1 Einleitung

Die unternehmensweite Informationsverarbeitung besteht heute aus vielen Systemen und Anwendungen. Diese sind oftmals nicht in einem Rechenzentrum konzentriert, sondern verteilen sich auf verschiedene Standorte.

1.1 Zentrale Überwachung dezentraler Systeme via SNMP

Während die dezentrale Installation von Servern früher meistens mit einer Verteilung der Betriebsverantwortung verbunden war, wird heute die Betriebsverantwortung zunehmend zentralisiert. Dies setzt voraus, dass die verteilten Systeme und Anwendungen über Kommunikationsverbindungen zentral überwacht und gesteuert werden können und sich die Betreuung der Komponenten (Bridges, Hubs, Router bis hin zu Servern) „vor Ort“ weitgehend ersetzen lässt. Diese Aufgabe wird von zentralen Management-Plattformen erfüllt, die sich aktuelle Informationen über die zu überwachenden Komponenten mithilfe sogenannter Agenten beschaffen. Ein Agent ist die Software, die auf der überwachten Komponente abläuft und die benötigten Informationen über die Komponente liefert. Management-Plattform und Agent kommunizieren über ein festgelegtes Protokoll miteinander. Als Standard hat sich hier das Protokoll SNMP (Simple Network Management Protocol) durchgesetzt. Der SNMP-Standard ermöglicht die Integration eines sehr heterogenen IT-Inventars von verschiedenen Herstellern in ein einheitliches Netzwerk-Management.

1.2 SNMP Management für BS2000/OSD

Mit den Produkten zum SNMP Management des BS2000/OSD bietet Fujitsu Siemens Computers die Voraussetzungen, auch BS2000/OSD-Systeme in das einheitliche Netzwerk-Management einzubeziehen.

Folgende Komponenten werden angeboten:

- Aufseiten der SNMP-Agenten:
  - SNMP-Basisagent
  - zahlreiche produktspezifische Agenten
- Aufseiten der Management-Plattform:
  - Die Anwendungen Console und Application Monitor, Performance Monitor, BCAM Monitor und Cluster Monitor.
  - Das Paket für die Integration des BS2000/OSD-Managements in Unicenter der Firma Computer Associates.

1.3 Zielgruppe

Das vorliegende Handbuch wendet sich an Netzplaner, -verwalter, -operateure und Systembetreuer, die BS2000/OSD-Systeme in ein SNMP-basiertes Netz-, System- und Anwendungsmanagement integrieren bzw. ein solches System bedienen wollen. Kenntnisse des Betriebssystems BS2000/OSD sowie der TCP/IP-Grundbegriffe werden vorausgesetzt.

1.4 Konzept des Handbuchs

Das vorliegende Handbuch ist wie folgt strukturiert:

- Kapitel 2: SNMP im Überblick

  Dieses Kapitel stellt die Grundlagen der SNMP-Architektur vor und gibt einen Überblick über Einsatzbereiche und Funktionalität von SNMP. Danach wird der Aufbau des SNMP-Agenten im BS2000/OSD beschrieben und die Produktstruktur des SNMP Managements für BS2000/OSD dargestellt, in der sich das Master-Subagenten-Prinzip des SNMP-Agenten widerspiegelt. Ein Überblick über die Bedienoberflächen des SNMP Managements im BS2000/OSD schließt das Kapitel ab.

- Kapitel 3: Integration von BS2000/OSD in SNMP

  Dieses Kapitel nennt die Software-Voraussetzungen für die Installation der SNMP Management-Produkte in BS2000/OSD und beschreibt die Installation der SNMP-Agenten.

- Kapitel 4: SNMP-Basisagenten für BS2000/OSD

  Dieses Kapitel beschreibt für die einzelnen SNMP-Basisagenten im BS2000/OSD jeweils Anwendungsbereich, Funktionalität, Konfiguration sowie Start- und Stopp-Kommando.

- Kapitel 5: Produktspezifische Agenten - Funktionale Erweiterungen zu SNMP V6.0

  Dieses Kapitel enthält die vollständige Beschreibung des neuen HIPLEX-Subagenten. Außerdem werden die funktionalen Erweiterungen des *open*UTM-Subagenten gegenüber SSA-OUTM-BS2 V5.0A beschrieben.

- Kapitel 6: SNMP Management

  Dieses Kapitel erläutert die drei Alternativen, die Ihnen für den Zugriff auf Management-Informationen zur Verfügung stehen und sich z.B. durch die notwendigen Systemanforderungen an die Management-Plattform unterscheiden:

  – Zugriff zum SNMP-Agenten über das World Wide Web

  – Management-Anwendungen

  – Integration in Management-Plattformen wie z.B. Unicenter von Computer Associates.

- Kapitel 7: Sicherheitsbewusste Nutzung von SNMP

  Dieses Kapitel beschreibt, was Sie für den sicherheitsbewussten Einsatz von SNMP beachten sollten.

1.5 Erweiterungen gegenüber der Vorgängerversion

Zur Version V6.0 des SNMP Managements für BS2000/OSD gibt es folgende Neuerungen und funktionale Erweiterungen:

- neue Subagenten:
  - Event-Subagent
  - Scheduler-Subagent
  - HIPLEX-Subagent
- funktionale Erweiterungen folgender Subagenten:
  - Console Monitor-Subagent
  - Application Monitor-Subagent
  - Subagent für *open*UTM
- neue Management-Anwendung Cluster Monitor
- Überarbeitung und funktionale Erweiterung folgender Management-Anwendungen:
  - Console Monitor, erweitert um Application Monitor:
    Console und Application Monitor
  - Performance Monitor
- Erweiterung der Integration in CA Unicenter
  - Anpassung an Unicenter NSM V3.0
  - Erweiterung um eine DSM-Policy zum Subagenten für *open*UTM
- erhöhte Sicherheit

  Die SNMP-Agenten sowie die Management-Anwendungen Console und Application Monitor, Performance Monitor und Cluster Monitor unterstützen die Sicherheitskonzepte von SNMPv3, u.a. Authentifizierung, Autorisierung und Zugriffskontrolle aller Abfragen oder Änderungen von Management-Objekten.

1.6 Typografische Gestaltungsmittel

In diesem Handbuch werden folgende Mittel zur Darstellung von funktional wichtigen Textteilen verwendet:

für Hinweistexte

ACHTUNG!

für Warnhinweise

kursive Schrift
: für Dateinamen, Namen von Auftragsfenstern, Parameterbezeichnungen, Menütitel und Menüeinträge sowie Kommandos und Variablen im Fließtext.

<spitze Klammern>
: kennzeichnen Variable, wenn Sie dafür Werte einsetzen müssen.

`dicktengleiche Schrift`
: für die Darstellung von Eingaben für das System, Systemausgaben und für Dateinamen in Beispielen.

kommando
: In der Syntaxbeschreibung für Kommandos werden diejenigen Bestandteile (Bezeichnungen von Kommandos und Parametern) fett dargestellt, die unverändert eingegeben werden müssen.

1.7 Readme-Datei

Funktionelle Änderungen und Nachträge der aktuellen Produktversion zu diesem Handbuch entnehmen Sie bitte ggf. der produktspezifischen Readme-Datei. Sie finden die Readme-Datei auf Ihrem BS2000/OSD-Rechner unter dem Dateinamen `SYSRME.`*`SBA-BS2.060.D`*. Die Benutzerkennung, unter der sich die Readme-Datei befindet, erfragen Sie bitte bei Ihrer zuständigen Systembetreuung. Die Readme-Datei können Sie mit dem Kommando `/SHOW-FILE` oder mit einem Editor ansehen oder auf einem Standarddrucker mit folgendem Kommando ausdrucken:

`/PRINT-DOCUMENT` *`dateiname`*`,LINE-SPACING=*BY-EBCDIC-CONTROL`

2 SNMP im Überblick

SNMP steht für **S**imple **N**etwork **M**anagement **P**rotocol und wurde als Protokoll für Netzmanagement-Dienste im TCP/IP-Internet entwickelt. Die Überwachung und Administration von LAN-Komponenten, wie z.B. Bridges, Routers und Hubs, in heterogenen Netzen mit TCP/IP-Protokollen war ursprünglich die einzige Aufgabe von SNMP. Inzwischen hat sich der Anwendungsbereich von SNMP um System- und Anwendungsmanagement bis hin zum Management von Middleware-Produkten wie Datenbanken und Transaktionsmonitoren erweitert.

Ähnlich wie bei TCP/IP, wo der Protokollname nicht nur die Protokolle selbst, sondern die Infrastruktur und das Rahmenwerk des gesamten TCP/IP-Netzes, d.h. des Internet, bezeichnet, steht auch der Name SNMP nicht nur für das Protokoll allein, sondern für das gesamte auf SNMP basierte Management-System.

Vorteile von SNMP

SNMP ist nicht mehr nur ein Management-Protokoll unter vielen, es ist vielmehr *das* Management-Protokoll in TCP/IP-Netzen. Dies hat u.a. folgende Gründe:

- SNMP ist standardisiert.
- SNMP ist weit verbreitet.
- SNMP erlaubt differenzierten Zugriff.
- SNMP ist leicht zu implementieren.

2.1 SNMP Management-Architektur

SNMP folgt einer Client-/Server-Architektur mit der Management-Plattform als Client und den Management-Agenten als Server (siehe Bild 1).

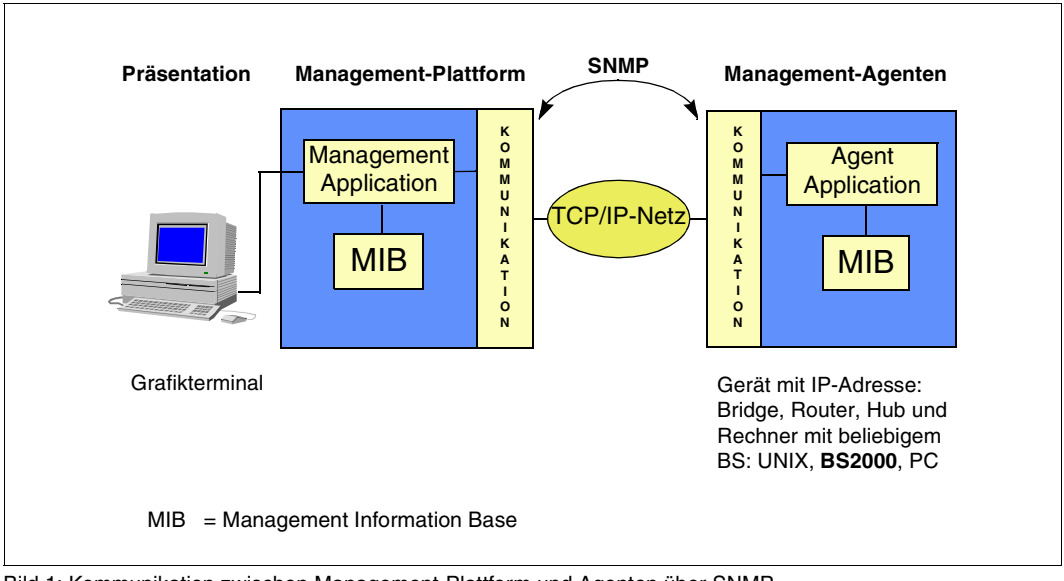

Bild 1: Kommunikation zwischen Management-Plattform und Agenten über SNMP

Management-Plattform

Zentraler Bestandteil einer SNMP-Installation ist die Management-Plattform. Als Leitstand mit Grafikterminals ermöglicht die Management-Plattform eine übersichtliche Darstellung der verwalteten Komponenten und eine komfortable Bedienung. Von der Management-Plattform aus lässt sich das Netz mit all seinen Komponenten, Systemen und Anwendungen überwachen und steuern.

Auf der Management-Plattform residieren SNMP-Manager, auch Management-Applikationen genannt, die via SNMP über ein TCP/IP-Netz mit den SNMP-Agenten kommunizieren. Auf jeder verwalteten Komponente liegt ein SNMP-Agent, der dem SNMP-Manager - auf Anfrage oder spontan - aktuelle Informationen über diese Komponente liefert. Die Initiative zur Steuerung der Aktivitäten liegt überwiegend beim SNMP-Manager, wodurch die Belastung der verwalteten Komponenten mit Management-Aufgaben gering gehalten wird.

SNMP-Manager

Der SNMP-Manager ist die Software, die Anforderungen an die einzelnen Agenten generiert und sie via SNMP an die betreffenden Agenten sendet.

SNMP-Manager empfangen zweierlei Nachrichten von den Agenten:

– Antworten auf gesendete Anforderungen

– Traps: Traps sind asynchrone Nachrichten, die der Agent in bestimmten Situationen ohne Anforderung durch den SNMP-Manager an diesen sendet.

Der SNMP-Manager zeigt die vom Agenten erhaltenen Informationen an und kann mit eigenen Aktionen auf sie reagieren. Das Spektrum der Anzeigemöglichkeiten reicht von der einfachen tabellarischen Ausgabe der Werte bis hin zu einer visuellen Darstellung der überwachten Systeme und Anwendungen in einem Netzbild mit Ereignisanzeige und ausgeprägtem Alarmmanagement.

SNMP-Agent

Ein SNMP-Agent ist die Software, die die vom SNMP-Manager gesendeten Anforderungen empfängt, ausführt und beantwortet. Der Agent hat den direkten Zugang zum überwachten Systemteil bzw. zur überwachten Komponente. In definierten Situationen senden die SNMP-Agenten ohne Anforderung asynchrone Nachrichten (Traps) an den Manager.

Moderne SNMP-Agenten, wie der in BS2000/OSD verwendete Agent, sind nach einem Master-Subagenten-Prinzip strukturiert (siehe Abschnitt „Produktstruktur“ auf Seite 12). Gegenüber einem monolithischen SNMP-Agenten vereinfacht dies die In- und Außerbetriebname von Teilfunktionen (Subagenten).

Management Information Base (MIB)

Für jede zu verwaltende Komponente und somit für jeden Subagenten wird eine eigene MIB benötigt. In der MIB sind die verwaltungsrelevanten Objekte der betreffenden Komponente festgelegt und die Objekteigenschaften beschrieben. Objekteigenschaften sind u.a. Objektname, Syntax, Zugriffsrechte und Status.

Es gibt folgende Arten von MIBs:

– Standard-MIBs, d.h. MIBs, die von Normungsgremien, speziell von denen des Internet, verabschiedet werden. Ein typisches Beispiel ist der Internet-Standard 17, die MIB-II (RFC1213) für TCP/IP-Netze.

– MIBs, die einen de-facto Standard darstellen.

– private MIBs, die hersteller-spezifische Erweiterungen enthalten.

Für viele Hard- und Softwarekomponenten liefert der Hersteller spezifische MIBs mit. Nähere Informationen zur MIB finden Sie im Handbuch „SNMP Management V5.0“.

Sicherheitsmechanismen

Die Berechtigung für einen lesenden oder schreibenden Zugriff des SNMP-Managers wird über einen sog. Community-Namen (Community String) gesteuert. Der Community-Name ist in jeder SNMP-Nachricht enthalten und weist den Absender der Nachricht als Mitglied einer bestimmten Gruppe, der Community, aus. Manager und Agenten dürfen nur miteinander kommunizieren, wenn sie derselben Community angehören.

Dieses relativ einfache Modell wird mit SNMPv3 zu einem umfangreichen Sicherheitskonzept erweitert. Damit können Sie, selbst wenn Sie als Protokoll SNMPv1 einsetzen, in den SNMP-Produkten für BS2000/OSD wesentliche SNMPv3-Funktionen nutzen, wie z.B.

– selektive Vergabe von Zugriffsrechten auf MIB-Variable,
– Festlegung von Zugriffsrechten für eine Gruppe von Management-Plattformen,
– detailliertes Versenden von Traps.

2.2 SNMP-Agent im BS2000/OSD

Moderne SNMP-Agenten, wie der im BS2000/OSD verwendete, sind nach einem Master-Subagenten-Prinzip strukturiert. Die Funktionalität des Agenten verteilt sich dabei wie folgt auf einen Masteragenten sowie einen oder mehrere Subagenten:

– Der Masteragent erledigt zentral die grundlegenden Aufgaben, wie die Abwicklung des SNMP-Protokolls, Sicherheitsfunktionen, Arbeitsverteilung etc.

– Jeder Subagent ist nur für einen bestimmten Teilbereich der überwachten Komponente zuständig. Dabei kommuniziert er nur mit dem SNMP-Masteragenten, der die Kommunikation von/zum SNMP-Manager durchführt.

Die Subagenten sind in sich abgeschlossen und können zu beliebigen Zeitpunkten gestartet und beendet werden. Dies optimiert Performance, Ausfallsicherheit und Skalierbarkeit des gesamten SNMP-Systems.

Die Master-Subagenten-Struktur des SNMP-Agenten spiegelt sich in der Produktstruktur des SNMP Managements für BS2000/OSD wider (siehe nachfolgender Abschnitt „Produktstruktur“).

2.3 Produktstruktur

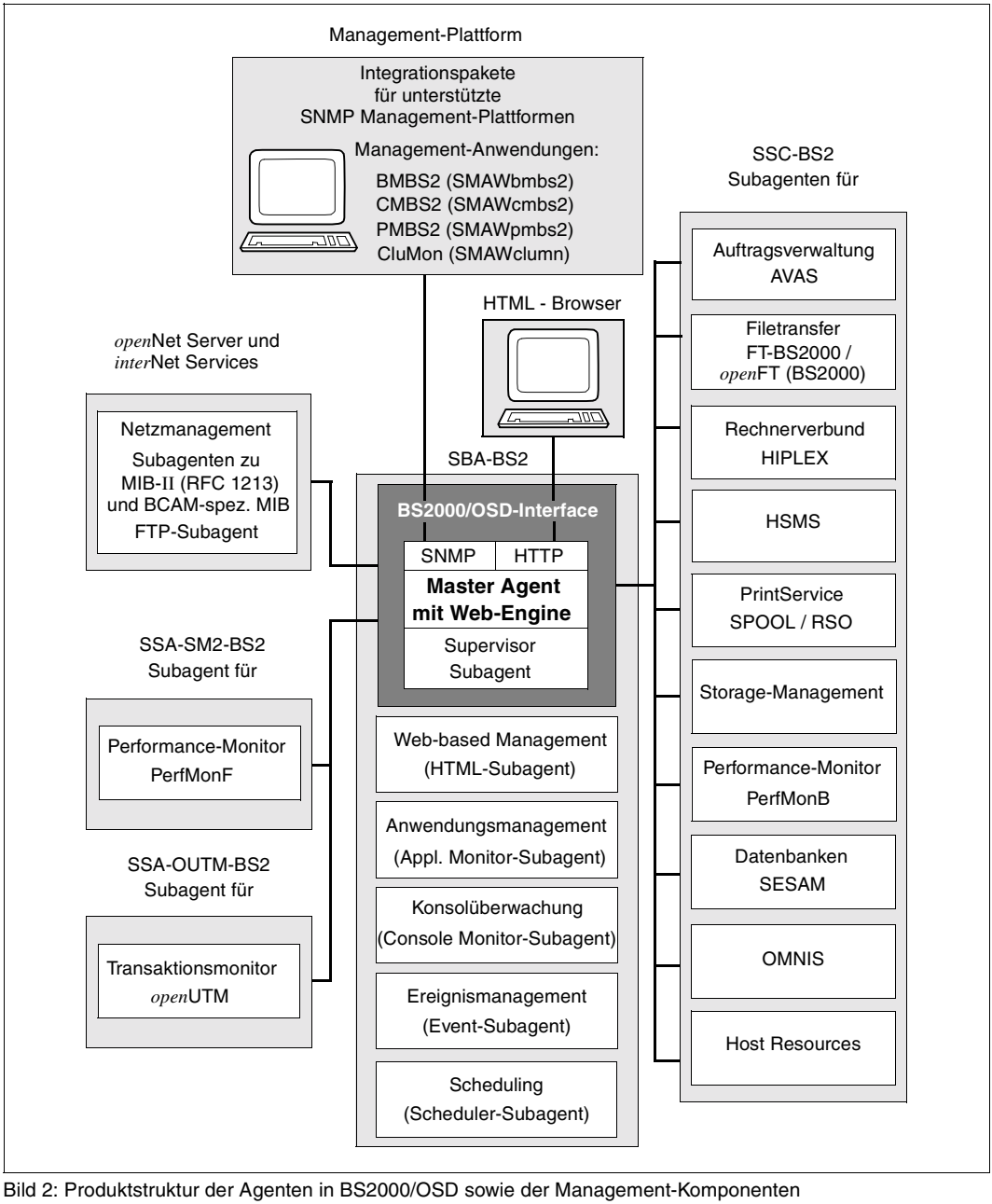

Bild 2: Produktstruktur der Agenten in BS2000/OSD sowie der Management-Komponenten

Die SNMP-Agenten werden in folgenden Liefereinheiten bereitgestellt:

- SBA-BS2 (SNMP-Basic-Agent BS2000)
- SSC-BS2 (SNMP-Standard-Collection BS2000)
- SSA-SM2-BS2 (SNMP-Subagent für den Performance-Monitor SM2)
- SSA-OUTM-BS2 (SNMP-Subagent für *open*UTM in BS2000/OSD)
- SNMP-Subagenten in den Produkten *open*Net Server und *inter*Net Services
- SNMP-Agent für den HNC (Highspeed Net Connector)

SBA-BS2 (SNMP-Basic-Agent BS2000)

Die Liefereinheit SBA-BS2 V6.0 enthält die Basisagenten:

– SNMP-Masteragent
– Supervisor-Subagent
– Application Monitor-Subagent
– Console Monitor-Subagent
– Event-Subagent
– Scheduler-Subagent
– HTML-Subagent

Im Folgenden werden die SNMP-Basisagenten kurz vorgestellt. Ausführliche Informationen zu Funktionalität, Konfiguration und Bedienung der einzelnen Basisagenten finden Sie im Kapitel „SNMP-Basisagenten für BS2000/OSD“ auf Seite 27. Die Beschreibung zum HTML-Subagenten finden Sie im Handbuch „SNMP Management V5.0“.

- Der **Masteragent** ist einerseits der BS2000/OSD-Kommunikationspartner der Management-Plattform, der das SNMP-Protokoll abwickelt. Andererseits steuert er die Kommunikation mit den Subagenten. Zusätzlich bietet er Zugriffe auf die System- und die SNMP-Gruppe der MIB-II (RFC 1213) sowie auf Objekte weiterer standardisierter SNMP-MIBs (RFC 2572 - RFC 2575) und erlaubt so die Überwachung des Systems und der SNMP-relevanten Werte. Ferner ermöglicht der Masteragent den Web-Zugriff auf Informationen aus den MIBs.
- Der **Supervisor-Subagent** überwacht alle anderen Subagenten im System und die von ihnen gemeldeten Ereignisse.
- Der **Application Monitor-Subagent** überwacht Benutzeranwendungen, BCAM-Anwendungen, DCAM-Anwendungen, Tasks, Jobvariablen und BS2000/OSD-Subsysteme. Außerdem überwacht er Logging-Dateien in BS2000/OSD, POSIX und NFS. Logisch zusammengehörige Objekte aus einem Business-Prozess können mit dem Application Monitor-Subagent als Gruppe zusammengefasst werden und sowohl gemeinsam als auch einzeln überwacht werden.

- Der **Console Monitor-Subagent** dient der Konsolüberwachung. Er bietet einerseits die Möglichkeit, Konsolmeldungen als SNMP-Traps weiterzuleiten; die Menge der zu erfassenden Meldungen kann dabei gezielt definiert werden. Andererseits können Sie von der Management-Plattform aus BS2000/OSD-Konsol-Kommandos absetzen und das Resultat dieser Kommandos abfragen.

  Die zugehörige Management-Anwendung ermöglicht die netzweite Darstellung von Konsolmeldungen sowie die komfortable Konsolbedienung und Anwendungsüberwachung aller integrierten BS2000/OSD-Systeme.

- Der **Event-Subagent** bietet einen Mechanismus, periodisch SNMP-Abfragen (SNMP GetRequests) auf MIB-Objekte anderer Subagenten auszuführen und einfache Aktionen anzustoßen, wenn bestimmte Bedingungen erfüllt sind. Bedingungen können die Existenz von Objekten oder das Über- oder Unterschreiten von Schwellenwerten sein. Dabei kann sowohl der aktuelle Einzelwert, als auch die Differenz zur letzten Abfrage gewertet werden.

- Der **Scheduler-Subagent** bietet einen Mechanismus, Änderungen an SNMP-Objekten (SNMP SetRequests) periodisch oder zu bestimmten Zeitpunkten auszuführen. Für periodische Operationen werden die Sekunden zwischen SNMP-Set-Operationen angegeben. Zeitpunkte werden durch die Angabe von Monat, Tag, Wochentag, Stunde und Minute festgelegt. So ist es z.B. möglich, einen Request jeden Montag um 6:00 Uhr auszuführen oder jeden letzten Freitag im Monat um 22:00 Uhr.

- Der **HTML-Subagent** ermöglicht die Definition kundenspezifischer Web-Seiten (Custom-Pages) für den Web-Zugang zu den Management-Informationen von BS2000/OSD.

Die Liefereinheit SBA-BS2 enthält außerdem drei SDF-Kommandos zum Versenden von Traps (siehe Handbuch „SNMP Management V5.0“).

SSC-BS2 (SNMP-Standard-Collection BS2000)

Mit SSC-BS2 V6.0 wird eine Sammlung von Subagenten für BS2000/OSD-spezifische Management-Aufgaben ausgeliefert.

Im Folgenden werden die Agenten der SNMP-Standard-Collection BS2000 kurz vorgestellt. Ausführliche Informationen zu Funktionalität, Konfiguration und Bedienung der einzelnen Agenten finden Sie im Handbuch „SNMP Management V5.0“. Der HIPLEX-Subagent ist ausführlich beschrieben auf Seite 80 im Kapitel „Produktspezifische Agenten - Funktionale Erweiterungen zu SNMP V6.0“.

- Der **AVAS-Subagent** überwacht den Gesamtzustand von AVAS, die zentralen Prozesse und Ablaufsteuerungen sowie die Jobnetze und Strukturelemente.
- Der ***open*FT (BS2000)-Subagent** liefert Informationen über FT-Systemparameter und Statistikdaten des laufenden Betriebs. Weitere Funktionen sind das Starten und Stoppen des FT, die Steuerung der Diagnose, das Ändern des Public-Key zur Verschlüsselung und das Ändern des Status eines FT-Partners.
- Der **Subagent für HIPLEX** informiert über die aktuelle Konfiguration im HIPLEX-Verbund, über den Status der am Verbund beteiligten Systeme und Umschalteinheiten und meldet alle relevanten Änderungen. Bei Statusänderungen sendet der HIPLEX-Subagent Traps.
- Der **HSMS-Subagent** ermöglicht das Lesen und Ändern von globalen HSMS-Daten. Ferner liefert er detaillierte Informationen über HSMS-Aufträge. Der Umfang der Aufträge kann durch die Auswahlkriterien „Zustand“ und „Herkunftsort“ eingeschränkt werden.
- Der **Subagent für Spool & Print Service** überwacht die Geräte für SPOOL und RSO und liefert Informationen zu Druckaufträgen.
- Der **Subagent für das Storage-Management** liefert Informationen über Pubsets und Platten. Außerdem kann der Subagent ausgewählte oder alle Pubsets und Platten überwachen.
- Der **Subagent Host Resources** liefert Informationen über das System, über Geräte und Datei-Systeme sowie über die installierte Software und meldet Zustandswechsel.
- Der **OMNIS-Subagent** überwacht Datenstationen, Partner und Anwendungen und ermöglicht die Administration von OMNIS selbst.
- Der **Subagent für das Management von SESAM/SQL-Datenbanken** liefert Informationen über SESAM/SQL-Datenbanken und SESAM/SQL-DBHs, mit denen diese Datenbanken prozessiert werden (RDBMS-MIB gemäß RFC 1697).
- Der **Subagent zur Performance-Basisüberwachung mit SM2** (PerfMonB) liefert Durchschnittswerte zur Überwachung des CPU-Verbrauchs und der I/O-Raten.

SSA-SM2-BS2 (SNMP-Subagent für den Performance-Monitor SM2)

Der **SM2-basierte Performance-Subagent** SSA-SM2-BS2 liefert Basisinformationen zum SM2 selbst, d.h. zum Status des Subsystems, zur Version, zur Größe des Messintervalls und zum Stichprobenzyklus. Die eigentlichen Messwerte entsprechen den SM2-bekannten Reportgruppen und informieren über

– die CPU-Auslastung,
– I/O-Aktivitäten,
– die Auslastung des Hauptspeichers und des virtuellen Adressraums,
– die Belegung des Hauptspeichers durch die vier Standardkategorien von Tasks,
– Ein- und Ausgabeoperationen auf periphere Geräte während eines Messintervalls,
– applikationsspezifische Daten von *open*UTM-Anwendungen,
– Verbrauchswerte einzelner Tasks.

Zur Darstellung und Bewertung der gelieferten Messwerte auf der Management-Plattform steht die auf der mitgelieferten CD-ROM enthaltene Management-Anwendung PMBS2 zur Verfügung, die auch die gleichzeitige Überwachung mehrerer BS2000/OSD-Systeme ermöglicht.

Funktionalität, Konfiguration und Bedienung des SNMP-Subagenten für den Performance-Monitor SM2 sind ausführlich beschrieben im Handbuch „SNMP Management V5.0“.

SSA-OUTM-BS2 (SNMP-Subagent für *open*UTM in BS2000/OSD)

Der ebenfalls zu den additiven Subagenten gehörende ***open*UTM-Subagent** im Produkt SSA-OUTM-BS2 bietet folgende Leistungen:

– Überwachung und Steuerung ausgewählter *open*UTM-Anwendungen,
– Informationen über Systemparameter, physikalische und logische Terminals, Terminal-Pools, Transaktionscodes, Transaktionsklassen, Benutzerdaten, Verbindungen und Statistikdaten,
– Änderung von Anwendungseigenschaften und Systemparametern,
– Sperren bzw. Entsperren von UTM-Datenstationen,
– Beenden einer *open*UTM-Anwendung.

Für *open*UTM in Reliant UNIX wird der *open*UTM-Subagent im Produkt SSA-OUTM-SX bereitgestellt.

Funktionalität, Konfiguration und Bedienung des SNMP-Subagenten für *open*UTM in BS2000/OSD sind ausführlich beschrieben im Handbuch „SNMP Management V5.0“.

Erweiterung der Funktionalität des SNMP-Subagenten für *open*UTM in BS2000/OSD finden Sie ab Seite 87 im Kapitel „Produktspezifische Agenten - Funktionale Erweiterungen zu SNMP V6.0“ .

SNMP-Subagenten für *open*Net Server und *inter*Net Services

Hier steht ein **MIB-II-Subagent gemäß RFC 1213** zum Netzmanagement zur Verfügung.

Außerdem werden angeboten:

– **BCAM-Subagent** (liefert Informationen zu BCAM-spezifischen Einstellungen und Werten)

– **FTP-Subagent** (SNMP-Subagent für den FTP-Server)

Funktionalität, Konfiguration und Bedienung des SNMP-Subagenten für *open*Net Server und *inter*Net Services sind ausführlich beschrieben im Handbuch „SNMP-Management für *open*NetServer und *inter*Net Services“.

Folgende Produkte werden als Ergänzung angeboten:

– TransView SNMP-Proxy-Agent für BS2000/PDN (TV-SPBP)
– HNC mit integriertem SNMP-Agenten

Nähere Informationen finden Sie im Handbuch „SNMP Management V5.0“ sowie in den entsprechenden produktspezifischen Handbüchern.

2.4 Bedienoberflächen

Über das Standard-Protokoll SNMP können BS2000/OSD-Systeme grundsätzlich an jede Management-Plattform angeschlossen werden, die SNMP beherrscht. Dies ist für alle markt-relevanten Management-Plattformen der Fall. Die Management-Plattformen der verschiedenen Hersteller bringen dabei ein unterschiedliches Leistungsspektrum ein. Die von Fujitsu Siemens Computers empfohlene strategische Management-Plattform Unicenter der Firma Computer Associates (CA) ist universell ausgerichtet und verfügt über ein ausgeprägtes Alarm-Management mit vielfältigen Möglichkeiten, Reaktionen an Ereignisse zu koppeln.

Integrationspakete

Für CA Unicenter bietet Fujitsu Siemens Computers ein Integrationspaket (SMBS2 bzw. SMAWsmbs2) an, das die automatische Integration des BS2000/OSD in diese Management-Plattformen ermöglicht. Dieses Integrationspaket enthält u.a. Ergänzungen zur Oberfläche.

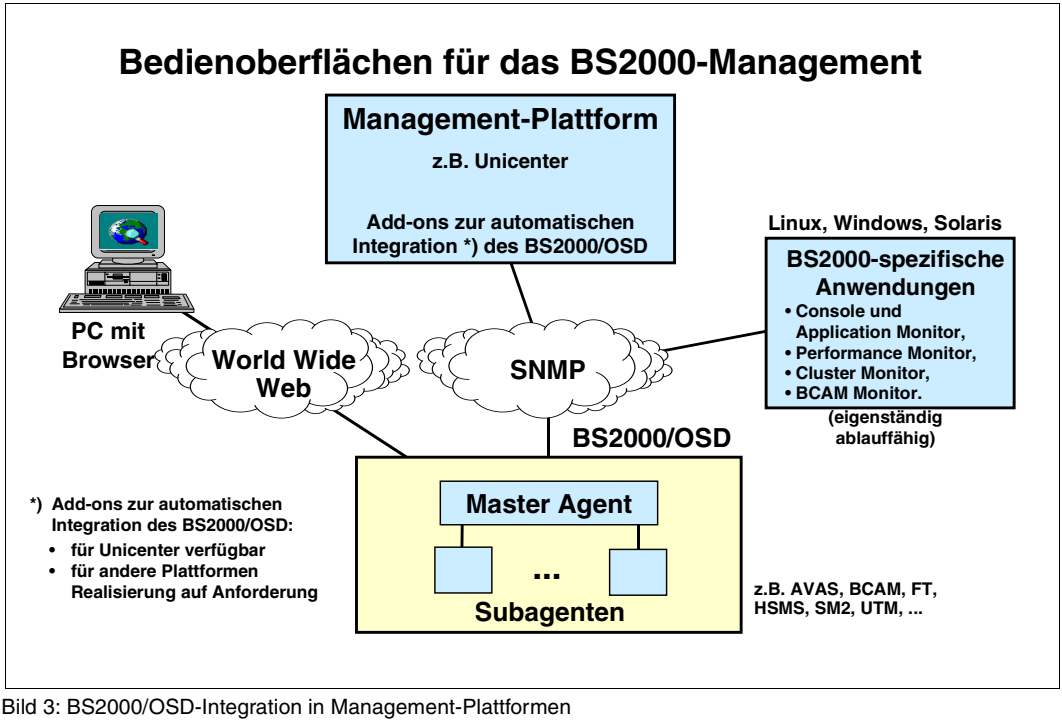

Bild 3: BS2000/OSD-Integration in Management-Plattformen

Übersicht über die Integrationspakete

Für die strategisch unterstützte Management-Plattform CA Unicenter werden folgende Integrationspakete angeboten:

Integrations-pakete	**zugehörige Agenten**	**Betriebssystem**
SMAWsmbs2	alle Subagenten	Solaris
SMBS2	alle Subagenten	Windows

Management-Anwendungen

Neben den Integrationspaketen bietet Fujitsu Siemens Computers in den Produkten BS2-SNMP-SO, BS2-SNMP-LX und BS2-SNMP-WIN für spezielle Subagenten eigene, auf die speziellen Eigenschaften und Aufgaben des jeweiligen Subagenten zugeschnittene Management-Anwendungen. Diese Management-Anwendungen ergänzen und verbessern die Darstellung und Handhabung der bestehenden Management-Plattform, können aber auch eigenständig auf einem Linux-, Solaris- oder Windows-System eingesetzt werden.

Übersicht über die Management-Anwendungen

Folgende spezielle Management-Anwendungen werden angeboten:

Management-Anwendungen	**Paket-Namen**	**zugehörige Agenten**	**Management-Plattform**	**Betriebssystem**
BCAM-Monitor BMBS2	BMBS2 bei Solaris: SMAWbmbs2	BCAM-Subagent, MIB-II-Subagent *open*Net Server	standalone / integriert	Solaris / Linux / Windows
Console und Application Monitor CMBS2	CMBS2 bei Solaris: SMAWcmbs2	Console Monitor-Subagent (SBA-BS2) Application Monitor-Subagent (SBA-BS2)	standalone / integriert	Solaris / Linux / Windows
Performance Monitor PMBS2	PMBS2 bei Solaris: SMAWpmbs2	Performance-Monitor-Subagent (SSA-SM2-BS2)	standalone / integriert	Solaris / Linux / Windows
Cluster Monitor CluMon	CluMon bei Solaris: SMAWclumn	HIPLEX-Subagent (SSC-BS2)	standalone / integriert	Solaris / Linux / Windows

Web-Zugriff auf Management-Informationen

Neben dem Zugriff über traditionelle SNMP Management-Anwendungen ermöglicht der Masteragent den Zugriff auf Management-Informationen via Web-Browser über das World Wide Web (WWW). Nähere Informationen zum Web-Zugriff finden Sie im Abschnitt „Web-Zugriff auf das BS2000/OSD-Management“ auf Seite 94 sowie im Handbuch „SNMP Management V5.0“.

3 Integration von BS2000/OSD in SNMP

Das BS2000/OSD-SNMP Management besteht aus folgenden Produkten für den Einsatz auf BS2000/OSD:

- SBA-BS2 V6.0
- SSC-BS2 V6.0
- SSA-SM2-BS2 V5.0
- SSA-OUTM-BS2 V5.0B

Außerdem umfasst das BS2000/OSD-SNMP Management die Pakete für die Management-Seite, die auf einer eigenen CD-ROM zusammen mit dem Produkt SBA-BS2 ausgeliefert werden bzw. im Internet zum Download verfügbar sind.

Die SNMP-Agenten sind hardware-unabhängig. Sie laufen auf allen Zentraleinheiten (inklusive der RISC-und SPARC-basierten Modelle), die von BS2000/OSD ab V2.0 bzw. OSD-SVP ab V2.0 unterstützt werden.

3.1 Software-Voraussetzungen

Software-Voraussetzungen für SBA-BS2

Der Einsatz des SNMP-Basic-Agent-BS2000 V6.0 setzt folgende Software voraus:

- BS2000/OSD-BC ≥ V2.0 bzw OSD-SVP ≥ V2.0
- POSIX-BC ≥ V1.0*
- SOCKETS(POSIX) ≥ 1.0*
- IMON ≥ V 2.0*
- SDF-P-BASYS V2.0B*
- JV ≥ V11.2 (optional)

Mit * gekennzeichnete Komponenten sind Bestandteil von BS2000/OSD-BC.

Software-Voraussetzungen für SSC-BS2

Der Einsatz der SNMP-Standard-Collection V6.0 setzt folgende Software voraus:

- BS2000/OSD-BC ≥ V2.0 bzw. OSD-SVP ≥ V2.0
- SBA-BS2 V6.0
- AVAS ≥ V3.0
- FT-BS2000 V6.2 bzw. *open*FT (BS2000) ≥ V6.0
- SPOOL ≥ V3.0*
- RSO ≥ V2.4
- HSMS ≥ V3.1
- OMNIS ≥ V8.1
- SDF-P-BASYS ≥ V2.0B*/**
- SESAM/SQL-Server ≥ V2.1B 0***
- SM2 ≥ V11.2
- JV ≥ V11.2
- HIPLEX-MSCF ≥ V1.0, HIPLEX-AF ≥ V3.0

Mit * gekennzeichnete Komponenten sind Bestandteil von BS2000/OSD-BC.
Mit ** gekennzeichnete Komponenten sind für den PrintService-Subagenten erforderlich.
Kennzeichnung mit *** bedeutet: Falls an einem Rechner mehrere DBHs überwacht werden sollen, ist außerdem der Einsatz von SESDCN erforderlich.

Software-Voraussetzungen für SSA-SM2-BS2

SSA-SM2-BS2 setzt SBA-BS2 V5.0 oder V6.0 und SM2 ab V11.2 in BS2000/OSD-BC ≥ V2.0 bzw. OSD-SVP ≥ V2.0 voraus.

Software-Voraussetzungen für SSA-OUTM-BS2

SSA-OUTM-BS2 setzt *open*UTM ≥ V3.3 und die entsprechende Version von UTM-D-SP voraus. Zusätzlich werden BS2000/OSD-BC ≥ V2.0 und SBA-BS2 V5.0 oder V6.0 vorausgesetzt.

Software-Voraussetzungen für die Subagenten für *open*Net Server und *inter*Net Services

Zum Einsatz des MIB-II-Subagenten ist SBA-BS2 ab V3.1 und DCAM ab V13.0 bzw. *open*Net Server V1.0 Voraussetzung. Ab DCAM V14.0 ist der BCAM-Subagent (Private MIB) ablauffähig.

Software-Voraussetzungen für die Integrationspakete SMBS2 und SMAWsmbs2

Die Software-Voraussetzungen für die Integrationspakete SMBS2 und SMAWsmbs2 sind beschrieben im Abschnitt „Voraussetzungen für die Integration“ auf Seite 110.

3.2 Installation der SNMP-Agenten in BS2000/OSD

Die Produkte SBA-BS2 und SSC-BS2 werden ebenso wie die additiven Subagenten SSA-SM2, SSA-OUTM-BS2 und die Subagenten für *open*Net Server und *inter*Net Services auf einem BS2000/OSD-Rechner installiert.

Die Installation von SBA-BS2, SSC-BS2, SSA-SM2-BS2 und SSA-OUTM-BS2 erfolgt mit dem Software-Liefer- und Informations-System SOLIS2. Die SOLIS2-Installation enthält, soweit erforderlich, BS2000/OSD-spezifische Arbeiten wie Subsystem-Katalog-Einträge etc.

Es ist darauf zu achten, dass im Subsystemkatalog ein Eintrag für das Subsystem SNMP erstellt wird.

Beachten Sie bitte, dass die interne Kommunikation zwischen Master- und Subagenten über die Portnummer 3161 abgewickelt wird. Insbesondere sollte die dynamische Portnummernvergabe von BCAM mit einem größeren Wert beginnen, der BCAM-Standardwert beträgt 4096.

Das Löschen der SINLIB nach der Installation führt zu Fehlern, da die Agenten die SINLIB auch für den Betrieb benötigen.

Die folgenden Abschnitte beschreiben die jeweiligen Installationsschritte für die Agentenseite. Die Installation des SNMP-Managers und der Management-Anwendungen ist beschrieben im Kapitel „SNMP Management“ auf Seite 93.

3.2.1 Installation von SBA-BS2 und SSC-BS2

Das Subsystem POSIX muss gestartet sein. Die ablauffähigen Agenten von SBA-BS2 befinden sich in der SINLIB.SBA-BS2.060. Diese enthält auch alle Elemente, die ins UFS installiert werden müssen. Die Installation erfolgt unter der Kennung SYSROOT bzw. TSOS (UID=0, GID=0) mit dem POSIX-Installationstool:

```
/CALL-PROCEDURE *LIB(LIB=$TSOS.SINPRC.POSIX-BC.<version>,ELEMENT=POSINST)
```

Funktion: POSIX-Programmpakete installieren
Produktname: SBA-BS2
Produktversion: 060

Die SINLIB.SSC-BS2.060 enthält die ablauffähigen Agenten und alle Elemente von SSC-BS2, die in das UFS installiert werden müssen. Die Installation erfolgt unter der Kennung SYSROOT bzw. TSOS (UID=0, GID=0) mit dem POSIX-Installationstool:

```
/CALL-PROCEDURE *LIB(LIB=$TSOS.SINPRC.POSIX-BC.<version>,ELEMENT=POSINST)
```

Funktion: POSIX-Programmpakete installieren
Produktname: SSC-BS2
Produktversion: 060

3.2.2 Installation von SSA-SM2-BS2

Die Installation erfolgt unter der Kennung SYSROOT bzw. TSOS (UID=0, GID=0) mit dem POSIX-Installationstool:

```
/CALL-PROCEDURE *LIB(LIB=$TSOS.SINPRC.POSIX-BC.<version>,ELEMENT=POSINST)
```

Funktion: POSIX-Programmpakete installieren
Produktname: SSA-SM2-BS2
Produktversion: 050

3.2.3 Installation von SSA-OUTM-BS2

Die Installation erfolgt unter der Kennung SYSROOT bzw. TSOS (UID=0, GID=0) mit dem POSIX-Installationstool:

```
/CALL-PROCEDURE *LIB(LIB=$TSOS.SINPRC.POSIX-BC.<version>,ELEMENT=POSINST)
```

Funktion: POSIX-Programmpakete installieren
Produktname: SSA-OUTM-BS2
Produktversion: 050

3.2.4 Versionswechsel

Die folgenden Hinweise zum Versionswechsel ergänzen die in den vorangegangenen Installationsabschnitten enthaltenen Informationen.

Umstieg von einer älteren SBA-BS2-Version auf V6.0

Auch beim Versionswechsel erfolgt die Installation über IMON oder durch Bekanntgabe der SYSSII-Datei (IMON V2.0). Bibliotheken können problemlos in die gewünschte Kennung eingespielt werden, da wegen unterschiedlicher Versionsbezeichnung Konflikte mit der Vorgängerversion ausgeschlossen sind.

Die Syntaxdatei der Vorversion muss durch die Syntaxdatei der Version 6.0 ersetzt werden. Hierzu sollten die Agenten beendet werden. Denn die Agenten der Vorgängerversion können nicht mehr mittels STOP-Kommando beendet werden, da die Version des Agenten mit der des zugehörigen Kommandoprogramms übereinstimmen muss.

Die Datei *snmpd.cnf* in */etc/snmp/agt* muss um die kundenspezifischen Einträge erweitert werden. Dazu sollte der Masteragent gestoppt werden, da dieser u.U. die Konfigurationsdatei überschreibt.

3.2.5 Deinstallation

Die Deinstallation erfolgt ebenfalls unter der Kennung SYSROOT bzw. TSOS (UID=0, GID=0) mit dem POSIX-Installationstool:

```
/CALL-PROCEDURE *LIB(LIB=$TSOS.SINPRC.POSIX-BC.<version>,ELEMENT=POSINST)
```

Funktion: POSIX-Programmpakete deinstallieren
Produktname: siehe entsprechender Name in den Abschnitten „Installation von ...“ auf Seite 25.
Produktversion: <prod-version>

<prod-version> bezeichnet die Versionsnummer des zu deinstallierenden Programmpakets.

4 SNMP-Basisagenten für BS2000/OSD

Voraussetzung für das SNMP Management im BS2000/OSD ist das Basisprodukt SNMP-Basic-Agent BS2000, das die SNMP-Basisagenten enthält.

Es gibt folgende SNMP-Basisagenten:

- Masteragent
- Supervisor-Subagent
- Application Monitor-Subagent
- Console Monitor-Subagent
- Event-Subagent
- Scheduler-Subagent

Der SNMP-Basic-Agent mit seinen SNMP-Subagenten im BS2000/OSD kann über das SNMP-Protokoll prinzipiell an alle Management-Plattformen angeschlossen werden. Basis hierfür sind die standardisierten und die BS2000/OSD-spezifischen MIBs im ASN.1-Format, die im Handbuch „SNMP Management V5.0“, Kapitel „Funktionen des BASIC-AGENT“, ausführlich beschrieben sind.

Ergänzend gibt es spezielle Produkte, die die Integration erleichtern (siehe Kapitel „SNMP Management“ auf Seite 93).

4.1 Masteragent

Der Masteragent bildet die Schnittstelle des SNMP-Agenten zum Netz und damit zu den Management-Plattformen.

4.1.1 Funktionalität des Masteragenten

Der Masteragent erfüllt folgende Funktionen:

- Abwickeln des SNMP-Protokolls und Kommunikation über ein TCP/IP-Netz mit dem SNMP-Manager auf der Management-Plattform
- Überprüfen der Zugangsberechtigungen
- Weiterleiten der Requests des SNMP-Managers an die zuständigen Subagenten
- Weiterleiten der Antworten und Traps der Subagenten an den SNMP-Manager

Ferner ermöglicht der Masteragent den Zugriff auf Management-Informationen über das World Wide Web (WWW). Somit lassen sich die von den Subagenten bereitgestellten Informationen sowohl über traditionelle SNMP Management-Anwendungen als auch via Web-Browser abfragen und ändern.

An den Masteragenten können auch benutzereigene Subagenten angeschlossen werden. In seiner Funktion als zentrale Management-Instanz und SNMP-Protokollmaschine realisiert der Masteragent außerdem die Objekte der System-Gruppe, der SNMP-Gruppe der MIB-II und des SNMP-Frameworks.

Zur System-Gruppe und der SNMP-Gruppe der MIB-II gehören u.a. folgende Informationen:

– Laufzeit des Agenten
– Name und Typ des Systems
– Anzahl der ein- und ausgehenden Pakete
– Anzahl der verschiedenen Protokollfehler
– Anzahl der Sicherheitsverstöße (z.B. Versuche, einen Agenten unter Angabe eines falschen Community-Namens abzufragen)

4.1.2 Konfiguration des Masteragenten

Konfigurationsdatei des Masteragenten ist die Datei *snmpd.cnf*. Sie befindet sich im POSIX-Dateisystem im Verzeichnis */etc/snmp/agt*. Neben den Parametern für die Security-Konfiguration (siehe Handbuch „SNMP Management V5.0“) enthält *snmpd.cnf* u.a. die Initial System Group und optional die Startanweisung für den Supervisor-Subagenten.

Die Datei *snmpd.cnf* sollte nur bei gestopptem Masteragenten editiert werden, da der Masteragent die Konfigurationsdatei u.U. zwischenzeitlich überschreibt.

Initial System Group

sysDescr	
sysLocation	Fujitsu Siemens Computers Mch-P *
sysContact	Help Desk *
sysObjectID	1.3.6.1.4.1.231.1.6
MAX_PDU_TIME	Wartezeit des Masteragenten auf eine Antwort des Subagenten, bevor er den Request verwirft
MAX_THREADS	gibt die maximale Anzahl der Threads an, die gleichzeitig bearbeitet werden können. Es wird empfohlen, die Anzahl etwa doppelt so groß wie die Anzahl der Subagenten zu wählen, da die Subagenten nur jeweils einen Request bearbeiten können.
MAX_OUTPUT_WAITING	gibt die maximale Anzahl Bytes an, die als Nachrichten vom Master gespeichert werden können, bevor ein „Overflow“ auftritt.
MAX_SUBAGENTS	definiert die maximale Anzahl der Subagenten, die sich an den Masteragenten anschließen dürfen.
RETRY_INTERVAL	RETRY_INTERVAL wird derzeit nicht genutzt.
snmpEnableAuthenTraps	2 : es werden keine Authentisierungsfehlertraps geschickt 1 : es werden Authentisierungsfehlertraps geschickt
subagent	Wenn der Supervisor-Subagent gestartet werden soll, muss hier der Name der Bibliothek angegeben werden: [:<catid>:] [$<userid>.]SYSLNK.SBA-BS2.060 oder für RISC-Maschinen: [:<catid>:] [$<userid>.]SRMLNK.SBA-BS2.060 oder für SPARC: [:<catid>:] [$<userid>.]SPMLNK.SBA-BS2.060

Voreinstellung der Initial System Group

* Passen Sie bitte nur die Werte sysLocation und sysContact Ihren Gegebenheiten an, der Wert sysObjectID sollte unverändert bleiben.

4.1.3 Start / Stopp des Masteragenten

Vor dem ersten Start des Masteragenten muss im BS2000/OSD die Datei */etc/snmp/agt/snmpd.cnf* an die eigene Konfiguration angepasst werden (siehe Seite 56).

Der Start des Masteragenten, wie auch der Start aller Subagenten, sollte im Hintergrund erfolgen, da sonst die Shell blockiert wird.

Starten des Masteragenten im BS2000/OSD:

/START-SNMP-MASTER
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54 without-gen-vers> , **CPU-LIMIT**=***STD** / <integer 1 .. 32767> / ***NO** , **JOB-CLASS**=***STD** / <name 1 .. 8> , **TIMER-INTERVAL**= 5 / <integer 1 .. 32767>

oder in der POSIX-Shell mit:

```
snmpdm
```

Der Start des Masteragenten, wie auch aller anderen Agenten, sollte im Hintergrund erfolgen, da sonst die Shell blockiert wird.

Stoppen des Masteragenten im BS2000/OSD:

/STOP-SNMP-MASTER
VERSION=***STD** / <product-version>

oder in der POSIX-Shell mit:

```
snmpdmcmd T
```

Beschreibung der Operanden:

VERSION=*STD / <product-version>
definiert die zu startende bzw. stoppende Version des Agenten. Diese Angabe wird derzeit nicht ausgewertet.

MONJV=*NONE / <filename 1..54 without-gen-vers>
Name der Jobvariable, die den Agenten überwachen soll. Die Angabe *NONE - keine Überwachung durch eine Jobvariable - ist Standard.

CPU-LIMIT=*STD / <integer 1 .. 32767> / *NO
Angabe der maximalen CPU-Laufzeit in Sekunden. Bei Angabe von *STD wird der generierte Standardwert verwendet.

JOB-CLASS=*STD / <name 1..8>
Jobklasse, mit der der Agent gestartet wird. Bei Angabe von *STD wird die generierte Standard-Jobklasse verwendet.

TIMER-INTERVAL=5 / <integer 1 .. 32767>
Zeitintervall, in dem der Agent überprüft, ob Anforderungen vom Kommandoprogramm vorliegen. Das Zeitintervall ist standardmäßig auf fünf Sekunden eingestellt.
Das Intervall wird vom Supervisor-Subagenten zur Überprüfung seiner Subagenten-Tabelle genutzt. Wurde vom Subagenten während der letzten fünf Minuten keine Nachricht empfangen, dann überprüft der Supervisor diesen Subagenten durch eine Anfrage.

4.2 Supervisor-Subagent

Aufgabe des Supervisor-Subagenten ist die Überwachung aller an den Masteragenten angeschlossenen Subagenten. Der Supervisor-Subagent ist besonders eng an den Masteragenten gekoppelt und registriert alle Ereignisse, die der Masteragent von den anderen Subagenten empfängt.

4.2.1 Funktionalität des Supervisor-Subagenten

Der Supervisor-Subagent kontrolliert in regelmäßigen Abständen, ob die anderen Subagenten erreichbar sind, und sendet in folgenden Fällen einen Trap an den SNMP-Manager:

– Ein Subagent meldet sich beim Masteragenten an oder ab.
– Ein Subagent ist nicht mehr erreichbar.

Damit genügt es, nur eine Instanz pro BS2000/OSD-System zu pollen. Wenn der Masteragent mit dem Supervisor-Subagenten aktiv ist, liefert der Supervisor-Subagent den Status aller anderen Subagenten. Dies reduziert erheblich die Netzbelastung, die von der aktiven Überwachung durch den SNMP-Manager verursacht wird.

Zu jedem angemeldeten Subagenten liefert der Supervisor-Subagent außerdem u.a. folgende Werte:

– Status des Subagenten (*active*, *disconnected*, *undefined*)
– Zeitpunkt der Anmeldung
– Zeitpunkt der letzten Kommunikation
– Anzahl der beantworteten Requests
– Anzahl der gesendeten Traps
– eine unterstützte OID (Object Identifier)

4.2.2 Konfiguration des Supervisor-Subagenten

Wenn sich ein Subagent beim Masteragenten anmeldet, informiert er den Masteragenten über alle von ihm unterstützten Object Identifiers (OID). Daraufhin identifiziert der Supervisor-Subagent den neu angemeldeten Subagenten über eine dieser OIDs.

In der Datei */etc/snmp/agt/supervis.cnf* kann diese OID sowie ein Name für den Subagenten festgelegt werden.

Der Eintrag ist wie folgt aufgebaut:

<name> <oid>

<name>
Name des Subagenten, der vom Supervisor-Subagenten überwacht werden soll.

<oid>
OID des Subagenten, der vom Supervisor-Subagenten überwacht werden soll.

Falls die Datei */etc/snmp/agt/supervis.cnf* nicht existiert, wird die kleinste unterstützte OID verwendet.

Beispiel

```
AppMon      1.3.6.1.4.1.231.2.23.5.3.0
```

4.2.3 Start / Stopp des Supervisor-Subagenten

Der Supervisor-Subagent wird stets zusammen mit dem Masteragenten gestartet oder beendet. Deshalb gibt es für den Supervisor-Subagenten kein eigenes Start- bzw. Stopp-Kommando. Der Start des Supervisor-Subagenten wird initiiert durch einen entsprechenden Eintrag in der Datei */etc/snmp/agt/snmpd.cnf*. Solange dieser Eintrag existiert, wird der Supervisor-Subagent immer automatisch mit dem Masteragenten gestartet und beendet.

4.3 Application Monitor-Subagent

Der Application Monitor-Subagent ist ein universeller Agent und keiner speziellen BS2000/OSD-Komponente zugeordnet.

4.3.1 Funktionalität des Application Monitor-Subagenten

Mit dem Application Monitor-Subagenten lassen sich folgende Einheiten (Objekte) überwachen, d.h. deren Zustände und Eigenschaften abfragen:

- Benutzeranwendungen
- BCAM-Anwendungen
- DCAM-Anwendungen
- Subsysteme
- Jobvariablen
- Protokolldateien

Jede Veränderung kann der Application Monitor-Subagent auch unaufgefordert als Trap an eine Management-Plattform schicken. Auf dieser Basis lassen sich Benutzeranwendungen und Tasks kontrollieren. Einträge in eine spezifische Datei können ebenfalls als Trap verschickt werden. Außerdem können Sie Gruppen zusammengehöriger Anwendungen als Einheit (Objekt) verwalten. Art und Umfang der Anwendungsüberwachung steuern Sie individuell über eine Konfigurationsdatei. Den Namen der Konfigurationsdatei teilen Sie dem Application Monitor-Subagenten im Start-Kommando mit.

Beide Möglichkeiten, Trap und Request, gestatten die Einbindung einer universellen Anwendungsüberwachung in das Alarmmanagement einer Management-Plattform.

Für den Application Monitor-Subagenten gibt es eine eigene Management-Anwendung, den Console und Application Monitor (siehe Seite 101).

4.3.2 Konfiguration des Application Monitor-Subagenten

4.3.2.1 Anweisungen für die Konfigurationsdatei

Die Konfigurationsdatei enthält Informationen darüber, welche Anwendungen, Tasks, Subsysteme, Jobvariablen und Protokolldateien überwacht werden sollen. Es können jeweils bis zu 256 Benutzer-, BCAM-Anwendungen, Jobvariablen und Protokolldateien sowie 128 DCAM-Anwendungen überwacht werden. Benutzer- und BCAM-Anwendungen sowie Tasks, die überwacht werden sollen, müssen mit Jobvariablen angestartet werden. Die Anzahl der zu überwachenden Subsysteme ist unbegrenzt.

Die Einträge in der Konfigurationsdatei werden über SDF-Anweisungen erzeugt. Mit der Anweisung //REMARK können Kommentare in der Konfigurationsdatei hinterlegt werden. Die letzte Anweisung der Datei muss immer die Anweisung //END sein. Anweisungen, die hinter der END-Anweisung stehen, werden ignoriert.

Überwachung	**Anweisung**	**Seite**
Anwendung	//ADD-APPLICATION-RECORD	39
DCAM-Anwendung	//ADD-DCAM-APPLICATION-RECORD	41
Subsystem	//ADD-SUBSYSTEM-RECORD	43
Protokolldatei	//ADD-LOG-FILE-RECORD	45
Jobvariable	//ADD-JV-RECORD	47
Gruppe von zusammengehörigen Anwendungen	//DEFINE-OBJECT	49
Trap-Format	//DEFINE-TRAP-FORMAT	51
Überwachungsprotokoll	//SET-TIMER-OPTIONS	52

Beispiel 1

Das folgende Beispiel finden Sie auch in der Bibliothek SINLIB.SBA-BS2.060

```
//REMARK Application Monitor, SDF-Configuration File
//REMARK
//REMARK Trap Format
//DEFINE-TRAP-FORMAT TYPE = (*GENERIC, *TVCC)
//REMARK
//REMARK Application Monitoring, Type BCAM
//ADD-APPLICATION-RECORD  -
//        APPLICATION-NAME = ANW1  -
//       ,VERSION = V1.0  -
//       ,TYPE = *BCAM  -
//       ,JV-NAME = MONJV  -
//       ,TRAP-CONDITION = (A, R)  -
//       ,WEIGHT=10  -
//
//REMARK Application Monitoring, Type USER
//ADD-APPLICATION-RECORD  -
//        APPLICATION-NAME = Applikation1  -
//       ,VERSION = V01.0A00  -
//       ,TYPE = *USER  -
//       ,JV-NAME = MJV1  -
//       ,TRAP-CONDITION = A  -
//       ,WEIGHT=5   -
//       ,ACKNOWLEDGE = *YES -
//
//ADD-APPLICATION-RECORD  -
//        APPLICATION-NAME = Applikation2  -
//       ,TYPE = *USER  -
//       ,JV-NAME = MJV2  -
//       ,TRAP-CONDITION = (T, A)  -
//
//REMARK Subsystem Monitoring
//ADD-SUBSYSTEM-RECORD  -
//        NAME = EDT  -
//
//ADD-SUBSYSTEM-RECORD  -
//        NAME = MAREN  -
//       ,VERSION = 08.1  -
//       ,TRAP-CONDITION = *NONE  -
//
//REMARK File Monitoring
//ADD-LOG-FILE-RECORD  -
//        NAME = /tmp/logfile1  -
//       ,APPLICATION-NAME = Datei1  -
```

```
//       ,MONITORING = *NO  -
//       ,FORMAT = *EBCDIC  -
//
//ADD-LOG-FILE-RECORD  -
//        NAME = $HUGO.LOGFILE2  -
//       ,MONITORING = *NO  -
//       ,FORMAT = *EBCDIC  -
//       ,PATTERN = '*important*' -
//
//REMARK Jobvariables
//ADD-JV-RECORD  -
//        JV-NAME = JOBVAR  -
//       ,PATTERN = ('*terminated*', '[1-5]00*')  -
//
//REMARK DCAM Application
//ADD-DCAM-APP A-NAME=d3str10$,HOST=camilla2,KEEP-CONNECTION=*NO -
//
//ADD-DCAM-APP A-NAME=$CONSOLE,HOST=D017ZE00, -
//        MSG='@CONSOLE,TSOS,''@@@@@@@@'',V01' -
//       ,WEIGHT=99  -
//
//REMARK Object
//DEFINE-OBJECT OBJECT-NAME=OB1,BCAM-APPLICATION=ANW1,          -
//        LOG-FILE=(/tmp/logfile1),  -
//        MONITORING-TIME=*INTERVAL(START=3:00,STOP=18:11,EX=SUN)
//END
```

Beispiel 2: Monitoring kritischer Anwendungen

Sie wollen eine ausgewählte, für Sie besonders wichtige Anwendung überwachen, um jederzeit über mögliche Ausfälle informiert zu sein. Die Anwendung wurde mit der Monitor-Jobvariablen ANWMONJV gestartet.

Ihre Anwendung tragen Sie wie folgt in die Konfigurationsdatei des Application Monitor-Subagenten ein:

```
//ADD-APPLICATION-RECORD -
//   ,APPLICATION-NAMW=ANW -
//   ,TYPE=*USER -
//   ,JV-NAME=ANWMONJV -
//   ,TRAP-CONDITION=(A,R,T)
```

Der Application Monitor-Subagent registriert dadurch jede Änderung der Monitor-Jobvariablen ANWMONJV. Durch den Start des Programms wird die Monitor-Jobvariable auf $R gesetzt. Die Statusänderung der Monitor-Jobvariablen leitet der Application Monitor als Trap an die Management-Plattform weiter.

Beispiel 3: Überwachung eines MAREN-Systems

Ein MAREN-System besteht u.a. aus folgenden Komponenten:

– Subsystem MAREN
– Steuerprogramm MARENCP
– automatische Freibandzuweisung MARENUCP

Darüber hinaus wird jede von der automatischen Freibandzuweisung reservierte VSN in der Jobvariablen TAPE.FILE.MAREN hinterlegt.

Folgende Definition eines Objekts „MAREN“ fasst diese Komponenten zusammen:

```
//DEFINE-OBJECT OBJECT-NAME = MAREN -
//   ,USER-APPLICATION = (MARENCP, MARENUCP) -
//   ,SUBSYSTEM = MAREN -
//   ,JV = TAPE.FILE.YES
```

4.3.2.2 Wechsel der Konfigurationsdatei im laufenden Betrieb

Änderungen der aktuellen Konfigurationsdatei im laufenden Betrieb können dem Application Monitor-Subagenten entweder durch Setzen des Objekts *appMonConfFile* oder per Kommando vorgenommen werden.

```
/START-APPMONCMD

       x "readConfig <filename>"
```

POSIX:

```
appmoncmd x "readConfig <filename>"
```

Bei Syntaxfehlern in *appMonConfFile* wird mit der ursprünglichen Konfiguration weitergearbeitet.

ADD-APPLICATION-RECORD

Die Anweisung //ADD-APPLICATION-RECORD benennt die BCAM- und Benutzeranwendungen, die überwacht werden sollen. Unter Anwendungen sind Programme oder Tasks zu verstehen.

//ADD-APPLICATION-RECORD
APPLICATION-NAME = <composed-name_1 .. 54_with-underscore>
, **VERSION** = ***NONE** / <product-version>
, **TYPE** = ***BCAM** / ***USER**
, **JV-NAME** = <filename_1 .. 54>
, **TRAP-CONDITION** = **A** / list-poss (6) : <name_1 .. 1>
, **WEIGHT**= **0 /** <integer 0 .. 999>
, **ACKNOWLEDGE** = ***NO** / ***YES**

APPLICATION-NAME=<composed-name_1..54_with-underscore>
bestimmt die Anwendung, die der Subagent überwachen soll.

VERSION=*NONE / <product-version>
Versionsnummer der Anwendung.
Standardwert: *NONE

TYPE=*BCAM / *USER
Typ der Anwendung.

JV-NAME = <filename_1 .. 54>
Jobvariable (MONJV), mit der die Anwendung bzw. die Task überwacht wird.

TRAP-CONDITION=A / list-poss (6) : <name_1 .. 1>
Zustände, bei denen ein Trap erzeugt werden soll.

WEIGHT= 0 / <integer 0 .. 999>
Gewichtung der für den Application Monitor-Subagenten spezifischen Traps. Beim Senden eines Traps versorgt der Application Monitor-Subagent beim generischen Trap das Trap-Objekt *appMonWeight* und die Trap-Nummer mit dem angegebenen Wert (siehe Handbuch „SNMP Management V5.0“). Sollen in einer Anwendung für verschiedene Ereignisse verschiedene Gewichte verwendet werden, dann muss die zugehörige //ADD-APPLICATION-RECORD-Anweisung mehrmals in der Konfigurationsdatei angegeben werden.
Standardwert: 0

ACKNOWLEDGE=*NO / *YES
Angabe, ob der Trap bestätigt werden muss. Es können nur Application Monitor-spezifische Traps bestätigt werden.
Standardwert: *NO

ADD-DCAM-APPLICATION-RECORD

Die Anweisung //ADD-DCAM-APPLICATION-RECORD benennt die DCAM-Anwendungen, die zyklisch überwacht werden sollen. Das Überwachungsintervall für DCAM-Anwendungen liegt im Standardfall beim 60-fachen Wert der Timer-Einstellung, beträgt also standardmäßig 5 Minuten. Mit der Anweisung //SET-TIMER-OPTIONS (siehe Seite 52) können Sie als Überwachungsintervall ein beliebiges Vielfaches der Timer-Einstellung festlegen.

Maximal können 128 DCAM-Anwendungen überwacht werden.

//ADD-DCAM-APPLICATION-RECORD
APPLICATION-NAME = <name_1 .. 8>
, **HOST**= ***OWN** / <name1 .. 8>>
, **KEEP-CONNECTION** = ***YES** / ***NO**
, **MSG**= ***NONE** / <c-string> / <x-string>
, **TRAP-CONDITION** = list-poss (2) : ***NOT-AVAILABLE** / ***AVAILABLE**
, **WEIGHT**= **0 / <integer 0 .. 999>**
, **ACKNOWLEDGE**= ***NO** / ***YES**

APPLICATION-NAME=<name_1..8>
bestimmt die DCAM-Anwendung, die der Subagent überwachen soll.

HOST=*OWN / <name1..8>
Rechner, auf dem die DCAM- Anwendung läuft
Standardwert: *OWN

KEEP-CONNECTION=*YES / *NO
Angabe, ob die Verbindung wieder abgebaut werden soll
Standardwert: *YES

MSG= *NONE / <c-string> / <x-string>
Verbindungsnachricht
Standardwert: *NONE

TRAP-CONDITION=*NOT-AVAILABLE / *AVAILABLE
Zustände, bei denen ein Trap erzeugt wird.
Standardwert: *NOT-AVAILABLE

WEIGHT= 0 / <integer 0 .. 999>
Gewichtung der für den Application Monitor-Subagenten spezifischen Traps. Beim Senden eines Traps versorgt der Application Monitor-Subagent beim generischen Trap das Trap-Objekt *appMonWeight* und die Trap-Nummer mit dem angegebenen Wert (siehe Handbuch „SNMP Management V5.0“).
Standardwert: 0

ACKNOWLEDGE= *NO / *YES
Angabe, ob der Trap bestätigt werden muss. Es können nur Application Monitor-spezifische Traps bestätigt werden.
Standardwert: *NO

ADD-SUBSYSTEM-RECORD

Die Anweisung //ADD-SUBSYSTEM-RECORD definiert die zu überwachenden Subsysteme. Das Überwachungsintervall liegt im Standardfall bei dem fünffachen Wert der Timer-Einstellung, beträgt standardmäßig also 25 Sekunden. Mit der Anweisung //SET-TIMER-OPTIONS (siehe Seite 52) können Sie als Überwachungsintervall ein beliebiges Vielfaches der Timer-Einstellung festlegen.

//ADD-SUBSYSTEM-RECORD
NAME = <structured-name 1 .. 8> / ***ALL**
, **VERSION** = ***NONE** / <product-version>
, **TRAP-CONDITION** = ***NONE** / list-poss (8) : ***CREATED** / ***NOT-CREATED** / ***IN-DELETE** / ***IN-CREATE** / ***IN-RESUME** / ***IN-HOLD** / ***NOT-RESUMED** / ***LOCKED**
, **WEIGHT= 0 / <integer 0 .. 999>**
, **ACKNOWLEDGE** = ***NO** / ***YES**

NAME=<structured-name 1..8> / *ALL
bestimmt das Subsystem, das der Subagent überwachen soll.

VERSION=*NONE / <product-version>
Versionsnummer des Subsystems
Standardwert: *NONE

TRAP-CONDITION=*NONE / list-poss (8) : *CREATED / *NOT-CREATED / *IN-DELETE / *IN-CREATE / *IN-RESUME / *IN-HOLD / *NOT-RESUMED / *LOCKED
Zustände, bei denen ein Trap erzeugt werden soll.
Standardwert: *NONE

ACHTUNG!
Bei der Angabe NAME=*ALL sollten Sie TRAP-CONDITION=*NONE verwenden, da andernfalls Performance-Probleme auftreten können.

WEIGHT= 0 / <integer 0 .. 999>
Gewichtung der für den Application Monitor-Subagenten spezifischen Traps. Beim Senden eines Traps versorgt der Application Monitor-Subagent beim generischen Trap das Trap-Objekt *appMonWeight* und die Trap-Nummer mit dem angegebenen Wert (siehe Handbuch „SNMP Management V5.0“). Sollen in einem Subsystem für verschiedene Ereignisse verschiedene Gewichte verwendet werden, dann muss die zugehörige //ADD-SUBSYSTEM-RECORD-Anweisung mehrmals in der Konfigurationsdatei angegeben werden.
Standardwert: 0

ACKNOWLEDGE=*NO / *YES
Angabe, ob der Trap bestätigt werden muss. Es können nur Application Monitor-spezifische Traps bestätigt werden.
Standardwert: *NO

ADD-LOG-FILE-RECORD

Die Anweisung //ADD-LOG-FILE-RECORD definiert die zu überwachenden Protokolldateien. Standardmäßig sendet der Application Monitor-Subagent bei jeder Änderung der Datei (Log-File) einen Trap. Es ist möglich, die Traps bzw. Einträge zu filtern. Mit der Anweisung //SET-TIMER-OPTIONS (siehe Seite 52) können Sie als Überwachungsintervall ein beliebiges Vielfaches der Timer-Einstellung festlegen.

//ADD-LOG-FILE-RECORD
NAME = <filename_1 .. 54> / <posix-pathname>
, APPLICATION-NAME = ***NONE** / <composed-name_1 .. 54_with-underscore>
, **MONITORING** = ***YES** / ***NO**
, **FORMAT** = ***EBCDIC** / ***ASCII**
, **PATTERN** = ***NONE** / list-poss (8) : <c-string_1 .. 256_with-lower-case>
, **WEIGHT**= **0 / <integer 0 .. 999>**
, **ACKNOWLEDGE** = ***NO** / ***YES**

NAME=<filename_1 .. 54> / <posix-pathname>
bestimmt die Protokolldatei, die der Subagent überwachen soll.

APPLICATION-NAME=*NONE / <composed-name_1 .. 54_with-underscore>
Name der Anwendung.
Standardwert: *NONE

MONITORING=*YES / *NO
Angabe, ob die Protokolldatei überwacht werden soll.

FORMAT=*EBCDIC / *ASCII
Format der Protokolldatei.
Standardwert: *EBCDIC

PATTERN = *NONE / list-poss (8) : <c-string_1 .. 256_with-lower-case>
Angabe eines oder mehrerer Suchmuster. Fehlt die Angabe PATTERN, so werden alle Einträge in eine Protokolldatei per Trap gemeldet.
Es sind folgende Wildcard-Angaben zulässig:

? : ersetzt ein beliebiges Zeichen
* : ersetzt eine beliebig lange Folge von Zeichen
[s] : ersetzt genau ein Zeichen aus der Zeichenkette s
[c1 - c2]: ersetzt ein beliebiges Zeichen aus dem Bereich von c1 bis c2
Das Zeichen "\" (Backslash) muss zur Entwertung der Sonderzeichen angegeben werden.
Es wird zwischen Groß- und Kleinschreibung unterschieden.
Standardwert: *NONE

WEIGHT= 0 / <integer 0 .. 999>
Gewichtung der für den Application Monitor-Subagenten spezifischen Traps. Beim Senden eines Traps versorgt der Application Monitor-Subagent beim generischen Trap das Trap-Objekt *appMonWeight* und die Trap-Nummer mit dem angegebenen Wert (siehe Handbuch „SNMP Management V5.0“).
Standardwert: 0

ACKNOWLEDGE=*NO / *YES
Angabe, ob der Trap bestätigt werden muss. Es können nur Application Monitor-spezifische Traps bestätigt werden.
Standardwert: *NO

ADD-JV-RECORD

Die Anweisung //ADD-JV-RECORD definiert die zu überwachenden Jobvariablen. Standardmäßig sendet der Application Monitor-Subagent jede Änderung einer Jobvariablen als Trap. Es ist jedoch möglich, die Traps zu filtern.

//ADD-JV-RECORD
JV-NAME = <filename_1 .. 54>
, **APPLICATION-NAME** = ***NONE** / <composed-name_1 .. 54_with-underscore>
, **PASSWORD** = ***NONE** / <c-string_1 .. 4 > / < x-string_1 .. 8>
, **PATTERN** = ***NONE** / list-poss (8) : <c-string_1 .. 256_with-lower-case>
, **WEIGHT**= **0 / <integer 0 .. 999>**
, **ACKNOWLEDGE** = ***NO** / ***YES**

JV-NAME = <filename_1 .. 54>
bestimmt die Jobvariable, die der Subagent überwachen soll.

APPLICATION-NAME = *NONE / <composed-name_1 .. 54_with-underscore>
Name der Anwendung.
Standardwert: *NONE

PASSWORD = *NONE / <c-string_1 .. 4 > / <x-string_1 .. 8>
Lesepasswort der Jobvariablen.
Standardwert: *NONE

PATTERN = *NONE / list-poss (8) : <c-string_1 .. 256_with-lower-case>
Angabe eines oder mehrerer Suchmuster. Fehlt die Angabe PATTERN, so werden alle JV-Änderungen per Trap gemeldet.
Es sind folgende Wildcard-Angaben zulässig:

? : ersetzt ein beliebiges Zeichen
* : ersetzt eine beliebig lange Folge von Zeichen
[s] : ersetzt genau ein Zeichen aus der Zeichenkette s
[c1 - c2]: ersetzt ein beliebiges Zeichen aus dem Bereich von c1 bis c2
Das Zeichen "\" (Backslash) muss zur Entwertung der Sonderzeichen angegeben werden.
Es wird zwischen Groß- und Kleinschreibung unterschieden.
Standardwert: *NONE

WEIGHT= 0 / <integer 0 .. 999>
Gewichtung der für den Application Monitor-Subagenten spezifischen Traps. Beim Senden eines Traps versorgt der Application Monitor-Subagent beim generischen Trap das Trap-Objekt *appMonWeight* und die Trap-Nummer mit dem angegebenen Wert (siehe „Handbuch „SNMP Management V5.0“).
Standardwert: 0

ACKNOWLEDGE = *NO / *YES
Angabe, ob der Trap bestätigt werden muss. Es können nur Application Monitor-spezifische Traps bestätigt werden.
Standardwert: *NO

DEFINE-OBJECT

Logisch zusammengehörige Bestandteile eines Prozesses (Anwendungen, Protokolldateien, Subsysteme und Jobvariablen) können mit der Anweisung DEFINE-OBJECT in einer Gruppe (Objekt) zusammengefasst werden. Alle in der DEFINE-OBJECT-Anweisung genannten Elemente müssen mit den entsprechenden //ADD...-Anweisungen ebenfalls in der Konfigurationsdatei definiert werden.

```
//DEFINE-OBJECT

OBJECT-NAME = <composed-name_1 .. 8_with-underscore>
, BCAM-APPLICATION = *NONE / list-poss(5): <composed_name_1 .. 54_with-underscore>
, USER-APPLICATION = *NONE / list-poss(5): <composed_name_1 .. 54_with-underscore>
, DCAM-APPLICATION = *NONE / list-poss(5): <name_1 .. 8>
, LOG-FILE = *NONE / list-poss(5): <filename_1 .. 54> / <posix-pathname>
, SUBSYSTEM = *NONE / list-poss(5): <structured-name_1 .. 8>
, JV = *NONE / list-poss(10): <filename_1 .. 54>
, MONITORING-TIME = *ALWAYS / *INTERVAL (...)
    *INTERVAL ( ...)
        , START-TIME = hh:mm
        , STOP-TIME = hh:mm
        , EXCEPT-DAYS = *NONE / list-poss(6): MON / TUE / WED / THU / FRI / SAT / SUN
, ACKNOWLEDGE= *NO / *YES
```

OBJECT-NAME = <composed-name_1 .. 8_with-underscore>
Name des Objekts.

BCAM-APPLICATION = *NONE / list-poss(5): <composed_name_1 .. 54_with-underscore>
BCAM-Anwendungen, die zu diesem Objekt gehören.
Standardwert: *NONE

USER-APPLICATION = *NONE / list-poss(5): <composed_name_1 .. 54_with-underscore>
Benutzeranwendungen, die zu diesem Objekt gehören.
Standardwert: *NONE

DCAM-APPLICATION = *NONE / list-poss(5): <name_1 .. 8>
DCAM-Anwendungen, die zu diesem Objekt gehören.
Standardwert: *NONE

LOG-FILE = *NONE / list-poss(5): <filename_1 .. 54> / <posix-pathname>
Protokolldateien, die zu diesem Objekt gehören.
Standardwert: *NONE

SUBSYSTEM = *NONE / list-poss(5): <structured-name_1 .. 8>
Subsysteme, die zu diesem Objekt gehören.
Standardwert: *NONE

JV = *NONE / list-poss(10): <filename_1 .. 54>
Job-Variablen, die zu diesem Objekt gehören.
Standardwert: *NONE

MONITORING-TIME = *ALWAYS / *INTERVAL (...)
spezifiziert den Zeitraum der Überwachung.
Standardwert: *ALWAYS

***INTERVAL (...)**
spezifiziert das Überwachungsintervall. Wenn STOP-TIME größer als START-TIME ist, zählen bei der Überprüfung der EXCEPT-DAYS die Stunden nach Mitternacht zum vorherigen Tag.

Beispiel:

Die Überwachungszeit erstreckt sich von 20:00 bis 3.00 außer Samstag und Sonntag. Die Überwachung endet daher am Samstag um 3:00 morgens und beginnt wieder am Montag um 20:00 abends.

START-TIME = HH:MM
Zeitpunkt, ab dem das Objekt überwacht werden soll.

STOP-TIME = HH:MM
Zeitpunkt, bis zu dem das Objekt überwacht werden soll.

EXCEPT-DAYS = *NONE / list-poss(6): MON / TUE / WED / THU / FRI / SAT / SUN
Wochentage, an denen das Objekt nicht überwacht werden soll.
Standardwert: *NONE

ACKNOWLEDGE= *NO / *YES
Angabe, ob der Trap bestätigt werden muss.
Standardwert: *NO

DEFINE-TRAP-FORMAT

Die Anweisung DEFINE-TRAP-FORMAT legt das Trap-Format für den Application Monitor-Subagenten fest.

//DEFINE-TRAP-FORMAT
TYPE = list-poss(2) ***GENERIC** / ***TVCC**

TYPE = list-poss(2) *GENERIC / *TVCC
legt das Trap-Format fest.
GENERIC: Es wird das Application Monitor-spezifische Trap-Format verwendet.
TVCC: Es wird das TV-CC-spezifische Trap-Format verwendet.
Standardwert: *GENERIC

SET-TIMER-OPTIONS

Der Application Monitor-Subagent verwendet einen Timer. Den Wert für den Timer spezifizieren Sie im START-Kommando des Application Monitor-Subagenten (siehe Seite 53). Die Anweisung SET-TIMER-OPTIONS legt das Überwachungsintervall (Polling-Faktor) fest. Das Überwachungsintervall bestimmt, nach wievielen Timer-Abläufen die nächste Überprüfung durchgeführt werden soll.

//SET-TIMER-OPTIONS
FILES = **1** / <integer> , **SUBSYSTEMS** = **5** / <integer> , **DCAM-APPLICATIONS** = **60**/ <integer>

FILES = 1 / <integer>
legt den Polling-Faktor für Dateien fest.
Standardwert: 1

SUBSYSTEMS = 5 / <integer>
legt den Polling-Faktor für Subsysteme fest.
Standardwert: 5

DCAM-APPLICATIONS = 60 / <integer>
legt den Polling-Faktor für DCAM-Anwendungen fest.
Standardwert: 60

4.3.3 Start / Stopp des Application Monitor-Subagenten

Der Application Monitor-Subagent wird in der POSIX-Shell oder im BS2000/OSD gestartet.

1. Starten im BS2000/OSD:

/START-SNMP-APPMON
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54 without-gen-vers> , **CPU-LIMIT**=***STD** / <integer 1 .. 32767> / ***NO** , **JOB-CLASS**=***STD** / <name 1 .. 8> , **FILE-NAME**=***NONE** / <filename 1 .. 54 without-gen-vers> , **TIMER-INTERVAL** = 5 / <integer 1 .. 32767>

2. Starten in der POSIX-Shell mit:

```
appmonagt [-f <inputfile>]
          [-t <int>]
```

Beendet wird der Application Monitor (unabhängig von der Umgebung, in der er gestartet wurde) im BS2000/OSD mit:

/STOP-SNMP-APPMON
VERSION=***STD** / <product-version>

oder in der POSIX-Shell mit:

```
appmoncmd T
```

Beschreibung der Operanden:

VERSION=*STD / <product-version>
definiert die zu startende bzw. stoppende Version des Agenten. Diese Angabe wird derzeit nicht ausgewertet.

MONJV=*NONE / <filename 1..54 without-gen-vers>
Name der Jobvariable, die den Agenten überwachen soll. Die Angabe *NONE - keine Überwachung durch eine Jobvariable - ist Standard.

CPU-LIMIT=*STD / <integer 1 .. 32767> / *NO
Angabe der maximalen CPU-Laufzeit in Sekunden. Bei Angabe von *STD wird der generierte Standardwert verwendet.

JOB-CLASS=*STD / <name 1..8>
Jobklasse, mit der der Agent gestartet wird. Bei Angabe von *STD wird die generierte Standard-Jobklasse verwendet.

FILE-NAME=*NONE / <filename 1..54 without-gen-vers>
Beim Start des Application Monitor kann eine Konfigurationsdatei angegeben werden (siehe Seite 35). Wird keine Konfigurationsdatei angegeben, werden all diejenigen Subsysteme überwacht, die beim Starten des Application Monitor-Subagenten dem BS2000/OSD bekannt waren. Die Konfigurationsdatei, definiert durch die Angabe <filename> bzw. <input-file>, muss im BS2000/OSD-Filesystem abgespeichert sein.

TIMER-INTERVAL=5 / <integer 1 .. 32767>
Zeitintervall, in dem der Agent überprüft, ob Anforderungen vom Kommandoprogramm vorliegen. Das Zeitintervall ist standardmäßig auf fünf Sekunden eingestellt.
Die Dateiüberwachung wird bei Ablauf des Zeitintervalls durchgeführt.

4.4 Console Monitor-Subagent

Wie der Application Monitor-Subagent ist auch der Console Monitor-Subagent ein universeller Agent und keinem speziellen Produkt zugeordnet. Er kommuniziert mit der BS2000/OSD-Konsole.

4.4.1 Funktionalität des Console Monitor-Subagenten

Mit dem Console Monitor-Subagenten können Konsolmeldungen des BS2000/OSD erfasst, zur Reduzierung der Netzlast nach verschiedenen Kriterien gefiltert und als Trap an die Management-Plattform gesendet werden. Umgekehrt kann der Administrator an einer Management-Plattform BS2000/OSD-Konsolkommandos absetzen und sich die Ergebnisse der Kommando-Ausführung anzeigen lassen.

Bei einigen Management-Plattformen lassen sich automatisch Daten aus einer Meldung extrahieren und in bestimmte Konsolkommandos einsetzen, die dann als automatische Reaktion an das BS2000/OSD-System zurückgesendet werden.

Für den Console Monitor-Subagenten gibt es eine eigene Management-Anwendung, den Console und Application Monitor (siehe Seite 101).

4.4.2 Konfiguration des Console Monitor-Subagenten

Über UCON erhält der Console Monitor-Subagent den Zugriff auf die Konsolkommandos von $CONSOLE.

Folgende Vorbereitungen sind notwendig, um dem Console Monitor-Subagenten den Zugriff auf die BS2000/OSD-Konsole zu ermöglichen:

– Operator-Kennung <operator-id> einrichten

– Zugangsberechtigung für die Operatorkennung freischalten

Operator-Kennung <operator-id> einrichten

```
/ADD-USER USER-ID=<operator-id>,                              -
          PROTECTION-ATTRIBUTE=*PAR(LOGON-PASSWORD=<pass>),   -
          ACCOUNT-ATTRIBUTES=*PAR(ACCOUNT=<account-nr>)
```

Die hier festgelegten LOGON-Attribute müssen in der Start-Anweisung des Console Monitor-Subagenten (siehe START-SNMP-CONSMON auf Seite 63) angegeben werden.

Zugangsberechtigung für die Operatorkennung freischalten

Für den Betrieb mit SECOS muss zusätzlich noch die Zugangsberechtigung für die Operator-Kennung zu $CONSOLE freigeschaltet werden:

```
/MOD-LOGON-PROTECTION USER-IDENTIFICATION=<operator-id>,     -
                      OPERATOR-ACCESS-PROG=*YES(PASSWORD-CHECK=*YES)
```

Der Klasse-2-Systemparameter NBBAPRIV muss auf den Standardwert N eingestellt sein.

4.4.2.1 Definition von Meldungsfiltern

Bei der Auswahl von Meldungen verwendet der Console Monitor-Subagent zwei Filtervarianten:

– positiver Meldungsfilter
  wählt Meldungen aus, die an die Management-Plattform geschickt werden sollen.

– negativer Meldungsfilter
  wählt Meldungen aus, die nicht an die Management-Plattform geschickt werden dürfen.

Positiver Meldungsfilter

Für die Auswahl der Meldungen, die an die Management-Plattform geschickt werden, stehen zwei Filtermöglichkeiten zur Verfügung:

– Routingcode (ist jeder Konsolmeldung zugeordnet)
– Meldungsschlüssel (identifiziert jede Meldung eindeutig)

Auswahlkriterium Routingcode

Jede Meldung ist einem bestimmten Routingcode zugeordnet. Operator-Rollen enthalten die Routingcodes derjenigen Meldungen, die an die Management-Plattform geschickt werden sollen. Die Operator-Rollen werden in der Start-Anweisung des Console Monitor-Subagenten (siehe START-SNMP-CONSMON auf Seite 63) angegeben. Die folgenden Anweisungen zeigen, wie Operator-Rollen erzeugt und der Operator-Kennung zugeordnet werden. Voraussetzung für das Absetzen der folgenden Anweisungen ist das Privileg SECURITY-ADMINISTRATION, das standardmäßig die Benutzerkennung SYSPRIV hat.

Erzeugen der Operator-Rolle:

```
/CREATE-OPERATOR-ROLE OP-ROLE=<op-role-name>,                -
                      ROUTING-CODES=.......
```

Zuordnung der Operator-Rollen zur Operator-Kennung:

```
/MODIFY-OPERATOR-ATTR USER-ID=<operator-id>,                 -
                      ADD-OPERATOR-ROLE=(<op-role-name1>,...,<op-role-namex>)
```

Bei Einsatz von SECOS muss außerdem der Operator-Kennung das Privileg OPERATING zugewiesen werden:

```
/SET-PRIVILEGE PRIV=OPERATING,USER-ID=<operator-id>
```

Auswahlkriterium Meldungsschlüssel

Die Meldungsschlüssel derjenigen Meldungen, die der Management-Plattform zugestellt werden sollen, werden in der positiven Meldungsfilter-Datei hinterlegt. Drei Filtermöglichkeiten stehen mit den folgenden Anweisungen zur Verfügung:

– *msgid*,
– *QUESTION*
– *TYPIO*

Der Name der Meldungsfilter-Datei wird dem Console Monitor bei dessen Start durch die Angabe MSG-FILTER mitgeteilt. Im laufenden Betrieb kann der Dateiname in dem MIB-Objekt *consMonMsgFilter* eingetragen werden.

Fehlt die Angabe einer Meldungsfilter-Datei beim Start des Console Monitor-Subagenten, werden alle Meldungen ausgegeben, deren Routingcode in der Operator-Rolle angegeben ist.

Enthält die Meldungsfilter-Datei keine bzw. keine gültigen Meldungsschlüssel, dann werden der Management-Plattform keine Traps zugestellt. Eine leere Meldungsfilter-Datei einzurichten ist nur dann sinnvoll, wenn Sie gleichzeitig den HIPLEX OP-Agenten zur Überwachung der BS2000/OSD-Konsolmeldungen im Einsatz haben, aber auf die Eingabe von Konsolkommandos mithilfe des Console Monitor nicht verzichten wollen.

Für die Meldungsfilter-Datei gelten folgende Namenskonventionen:

/BS2/<datei>	BS2000/OSD-Datei
[:<catid>:]$<userid>.<datei>	BS2000/OSD-Datei
*POSIX(<datei>)	POSIX-Datei
/<pfad>/<datei>	POSIX-Datei
<datei>	in diesem Fall ist ausschlaggebend, in welcher Umgebung der Subagent gestartet wurde.

Aufbau des positiven Meldungsfilters

msgid

```
<msgid [wgt] [SOURCE=src] [DEVICE=dev] [PATTERN=/pat1[/..patx]] [ACKNOWLEDGE=YES]>
```

msgid Angabe eines Meldungsschlüssels.

Bei der Angabe von Meldungsschlüsseln sind folgende Wildcard-Angaben zulässig:

? :	ersetzt ein beliebiges Zeichen
* :	ersetzt eine beliebig lange Folge von Zeichen
[s] :	ersetzt genau ein Zeichen aus der Zeichenkette s
[c1 - c2]:	ersetzt ein beliebiges Zeichen aus dem Bereich von c1 bis c2

Das Zeichen "\" (Backslash) muss zur Entwertung der Sonderzeichen angegeben werden. Es wird zwischen Groß- und Kleinschreibung unterschieden.

wgt Angabe eines Meldungsgewichts (weight). Den Meldungsschlüsseln kann ein Gewicht zugeordnet werden. Dieses Gewicht wird im Trap-String der eigentlichen Meldung vorangestellt. Damit hat der Anwender die Möglichkeit, die Wichtigkeit der Meldungen selbst einzustellen und entsprechend an der Management-Plattform darzustellen. Fehlt die Gewichtsangabe, erhält der Meldungsschlüssel standardmäßig den Wert 0.

Die Angabe wird als Integer mit dem Wertebereich 0 - 999 erwartet.

src Angabe eines Quellennamens (source). Im Trap-String wird Quelle mit BS2-<source> versorgt. Fehlt diese Angabe, wird der Standardwert *BS2Console* eingesetzt. Mit dieser Angabe können Sie einen Alarm gezielt auf ein Objekt im Netzbild lenken. Die Angabe erfolgt alphanumerisch in der Länge 1 - 12 (siehe „Handbuch „SNMP Management V5.0“).

pat Angabe eines oder mehrerer Suchmuster (pattern).

? :	ersetzt ein beliebiges Zeichen
* :	ersetzt eine beliebig lange Folge von Zeichen
[s] :	ersetzt genau ein Zeichen aus der Zeichenkette s
[c1 - c2]:	ersetzt ein beliebiges Zeichen aus dem Bereich von c1 bis c2

Das Zeichen "\" (Backslash) muss zur Entwertung der Sonderzeichen angegeben werden. Es wird zwischen Groß- und Kleinschreibung unterschieden.

dev Ist ein DEVICE angegeben, sendet der Console Monitor-Subagent diesen Trap mit der DEVICE-Angabe als Community (siehe Handbuch „SNMP Management V5.0“).

ACKNOWLEDGE=YES

Durch die Angabe ACKNOWLEDGE=YES wird dem Subagenten angezeigt, dass dieser Trap bestätigt werden muss.

QUESTION

Question filtert alle Meldungen heraus, die eine Frage beinhalten, d.h. die eine Antwort erwarten.Tritt eine Frage auf, überprüft der Console Monitor zuerst, ob ein Muster der QUESTION-Einträge passt. Ist das nicht der Fall, werden dem Meldungstyp entsprechend die MSGID-Einträge oder die TYPIO-Einträge durchsucht.

```
<QUESTION [wgt] [SOURCE=src] [DEVICE=dev] [PATTERN=/pat1[/..patx]]
[ACKNOWLEDGE=YES]>
```

QUESTION Meldungsschlüssel einer Konsolanfrage

wgt	siehe oben
src	siehe oben
dev	siehe oben
pat	siehe oben
ACKNOWLEDGE=YES	siehe oben

Beispiel:

`<QUESTION PATTERN=[0-9]*>` Auswahl aller Fragen, die mit einer Ziffer beginnen.

TYPIO

Eine Sonderstellung nehmen so genannte TYPE I/O-Meldungen ein. Zu den TYPE I/O-Meldungen zählen beispielsweise Nachrichten, die mit /SEND-MSG der BS2000/OSD-Konsole zugestellt werden. Die Weiterleitung der TYPE I/O-Meldungen als SNMP-Trap wird ebenfalls über die Meldungsfilter-Datei gesteuert. Der Eintrag für eine TYPE I/O-Meldung ist folgendermaßen aufgebaut:

```
<TYPIO [wgt] [SOURCE=src] [DEVICE=dev] [PATTERN=/pat1[/..patx]] [ACKNOWLEDGE=YES]>
```

wgt	siehe oben
src	siehe oben
dev	siehe oben
pat	siehe oben
ACKNOWLEDGE=YES	siehe oben

Beispiel:

```
<TYPIO PATTERN=/*abc*/xyz>
<TYPIO PATTERN=/Hallo*>
<TYPIO PATTERN=/?\?*>
```

Alle TYPE I/O-Meldungen, die den String "abc" enthalten, nur aus "xyz" bestehen, mit "Hallo" beginnen oder an zweiter Stelle ein Fragezeichen haben, werden als Trap der Management-Plattform zugestellt.

Ein Beispiel für einen Meldungsfilter finden Sie in der Bibliothek SINLIB.SBA-BS2.060.

Negativer Meldungsfilter

Für den Console Monitor-Subagenten wird auch ein negativer Meldungsfilter angeboten. Die Meldungsschlüssel derjenigen Konsolmeldungen, die nicht an die Management-Plattform durchgereicht werden sollen, werden in der negativen Meldungsfilter-Datei hinterlegt. Jeder Meldungsschlüssel kann optional um ein oder mehrere Suchmuster ergänzt werden. Fragen können nicht unterdrückt werden. Das MIB-Objekt *consMonNegMsgFilter* verweist auf den Namen der negativen Meldungsfilter-Datei. Der Name der negativen Meldungsfilter-Datei wird beim Start des Console Monitor-Subagenten mit dem Operanden SUPPRESS-MSG-FILE definiert. Diese Definition kann nur beim Start des Console Monitor-Subagenten, aber nicht im laufenden Betrieb geändert werden.

Die Länge eines Eintrags darf maximal 179 Zeichen betragen.

```
<msgid [PATTERN=/pat1[/..patx]]> [<msgid [PATTERN=/pat1[/..patx]]>] ...
```

pat siehe oben

Trap-Format

Zusätzlich wird das Trap-Format in der Meldungsfilter-Datei festgelegt:

```
TRAP-FORMAT=GENERIC / TVCC / ALL
```

GENERIC
: Es wird nur der für den Application Monitor-spezifische Trap verwendet. GENERIC ist Standardwert

TVCC
: Es wird nur TVCC-Trap-Format verwendet.

ALL
: Es werden beide Trap-Formate verwendet.

4.4.2.2 Ändern der Meldungsfilter-Datei im laufenden Betrieb

Änderungen der aktuellen Meldungsfilter-Datei im laufenden Betrieb können mit der Console Monitor-Anwendung entweder durch Setzen des Objekts *consMonMsgFilter* oder per Kommando vorgenommen werden.

```
/START-CONSMONCMD
       x "readConfig <filename>"
```

POSIX:

```
consmoncmd x "readConfig <filename>"
```

Bei Syntaxfehlern in der Meldungsfilter-Datei *consMonMsgFilter* wird mit der ursprünglichen Meldungsfilter-Datei weitergearbeitet.

Beispiel: Filtern von Konsolmeldungen

Die Meldung EXC0858 soll nur an die Management-Plattform geschickt werden, wenn sie weder den String "CLAQ" noch den String "TEST" enthält. Der Trap soll mit der Trapnummer 99 geschickt werden und als Quelle soll "Hardware" eingetragen sein.

Sie erreichen dies wie folgt:

- Tragen Sie im positiven Meldungsfilter ein: `<EXC0858 99 SOURCE=Hardware>`
- Tragen Sie in der negativen Meldungsfilterdatei ein:

  ```
  <EXC0858 PATTERN = *CLAQ* / *TEST*>
  ```

4.4.3 Start / Stopp des Console Monitor-Subagenten

Der Console Monitor-Subagent wird in der POSIX-Shell oder im BS2000/OSD gestartet.

1. Starten im BS2000/OSD:

/START-SNMP-CONSMON
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54> , **CPU-LIMIT**=***STD** / <integer 1 .. 32767> / ***NO** , **JOB-CLASS**=***STD** / <name 1 .. 8> , **OPERATOR-ID**= <name 1 .. 8> , **PASSWORD**=***NONE** / <c-string 1 .. 8> / ***SECRET** , **OPERATOR-ROLE**= **list-poss(10)** <name 1 .. 8> , **MSG-FILTER**=***NONE** / <filename 1 .. 54> / <posix-pathname> , **SUPPRESS-MSG-FILE** = ***NONE** / <filename 1 .. 54> / <posix-pathname> , **TIMER-INTERVAL** = 5 / <integer 1 .. 32767>

2. Starten in der POSIX-Shell mit:

```
consmonagt  -o <operid>
            [-t <int>]
            [-p <password>]
            [-f <msg-filter>]
            [-n <negative-msg-filter>]
            <op-role1> [,<op-role2>, ...., <op-role10>]
```

Beendet wird der Console Monitor (unabhängig von der Umgebung, in der er gestartet wurde) im BS2000/OSD mit:

/STOP-SNMP-CONSMON
VERSION=***STD** / <product-version>

oder in der POSIX-Shell mit:

```
consmoncmd T
```

Beschreibung der Operanden:

VERSION=*STD / <product-version>
definiert die zu startende bzw. stoppende Version des Agenten. Diese Angabe wird derzeit nicht ausgewertet.

MONJV=*NONE / <filename 1..54 without-gen-vers>
Name der Jobvariable, die den Agenten überwachen soll.
Die Angabe *NONE - keine Überwachung durch eine Jobvariable - ist Standard.

CPU-LIMIT=*STD / <integer 1 .. 32767> / *NO
Angabe der maximalen CPU-Laufzeit in Sekunden.
Bei Angabe von *STD wird der generierte Standardwert verwendet.

JOB-CLASS=*STD / <name 1..8>
Jobklasse, mit der der Agent gestartet wird.
Bei Angabe von *STD wird die generierte Standard-Jobklasse verwendet.

OPERATOR-ID=<name 1 .. 8>
Benutzerkennung, mit der sich der Subagent bei $CONSOLE anmeldet.

PASSWORD=*NONE / <c-string 1 .. 8> / *SECRET
Definition des Passworts, das den Subagenten zum Zugriff auf $CONSOLE berechtigt. Die Standardangabe *NONE sorgt dafür, dass kein Passwort angegeben werden muss. *SECRET bewirkt, dass das Feld zur Passworteingabe dunkel gesteuert ist.

OPERATOR-ROLE=list-poss(10) <name 1 .. 8>
Name der Operator-Rolle, die die zur Konsolüberwachung relevanten Routingcodes enthält.

MSG-FILTER=*NONE / <filename 1 .. 54> / <posix-pathname>
Name der Datei (<filename> bzw. <posix-pathname>), die die relevanten Meldungsschlüssel enthält.
*NONE (Standardwert) bedeutet, es wird keine Datei mit Meldungsschlüsseln zugewiesen.

SUPPRESS-MSG-FILE=*NONE / <filename 1 .. 54> / <posix-pathname>
Die mit <filename> bzw. <posix-pathname> definierte Datei enthält die zu unterdrückenden Konsolmeldungsschlüssel.
*NONE (Standardwert) bedeutet, es wird keine Datei mit zu unterdrückenden Meldungsschlüsseln zugewiesen.

TIMER-INTERVAL=5 / <integer 1 .. 32767>
Zeitintervall, in dem der Agent überprüft, ob Anforderungen vom Kommandoprogramm vorliegen. Das Zeitintervall ist standardmäßig auf fünf Sekunden eingestellt.

4.5 Event-Subagent

Mit dem Event-Subagenten können Sie MIB-Objekte anderer Subagenten durch periodische SNMP-Abfragen überwachen und einfache Aktionen ausführen, sobald bestimmte Bedingungen (Trigger Tests) erfüllt sind. Aktionen können das Senden von Traps oder die Ausführung von SNMP-Set-Operationen sein.

Bild 4 veranschaulicht das Zusammenwirken von Masteragent und Event-Subgent bei der Subagentenüberwachung.

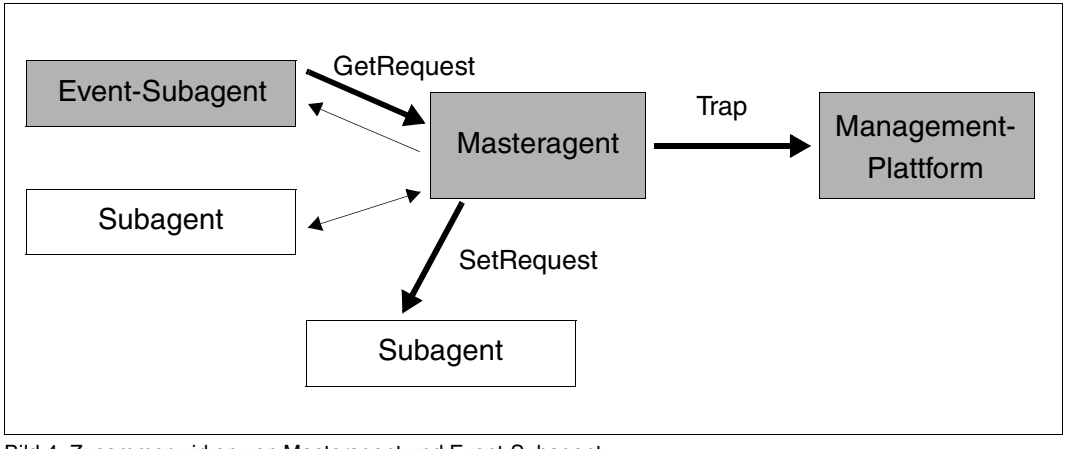

Bild 4: Zusammenwirken von Masteragent und Event-Subagent

4.5.1 Funktionalität des Event-Subagenten

Der Event-Subagent implementiert die Event MIB (RFC2981). Die Event-MIB ist in die folgenden beiden Abschnitte (Sections) unterteilt, die den Funktionsumfang des Event-Subagenten festlegen:

– Trigger-Section
– Event-Section

Diese Sections werden über Tabellen in einer Konfigurationsdatei konfiguriert.

Trigger Section

Die Trigger Section definiert die zu überwachenden MIB-Objekte sowie die Bedingungen (Trigger Tests), bei denen ein Ereignis auslöst wird, wie z.B.

- **TriggerValueID** spezifiziert die OID des zu prüfenden MIB-Objekts.
- **TriggerTest** spezifiziert die zu testenden Bedingungen, z.B.
  - existence: ob das in TriggerValueID spezifizierte MIB-Objekt existiert.
  - boolean: ob der Wert dieses MIB-Objekts mit einem in der BooleanTable definierten Wert (TriggerBooleanValue) übereinstimmt.
  - threshold: ob dieser Wert einen in der TresholdTable definierten Schwellenwert über- oder unterschreitet.
- **TriggerSampleType** spezifiziert, ob der Vergleichswert als absoluter Wert (absolute) oder als Differenz (delta) zu einem bei einer früheren Abfrage ermittelten Wert interpretiert werden soll.
- **TriggerFrequency** spezifiziert das Zeitintervall (in Sekunden) zwischen zwei aufeinanderfolgenden Abfragen.

Abhängig vom Test-Typ sind Einträge in einer weiteren Tabelle erforderlich:

- **ExistenceTable:** Objekt existiert / verschwindet / ändert Wert.
- **BooleanTable:** Vergleichstest des Objekt- bzw. Delta-Werts mit dem Vergleichswert.
- **ThresholdTable:** Objekt- bzw. Delta-Wert überschreitet / unterschreitet Grenzwert.

Bei positivem Testergebnis wird für die auszuführende Aktion ein passender Eintrag in der EventTable gesucht.

Event Section

Die Event Section definiert im Objekt EventAction, welche Aktion - SNMP-Trap und/oder SNMP-SetRequest - als Reaktion auf einen erfolgreichen Trigger Test ausgelöst werden soll:

- **notification** (SNMP-Trap): Die OID des zu sendenden Traps wird in der EventNotificationTable festgelegt.
- **set** (SNMP-SetRequest): Die OID des MIB-Objekts und der zu setzende Wert werden in derEventSetTable festgelegt.

Notifications

Die Event-MIB bietet folgende Traps an, die als Reaktion auf ausgelöste Ereignisse gesendet werden können.

- **TriggerSenseAlarm** meldet, dass der Trigger, der ein Objekt überwacht, ausgelöst wurde.
- **TriggerRaisingAlarm** meldet, dass der Schwellenwert überschritten wurde.
- **TriggerFallingAlarm** meldet, dass der Schwellenwert unterschritten wurde.

Die bei einer Notification mitgegebenen Objekte sind in der **Objects Table** eingetragen und werden angegeben im jeweiligen Eintrag in der

– Trigger Table
– ExistenceTable / Boolean Table / Threshold Table und / oder
– Notification Table

4.5.2 Konfiguration des Event-Subagenten

Die Beschreibung der Konfiguration des Event-Subagenten finden Sie als Kommentar in der mitgelieferten Beispiel-Konfigurationsdatei des Event-Subagenten.

Beispiele

Mit MIB-II: Trap senden, wenn

- rUtilization = ((delta(iflnOctets)*8) / (ifSpeed * delta(seconds)) >x
- rErrorRate = ((delta(iflnErrors)) / (delta(seconds)) > x)
- comSecurit = (delta(snmplBadCommunityNames) > 0)

Mit Sar MIB: Trap senden, wenn

- idleTime = (sm“TimeIOmachTabldleTime < x)

Für das erste Beispiel ist die folgende Konfiguration erforderlich:

```
rUtilization = ((delta(iflnOctets)*8) / (ifSpeed * delta(seconds)) > 20%


mteTriggerEntry
         owner1                    TriggerOwner
         tInUtil                   Triggername
         "Utilization if1"         Kommentar
         20                        Test: threshold
         2                         Sampletype: delta
         1.3.6.1.2.1.2.2.1.10.1    zu überwachende OID: ifInOctets von interface 1
         –                         –
         –                         –
         –                         –
         –                         –
         100                       Frequency: 100 sec
         –                         –
         –                         –
         1                         trigger enabled
         1                         active

mteTriggerThresholdEntry
(erweitert die Trigger-Tabelle über gemeinsamen Index 'owner1 tInutil')
          1                        StartUp: rising
          –                        –
          –                        –
          25000000                 DeltaRising: 25.000.000 InOctets/100 sec.
          –                        –
          –                        –
          –                        –
          –                        –
```

```
          -                          -
          -                          -
          -                          -
          owner1                     DeltaRisingEventOwner (Index in EventTable)
          evInUtil                   DeltaRisingEvent      (Index in EventTable)
          -                          -
          -                          -
          owner1                     TriggerOwner  (Index in TriggerTable)
          tInUtil                    TriggerName   (Index in TriggerTable)

mteEventEntry
          evInUtil                   Name        (Index in dieser EventTable)
          "Utilization if1"          Kommentar
          80                         Actions: notification
          1                          true
          1                          active
          owner1                     Owner       (Index in dieser EventTable)


mteEventNotificationEntry
(erweitert die EventTabelle über gemeinsamen Index 'owner1 evInUtil')
          1.3.6.1.2.1.88.2.0.2       Notification: mteTriggerRising
          owner1                     ObjectsOwner (Index in ObjectTable)
          oInUtil                    ObjectsName  (Index in ObjectTable)
          owner1                     EventOwner   (Index in EventTable)
          evInUtil                   EventName    (Index in EventTable)

mteObjectsEntry
          oInUtil                    Name      (Index in dieser ObjectTable)
          1                          Subindex  (Index in dieser ObjectTable)
          1.3.6.1.2.1.2.2.1.10.1     OID: ifInOctets von interface 1
          2                          -
          1                          active
          owner1                     Owner     (Index in dieser ObjectTable)

mteObjectsEntry
          oInUtil                    Name      (Index in dieser ObjectTable)
          2                          Subindex  (Index in dieser ObjectTable)
          1.3.6.1.2.1.2.2.1.5.1      OID: ifSpeed von interface 1
          2                          -
          1                          active
          owner1                     Owner     (Index in dieser ObjectTable)
```

4.5.3 Start / Stopp des Event-Subagenten

Der Event-Subagent wird in der POSIX-Shell oder im BS2000/OSD gestartet.

Für das Starten des Event-Subagenten ist das Privileg NET - ADMINISTRATION erforderlich.

1. Starten im BS2000/OSD:

/START-SNMP-EVENTAGT
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54 without-gen-vers> , **CPU-LIMIT**=***STD** / <integer 1 .. 32767> / ***NO** , **JOB-CLASS**=***STD** / <name 1 .. 8> **, FILE-NAME**=***STD** / <posix-pathname_1..1023>

2. Starten in der POSIX-Shell mit:

```
eventagt  [-f <config-file>]
```

Beendet wird der Event-Subagent (unabhängig von der Umgebung, in der er gestartet wurde) im BS2000/OSD mit:

/STOP-SNMP-EVENTAGT
VERSION=***STD** / <product-version>

oder in der POSIX-Shell mit:

```
eventcmd T
```

Beschreibung der Operanden:

VERSION=*STD / <product-version>
definiert die zu startende bzw. stoppende Version des Agenten. Diese Angabe wird derzeit nicht ausgewertet.

MONJV=*NONE / <filename 1..54 without-gen-vers>
Name der Jobvariable, die den Agenten überwachen soll. Die Angabe *NONE - keine Überwachung durch eine Jobvariable - ist Standard.

CPU-LIMIT=*STD / <integer 1 .. 32767> / *NO
Angabe der maximalen CPU-Laufzeit in Sekunden.
Bei Angabe von *STD wird der generierte Standardwert verwendet.

JOB-CLASS=*STD / <name 1..8>
Jobklasse, mit der der Agent gestartet wird.
Bei Angabe von *STD wird die generierte Standard-Jobklasse verwendet.

FILE-NAME=*STD / <posix-pathname_1..1023>
Name der Konfigurationsdatei.
Bei Angabe von *STD wird die Datei */etc/snmp/agt/event.cnf* verwendet.

4.6 Scheduler-Subagent

Der Scheduler-Subagent ermöglicht es, Änderungen an Objekten periodisch oder zu definierten Zeitpunkten auszuführen. So ist es beispielsweise möglich, eine SNMP-Set-Operation jeden Montag um 6:00 Uhr abzusetzen oder auch an jedem letzten Freitag im Monat um 22 Uhr. Ein solches Scheduling kann durch Modifizieren eines Kontrollobjekts in oder außer Kraft gesetzt werden. Dies ermöglicht ein vorkonfiguriertes Scheduling, das von anderen Management-Funktionen aktiviert oder deaktiviert werden kann.

Typischer Anwendungsfall für die vom Scheduler-Subagenten gesteuerten SNMP-Set-Operationen ist das Ändern des Adminstatus von MIB-Instanzen zu festgelegten Zeitpunkten, z.B.:

– Status eines Interfaces durch Setzen von *ifAdminStatus*
– Überwachungsstatus im Event-Subagenten durch das Setzen von *mteTriggerEnabled*
– Status einer Anwendungsüberwachung des Application Monitor-Subagenten durch Setzen von *appMonLogFState*

Bild 5 veranschaulicht die Arbeitsweise des Scheduler-Subagenten.

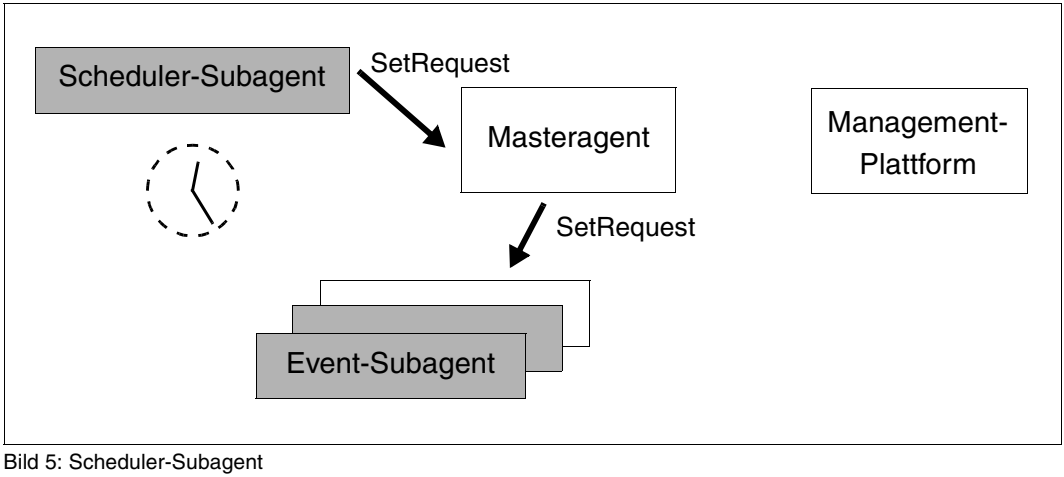

Bild 5: Scheduler-Subagent

4.6.1 Funktionalität des Scheduler-Subagenten

Der Scheduler-Subagent implementiert die Scheduler MIB (RFC2591), die folgende Arten des Scheduling unterstützt:

– Periodisches Scheduling
– Scheduling auf der Basis von kalendarischen Daten
– Einzel-Scheduling

Periodisches Scheduling

Periodisches Scheduling basiert auf festgelegten Zeitintervallen zwischen zwei aufeinanderfolgenden, vom Scheduler-Subagenten initiierten SNMP-Set-Operationen. Ein Zeitintervall definieren Sie durch die Anzahl der Sekunden, die zwischen zwei aufeinander folgenden SMNP-Anweisungen vergehen.

Scheduling auf Basis kalendarischer Daten

Scheduling auf Basis kalendarischer Daten initiiert Aktionen an festgelegten Wochentagen oder bestimmten Tagen eines Monats. Einen kalendarischen Zeitpunkt spezifizieren Sie durch Angabe von Monat, Tag, Wochentag, Stunde und Minute.

Für jedes Datum können Sie eine Vielzahl von Werten spezifizieren und auf diese Weise ein komplexes Scheduling definieren. Das Scheduling kann beispielsweise an einem festgelegten Wochentag alle 15 Minuten eine bestimmte Aktion anstoßen.

Datumsangaben, basierend auf Monaten, Tagen und Wochentagen können Sie mithilfe der folgenden Scheduler MIB-Objekte des Typs BITS festlegen:

– *schedMonth*
– *schedDay*
– *schedWeekDay*

Das Setzen mehrerer Bits in einem dieser MIB-Objekte hat die Wirkung einer logischen ODER-Knüpfung. Wenn Sie beispielsweise in *schedWeekDay* die Bits monday (1) und friday (5) setzen, initiiert das Scheduling die Aktionen genau an Montagen und Freitagen.

Die objekt-übergreifende Kombination der Bitfelder von *schedMonth*, *schedDay* und *schedWeekDay* hat den Effekt einer logischen UND-Verknüpfung. Wenn Sie z.B. die Bits june (5) und july (6) in *schedMonth* setzen und die Bitfelder monday (1) und friday (5) in *schedWeekDay* setzen, beschränkt sich das Scheduling darauf, Aktionen ausschließlich montags und freitags in den Monaten Juni und Juli zu initiieren.

Wildcard-Funktionalität bei Datumsangaben erzielen Sie, wenn Sie alle Bits auf „1“ setzen.

Einzel-Scheduling

Einzel-Scheduling ähnelt dem Scheduling auf Basis kalendarischer Daten. Der Unterschied besteht darin, dass Einzel-Scheduling sich nach dem Anstoßen einer Aktion automatisch außer Kraft setzt.

Aktionen

Die vom Scheduling initiierten Aktionen modellieren SNMP-Set-Operationen auf MIB-Objekte, deren OID im Objekt *schedVariable* konfiguriert wird. Der zu setzende Wert wird im Objekt *schedValue* spezifiziert. In dieser MIB definierte Aktionen sind auf Objekte des Typs INTEGER beschränkt. Diese Einschränkung mindert jedoch nicht die Verwendbarkeit der Scheduler MIB. So ist einfaches Scheduling möglich, wie z.B. in Betrieb/ausser Betrieb-Scheduling für Ressourcen, zu denen es ein korrespondierendes Status-MIB-Objekt (z.B. *ifAdminStatus*) gibt.

4.6.2 Konfiguration des Scheduler-Subagenten

Die Beschreibung der Konfiguration des Scheduler-Subagenten finden Sie als Kommentar in der mitgelieferten Beispiel-Konfigurationsdatei des Scheduler-Subagenten.

```
# Entry type: schedEntry
# Format: schedOwner            String   (Index)
#         schedName             String   (Index)
#         schedDescr            String
#         schedInterval         Integer  (fuer schedType == periodic(1))
#         schedWeekDay          BITS     (fuer schedType == calendar(2), oneshot(3))
#         schedMonth            BITS     (fuer schedType == calendar(2), oneshot(3))
#         schedDay              BITS     (fuer schedType == calendar(2), oneshot(3))
#         schedHour             BITS     (fuer schedType == calendar(2), oneshot(3))
#         schedMinute           BITS     (fuer schedType == calendar(2), oneshot(3))
#         schedContextName      String   (not yet)
#         schedVariable         OID      (das zu setzende Objekt)
#         schedValue            SR_INT32 (der zu setzende Wert)
#         schedType             SR_INT32 (periodic(1), calendar(2), oneshot(3))
#         schedAdminStatus      SR_INT32 (enabled(1), disabled(2))
#         schedStorageType      SR_INT32 (other(1), volatile(2), nonVolatile(3),
#                                                   permanent(4), readOnly(5))

# Beispiel für BITS-Codierung
# Wochentag: Montag und Donnerstag wird codiert als 48
#   Sonntag
#|Montag
#   |   |   Dienstag
#   |   |   |   Mittwoch
#   |   |   |   |   Donnerstag
#   |   |   |   |   |   Freitag
#   |   |   |   |   |   |   Sonnabend
#   0   1   0   0   1   0   0
#   |           |   |
#   -------------   -----------
#         |               |
#         4               8                => 48
```

Beispiele

Scheduler-Subagent mit MIB-II: Interface am Wochenende herunterfahren.

```
ifOperStatus = down jeden Freitag 18:30 Uhr
ifOperStatus = up jeden Montag 07:30 Uhr
```

Scheduler-Subagent mit Event MIB: Überwachung am Wochenende ausschalten.

```
mteTriggerEnabled.xxx = false jeden Freitag 18:30 Uhr
mteTriggerEnabled.xxx = true jeden Montag 07:30 Uhr
```

Scheduler-Subagent mit Crit-Appl-MIB: Anwendungsüberwachung zu bestimmten Zeiten einschalten:

```
status = (appMonLogFState == start-begin)
```

Für das erste Beispiel (ifOperStatus = down jeden Freitag 18:30 Uhr) ist die folgende Konfiguration erforderlich:

```
Entry type: schedEntry
Format: schedOwner            Owner    (Index)
        schedName             IfDown   (Index)
        schedDescr            "     "

        schedInterval         0
        schedWeekDay          04                                  - Freitag
        schedMonth            FF:F0                               - jeder
        schedDay              FF:FF:FF:FE:00:00:00:00             - jeder
        schedHour             00:00:20                            - 18
        schedMinute           00:00:00:02:00:00:00                - 30

        schedContextName      -
        schedVariable         iso.3.6.1.2.1.2.2.1.7.1             - ifOperStatus
        schedValue            2                                   - down
        schedType             calendar                            - Kalender

        schedAdminStatus      enabled
        schedStorageType      readonly
```

4.6.3 Start / Stopp des Scheduler-Subagenten

Der Scheduler-Subagent wird in der POSIX-Shell oder im BS2000/OSD gestartet.

Für das Starten des Scheduler-Subagenten ist das Privileg NET- ADMINISTRATION erforderlich.

1. Starten im BS2000/OSD:

/START-SNMP-SCHEDULER
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54 without-gen-vers> , **CPU-LIMIT**=***STD** / <integer 1 .. 32767> / ***NO** , **JOB-CLASS**=***STD** / <name 1 .. 8> **, FILE-NAME**=***STD** / <posix-pathname_1..1023>

2. Starten in der POSIX-Shell mit:

```
schedagt  [-f <config-file>]
```

Beendet wird der Scheduler-Subagent (unabhängig von der Umgebung, in der er gestartet wurde) im BS2000/OSD mit:

/STOP-SNMP-SCHEDULER
VERSION=***STD** / <product-version>

oder in der POSIX-Shell mit:

```
schedcmd T
```

Beschreibung der Operanden:

VERSION=*STD / <product-version>
definiert die zu startende bzw. stoppende Version des Agenten. Diese Angabe wird derzeit nicht ausgewertet.

MONJV=*NONE / <filename 1..54 without-gen-vers>
Name der Jobvariable, die den Agenten überwachen soll. Die Angabe *NONE - keine Überwachung durch eine Jobvariable - ist Standard.

CPU-LIMIT=*STD / <integer 1 .. 32767> / *NO
Angabe der maximalen CPU-Laufzeit in Sekunden. Bei Angabe von *STD wird der generierte Standardwert verwendet.

JOB-CLASS=*STD / <name 1..8>
Jobklasse, mit der der Agent gestartet wird. Bei Angabe von *STD wird die generierte Standard-Jobklasse verwendet.

FILE-NAME=*STD / <posix-pathname_1..1023>
Name der Konfigurationsdatei. Bei Angabe von *STD wird die Datei */etc/snmp/agt/scheduler.cnf* verwendet.

5 Produktspezifische Agenten - Funktionale Erweiterungen zu SNMP V6.0

Dieses Kapitel beschreibt

- den HIPLEX-Subagenten sowie
- den erweiterten Funktionsumfang des *open*UTM-Subagenten in SNMP V6.0.

Der Funktionsumfang der anderen produktspezifischen Agenten - Agenten der SNMP Standard-Collection und Subagent für die Performance-Überwachung mit SM2 - bleibt gegenüber der SNMP V5.0 unverändert. Die Informationen zu diesen Subagenten im Handbuch „SNMP Management V5.0“ sind deshalb weiterhin uneingeschränkt gültig.

5.1 HIPLEX-Subagent

Der HIPLEX-Subagent überwacht Hochverfügbarkeits-Cluster aus BS2000/OSD-Systemen und liefert Informationen über die aktuelle Clusterkonfiguration aus der Sicht eines Teilsystems. Wichtige Ereignisse und Änderungen im Cluster meldet der HIPLEX-Subagent durch Traps an die Management-Plattform und ermöglicht so ein effizientes Clustermonitoring. Der HIPLEX-Subagent verwendet Schnittstellen der Produkte HIPLEX MSCF und HIPLEX AF. Ein sinnvoller Einsatz des HIPLEX-Subagenten setzt diese beiden Produkte in den Versionen 1.0 (HIPLEX MSCF) bzw. 3.0 (HIPLEX AF) voraus.

5.1.1 Funktionalität des HIPLEX-Subagenten

Der HIPLEX-Subagent unterstützt die HaCl-MIB (High Availability Cluster MIB). Es handelt sich dabei um eine allgemein definierte MIB zur Überwachung und Darstellung der Konfiguration in Hochverfügbarkeitsclustern. Die nachfolgende Beschreibung gibt einen Überblick, wie der HIPLEX-Subagent die HaCl-MIB unterstützt, und weist auf die Besonderheiten hin, die sich aus den Konzepten von HIPLEX MSCF und HIPLEX AF ergeben.

Objekte der HaCl-MIB

Der Objektbereich der HaCl-MIB gliedert sich in die folgenden Teilbereiche:

- Allgemeine Daten zum Subagenten und zum Cluster
- Systemtabelle
- Anwendungstabelle
- Ressourcentabelle
- Tabelle zur Anzeige des Anwendungsstatus auf den verbundenen Systemen
- Tabelle zur Anzeige des Ressourcenstatus auf den verbundenen Systemen

Allgemeine Daten zum Subagenten und zum Cluster

Zum Subagenten werden folgende Parameter angezeigt:

- Version des Subagenten
- Beschreibungstext
- Subagententyp
- Informationsbereich
- Notifikationsbereich
- Name des lokalen Systems

Der HIPLEX-Subagent liefert die Informationen über die Clusterkonfiguration aus der Sicht des lokalen Systems.

Informations- und der Notifikationsbereich können durch das Setzen zweier MIB-Objekte bestimmt werden.

Folgende Werte können eingestellt werden:

no: Es wird keine systemspezifische Information geliefert bzw. es wird kein Trap mit systemspezifischen Informationen gesendet.

local: Es werden nur Informationen angezeigt bzw. Traps gesendet, die das lokale System betreffen.

down: Die Umschalteinheit ist deaktiviert auf allen Systemen, auf denen sie definiert ist.

subset/all: Es werden alle verfügbare Informationen geliefert bzw. für alle Ereignisse werden Traps gesendet.

Der Notifikationsbereich kann niemals größer als der Informationsbereich sein.

Zum Cluster werden die folgenden allgemeinen Parameter angezeigt:

- Clustername
- Beschreibungstext
- Clustertyp
- Clusterversion
- Clusterhersteller
- Status des Clustermonitoring

Als Clustername wird der Name des XCS-Verbundes angezeigt, wenn das System zu einem solchen Verbund gehört.

Der Status des Clustermonitoring zeigt an, ob die Hauptprozedur von HIPLEX AF gestartet ist.

Die Hauptprozedur ist Voraussetzung für die Teilnahme an einem HIPLEX AF-Verbund. Nur unter dieser Voraussetzung können Informationen zu Umschalteinheiten und Objekten geliefert werden.

Systemtabelle

Die Systemtabelle liefert eine Übersicht über alle Systeme, zu denen eine CCS- oder XCS-Verbindung besteht.

Angezeigt werden:

– Name des Systems
– Beschreibungstext „CCS-Partner“ oder „XCS-Partner“
– Betriebsstatus
– installiertes Betriebssystem
– Version des Betriebssystems

Die Grundinformation dieser Tabelle entspricht der MSCF-Konfiguration. Da in einem HIPLEX AF-Verbund BS2000/OSD-Systeme mit UNIX-Systemen als Co-Systeme miteinander gekoppelt werden können, werden auch UNIX-Systeme angezeigt. Die Informationen zu diesen Systemen werden angezeigt, soweit sie verfügbar sind.

Anwendungstabelle

Die Anwendungstabelle enthält Informationen zu den Umschalteinheiten.

Angezeigt werden:

– Name der Umschalteinheit
– Beschreibungstext
– aktueller Status
– aktuelles Worksystem

Umschalteinheiten sind in HIPLEX AF jeweils auf Pubsets definiert. Der angezeigte Name wird deshalb aus der Catalog-Id und dem eigentlichen Namen der Umschalteinheit gebildet.

Der Beschreibungstext ist der Description-Text aus der BEGIN-SWITCH-UNIT-DEFINITION-Anweisung in der Definition der Umschalteinheit.

Der angezeigte Status hat folgende Bedeutung:

online: Die Umschalteinheit hat auf einem System den Status „work“.

offline: Die Umschalteinheit hat auf keinem System den Status „work“, aber auf einem System den Status „stand-by“.

down: Die Umschalteinheit ist deaktiviert auf allen Systemen, auf denen sie definiert ist.

faulted: Die Umschalteinheit wurde auf einem System fehlerhaft beendet und ist auf allen weiteren Systemen deaktiviert.

Das Worksystem kann nur angegeben werden, wenn der Status „online“ ist.

Ressourcentabelle

Die Ressourcentabelle enthält eine Aufstellung aller Objekte zu allen Umschalteinheiten, deren Umschaltprozedur auf dem lokalen System gestartet ist.

Zu jedem Objekt werden die folgenden Informationen geliefert:

– Objektname
– Beschreibungstext
– Objektstatus
– Typbeschreibung

Der Beschreibungstext ist der Description-Text aus den //ADD...-Anweisungen (siehe Handbuch „HIPLEX AF Hochverfügbarkeit von Anwendungen in BS2000/OSD“), mit der die Objekte in der Definition der Umschalteinheit spezifiziert werden.

Die Typbeschreibung enthält einen der folgenden Werte:

– BS2000 Application
– UNIX Application
– BS2000 Action
– UNIX Action
– Device
– Virtual Host
– Action on System Crash
– System Action

Bei HIPLEX AF wird nur der Status der BS2000/OSD- und UNIX- Anwendungen überwacht. Alle anderen Objekte haben deshalb den Status „not-monitored“.

Tabelle zur Anzeige des Anwendungsstatus auf den verbundenen Systemen

Diese Tabelle enthält für jede Umschalteinheit zu jedem System, auf dem sie definiert ist, einen Eintrag mit der Statusanzeige.

Tabelle zur Anzeige des Ressourcenstatus auf den verbundenen Systemen

Diese Tabelle wird vom HIPLEX-Subagenten nicht unterstützt. In einem HIPLEX AF-Cluster ist der Ressourcen- oder Objektstatus nur auf dem Worksystem einer Anwendung definiert.

Notifications

Die folgenden Änderungen im Cluster kann der HIPLEX-Subagent durch einen Trap an die Management-Plattform melden:

– Clusterinformation ist verfügbar: Das Subsystem MSCF ist gestartet.

– Clusterinformation ist nicht verfügbar: Das Subsystem MSCF wurde gestoppt.

– Der Status des Clustermonitoring hat sich geändert: Die Hauptprozedur von HIPLEX AF wurde gestartet, gestoppt oder hat sich aufgrund eines Fehlers beendet.

– Der Status eines Systems hat sich geändert: Die Verbindung zu einem System wurde (wieder) hergestellt oder ist verlorengegangen.

– Der Status einer Anwendung hat sich auf einem System geändert: Der Status einer Umschalteinheit hat sich auf einem System geändert.

– Der Status einer Ressource hat sich geändert: Eine BS2000/OSD- oder UNIX-Anwendung wurde gestartet, gestoppt oder hat sich aufgrund eines Fehlers beendet.

5.1.2 Konfiguration des HIPLEX-Subagenten

Der HIPLEX-Subagent benötigt keine Konfiguration. Alle wichtigen Ablaufparameter können Sie mit dem Start-Kommando einstellen (siehe nachfolgender Abschnitt).

5.1.3 Start / Stopp des HIPLEX-Subagenten

Starten des HIPLEX-Subagenten im BS2000/OSD:

/START-SNMP-HIPLEX
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54> , **CPU-LIMIT**=***STD** / <integer 1 .. 32767> / ***NO** , **JOB-CLASS**=***STD** / <name 1 .. 8> , **TIMER-INTERVAL**= 5 / <integer 1 .. 32767> , **HIPLEX-AF-LIBRARY**=***STD** / <full-filename 1 .. 54 without-generation-version> , **INFORMATION-SCOPE**=***ALL** / ***LOCAL** / ***NO** , **NOTIFICATION-SCOPE**=***ALL** / ***LOCAL** / ***NO**

oder in der POSIX-Shell mit:

```
hiplexagt [-t int] [-l <HIPLEX-AF library>] [-i info-scope] [-n notification-scope]
```

`info-scope` und `notification-scope` können jeweils die Werte „no“, „all“ oder „local“ annehmen.

Stoppen des HIPLEX-Subagenten im BS2000/OSD:

/STOP-SNMP-HIPLEX
VERSION=***STD** / <product-version>

oder in der POSIX-Shell mit:

```
hiplexcmd T
```

Beschreibung der Operanden:

VERSION=*STD / <product-version>
definiert die zu startende bzw. stoppende Version des Agenten. Diese Angabe wird derzeit nicht ausgewertet.

MONJV=*NONE / <filename 1..54>
Name der Jobvariable, die den Agenten überwachen soll. Die Angabe *NONE - keine Überwachung durch eine Jobvariable - ist Standard.

CPU-LIMIT=*STD / <integer 1 .. 32767> / *NO
Angabe der maximalen CPU-Laufzeit in Sekunden. Bei Angabe von *STD wird der generierte Standardwert verwendet.

JOB-CLASS=*STD / <name 1..8>
Jobklasse, mit der der Agent gestartet wird. Bei Angabe von *STD wird die generierte Standard-Jobklasse verwendet.

TIMER-INTERVAL=5 / <integer 1 .. 32767>
Zeitintervall, in dem der Agent überprüft, ob Anforderungen vom Kommandoprogramm vorliegen. Das Zeitintervall ist standardmäßig auf fünf Sekunden eingestellt.

HIPLEX-AF-LIBRARY=*STD / <full-filename 1..54 without-gen-vers>
Pfadname der SYSLIB von HIPLEX AF. Bei Angabe von *STD wird der Name über IMON ermittelt.

INFORMATION-SCOPE=*ALL / *LOCAL / *NO
Umfang der Information, die vom Subagenten geliefert wird.
Die Standardangabe *ALL liefert alle verfügbare Information. *LOCAL bewirkt, dass nur Informationen angezeigt werden, die das lokale System betreffen. *NO liefert keine systemspezifische Information.

NOTIFICATION-SCOPE=*ALL / *LOCAL / *NO
Umfang der Traps, die vom Subagenten gesendet werden.
Bei der Standardangabe *ALL werden für alle Ereignisse Traps geschickt. Bei *LOCAL werden nur Traps geschickt, die das lokale System betreffen, und bei *NO werden keine Traps mit systemspezifischen Informationen geschickt.

5.2 Subagent für *open*UTM - erweiterte Funktionalität in SNMP V5.0B

Die Funktionalität des Subagenten für *open*UTM wurde in folgenden Punkten erweitert:

- Mehrere *open*UTM-Anwendungen können überwacht werden.
- Die überwachten Anwendungen können angezeigt werden.
- Bei Zustandsänderungen können Traps versendet werden.

Auf Grund des erweiterten Funktionsumfangs ergeben sich Änderungen gegenüber SNMP V5.0 bei:

– Konfiguration des *open*UTM-Subagenten

– Start des *open*UTM-Subagenten

– *open*UTM-MIB

5.2.1 Konfiguration des *open*UTM-Subagenten

Für die Überwachung mehrerer *open*UTM-Anwendungen wird eine Konfigurationsdatei benötigt, in der jede überwachte *open*UTM-Anwendung spezifiziert werden muss.

Die Einträge in der Konfigurationsdatei werden mit der SDF-Anweisung //ADD-APPLICATION-RECORD erzeugt. Mit der Anweisung //REMARK können Kommentare in der Konfigurationsdatei hinterlegt werden. Die Datei muss mit der Anweisung //END abgeschlossen werden.

ADD-APPLICATION-RECORD

Die Anweisung //ADD-APPLICATION-RECORD benennt die *open*UTM-Anwendungen, die überwacht werden sollen.

//ADD-APPLICATION-RECORD
APPLICATION-NAME = <structured-name 1..8>
, **FILEBASE**= ***SAME** / <full-filename_1..54>
, **USER-ID**= **TSOS** / <name_1 .. 8>
, **TRAP-CONDITION** = list-poss (3): ***ABNORMAL-TERMINATED / *NORMAL-TERMINATED / *RUNNING**

APPLICATION-NAME=<structured-name 1.. 8>
bestimmt die *open*UTM-Anwendung, die der Subagent überwachen soll.

FILEBASE= *SAME / <full-filename_1..54>
Basisname der A-Teile der KDCFILE.
*SAME ist Standard.

USER-ID=TSOS / <name 1 .. 8>
Kennung, unter der die *open*UTM-Anwendung gestartet wird.
TSOS ist Standard.

TRAP-CONDITION= list-poss (3): *ABNORMAL-TERMINATED / *NORMAL-TERMINATED / *RUNNING
Zustände, bei denen ein Trap erzeugt werden soll.
*ABNORMAL-TERMINATED ist Standard.

5.2.2 Start / Stopp des *open*UTM-Subagenten

Starten des *open*UTM-Subagenten im BS2000/OSD:

/START-SNMP-UTM
VERSION=***STD** / <product-version> , **MONJV**=***NONE** / <filename 1 .. 54 without-gen-vers> , **CPU-LIMIT**=***STD** / <integer 1 .. 32767> / ***NO** , **JOB-CLASS**=***STD** / <name 1 .. 8> , **FILE-NAME**=***NONE** / <filename 1 .. 54 without-gen-vers> **,TIMER-INTERVAL** = 5 / <integer 1 .. 32767>

oder in der POSIX-Shell mit:

```
utmagt [-t <int>] [-f filename]
```

Stoppen des *open*UTM-Subagenten im BS2000/OSD:

/STOP-SNMP-UTM
VERSION=***STD** / <product-version>

oder in der POSIX-Shell mit:

```
utmcmd T
```

Beschreibung der Operanden:

VERSION=*STD / <product-version>
definiert die zu startende bzw. stoppende Version des Agenten. Diese Angabe wird derzeit nicht ausgewertet.

MONJV=*NONE / <filename 1..54 without-gen-vers>
Name der Jobvariable, die den Agenten überwachen soll. Die Angabe *NONE - keine Überwachung durch eine Jobvariable - ist Standard.

CPU-LIMIT=*STD / <integer 1 .. 32767> / *NO
Angabe der maximalen CPU-Laufzeit in Sekunden. Bei Angabe von *STD wird der generierte Standardwert verwendet.

JOB-CLASS=*STD / <name 1..8>
Jobklasse, mit der der Agent gestartet wird. Bei Angabe von *STD wird die generierte Standard-Jobklasse verwendet.

FILE-NAME=*NONE / <filename 1 .. 54 without-gen-vers>
Beim Start des Agenten kann eine Konfigurationsdatei angegeben werden (siehe Seite 88). Die Konfigurationsdatei muss im BS2000/OSD-Dateisystem gespeichert sein.
Standard: Es wird keine Konfigurationsdatei verwendet.

TIMER-INTERVAL=5 / <integer 1 .. 32767>
Zeitintervall, in dem der Agent überprüft, ob Anforderungen vom Kommandoprogramm vorliegen. Das Zeitintervall ist standardmäßig auf fünf Sekunden eingestellt.

5.2.3 Erweiterungen der *open*UTM-MIB

Die *open*UTM-MIB wurde erweitert um

– Anzeige überwachter Anwendungen
– Globale Daten
– Trap

Anzeige überwachter Anwendungen

Bereits zu SNMP V5.0 enthält die *open*UTM-MIB eine Tabelle der *open*UTM-Anwendungen. Diese Objekte werden zu SNMP V5.0B auch im BS2000/OSD unterstützt.

Es werden alle in der Konfigurationsdatei eingetragenen Anwendungen mit Namen und Zustand angezeigt. Die Objekte *utmApplSharedMemKey*, *utmApplSharedMemSegSize* und *utmApplSemId* werden nicht mehr unterstützt. *utmApplHomeDir* wird weiterhin nur in Reliant UNIX unterstützt.

Globale Daten

Es gibt eine neue Objektgruppe *utmGlobalData* mit folgendem Inhalt:

- utmGlobalSubagentVersion:
  Versionsnummer des SNMP-Subagenten und Art des Betriebssystems
- utmGlobalConfFile
  Name der Konfigurationsdatei. Dieses Objekt ist als „read-write“ definiert, so dass ein Wechsel der Konfigurationsdatei im laufenden Betrieb möglich ist.

Trap

Der neue Trap hat folgende Struktur

1. Enterprise: 1.3.6.1.4.1.231.2.19.20.2

2. Trapnummer: abhängig vom Zustandsübergang der *open*UTM-Anwendung:

1: abnormal beendet
2: normal beendet
3: gestartet

3. Variablen-Bindung: utmApplication 1.3.6.1.4.1.231.2.19.20.1.1 (OCTET STRING)
Name der betroffenen *open*UTM-Anwendung

6 SNMP Management

Zum SNMP Management für BS2000/OSD stehen Ihnen drei Alternativen zur Verfügung:

- Der **Web-Zugriff** auf das BS2000/OSD-Management ermöglicht es Ihnen, sich an jedem Ort, gleichgültig, ob zentral in einem Rechenzentrum oder dezentral, zuhause oder unterwegs, umfassend und jederzeit aktuell über den Zustand Ihrer BS2000/OSD-Systeme zu informieren. Zusätzlich können Sie auch steuernd eingreifen. Dabei sind alle Zugriffe sowohl über SNMP wie über Web durch strenge Sicherheitsmechanismen geschützt.
- Zu einer Reihe von Subagenten gibt es **Management-Anwendungen**, die speziell auf die Funktionalität des korrespondierenden Subagenten zugeschnitten sind. Damit können Sie sich die vom Subagenten gelieferten Werte z.B. tabellarisch oder auch in grafisch ansprechender Form anzeigen lassen. Die Management-Anwendungen sind standalone ablauffähig, können aber auch in eine Management-Plattform integriert werden.
- Bei der **Integration in die Management-Plattform Unicenter** stehen Ihnen die Funktionen des Unicenter Netz- und Systemmanagements in vollem Umfang auch für BS2000/OSD-Systeme zur Verfügung. Das SNMP Management des BS2000/OSD lässt sich in wenigen Minuten in Unicenter 3.0 oder eine der Vorgängerversionen integrieren. In einem geführten Dialog können Sie wahlweise die voreingestellten Parameter des Installationspakets SMBS2 nach Ihren Wünschen verändern oder die Standardeinstellungen des Pakets übernehmen. Zeitaufwändiges und viel Detailwissen erforderndes Einbringen und Konfigurieren „von Hand“ ist nicht notwendig.

Nachfolgend sind die einzelnen Zugänge zum SNMP Management für BS2000/OSD näher beschrieben.

6.1 Web-Zugriff auf das BS2000/OSD-Management

Neben der Verarbeitung von SNMP-Requests ermöglicht der SNMP-Masteragent auch den Zugriff auf die Managementinformationen über das World Wide Web (WWW). Somit lassen sich die von Subagenten bereitgestellten Informationen sowohl über eine SNMP Management-Plattform als auch mit einem Web-Browser abfragen und ändern.

6.1.1 Zwei verschiedene Arten von Requests

Der Masteragent hört das Netz auf zwei verschiedene Arten von Requests ab:

– Auf dem SNMP-Port (normalerweise UDP 161) erwartet der Masteragent die SNMP-SetRequests- und SNMP-GetRequests.

  Als Antwort auf die SNMP-Requests sendet der Masteragent SNMP-GetResponse-Nachrichten.

– Auf dem Web-basierten Management-Port (normalerweise TCP 280) erwartet der Masteragent HTTP-Verbindungsanforderungen.

  Als Antwort auf eine HTTP-Nachricht sendet der Masteragent eine HTML-Seite an den Browser zurück. Diese HTML-Seite kann eine vordefinierte, benutzerspezifische Web-Seite (Custom-Page) oder eine automatisch generierte Web-Seite (Subtree-Page) sein.
  Der Teil des Masteragenten, der für die Verarbeitung von HTTP-Nachrichten zuständig ist, heißt HTTP-Engine.

HTTP-Requests werden in gleicher Weise verarbeitet wie SNMP-Requests. Nach Auswertung eines SNMP- oder HTTP-Requests legt der Masteragent die relevanten Bestandteile des Requests in einer internen Warteschlange ab und beschafft sich die Information von den Subagenten auf dem üblichen Weg. Sobald der Masteragent vom Subagenten die Informationen erhalten hat, generiert er, je nach Typ des Requests, eine SNMP-GetResponse-Nachricht oder eine HTML-Seite und sendet diese mit den gewünschten Informationen an den Sender der ursprünglichen Nachricht zurück. Für den Subagenten besteht kein Unterschied zwischen SNMP-Requests oder Requests aus dem Web.

Bild 6 zeigt den Zusammenhang zwischen der SNMP- und der Web-Schnittstelle des BS2000/OSD-Agenten.

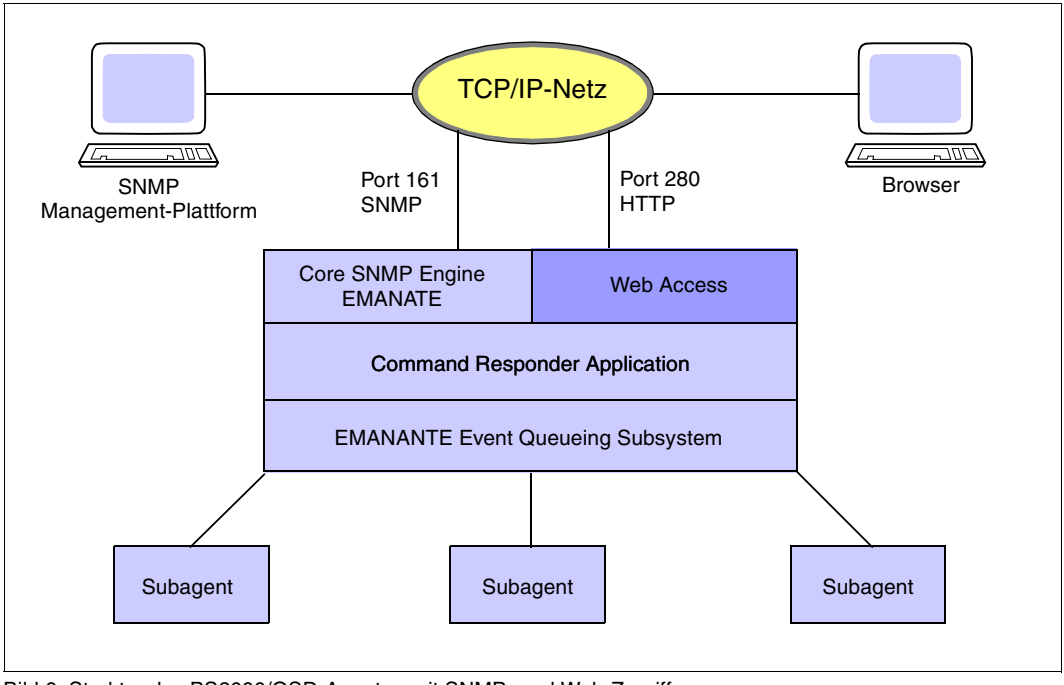

Bild 6: Struktur des BS2000/OSD-Agenten mit SNMP- und Web-Zugriff

6.1.2 Verbindung zum BS2000/OSD-Web-Agenten aufbauen

Für den Verbindungsaufbau zum BS2000/OSD-Web-Agenten (DR-Web-Entity) geben Sie an Ihrem Web-Browser Netzadresse und Portnummer wie folgt ein:

http://*netzadresse*:*portnummer*

Beispielsweise ist http://D016ZE07:280 die Adresse des Web-Agenten auf dem System D016ZE07.

Benutzername und Kennwort eingeben

Nach dem Verbindungsaufbau werden Sie aufgefordert, am Browser Benutzername und Kennwort einzugeben. Eine Dialogbox wird angezeigt (siehe folgendes Bild).

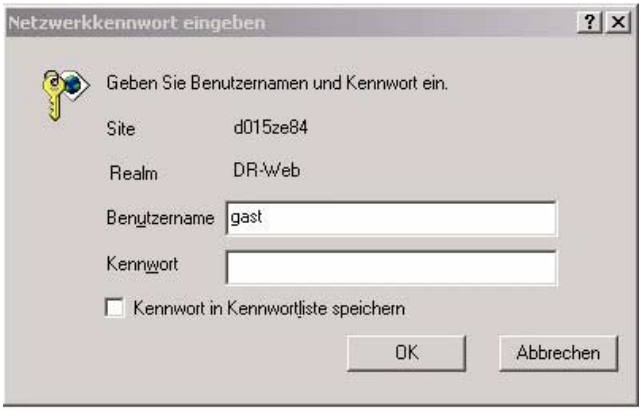

Bild 7: Eingabe von Benutzernamen und Kennwort

Benutzername und Kennwort müssen am Agenten konfiguriert sein. Die Konfiguration ist bei Auslieferung so voreingestellt, dass als gültige Eingabe der Benutzername „gast“ und als Kennwort der leere String akzeptiert wird.

- Tragen Sie bitte im Feld für die Benutzerkennung „gast“ ein und lassen Sie das Kennwortfeld leer.
- Drücken Sie den *OK*-Button.

Nach erfolgreicher Anmeldung präsentiert der Web-Agent am Browser den Begrüßungsbildschirm.

Begrüßungsbildschirm des BS2000/OSD-Web-Agenten

Bild 8 zeigt den Standard-Begrüßungsbildschirm mit Hyperlinks auf Subtree- und Custom-Page-Zweig sowie auf den Trap-Empfang.

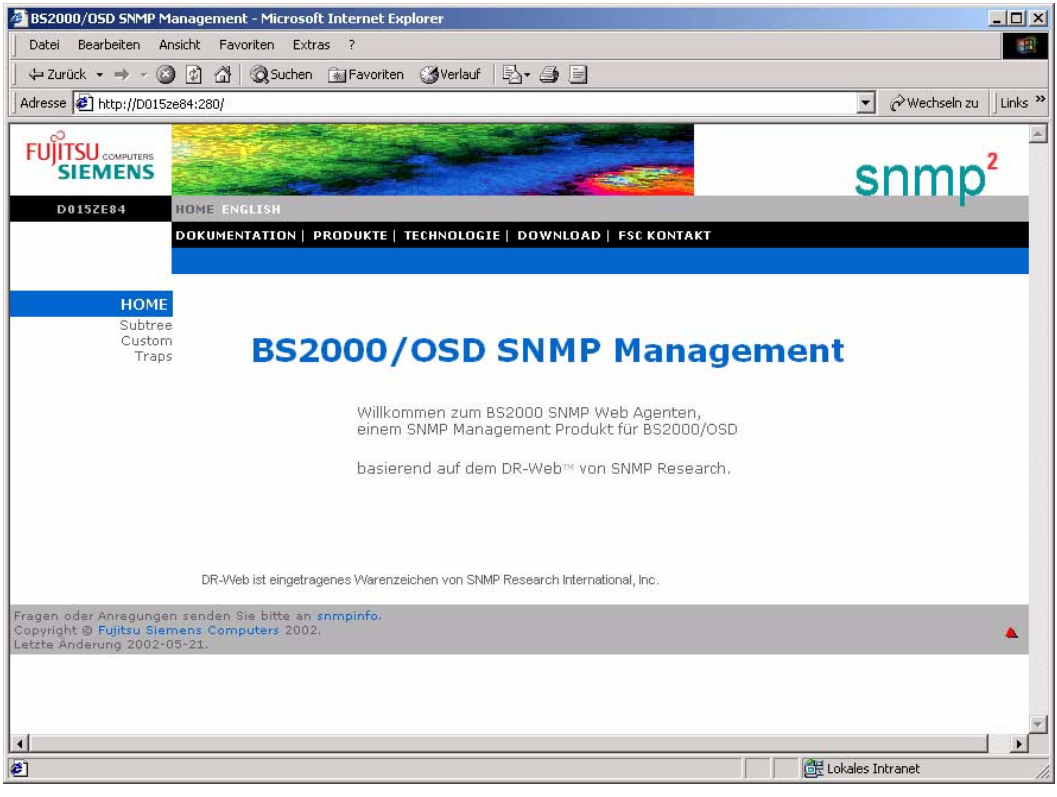

Bild 8: Begrüßungsbildschirm des Web-Agenten

6.1.3 Subtree-Funktionalität, Custom-Pages und Trap-Anzeige

Im Begrüßungsbildschirm des BS2000/OSD-Web-Agenten können Sie unter drei Links auswählen:

- *Subtree* führt Sie zur Anzeige der Standardseiten, die der Masteragent aus den MIB-Definitionen generiert (Subtree-Funktionalität).
- *Custom* führt zu den benutzerspezifischen Seiten, die mithilfe des HTML-Subagenten und der HTML-MIB definiert sind (Custom-Page-Funktionalität).
- *Traps* öffnet eine Seite für die Trap-Anzeige. Dafür wird ein Plug-in heruntergeladen mit einem Java-Applet, das den Trap-Empfang ermöglicht.

Subtree-Funktionalität

Die Subtree-Page enthält Hyperlinks auf Subtree-URLs, mit denen Sie sich die über den Web-Agenten zugänglichen Management-Informationen anzeigen lassen können. Diese Seite ist so vorkonfiguriert, dass ein schneller Zugang zu allen vom SNMP Management in BS2000/OSD unterstützten MIBs gewährleistet ist. Näheres hierzu finden Sie im Handbuch „SNMP Management V5.0“.

Custom-Page-Funktionalität

Wenn Sie auf Ihrem System den HTML-Subagenten einsetzen, steht Ihnen mit einem Klick auf den Hyperlink *Custom* des DR-Web-Begrüßungsbildschirms die Custom-Page-Funktionalität zur Verfügung. Damit können Sie die vorkonfigurierten Web-Seiten nutzen oder eigene Web-Seiten (Custom-Pages) erstellen, die neben allen Bestandteilen (Text, Grafik etc.) zusätzlich Makros für den Zugriff auf einzelne MIB-Objekte enthalten. Darüber hinaus können Sie Informationen nach individuellen Gesichtspunkten gruppieren. Näheres zur Konfigurationen von Custom-Pages sowie zum HTML-Subagenten und zur HTML-MIB finden Sie im Handbuch „SNMP Management V5.0“.

Trap-Anzeige im Web-Browser

Über die DR-Web-Schnittstelle können Sie eine Web-Seite aufrufen, die eingehende Traps in einer Tabelle anzeigt. Diese Tabelle ist als Java-Applet realisiert. Aufgrund der Zugriffsbeschränkungen, die für die Sicherheit von Java-Applets gefordert werden, können nur Traps empfangen werden, die von dem System stammen, von dem aus die Web-Seite geladen wurde. Näheres hierzu finden Sie im Handbuch „SNMP Management V5.0“.

6.2 Management-Anwendungen

Dieser Abschnitt beschreibt die Management-Anwendungen

- BCAM-Manager (BMBS2),
- Console und Application Monitor (CMBS2),
- Performance Monitor (PMBS2) und
- Cluster Monitor (CluMon).

Außerdem wird die Installation des Interpreters Tcl-Set V6.0 beschrieben, die Voraussetzung ist für die Nutzung der genannten Management-Anwendungen.

6.2.1 Installation des Interpreters Tcl-Set

Den Interpreter Tcl-Set (tclset bzw. SMAWtcl) installieren Sie wie folgt:

- In Windows verwenden Sie den Setup-Aufruf:

  ```
  tclset6A00.exe
  ```

- In Solaris verwenden Sie den Aufruf:

  ```
  pkgadd -d <pathname>/SMAWtcl-6.0.stream
  ```

- In Linux verwenden Sie den Aufruf:

  ```
  rpm -i snmptcl-6.0.rpm
  ```

6.2.2 BCAM-Manager

Der BCAM-Manager (BMBS2) ist eine SNMP Management-Anwendung, die für die Arbeit mit folgenden MIBs optimiert ist:

- MIB-II
- private BCAM-MIB

Installation des BCAM-Managers

Den BCAM-Manager (BMBS2 bzw. SMAWbmbs2) installieren Sie wie folgt:

- In Windows verwenden Sie den Setup-Aufruf:

  ```
  bmbs250A00.exe
  ```

- In Solaris verwenden Sie den Aufruf:

  ```
  pkgadd -d <pathname>/SMAWbmbs2-5.0.stream
  ```

- In Linux verwenden Sie den Aufruf:

  ```
  rpm -i snmp-5.0.rpm
  ```

i Die Installation des BCAM-Managers setzt einen Interpreter Tcl-Set ≥ V 5.0 voraus.

Funktionalität des BCAM-Managers

Der BCAM-Manager bietet folgende Funktionalität:

- Überwachung mehrerer Systeme
- Unterstützung beliebiger MIBs
- Definition von Kurzbezeichnungen für jede MIB-Variable
- Definition von Standardvariablen für jede MIB-Gruppe oder -Tabelle
- Suche nach Tabelleninstanzen anhand beliebiger Kriterien
- Suche nach Tabelleninstanzen über Kriterien aus anderen Tabellen für die BCAM-MIB
- Definition und Speicherung von Suchanfragen
- Definition von Grafikfunktionen mit Balken- und Liniendiagrammen
- Automatische Suche nach neuen Tabelleninstanzen

Ausführliche Informationen zum BCAM-Manager finden Sie im Handbuch „SNMP-Management für *open*Net Server und *inter*Net Services“.

6.2.3 Console und Application Monitor

Der Console und Application Monitor BS2000 (CMBS2) ist eine SNMP Management-Anwendung für die Überwachung der BS2000/OSD-Konsole sowie von Applikationen.

Der Console und Application Monitor arbeitet mit zwei speziellen Subagenten zusammen:

- Console Monitor Subagent in BS2000/OSD
- Application Monitor Subagent in BS2000/OSD bzw. AppMon-SX-Subagent auf Reliant UNIX

Für die Kommunikation zwischen der CMBS2-Anwendung und dem zugehörigen Subagenten wird das SNMP-Protokoll verwendet.

Ausführliche Informationen zur Bedienung des Console und Application Monitor finden Sie in der Online-Hilfe zu dieser SNMP Management-Anwendung.

Bild 9 zeigt das Trap-Fenster des Console und Application Monitor. Das Trap-Fenster ist zugleich das Hauptfenster von CMBS2. Von ihm ausgehend können Fenster für weitere Aktionen geöffnet werden.

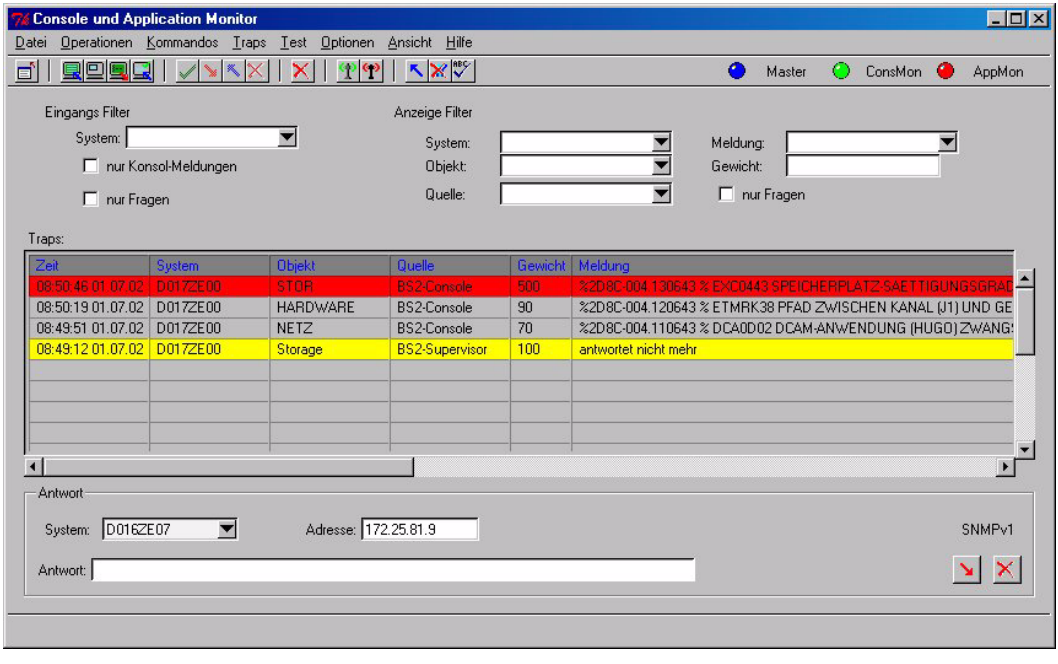

Bild 9: Hauptfenster des Console und Application Monitor (CMBS2)

Voraussetzungen

- Für den Betrieb des Console und Application Monitors sind Voraussetzung:
  - Auf BS2000/OSD-Seite: ein gestarteter SNMP Masteragent sowie der Console Monitor Subagent oder der Application Monitor Subagent
  - Auf Reliant UNIX-Seite: AppMon-SX-Subagent
- Für den Mechanismus der Trap-Bestätigung (siehe Seite 103) wird zusätzlich der gestartete Supervisor Subagent im BS2000/OSD benötigt.

Installation des Console und Application Monitor

Den Console und Application Monitor (CMBS2 bzw. SMAWcmbs2) installieren Sie wie folgt:

► In Windows verwenden Sie den Setup-Aufruf:

```
cmbs26A00.exe
```

► In Solaris verwenden Sie den Aufruf:

```
pkgadd -d <pathname>/SMAWcmbs2-6.0.stream
```

► In Linux verwenden Sie den Aufruf:

```
rpm -i snmpcmbs2-6.0.rpm
```

Überwachung der Konsole

Der Console Monitor Subagent erfasst die Konsol-Meldungen und sendet sie einzeln als Trap an den Console und Application Monitor. Die Grundfunktionalität des Console und Application Monitors besteht in der Anzeige dieser Meldungen. Über die Anzeige hinaus bietet der Console und Application Monitor folgende Möglichkeiten:

– Meldungen filtern
– bestimmte Meldungen optisch hervorheben und protokollieren
– Konsol-Fragen komfortabel beantworten

Konsol-Kommandos

Mithilfe des Console und Application Monitors und des Console Monitor Subagenten in BS2000/OSD können Sie Konsol-Kommandos ausführen. Falls Sie auf einem Agentensystem den AppMon-SX Subagenten einsetzen, können Sie dort UNIX Shell Kommandos absetzen. Es ist ferner möglich, Kommandos vorzudefinieren.

Überwachung von Applikationen

Der Application Monitor Subagent im BS2000/OSD überwacht Benutzer- und BCAM-Anwendungen (mithilfe von Monitor-Jobvariablen), DCAM-Anwendungen, Subsysteme, Jobvariablen und Logging-Dateien.

Mithilfe des AppMon-SX Subagenten können Sie Protokolldateien in Reliant UNIX kontrollieren. Jede Veränderung kann der Application Monitor Subagent durch einen Trap an den Console und Application Monitor melden.

Auch hier bieten sich Ihnen folgende Möglichkeiten:

– Meldungen filtern
– bestimmte Meldungen optisch hervorheben und protokollieren
– Konsolfragen komfortabel beantworten

Unabhängig davon können Sie von der Management-Plattform aus den Status der überwachten Subsysteme und Benutzeranwendungen durch Polls an den Application Monitor Subagenten abfragen.

Automatische Reaktionen

Als Antwort auf die Problemmeldungen, die der Application Monitor-Subagent via Trap meldet, können Sie automatische Reaktionen definieren. Hierfür geben Sie Kriterien an, wann durch einen Trap eine Reaktion ausgelöst werden soll, und definieren die Reaktion. Folgende Möglichkeiten stehen u.a. zur Auswahl:

– BS2000-Kommando senden
– lokales Kommando ausführen
– Mail schicken
– Eintrag in eine Protokolldatei schreiben
– beliebiges Tcl-Script ausführen
– Trap senden

Trap-Bestätigung

Die asynchrone Meldung von Problemen via Traps eines Agenten ist äußerst performant, da die Netzbelastung auf ein Minimum reduziert wird. Problematisch ist jedoch, dass die Information verloren geht, wenn zum Zeitpunkt des Trap-Sendens keine Management-Plattform eingeschaltet oder wenn die Kommunikation zu ihr fehlerhaft ist.

Damit eine Trap-Information unter den geschilderten Umständen dennoch nicht verloren geht, können Sie einen Trap als „zu bestätigend“ deklarieren. In diesem Fall gibt der Subagent dem Trap eine interne Information mit. Der Console und Application Monitor kann diese Information erkennen und sendet daraufhin automatisch einen Set-Request an den Agenten. Erhält der Subagent den Request, dann gilt für ihn der Trap als bestätigt. Nicht bestätigte Traps speichert der Subagent bis zum Eintreffen einer Bestätigung.

6.2.4 Performance Monitor

Der Performance Monitor BS2000 (PMBS2) ist eine SNMP Management-Anwendung für die Performance-Überwachung im BS2000/OSD. Die PMBS2-Anwendung ist speziell auf die Funktionalität des Performance Monitor Subagenten im BS2000/OSD zugeschnitten. Für die Kommunikation zwischen der PMBS2-Anwendung und dem Performance Monitor Subagenten im BS2000/OSD wird das SNMP-Protokoll verwendet. Einen kurzen Überblick über das SNMP-Protokoll und das SNMP Management-Konzept finden Sie im Abschnitt „SNMP Management-Architektur“ auf Seite 8.

Ausführliche Informationen zur Bedienung des Performance Monitor finden Sie in der Online-Hilfe zu dieser SNMP Management-Anwendung.

Vorraussetzungen

Voraussetzung für den Betrieb der Performance Monitor-Anwendung ist auf BS2000/OSD-Seite ein gestarteter SNMP-Masteragent, der Performance Monitor-Subagent sowie ein gestartetes SM2-Subsystem.

Installation des Performance Monitor

Den Performance Monitor (PMBS2 bzw. SMAWpmbs2) installieren Sie wie folgt:

- In Windows verwenden Sie den Setup-Aufruf:

  ```
  pmbs26A00.exe
  ```

- In Solaris verwenden Sie den Aufruf:

  ```
  pkgadd -d <pathname>/SMAWpmbs2-6.0.stream
  ```

- In Linux verwenden Sie den Aufruf:

  ```
  rpm -i snmppmbs2-6.0.rpm
  ```

Tabellarische und grafische Darstellung

Die Grundfunktionalität der Performance-Monitor-Anwendung besteht in der tabellarischen und insbesondere in der grafischen Darstellung der SM2-Messwerte, die durch den Performance Monitor Subagenten auf BS2000/OSD-Seite bereitgestellt werden. Die aktuellen Messwerte eines oder auch mehrerer BS2000/OSD-Systeme werden dabei in Form von Formularen und Tabellen oder grafisch als Balken- und Kurvendiagramme dargestellt. Dabei können Sie beliebige Zeitintervalle für die automatische Aktualisierung der Anzeigen festlegen.

Die darzustellenden Objekte und Werte sind dabei in der SNMP Performance MIB exakt definiert. Auf diese MIB greifen sowohl die PMBS2-Anwendung auf der Management-Plattform, als auch der Performance Monitor Subagent im BS2000/OSD zu. Der Zugriff auf die MIB-Objekte durch den Performance Monitor erfolgt dabei derzeit nur lesend. Eine detaillierte Auflistung der MIB-Objekte der Performance MIB finden Sie im Handbuch „SNMP Management V5.0“.

Bild 10 zeigt die grafische Darstellung der CPU-Auslastung durch den Performance Monitor.

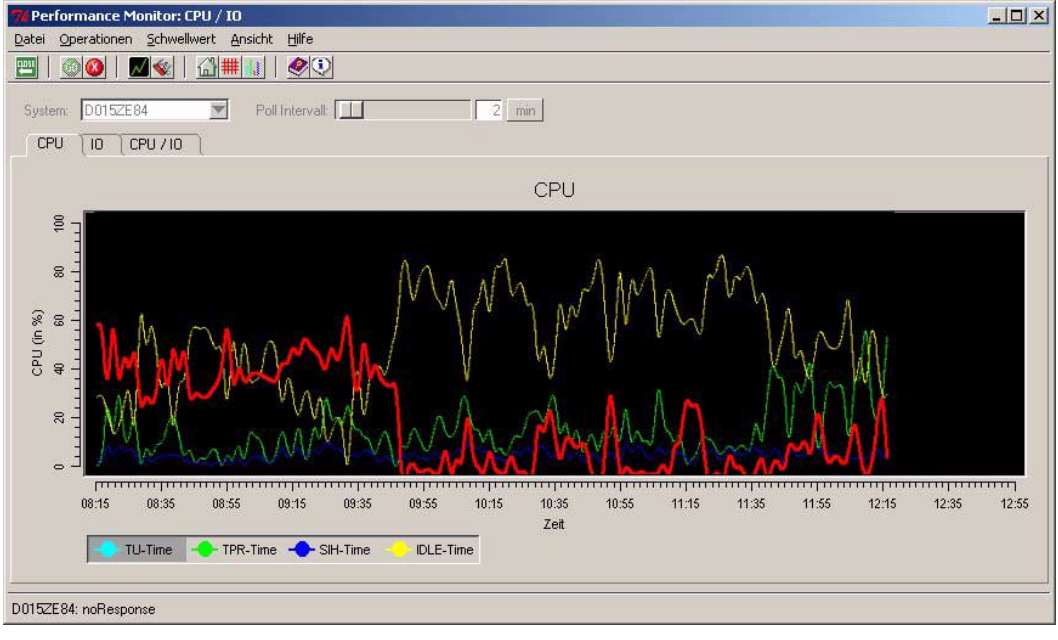

Bild 10: Anzeige der CPU-Auslastung durch den Performance Monitor (PMBS2)

Schwellenwerte und Reaktionen

Zusätzlich zur Grundfunktionalität für die Darstellung aktueller SM2-Messwerte ermöglicht es der Performance Monitor, kritische Schwellenwerte zu definieren und zugeordnete Reaktionen auf Über- oder Unterschreitung dieser Schwellenwerte festzulegen. Dabei können Sie bei der Art der Reaktion frei zwischen mehreren Möglichkeiten wählen.

Mögliche Reaktion sind z.B.

- Protokolldatei-Einträge,
- akustische Signale,
- Ausführung bestimmter Kommandos,
- Erzeugung von SNMP-Traps.

Traps als asynchrone Ereignismeldungen können an weitere SNMP Management-Anwendungen gesendet werden, so dass sich die Performance Monitor-Anwendung nahtlos in das SNMP Management einfügt.

6.2.5 Cluster Monitor

Der Cluster Monitor (CluMon) ist eine SNMP Management-Anwendung zur Überwachung von Hochverfügbarkeits-Clustern im BS2000/OSD und auf Reliant UNIX. Er arbeitet mit dem HIPLEX-Subagenten im BS2000/OSD zusammen. Für die Kommunikation zwischen der CluMon-Anwendung und dem HIPLEX-Subagenten im BS2000/OSD wird das SNMP-Protokoll verwendet.

Ausführliche Informationen zur Bedienung des Cluster Monitor finden Sie in der Online-Hilfe zu dieser SNMP Management-Anwendung.

Voraussetzungen

Voraussetzung für den Betrieb des Cluster Monitors ist ein gestarteter SNMP-Masteragent, der HIPLEX-Subagent im BS2000/OSD oder der entsprechende Subagent auf Reliant UNIX.

Installation des Cluster Monitor

Den Cluster Monitor (CluMon bzw. SMAWclumn) installieren Sie wie folgt:

- In Windows verwenden Sie den Setup-Aufruf:

  ```
  clumon1A00.exe
  ```

- In Solaris verwenden Sie den Aufruf:

  ```
  pkgadd -d <pathname>/SMAWclumn-1.0.stream
  ```

- In Linux verwenden Sie den Aufruf:

  ```
  rpm -i snmpclumn-1.0.rpm
  ```

Cluster-Überwachung

HIPLEX-Subagenten befinden sich auf den Systemen des Clusters. Diese Agenten liefern Informationen über die aktuelle Clusterkonfiguration aus der Sicht dieses Teilsystems. Die Cluster Monitor-Anwendung wertet die Informationen aller im Cluster arbeitenden HIPLEX-Subagenten aus und präsentiert eine Gesamtsicht des Clusterzustands (siehe Bild 11). Dies ist auch dann noch möglich, wenn ein System im Clusterverbund ausfällt. Die grafische Darstellung gestattet ein schnelles Erfassen der Konfiguration und aufgetretener Probleme im Clusterverbund.

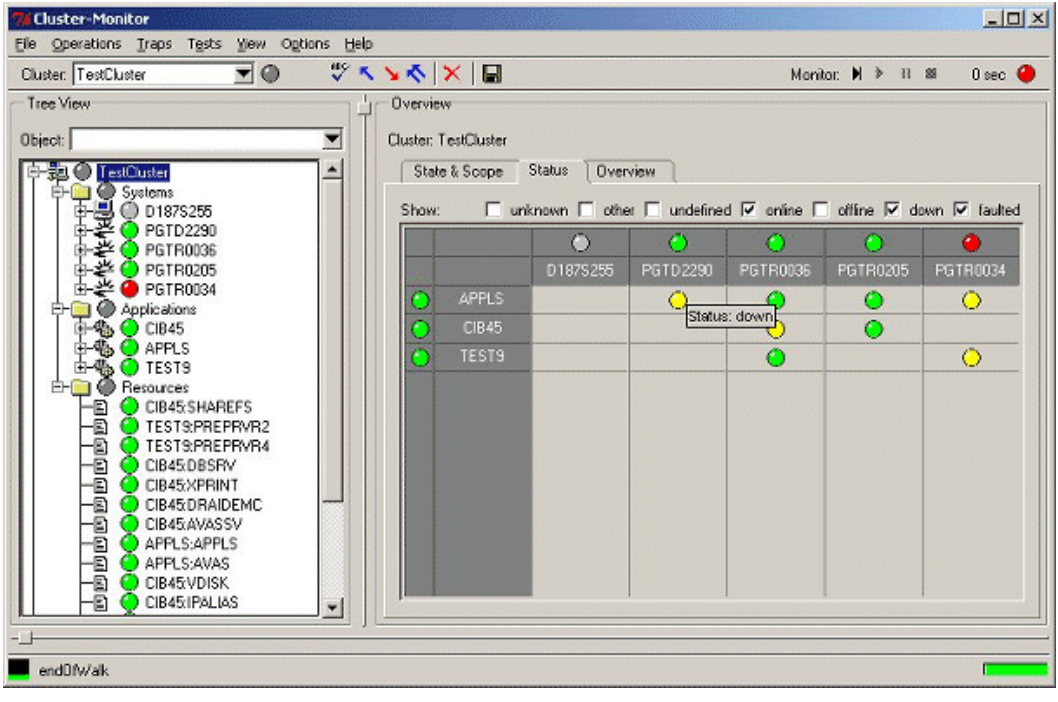

Bild 11: Gesamtsicht des Cluster-Zustands

Der Cluster-Monitor ermöglicht verschiedene Sichten:

– Sicht auf den Cluster
– Sicht auf die Systeme
– Sicht auf die Applikationen
– Sicht auf die Ressourcen

Sicht auf den Cluster

Die Sicht auf den Cluster bietet Ihnen

– eine tabellarische Auflistung der von den Agenten der einzelnen Systeme zur Verfügung gestellten Informationen,

– eine Matrixdarstellung der Zustände der Applikationen auf den einzelnen Systemen,

– eine komprimierte Übersicht über die Status der am Cluster beteiligten Systeme und Applikationen.

Sicht auf die Systeme

Die Sicht auf die Systeme bietet Ihnen

– eine Darstellung der Clusterkonfiguration als Verbund von Systemen,

– eine tabellarische Übersicht über die Daten der am Cluster beteiligten Systeme wie Name, Status, Betriebssystem etc.

Sicht auf die Applikationen und Ressourcen

Ähnlich wie bei den Systemen erhalten Sie hier eine Übersicht über die am Verbund beteiligten Applikationen und Ressourcen mit den zugehörigen Daten und Status.

Monitoring

Für die automatische Aktualisierung der Anzeige können Sie ein Monitoring via Poll einschalten. Die Pollfrequenz ist dabei frei wählbar.

Traps

Wichtige Ereignisse und Statusänderungen im Cluster werden von den HIPLEX-Subagenten als Traps an die Management-Plattform gemeldet. Die asynchrone Meldung von Problemen via Traps des Agenten ist äußerst performant, da dies die Netzbelastung auf ein Minimum reduziert. Die Traps werden im Cluster Monitor angezeigt. Darüber hinaus aktualisiert der Cluster Monitor mithilfe der Trap-Informationen die Zustandsdarstellung des Clusters.

6.3 Integration in Unicenter

Die Integrationspakete SMBS2 (für Windows NT bzw. Windows2000) und SMAWsmbs2 (für Solaris) enthalten ergänzende Teile zur Einbindung des Systemmanagements für BS2000/OSD in die Management-Plattform Unicenter von Computer Associates.

6.3.1 Voraussetzungen für die Integration

Für die Integration der BS2000/OSD-Systeme in Unicenter müssen die folgenden Voraussetzungen erfüllt sein:

- Auf der Managementseite sind erforderlich:
  - Unicenter 3.0 (WindowsNT V4.0, Windows2000) oder

    Unicenter TNG 2.2. 2.4 oder 2.4.1 (WindowsNT V4.0, Windows2000 oder Solaris ab V2.6) oder
  - Integrationspaket SMBS2 V5.0B für das SNMP Management des BS2000/OSD.
- In BS2000 ab OSD V2.0 sind erforderlich:
  - SNMP-Basic-Agent BS2000 SBA-BS2 V6.0

  sowie optional:
  - SNMP-Standard-Collection BS2000 SSC-BS2 V6.0
  - SNMP-Subagent SM2 SSA-SM2-BS2 V5.0B
  - SNMP-Subagent für *open*UTM (BS2000) SSA-OUTM-BS2 V5.0B

6.3.2 Installation der Integrationspakete

SMBS2 bzw. SMAWsmbs2 installieren Sie wie folgt:

- SMBS2 installieren Sie in Windows NT bzw. Windows 2000 mit dem Setup-Aufruf:

  ```
  smbs2_setup.exe
  ```

- SMAWsmbs2 installieren Sie in Solaris mit dem Aufruf:

  ```
  pkgadd -d <Pfadname>/SMBS2-SO.stream.5.0B00
  ```

Damit werden automatisch die zusätzlich benötigten Dateien in das Installationsverzeichnis von Unicenter übertragen und die erforderlichen Anpassungen an die Konfiguration durchgeführt.

Zu Beginn der Installation spezifizieren Sie in einem geführten Dialog wichtige Installationsparameter. Für alle Parameter gibt es sinnvolle Standardeinstellungen, die Sie übernehmen können.

6.3.3 Integration in die Unicenter-Komponenten

SMBS2 und SMAWsmbs2 enthalten mehrere Elemente, die den „World View“, das „Enterprise Management“, die „Agent Technology“ und den Unicenter Explorer ergänzen und deren Konfiguration erweitern.

World View

Für das Repository wird eine neue Hostklasse mit dem Namen „SiemensBS2000“ sowie eine Anzahl von Agentenklassen eingeführt. Diese neuen Klassen besitzen einige Erweiterungen gegenüber ihren Oberklassen „Host“ und „Agent“. Sie sind mit speziellen Ikonen für das 2D- und 3D-Netzbild und den Unicenter Explorer verknüpft.

Das Standard-Popup-Menü für Host-Objekte ist für BS2000/OSD-Objekte um einige Funktionen erweitert. BS2000/OSD-Objekte können sowohl manuell als auch mit der automatischen Discovery in das Repository und das Netzbild eingefügt werden. Für die Objektansicht (Object View) werden die BS2000/OSD-MIBs bereitgestellt.

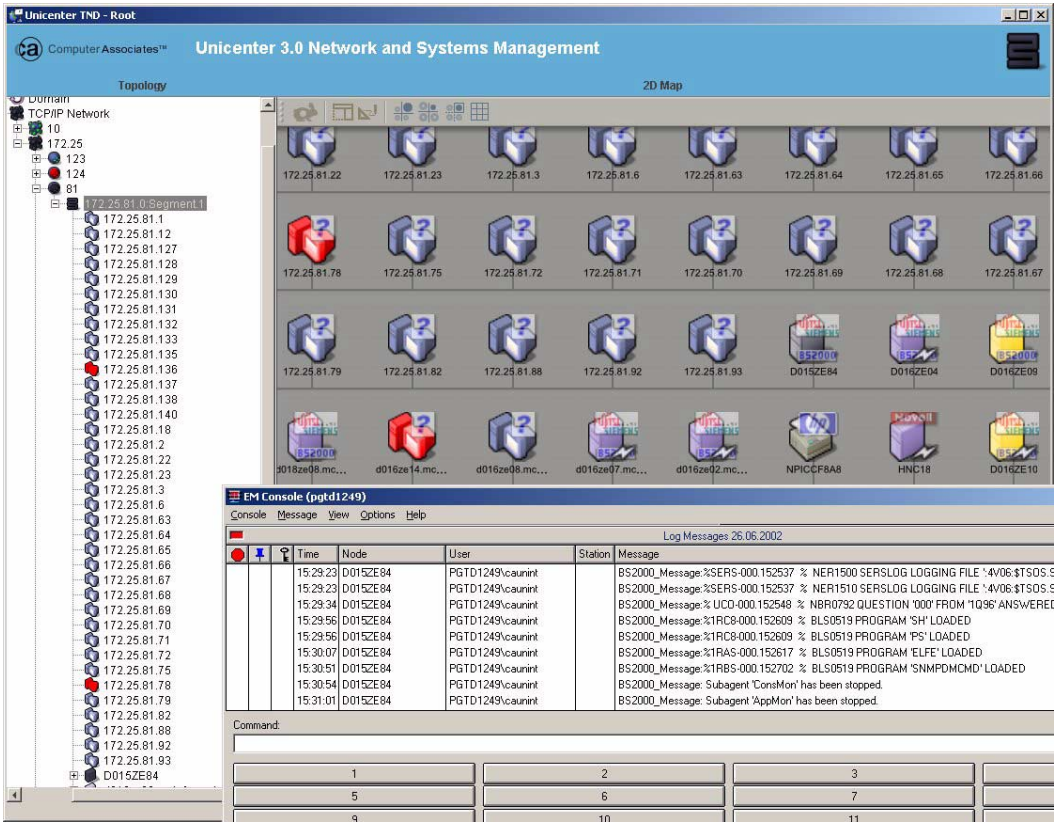

Bild 12: BS2000/OSD-Systeme im 2D-Netzbild des Unicenter Explorer und Event Console mit BS2000-Meldungen

Enterprise Management

Für die Überwachung von BS2000/OSD-Systemen an der Ereigniskonsole enthält SMBS2 Meldungsformate zu allen Traps des BS2000/OSD-SNMP-Agenten. Insbesondere können alle BS2000/OSD-Konsolmeldungen, die der Console Monitor Subagent als Traps weitersendet, an der Unicenter-Ereigniskonsole angezeigt werden. Die BS2000/OSD-Meldungen sind mit besonderen Attributen versehen, so dass Sie die Meldungen ohne größeren Aufwand mit Reaktionen verknüpfen können.

Agententechnologie (Agent Technology)

Analog zum „World View“ wird die Hostklasse „Siemens-BS2000“ definiert, und dieser Hostklasse werden zehn Agentenklassen mit DSM-Policies zugeordnet. Dabei handelt es sich um zwei im Lieferumfang von Unicenter enthaltene Agentenklassen und acht neu eingerichtete Klassen. Zu jeder der neuen Agentenklassen gehören eine Agenten- und eine Knotenansicht.

Die Knotenansicht (Node View) zu einem BS2000/OSD kann folgende Objekte enthalten:

- Ping
- Mib2
- Application Monitor
- AVAS
- HSMS
- Omnis
- RDBMS
- Storage
- Supervisor
- UTM

Ping

Diese Policy realisiert eine allgemeine Überwachung des Systems durch regelmäßige Pings. Wenn sich das Antwortverhalten des Systems ändert, wird der Objektstatus geändert.

Mib2

Es gibt Unterobjekte zu allen Interfaces in der Interfacetabelle der MIB-II, die den Status der Interfaces anzeigen.

Application Monitor

Es werden vier Unterobjekte zu Subsystemen sowie zu BCAM-, Benutzer- und DCAM-Anwendungen erzeugt. Diese Objekte haben Unterobjekte zu den einzelnen überwachten Subsystemen, BCAM-, Benutzer- und DCAM-Anwendungen. Diese Unterobjekte zeigen den Status der überwachten Subsysteme sowie BCAM-, Benutzer- und DCAM-Anwendungen an.

AVAS

Es gibt ein Objekt, das den Gesamtstatus von AVAS anzeigt.

HSMS

Es gibt ein Objekt, das die Verfügbarkeit des HSMS-Subagenten anzeigt.

OMNIS

Zu allen überwachten OMNIS-Systemen werden Unterobjekte erzeugt, und zu jedem OMNIS gibt es jeweils ein Unterobjekt zu einer Trap-Klasse. Angezeigt wird der allgemeine Status für alle überwachten OMNIS-Systeme. Die Unterobjekte zu den Trap-Klassen ändern ihren Status, sobald ein Trap dieser Klasse empfangen wird.

RDBMS

Es gibt Unterobjekte zu allen Datenbank-Servern. Zu jedem Datenbank-Server werden Datenbank-Unterobjekte erzeugt. Mit dem Status dieser Datenbank-Unterobjekte wird die Verfügbarkeit der Datenbank-Server für die Datenbanken angezeigt.

Storage

Es werden zwei Unterobjekte zu Pubsets und Privatplatten erzeugt. Zu diesen Unterobjekten gehören jeweils Unterobjekte zu den überwachten Pubsets und den überwachten Privatplatten:

– Der Status der Pubset-Objekte zeigt den erreichten Saturation-Level des Pubsets an.

– Der Status der Privatplatten-Objekte zeigt die Verfügbarkeit der Privatplatten an.

Supervisor

Es gibt Unterobjekte zu allen Subagenten in der Subagententabelle, die den Status der Subagenten anzeigen.

openUTM

Es gibt Unterobjekte zu allen überwachten *open*UTM-Anwendungen, die den Status der *open*UTM-Anwendung anzeigen.

Auf der nächsten Seite finden Sie ein Beispiel für die NodeView-Anzeige in Unicenter.

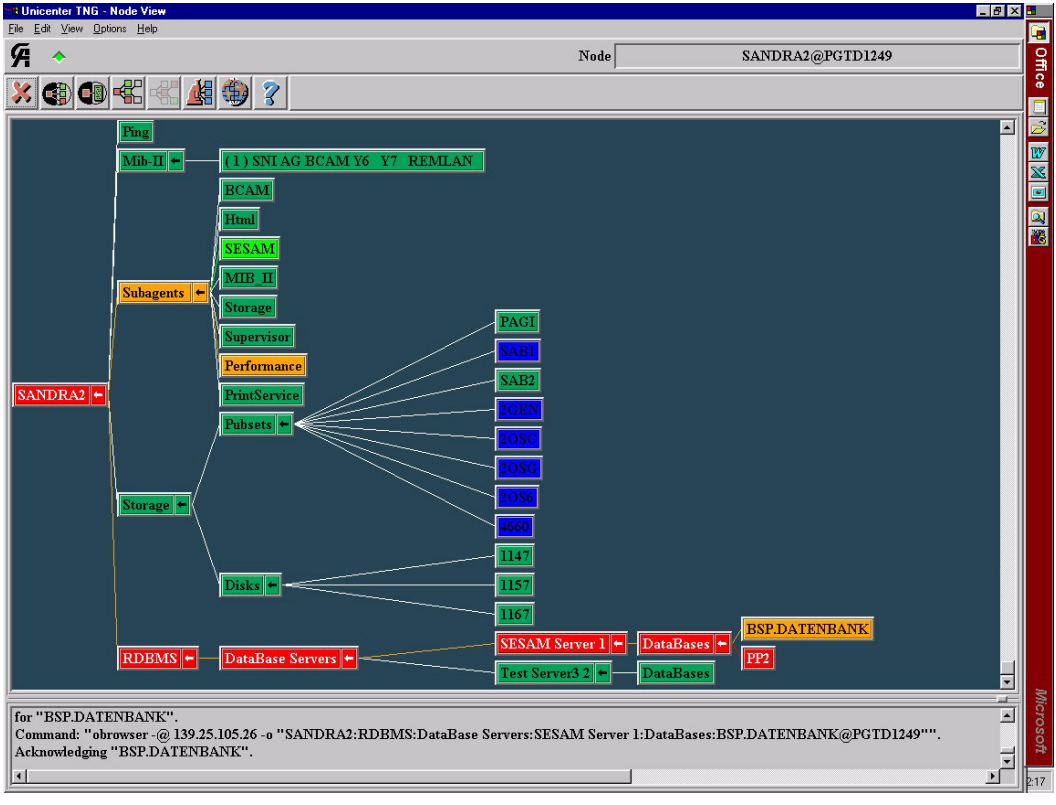

Bild 13: NodeView-Anzeige in Unicenter

7 Sicherheitsbewusste Nutzung von SNMP

Dieses Kapitel gibt Hinweise und Empfehlungen zur sicheren Nutzung des SNMP-basierten BS2000/OSD-Managements. Es ist jedoch keine Anleitung zur Sicherheitsanalyse oder zur Aufstellung eines Sicherheitsregelwerks. Diese beiden wichtigen Themen gehen über den Rahmen des vorliegenden Handbuchs hinaus.

Die funktionellen Details zur sicherheitsbewussten Einstellung der Konfigurationsparameter des SNMP-Agenten in BS2000/OSD finden Sie im Handbuch „SNMP Management V5.0“, Abschnitt „3.3.1 Security-Konfiguration“.

Details zu den korrespondierenden Einstellungen an der Management-Plattform sind im Abschnitt „Integration in Unicenter“ (siehe Seite 110) des vorliegenden Handbuchs beschrieben.

7.1 Sicherheit als Prozess

Sicherheit ist nicht nur ein Produkt oder eine Lösung, sondern auch ein Prozess, vergleichbar den Prozessen für das Qualitätsmanagement. Nur ein permanentes Sicherheitsmanagement kann dauerhafte Sicherheit gewährleisten.

Gegen versehentliche oder willkürliche äußere und innere Bedrohungen verfolgt der Prozess „Sicherheit für schützenswerte Objekte“ folgende Zielsetzungen:

– Vertraulichkeit
– Integrität
– Verfügbarkeit
– verantwortliche Nutzung

Dabei lehnt sich der Prozess eng an die Vorgaben des Sicherheitsregelwerks an.

Ein umfassendes Sicherheitsregelwerk enthält folgende Prozessschritte:

- vorbeugende Maßnahmen
- Feststellung
- Reaktion

Vorbeugende Maßnahmen

Zu den vorbeugenden Maßnahmen gehört die korrekte Konfigurierung der Sicherheitsparameter entsprechend den Sicherheitsregeln. Vorbeugende Maßnahmen allein gewährleisten jedoch noch keine Sicherheit, da die Sicherheitsvorkehrungen nie perfekt sein können.

Feststellung

Die Feststellung umfasst die Protokollierung des operativen Betriebs und die regelmäßige Prüfung der Protokolle auf sicherheitsrelevante Zwischenfälle, wie z.B. wiederholte Verletzungen des Zugriffsschutzes (Eindringversuche).

Reaktion

Die Reaktion überprüft

– die Sicherheitsregeln auf Angemessenheit und Wirksamkeit,
– eine eventuelle Aufwertung der vorbeugenden Maßnahmen sowie
– eine eventuelle Verschärfung der Mechanismen zur Feststellung.

7.2 Empfehlungen zur allgemeinen Netz- und Systemsicherheit

Mit dem SNMP-Protokoll kommunizieren Sie über das Internet. Damit setzen Sie die beteiligten Systeme den möglichen Angriffen und Gefahren des öffentlichen Internet aus.

Empfehlung

Stellen Sie die zu verwaltenden BS2000/OSD-Systeme und die Management-Plattform in ein ausschließlich von Ihnen selbst kontrolliertes Teilnetz, zum Beispiel hinter eine Firewall. Auf diese Weise können Sie den Zugang zu Ihren Systemen einfacher und zentraler kontrollieren und mögliche Angriffe abwehren.

Die im SNMP-Agenten implementierten Sicherheitsfunktionen beruhen darauf, dass seine Konfigurations- und Programmdateien nur für privilegierte Nutzer - meist ist dies ein Administrator - zugreifbar sind. Die Installation richtet diese Dateien mit den korrekten Rechten ein.

Empfehlung

Vergewissern Sie sich in regelmäßigen Abständen, dass die Konfigurations- und Programmdateien des SNMP-Agenten in BS2000/OSD nur für privilegierte Nutzer zugreifbar sind.

7.3 Empfehlungen für die sichere Nutzung des SNMP-Service

SNMP ist bei naiver Nutzung unsicher. Die Standardkonfiguration, die bei minimaler Konfiguration nach der Installation wirksam wird, ist ein Kompromiss zwischen Sicherheit und umfassender Interoperabilität im SNMP-Verbund, der zu Gunsten der Interoperabilität ausfällt.

Vermeiden Sie daher den operativen Betrieb mit der Minimalkonfiguration des BS2000/OSD-Systems. Wenn Sie die Konfiguration am verwalteten BS2000/OSD-System ändern, passen Sie bitte auch die korrespondierenden Einstellungen an der Management-Plattform an.

Empfehlung

Ändern Sie die Minimalkonfiguration des SNMP-Agenten sowie die korrespondierende Einstellungen an der Management-Plattform entsprechend den Vorgaben Ihres Sicherheitsregelwerks.

Bei folgenden Konfigurationsparametern sollten Sie unter Sicherheitsaspekten besonders aufmerksam verfahren:

- Community Strings für den Empfang von SNMP-Anforderungen
- Community Strings und Zugriffskontrolle auf MIB-Objekte
- Community Strings und Absenderadressen
- Empfängeradressen für SNMP-Traps
- Versenden von Authentication Failure Traps aktivieren

7.3.1 Community Strings für den Empfang von SNMP-Requests

Unter einer Community versteht man in der SNMP-Terminologie eine Gruppe, bestehend aus einer oder mehreren Management-Plattformen und mehreren von diesen Plattformen betreuten SNMP-Agenten.

Jede Community wird durch einen sog. Community String identifiziert. Der Community String ist unverschlüsselter Bestandteil jedes SNMP-Requests und weist den Absender des Requests als Mitglied der betreffenden Community aus. Die Berechtigung für einen lesenden oder schreibenden Request, den eine Management-Plattform an einen SNMP-Agenten sendet, wird über diesen Community String geregelt.

Mit dem Community String steht ein einfacher Authentisierungsmechanismus in SNMP zur Verfügung. So dürfen Management-Plattform und SNMP-Agent nur dann miteinander kommunizieren, wenn Sie derselben Community angehören, d.h. der SNMP-Agent akzeptiert SNMP-Requests nur von Management-Plattformen, deren Community Strings ihm bekannt, also vorkonfiguriert, sind.

Da der Community String unverschlüsselt mit der SNMP-Nachricht versendet wird, besteht immer die Gefahr seiner unberechtigten Verwendung. Dies kann für eine sicherheitsbewusste Nutzung von SNMP problematisch sein. Andererseits verwenden die meisten Communities ohnehin den voreingestellten Community String "public".

Empfehlung

Wählen Sie geeignete Communities entsprechend Ihrer System- und Betriebsorganisation und ordnen Sie passende Community Strings zu. Ändern Sie den Community String entsprechend den Vorgaben Ihres Regelwerkes, ähnlich wie Sie es z.B. von Passwörtern kennen. Beachten Sie, dass Sie den Community String in allen beteiligten Systemen der Community ändern müssen.

Empfehlung

Wenn es Ihr Umfeld aus SNMP-Agenten und Management-Plattform(en) erlaubt, sollten Sie die benutzerspezifische Authentisierung des SNMPv3-Protokolls verwenden.

7.3.2 Community Strings und Kontrolle des Zugriffs auf MIB-Objekte

Community Strings können mit Zugriffrechten „read-only“, „read-write“ usw. versehen werden. Entsprechend den Zugriffrechten dürfen SNMP-Requests, die diesen Community String enthalten, die „read-only“ definierten Objekte lesen und die „read-write“ definierten Objekte lesen und ändern. Ohne weitere Vorkehrungen können dann *alle* zugreifbaren Objekte gelesen oder *alle* änderbaren Objekte gelesen und/oder geändert werden.

Empfehlung

Nutzen Sie die Möglichkeit, an bestimmte Community-Strings selektive Lese- bzw. Schreibrechte zu vergeben für

- MIB-Zweige,
- Objekte,
- Objektinstanzen (in Tabellen).

Empfehlung

Wenn es Ihr Umfeld aus SNMP-Agenten und Management-Plattform(en) erlaubt, sollten Sie die benutzerspezifische Autorisierung des SNMPv3-Protokolls verwenden.

7.3.3 Community Strings und Absenderadressen

Die IP-Adressen der autorisierten Management-Plattformen können Sie explizit vorkonfigurieren. Damit veranlassen Sie den SNMP-Agenten, SNMP-Anforderungen nur von diesen Systemen zu akzeptieren.

Die Management-Plattformen müssen dabei konstante IP-Adressen besitzen. Eine dynamische Zuweisung via Dynamic Host Configuration Protocol (DHCP) ist nicht möglich.

Empfehlung

Nutzen Sie die Prüfung der Absenderadressen der Management-Plattformen, indem Sie deren IP-Adressen im SNMP-Agenten konfigurieren.

7.3.4 Empfängeradressen für SNMP-Traps

Die IP-Adressen der autorisierten Management-Plattformen, an die ein SNMP-Agent SNMP-Traps senden soll, können Sie explizit vordefinieren. Die Management-Plattformen müssen dabei konstante IP-Adressen besitzen. Eine dynamische Zuweisung via Dynamic Host Configuration Protocol (DHCP) ist nicht möglich.

Empfehlung

Definieren Sie die Empfängeradressen der Management-Plattformen vor, die SNMP-Traps empfangen sollen. Konfigurieren Sie hierfür die IP-Adressen dieser Management-Plattformen im SNMP-Agenten.

7.3.5 Community String für SNMP-Traps

Sie können den Community String konfigurieren, den ein SNMP-Agent als Teil eines SNMP-Traps an die Management-Plattform sendet. Die Management-Plattform wird dann nur SNMP-Traps mit diesem Community String akzeptieren.

Empfehlung

Wählen Sie geeignete Communities entsprechend Ihrer System- und Betriebsorganisation. Ändern Sie den Community String entsprechend den Vorgaben Ihres Regelwerkes, ähnlich wie Sie es z.B. bei Passwörtern kennen. Beachten Sie, dass Sie den Community String in allen beteiligten Systemen der Community ändern müssen.

7.3.6 Authentication Failure Traps aktivieren

Der SNMP-Agent prüft jeden SNMP-Request entsprechend der konfigurierten Sicherheitsparameter und -optionen (siehe vorhergehende Abschnitte). Falls der SNMP-Request die Prüfungen passiert, bearbeitet ihn der Agent. Andernfalls verwirft der Agent den SNMP-Request und sendet einen Authentication Failure Trap.

Per Voreinstellung ist das Versenden von Authentication Failure Traps deaktiviert. Bei aktiviertem Versenden schickt der SNMP-Agent diesen Trap an alle konfigurierten Empfänger.

Empfehlung

Konfigurieren Sie den Parameter `snmpEnableAuthenTraps` so, dass das Versenden von Authentication Failure Traps aktiviert ist.

Konfigurieren Sie einen geeigneten Empfänger für solche Traps.

Prüfen Sie in regelmäßigen Abständen, ob diese Traps aufgetreten sind und analysieren Sie alle derartigen Fälle.

Literatur

BS2000/OSD-Handbücher

SNMP Management V5.0
SNMP Management für BS2000/OSD
Benutzerhandbuch

Zielgruppe
Das Handbuch wendet sich an Netzverwalter, -operateure und Systemverwalter, die BS2000-Systeme in ein SNMP-basiertes Management integrieren bzw. ein solches System bedienen wollen.
Inhalt
Dieses Handbuch beschreibt einerseits die Einbettung von SBA-BS2, SSC-BS2, SSA-SM2-BS2 und SSA-OUTM-BS2 in BS2000/OSD und die zum Betrieb notwendigen Installations- und Konfigurationsschritte sowie den Betrieb selbst. Die zur Überwachung notwendigen Agenten und ihre MIBs werden detailliert vorgestellt. Andererseits wird die Installation und Konfiguration der entsprechenden Management-Anwendungen auf den Management-Plattformen Unicenter TNG, TransView SNMP und HP OpenView beschrieben.
Weitere zentrale Themen des Handbuchs sind der Zugriff auf Management-Informationen über das World Wide Web sowie der Trap-Server für Solaris und Reliant UNIX.

***open*Net Server V2.0** (BS2000/OSD)
BCAM V16.0A Band 1
Benutzerhandbuch

Zielgruppe
Das Handbuch richtet sich an Netzplaner, -generierer und -verwalter, die in BS2000-Systemen BCAM betreiben.
Inhalt
BCAM Band 1 beschreibt BCAM selbst, seine Einbettung in TRANSDATA und TCP/IP- und ISO-Netze, sowie Generierungs- und Administrationstätigkeiten. Generierungsbeispiele verdeutlichen die Beschreibung. Es werden BCAM-Tools zur Generierung und Diagnose beschrieben.

***open*Net Server V2.0** (BS2000/OSD)
BCAM V16.0A Band 2
Referenzhandbuch

Zielgruppe
Das Handbuch richtet sich an Netzoperateure, -generierer und -verwalter, die in BS2000-Systemen BCAM betreiben.
Inhalt
BCAM Band 2 baut auf Band 1 auf und beschreibt ausführlich die zur Generierung und zum Betrieb nötigen BCAM-Kommandos. Es werden die zur statischen Generierung nötigen KOGS-Makros vorgestellt und die BCAM-Fehlermeldungen aufgelistet.

***open*Net Server V2.0, *inter*Net Services V2.0**
SNMP-Management für *open*Net Server und *inter*Net Services
Benutzerhandbuch

Zielgruppe
Das Handbuch richtet sich an Netz- und Systemverantwortliche, die ein SNMP-basiertes Netz- und Systemmanagement nutzen möchten.
Inhalt
Das Handbuch beschreibt detailliert die mit *open*Net Server ausgelieferten MIBs, die mit *inter*Net Services ausgelieferte FTP-MIB, die Installation und den Betrieb der Subagenten. Ein eigenes Kapitel behandelt ausführlich die Bedienung des BCAM Managers.

***inter*Net Services V2.0** (BS2000/OSD)
Administratorhandbuch

Zielgruppe
Das Handbuch richtet sich an Netzplaner, -generierer und -verwalter, die in BS2000/OSD Internet Services betreiben wollen.
Inhalt
Das Handbuch beschreibt die Funktionalität der Internet Services BOOTP/DHCP, TFTP, DNS, FTP, LDAP und NTP in BS2000/OSD. Installation, Administration, Betrieb, Logging- und Diagnose-Möglichkeiten der einzelnen Komponenten sowie FTP-Exit und TELNET-Exits sind weitere Themen dieses Handbuchs.

***inter*Net Services V2.0** (BS2000/OSD)
Benutzerhandbuch

Zielgruppe
Das Handbuch richtet sich an Netzplaner, -generierer und -verwalter sowie Nutzer, die die Internet Services in Verbindung mit BS2000/OSD nutzen wollen.
Inhalt
Das Handbuch stellt die Komponenten von *inter*Net Services vor. Ausführlich werden die Nutzung von FTP, der FTAC-Schnittstelle für FTP und TELNET beschrieben. Netzverwalter benötigen dieses Handbuch zusätzlich zum Administratorhandbuch.

HIPLEX AF (BS2000/OSD)
Hochverfügbarkeit von Anwendungen in BS2000/OSD
Produkthandbuch

Zielgruppe
Dieses Handbuch wendet sich an Systemverwalter und Organisatoren in Rechenzentren.
Inhalt
Das Handbuch macht mit den Voraussetzungen für das Umschalten von Anwendungen und mit der Bedienung von HIPLEX AF vertraut. Da es das konkrete Know-how vermittelt, wie Anwendungen umschaltbar einzurichten sind (Organisation, Generierung, Prozeduranpassung), lässt es sich auch einsetzen, um Anwendungen auf ein möglichst sicheres manuelles Umschalten vorzubereiten

DSSM/SSCM
Verwaltung von Subsystemen in BS2000/OSD
Benutzerhandbuch

Zielgruppe
Das Handbuch wendet sich an die Systembetreuung und die Softwareberatung des BS2000/OSD.
Inhalt
Es werden das Subsystemkonzept des BS2000/OSD, die Dynamische Subsystemverwaltung DSSM und die Subsystemkatalog-Verwaltung SSCM mit den dazugehörigen Kommandos und Anweisungen beschrieben.

SPOOL V4.1A (BS2000/OSD)
Benutzerhandbuch

Zielgruppe
Das Handbuch wendet sich an nichtprivilegierte Anwender, den Spool & Print-Verwalter, den RSO-Geräteverwalter und die Systembetreuung.
Inhalt
Es wird der Betrieb von SPOOL beschrieben.

RSO V3.2A (BS2000/OSD)
Remote SPOOL Output
Benutzerhandbuch

Zielgruppe
Das Handbuch wendet sich an nichtprivilegierte Benutzer, RSO-Geräteverwalter, SPOOL-Verwalter und Systembetreuer des BS2000/OSD.
Inhalt
Für die einzelnen Anwendergruppen werden die Aufgaben und Möglichkeiten zur Nutzung und Steuerung von dezentralen Druckern (RSO-Drucker) beschrieben. Das Handbuch enthält die Druckermerkmale aller RSO-Drucker.

AVAS V5.0A (BS2000/OSD)
AVAS Funktionen
Benutzerhandbuch

Zielgruppe
AVAS-Benutzer
Inhalt
– Überblick über die AVAS-Funktionen
– Festlegung und Abwicklung der Produktion
– Kurzer Abriss über die Administration
Bestellnummer
U5184-J-Z125-8

AVAS V5.0A / AVAS-SV V4.1C
(BS2000/OSD, UNIX, Windows NT)
AVAS für den Administrator
Systemverwalterhandbuch

Zielgruppe
AVAS-Administratoren
Inhalt
– Alle Aufgaben des AVAS-Administrators – von der Generierung des Systems bis zur Administration des AVAS-Systems
– Dienstprogramm AVAS-QUER
– Kopplung von AVAS mit MAREN
– AVAS-Reports
– BATCH-Funktionen
– Externe Erstellung von AVAS-Elementen
– Programmschnittstelle
– AVAS-SV
Bestellnummer
U21462-J-Z125-5

AVAS (BS2000/OSD)
Auftragsverwaltungs- und Abwicklungssystem
Einführung

Zielgruppe
Alle, die das AVAS-System kennenlernen wollen.
Inhalt
Einführung in das AVAS-System. Das Handbuch zeigt den Kundennutzen von AVAS auf, stellt die wesentlichen Funktionen vor, empfielt eine Vorgehensweise bei der AVAS-Einführung und stellt Produkte im AVAS-Umfeld vor. Als Nachschlagewerk dienen die AVAS-Benutzerhandbücher.

***open*FT V8.0 für BS2000/OSD**
Enterprise File Transfer in der offenen Welt
Benutzerhandbuch

Zielgruppe
Das Handbuch richtet sich an Benutzer, die mit *open*FT Dateien übertragen oder Dateimanagement betreiben möchten.
Inhalt
Das Benutzerhandbuch stellt die Leistungen von *open*FT vor. Die Beschreibung beinhaltet auch die optionalen Komponenten *open*FT-AC für den Zugangs- und Zugriffsschutz und *open*FT-FTAM zur Unterstützung der FTAM-Funktionalität. Die Kommandoschnittstelle und Meldungen werden ausführlich dargestellt.

***open*FT V8.0 für BS2000/OSD**
Enterprise File Transfer in der offenen Welt
Installation und Administration
Systemverwalterhandbuch

Zielgruppe
Das Handbuch richtet sich an Verwalter, die auf ihren BS2000-Rechnern *open*FT, *open*FT-FTAM und *open*FT-AC betreiben möchten.
Inhalt
Das Systemverwalterhandbuch beschreibt Installation und Inbetriebnahme von *open*FT und den optionalen Komponenten *open*FT-AC und *open*FT-FTAM. Betrieb und Steuerung des *open*FT-Systems werden eingehend vorgestellt. Die Kommandoschnittstelle enthält die Beschreibung aller Administratorkommandos.

OMNIS/OMNIS-MENU (TRANSDATA, BS2000)
Administration und Programmierung
Benutzerhandbuch

Zielgruppe
– OMNIS-Administrator
– Programmierer
Inhalt
– Grundlagen der Administration von OMNIS und OMNIS-MENU
– OMNIS-Dienstprogramme
– Anwenderschnittstelle zur Erweiterung des Funktionsumfangs von OMNIS
– Meldungen

SESAM/SQL-Server (BS2000/OSD)
Datenbankbetrieb
Benutzerhandbuch

Zielgruppe
Das Handbuch wendet sich an den SESAM/SQL-Systemverwalter.
Inhalt
Das Handbuch beschreibt, welche Möglichkeiten der Systemverwalter hat, den Datenbankbetrieb zu steuern und zu überwachen.

SM2 (BS2000/OSD)
Software Monitor
Band 1: Verwaltung und Bedienung

Zielgruppe
Anwender und Systembetreuung
Inhalt
Das Messsystem SM2 liefert dem Benutzer statistische Daten über die Leistung des DV-Systems und die Auslastung der Betriebsmittel. Im Band 1 werden die Bedienung des Messmonitors SM2, die SM2-Messprogramme und die SM2-Bildschirmreports beschrieben.
Zur Auswertung und Darstellung der SM2-Messwerte siehe Band 2 des Handbuchs.

SM2 (BS2000/OSD)
Software Monitor
Band 2: SM2-Messwerte auswerten und darstellen

Zielgruppe
Anwender und Systembetreuung
Inhalt
Das Messsystem SM2 liefert dem Benutzer statistische Daten über die Leistung des DV-Systems und die Auslastung der Betriebsmittel. Im Band 2 des Handbuchs werden das Dienstprogramm SM2U1 zum Aufbereiten und Verwalten der SM2-Messwertedateien und die Auswerteprogramme SM2R1, SM2R1-PC, SM2ONLINE-PC und SM2-PA beschrieben.
Zur Verwaltung und Bedienung von SM2 siehe Band 1 des Handbuchs.

*open***UTM** (BS2000/OSD)
Anwendungen generieren und betreiben
Benutzerhandbuch

Zielgruppe
Das Handbuch richtet sich an Anwendungsplaner, Fachprogrammierer, Administratoren und Anwender von UTM-Anwendungen.
Inhalt
Das Handbuch beschreibt die Generierung von UTM-Anwendungen mit verteilter Verarbeitung, die Tools, die *open*UTM dazu zur Verfügung stellt und die UTM-Objekte, die bei der Generierung erzeugt werden. Außerdem enthält das Handbuch alle Informationen, die für die Strukturierung, den Betrieb und die Kontrolle einer UTM-Produktivanwendung benötigt werden.

Literatur zu Unicenter TNG

Wegen Literatur zu Unicenter TNG wenden Sie sich bitte an die Firma Computer Associates.

www.ca.com

Sonstige Literatur

Douglas Steedman
Abstract Syntax Notation One (ASN.1): The Tutorial and Reference
Isleworth, 1990
(ISBN 1-871802-06-7)

Marshall T. Rose
The Simple Book: An Introduction to Management of TCP/IP-based Internets
Prentice-Hall
(ISBN 0-13-812611-9)

RFCs

Umfassende Informationen zu den Request for Comments (RFCs) finden Sie auf der Home Page der Internet Engineering Task Force (IETF):

www.ietf.org

.

Stichwörter

W

Z

Inhalt

SNMP Management V6.0 (BS2000/OSD)

Benutzerhandbuch

Zielgruppe
Das Handbuch wendet sich an Netzverwalter, -operateure und Systemverwalter, die BS2000/OSD-Systeme in ein SNMP-basiertes Management integrieren bzw. ein solches System bedienen wollen.
Inhalt
Das Handbuch beschreibt die Erweiterungen des SNMP Managements V6.0 für BS2000/OSD gegenüber der Version 5.0:

– neue sowie funktional erweiterte Subagenten und Managament Anwendungen
– erweiterte Integration in die Management-Plattform CA Unicenter
– erhöhte Sicherheit

Dort, wo sich die Funktionalität der Version 6.0 des SNMP-Managements für BS2000/OSD gegenüber der Version 5.0 nicht geändert hat, ist die Beschreibung im Handbuch „SNMP Management V5.0 SNMP Management für BS2000/OSD“ weiterhin gültig.

Ausgabe: Juli 2002

Datei: snmp.pdf

Dieses Handbuch wurde erstellt von
cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

Fujitsu Siemens Computers GmbH
Handbuchredaktion
81730 München

Kritik
Anregungen
Korrekturen

Fax: 0 700 / 372 00000

e-mail: DOCetc@mchp.siemens.de
http://manuals.fujitsu-siemens.com

Absender

Kommentar zu SNMP Management V6.0
SNMP Management für BS2000/OSD

Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format *...@ts.fujitsu.com*.

The Internet pages of Fujitsu Technology Solutions are available at *http://ts.fujitsu.com/*...
and the user documentation at *http://manuals.ts.fujitsu.com*.

Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form *...@ts.fujitsu.com*.

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter *http://de.ts.fujitsu.com/*..., und unter *http://manuals.ts.fujitsu.com* finden Sie die Benutzerdokumentation.