en version 8.2 avec le groupe de correctifs de niveau 4 ou supérieur.

Trafic réseau

Le trafic réseau reste inchangé dans des conditions normales, mais augmente pendant la phase de réexécution, en fonction de votre choix et uniquement dans des conditions particulières.

La phase de réexécution est une partie essentielle du traitement effectué par la commande **switchmgr**. Elle se produit lorsque le nouveau gestionnaire de domaine traite son fichier Symphony en fonction de ses copies des messages reçus, lorsqu'il tente de mettre à jour sa copie du fichier Symphony.

Dans des conditions normales, la fiabilité sortante ne crée pas de trafic réseau supplémentaire car les messages sont uniquement conservés pour une éventuelle opération de réexécution. Les multiples connexions entrantes ne génèrent pas de trafic supplémentaire car le trafic qui avait été précédemment copié par le gestionnaire de domaine dans le membre *FullStatus* est à présent copié directement dans les membres *FullStatus* par les agents tolérants aux pannes.

Pendant la phase de réexécution, le protocole de connexion démarré par le gestionnaire de domaine de secours inclut une nouvelle phase de réexécution des messages non envoyés par le gestionnaire de domaine en panne. L'impact de la réexécution du message peut être important, en fonction du nombre de messages consignés dans l'ancien gestionnaire de domaine.

Espace disque

L'utilisation de l'espace disque augmente en deux points du réseau suite à l'activation de la tolérance aux pannes supplémentaire.

Les points concernés sont les suivants :

- Sur un seul agent tolérant aux pannes. Ici, en plus de la file d'attente `tomaster.msg`, de nouvelles files sont créées pour les autres agents tolérants aux pannes à *FullStatus*. Vous n'avez pas à tenir compte de ces files d'attente car l'impact sur un seul agent est négligeable.
- Sur les agents tolérants aux pannes à *FullStatus* agissant en tant que gestionnaires de domaine de secours. De nouveaux fichiers de messages ftbox sont créés. Le trafic montant vers le gestionnaire de domaine supérieur figure dans `ftbox/ftup.msg`, et le trafic descendant vers le gestionnaire de domaine inférieur dans `ftbox/ftdown.msg`.

Evolutivité

Dans un environnement comportant un grand nombre d'objets de planification, les impacts sont les suivants :

- «Impact sur **JnextPlan**», à la page 455
- «Impact sur la génération de rapports», à la page 455
- «Impact sur le déploiement de la règle d'événement», à la page 455

La résolution de ces incidents implique souvent d'apporter les modifications suivantes :

- «Augmentation de la taille de segment du serveur d'applications», à la page 456
- «Augmentation de la capacité maximale du journal DB2», à la page 457

Impact sur JnextPlan

L'impact principal sur la performance provoqué par un important réseau de postes de travail exécutant de nombreux travaux sur une période de production de plusieurs jours se ressent sur **JnextPlan**. Le facteur clé est le nombre d'instances de Planificateur de travaux que le **JnextPlan** doit gérer. **JnextPlan** doit traiter chacune de ces instances et le temps nécessaire pour ce faire est un facteur qui ne peut être réduit qu'en s'assurant que le gestionnaire de domaine maître et la base de données se trouvent sur les ordinateurs les plus puissants possible, et que les communications, qu'elles soient locales ou distantes, entre le gestionnaire de domaine maître et la base de données sont optimisées.

Toutefois, certaines mesures spécifiques doivent être prises à mesure que le nombre de travaux ou d'instances du Planificateur de travaux augmente.

Le plan contient plus de 40.000 travaux

Dans ce cas, vous devez augmenter la taille de segment Java utilisée par le serveur d'applications. La valeur par défaut est 512 Mo et vous devez au moins la doubler lorsque le nombre de travaux dépasse les 40.000. Suivez la procédure décrite dans «Augmentation de la taille de segment du serveur d'applications», à la page 456.

Vous disposez d'un grand nombre d'instances de Planificateur de travaux dans le plan

DB2 Les fichiers journaux de transaction DB2 par défaut ne peuvent pas gérer davantage de transactions que celles qui sont générées par environ 180 000 instances de Planificateur de travaux. Vous devez modifier les paramètres qui contrôlent les tailles ou les nombres de fichiers journaux à créer, voire les deux. Suivez la procédure décrite dans «Augmentation de la capacité maximale du journal DB2», à la page 457.

Oracle Le nombre de transactions pouvant être gérées par les fichiers journaux Oracle dépendent de la configuration de la base de données Oracle. Pour plus d'informations, voir la documentation Oracle.

Remarque : Si les circonstances évoluent et que le nombre d'instances de Planificateur de travaux gérées par **JnextPlan** tombe à moins de 180 000, envisagez de redéfinir les paramètres de taille de segment du journal et du serveur d'applications sur leurs valeurs par défaut, afin d'éviter tout problème de performances.

Impact sur la génération de rapports

Lors du traitement d'un rapport, de la mémoire supplémentaire est requise pour traiter un grand nombre d'objets de planification. Le point critique est d'environ 70 000 objets. Cet incident peut être traité en augmentant la taille de segment Java utilisée par le serveur d'applications. Suivez la procédure décrite dans «Augmentation de la taille de segment du serveur d'applications», à la page 456.

Impact sur le déploiement de la règle d'événement

Lors du déploiement d'un grand nombre de règles d'événement, de la mémoire supplémentai est requise. Le point critique est d'environ 8 000 règles. Cet incident peut être traité en augmentant la taille de segment Java utilisée par le serveur d'applications. Suivez la procédure décrite dans «Augmentation de la taille de segment du serveur d'applications», à la page 456.

Augmentation de la taille de segment du serveur d'applications

Suivez cette procédure pour augmenter la taille de segment Java :

1. Connectez-vous à l'ordinateur sur lequel Tivoli Workload Scheduler est installé en tant que l'utilisateur suivant :

   Systèmes d'exploitation Windows :
   Tout utilisateur du groupe *Administrators*.

   Systèmes d'exploitation UNIX :
   root

2. Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413) ou en exécutant :

   Sur les système d'exploitation Windows :
   `TWA_home\wastools\stopWas.bat`

   Systèmes d'exploitation UNIX :
   `TWA_home/wastools/stopWas.sh`

3. Ouvrez le fichier

   `<chemin_profil_WAS>/config/cells/TWSNodeCell/nodes/TWSNode/servers/server1/server.xml`

   où la valeur par défaut pour le chemin *chemin_profil_WAS* est `<TWA_home>/WAS/TWSprofile`.

   Localisez les lignes suivantes :

   ```
   <jvmEntries xmi:id="...."
   verboseModeClass="false"
   verboseModeGarbageCollection="false"
   verboseModeJNI="false"
   initialHeapSize="256"
   maximumHeapSize="1024"
   runHProf="false"
   hprofArguments=""
   debugMode="false"
   debugArgs="...."/>
   </jvmEntries>
   ```

4. Editez les zones `initialHeapSize` et `maximumHeapSize` pour au moins les valeurs indiquées dans le tableau 76.

Tableau 76. Taille de segment de mémoire pour une **machine virtuelle Java 64 bits**

Mémoire vive (Go)	initialHeapSize (Mo)	maximumHeapSize (Mo)
2	512	1024
4	1024	2048
8	2048	4096

   Remarque : Assurez-vous que l'utilisation de la mémoire vive sur l'ordinateur peut accepter l'accroissement de la taille que vous avez indiqué. Si vous disposez d'une **machine virtuelle Java 32 bits** sur votre ordinateur, la valeur maximale pour `maximumHeapSize` est *1536* Mo.

5. Enregistrez le fichier `server.xml`

6. Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413) ou en exécutant :

   Sur les système d'exploitation Windows :
   <*TWA_home*>\wastools\startWas.bat

   Systèmes d'exploitation UNIX :
   <*TWA_home*>/wastools/startWas.sh

Augmentation de la capacité maximale du journal DB2

La base de données Tivoli Workload Scheduler DB2 utilise un journal de transactions dont la taille maximale est essentielle à la réussite de l'exécution de **JnextPlan** sur les bases de données très volumineuses.

Le journal par défaut est composé de 40 fichiers journaux primaires, toujours présents, et de 20 fichiers journaux secondaires, créés à la demande. Chaque fichier faisant environ 4 Mo, la capacité maximale de journalisation en utilisant tous les fichiers journaux "secondaires" en plus des fichiers primaires est de (40 + 20) x 4 Mo = 240 Mo.

L'espace de journal utilisé par **JnextPlan** dépend de la taille du plan de préproduction. 1000 instances environ de Planificateur de travaux génèrent des transactions qui occupent 1 Mo d'espace dans le fichier journal. Par conséquent, les fichiers journaux possèdent par défaut une capacité maximale théorique de 240 000 instances de Planificateur de travaux. Toutefois, dans la pratique, il convient d'allouer un espace au moins 25 % supérieur à celui indiqué par l'algorithme, afin que la capacité par défaut des fichiers journaux soit d'environ 180 000 instances de Planificateur de travaux.

Si **JnextPlan** approche ou dépasse ce niveau, vous devrez libérer davantage d'espace pour le journal DB2.

Outre le calcul ci-dessus, vous pouvez également déterminer l'espace de journalisation réellement utilisé par une instance particulière de **JnextPlan** et baser vos exigences en termes de taille de journal sur ce chiffre.

Détermination de l'utilisation réelle des fichiers journaux DB2

Ci-dessous est présentée la procédure de vérification de la quantité d'espace utilisée par une instance aboutie de la commande **JnextPlan** :

1. Après l'exécution de **JnextPlan**, connectez-vous à l'ordinateur sur lequel le serveur Tivoli Workload Scheduler DB2 est installé, en tant que propriétaire de l'instance DB2 (UNIX) ou en tant qu'administrateur DB2 (Windows).
2. Ouvrez une fenêtre de ligne de commande DB2 ou de shell de la façon suivante :

   UNIX Procédez comme suit :

   a. Lancez la commande **su - db2inst1** ou accédez au sous-répertoire sqllib du répertoire racine du propriétaire de l'instance DB2 (par défaut, *db2inst1*)
   b. Exécutez la commande **. ./db2profile**

   Windows
   Dans le menu **Démarrer**, sélectionnez **Programmes → IBM DB2 → Outils ligne de commande → Fenêtre Commande**
3. Exécutez la commande suivante :

```
db2 "get snapshot for database on TWS" > snapdb.txt
```

où "TWS" doit être remplacé par le nom réel de la base de données, s'il est différent

4. Ouvrez le fichier `snapdb.txt` et recherchez une section comme la suivante :

```
Log space available to the database (Bytes)= 244315359
Log space used by the database (Bytes)     = 484641
Maximum secondary log space used (Bytes)   = 0
Maximum total log space used (Bytes)       = 581636
Secondary logs allocated currently         = 0
```

La valeur indiquée dans "Maximum total log space used" représente l'espace utilisé pour les journaux DB2. Cet espace doit être alloué à DB2 à l'aide des fichiers journaux principaux. Par conséquent, vous devez modifier le nombre et la taille des fichiers journaux principaux en fonction des exigences minimales.

De plus, il est recommandé d'allouer un espace de journal secondaire à DB2. La moitié du nombre de fichiers journaux secondaires alloués pour les fichiers principaux semble être un bon compromis.

La commande snapshot décrite à l'étape 3, à la page 457 peut être exécutée à tout moment pour conserver une trace de l'utilisation en cours de l'espace de journal DB2, sans impact notable sur les performances. Tous les éléments métriques qui s'affichent sont utiles pour surveiller à tout moment l'allocation en cours des journaux DB2 principaux et secondaires et déterminer toutes les modifications requises.

Procédure de modification de la capacité de journalisation DB2 maximale

Procédez comme suit :

1. Connectez-vous à l'ordinateur sur lequel le serveur Tivoli Workload Scheduler DB2 est installé, en tant que propriétaire de l'instance de DB2 (UNIX) ou en tant qu'DB2 Administrateur (Windows).
2. Ouvrez une fenêtre de ligne de commande DB2 ou de shell de la façon suivante :

   UNIX Procédez comme suit :

   a. Lancez la commande **su - db2inst1** ou accédez au sous-répertoire `sqllib` du répertoire racine du propriétaire de l'instance DB2 (par défaut, *db2inst1*)

   b. Exécutez la commande **. ./db2profile**

   Windows

   Dans le menu **Démarrer**, sélectionnez **Programmes** → **IBM DB2** → **Outils ligne de commande** → **Fenêtre Commande**

3. Exécutez les commandes suivantes :

```
db2 update db cfg for <database_name> using LOGFILSIZ <log_file_size>
db2 update db cfg for <database_name> using LOGPRIMARY <primary_log_files>
db2 update db cfg for <database_name> using LOGSECOND <secondary_log_files>
```

Où :

<database_name>

Nom de la base de données :

- Si l'exécution s'effectue à partir de l'ordinateur sur lequel le serveur DB2 est installé, le nom par défaut installé est *TWS*. Indiquez cette valeur à moins que vous ne l'ayez modifiée.

- Il est déconseillé d'exécuter cette procédure sur l'ordinateur où est installé le client DB2, mais si vous le faites, le nom par défaut installé est *TWS_DB*. Indiquez cette valeur à moins que vous ne l'ayez modifiée.

<taille_fichier_journal>
Taille du fichier journal en pages de 4 Ko. La valeur par défaut est de 1000 (d'où la taille du fichier journal par défaut de 4 Mo). Consultez la documentation DB2 pour plus d'informations sur les implications liées au choix d'une taille de fichiers journaux différente. La valeur maximale est de 262 144 (pour une taille de fichier journal maximale d'1 Go).

<fichiers_journaux_primaires>
Nombre de fichiers journaux primaires. La valeur par défaut est 40. Le nombre maximum total de fichiers journaux que DB2 peut traiter (primaires ou secondaires) est de 256. Par conséquent, le journal est limité à 256 Go, soit environ 256 millions d'instances de Planificateur de travaux ! (256 fichiers max x taille de fichier max de 1 Go)

<fichiers_journaux_secondaires>
Nombre de fichiers journaux secondaires. La valeur par défaut est 20. Si l'espace disponible sur le système de fichiers est suffisant, ces fichiers journaux supplémentaires sont alloués de manière dynamique par DB2 au besoin (ce qui entraîne un léger impact sur la performance de **JnextPlan**). Compte tenu du fait qu'ils ne sont créés qu'en cas de besoin, il est préférable d'augmenter le nombre de fichiers secondaires plutôt que le nombre de fichiers primaires. En général, vous allouez 50 % de la valeur du fichier journal principal.

Lors su calcul de l'allocation de fichiers journaux, allouez au moins 25 % d'espace en plus de celui dont vous pensez avoir besoin afin d'éviter qu'un calcul légèrement erroné ne provoque la défaillance de **JnextPlan**.

Exemple : si vous avez déterminé, en fonction de la procédure décrite dans «Détermination de l'utilisation réelle des fichiers journaux DB2», à la page 457, que **JnextPlan** utilise actuellement 320 Mo, vous pouvez procéder au calcul suivant :

a. Augmentez 320 Mo de 25 %, ce qui donne 400 Mo
b. Déterminez si vous voulez plus de fichiers journaux et/ou des fichiers journaux plus volumineux, par référence à la documentation DB2. Par exemple, vous pouvez choisir d'allouer 40 fichiers de 10 Mo, 80 fichiers de 5 Mo ou 100 fichiers de 4 Mo. Pour cet exemple, supposons que vous avez choisi d'allouer 80 fichiers de 5 Mo. La valeur de LOGPRIMARY est donc de 80.
c. Déterminez la taille du fichier journal en pages de 4 Ko donnant une taille de fichier journal de 5 Mo - la valeur de LOGFILSIZ est donc de 1250.
d. Déterminez le nombre de fichiers journaux secondaires requis. Si vous vous conformez aux 50 %, la valeur de LOGSECOND est 40.

4. Connectez-vous à l'ordinateur sur lequel Tivoli Workload Scheduler est installé en tant que l'utilisateur suivant :

UNIX root

Windows
Tout utilisateur du groupe *Administrators*.

5. Accédez au répertoire : `<TWA_home>/wastools`

6. Arrêtez WebSphere Application Server à l'aide de la commande **conman stopappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)
7. Sur l'ordinateur où est installé le serveur DB2, arrêtez et redémarrez DB2 de la façon suivante :
   a. Assurez-vous qu'aucune autre application n'utilise cette instance de DB2, ou si c'est le cas, qu'elles peuvent être arrêtées.
   b. Emettez la commande suivante :
      ```
      db2stop
      ```
   c. Emettez la commande suivante :
      ```
      db2start
      ```

   Remarque : Il est vivement conseillé d'arrêter et démarrer DB2. Si cela vous pose problème, vous devez au moins déconnecter toutes les applications de l'instance DB2 et les reconnecter. DB2 appliquera de nouveaux paramètres lors de la reconnexion. Le cas échéant, utilisez la commande suivante pour forcer la déconnexion de toutes les connexions ouvertes :
   ```
   db2 "force application all"
   ```
8. Démarrez WebSphere Application Server à l'aide de la commande **conman startappserver** (voir «Démarrage et arrêt du serveur d'applications et de **appservman**», à la page 413)

Rapports multiples de plan de production Dynamic Workload Console

A partir de Dynamic Workload Console, vous pouvez lancer des rapports de plan de production. Ils utilisent une quantité importante de temps UC. S'ils sont demandés pour l'ensemble du plan, leur génération peut également prendre beaucoup de temps. L'exécution simultanée de plusieurs rapports peut avoir un impact important sur les performances du gestionnaire de domaine maître.

Si vous remarquez une baisse des performances, vous pouvez déterminer si des rapports sont en cours d'exécution en vérifiant les fichiers de travail des rapports de la façon suivante :

1. Accédez au répertoire temporaire du système d'exploitation
2. Recherchez les fichiers qui utilisent le modèle de nom suivant :
   ```
   TWS-<sequential_number>-extr
   ```

   Un de ces fichiers de travail est ouvert pour chaque rapport en cours d'exécution. Les fichiers sont supprimés une fois le rapport terminé.
3. Vérifiez les dates de ces fichiers, et tenez compte uniquement des fichiers récents (si un rapport échoue à tout moment de la production, son fichier reste dans le répertoire temporaire jusqu'au redémarrage suivant du gestionnaire de domaine maître ou jusqu'à ce que vous procédiez à un nettoyage du système d'exploitation qui supprime tous les fichiers du répertoire temporaire).

Il n'existe aucune action directe à effectuer, étant donné que vous devez attendre que le rapport s'achève pour que les performances s'améliorent.

Toutefois, si vous remarquez que de nombreux rapports sont émis, vous êtes peut-être dans le cas suivant :

1. Un utilisateur émet une demande de rapport et s'attend à ce qu'il soit disponible immédiatement

2. Lorsque le rapport ne s'affiche pas immédiatement, l'utilisateur pense qu'un incident s'est produit, ferme et ré-ouvre le navigateur pour relancer le rapport. Le fait de fermer le navigateur n'interrompt pas la production du rapport.
3. Il arrive que l'utilisateur répète cette opération plusieurs fois.

Dans ce cas, vous devez rappeler à l'utilisateur que la production de rapports volumineux demande beaucoup de temps, et qu'il doit être patient.

Dynamic Workload Console - ajustement des paramètres relatifs au délai d'attente de la session

La valeur affectée aux paramètres relatifs au délai d'attente permet de définir la durée (en minutes) au bout de laquelle un utilisateur est automatiquement déconnecté de WebSphere Application Server. Si vous comptez exécuter des opérations de longue durée, connecter simultanément plusieurs utilisateurs à Dynamic Workload Console ou encore obtenir de faibles performances sur le système sur lequel Dynamic Workload Console est installée, vous pouvez être amené à augmenter ces valeurs.

Procédez comme suit pour modifier les valeurs affectées aux paramètres relatifs au délai d'attente :

1. Ouvrez le fichier de configuration :

   ```
   <rép_profil_JazzSM>\config\cells\JazzSMNode01Cell\
              nodes\JazzSMNode01\servers\server1\server.xml
   ```

   où la valeur par défaut de *<rép_profil_JazzSM>* est :

   Sur les systèmes d'exploitation Windows

   ```
   C:\Program Files\IBM\JazzSM\profile
   ```

   Sur les systèmes d'exploitation UNIX

   ```
   /opt/IBM/JazzSM/profile
   ```

2. Dans le fichier, recherchez le paramètre `invalidationTimeout` dans la balise suivante :

   ```
   <tuningParams xmi:id="TuningParams_1188622510500"
                       usingMultiRowSchema="false"
                       maxInMemorySessionCount="1000"
                       allowOverflow="true"
                       scheduleInvalidation="false"
                       writeFrequency="TIME_BASED_WRITE"
                       writeInterval="10"
                       writeContents="ONLY_UPDATED_ATTRIBUTES"
                       invalidationTimeout="30">
   ```

   Il s'agit du paramètre définissant le délai d'attente de la session HTTP. Par défaut, `invalidationTimeout` est défini sur 30, ce qui signifie qu'un utilisateur est automatiquement déconnecté au bout de 30 minutes d'inactivité.
3. Définissez le paramètre `invalidationTimeout` sur une valeur qui convient pour votre environnement et les activités que vous prévoyez d'effectuer.
4. Enregistrez le fichier.
5. Ouvrez le fichier de configuration :

   ```
   <rép_profil_JazzSM>\config\cells\JazzSMNode01Cell\applications\isclite.ear\
                         deployments\isclite\deployment.xml
   ```

6. Dans le fichier, recherchez le paramètre `invalidationTimeout` dans la balise suivante :

```
<tuningParams xmi:id="TuningParams_1188878529796"
                      usingMultiRowSchema="false"
                      maxInMemorySessionCount="1000"
                      allowOverflow="true"
                      scheduleInvalidation="false"
                      writeFrequency="TIME_BASED_WRITE"
                      writeInterval="10"
                      writeContents="ONLY_UPDATED_ATTRIBUTES"
                      invalidationTimeout="30">
```

Par défaut, `invalidationTimeout` est défini sur 30, ce qui signifie qu'un utilisateur est automatiquement déconnecté au bout de 30 minutes d'inactivité.

7. Définissez le paramètre `invalidationTimeout` sur une valeur qui convient pour votre environnement et les activités que vous prévoyez d'effectuer.
8. Enregistrez le fichier.
9. Ouvrez le fichier de configuration :

   `<rép_profil_JazzSM>\config\cells\JazzSMNode01Cell\security.xml`
10. Dans le fichier, recherchez le paramètre `timeout` dans la balise suivante :

```
<authMechanisms xmi:type="security:LTPA"
            xmi:id="LTPA_1" OID="oid:1.3.18.0.2.30.2"
            authContextImplClass="com.ibm.ISecurityLocalObjectTokenBaseImpl
.WSSecurityContextLTPAImpl"
            authConfig="system.LTPA"
            simpleAuthConfig="system.LTPA"
            authValidationConfig="system.LTPA"
            timeout="120"
            keySetGroup="KeySetGroup_lab237165Node01_1">
```

Par défaut `timeout` est défini sur 120, ce qui signifie qu'un utilisateur est automatiquement déconnecté au bout de 120 minutes d'inactivité qu'il ait ou non effectué des actions sur WebSphere Application Server.

11. Dans la section suivante du fichier, définissez le délai d'attente sur une valeur qui convient pour votre environnement et les activités que vous prévoyez effectuer.
12. Enregistrez le fichier.
13. Redémarrez le serveur WebSphere Application Server.

Chapitre 12. Disponibilité

Ce chapitre décrit les facteurs qui peuvent affecter la disponibilité de Tivoli Workload Scheduler sur un poste de travail. Il traite les sujets suivants :

Résolution du compte utilisateur sous le système d'exploitation Windows

Tivoli Workload Scheduler doit résoudre le compte utilisateur sur les systèmes d'exploitation Windows pour vérifier les informations de sécurité.

Les utilisateurs Windows peuvent être classés en utilisateurs de domaine ou utilisateurs locaux. Les utilisateurs de domaine sont définis dans le contrôleur de domaine, tandis que les utilisateurs locaux sont définis dans les postes de travail du réseau.

Pour un utilisateur de domaine, Tivoli Workload Scheduler demande au contrôleur de domaine principal (ou à tout contrôleur de domaine pour Windows 2000 ou 2003 Active Directory) d'identifier un contrôleur de domaine disponible. Il utilise alors cette identité de contrôleur de domaine pour compléter la structure de l'utilisateur.

Pour un utilisateur local, Tivoli Workload Scheduler s'adresse au poste de travail local. Généralement, Tivoli Workload Scheduler spécifie deux cas : un pour l'utilisateur Tivoli Workload Scheduler et un pour l'utilisateur streamlogon.

Voici la liste des étapes effectuées par Tivoli Workload Scheduler pour authentifier les utilisateurs Windows ainsi que les API concernées :

1. Tivoli Workload Scheduler rechercher l'utilisateur dans le domaine de référence. Pour l'utilisateur de domaine, le domaine de référence correspond au nom du réseau Windows. Pour l'utilisateur local, il s'agit du nom du poste de travail local.

   API : `LookupAccountName`.
2. Si l'utilisateur est un utilisateur de domaine, Tivoli Workload Scheduler demande au contrôleur de domaine d'ordonner à tout contrôleur de domaine disponible de résoudre le compte pour l'utilisateur du domaine de référence.

   API : `NetGetAnyDCName` pour Windows ou `DsGetDcName` pour Windows 2000 ou 2003.
3. Tivoli Workload Scheduler demande au contrôleur de domaine (ou au poste de travail local si l'utilisateur est au niveau local) des informations sur l'utilisateur.

   API : `NetUserGetInfo`.

   Remarque : Sous Windows 2000 et 2003, les droits d'accès à cette API se trouvent dans le groupe `BUILTIN\"Pre-Windows 2000 compatible access"`.

Utilisation d'un répertoire temporaire sous UNIX

Lorsque vous effectuez des opérations Tivoli Workload Scheduler sous UNIX, des fichiers temporaires sont créés dans le répertoire temporaire du poste de travail local. Assurez-vous que l'<*utilisateur_TWS*> qui exécute ces opérations dispose d'un accès à ce répertoire en *lecture* et en *écriture*.

Remarques

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, programme ou service IBM n'est pas destinée à affirmer ou à laisser entendre que seul ce produit, programme ou service IBM peut être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou avoir des brevets en instance couvrant les produits décrits dans ce document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations
IBM Canada Ltd
3600 Steeles Avenue East
Markham, Ontario
L3R 9Z7 Canada

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit auprès d'IBM à l'adresse suivante :

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales.

LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT" SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Certaines juridictions n'autorisent pas l'exclusion des garanties implicites ou explicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Il est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM contenues dans le présent document sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758
Etats-Unis

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'ICA, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

Marques

IBM, le logo IBM et ibm.com sont des marques d'International Business Machines Corporation aux Etats-Unis et/ou dans d'autres pays. Si ces marques et d'autres marques d'IBM sont accompagnées d'un symbole de marque (® ou ™), ces symboles signalent des marques d'IBM aux Etats-Unis à la date de publication de ce document. Ces marques peuvent également exister et éventuellement avoir été enregistrées dans d'autres pays. La liste actualisée de toutes les marques IBM est disponible sur la page Web "Copyright and trademark information" à l'adresse http://www.ibm.com/legal/copytrade.shtml.

Intel, le logo Intel, Intel Inside, le logo Intel Inside, Intel Centrino, le logo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium et Pentium sont des marques d'Intel Corporation ou de ses filiales aux Etats-Unis et dans certains autres pays.

Java ainsi que tous les logos et toutes les marques incluant Java sont des marques d'Oracle et/ou de ses sociétés affiliées.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

Microsoft, Windows, Windows NT et le logo Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

UNIX est une marque enregistrée de The Open Group aux Etats-Unis et/ou dans certains autres pays.

Index

Caractères spéciaux

Nombres

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

Numéro de programme : 5698-WSH

SC11-6396-05