

FUJITSU Software openFT

(BS2000, z/OS, Windows, Unix Systems)

Version 12.0C00
July 2015

Readme file

All rights reserved, including intellectual property rights.
Technical data subject to modifications and delivery subject to availability. Any liability that the data and illustrations are complete, actual or correct is excluded. Designations may be trademarks and/or copyrights of the respective manufacturer, the use of which by third parties for their own purposes may infringe the rights of such owner.

© 2015 Fujitsu Technology Solutions GmbH

1	Introduction	3
2	Deactivate restart function	4
2.1	Partner-specific deactivation of the restart	4
2.2	Global deactivation of the recovery	6
3	Extended support of the Application Entity Title on Windows and Unix Platforms	10
3.1	Application Entity Title formats	10
3.2	Sender check with the Application Entity Title	11
3.3	Setting of the Calling Application Entity Title	12
3.4	Display of the Application Entity Title for FTAM partner	13
4	Support of FarSync X.25 under Linux and Windows	14
4.1	Configuration of the FarSync X.25 transport system in openFT	14
4.1.1	Command ftmodo	14
4.1.2	openFT Explorer – Operating parameters	16
4.2	Partner address of an X.25 partner	16
4.2.1	Command ftaddptn	17
4.2.2	Command ftmodptn	20
4.2.3	Command ftshwptn – normal output	22
4.2.4	Command ftshwptn – CSV output	24
4.2.5	openFT Explorer - Partner list entry	25
4.3	Local X.25 address	26
4.3.1	Command ftmodo	27
4.3.2	Command ftshwo – normal output	31
4.3.3	Command ftshwo – CSV output	33
4.3.4	openFT Explorer – Operating parameters	35
4.4	Extension of the ftinfo command	36
5	Scope ID and line number	37
6	Single-user mode on Unix Platforms	38
6.1	Installing openFT	38
6.2	Switching to single-user mode	38
6.3	Creating and administering additional openFT instances in single-user mode	39
6.4	Returning to multi-user mode	40
6.5	Returning permission for remote administration	40
6.6	SNMP in single-user mode	41
6.7	Changes for single-user mode under Solaris	41
6.8	ftalarm in single and multi-user mode	41
7	SNMP under Linux	42
8	Release of openFT for 64-bit Linux	43
9	Installation on Unix Systems	44
10	Exporting public keys	45
11	Delete diagnosis information and console messages	46
12	Configuration of minimum RSA key length for openFT protocol	47
12.1	Command syntax for Unix / Windows / POSIX:	47
12.2	Command syntax for BS2000 / z/OS:	47
12.3	Output of the ftshwo / SHOW-FT-OPTIONS command:	48
12.4	openFT GUI support :	48
12.5	openFT z/OS panels support :	49
13	ftmonitor polling time change	50
14	Extended ftinfo output	51
15	Case extension to the tool ft_mget	52
16	Extension of logging functionality to the ‘Logging’ node	53
17	openFT Explorer recognizes newly created openFT instances automatically	54
18	Unattended installation of openFT on Windows Platforms	55

1 Introduction

This document describes the new features since openFT 12.0B on the Windows, Unix, BS2000 and z/OS systems.

2 Deactivate restart function

It is sensible in many situations to deactivate the restart function for the openFT and FTAM protocol for asynchronous file transfer jobs as well. If, for example, an openFT Initiator instance is coupled to a load balancer which selects one of the many responders depending on the load, there is no guarantee that this load balancer at restart selects the same responder instance which processed the job at the beginning. The restart would fail and a job "corpse" remains in the original responder.

2.1 Partner-specific deactivation of the restart

The partner-specific deactivation of the restart has an impact on outbound jobs. It can be switched on and off using the commands `ftaddptn` and `ftmodptn`:

```
ftaddptn <name>
    ...
    [ -rco=n | -rco=f | -rco= ]

ftmodptn <name>
    ...
    [ -rco=n | -rco=f | -rco= ]
```

The parameter `-rco` (recovery outbound) has the following meaning:

- rco=n the restart is always activated for this partner for outbound jobs.
- rco=f the restart is deactivated for this partner for outbound jobs.
- rco= the restart operability for outbound jobs depends on the setting in the operating parameters (default setting).

The parameter has no impact if the implementation of the file transfer protocol (FTP) or type of job (e.g. preprocessing, synchronous orders) does not permit a restart.

The setting of the parameter is shown with the long output and CSV output of the command `ftshwptn`:

```
ftshwptn -l FTESP1

NAME      STATE SECLEV  PRI  TRACE  LOC  REM  P-CHK  ADDRESS
          INBND REQU-P   RECOV
FTESP1    ACT   STD    NORM FTOPT   0   0      ftam://d012ze28
          ACT   STD    FTOPT
```

The following outputs are possible with **RECOV**:

- FTOPT The global setting in the operating parameters applies
- ON The recovery is activated
- OFF The recovery is deactivated

This parameter is shown in the column **OutboundRecovery** in the CSV output with the possible values `*FTOPT`, `*ON` and `*OFF`.

In **openFT Explorer** the parameter is supported by the new field **Recovery Outbound** on the dialog page **General** of the Partner list entry dialog. The field has three radio buttons **Standard** (corresponds to `-rco=`), **Activated** (corresponds to `-rco=n`) and **Deactivated**

(corresponds to -rco=f). The FT administrator can set a value here for openFT instances as of V12.0B. No change is possible for other users and older versions.

As of openFT V12.0B, this functionality was only available for Windows and UNIX systems. In openFT V12.0C, this functionality is extended to **BS2000** and **z/OS** systems for the commands, **MODIFY-FT-PARTNER / FTMODPTN** and **ADD-FT-PARTNER / FTADDPTN**.

Command syntax for ADD-FT-PARTNER in BS2000 / z/OS:

```
ADD-FT-PARTNER / FTADDPTN
...
RECOVERY-OUTBOUND= *BY-FT-OPTIONS / *ON / *OFF
```

OPERANDS

RECOVERY-OUTBOUND=

This parameter allows a partner to restart asynchronous file transfer if interrupted in between. Default setting following installation: FTOPT

RECOVERY-OUTBOUND= *BY-FT-OPTIONS

The global settings apply for the partner.

RECOVERY-OUTBOUND= *ON

Outbound Recovery functions are enabled for the specific partner. This function only works if the global openFT outbound recovery function is also activated.

RECOVERY-OUTBOUND= *OFF

Outbound Recovery functions are disabled for the specific partner.

Command syntax for MODIFY-FT- PARTNER in BS2000 / z/OS:

```
MODIFY-FT- PARTNER / FTMODPTN
...
RECOVERY-OUTBOUND = *UNCHANGED / *ON / *OFF/ *BY-FT-OPTIONS
```

OPERANDS

RECOVERY-OUTBOUND=

This parameter allows a partner to restart asynchronous file transfer if interrupted in between. Default setting following installation: FTOPT

RECOVERY-OUTBOUND=*UNCHANGED

The existing outbound Recovery functions remain unchanged.

RECOVERY-OUTBOUND=* ON

Outbound Recovery functions are enabled for the specific partner. This function only works if the global openFT outbound recovery function is also activated.

RECOVERY-OUTBOUND=* OFF

Outbound Recovery functions are disabled for the specific partner.

RECOVERY-OUTBOUND=* BY-FT-OPTIONS

The global setting for recovery function applies for the specific partner.

**Output of the SHOW-FT-PARTNER / FTSHWPTN in BS2000 / z/OS:
These are the different outputs for Outbound Recovery.**

NAME	STATE	SECLEV	PRI	TRACE	LOC	REM	P-CHK	ADDRESS	
	INBND	REQU-P		RECOV				ROUTING	IDENTIFICATION
HCLLIN1	ACT	STD	NORM	FTOPT	0	0	FTOPT	HCLLIN1	
	ACT	STD		FTOPT					HCLLIN1

NAME	STATE	SECLEV	PRI	TRACE	LOC	REM	P-CHK	ADDRESS	
	INBND	REQU-P		RECOV				ROUTING	IDENTIFICATION
HCLLIN1	ACT	STD	NORM	FTOPT	0	0	FTOPT	HCLLIN1	
	ACT	STD		ON					HCLLIN1

NAME	STATE	SECLEV	PRI	TRACE	LOC	REM	P-CHK	ADDRESS	
	INBND	REQU-P		RECOV				ROUTING	IDENTIFICATION
HCLLIN1	ACT	STD	NORM	FTOPT	0	0	FTOPT	HCLLIN1s	
	ACT	STD		OFF					HCLLIN1

CSV- OUTPUT of SHOW-FT-PARTNER / FTSHWPTN in BS2000 / z/OS:

Column	Type	Value
OutboundRecovery	String	*ON / *OFF/ *FTOPT

OPS- OUTPUT of SHOW-FT-PARTNER in BS2000:

Element	Type	Value
RECOVERY-OUTBOUND	String	*ON / *OFF / *BY-FT-OPT

2.2 Global deactivation of the recovery

The recovery can be globally deactivated separated according to outbound and inbound jobs:

```
ftmodo ...
    [ -rco=n | -rco=f ] [ -rci=n | -rci=f ]
```

The parameters -rco (recovery outbound) and -rci (recovery inbound) have the following meaning:

- rco=n The recovery for outbound jobs is activated for all partners for which there is no own partner-specific setting
- rco=f The recovery for outbound jobs is deactivated for all partners for which there is no own partner-specific setting
- rci=n The recovery for inbound jobs is activated
- rci=f The recovery for inbound jobs is deactivated

After a new installation the recovery is activated for outbound as well as inbound.

The settings of the two parameters are displayed in the output of the ftshwo command:

```
ftshwo
STARTED PROC-LIM  CONN-LIM  ADM-CLIM  RQ-LIM  MAX-RQ-LIFE  TU-SIZE  CCS-NAME
YES      NONE      16        8        2000        30      65535    ISO88591
PTN-CHK  DYN-PART  SEC-LEV   FTAC-LOG FT-LOG  ADM-LOG   USE TNS  USE CMX
```

STD	ON	B-P-ATTR	ALL	ALL	ALL	NO	NO	
OPENFT-APPL		FTAM-APPL		FTP-PORT		ADM-PORT		ADM-CS
*STD		*STD		1121		11000		NO
ACTIVE		INACT		DISABLED		INACT		
RSA-PROP	RSA-MIN	ENC-MAND						
2048	0	NO						
HOST-NAME		IDENTIFICATION / LOCAL SYSTEM NAME						
*NONE		D012ZE26.due.fxy.net / \$FJAM,D012ZE26						
DEL-LOG	ON	AT	RETPD	RECOVERY	ADM-TRAP-SERVER			
OFF	DAILY	00:00	0	IN+OUT	*NONE			
TRAP:	SS-STATE	FT-STATE	PART-STATE	PART-UNREA	RQ-STATE	TRANS-SUCC	TRANS-FAIL	
CONS	OFF	OFF	OFF	OFF	OFF	OFF	OFF	
ADM		OFF	OFF	OFF	OFF	OFF	OFF	
FUNCT:	SWITCH	PARTNER-SELECTION		REQUEST-SELECTION	OPTIONS	OPTIONS-LL		
MONITOR	OFF	NONE		ALL				
TRACE	OFF	ALL		ALL	NONE	OFF		

The following outputs are possible with **RECOVERY**:

- IN+OUT Both inbound as well as outbound recovery are activated
- IN Only inbound recovery is activated
- OUT Only outbound recovery is activated
- NO Neither outbound nor inbound recovery is activated

In the CSV output these settings (with *ON or *OFF) are shown in the columns **OutboundRecovery** and **InboundRecovery**.

In **openFT Explorer** the two parameters are supported by the new field **Recovery** on the dialog page **General** of the dialog Operating parameters. The field comprises two check buttons **Inbound activated** and **Outbound activated**.

Setting the option **Inbound activated** corresponds to -rci=n and setting **Outbound activated** corresponds to -rco=n. For both options the FT administrator can set a value for openFT instances as of V12.0B. No change is possible for other users and older versions.

As of openFT V12.0B, this functionality was only available for Windows and UNIX systems. In openFT V12.0C, this functionality is extended to **BS2000** and **z/OS** systems to the commands, **MODIFY-FT-OPTIONS / FTMODOPT**.

Command syntax for MODIFY-FT-OPTIONS in BS2000 / z/OS:

```

MODIFY-FT-OPTIONS / FTMODOPT
...
RECOVERY-OUTBOUND = *UNCHANGED / *ON / *OFF
RECOVERY-INBOUND = *UNCHANGED / *ON / *OFF
    
```

OPERANDS

RECOVERY-OUTBOUND=

This parameter allows to restart all asynchronous file transfer request if interrupted in between.

Default setting following installation: *ON

RECOVERY-OUTBOUND=*UNCHANGED

The existing outbound Recovery functions remain unchanged.

RECOVERY-OUTBOUND=*ON

Outbound Recovery functions are enabled for global settings.

Note: Outbound Recovery functions only works if inbound recovery functions is enabled in remote system.

RECOVERY-OUTBOUND=*OFF

Outbound Recovery functions are disabled for global settings.

RECOVERY-INBOUND=

This parameter allows to restart an asynchronous file transfer request coming from a remote system having outbound recovery enabled, if transfer is interrupted in between.
 Default setting following installation: *ON

RECOVERY- INBOUND =*UNCHANGED

The existing inbound Recovery functionality remains unchanged.

RECOVERY- INBOUND =*ON

Inbound Recovery functionality is globally enabled.

RECOVERY- INBOUND =*OFF

Inbound Recovery functionality is globally disabled.

Output of the SHOW-FT-OPTIONS command in BS2000:

STARTED	PROC-LIM	CONN-LIM	ADM-CLIM	RQ-LIM	MAX-RQ-LIFE	TU-SIZE	
YES	2	16	8	2000	30	65535	
PTN-CHK	DYN-PART	SEC-LEV	FTAC-LOG	FT-LOG	ADM-LOG		
STD	ON	B-P-ATTR	ALL	ALL	ALL		
OPENFT-APPL		FTAM-APPL		FTP-PORT		ADM-PORT	
*STD		*STD		21		11000	
ACTIVE		ACTIVE		DISABLED		ACTIVE	
RSA-PROP	RSA-MIN	ENC-MAND					
2048	0	NO					
HOST-NAME		IDENTIFICATION					
*NONE		HCLTEST1					
DEL-LOG	ON	AT	RETPD	RECOVERY	ADM-TRAP-SERVER		
OFF	DAILY	00:00	14	IN+OUT	*NONE		
TRAP:	SS-STATE	FT-STATE	PART-STATE	PART-UNREA	RQ-STATE	TRANS-SUCC	TRANS-FAIL
CONS	ON	OFF	OFF	OFF	OFF	OFF	OFF
SNMP	OFF	OFF	OFF	OFF	OFF	OFF	OFF
ADM		OFF	OFF	OFF	OFF	OFF	OFF
FUNCT:	SWITCH	PARTNER-SELECTION		REQUEST-SELECTION		OPTIONS	
MONITOR	OFF	ALL		ALL			
TRACE	OFF	ALL		ALL		NONE	

Output of the FTSHWOPT command in z/OS:

STARTED	PROC-LIM	CONN-LIM	ADM-CLIM	RQ-LIM	MAX-RQ-LIFE	TU-SIZE	CCS-NAME
YES	2	16	8	2000	30	65535	IBM1047
PTN-CHK	DYN-PART	SEC-LEV	FTAC-LOG	FT-LOG	ADM-LOG		
STD	ON	B-P-ATTR	ALL	ALL	ALL		
OPENFT-APPL		FTAM-APPL		FTP-PORT		ADM-PORT	
*STD		*NONE		1221		11000	
ACTIVE		NAVAIL		DISABLED		ACTIVE	
RSA-PROP	RSA-MIN	ENC-MAND					
2048	0	NO					
HOST-NAME		IDENTIFICATION / LOCAL SYSTEM NAME					
OSA2FT		FJMF TANK / \$FJAM, FJMF TANK					
DEL-LOG	ON	AT	RETPD	RECOVERY	ADM-TRAP-SERVER		
OFF	DAILY	00:00	14	IN+OUT	*NONE		
TRAP:	SS-STATE	FT-STATE	PART-STATE	PART-UNREA	RQ-STATE	TRANS-SUCC	TRANS-FAIL
CONS	OFF	OFF	OFF	OFF	OFF	OFF	OFF
ADM		OFF	OFF	OFF	OFF	OFF	OFF
FUNCT:	SWITCH	PARTNER-SELECTION		REQUEST-SELECTION		OPTIONS	

MONITOR	OFF	ALL	ALL	
TRACE	OFF	ALL	ALL	NONE

CSV- OUTPUT of SHOW-FT- OPTIONS / FTSHWOPT in BS2000 / z/OS:

Column	Type	Value
OutboundRecovery	String	*ON / *OFF
InboundRecovery	String	*ON / *OFF

OPS-OUTPUT of SHOW-FT-OPTIONS in BS2000:

Element	Type	Value
RECOVERY-OUTBOUND	String	*ON / *OFF
RECOVERY-INBOUND	String	*ON / *OFF

3 Extended support of the Application Entity Title on Windows and Unix Platforms

With openFT protocol the initiator is only determined via the instance identification on the inbound side (with normal sender check) or primary (with extended sender check). The determination of the sender with the FTAM protocol was previously only via the partner address. If the partner address of the initiator is now changed from time to time, FTAM can have problems with the partner allocation and also with the recovery.

The Application Entity Title now offers the option of implementing the identification of an initiator which is independent of the partner address for FTAM as well.

3.1 Application Entity Title formats

The Application Entity Title (AET) is divided into the Application Process Title and the optional Application Entity Qualifier. The Application Process Title is an object identifier (a sequence of non-negative integer figures); the Application Entity Qualifier is a non-negative integer number. In addition to this numeric format there is also a transparent format (also known as Directory Format) for Application Process Title and Application Entity Qualifier. If an instance identification specified in a FTAM partner list entry does not correspond to the previously supported numeric format nn.nn.nn or nn.nn.nn..nn (2 to 10 numbers separated by dots; optional is then a number for the Application Entity Qualifier separated by 2 dots), the following applies:

If, with a fundamental numeric specification for an object identifier, non-permitted values (e.g. a value not equal to 0, 1 or 2 in the first component) or more than 10 components are specified, setting the Application Entity Title is rejected. However, the value range specially specified for the Application Entity Title as of the third component between 0 and 67108863 is not checked.

Specifications beginning with a blank or where a string (which begins with a blank) is attached to a fundamental numeric specification are rejected. These formats are reserved for any extensions.

All other specifications which contain a character not equal to a decimal number or dot are sent and evaluated as ISO646 string in the transparent format of the Application Process Title.

These identifications in the transparent format are only evaluated as Application Process Title as far as the last occurrence of two sequential dots. The character string behind the last sequence of two dots is interpreted as Application Entity Qualifier in the transparent format. Exception: the identification ends with .. or ..#

Examples: (specifications in transparent format are shown in hexadecimal form as they would appear in the trace. APT means "Application Process Title" and AEQ means "Application Entity Qualifier"):

-id=1.3.5.6.2	APT = iso identified-organization 5 6 2
-id=1.3.5.6.2..62	APT = iso identified-organization 5 6 2 AEQ = 62
-id=01.03.05.06	APT = iso identified-organization 5 6 *)
-id=1.3.5.6.	APT = iso identified-organization 5 6 0 *)
-id=1.3.5.6..	APT = iso identified-organization 5 6 AEQ = 0 *)
-id=1.3.5.6..#	APT = 0x312e332e352e36 *)
-id=9.3.4.5.2	is rejected as only 0, 1 or 2 would be permitted at the first digit
-id=1.2.3.4.5.6.7..8.9	APT = 0x312e322e332e342e352e362e37 AEQ = 0x382e39
-id=1.2.3.4.5.6.7.8.9.10.11	is rejected
-id=%ip139.28.87.55	APT = 0x2569703133392e32382e38372e3535

-id=emil	APT = 0x656d696c
-id=1.3.5A	APT = 0x312e332e3541
„-id=1.3.5 A“	is rejected
„-id=Emil Huber“	APT = 0x456d696c204875626572
-id=Emil.Huber	APT = 0x456d696c2e4875626572
-id=Emil..Huber	APT = 0x456d696c AEQ = 0x4875626572
-id=Emil..	APT = 0x456d696c *
-id=Emil..#	APT = 0x456d696c *

*) These specifications are not suitable for a sender check via Application Entity Title if the partner is openFT. Instead, leading zeros, and strings ending with . or .. must be avoided in numeric specifications, and - with transparent format .. or ..# at the end must not exist.

3.2 Sender check with the Application Entity Title

It is possible in previous versions of openFT to specify Application Entity Titles for partners; but there is no evaluation of AETs. This continues to apply for standard settings in the operating parameters; FTAM partners are always identified via their sender address. The partner check option (-ptc) in the ftmodo command has been extended:

```
ftmodo ...
      [ -ptc=i | -ptc=a | -ptc=t | -ptc=b ]
```

You can use this to modify the global settings for the sender check. These settings only have an effect for named partners which are connected via the openFT or FTAM protocol and which do not work with authentication (with openFT protocol). With FTAM partners, the setting is only then effective if the partner identifies itself with a Calling Application Entity Title which does not correspond to the nil AP Title.

- ptc=i (identification) Only the identification is checked with partners with the openFT protocol. With FTAM partners the transport address is checked; any specified Calling Application Entity Title is ignored.
- ptc=a (address) The transport address is checked with partners with the openFT protocol in addition to the identification. With FTAM partners the transport address is checked; any specified Calling Application Entity Title is ignored.
- ptc=t (application entity title) Only the identification is tested with partners with the openFT protocol. With FTAM partners only the Calling Application Entity Title is checked if it is specified and does not correspond to the nil AP title; otherwise, the transport address is checked.
- ptc=b (both) The transport address is checked with partners with the openFT protocol in addition to the identification. With FTAM partners both the Calling Application Entity Title as well as the transport address is checked if a Calling AET is specified which is not equal to the nil AP title; otherwise, the transport address is checked.

With ftshwo -csv these options are shown in the column **PartnerCheck** as *STD, *TRANSP-ADDR, *AET and *BOTH. In the standard output of the ftshwo command, they are shown under **PTN-CHK** as STD, ADDR, AET and T+AE.

For FTAM partners there are no partner-specific settings for the sender check.

If -ptc=t or -ptc=b has been set, any identifications specified when entering or modifying FTAM partners are checked for uniqueness. If the Application Entity Title is used for partner identification, we recommend setting the option -ptc at this value from the start as existing ambiguities could persist.

In the **openFT Explorer** the field **Partner Check** on the dialog page **General** of the dialog Operating parameters has been extended. In addition to the previously available radio buttons **Identification** and **Address** there are also the alternatives **AET and Identification** (corresponds to -ptc=t) as well as **AET and Address** (corresponds to -ptc=b). The two new alternatives can only be set by the FT administrator for openFT instances as of V12.0B. These options are not available for other users and older versions.

3.3 Setting of the Calling Application Entity Title

The own Application Entity Title can be specified in the ftmodo command. It is sent in the initiator role as "calling AET" and in the responder role as "responding AET":

```
ftmodo ...
-aet=@n|i|<application entity title 1..64>
```

The specifications have the following meaning:

- aet=@n The previous regulations apply. If -ae=y is set, the nil APTitle is used. If -ae=n is set, no Application Entity Title is sent.
- aet=@i The instance identification is used as Application Entity Title (ftmodo -id=...). Syntactically incorrect specifications in the instance identification for the Application Entity Title should be avoided in this situation. In most of these situations, the nil APTitle is currently set instead; this can change in future versions.
- aet=<AET> The explicit specification of a Calling/Responding Application Entity Title is possible.
The format rules apply as described above.

In the CSV output this setting is output in the column ApplicationEntityTitle. In addition to the explicit output of an Application Entity Title which is enclosed in inverted commas, the outputs *NSPEC and *IDENTIFICATION are possible.

The settings for the Application Entity are not in the standard output of ftshwo. This can be output via its own option:

```
ftshwo -ae
```

Example output:

```
LocalAET                                = Emil.Huber..private
AETitle format 1 (transparent)
Application Process Title               = Emil.Huber
Application Entity Qualifier = private
```

The first line "LocalAET" contains the AET specification as a whole – either as reference to the identification (*IDENTIFICATION) or as explicit string. It is missing if no specification for an Application Entity Title has been made (ftmodo -aet=@n).

The next line specifies the format of the Application Entity Title:

- no AETitle No Calling AET is sent
- nil AETitle The nil APTitle is sent as Calling AET
- AETitle format 1 (transparent) The Calling AET is defined in the transparent format
- AETitle format 2 (numeric) The Calling AET is defined in the numeric format

The two following (optional) lines output the Application Process Title and the Application Entity Qualifier.

openFT Explorer has the field **Application Entity Title** to set the Application Entity Title on the dialog page **Options** of the dialog Operating parameters. You can select from the following alternatives:

Nil

The nil APTitle is sent as Calling AET; it corresponds to the ftmodo parameters -ae=y -aet=@n

None

No Calling AET is sent; it corresponds to the ftmodo parameters -ae=n -aet=@n

Identification

The instance identification is sent as Calling AET; it corresponds to ftmodo -aet=@i

AET with field to specify and display the Application Entity Title

The specified AET is used; it corresponds to ftmodo -aet=<AET>

3.4 Display of the Application Entity Title for FTAM partner

The specification of identification in a partner list entry for FTAM partners is evaluated as Application Entity Title for this partner: this is sent as called AET with a local job, and it can be used for the sender check with a remote job.

If an individual FTAM partner is output in long format:

```
ftshwptn <ftam-Partner> -l
```

The entered identification is output decoded as Application Entity Title.

Example:

NAME	STATE	SECLEV	PRI	TRACE	LOC	REM	P-CHK	ADDRESS
	INBND	REQU-P		RECOV				ROUTING IDENTIFICATION
myftam	ACT	STD	NORM	FLOPT	0	0		ftam://d012ze28.due.fxy.net
	ACT	SERIAL		FLOPT				1.0.795.323.64
AETitle format 2 (numeric)								
Application Process Title = 1.0.795.323.64								

If the identification cannot be read as valid Application Entity Title, this additional output is missing. It is also missing if several partners are to be listed.

4 Support of FarSync X.25 under Linux and Windows

As of openFT V12.0B00 FarSync X.25 cards from the manufacturer FarSite are directly supported by openFT under Linux and Windows. The coupling method XOT (X.25 via TCP/IP) is also supported under Linux by using the FarSync XOT Runtime. PCMX is no longer required for this. In order to enable the usage of the direct X.25 support in openFT, CMX operation must be switched off (ftmodo -cmx=n).

Under **Windows** a maximum of 4 FarSync X.25 cards (adapters) with a maximum of 4 lines each are supported. All in all, 16 lines are supported. Adapters and lines are numbered from 0 to 3. Each line is thus uniquely identified by a combination of adapter number and line number (0-3).

The respective latest version of FarSync X.25 software for Windows is recommended for use with openFT. This is currently

- 4.0 As of Windows XP up to and including Windows Vista.
- 4.2.0 As of Windows 7

However, under **Linux** a separate device is configured for each line. A maximum of 16 lines are supported under Linux. The lines are numbered from 0 to 15. Each line is thus uniquely identified via its own number.

The minimum required version of FarSync X.25 software for Linux is 3.2.9.

The required X.25 software can be downloaded from
http://www.farsite.com/custsupp/Download_X.25_software.htm.

At least FarSync XOT Runtime version 3.2 is required for the support of XOT (X.25 via TCP/IP) under Linux. A coupling to the FarSync X.25 Gateway or another device from any manufacturer which does support XOT is thus possible. A support of the FarSync XOT Runtime under Windows is not possible as the XOT Runtime cannot be addressed via the NCB interface required by openFT.

The FarSync XOT Runtime provides as default under Linux a virtual X.25 interface. The configuration of openFT for using FarSync XOT Runtime is identical with the configuration for using FarSync X.25 cards. In order to address XOT Runtime only the adapter number (standard value: Adapter 0) of the XOT adapter (of the virtual X.25 interface) must be specified. An entry for XOT routing must also be created in the FarSync configuration program, i.e. for outgoing connections, an allocation must be created between a DTE target address and the IP target address of the XOT partner.

Note: Manual changes in the configuration files of the XOT-Runtime enable the use of several XOT adapters. Due to problems in the XOT-Runtime it is not recommended to use multiple XOT adapters for openFT.

4.1 Configuration of the FarSync X.25 transport system in openFT

A separate DTE address can be assigned to each X.25 line on the FarSync X.25 card, which is sent with outgoing connections as DTE address of the sender (Calling DTE Address) in the X.25 connection set-up. The assignment of a DTE address to a line is optional. If no DTE address can be assigned to a line, then no DTE sender address is sent in the X.25 connection set-up.

4.1.1 Command ftmodo

A DTE address can be assigned to a line on the FarSync X.25 card using the ftmodo command:

The ftmodo commando was thus extended to include the option -x25.

Under **Windows** the ftmodo command looks like this:

```
ftmodo ...
[ -x25=[<0..3>:<0..3>[=<DTE 1..15>]..[,<0..3>:<0..3>[=<DTE 1..15>]]] ]
```

With the option -x25 a DTE address can be assigned to one or more lines. If DTE addresses are specified for several lines, the individual values must be separated from each other by a comma. A maximum of 16 lines can be simultaneously specified.

-x25=<Adapter>:<Line>=<DTE address>
 assigns a DTE address to an X.25 line which is uniquely identified by a combination of adapter number and line number.

Default setting after a new installation: a DTE address is not assigned to any line.

-x25=<Adapter>:<Line>=
 resets the assigned DTE address to an X.25 line which is uniquely identified by a combination of adapter number and line number, i.e. the DTE address is deleted.

-x25=
 resets all DTE addresses for all X.25 lines. All assigned DTE addresses are deleted.

Example: On the FarSync X.25 adapter number 0 the DTE address 1234 is to be assigned to line number 2.

```
ftmodo -x25=0:2=1234
```

Example: On the FarSync X.25 adapter number 2 the DTE address 111111 is to be assigned to line 0 and the DTE address 222222 is to be assigned to line 1.

```
ftmodo -x25=2:0=111111,2:1=222222
```

Example: On the Far Sync X.25 adapter 2 the DTE address stored for line 0 is to be deleted.

```
ftmodo -x25=2:0=
```

Example: All stored DTE addresses for all lines are to be deleted.

```
ftmodo -x25=
```

Under **Linux** the ftmodo command is as follows:

```
ftmodo ...
[ -x25=[<0..15>[=<DTE 1..15>]..[,<0..15>[=<DTE 1..15>]]] ]
```

Under Linux all lines for all adapters are numbered starting with 0. A DTE address can be assigned to one or more lines. If DTE addresses are specified for several lines, the individual values must be separated from each other by a comma. A maximum of 16 lines can be simultaneously specified.

-x25=<Adapter>=<DTE address>
 assigns a DTE address to an X.25 line which is uniquely identified by an adapter number.

Default setting after a new installation: a DTE address is not assigned to any line.

`-x25=<Adapter>=`

resets the DTE address assigned to an X.25 line which is uniquely identified via the adapter number, i.e. the DTE address is deleted.

`-x25=`

resets all DTE addresses for all X.25 lines. All assigned DTE addresses are deleted.

Example: The DTE address 1234 is to be assigned to FarSync X.25 adapter number 0.

```
ftmodo -x25=0=1234
```

Example: The DTE address 111111 is to be assigned to FarSync adapter number 2 and the DTE address 222222 is to be assigned to adapter 3.

```
ftmodo -x25=2=111111,3=222222
```

Example: The DTE address assigned to FarSync X.25 adapter 2 is to be deleted.

```
ftmodo -x25=2=
```

Example: All stored DTE addresses for all lines are to be deleted.

```
ftmodo -x25=
```

4.1.2 openFT Explorer – Operating parameters

In openFT Explorer the Operating parameters dialog has been extended to include the dialog page **X.25**. This dialog page is only available for openFT instances which support the FarSync X.25 transport system.

The configured FarSync adapters are displayed in a table on the dialog page in the section **Adapter configuration**. Under Windows the table has the columns **Adapter**, **Line** and **DTE address**. Under Linux there are only the columns Adapter and DTE address. The configuration of the adapters can be modified via the buttons **Add**, **Remove** and **Edit**.

Add... calls the dialog **Configure adapter** in order to define a new adapter or line. The dialog has fields according to the table columns. Use the fields **Adapter** and **Line** to select the line that is to be configured via selection lists. In the field **DTE address** an address for the selected line must be configured as a 1 to 15-digit decimal number.

Remove deletes (if necessary after a check) the line that has been selected in the table.

Edit calls the dialog **Configure adapter** for the line that has been selected in the table. Only the field DTE address can be modified.

4.2 Partner address of an X.25 partner

The host part in the partner address has the following structure when using the FarSync X.25 transport system:

Windows:

- DTE address and local line number with the prefix %x25, for example
%x25[123456%0:3]

Linux:

- DTE address and local line number with the prefix %x25, for example
%x25[123456%0]

The square brackets [..] must be specified. The DTE target address of the partner system must be specified as the first value in the brackets. The DTE address is a mandatory parameter and must be specified. The **local** line number of the FarSync X.25 card must be specified after the percentage via which the connection to the partner system is to be set up. The specification of the local line number is optional. If none is specified, the line 0:0 is automatically used in Windows and the line 0 is used in Linux.

If optional X.25 facilities or a special value for the transport protocol class are to be specified when setting up the connection, it is **essential** for the partner system to create a named partner list entry.

The ftaddptn command has been extended to create a partner list entry for a X.25 partner. Please note: the specification of the following options is only permitted if the partner address (option -pa) contains a valid X.25 address (beginning with %x25).

4.2.1 Command ftaddptn

Under **Windows** the ftaddptn command has been extended to include the following options:

```
ftaddptn ...
[ -nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] | 2..40 ]
[ -cl=0/- | -cl=2/0 | -cl=2/2 ]
[ -ws=<1..127> ]
[ -ps=16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 | 4096 ]
[ -cud=<2..32> ]
[ -cug=<0..9999> ]
[ -thr=75 | 150 | 300 | 600 | 1200 | 2400 | 4800 | 9600 |
    19200 | 48000 | 64000 | 128000 | 192000 ]
[ -rch=y | -rch=n ]
[ -sif=<0..3>:<0..3>[,<0..3>:<0..3>]..[,<0..3>:<0..3>] ]
```

The options have the following meaning:

-nsap=network address of the partner system

The network address (NSAP = Network Service Access Point) can be specified in two formats. As OSI network address comprised of the components AFI (Authority and Format Identifier), IDI (Initial Domain Identifier) and DSP (Domain-Specific Part) or in free format as a hexadecimal string. The NSAP is sent as so-called "Called NSAP" in the X.25 connection set-up.

The specification of the NSAP is optional. If the network address of the partner system is not specified, then no value is sent for the NSAP of the partner system in the X.25 connection set-up.

In the OSI format the individual NSAP components (AFI, IDI and DSP) must be specified separated by dots. Optional parts of an NSAP can be omitted but the leading dot must be specified.

AFI:

The Authority and Format Identifier for the NSAP of the remote address is specified. You can obtain the corresponding specification from your network operator or partner. The AFI value defines the length and possible values of the IDI and the length and format of the DSP. The AFI is an NSAP mandatory parameter. The values 36 to 59 are supported for the AFI.

IDI:

The Initial Domain Identifier for the NSAP of the remote address is specified. You can obtain the corresponding specification from your network operator or partner.

DSP:

The Domain-Specific Part for the NSAP of the remote address is specified. You can obtain the corresponding specification from your network operator or partner.

The possible AFI, IDI and DSP values can be seen in the following table. Please note that only even DSP digit numbers are permitted for the hexadecimal DSP syntax (even if the maximum value is not reached!).

AFI	IDI Min. Length	IDI Max. Length	IDI Set of values	DSP Min. Length	DSP Max. Length	DSP Set of values
36	1	14	Decimal	0	24	Decimal
37	1	14	Decimal	2	24 (12 x 2)	Hexadecimal
38	3	3	Decimal	1	35	Decimal
39	3	3	Decimal	2	34 (17 x 2)	Hexadecimal
40	1	8	Decimal	0	30	Decimal
41	1	8	Decimal	2	30 (15 x 2)	Hexadecimal
42	1	12	Decimal	0	26	Decimal
43	1	12	Decimal	2	26 (13 x 2)	Hexadecimal
44	1	15	Decimal	0	23	Decimal
45	1	15	Decimal	2	22 (11 x 2)	Hexadecimal
46	4	4	Decimal	1	34	Decimal
47	4	4	Decimal	2	34 (17 x 2)	Hexadecimal
48	0	0	Decimal	1	38	Decimal
49	0	0	Decimal	2	38 (19 x 2)	Hexadecimal
50	0	0	Decimal	2	38 (19 x 2)	Hexadecimal
51	0	0	Decimal	2	38 (19 x 2)	Hexadecimal
52	1	14	Decimal	0	24	Decimal
53	1	14	Decimal	2	24 (12 x 2)	Hexadecimal
54	1	8	Decimal	0	30	Decimal
55	1	8	Decimal	2	30 (15 x 2)	Hexadecimal
56	1	12	Decimal	0	26	Decimal
57	1	12	Decimal	2	26 (13 x 2)	Hexadecimal
0.58	1	15	Decimal	0	23	Decimal
59	1	15	Decimal	2	22 (11 x 2)	Hexadecimal

-cl=transport protocol class

The proposed and alternative transport protocol class according to ISO 8073 can be defined for the connection set-up to the remote application.

The specification of the transport protocol class is optional. The transport protocol class 2/0 is used as default.

Select	If
2/2	class 2 proposed and is to be accepted as alternative class
2/0	Class 2 proposed and class 0 is to be accepted as alternative class
0/-	Class 0 proposed and is to be accepted as alternative class

-ws=window size

You can define the window size. A differentiation between incoming/outgoing window size is not possible. The format is a decimal number.

The value range is 1 - 127.

The specification of the window size is optional.

-ps=packet size

The package size can be selected. A differentiation between incoming/outgoing package size is not possible.

The value range is: 16, 32, 64, 128, 256, 512, 1024, 2048, 4096.

The specification of the package size is optional.

-cud=user data for the X.25 connection setup

The transport protocol identification (max. 32 hexadecimal digits) can be specified if it is expected with the X.25 connection-set up to the remote application. It is sent with the connection set-up as Call User Data.

The specification of the user data is optional.

-cug=closed user group

You can define a closed user group consisting of up to 4 decimal digits.

The value range is 0 - 9999.

The specification of the closed user group is optional. If the closed user group is not specified, no value is specified in the X.25 connection set-up.

-thr=throughput class

The throughput class can be selected. A differentiation between incoming/outgoing throughput class is not possible.

The value range is: 75, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 48000, 64000, 128000, 192000.

The specification of the throughput class is optional. If the throughput class is not specified, no value is specified in the X.25 connection set-up.

-rch=reverse charging

You can define the charge transfer.

The value range is: y (yes) or n (no).

The specification of the charge transfer class is optional. If the charge transfer is not specified, no value is specified in the X.25 connection set-up.

-sif=alternative line

Specify the line which is to be used as alternative line if there is a failed connection set-up to the remote system. Up to 5 lines can be specified separated by a comma.

If the connection set-up via the line specified in the partner address does not work, the connection set-up is attempted using the sequence of line numbers specified here.

A line is uniquely defined via the combination of adapter number: line number.

The adapter and line numbers specified here must be configured accordingly in the configuration program for the FarSync X.25 cards.

The specification of one or more alternative lines is optional.

Example:

```
ftaddptn mchx25 -pa=%x25[123456789012345%0:0] -nsap=43.123.45678901
-cud=12345678901234567890123456789012 -cug=9999 -rch=n -ws=7
-ps=4096 -thr=192000 -cl=2/2 -sif=1:0,2:0
```

Under **Linux** the ftaddptn command has been extended to include the following options:

```
ftaddptn ...
[ -nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] | 2..40 ]
[ -cl=0/- | -cl=2/0 | -cl=2/2 ]
[ -ws=<1..127> ]
[ -ps=16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 | 4096 ]
[ -cud=<2..32> ]
[ -cug=<0..9999> ]
[ -thr=75 | 150 | 300 | 600 | 1200 | 2400 | 4800 | 9600 |
      19200 | 48000 | 64000 | 128000 | 192000 ]
[ -rch=y | -rch=n ]
[ -sif=[0][,1][,2][,3]..[,15] ]
```

The options with ftaddptn command in Linux have the same meaning as under Windows. However, the description for the option -sif differs from Windows due to the different line numbers:

-sif=alternative line

Specify the line which is to be used as alternative line if there is a failed connection set-up to the remote system. Up to 5 lines can be specified separated by a comma. If the connection set-up via the line specified in the partner address does not work, the attempt to set up the connection is continued using the sequence of the following line numbers.

A line is uniquely defined via the adapter number.

The adapter numbers specified here must be configured accordingly in the configuration program for the FarSync X.25 cards.

The specification of one or more alternative lines is optional.

Example:

```
ftaddptn mchx25 -pa=%x25[123456789012345%0] -nsap=43,123,45678901
-cud=12345678901234567890123456789012 -cug=9999 -rch=n -ws=7
-ps=4096 -thr=192000 -cl=2/2 -sif=1.2
```

4.2.2 Command ftmodptn

The `ftmodptn` command under **Windows** has been extended to include the following options for the modification of a partner list entry with an X.25 partner address:

```
ftmodptn ...
  [ -nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] | 2..40 ]
  [ -cl=0/- | -cl=2/0 | -cl=2/2 ]
  [ -ws=<1..127> ]
  [ -ps=16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 | 4096 ]
  [ -cud=<2..32> ]
  [ -cug=<0..9999> ]
  [ -thr=75 | 150 | 300 | 600 | 1200 | 2400 | 4800 | 9600 |
    19200 | 48000 | 64000 | 128000 | 192000 ]
  [ -rch=y | -rch=n ]
  [ -sif=[0][,1][,2][,3]..[,15] ]
```

Caution: If the type of partner address (option `-pa`) is changed for the `ftmodptn` command to the extent that e.g. an X.25 address becomes a TCP/IP-RFC1006 address, all X.25-specific address parameters will be deleted.

`-nsap`=network address of the partner system

See description of the parameter for the command `ftaddptn`.

`-nsap`=

Deletes the setting for the network address of the partner system.

`-cl`=transport protocol class

See description of the parameter for the command `ftaddptn`.

`-cl`=

Deletes the setting for the transport protocol class

`-ws`=window size

See description of the parameter for the command `ftaddptn`.

`-ws`=

Deletes the setting for the window size

`-ps`=packet size

See description of the parameter for the command `ftaddptn`.

`-ps`=

Deletes the setting for the packet size

`-cud`=user data for the X.25 connection setup

See description of the parameter for the command `ftaddptn`.

`-cud`=

Deletes the setting for the user data

`-cug`=closed user group

See description of the parameter for the command `ftaddptn`.

`-cug`=

Deletes the setting for the closed user group

`-thr`=throughput class

See description of the parameter for the command `ftaddptn`.

`-thr`=

Deletes the setting for the throughput class

`-rch`=reverse charging

See description of the parameter for the command `ftaddptn`.

`-rch`=

Deletes the setting for the reverse charging

`-sif`=alternative line

See description of the parameter for the command `ftaddptn`.

-sif=
 Deletes the setting for the alternative line.

Example Windows:

```
ftmodptn mchx25 -nsap= -ws=2 -ps=128 -thr=192000 -sif=3:0
```

Under **Linux** the ftmodptn command has been extended to include the following options:

```
ftmodptn ...
[ -nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] | 2..40 ]
[ -cl=0/- | -cl=2/0 | -cl=2/2 ]
[ -ws=<1..127> ]
[ -ps=16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 | 4096 ]
[ -cud=<2..32> ]
[ -cug=<0..9999> ]
[ -thr=75 | 150 | 300 | 600 | 1200 | 2400 | 4800 | 9600 |
    19200 | 48000 | 64000 | 128000 | 192000 ]
[ -rch=y | -rch=n ]
[ -sif=[0][,1][,2][,3]..[,15] ]
```

The options for the ftmodptn command in Linux have the same meaning as under Windows. However, the description for the option -sif differs from Windows due to the different line numbers:

-sif=alternative line
 See description of the parameter for the command ftaddptn for Linux.
 -sif=
 Deletes the setting for the alternative line.

Example Linux:

```
ftmodptn mchx25 -nsap= -ws=2 -ps=128 -thr=192000 -sif=3
```

4.2.3 Command ftshwptn – normal output

The ftshwptn command under Linux and Windows has been extended to include the display option -pa for the display of the address-specific additional parameters:

```
ftshwptn ...
[ -l | -csv | -px | -pw | -p2 | -pz | -pa ]
```

Example Windows:

```
ftshwptn mchx25 -pa
NAME = mchx25
TYPE = X.25 [FarSync] ID = 0
DTE = 123456789012345
NSAP = 4300000000012345678901
AFI = 43
```

```

    IDI    = 123
    DSP    = 45678901
    CUD    = 03010100
    CLASS  = 2/2      WSIZE    = 7      PSIZE    = 4096
    CUG    = 9999     THPUTCL  = 192000  REVCHRG  = NO
    IF     = 0:0      SPARE-IF = 1:0,2:0

```

Example Linux:

```

ftshwptn mchx25 -pa
NAME = mchx25
  TYPE    = X.25 [FarSync]          ID      = 0
  DTE     = 123456789012345
  NSAP    = 43000000000012345678901
    AFI    = 43
    IDI    = 123
    DSP    = 45678901
  CUD     = 03010100
  CLASS   = 2/2      WSIZE    = 7      PSIZE    = 4096
  CUG     = 9999     THPUTCL  = 192000  REVCHRG  = NO
  IF      = 0        SPARE-IF = 1,2

```

Explanation:

NAME

Name of the partner list entry

TYPE

Address type

X.25 [FarSync] X.25 address for the FarSync X.25 transport system

TCP/IP IPv4 or IPv6 address for the transport system TCP/IP-RFC1006

HOST/TNS Host or TNS name

ID

Index of address extension. Is only used for diagnosis purposes.

DTE

DTE address of the partner system

NSAP

NSAP address of the partner system. If the NSAP is specified as the OSI network address, this is then followed by the individual values for AFI, IDI and DSP.

AFI

Authority and format identifier of the NSAP.

IDI

Initial domain identifier of the NSAP

DSP

Domain-specific part of the NSAP

CUD

User data for the X.25 connection setup

CLASS

Transport protocol class

WSIZE

Window size

PSIZE

Packet size

CUG
Closed user group

THPUTCL
Throughput class

REVCHRG
Reverse charging

IF
Local line on the FarSync X.25 card that is used to set up the connection

SPARE-IF
Alternative line and list with alternative lines on the FarSync X.25 card, via which a further connection setup is initiated as an alternative in the case of a failed connection setup.

4.2.4 Command ftshwptn – CSV output

The CSV output has been extended to include the following values for the display of the X.25 parameters:

Example:

```
ftshwptn mchx25 -csv
```

```
PartnerName;Sta;SecLev;Trace;Loc;Rem;Processor;Entity;NetworkAddr;P
ort;PartnerCheck;TransportSel;LastAccessDate;SessionSel;Presentatio
nSel;Identification;SessRout;PartnerAddr;Check;AuthMand;Priority;AS
3;AuthLev;InboundSta;RequProc;OutboundRecovery;ForeignPartner;Scope
Id;AddrType;ExtensionID;DteAddress;NsapAddress;CallUserData;Transpo
rtClass;WindowSize;PacketSize;ClosedUserGroup;ThroughputClass;Rever
seCharging;SpareInterface
"mchx25";*ACT;*STD;*FTOPT;0;0;;; "%x25 [123456789012345]";;*FTOPT;"$f
jam";2015-04-
01;;; "%x25 [123456789012345]";; "%x25 [123456789012345%0:0]";*FTOPT;*N
O;*NORM;*NO; *ACT;*STD;*FTOPT;*NO;"0:0";"*X25";3;"123456789012345";
"43.123.45678901";"12345678901234567890123456789012";"2/2";7;4096;9
999;192000;*NO;"1:0,2:0"
```

The following table shows the CSV output format for the X.25 parameters of a partner list entry:

Column	Type	Values and meaning	Parameters
AddrType	String	*X25/*TCP/IP/*TNS Type of address	TYPE
ExtensionID	Number	Index of address extension (for internal use only)	ID
DteAddress	String	DTE address of 1 to 15 decimal digits in length	DTE
NsapAddress	String	NSAP OSI network address in the format AFI.IDI.DSP and as hexadecimal string in free format	NSAP and AFI.IDI.DSP
CallUserData	String	User data for the connection setup (e.g.	CUD

		transport protocol identification)	
TransportClass	String	2/2 / 2/0 / 0/- Transport protocol class	CLASS
WindowSize	Number	1..127 Window size	WSIZE
PacketSize	Number	16 / 32 / 64 / 128 / 256 / 512 / 1024/ 2048 / 4096 Packet size	PSIZE
ClosedUserGroup	Number	0..9999 Closed user group	CUG
ThroughputClass	Number	75 / 150 / 300 / 600 / 1200 / 2400 / 4800 / 9600 / 19200 / 48000 / 64000 / 128000 / 192000 Throughput class	THPUTCL
ReverseCharging	String	Reverse charging *NO / *YES	REVCHRG
SpareInterface	String	Alternative line and a list of comma separated values with alternative lines	SPARE-IF

4.2.5 openFT Explorer - Partner list entry

In openFT Explorer the **Partner list entry** dialog has been extended to include the dialog page **X.25**. This dialog page is only available for openFT instances that support the FarSync X.25 transport system. Furthermore, the FarSync X.25 transport system must be installed on the computer. If that is the case, the configured partner address (**Address** field on the **General** dialog page) must also correspond to the format of an X.25 address so that the X.25 dialog page can be called up. The form of the address must be "%x25[*]:" "*" stands for a decimal string of digits of 1 to 15 characters in length, to which an expression in format "%<0..3>:<0..3>" can be attached in Windows and "%<0..15>" in Linux in order to define an adapter. Optionally, one of the protocol prefixes "openft://" and "ftam://" can also be specified in each case.

Examples of valid X.25 addresses:

%x25[12345%0:1] (on Windows)
ftam://%x25[6789%12] (on Linux)

The X.25 dialog page allows the configuration of the parameters of the commands ftaddptn and ftmodptn as described in the previous sections. It contains the following fields:

NSAP (Network Service Access Point)

This option should be selected in order to define an NSAP. The NSAP fields described below are only available after the option has been selected.

An NSAP can either be defined in free format or in OSI format.

The definition in OSI format is the default setting.

Corresponds to the parameter -nsap.

Free Format

Value range: 4 to 40 hexadecimal digits

OSI Format

In OSI format an NSAP should be defined as consisting of an AFI as well as an optional IDI and DSP.

AFI

(Authority and Format Identifier)
Selection list with the supported AFI values

IDI (...)

(Initial Domain Identifier)
The corresponding address format (X.121, ISO_DCC, F.69, E.163, E.164, ISO_ICD or locally) is displayed between the brackets of the field name depending on the set AFI. Value range: depends on the value selected for AFI, see table in section 4.2.1.

DSP

(Domain-Specific Part)
Value range: depends on the value selected for AFI, see table in section 4.2.1.

Transport Protocol Identification

Value range: 0 to 32 hexadecimal digits
Corresponds to the parameter -cud

Transport Protocol Class

Selection list with the alternatives **Class 0**, **Class 2 (0 possible)** or **Class 2**.
Default setting: Class 2 (0 possible)
Corresponds to the parameter -cl. Class 0 stands for -cl=0/-. Class 2 (0 possible) stands for -cl=2/0. Class 2 stands for -cl=2/2.

Alternative Lines

Value range: up to five alternative lines as a list with comma separated values.
Under Windows each alternative line must be specified in format <0..3>:<0..3> and under Linux in format <0..15>.
Corresponds to the parameter -sif.

Packet Size

Selection list with the permitted packet sizes as well as the default value <none>.
Corresponds to the parameter -ps.

Window Size

Selection list with the permitted window sizes as well as the default value <none>.
Corresponds to the parameter -ws.

Throughput Class

Selection list with the permitted throughput classes as well as the default value <none>.
Corresponds to the parameter -thr.

Closed User Group

Value range: Decimal digit between 0 and 9999
Corresponds to the parameter -cug.

Reverse Charging

Selectable option. The option is not selected as a default setting.
Corresponds to the parameter -rch.

4.3 Local X.25 address

The X.25 transport system can be used by the openFT and FTAM protocol. It cannot be used by the FTP and FTADM protocol.

4.3.1 Command ftmodo

To enable the use of the X.25 transport system to be configured for the openFT and the FTAM protocol the ftmodo command has been extended to include the following options:

Windows:

```
ftmodo ...
[ -openftx25=y | -openftx25=n ]
[ -openftx25lif=[0][,1][,2][,3] ]
[ -openftx25lnb=<1..99> ]
[ -openftx25cl=0/- | -openftx25cl=2/0 | -openftx25cl=2/2 ]
[ -openftx25nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] |
  2..40 ]
[ -ftamx25=y | -ftamx25=n ]
[ -ftamx25lif=[0][,1][,2][,3] ]
[ -ftamx25lnb=<1..99> ]
[ -ftamx25cl=0/- | -ftamx25cl=2/0 | -ftamx25cl=2/2 ]
[ -ftamx25nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] |
  2..40 ]
```

The options have the following meaning:

-openftx25=y | -openftx25 = n

You can enable and disable the use of the X.25 transport system for the openFT protocol here.

y This selection enables the use of the X.25 transport system to be activated for the openFT protocol. The openFT protocol attaches to the X.25 transport system.

n The use of the X.25 transport system is deactivated by the openFT protocol. No attach to the X.25 transport system by the openFT protocol takes place.

Default setting after new installation: *-openftx25=n*

-openftx25lif=[0][,1][,2][,3]

Here you can specify the adapter numbers of the FarSync X.25 cards, which the openFT protocol in the asynchronous openFT server is attached to in order to accept incoming connections. If no adapter has been selected for the openFT protocol, the incoming X.25 connections for the openFT protocol are then **not** accepted.

Default setting after new installation: *-openftx25lif=0*

-openftx25lif=

Resets for the openFT protocol the setting for the FarSync X.25 cards, on which the incoming connections are to be accepted, i.e. incoming X.25 connections for the openFT protocol are not accepted.

-openftx25lnb=number of list() calls

The number of list() calls per FarSync X.25 card is specified here.
Possible values: 1 to 99

Default setting after new installation: *-openftx25lnb=5*

To enable an incoming connection request to be accepted an application must place at least one list() call via the FarSync program interface. If a connection request is made, this is then reported by the list() call and the connection is established. However, each list() call can only be used for one connection and must therefore be replaced by a new list() call. This is done immediately after the connection has been accepted.

However, with a high load it is possible for there to be another connection request precisely in this brief period between the acceptance of an incoming connection and the placing of a new list() call. This is rejected by the driver of the FarSync card, as the driver does not in principle buffer any incoming connection requests.

This behavior is stipulated by the design of the FarSync program interface. This problem can be remedied by placing several list() calls per adapter. As standard, 5 list() calls are placed in each case for each adapter that is configured with the option `-openftx25lif`.

`-openftx25cl=0/-` | `-openftx25cl=2/0` | `-openftx25cl=2/2`

This is where you specify for the openFT protocol which transport class is accepted for incoming connections.

- 0/- Transport class 0 is to be used for incoming transport connections. In this case, incoming connections are (if possible) set down to transport class 0. If this is not possible, the connection request is rejected.
- 2/0 Both transport class 2 and 0 can be used for incoming transport connections.
- 2/2 Transport class 2 is to be used for incoming transport connections, i.e. only connections with transport class 2 are accepted. Incoming connections with transport class 0 are rejected.

Default setting after new installation: `-openftx25cl=2/0`

`-openftx25nsap=`network address of the local openFT application

You can enter the network address (NSAP) of the local openFT application here, i.e. for the openFT protocol. The NSAP is used as follows:

- To identify the sender for a connection setup (Calling NSAP) via the openFT protocol. If no NSAP is entered, then "Calling NSAP" is not included in outgoing connections.
- To identify the receiver for incoming connections (Called NSAP) via the openFT protocol.
Please note that an incoming connection is only setup if the "Called NSAP" supplied by the partner matches the NSAP specified here. If e.g. no NSAP is specified here, the partner may not send a "Called NSAP", either.

Setup and format of the NSAP is identical to the description of the NSAP for the command `ftaddptn`, option `-nsap`.

Default setting after new installation: `-openftx25nsap=`

`-openftx25nsap=`

Resets for the openFT protocol the setting for the local NSAP.

`-ftamx25=y | -ftamx25 = n`

You can enable and disable the use of the X.25 transport system for the FTAM protocol here.

- y This selection enables the use of the X.25 transport system to be activated for the FTAM protocol. The FTAM protocol attaches to the X.25 transport system.
- n The use of the X.25 transport system is deactivated by the FTAM protocol. No attach to the X.25 transport system by the FTAM protocol takes place.

Default setting after new installation: `-ftamx25=n`

`-ftamx25lif=[0],[1],[2],[3]`

Here you can specify the adapter numbers of the FarSync X.25 cards, which the FTAM protocol in the asynchronous openFT server is attached to in order to accept incoming connections. If no adapter has been selected for the FTAM protocol, the incoming X.25 connections for the FTAM protocol are then **not** accepted.

Default setting after new installation: `-ftamx25lif=0`

`-ftamx25lif=`

Resets for the FTAM protocol the setting for the FarSync X.25 cards, on which the incoming connections are to be accepted, i.e. incoming X.25 connections for the FTAM protocol are not accepted.

`-ftamx25lnb=number of list() calls`

The number of list() calls per FarSync X.25 card can be specified for the FTAM protocol.

Possible values: 1 to 99

Default setting after new installation: `-ftamx25lnb=5`

To enable an incoming connection request to be accepted an application must place at least one list() call via the FarSync program interface. If a connection request is made, this is then reported by the list() call and the connection is established. However, each list() call can only be used for one connection and must therefore be replaced by a new list() call. This is done immediately after the connection has been accepted.

However, with a high load it is possible for there to be another connection request precisely in this brief period between the acceptance of an incoming connection and the placing of a new list() call. This is rejected by the driver of the FarSync card, as the driver does not in principle buffer any incoming connection requests.

This behavior is stipulated by the design of the FarSync program interface. This problem can be remedied by placing several list() calls per adapter. As standard, 5 list() calls are placed in each case for each adapter that is configured with the option `-ftamx25lif`.

`-ftamx25cl=0/- | -ftamx25cl=2/0 | -ftamx25cl=2/2`

This is where you specify for the FTAM protocol which transport class is accepted for incoming connections.

- 0/- Transport class 0 is to be used for incoming transport connections. In this case, incoming connections are (if possible) set down to transport class 0. If this is not possible, the connection request is rejected.
- 2/0 Both transport class 2 and 0 can be used for incoming transport connections.
- 2/2 Transport class 2 is to be used for incoming transport connections, i.e. only connections with transport class 2 are accepted. Incoming connections with transport class 0 are rejected.

Default setting after new installation: *-ftamx25cl=2/0*

-ftamx25nsap=network address of the local FTAM application

You can enter the network address (NSAP) of the local FTAM application here, i.e. for the FTAM protocol. The NSAP is used as follows:

- To identify the sender for a connection setup (Calling NSAP) via the FTAM protocol. If no NSAP is entered, then "Calling NSAP" is not included in outgoing connections.
- To identify the receiver for incoming connections (Called NSAP) via the FTAM protocol.

Please note that an incoming connection is only setup if the "Called NSAP" supplied by the partner matches the NSAP specified NSAP here. If e.g. no NSAP is specified here, the partner may not send a "Called NSAP", either.

Setup and format of the NSAP is identical to the description of the NSAP for the command *ftaddptn*, option *-nsap*.

Default setting after new installation: *-ftamx25nsap=*

-ftamx25nsap=

Resets for the FTAM protocol the setting for the local NSAP.

Linux:

```
ftmodo ...
[ -openftx25=y | -openftx25=n ]
[ -openftx25lif=[0][,1][,2][,3]..[,15] ]
[ -openftx25lnb=<1..99> ]
[ -openftx25cl=0/- | -openftx25cl=2/0 | -openftx25cl=2/2 ]
[ -openftx25nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] |
  2..40 ]

[ -ftamx25=y | -ftamx25=n ]
[ -ftamx25lif=[0][,1][,2][,3]..[,15] ]
[ -ftamx25lnb=<1..99> ]
[ -ftamx25cl=0/- | -ftamx25cl=2/0 | -ftamx25cl=2/2 ]
[ -ftamx25nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] |
  2..40 ]
```

The options have the identical meaning as under Windows. There are differences in the options *-openftx25lif* and *-ftamx25lif*, as the value range is different here. In Linux adapter numbers between 0 and 15 can be specified here.

4.3.2 Command ftshwo – normal output

The command ftshwo has been extended for the display of the parameters for the transport system X.25 with FarSync cards:

```
ftshwo ...
      [ -csv | -px | -pw | -p2 | -pz | -x25 ]
```

Example:

```
ftshwo -x25
```

WINDOWS :

```
ADAPTER LINE DTE
0        0    12345
0        1    54321
1        0    22222
1        1    33333
```

```
OPENFT-APPL
USE X.25  NUM-LISTEN  CLASS  ADAPTER
NO        3          0/-    0,1
NSAP     = 4300000000012345678901
AFI      = 43
IDI      = 123
DSP      = 45678901
```

```
FTAM-APPL
USE X.25  NUM-LISTEN  CLASS  ADAPTER
YES       4          2/0    1
NSAP     = 4300000000032110987654
AFI      = 43
IDI      = 321
DSP      = 10987654
```

Explanation:

- ADAPTER** Number of the FarSync X.25 adapter.
- LINE** Number of the line on the appropriate FarSync X.25 adapter.
- DTE** DTE address which is allocated to the line that is clearly identified from the combination of adapter number and line number.
- OPENFT-APPL** FarSync X.25-specific settings for the openFT protocol.
- USE X.25** Specifies whether the openFT protocol attaches to the FarSync X.25 transport system in order to use it.
 - YES** The openFT protocol attaches to the FarSync X.25 transport system.
 - NO** The openFT protocol does not attach to the FarSync X.25 transport system.
- NUM-LISTS** Specifies the number of list() calls per FarSync X.25 adapter by the openFT protocol.

CLASS

Specifies the transport class that is to be used for the openFT protocol in the case of incoming connections.

ADAPTER

List of FarSync X.25 adapters, which the openFT protocol attaches to in order to accept incoming connections.

NSAP

NSAP address of the local openFT protocol. If the NSAP is specified as the OSI network address, this is then followed by the individual values for AFI, IDI and DSP.

AFI

Authority and format identifier of the NSAP.

IDI

Initial domain identifier of the NSAP.

DSP

Domain-specific part of the NSAP.

FTAM-APPL

FarSync X.25-specific settings for the FTAM protocol.

USE X.25

Specifies whether the FTAM protocol attaches to the FarSync X.25 transport system in order to use it.

YES The FTAM protocol attaches to the FarSync X.25 transport system.

NO The FTAM protocol does not attach to the FarSync X.25 transport system.

NUM-LISTS

Specifies the number of list() calls per FarSync X.25 adapter by the FTAM protocol.

CLASS

Specifies the transport class that is to be used for the FTAM protocol in the case of incoming connections.

ADAPTER

List of FarSync X.25 adapters, which the FTAM protocol attaches to in order to accept incoming connections.

NSAP

NSAP address of the local FTAM protocol. If the NSAP is specified as the OSI network address, this is then followed by the individual values for AFI, IDI and DSP.

AFI

Authority and format identifier of the NSAP.

IDI

Initial domain identifier of the NSAP.

DSP

Domain-specific part of the NSAP.

Linux:

```
ADAPTER DTE
0      12345
1      54321
2      22222
3      33333
```

OPENFT-APPL

```
USE X.25  NUM-LISTEN  CLASS  ADAPTER
NO        3          0/-   0,1
NSAP     = 43000000000012345678901
AFI      = 43
IDI      = 123
DSP      = 45678901
```

FTAM-APPL

```
USE X.25  NUM-LISTEN  CLASS  ADAPTER
YES       4          2/2   2
NSAP     = 43000000000032110987654
AFI      = 43
```



```

IDI    = 321
DSP    = 10987654

```

The meaning of the parameters under Linux is identical to that under Windows. In contrast to Windows there is not a Line column under Linux in the output of the DTE addresses that are allocated to the lines, because in Linux a separate number (adapter) is allocated to each line.

4.3.3 Command ftshwo – CSV output

The CSV output has been extended to include the following values for the display of the FarSync X.25 parameters:

Example:

Windows:

```

ftshwo -csv
...
X25Config;openftApplUseX25;openftApplX25ListInt;openftApplX25ListNum;
openftApplX25Class;openftApplX25Nsap;ftamApplUseX25;ftamApplX25ListInt;
ftamApplX25ListNum;ftamApplX25Class;ftamApplX25Nsap
"0:0=12345,0:1=54321,1:0=22222,1:1=33333";*NO;"0,1,2";2;"0/-";
"43.123.45678901";*YES;"2";4;"2/2";"43.321.10987654"

```

Linux:

```

ftshwo -csv
...
X25Config;openftApplUseX25;openftApplX25ListInt;openftApplX25ListNum;
openftApplX25Nsap;ftamApplUseX25;ftamApplX25ListInt;ftamApplX25ListNum;
ftamApplX25Nsap
"0=12345,1=54321,2=22222,3=33333";*NO;"0,1,2";2;"43.123.45678901";*
YES;"2";4;"43.321.10987654"

```

The output under Linux is identical to that under Windows. There is only one difference in the columns `openftApplX25ListInt` and `ftamApplX25ListInt`. The difference, however, only exists in the output values, because the lines in Windows and in Linux are referred to differently.

The following table shows the CSV output format for the X.25 operating parameters:

Column	Type	Values and meaning	Parameters
X25Config	String	A list of comma separated values with DTE addresses, which are allocated to the respective lines of the FarSync X.25 card	Table with the DTE addresses per line
openftApplUseX25	String	*YES / *NO Attach of the openFT protocol to the FarSync X.25 transport system	OPENFT-APPL USE X.25
openftApplX25ListInt	String	A list of comma separated values with the adapter numbers, which the openFT protocol is to attach to in order to accept incoming connections	OPENFT-APPL ADAPTER
openftApplX25ListNum	Number	Number of list() calls that the openFT protocol makes per FarSync X.25 adapter in order to accept incoming connections	OPENFT-APPL NUM-LISTS
openftApplX25Class	String	0/- / 2/0 / 2/2 Transport class for incoming connections via the openFT protocol	OPENFT-APPL CLASS
openftApplX25Nsap	String	NSAP Local OSI network address for the openFT protocol in AFI.IDI.DSP format and a hexadecimal string in free format	OPENFT-APPL NSAP
ftamApplUseX25	String	*YES / *NO Attach of the FTAM protocol to the FarSync X.25 transport system	FTAM-APPL USE X.25
ftamApplX25ListInt	String	A list of comma separated values with the adapter numbers, which the FTAM protocol is to attach to in order to accept incoming connections	FTAM-APPL ADAPTER
ftamApplX25ListNum	Number	Number of list() calls that the FTAM protocol makes per FarSync X.25 adapter in order to accept incoming connections	FTAM-APPL NUM-LISTS
ftamApplX25Class	String	0/- / 2/0 / 2/2 Transport class for incoming connections via the FTAM protocol	FTAM-APPL CLASS

ftamApplX25Nsap	String	NSAP Local OSI network address for the FTAM protocol in AFI.IDI.DSP format and a hexadecimal string in free format	FTAM-APPL NSAP
-----------------	--------	---	-------------------

4.3.4 openFT Explorer – Operating parameters

In openFT Explorer the Operating parameters dialog has been extended to include the dialog page **X.25**. This dialog page is only available for openFT instances that support the FarSync X.25 transport system. Furthermore, the FarSync X.25 transport system must be installed on the computer.

Apart from the table of configured FarSync adapters (see section 4.1.2), configuration parameters for the use of the X.25 transport systems are also shown for the openFT and the FTAM protocol. In detail, the following fields are available – separated for the two protocols openFT and FTAM:

Use X.25 transport system

This option enables the use of the X.25 transport system to be activated for the respective protocol. A definition exists as to whether the protocol attaches to the X.25 transport system. Corresponds to the parameters -openftx25 for the openFT protocol and -ftamx25 for the FTAM protocol.

NSAP (Network Service Access Point)

This option should be selected in order to define an NSAP. The NSAP fields described below are only available after the option has been selected.

An NSAP can either be defined in free format or in OSI format.

The definition in OSI format is the default setting.

Corresponds to the parameters -openftx25nsap for the openFT protocol and -ftamx25nsap for the FTAM protocol.

Free Format

Value range: 4 to 40 hexadecimal digits

OSI Format

In OSI format an NSAP should be defined as consisting of an AFI as well as an optional IDI and DSP.

AFI

(Authority and Format Identifier)

Selection list with the supported AFI values

IDI (...)

(Initial Domain Identifier)

The corresponding address format (X.121, ISO_DCC, F.69, E.163, E.164, ISO_ICD or locally) is displayed between the brackets of the field name depending on the set AFI. Value range: depends on the value selected for AFI, see table in section 4.2.1.

DSP

(Domain-Specific Part)

Value range: depends on the value selected for AFI, see table in section 4.2.1.

Accept connections on adapters

List of FarSync X.25 adapters the openFT or FTAM protocol attaches to in order to accept incoming connections.

Value range: adapter numbers 0 .. 3 (Windows) and 0 .. 15 (Linux), comma separated values.

Standard value: 0, i.e. incoming X.25 connections for the respective protocol are accepted on adapter 0.

Corresponds to the parameters -openftx25lif for the openFT protocol and -ftamx25lif for the FTAM protocol.

Number of list calls per adapter

Specifies the number of list() calls per FarSync X.25 adapter by the openFT and FTAM protocol.

Value range: 1 .. 99

Standard value: 5

Corresponds to the parameters -openftx25lnb for the openFT protocol and -ftamx25lnb for the FTAM protocol.

Accept transport connections of class

Selection list with the alternatives **Class 0** or **Class 2**, **Class 0** or **Class 2**.

Default setting: Class 2, Class 0

Corresponds to the parameters -openftx25cl for the openFT protocol and -ftamx25cl for the FTAM protocol. Class 0 stands for the value 0/-. Class 2, Class 0 stands for 2/0. Class 2 stands for 2/2.

4.4 Extension of the ftinfo command

To enable an application to determine whether openFT on the respective operating system supports the FarSync X.25 transport system the output of the ftinfo command has been extended by a column:

Column	Type	Values and meaning
NativeX25	String	*YES / *NO openFT supports the FarSync X.25 transport system directly and FarSync X.25 is installed on the processor

The value *YES is only output for NativeX25 on the operating systems Windows and Linux (x86 and x86_64) if the FarSync X.25 transport system is also actually installed on the computer.

If *NO is output for NativeX25, then no parameters are displayed in the dialogs in openFT Explorer for the FarSync X.25 transport system. However, the commands ftmodo, ftshwo, ftaddptn, ftmodptn and ftshwptn continue to accept the X.25-specific parameters on the operating systems Windows and Linux (x86 and x86_64).

5 Scope ID and line number

The scope ID in IPv6 addresses and the line number (or adapter number:line number) in X.25 address specifications are integrated in the "Host" within the partner address, but are not taken into consideration when determining a partner list entry via the partner address, unless a scope ID or line number is specified in the partner address. This is a change in the previous behavior for IPv6, which is a particular advantage for the inbound resolution of the sender via the partner address.

Example: A partner was entered in the partner list with the following command:

```
ftaddptn ftampart -pa=ftam://%ip6[FE80::222:333:444:555%eth0]
```

If this partner now sends a message, he answers with the sender address
ftam://%ip6[FE80::222:333:444:555]

In previous versions the sender was not able to be allocated to the partner entry ftampart, because the scope ID caused trouble. The partner entry ftampart is now found because its scope ID specification is ignored.

Please note the following:

If a unique allocation is required via the partner address, one and the same IPv6 or DTE address (with the same selectors) and with a different scope ID or line number should not occur twice in partner list entries. If we are dealing with a link-local IPv6 address here, the scope ID in operation without CMX is nevertheless passed up in the sender address as of V12.0B, thus enabling unique partner allocation even with identical link-local addresses plus different scope IDs.

In openFT or FTADM partner entries the scope ID or line number is not transferred to the identification in the case of defaulting. It is also no longer included in the CSV output of the ftshwptn command in the "NetworkAddr" column, but in the "PartnerAddr" column. There is now also a new column "Scopeld" in the CSV output of the ftshwptn command:

Column	Type	Values and meaning	Parameters
Scopeld	String	Scope ID (for IPv6) or line number (on Windows adapter number:line number) (X.25)	Included in the partner address

6 Single-user mode on Unix Platforms

As standard openFT runs on Unix platforms in multi-user mode under a privileged ID (root) and then temporarily changes into the role of a user if openFT functions are to be executed for the former. As of V12.0B it is also possible to let openFT run in single-user mode. This means that openFT runs completely under a specific ID, and also means that access options, the executing of commands, etc. are consequently restricted to the permissions of this ID. This ID is referred to below as **openFT ID**.

The feature as to whether openFT runs in single or multi-user mode can be switched using the openFT command `ftsetmode` and applies for all openFT instances. However, it is possible to define an individual openFT ID for this when creating a new instance in single-user mode.

Other IDs than the openFT ID have no access to functions of the appropriate openFT instance (exceptions: `ftcrei`, `ftdeli`, `ftmodi`). The invoking of openFT commands from unauthorized IDs is rejected with exit code 254. Access to the openFT instance is only possible on the inbound side via FTAC profiles, which belong to the openFT ID and in which no explicit user password is specified. Accesses that specify user ID and user password are not possible.

In single-user mode the openFT ID is at the same time the FT Administrator; merely the creation, deletion and modification of openFT instances are not possible in the openFT ID.

In single-user mode the openFT ID is also the FTAC administrator. Permission profiles for the openFT ID can be administered as required. Existing profiles for other IDs can also be deleted and modified, inasmuch as the modification does not refer to the user ID. In the permission record for the openFT ID the limit values for administrator and user are changed simultaneously if `ftmoda` is invoked.

In single-user mode permission records and permission profiles for external IDs are invalid, but continue to exist and become "active" again upon returning to multi-user mode.

To enable the openFT instance to work as a central administration server in single-user mode the appropriate permission is returned when transferring to single-user mode, providing it had a different ID to the openFT ID. If required, the openFT ID can then give itself this permission (`ftmoda -admpriv=y`).

6.1 Installing openFT

openFT is always installed under root. openFT runs in multi-user mode after a new installation and an update installation. The administrator can then switch openFT to single-user mode using the openFT command `ftsetmode`.

If openFT V12.0B is already installed and openFT is running in single-user mode, single-user mode is retained if openFT V12.0B is installed again.

6.2 Switching to single-user mode

The administrator can use the following command to switch openFT from multi-user mode to single-user mode:

```
/opt/openFT/bin/ftbin/ftsetmode -s -ua=<openFT user ID>
```

In this case the standard instance and all other active instances are allocated to the openFT ID.

The following applies for each active instance:

FT and FTAC administrators are set to the openFT ID. If the openFT ID was ADM administrator in multi-user mode, it also remains this in single-user mode. The ADM administration permission is returned for all other IDs, including `root`.

Note:

If necessary, `ftsetmode` ends the openFT processes of all active instances prior to changing to single-user mode (`ftagt`, `openFT`, `openFT Explorer`, `ftscript jobs`, `ftexec` and `ncopy` commands). However, the `ftsetmode` command should not be invoked during ongoing openFT operation.

The `Ftscript` user options (including the openFT ID) are deleted. The `Ftscript` runs of unauthorized IDs can no longer be accessed using openFT resources.

After the switch openFT is restarted for the instances, for which it had been started before the switch. The `ftagt` process is no longer started (see 6.6).

The start of openFT when booting the system is in single-user mode under the ID of the instance owner.

6.3 Creating and administering additional openFT instances in single-user mode

Additional openFT instances in single-user mode are created and administered using the instance commands.

ftcrei

Also in single-user mode only `root` is permitted to create a new openFT instance with the command `ftcrei`. In this case, the new parameter `-ua` must be specified in order to define the owner of the instance:

```
ftcrei -h | <Instanz 1..8> ... -ua=<user ID 1..32>
```

Likewise, only `root` is also permitted to reactivate an openFT instance with the command `ftcrei`. The parameter `-ua` may not be specified here. In multi-user mode the parameter is generally not permitted. When reactivating an instance in multi-user mode `ftcrei` ensures that `root` is always the owner of the instance. If the owner to date was not `root`, `root` is made the FT and FTAC administrator; and if the previous owner was the ADM administrator, this permission is then transferred to `root`.

ftdeli

Only `root` is permitted to deactivate an openFT instance with the command `ftdeli` in single-user mode.

ftupdi

Only the owner of this instance is permitted to update an openFT instance with `ftupdi` in single-user mode. Even `root` may not update an instance in single-user mode if it does not belong to `root`.

ftmodi

On Unix systems the command `ftmodi` can be used in single-user mode to allocate an openFT instance to another user. For this purpose it was extended to include the parameter `-ua`. This is only available to `root` and only in single-user mode.

```
ftmodi -h | <Instanz 1..8> ... [ -ua=<user ID 1..32> ]
```

The new owner of the instance also becomes its FTAC administrator. If the previous owner of the instance had ADM administrator permission, this is then transferred to the new owner.

Both the changing of the owner of an instance with `ftmodi` and the deactivation of an instance using `ftdeli` should as a matter of principle not be done during ongoing openFT operation. However, an asynchronous openFT server or Ftscript jobs running at the time the command is executed will be stopped.

The Ftscript user options (including those of the openFT ID) are deleted. The Ftscript runs of unauthorized IDs can no longer be accessed using openFT resources.

ftshwi

The long output of the command `ftshwi` (if the switch `-l` is specified) also shows the owner of the instances in single-user mode. In this case, the output looks e.g. like this:

Instance	Owner	Address
-----	-----	-----
aba	aba /var/openFT/aba	host05
std	fts /var/openFT/std	host04

ftseti

In single-user mode a user may only set those instances that belong to him with `ftseti`.

6.4 Returning to multi-user mode

The administrator can use the following command to switch the openFT to multi-user mode:

```
/opt/openFT/bin/ftbin/ftsetmode -m
```

After invoking this command all active openFT instances work in multi-user mode. The following applies for each active instance, including the standard instance: FT and FTAC administrators are set to `root`. If the openFT ID was the ADM administrator in single-user mode, the ADM administration permission is returned.

Note: If necessary, `ftsetmode` ends the openFT activities in all active instances (openFT, openFT Explorer, `ftscript` jobs, `ftexec` and `ncopy` commands). However, the `ftsetmode` command should not be invoked during ongoing openFT operation. After the switch openFT is restarted for the instances, for which it had been started before the switch.

6.5 Returning permission for remote administration

The ADM administrator can use the command `ftmoda` to forward his permission to another ID. As of V12.0B he also has the option of returning the ADM administration permission:

```
ftmoda -admpriv=n
```

All the profiles defined with `-ff=c` remain saved but cannot be used, as they cannot be allocated to a valid ADM administrator. The configuration of the remote administration server is retained.

The FTAC administrator can now define a new ADM administrator. As a result, the FTAC profiles saved for remote administration become active again. They are allocated to the new ADM administrator. It is not necessary to reimport the configuration of the remote administration server.

6.6 SNMP in single-user mode

SNMP is not supported in single-user mode.

6.7 Changes for single-user mode under Solaris

In single-user mode the openFT instances that are enabled in the SMF are started "transiently" for the appropriate openFT ID. "Transiently" means that the openFT instances in single-user mode are not monitored by the SMF.

In single-user mode the openFT ID can start or stop the openFT instance at will with the commands `ftstart` and `ftstop`. The settings in the SMF are unchanged.

Only the system administrator (`root`) can use the known SMF commands to define whether the openFT instance is started (enabled) or not started (disabled) during booting:

- `svcadm disable openFT:<instance>`
- `svcadm enable openFT:<instance>`

When booting the system the settings defined by the system administrator apply.

When switching to multi-user mode `ftstart` and `ftstop` are also used to execute the corresponding `svcadm` command.

6.8 ftalarm in single and multi-user mode

The starting of the monitoring function with `ftalarm -s <number>` must be effected in multi-user mode by the FT administrator and in single-user mode by the owner of the invoking instance. If a switch is made from single-user mode to multi-user mode or vice versa, or if an openFT instance is allocated to another user ID in single-user mode, the `ftalarm` function should in each case be deactivated with `ftalarm -t`, because although it is still suspended in the cron function after the switch, it would however no longer be effective.

`ftalarm -t` can be invoked from any user ID. As a result, it is also possible to remove invalid `ftalarm` entries or those that have become invalid from the cron function.

`ftalarm -i` can be used to test whether the `ftalarm` function is always activated for the invoking user and the set openFT instance. If this is the case, the appropriate crontab entry is output, e.g.

```
* * * * * OPENFTINSTANCE=std /opt/bin/ftalarm -r 47
```

Otherwise nothing is output.

Although `ftalarm` writes entries in the `conslog` file in single-user mode, it nevertheless issues no messages to the console.

7 SNMP under Linux

The SNMP functionality is also supported under Linux (x86) (32 bit) with V12.0B.

8 Release of openFT for 64-bit Linux

A new delivery unit for 64-bit Linux (x86_64) is released as of openFT V12.0B.
openFT runs in 64-bit mode here.

9 Installation on Unix Systems

Update installation:

An update installation is performed if openFT V10, V11 or V12 has already been installed. The logging file is retained in the case of an update from V12. Interrupted ftscript runs from V12 start up again when openFT is started. If the FTAM and / or FTP functionality is installed in openFT V12, then it is also installed in openFT V12.0C after the update installation.

10 Exporting public keys

As of V12.0B the openFT Explorer supports the export of public keys of the local instance. This makes it easier for the FT administrator to provide partners with the keys required to authenticate their own instance.

There is for this purpose the new command **Export Public Key...** in the submenu **Key Management** of the **Administration** menu. The command opens the dialog **Export Public Key**.

For each of the (max. 3) existing sets of key pairs the dialog shows the appropriate public keys with bit lengths of 768, 1024 and 2048. Just one of these (max. 9) keys can be selected for export. Furthermore, the file to which the required key is to be exported must be defined in the **Output File** field. A file in the home directory with the name of the instance ID is provided as default.

Click the **Export** button to copy the file with the selected public key to the place defined in **Output File**.

11 Delete diagnosis information and console messages

As of V12.0B the openFT Explorer supports the deletion of diagnosis information and console messages of the set local instance.

There are two commands **Show Diagnosis Information** (as before) and **Delete Diagnosis Information** in the submenu **Diagnosis Information** of the **Administration** menu in order to show and delete diagnosis information. Both commands are only available to the FT administrator and can only be executed if at least one diagnosis record exists. The **Delete Diagnosis Information** command deletes – if configured after confirmation – the file DIAGFILE with the diagnosis records from the instance directory.

There are two commands **Show Console Messages** (as before) and **Delete Console Messages** in the submenu **Console Messages** of the **Administration** menu in order to show and delete console messages. Both commands are only available to the FT administrator and can only be executed if at least one console message exists. The **Delete Console Messages** command deletes – if configured after confirmation – the file conslog with the console messages from the instance directory.

12 Configuration of minimum RSA key length for openFT protocol

As of 12.0C it is possible for an openFT instance to require a minimum RSA key length for the negotiation of the openFT session encryption key. If one of the acting instances requires a minimum key length by configuration, the openFT protocol will assure, that the AES key negotiation uses RSA encryption of at the minimum this configured length.

When one of the partners has either no valid RSA key or has switched off encryption using the `-kl / RSA-PROPOSED` option while the other communication partner requires a minimum key length, no session connection between the two partners will be possible. This also applies to the `ftping` command which will only be able to verify the transport connection between the partners.

As of V12.0C it is possible to configure a minimum RSA key length in the openFT options. Therefore a new parameter was inserted to the commands, `ftmodo / MODIFY-FT-OPTIONS`.

The configured key length for RSA proposal must be greater or equal to the configured minimum key length. If any of these parameters is being modified and this constraint is not observed, a warning will be issued and the proposal key length will be adapted to the minimum key length.

12.1 Command syntax for Unix / Windows / POSIX:

```
ftmodo -h |
  [ -kl=0 | -kl=768 | -kl=1024 | -kl=2048 ]
  [ -klmin=0 | -klmin=768 | -klmin=1024 | -klmin=2048 ]
  ...
```

Parameter:

```
[ -klmin=0 ]           No minimum key length is configured. Any key length and
                       even requests without encryption will be accepted.

[ -klmin=768 | 1024 | 2048 ]   Only keys of the specified length or
                               larger ones will be accepted. If the initiator uses a
                               key of a lower length there will be a counter proposal
                               by the responder of the session.
                               Sessions without encryption will not be accepted.
```

12.2 Command syntax for BS2000 / z/OS:

```
/MODIFY-FT-OPTIONS / FTMODEOPT
...
KEY-LENGTH = *PARAMETERS()
             RSA-PROPOSED = *UNCHANGED / 0 / 768 / 1024 / 2048
             ,RSA-MINIMUM = *UNCHANGED / 0 / 768 / 1024 / 2048
...
```

KEY-LENGTH

Configuration of the length of RSA keys used for the negotiation of the AES session key

RSA-PROPOSED

Length of the RSA key used for encryption. This key is used only to encrypt the AES key which is agreed on between the partners (or the DES key up to and including openFT V7.0). openFT uses the AES key to encrypt the request description data and possibly also the file contents.

Default setting after new installation: 2048

RSA-PROPOSED = 0

Switch off encryption

RSA-PROPOSED = 768 / 1024 / 2048

Length (in bit) of the RSA key used for the negotiation of the AES session key.

RSA-MINIMUM

Minimum length (in bit) of RSA key that will be accepted from a communication partner for the negotiation of the AES session key.

Default setting after new installation: 0

RSA-MINIMUM = 0

No minimum key length is configured. Any key length and even requests without encryption will be accepted.

RSA-MINIMUM = 768 / 1024 / 2048

Only keys of the specified length or larger ones will be accepted. If the initiator uses a key of a lower length there will be a counter proposal by the responder of the session. Sessions without encryption will not be accepted.

12.3 Output of the ftshwo / SHOW-FT-OPTIONS command:

All encryption parameters were moved to a separate line in the standard output of ftshwo.

```
ftshwo
STARTED PROC-LIM CONN-LIM ADM-CLIM RQ-LIM MAX-RQ-LIFE TU-SIZE CCS-NAME
YES NONE 16 64 2000 30 65535 ISO88591
PTN-CHK DYN-PART SEC-LEV FTAC-LOG FT-LOG ADM-LOG USE TNS USE CMX
STD ON B-P-ATTR ALL ALL ALL NO NO
OPENFT-APPL FTAM-APPL FTP-PORT ADM-PORT ADM-CS
*STD *STD 21 11000 YES
ACTIVE ACTIVE DISABLED ACTIVE
RSA-PROP RSA-MIN ENC-MAND
2048 0 NO
HOST-NAME IDENTIFICATION / LOCAL SYSTEM NAME
*NONE unixsystem / $FJAM,unixsystem
DEL-LOG ON AT RETPD RECOVERY ADM-TRAP-SERVER
OFF DAILY 00:00 14 IN+OUT *NONE
TRAP: SS-STATE FT-STATE PART-STATE PART-UNREA RQ-STATE TRANS-SUCC TRANS-FAIL
CONS OFF OFF OFF OFF OFF OFF OFF OFF
ADM OFF OFF OFF OFF OFF OFF OFF OFF
FUNCT: SWITCH PARTNER-SELECTION REQUEST-SELECTION OPTIONS OPTIONS-LL
MONITOR OFF ALL ALL ALL
TRACE OFF ALL ALL NONE OFF
```

CSV-output:

There was a new column inserted for the output of RSA key minimum:

Column	Type	Value
RSAMinimum	Number	0 / 768 / 1024 / 2048

OPS output:

Element	Type	Value
RSA-MINIMUM	Integer	0 / 768 / 1024 / 2048

12.4 openFT GUI support :

The new frame 'RSA Min Key Length' is inserted in the Operating Parameters dialog window for the configuration of minimum RSA key length. The frame comprises of four radio buttons representing the key length of **0**, **768**, **1024** and **2048**.

If minimum key length of '0' is selected then no minimum key length is configured. Any key length and even requests without encryption will be accepted.

If RSA Minimum Key Length of '768/1024/2048' is selected then RSA Key Length of the equal length and larger key values will be enabled for the selection. The RSA Key Length with lesser values than RSA Minimum Key Length are disabled from the selection.

The configured key length for RSA Key Length must be greater or equal to the configured minimum key length. If configured RSA Minimum Key Length is greater than RSA Key Length, then RSA Key Length value will be adapted to the RSA Minimum Key Length value, and warning message "The RSA key length was aligned to the minimum key length" will be displayed to the user during submission.

This new frame is disabled for openFT partners with versions lower than V12.0C.

12.5 openFT z/OS panels support :

The new parameter 'RSA-MINIMUM' has been inserted into the panels for the configuration of minimum RSA key length.

The permissible values for RSA-MINIMUM parameter are **0**, **768**, **1024** and **2048**.

13 ftmonitor polling time change

As of openFT V12.0C the graphic openFT monitor supports the change to another polling interval without restarting the program and while retaining the window, view and, where applicable, choice of measured value. However, the charts are recreated and the previously obtained values are no longer displayed. If a partner instance is monitored then the connection to the partner that was established for this purpose is closed and a new connection is opened.

In order to set a different polling interval from the openFT monitor click on View->Change Polling Interval in the main menu. A submenu with various possible values for the polling interval appears, from which the user can select the desired value. If the value which the user requested is not in the selection list, select the bottom entry of the selection list "Other value..." A dialog, in which the user can enter the desired value, then opens.

After brief reinitialization, entry of measured values restart with the changed polling interval.

14 Extended ftinfo output

The output of the command ftinfo is extended by the column Crypt. If openFT-CR is installed, which means that the user data can be transferred encrypted, this column contains the value *YES, otherwise *NO.

```
ftinfo  
  
CmdUiVer;CmdTiVer;OsType;UserId;IsFtAdm;IsFtacAdm;FtLang;CcsName;Home;Li  
mited;IsAdmAdm;ProdVer;SrcVer;Inst;TimeOffset;FtScriptDir;NativeX25;Sing  
leUser;Crypt  
1202;100;"Unix";"root";1;1;"en";"ISO88591";"/";*NO;0;"12.0C00";"311";"st  
d";3600;"/";*NO;*NO;*YES
```

In the openFT explorer this information is displayed in the dialog "About openFT Explorer" which is an item of the Help menu.

15 Case extension to the tool ft_mget

The tool ft_mget is extended to include the option `-case`; this sets the consideration or non-consideration of upper case / lower case in the file name pattern.

```
ft_mget -h |
        [ -async ]
        [ -case=y | -case=n ]
        ...
```

Description

<code>-case=y</code>	The file name pattern specified for the source file is compared with the file names received from the remote system with due regard to upper case / lower case.
<code>-case=n</code>	The file name pattern specified for the source file is compared with the file names received from the remote system without regard to upper case / lower case.
Default:	If <code>-case</code> is not specified, the platform of the remote system is determined as before and whether upper case / lower case for file names should be considered is determined as a result.

The option `-case` does not influence determination of the directory name.

16 Extension of logging functionality to the 'Logging' node

As of openFT V12.0B, the logging functionalities such as Switch Logging File, Determine Logging Extent and Automatic Deletion of Logging Records are available via the 'Operating Parameters' dialog, tab 'Logging'.

In openFT V12.0C, the logging functionalities are extended to the context menu of the node 'Logging' by a menu entry **Settings...** allowing to open the 'Operating Parameters' dialog and position on its 'Logging' tab automatically.

17 openFT Explorer recognizes newly created openFT instances automatically

As of openFT V12.0C the drop down menu for choosing an instance is always available. Newly created instances are added to the menu without restarting openFT Explorer.

18 Unattended installation of openFT on Windows Platforms

As of openFT V12.0C00 the unattended installation has been changed as follows:

- Packages openFTde.mst and PCMX-32de.mst are no longer delivered
- For installing the German language use TRANSFORMS=:de
- Default language: If German or English is set as locale in operating system this language is taken as default language for openFT.

Examples:

1. You start unattended installation of CMX without user interaction with

```
msiexec /i PCMX-32.msi /qn
```

2. You start unattended installation of the German language version of openFT (without user interaction) in the default directory %Program Files%\openFT with

```
msiexec /i openFT.msi TRANSFORMS=:de /qn
```

3. Enter the following command to start unattended installation without user interaction of the German language version of openFT including SNMP in the default directory:

```
msiexec /i openFT.msi ADDLOCAL=SNMP TRANSFORMS=:de /qn
```

4. For the unattended installation, without user interaction, of openFT with SNMP in German in the default directory including specification of the openFT and optional FTAM license keys, enter the following command:

```
msiexec /i openFT.msi ADDLOCAL=SNMP  
LICENSEKEY=12345-12345-12345-12345-12345  
FTAMLICENSEKEY=54321-54321-54321-54321-54321  
TRANSFORMS=:de /qn
```