English

Fujitsu Software

openFT (Unix- and Windows-Systeme)

Command Interface

User Guide

Valid for:
openFT V12.1C80

November 2024

Comments… Suggestions… Corrections…

The User Documentation Department would like to know your opinion on this manual. Your feedback helps us to optimize our documentation to suit your individual needs.

Feel free to send us your comments by e-mail to: bs2000services@fujitsu.com.

Certified documentation according to DIN EN ISO 9001:2015

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2015.

Copyright and Trademarks

Table of Contents

Command Interface (Unix and Windows Systems)

1 Preface

The openFT product range transfers and manages files

- automatically,
- securely, and
- cost-effectively.

The reliable and user-friendly transfer of files is an important function in a high-performance computer network. The corporate topologies consist of networked PC workstations, which are usually additionally linked to a mainframe or Unix based server or Windows server. This allows much of the processing power to be provided directly at the workstation, while file transfer moves the data to the mainframe for further processing there as required. In such landscapes, the locations of the individual systems may be quite far apart. Fujitsu offers an extensive range of file transfer products - the openFT product range - for the following system platforms:

- BS2000®
- Linux® (Intel x86_64)
- Microsoft® Windows™ 10, Windows Server 2016 and 2019
- z/OS (IBM®)

1.1 Brief description of the product

The openFT product range comprises the following products:

FUJITSU Software openFT (Unix systems) is the file transfer product for systems with a Unix based operating system.

FUJITSU Software openFT (Windows) is the file transfer product for Microsoft's Windows systems.

FUJITSU Software openFT (BS2000) is the file transfer product for computers using the operating system BS2000.

FUJITSU Software openFT (z/OS) is the file transfer product for computers using the operating system z/OS.

The manual applies to the file transfer products openFT(Unix systems) and openFT(Windows).

All openFT products communicate with each other using the openFT protocol (previously only known as FTNEA) as laid down by Fujitsu. Since a number of FT products from other software vendors also support these protocols, many interconnection options are available.

openFT allows the use of the following transport protocols:

- TCP/IP
- ISO TP0/2 (not on z/OS)
- ISO TP4 (not on z/OS)
- SNA (only on z/OS)

The range of functions made available by openFT can be extended by:

- FTAC:

  FTAC provides extended system and data access protection. FTAC stands for File Transfer Access Control.
  On BS2000 systems, FTAC is provided by the add-on product openFT-AC.
  On z/OS, FTAC is provided by the add-on product openFT-AC.
  On BS2000 systems and on z/OS, FTAC is provided by the add-on product openFT-AC.
  On Unix and Windows systems, FTAC is integrated in openFT.
- openFT-FTAM (not available on z/OS):

  openFT supports the FTAM file transfer protocol (File Transfer Access and Management) standardized by ISO (International Organization for Standardization). This makes it possible to interconnect with even more systems from other vendors whose file transfer products support the same standard.
- openFT-FTP:

  openFT also supports the FTP functionality. This makes it possible to interconnect with other FTP servers.
- openFT-CR:
  openFT-CR is required up to version V12.1B if encrypted file transfer is requested.

> **Attention**
> As of openFT version V12.1C, openFT-CR is no longer delivered because the functionality has been integrated in openFT V12.1C.
>
> All references to openFT-CR therefore only apply to openFT versions <= V12.1B.

1.2 Target group

This manual is intended for those who want to use the command openFT interface on a Unix or Windows system in order to transfer files and to administer openFT.

The manual covers Linux systems and Oracle Solaris systems as well as porting to other Unix platforms such as AIX or HP-UX.

> i As of openFT version V12.1B, the release is only for Linux x86_64. Since openFT version V12.1C20, Solaris (Sparc) is also included in the release again. Further platforms are available on request from your Fujitsu sales representative.

The operating system-dependent differences are described in detail in the Release Notices supplied on the internet and the respective product CD.

1.3 Concept of openFT manuals

openFT - Concepts and Functions

This manual is intended for those who want to get familiar with the capabilities of openFT and want to understand the openFT functions. It describes:

- the concept of openFT as a Managed File Transfer
- the scope of work and main features of the openFT product family
- the openFT-specific terms

openFT (Unix and Windows Systems) - Installation and Operation

This manual is intended for the FT, FTAC and ADM administrator on Unix and Windows systems. It describes:

- how to install openFT and its optional components
- how to operate, control and monitor the FT system and the FTAC environment
- the configuration and operation of a remote administration server and a ADM trap Server

openFT (BS2000) - Installation and Operation

This manual is intended for the FT and FTAC administrator on BS2000 systems. It describes:

- how to install openFT and its optional components on the BS2000 system
- how to operate, control and monitor the FT system and the FTAC environment
- the accounting records

openFT (z/OS) - Installation and Operation

This manual is intended for the FT and FTAC administrator on z/OS. It describes:

- how to install openFT and its optional components, including the requirements for using the product
- how to operate, control and monitor the FT system and the FTAC environment
- the openFT and openFT-AC messages for the FT administrator
- additional sources of information for the FT administrator, such as the accounting records and the logging information

openFT (Unix and Windows Systems) - Command Interface

This manual is intended for the openFT users on Unix and Windows systems and describes:

- the conventions for file transfer to computers with different operating systems
- the openFT commands on Unix and Windows systems
- the messages of the various components

The description of the openFT commands also applies to the POSIX interface on BS2000 systems.

openFT (BS2000) - Command Interface

This manual is intended for the openFT users on BS2000 systems and describes:

- the conventions for file transfer to computers with different operating systems

- the openFT commands on BS2000 systems
- the messages of the various components

openFT (z/OS) - Command Interface

This manual is intended for the openFT users on z/OS systems and describes:

- the conventions for file transfer to computers with different operating systems
- the openFT commands on z/OS
- the menu interface for the FT administrator and the FT user
- the program interface for the FT user
- the messages of the various components

openFT (BS2000) - Program Interface

This manual is intended for the openFT programmer and describes the openFT and openFT-AC program interfaces on BS2000 systems.

openFT (Unix and Windows Systems) - C and Java Program Interface

This manual is intended for C and Java programmers on Unix and Windows systems. It describes the C program interface and the main features of the Java interface.

openFT (Unix and Windows Systems) - openFT-Script Interface

This manual is intended for XML programmers and describes the XML statements for the openFT-Script interface.

> **i** Many of the functions described in the manuals can also be executed via the openFT graphical interface, the openFT Explorer. The openFT Explorer is available on Unix systems and Windows systems. You can use the openFT Explorer to operate, control and monitor the FT system and the FTAC environment of remote openFT installations on any system platform independent from the local system, A detailed online help system that describes the operation of all the dialogs is supplied together with the openFT Explorer.

1.4 Changes since the last version

This section describes the changes in openFT V12.1 compared to openFT V12.0A.

> The functional extensions to the openFT commands, whether they relate to administrators or users, are also available in the openFT Explorer which is provided on Unix and Windows systems. For details, see the *New functions* section in the associated online help system.
> On z/OS, the functional extensions are also available in the menu system (panels).

1.4.1 Changes for all platforms

- Extended Unicode support

  On all Unicode capable systems, file names, FTAC transfer admissions and follow-up processing may consist of Unicode characters. To permit this, the function "Encoding Mode" has been introduced in order to represent the Unicode names correctly on all involved systems.

  The command interfaces have been extended as follows:

  - All platforms:

    The new field FNC-MODE in the long output of log records displays the encoding mode for the file name (commands *ftshwl*, SHOW-FT-LOGGING-RECORDS and FTSHWLOG). On BS2000 systems, the OPS variables have been extended by the elements FNC-MODE and FNCCS.
  - BS2000 and z/OS systems (from openFT V12.1C): the new option FILE-NAME-ENCODING supports different character encoding for the specification of remote file names, pre, post and follow-up processing for transfer commands for the following commands:
    - TRANSFER-FILE-SYNCHRONOUS / FTSCOPY
    - TRANSFER-FILE / FTACOPY
    - CREATE-REMOTE-DIR / FTCREDIR
    - DELETE-REMOTE-DIR / FTDELDIR
    - MODIFY-REMOTE-DIR-ATTRIBUTES / FTMODDIR
    - MODIFY-REMOTE-FILE-ATTRIBUTES / FTMOD
    - SHOW-REMOTE-FILE-ATTRIBUTES / FTSHW
    - DELETE-REMOTE-FILE / FTDEL
    - EXECUTE-REMOTE-CMD / FTEXEC
    - EXECUTE-REMOTE-FTADM-CMD / FTADM.
  - Unix systems and Windows systems:
    - New option *-fnc* in order to set the encoding mode in a file transfer, file management or administration request. This option is available for the commands *ft*, *ftadm*, *ftcredir*, *ftdel*, *ftdeldir*, *ftexec*, *ftmod*, *ftmoddir*, *ftshw* and *nopy*.The encoding mode is displayed in the output of the following commands (in addition to *ftshwl*): *ftshw* and *ftshwr* (FNC-MODE field).
      The number of not mapped file names is displayed using *ftshw -sif*.
    - New attribute *CmdMode* in the configuration of remote administration server to define the (recommended) encoding mode for administered openFT instances. The encoding mode is displayed in the output of the *ftshwc* command (MODE field).
      This function is also available in the configuration editor of the openFT Explorer.
    - In Unix systems, it is also possible to set the character set which is to be used for inbound requests in character mode. To do this, the new option *-fnccs* in the *ftmodo* command has been introduced.

      The character set which is currently set for inbound requests in character mode is displayed in *ftshwo*, FN-CCS-NAME field.
    - Unicode for C and JAVA

- BS2000 and z/OS systems: for library elements the log records display the element name, type and version in addition to the library name .

- For inbound requests, the long output and CSV output of log records display the address of the partner system in the new field PTNR-ADDR. On BS2000 systems, the partner address is also displayed in the OPS variable PARTNER-ADDRESS.
- Deactivation of the restart functions

  The restart function can be deactivated for asynchronous file transfer requests via the openFT or FTAM protocol. The restart can be set partner-specifically for outbound requests and globally for inbound and outbound requests. To permit this, the following commands have been modified:

  Unix and Windows systems:

  - *ftaddptn* and *ftmodptn*: New option *-rco*
  - *ftmodo*: New options *-rco* and *-rci*

  BS2000 and z/OS systems:

  - ADD-FT-PARTNER/MODIFY-FT-PARTNER and FTADDPTN/FTMODPTN:
    New operand RECOVERY-OUTBOUND
  - MODIFY-FT-OPTIONS and FTMODOPT:
    New operands RECOVERY-OUTBOUND and RECOVERY-INBOUND

- Minimum RSA key length for openFT protocol

  An openFT instance can require a minimum RSA key length for the openFT session encryption. The minimum RSA key length can be defined in the operating parameters. To permit this, the following commands have been modified:

  Unix and Windows systems:

  - *ftmodo*: New option *-klmin*

  BS2000 and z/OS systems:

  - MODIFY-FT-OPTIONS and FTMODOPT: New parameters RSA-PROPOSED and RSA-MINIMUM for the KEY-LENGTH operand.

- Minimum AES key length for openFT protocol

  An openFT instance can require a minimum AES key length for the openFT session encryption. The minimum AES key length can be defined in the operating parameters. To permit this, the following commands have been modified:

  Unix and Windows systems:

  - *ftmodo*: New option *-aesmin*

  BS2000 and z/OS systems:

  - MODIFY-FT-OPTIONS and FTMODOPT: New parameter AES-MINIMUM for the KEY-LENGTH operand.

- Encryption for file management requests

  For file management requests the file and directory list attributes can be transferred encrypted..This property can also set in admission profiles.

  To permit this, the following commands have been modified:

  Unix and Windows systems:

  - *ftshw*: New option *-c*
  - *ftcrep* and *ftmodp*: New option *-cm*
  - *ftshwp*: New parameter FILE-AT-ENC in the long output and new parameter *FileAtEnc* in the csv output.

  BS2000 and z/OS systems:

  - SHOW-REMOTE-FILE -ATTRIBUTES and FTSHW:
    New operand FILE-ATTR-ENCRYPTION
  - CREATE-FT-PROFILE and MODIFY-FT-PROFILE as well as FTCREPF and FTMODPRF: New operand FILE-ATTR-ENCRYPTION
  - SHOW-FT-PROFILE and FTSHWPRF: New parameter FILE-AT-ENC in the long output and new parameter *FileAtEnc* in the csv output.

  For getting multiple files form the remote system using ft_mget, GET-REMOTE-FILES or FTMGET the encryption option (-c or DATE-ENCRPTION) also applies to file and directory list attributes.
- From openFT V12.1C the CRYPT functionallity is integrated in openFT. Therefore openFT-CR is no longer delivered.

Changes in openFT version V12.1C10:

- For local outbound requests, the long output and CSV output of log records display the remote filename in the new field REMOTE-FN. On BS2000 systems, the remote filename is also displayed in the OPS variable F-REM-NAME.

Extensions from openFT version V12.1C70:

- Extensions for adding and modifying an openFT partner entry (FTADDPTN/FTMODPTN and FTSHWPTN) on BS2000 system, Unix and Windows platforms, but not (!) on z/OS system. Namely the partner-specific specification of an RSA key whose length is always selected for this partner when connecting to this partner (RSA-PROP), and what minimum RSA key length is expected from this partner (RSA-MIN) when a connection comes in from it. Both specifications have priority over the global setting, visible in the FTSHWOPT command output.

1.4.2 Changes for Unix and Windows platforms

- Transferring directories:
  - Directories can be transferred between Unix and Windows systems. To permit this, the commands *ft* and *ncopy* have been extended with the option *-d*.
  - The new field PROGRESS in the output of the *ftshwr* command displays the progress of (asynchronous) directory transfer.
  - The new option *ftmodo -ltd* has been introduced to set the logging scope for directory transfer.
  - The new value *ftshwl -ff=T* selects log records of directory transfer requests. In addition, the *ftshwl* output has been extended to the field TRANSFILE (long output) as well as the FT function values TD, SD, SF (short output) and the value FUNCTION=TRANSFER-DIR (long output).
- Transferring multiple files via FTAM:

  Multiple files can be transferred synchronously between Unix and Windows systems using the FTAM protocol. This is controlled by a specific file name syntax of the *ncopy* command.
- Extension of the openFT-Script commands
  - The FT administrator can set limits of openFT requests. To permit this, the command *ftmodsuo* has been extended to the options *-u*, *-thl* and *-ftl*.
  - *ftshwsuo* displays the limits currently set.
- The *ftshwk* command displays the partner name for public keys of partner systems.
- FarSync X25 support

  FarSync X.25 cards from the manufacturer FarSite are directly supported by openFT on Linux and Windows systems. PCMX is no longer required for this. The connection method XOT (X.25 via TCP/IP) is also supported on Linux by using the FarSync XOT Runtime.
  To permit this, the commands *ftaddptn*, *ftmodptn*, *ftmodo*, *ftshwptn* and *ftshwo* have been extended.
- Extended support of the Application Entity Title

  The Application Entity Title (AET) now can be used for checking the partner address of FTAM partners. To permit this, the *ftmodo* command has been modified by extending the *-ptc* (partner check) option and adding the *-aet* option for specifying the AET. The *ftshwo* command has been extended by the *-ae* option.
- The maximal length of the command string in the *ftexec* command has been raised to 8191.
- Other changes
  - Modified partner checking for partners which are addressed via IPv6 with scope ID or via X.25 with line number. By this, a unique identification via the partner address is always possible.
  - The *ft_mget* command has been extended by the *-case* option which controls the consideration of the upper case / lower case in the file name pattern.
  - The ADM administrator now can return the permission for remote administration (*ftmoda -admpriv=n*). The configuration of the remote administration server is retained.

Changes in openFT version V12.1C20:

- Support of RSA keys with length 3072 and 4096.

Changes in openFT version V12.1C30:

- On Windows systems the openFT administrator rights can be transferred to any user.
  The *ftmodo* command has been extended for this purpose.

Changes in openFT version V12.1C80:

- The *ftshwlic* command shows the remaining days of the demo period if no license is available.
- So far until openFT 12.1C70 there was only one openFT administrator (as user or group of users) for whole openFT. Starting from version 12.1C80, every instance will have separate openFT administrator.

  The administrator of the STD instance will be at the same time global administrator, who can handle global openFT settings.
- Since openFT 12.1C80, it will be possible to back-up the entire openFT configuration with “ftbackup” command and then restore it with the “ftrestore” command. Configuration for all instances and all users will be saved, excluding licenses.

1.4.3 Changes for Unix platforms

- Single-user mode

  On Unix systems, the administrator can switch between the multi-user mode (default) and the single-user mode using the *ftsetmode* command. In single-user mode openFT runs completely under a specific user ID (the so called openFT ID) which is also FT and FTAC administrator. To permit creating and administering additional openFT instances in single-user mode, the commands *ftcrei* and *ftmodi* have been extended by the option *-ua* for specifying the user ID of a new instance.
- SNMP is no longer supported on Unix platforms.
- Support of systemd.

Changes in openFT version V12.1C10:

- Support of Openshift.

Changes in openFT version V12.1C30:

- Installation using license keys.

Changes in openFT version V12.1C40:

- On Unix systems, as already in Windows systems, the FT administrator rights can be transferred to any user.
  The *ftmodo* command has been extended for this purpose.

Changes in openFT version V12.1C60:

- The FT administrator now can assign a Linux group as FT administrator, every user in the group will have FT administrator rights.
  The *ftmodo* command has been extended for this purpose.

1.4.4 Changes for BS2000 systems and z/OS

- New commands GET-REMOTE-FILES (BS2000 systems) and FTMGET (z/OS) for synchronous or asynchronous fetching of multiple files specified by wildcards from a remote system.
- New diagnostics command FTPING on BS2000-POSIX and z/OS for testing the openFT connection to a remote partner.

Changes in openFT version V12.1C60:

- Support of RSA keys with length 3072 and 4096 bit.

1.4.5 Changes for z/OS

- The PARM member of the z/OS parameter file has been changed as follows:
  - New key word JOB_JOBCLASS for follow-up processing jobs, preprocessing jobs, postprocessing jobs and print jobs.
  - New key word LISTPARM for setting of a default printer (LISTING=*STD in a FT request).
  - The key word JOB_MSGCLASS now applies to preprocessing jobs and postprocessing jobs.
- For FJBATCH in z/OS as of V2.1, you can use the PARMDD parameter instead for the PARM parameter.
- NCOPY and FTACOPY: New value LISTING=*STD in LOCAL-PARAMETER in order to assign a printer defined via LISTPARM.
- openFT (z/OS) is now supporting host names with up to 80 characters in length. This applies both to the internal communication in z/OS and to connections to z/OS partners.
- The member TNCTCPIP of the z/OS parameter file is no longer supported, therefore the description has been dropped.

Changes in openFT version V12.1C30:

- The PARM member of the z/OS parameter file has been changed as follows:
  - New key word TZSTRING for support of timezone handling to avoid missing lines when reading logging records that were written during the change from daylight saving time to winter time.

1.4.6 New functions that are only available in the openFT Explorer

- Exporting public keys

  The FT administrator can export public keys of the local openFT instance using the *Key Management - Export Public Key* command in the *Administration* menu.

- Deleting diagnosis information and console messages

  The FT administrator can delete diagnosis information and console messages using the commands *Delete Diagnosis Information* and *Delete Console Messages* in the *Administration* menu.

- The logging is also available in the object tree of the openFT Explorer.

Please refer to the online help for more details.

1.5 Notational conventions

The following notational conventions are used throughout this manual:

`typewriter font`

: typewriter font is used to identify entries and examples.

italics

: In running text, names, variables and values are indicated by italic letters, e.g. file names, instance names, menus, commands and command options.

indicates notes.

Indicates warnings.

Additional conventions are used for the command descriptions, see section “Entering command”.

1.6 Internet

Current information on the Internet

Current information on the openFT family of products can be found in the internet under http://www.fujitsu.com/ts/openFT.

2 Introduction to the command interface

This chapter gives an overview of using the command interface on Unix and Windows systems and at the POSIX interface on BS2000 systems:

- Overview of the commands
- Entering commands
- Specifying partner addresses
- Entering the authorization data for the partner system
- Preprocessing and postprocessing
- Commands for follow-up processing
- Instance identification
- Output in CSV format
- Notes on FTP partners

2.1 Overview of the commands

The following overview shows a list of all commands according to the various tasks.

2.1.1 Commands for all systems

Administer openFT	
ftstart	Start asynchronous openFT server
ftstop	Stop asynchronous openFT server
ftshwo	Display operating parameters
ftmodo	Modify operating parameters
ftshwd	Display diagnostic information
fttrace	Evaluate trace files
Administer partners	
ftaddptn	Enter a partner in the partner list
ftshwptn	Display partner properties
ftmodptn	Modify partner properties
ftremptn	Remove a partner from the partner list
Administer key pair sets for authentication	
ftcrek	Create key pair set
ftimpk	Import keys
ftshwk	Show key properties
ftmodk	Modify keys
ftupdk	Update public keys
ftdelk	Delete key pair set
Remote administration and ADM traps	
ftadm	Enter a remote administration command
ftshwc	Display remote administrable openFT instances
ftshwatp	Display ADM traps
ftexpc	Export configuration of the remote administration server
ftimpc	Import configuration of the remote administration server
File transfer and request queue managing	

ncopy / ftscopy	Issue synchronous file transfer request
ft / ftacopy	Issue asynchronous file transfer request
ftcanr	Cancel asynchronous file transfer requests
ftalarm	Report failed requests
ftmodr	Change the order of the requests in the request queue
ftshwr	Display the properties and status of requests
Remote command execution	
ftexec	Execute operating system commands in remote system
File management	
ftcredir	Create remote directories
ftshw	Display attributes of a file / a directory in the remote system
ftshwf	Display the FTAM attributes of a local file
ftmod	Modify file attributes in a remote system
ftmoddir	Modify the attributes of remote directories
ftmodf	Modify the FTAM attributes of a local file
ftdel	Delete a file in a remote system
ftdeldir	Delete remote directories
Logging	
ftshwl	Display log records or log files
ftdell	Delete log records or log files
fthelp	Display information on the reason codes in the log records
FTAC function	
ftcrep	Create FT profile
ftshwp	Display FT profile
ftmodp	Modify FT profile
ftdelp	Delete FT profile
ftshwa	Display admission set

ftmoda	Modify admission set
ftexpe	Export FT profiles and admission sets
ftshwe	Display FT profiles and admission sets from a file
ftimpe	Import FT profiles and admission sets
Administer instances	
ftseti	Set an instance
ftshwi	Output information on instances
ftmodi	Modify an instance
ftupdi	Update the instance directory
ftdeli	Deactivate an instance
Display measurement data	
ftshwm	Display measurement data of the openFT operation
ftmonitor	Display measurement data of the openFT operation on openFT Monitor
Output of general information and miscellaneous commands	
ftinfo	Output information about the openFT system
ftshwo	Display operating parameters
ftshwptn	Display partner properties
ftedit	Load local or remote files in the openFT editor
ftmsg	Output message box on a graphical display
openFT	Start openFT Explorer

As the **administrator**, you may execute the commands listed below with the additional options to perform the corresponding action **system-wide**. This means that:

You can use *ftcanr* to delete any desired file transfer requests.

You can use *ftcrep* to create FT profiles for any login names

You can use *ftdelp* to delete any FT profiles.

You can use *ftmoda* to modify and privilege any of the admission sets.

You can use *ftmodp* to modify any of the FT profiles.

You can use *ftmodr* to change the order of all requests in the request queue independent of the login name.

You can use *ftshwa* to display any of the admission sets.

You can use *ftshwl* to display any of the log records.

You can use *ftshwp* to display any of the FT profiles.

You can use *ftshwr* to obtain information about all the requests for all user IDs.

The following commands are valid for Windows and Unix systems as of openFT version V12.1C30:

License key administration	
ftaddlic	Add license keys
ftremlic	Remove license keys
ftshwlic	Show license keys

The following commands are valid for Windows and Unix systems as of openFT version V12.1C80:

Back up and restore the entire openFT configuration	
ftbackup	Backing up the openFT configuration
ftrestore	Restoring the openFT configuration

2.1.2 Specific commands for Unix systems

> As of openFT Version V12.1C30, install.ftam and install.ftp do not exist anymore, because they are replaced with the license system. Running install.ftp and install.ftam on openFT V12.1C30 or later will have no effect.

ftlang	Change default language setting
ftsetjava	Manage link to the Java executable
install.ftam	Install/deinstall openFT-FTAM
install.ftp	Install/deinstall openFT-FTP
ftsetmode	Switch between multi-user and single-user operation

2.1.3 Specific commands for Windows systems

ftsetpwd	Store user password

2.2 Entering commands

The command syntax essentially corresponds to the output that you get when you specify the command with *-h* option.

Notational conventions

< >	angle brackets are used for parameters which you may replace with current values. You must not specify the angle brackets < > and the permissible value ranges.
[ ]	enclose optional entries. The effect on the function of the command is described for the individual parameters.
|	stands for alternatives. You may specify only one of the values indicated.
Bold typeface	This is used in the "Description" sections for individual characters or strings that must be specified in exactly the form given, e.g. options or values. In running text, these are then shown in *italics*.

Special characters and blanks

If you enter special characters or blanks in a openFT command you must note the following:

- Special characters must be handled specifically if they can be control characters for the corresponding command shell. I.e. they must be either escaped individually or enclosed in order that the shell does not interpret them.
- Blanks act as separator for the command options and must be enclosed otherwise openFT interprets all characters following the blank as option.

For escaping and enclosing you have the following possibilities:

- You escape single special characters using the backslash (\). If the backslash itself is the special character it must be also escaped (\\).
- Enclosing depends on the platform:
  - Unix systems: single quotes or double quotes, e.g.:

    ```
    ft 'partner1!file'BLANK'002' 'userid,,&xyz12'
    ```

    If an entry also contains single quotes (') then it makes sense to enclose the entire entry in double quotes (").
  - Windows systems: double quotes, e.g.:

    ```
    ftexec ux1 "ftshwl -nb=12" Transunix1
    ```

    If the entry also contains double quotes (") then they must be escaped with backslash (\).

Hexadecimal entries

The transfer admission, the user ID, the password or the management password can also be specified in hexadecimal format in the form x'...' or X'...'.

The following applies to Unix systems:

- If you enter the password directly, the single quotes must be escaped, e.g. X\'c6d9e4c5\', except the complete input has been enclosed.
- If your entry is not displayed (e.g. when prompting the password on the screen), the single quotes must not be escaped.

Example

- Unix systems: x\'f1f2f3f4f5f6f6f8\'.
- Windows systems: x'f1f2f3f4f5f6f6f8'.

Sequence of entries

The sequence of entries in the command is arbitrary.
Exceptions to this are specifications that do **not** start with a minus sign in the command syntax description if there is more than one such specification, e.g.:

- the source and destination of a request (e.g. local and remote file name, partner name,...)
- the authorization to access the remote system, i.e., the transfer admission or the system login.

Continuation lines on Unix systems

When there is a large number of parameters, openFT commands can be very long. If you want to use the keyboard to enter commands that are longer than 256 characters, you will need to work with continuation lines. You can obtain these by entering the sequence "\" (backslash) followed by Return.

Lengths on Windows systems

In Windows systems, openFT administers commands, follow-up-processing commands and file names using the character set UTF-8. The maximum lengths are therefore based on the UTF-8 corresponding representation. Characters that are habitually used but that are not present in the ISO646 character sets (ASCII characters) have a length of two or three bytes in UTF-8 (e.g. the Euro symbol).

File name

You can specify an absolute or relative file name. The file name specified in the local and remote systems may have a maximum length of 512 characters based on the length of the absolute path name. Please note that although long file names can be specified at the openFT interfaces, not all platforms support this maximum length. For example Unix systems permit up to 512 characters whereas Windows systems only permit 256 characters.

If the file name contains blanks, they must be set in double quotes ("), e.g. "file name".

Notes for Windows systems

- A local file name in UTF-8 representation may not exceed 512 bytes.
- The specification of UNC names is also possible.

%UNIQUE variable

If a file name ends with %unique or %UNIQUE, this string will be replaced by another string, which varies with each new call.

This string is 14 characters long in Unix systems, 18 characters long in Windows systems, 22 characters long in BS2000 systems and 15 or 8 characters long (for libraries) in z/OS systems.

If the receiving system is a Unix or Windows system, a suffix may follow %unique or %UNIQUE separated by a dot, e.g. "file1%unique.txt". This suffix must not contain any dot.

Only the already converted file name is displayed in both the log and the messages.

Date

The date must be numeric; exactly 8 characters in the form `yyyymmdd` with:`yyyy` for year, `mm` for month and `dd` for day

> Note that for all date entries, you may only specify values up to and including 20380119 (January 19, 2038)

Local user ID

The maximum length is system-dependent: In Unix systems, a maximum of 32 characters with first 8 characters being unique; in Windows systems, a maximum of 36 characters. When the user ID is entered in hexadecimal format, the maximum length is 64 characters + 3 characters for hexadecimal format (X' '), see also Hexadecimal entries.

Local FTAC Transfer admission

The FTAC transfer admission usually consists of printing characters and may not start with a hyphen, minimum 8 characters. The maximum length is system-dependent: In Unix systems, a maximum of 32 characters; in Windows systems, a maximum of 36 characters. When the transfer admission is entered in hexadecimal format, the maximum length is 64 characters + 3 characters for hexadecimal format (X' '). If a transfer admission consists of non-printing characters then it must be specified in hexadecimal format, see Hexadecimalentries.

Profile name

the profile name must be alphanumeric (a..z, A..Z, 0..9), up to 8 characters.

Input of openFT commands via shell procedures

When openFT commands are input via shell procedures please note the following:

Unix systems

Shell procedures with UTF-8-coded data are not accepted on Unix systems in some cases, as they are allegedly binary and cannot be executed (cannot execute binary files).

Unfortunately, this happens especially when the LANG variable is set for the procedure call in such a way that it locally indicates UTF-8 coding. This was observed on Linux systems for example with */bin/sh* and with *bash*; *ksh* on the other hand can also run UTF-8 shell procedures.

It often helps to explicitly set the LANG variable again at the start of the shell procedure, e.g. *export LANG=en_US.utf8*. If the shell procedure is running, the commands are processed on a byte-by-byte basis without code transformation - as if they had been entered successively in a console window.

Windows systems

openFT command parameters from a shell procedure on Windows systems are recoded by the system to UCS-2. The character set from which this takes place depends on the system settings. The OEM character set CP850 is usually the default setting in the West/Central European language area.

It is possible to vary the character set before calling the shell procedure, or also within the procedure, by using the console command *chcp*. You can use *chcp 65001* for example to run procedures coded in UTF-8. These procedures must use CRLF as a line separator; other line separators (e.g. only LF) are under certain circumstances not processed correctly by

Windows. The code table set using chcp is of no significance for the direct entry of openFT commands in a console window; the interpretation corresponds to the description in the console window.

It is recommended to create and edit procedures coded in UTF-8 using the openFT edtior (with *ftedit -ccs=utf8*), because Windows editors mostly write a BOM (byte order mask) at the start of the procedure file, which is interpreted as part of the first command during the call.

2.3 Output of openFT commands

The following applies for messages and output of openFT commands in the local system:

Unix systems

Messages and display output appear as before in local character code. Recoding does not take place. Exceptions: the output of a locally invoked ftshwc command, which is converted from UTF-8 to the locally set character code as of V12.1, as well as the output data streams when displaying remote file contents and when remotely executing commands (*ftexec*, *ftadm*, pre-processing); the local CCS specifications in the appropriate calls are regarded here as the target code.

If you change the local code setting, then you must also expect the output appearance of an openFT command to change. If for example an openFT request with the local file name "Köln" was called with UTF-8 as the local character code, and the character code changed in the meantime to ISO8859-1 when viewing the appropriate logging entry, the local file name now appears in the logging record as "KÄ¶ln".

Windows systems

Console output appears as wide characters (WideChar), i.e. in a UCS-2 character code internal to the operating system. This enables all the characters to be output that can be represented in the set graphic character set. This also applies for file contents, which are obtained via an ncopy command and output on stdout if UTF8 was specified as the local CCS name.

Messages and output that are rerouted to a file or pipe are output according to the environment variable OPENFTOUT as per the following rule:

OPENFTOUT = OLD	Output as before, i.e. in OEM for *stderr* or table output, and in local ANSI coding for *stdout* and csv output. Characters outside this character set are replaced; usually by a question mark.
OPENFTOUT not defined	as OPENFTOUT = OLD
OPENFTOUT = UTF8	Output in UTF-8

2.4 Specifying partner addresses

The following applies to the addressing of a partner system:

- You can specify the name of the partner from the partner list provided that the partner has been entered in the partner list.

  A partner has to be entered in the partner list by the FT administrator. For this purpose, the FT administrator can use the following commands:

  - ftaddptn
  - ftmodptn
- You can access a partner directly via its address in FT or file management requests even if it is not entered in the partner list. This is only possible if the “dynamic partner” function is enabled via operating parameters.

Partner addresses

A partner address has the following structure:

[protocol://]host[:[port].[tsel].[ssel].[psel]]

host (= computer name, see "Specifying partner addresses") is mandatory; all other specifications are optional. In many cases, the other specifications are covered by the default values, so that the host name suffices as the partner address, see “Examples” . Final ’.’ or ’:’ can be omitted.

The individual components of the address have the following meanings:

protocol://

Protocol stack via which the partner is addressed. Possible values for *protocol* (uppercase and lowercase are not distinguished):

openft

openFT partner, i.e. communication takes place over the openFT protocol.

ftam

FTAM partner, i.e. communication takes place over the FTAM protocol.

ftp

FTP partner, i.e. communication takes place over the FTP protocol.

ftadm

ADM partner, i.e. communication takes place over the FTADM protocol for remote administration and ADM traps.

Default value: **openft**

Exception: if a global name from the TNS is used for *host* and a presentation selector is assigned to this name in the TNS then **ftam** is the default value.

host

Computer name via which the partner is addressed. Possible entries:

- Internet host name (e.g. DNS name), length 1 to 80 characters

- Global name from the Transport Name Service (TNS), up to 78 characters long, with full support for the 5 name parts. In this event, the following applies:
  - TNS must be activated (*ftmodo -tns=y*) and operation with CMX must be enabled to allow a global name from the TNS to be used in requests. In this case, the TNS name takes precedence over the Internet host name.
  - The partner address must end with *host* and must not contain any other address components, such as *port* , *tsel* etc.
  - *ftp* is not permitted for *protocol* , as openFT-FTP does not support TNS operation.
  - If the TNS entry contains a presentation selector for this global name, only *ftam* is permitted for *protocol* .
  - If the TNS entry does not contain a presentation selector, *ftam* is not permitted for *protocol* .

  Note for Windows systems

  If you are using TranSON, the partner is only available over the TNS. To do this, a proxy must be entered in TNS. For further information, refer to the online Help system for the "TNS User Interface" application of PCMX-32.
- IPv4 address with the prefix %ip, e.g. %ip139.22.33.44
  You should always specify the IP address with the prefix %ip since the specification is then immediately treated as the IP address. Omitting this prefix results in performance impairments since in this case a search is initially performed in the TNS and then in the hosts file. The pathname is / *etc/hosts* (Unix systems) or *%SystemRoot%\system32\drivers\etc\hosts* (Windows systems).
  The IP address must always be specified as a sequence of decimal numbers separated by dots and without leading zeros.
- IPv6 address with the prefix %ip6, e.g.
  `%ip6[FEDC:BA98:7654:3210:FEDC:BA98:7654:3210]` (IPv6) or
  `%ip6[FE80::20C:29ff:fe22:b670%5]` (IPv6 with ccope ID)

  The square brackets [..] must be specified.

  The scope ID designates the local network card via which the remote partner can be accessed in the same LAN segment. It must be appended to the address with a % character. In Windows systems, this is a numerical value (e.g. 5). On other s ystems, it may also be a symbolic name (e.g. *eth0*). The scope ID can be identified using the *ipconfig* command (Unix systems) or the *ifconfig* command (Windows systems).

  If a unique allocation is required via the partner address, one and the same IPv6 address (with the same selectors) and with a different scope ID 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, thus enabling unique partner allocation even with identical link-local addresses plus different scope IDs.

  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 openFT or FTADM partner entries the scope ID or line number is not transferred to the identification in case of defaulting.

- 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:
  %x25[DTE address%line number]

  The prefix %x25, the square brackets [...] and the DTE address are mandatory and must be specified. The specification of the local line number of the FarSync X.25 card after the percentage is optional, the standard is 0:0 (Windows) i.e. 0 (Linux).

  Example for Windows systems
  %x25[123456%0:3]

  Example for Linux systems
  %x25[123456%0]

  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.

  If a unique allocation is required via the partner address, one and the same DTE address (with the same selectors) and with a different line number should not occur twice in partner list entries.
- TNS name from the z/OS library (TNSTCPIP member), up to 8 characters in length.

port

When a connection is established over TCP/IP, you can specify the port name under which the file transfer application can be accessed in the partner system.
Permitted range of values: 1 through 65535.

Default value:

- **1100** for openFT partners.
  A different default value can also be set in the operating parameters using the following command: *ftmodo -ftstd=*
- **4800** for FTAM partners.
- **21** for FTP partners
- **11000** for ADM partners

tsel

Transport selector under which the file transfer application is available in the partner system. The transport selector is only relevant for openFT partners and FTAM partners . You can specify the selector in printable or hexadecimal format (0xnnnn...).

The specification will depend on the type of partner:

- openFT partner:
  Length, 1 through 8 characters; alphanumeric characters and the special characters # @ $ are permitted. A printable selector will be coded in EBCDIC in the protocol and may be padded with spaces internally to the length of eight characters.

  Default value: **$FJAM**

- FTAM partner:
  Length 1 to 10 characters; a printable selector will be coded as variable length
  ASCII in the protocol. Exception: T-selectors that start with $FTAM (default value) are coded in EBCDIC and padded with spaces to the length of 8 characters.

  All alphanumeric characters and the special characters @ $ # _ - + = and * can be used with ASCII selectors.

  Default value: **$FTAM**

  Note :
  As a rule, **SNI-FTAM** must be specified for Windows partners with openFT-FTAM up to V10. As of openFT-FTAM V11 for Windows, the default value has been
  changed to **$FTAM** and can therefore be omitted.

Printable transport selectors are always used in uppercase in openFT even if they are specified or output in lowercase.

ssel

Session selector under which the file transfer application is accessible in the partner system. You can specify the selector in printable or hexadecimal format (0xnnnn...). Length, 1 through 10 characters; alphanumeric characters and the special characters @ $ # _ - + = * are permitted. A printable selector is encoded in ASCII with a variable length in the log.

Default value: empty

Printable session selectors are always used in uppercase in openFT even if they are specified or output in lowercase.

psel

Only relevant for FTAM partners.
Presentation selector under which the file transfer application is available in the partner system. You can specify the selector in printable or hexadecimal format (0xnnnn...). Length, 1 through 10 characters; alphanumeric characters and the special characters @ $ # _ - + = * are permitted. A printable selector is interpreted as ASCII with a variable length in the log.

Default value: empty

Printable presentation selectors are always used in uppercase in openFT even if they are specified or output in lowercase.

Examples

The partner computer with the host name FILESERV is to be addressed over different protocols/connection types:

Connection type/protocol	**Address specification**
openFT partner	FILESERV

FTAM partner (Windows system as of V11.0, BS2000 or Unix system with default setting as of V11.0)	ftam : / /FILESERV
FTAM partner (Windows system with default setting up to V10.0)	ftam : / /FILESERV:.SNI-FTAM
Third-party FTAM partner	ftam: / /FILESERV:102.TS0001.SES1.PSFTAM
FTP partner	ftp : / /FILESERV
FTADM partner	ftadm://FILESERV

2.5 Entering the authorization data for the partner system

The authorization data can be specified via login/LOGON authorization or via FTAC transfer admission, see the following table:

System	FTAC transfer admission	Login name	Account number	Password
BS2000	8 - 32 character long C string or 15 - 64 character long X string	1 - 8 alphanumeric characters	1 - 8 alphanumeric characters	1 - 32 character long C string or 1 - 16 character long X string
Unix based	8 - 32 characters long C string (Unicode characters are also permitted) or 15 - 64 characters long X string	1 - 32 characters	Unix systems do not recognize any account numbers locally	Alphanumeric characters (the length is system dependent), a distinction is made between uppercase and lowercase
Windows	8 - 36 characters (Unicode characters are also permitted)	1 - 36 characters, possibly with leading domain name (DOM\)	Windows does not recognize any account numbers locally	8 - 32 character long C string or 15 - 64 character long X string
z/OS	8 - 32 character long C string or 15 - 64 character long X string	1 - 8 alphanumeric characters	max. 40 characters, uppercase, digits and special characters $, @, #	1 - 8 alphanumeric characters

Examples

If you do not possess FTAC transfer admission then you can specify the LOGON/login authorization for the individual platforms using the following syntax:

- BS2000 systems:

  ```
  userid,[account-number][,'password']
  ```

  You can omit the account number if the user has a default account number for the BS2000 timesharing mode and you want to use this default account number.

- Unix systems

  ```
  userid[,,password]
  ```

- Windows systems:

  `userid[,,password]`

  The user ID consists of a user name (In the case of local IDs, the " `hostname\` " must not be entered in front of the user ID.) or, if a user ID in a LAN Manager or Windows domain is accessed, it consists of the domain name followed by a backslash (\) and the user name.

  Remember to escape the backslash on Unix systems (\\).
- OS/390 and z/OS:

  `userid,account-number[,password]`

  The accounting number is optional with more recent z/OS versions.
- In the case of other partner systems, your specifications depend on the conventions used in the partner system.

Inbound access using the default FTP client

If you wish to access an openFT server from a standard FTP client, you should note the following:

- Establishing a connection
  If the default listener port 21 is set on the openFT FTP server, enter the following from the shell (Unix systems), from the command prompt (Windows) or on command level (BS2000 and z/OS):

  `ftp` *hostname*

  hostname is the host name of the openFT FTP server.

  If a listener port other than 21 is set on the openFT FTP server, you need two commands to establish a connection:

  `ftp`

  `ftp> open` *hostname port-number*
- Login
  If you log in without an FTAC transfer admission, enter the login data interactively as usual (user ID and any password that is required and/or account number). If you log in using an FTAC transfer admission, enter the FTAC transfer admission under *User* and leave the *Password* empty.

  Example

  ```
  User: ftpuser1
  Password: (empty)
  ```

  With openFT FTP servers as of V11, you can enter the value *$ftac* under *User* and the FTAC transfer admission under *Password* .

  Example

  ```
  User: $ftac
  Password: ftpuser1
  ```

2.6 Preprocessing and postprocessing

If preprocessing or postprocessing runs in a Unix or Windows system then the following applies:

- During preprocessing the data is by default output to *stdout*. You can, however, also output the data created by preprocessing in a temporary file created by openFT. You can detect the name of this file via the variable % TEMPFILE and give it over to the preprocessing as a calling parameter. The temporary file is then transferred to the partner system.
- During postprocessing, the data is read from *stdin* by default. In this case, it must possess a format which can be processed by *stdin*. However, it is also possible to output the transferred data to a temporary file created by openFT. You can detect the name of this file via the variable %TEMPFILE and give it over to the postprocessing as a calling parameter. The postprocessing then can read the data from the specified temporary file and can process it.

2.7 Commands for follow-up processing

Maximum length

The total number of entries for local follow-up processing, i.e. for *-ls* and *-lf*, may not exceed 1000 characters.

The total number of characters for remote follow-up processing, i.e. for *-rs* and *-rf*, may not exceed 1000 characters, but this maximum value may be lower if a FT version < V10 is used in the remote system.

This maximum length applies to the representation in UTF-8, see also section “Entering commands”.

Syntax rules

- Unix systems

  The entries for follow-up processing must be enclosed in single quotes (') or double quotes ("). If the entry for follow-up processing also contains single quotes (') such as the single quotes in a BS2000 password, the entire entry must be enclosed in double quotes (").

- Windows systems

  The entries for follow-up processing must be enclosed in double quotes ("). If the followup processing specification contains quotes or backslashes, these must be escaped with a backslash (\).

Variables

- When starting follow-up processing in the remote system, the specified variables are first substituted, and the follow-up processing commands are then executed. The following variables are permitted:

  %FILENAME

  File name in the relevant system. The entry is automatically taken from the command. If you specified the variable *%UNIQUE* (or *%unique*) for the remote file name during transfer, the *%FILENAME* variable contains the already converted (i.e. unique) file name.

  %PARTNER

  Name or address of the partner system in long form, i.e. with dynamic partners, all address components are taken (protocol prefix, port number, selectors, ...). *%PARTNER* is substituted by the name of the initiator system (with the name as known in the partner system).

  %PARTNERAT

  Name or address of the partner system in short form, i.e. with dynamic partners, only the *host* address component is taken, see section “Specifying partner addresses”. In addition, each character is replaced by a '@' if it is neither a letter nor a digit or a period.

  %RESULT

  Message number of the request, as required by the system concerned. If, for example, a send request is successfully executed, the value of *%RESULT* in the local system contains the message number 0 (in openFT V10 and higher).

  If the partner is an openFT on a BS2000 system, you may also use the variables *%ELEMNAME*, *%ELEMVERS* and *%ELEMTYP.*

- Follow-up processing in a Unix system does not involve execution of the sequence of commands stored in the . profile file. Only the default values of the $HOME, $LOGNAME, $PATH and $USER shell variables are available, as well as the shell variables LANG and TZ as they were set by ftstart in the remote system. The shell or called programs may set further environment variables.

- When follow-up processing is processed on a Windows system, only the system environment variables are available, not the user variables. In addition, the userspecific Registry entries are not loaded before follow-up processing is executed.

Which commands can be entered?

- You can enter all commands of the corresponding operating system.
- You can enter openFT commands, since the search path (PATH variable) for follow-up processing commands is preceded by the directory which contains the openFT commands of the respective instance, e.g. */var/openFT /instance/openFT/bin* in Unix systems, where instance means the name of the corresponding instance.
- When specifying BS2000 commands, remember to insert a slash (/) at the beginning of the command.
- Special considerations in Windows systems:
  - Any program can be started, e.g. a shell command, a program (*.exe* or *.com*) or a batch procedure (*.bat* or *. cmd*). If the command requires a path specification, then use the absolute path.
  - Before calling the follow-up processing, it is also possible to switch to another directory as follows:

    cd *path-name*;*command*

    path-name is then used as the current directory. There must not be a blank between the semicolon and the command. *path-name* must not be a directory which is addressed using a UNC name.
    Exception: The UNC checking is deactivated on the system on which the command is to be executed. To do this, the registry value described under
    http://support.microsoft.com/kb/156276/de has to be generated.

    If the HOME directory is a network drive then *cmd.exe* may issue a warning and command execution may not take place on the network drive but at another directory.
  - If you wish to execute shell-internal Windows commands (e.g. *move* or *copy*), remember that you must specify the command processor *cmd.exe /c* at the start of the command.

Impacts on the FT request

If a prefix or suffix was defined in the profile, the character set available for specifying followup processing in the FT request is restricted to:

- alphanumeric characters (letters (including blanks) and digits)
- the special characters + = / ! - , @ _ " $ ' (all systems) and \ : # (only Windows systems)
- a period (.) between alphanumeric characters

Follow-up-processing for FTAM and FTP

With requests for FTAM and FTP partners, the follow-up processing function is not available in the remote system (exception: -rs='*DELETE' for FTAM receive requests to delete the send file after successful processing). If FTAC is used in the remote system, this restriction can be avoided by creating an FT profile in the remote system and defining follow-up processing for it.

2.8 Instance identification

An instance ID may have a maximum length of 64 characters and may be comprised of alphanumeric characters and the special characters. It is advisable to use only the special characters ".", "-", ":" and "%". The first character must be alphanumeric or be the special character “%”. The character “%” can only be used as an initial character. An alphanumeric character must follow a “.”.

In order to ensure the network-wide, uniqueness of the instance ID, you should proceed as follows when allocating the instance IDs:

- If the openFT instance has a network address with a **DNS name** you should use this as the ID. You can create an “artificial” DNS name for an openFT instance, by placing another part of a name in front of an existing “neighboring” DNS name, separated by a period.
- If the openFT instance does not have a DNS name, but is connected to a TCP/IP network, you should use the following ID.
  - IPv4: **%ip***n.n.n.n* (*n.n.n.n* is the IPv4 address of the local openFT instance without leading zeros in the address components).
  - IPv6: **%ip6[***x***:***x***:***x***:***x***:***x***:***x***:***x***:***x***]** (without scope ID) or
    IPv6: **%ip6[***x***:***x***:***x***:***x***:***x***:***x***:***x***:***x***%***s***]** (with Scope ID)
    where *x:x:x:x:x:x:x:x* is the IPv6 address of the local openFT instance and *s* is the scope ID of the local network card.

2.9 Output in CSV format

For some Show commands, openFT offers output in CSV format. CSV (**C**haracter **S**eparated **V**alues) is a popular format in the PC environment in which tabular data is defined by lines. Output in CSV format is offered for the following commands:

- ftshw
- ftshwa
- ftshwact
- ftshwatp
- ftshwc
- ftshwe
- ftshwk
- ftshwl
- ftshwlic
- fshwm
- ftshwo
- ftshwp
- ftshwptn
- ftshwr
- ftshws
- ftshwsuo

Many programs such as spreadsheets, databases, etc., can import data in CSV format. This means that you can use the processing and presentation features of such programs on the data output by the above commands.

The output fields are described in the chapter “Structure of CSV outputs”.

Every record is output as a line, and each record contains information on an object. If data is present, the first line always contains the header with the field names of each of the columns. **Only the field names are guaranteed, not the order of fields in a record**. In other words, the order of fields is determined by the order of the field names in the header line. Fields within an output line are separated by semicolons (;).

One example of a possible evaluation procedure is supplied as a reference template in the Microsoft Excel format in the following file:

/opt/openFT/samples/msexcel/ftaccnt.xlt (Unix systems)
openFT installation directory\samples\msexcel\ftaccnt.xlt (Windows systems)

The template evaluates a CSV log file by means of an automatically running macro. The result shows the number of inbound and outbound requests and the Kilobytes transferred in each case for all users.

2.10 Notes on FTP partners

If the FTP protocol is used then only communication via TCP/IP is possible. Furthermore, a number of special considerations apply when FTP servers are used compared to openFT partners. These are for the most part due to limitations in the FTP protocol:

- No restart is performed.
- Encryption is only possible for outbound requests to an FTP server that provides support for Secure FTP with the TLS protocol. This requires openFT-CR delivery unit to be installed.
- If encryption of the user data is required and the FTP server does not provide encryption, the request is rejected. If encrypted transfer of the user data is required, the login data is also encrypted. If encryption of the user data is not required, the login data is encrypted if the FTP server provides this. No mutual authentication is carried out.
- Coded character sets are only supported locally; specifications for the partner system cannot be transported by the FTP protocol.
- When files with a record structure are transferred in binary format, the record structure is lost. The contents of the records are stored in the destination file as a byte stream.
- File attributes are not supported by the FTP protocol. This means that the modification date and maximum record length are not taken over for the destination file.
- If the *ftexec* command is issued to a mainframe over the FTP protocol, the *-t* option must be used. The *-b* option (default) is rejected in the remote system with a message indicating that the file structure is not supported.
- Follow-up processing is only possible on the local system or by specifying the FTAC profiles.
- The modification date cannot be taken over for the destination file. As a result, the modification date of the destination file is set to the transfer date. This is of particular importance when comparing file hierarchies.
- If an FTP server does not provide the information as to whether a symbolic link refers to a file or a directory when listing directories, the link is by default shown as a file in openFT Explorer (on Unix and Windows systems).
- The maximum record length of the send file is not passed to the receiving system. This has an impact when transferring files to a mainframe system such as BS2000 or z/OS. In this case, the default maximum record length applies in the receiving system.
- The size of the send file is not passed to the receiving system. This has an impact when transferring files to a mainframe system such as BS2000 or z/OS. The maximum file size is derived from the default value that is used by openFT for primary and secondary allocation and by the maximum number of file extents defined by the system, see openFT manual "Concepts and Functions". If a file exceeds this size, the request is cancelled with the message: “File gets no more space”.
- The 'do not overwrite' option can have a different effect because this option cannot be passed to the responder, and the initiator must check whether the file already exists in the partner system. This has the following consequences:
  - It is possible for a request with the ’do not overwrite’ option to overwrite a file that has been created by a third party in the period between the check being performed by the initiator and the actual transfer.
  - If 'overwrite' is specified in an admission profile and if the file to be transferred does not yet exist, a request using this profile will still be executed, even if 'do not overwrite' is set in the request.

Please note that the other openFT functions (preprocessing and postprocessing, FTAC, etc.) can only be used if openFT is used as the FTP server on the system, where preprocessing and postprocessing are to be performed.

Problems may also occur when addressing FTP servers which send an unexpected layout when listing directories.

3 openFT commands

This chapter contains all openFT commands.

The section "Note on usage" lists (amongst others) the user groups for which the command is intended or allowed:

- FT user
- FT administrator
- FTAC user
- FTAC administrator
- ADM administrator

3.1 ft

Note on usage

Function: Asynchronous file transfer

User group: FT user

Alias name: *ftacopy*

Functional description

The *ft* command is used to issue asynchronous file transfer requests for sending a file or a directory to a remote system or for fetching a file or a directory from a remote system. In addition, you can use the preprocessing, postprocessing or follow-up processing capabilities to execute operating system commands in the local or remote system. Once openFT has stored the request in the request queue, your user process will be available again. openFT performs the actual transfer operation asynchronously to your user process at the earliest opportunity or at a time you specify, provided resources are free and the partner is available.

openFT acknowledges receipt of the request by default, with the output of the following message on the screen (*stderr*) of the user who issued the request

`ft: Request` *request ID* `accepted.`

request ID

is replaced by the transfer identification of the transfer request.

After acknowledgment of the request, the user process continues to run.

Note for Unix systems

If you want, you can use the *-m* option to tell openFT to send a result notification to the initiator's mail box if the request is processed successfully and/or unsuccessfully.

Note for Windows systems

The ft command usually cannot be executed in PowerShell because it is overwritten by the system command ft (Format Table). It is recommended to use the alias name ftacopy.

If openFT rejects your request, an error message will be displayed explaining why it was rejected (see chapter “Messages”).

The maximum number of requests that can be stored in the request queue is specified in the operating parameters. You can raise the default value of 2000 up to a maximum of 32000 (see the *ftmodo* command). Any further requests are rejected.

You can also obtain the result of an *ft* request by using the log function (see *ftshwl* command).

> **i** A number of special issues and restrictions apply for transfer requests with FTP partners. For details, see section “Notes on FTP partners”.

Only one file can be fetched from a remote system for each *ft* command. If you want to fetch several files asynchronously, use the *ft_mget* command.

Format

ft -h |

[-t | -u | -b] [-x]
[-o | -e | -n]
[-k | -z][-c][-N][-S][-m=n | -m=f | -m=a] *)
[-d]
[<file name 1..512> <partner 1..200>![<file name 1..512>]] |
[<partner 1..200>![<file name 1..512>] <file name 1..512>]
[<transfer admission 8..67> | @n | @d |
<user ID 1..67>[,[<account 1..64>][,[<password 1..64>]]]]
[-p=[<password 1..64>]][-di]
[-lc=<CCS name 1..8>][-rc=<CCS name 1..8>]
[-ls=<follow-up proc 1..1000>][-lf=<follow-up proc 1..1000>
]
[-rs=<follow-up proc 1..1000>][-rf=<follow-up proc 1..1000>
]
[-r=v[<1..65535>] | -r=f[<1..65535>] | -r=u[<1..65535>] |
-r=<1..65535>]
[-tff=b | -tff=s][-trf=u]
[-tb=n | -tb=f | -tb=a]
[-av=i | -av=d][-ac=<new account 1..64>]
[-am=[r][i][p][x][e][a][c][d] | -am=@rw | -am=@ro]
[-lq=<legal qualification 1..80>]
[-cp=[<password 1..64>]][-pr=n | -pr=l]
[-sd=yyyymmdd | +<start date 0..ddddd>]
[-st=[+]<start time hhmm>]
[-cd=yyyymmdd | +<cancel date 0..ddddd>]
[-ct=[+]<cancel time hhmm>]
[-fnc=t | -fnc=c]
[-md]

*) The options *-N* and *-m* are only available on Unix systems

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

[**-t** | **-u** | **-b**] [**-x**]

Identifies the type of file in the local operating system.

If you send a file to an FTAM partner without specifying a file type, the file type is determined by the structure entries of the send file. The structure entries can be displayed by outputting the local FT attributes (*ftshwf file name -l*). If there are no structure entries, the default value is *-t*. If you fetch a file from an FTAM partner without specifying a file type, the file type is determined by the file attributes in the FTAM partner. For more detailed information about file types when dealing with FTAM partners, see the openFT manual "Concepts and Functions".

-t (default value with openFT partners)

The file contains text with variable-length records.

Records end with the linefeed character \n on Unix systems.

On Windows systems, records end with the following characters:

- CRLF (X'0D0A') when sending and/or fetching a file
- LF (X'0A'), only possible when sending a file

-u

The file contains user-structured binary data with variable-length records. Each record starts with 2 bytes which contain the length data of the record.

-b

The file contains user-structured binary data with variable-length records.

If you specify the *-b* switch together with *-r* (maximum record length), the file contains binary data with record length specified for *-r*. The size of the send file must be a multiple of this record length.

-x

The send file is transferred in a transparent file format and is stored in the destination system, i.e. this is a file whose attributes are transparent for the local system.
The local system here acts as a storage and/or transport medium.

If a file is transparently retrieved with *-x* for local buffering, then it must be sent again to the remote system in binary form (i.e. with *-b*).

-o | **-e** | **-n**

Indicates whether a destination file is to be newly created, overwritten or extended.

-o (default value)

The destination file will be overwritten or newly created if it does not already exist.

In case of directory transfer (*-d* option), the target files are overwritten if the specified directory and the files in this directory already exist. Otherwise, the target directory, subdirectories (if they may exist) and the files are newly created. Files and subdirectories which only exist in the target directory remain unchanged.

-e

The transferred file will be appended to an existing destination file. If this destination file does not exist, it will be newly created.

-e is not permitted in case of directory transfer (*-d* option).

-n

The destination file will be newly created and written. If the destination file already exists, the request will be rejected. In this way, you can protect a file from being overwritten inadvertently.

In case of directory transfer (*-d* option), the target directory and the files are newly created. If the target directory already exists, the request is rejected.

-k

Indicates that identical characters repeated consecutively are to be transferred in compressed form (byte compression). In the case of connections to partners which do not support this type of compression, no compression are used automatically.

-z

Indicates that zip compression is used. In the case of connections to partners which do not support this type of compression, byte compression (corresponds to the option *-k*) or no compression are used automatically.

-c

Indicates that the data are also encrypted for file transfer. To do this, the openFT-CR module must have been be installed. The encryption of the request description data is not affected by this option. If the partner system does not support data encryption, the request is rejected.

-N

(only on Unix systems) Suppresses result messages being deposited in the mailbox of the user who issued the request. *-N* is the same as *-m=n*, but is still supported for compatibility reasons.

-S

Suppresses file transfer messages to *stderr*.

-m=n | **-m=f** | **-m=a**

(only on Unix systems) This indicates whether the result message is to be deposited in the mailbox of the user who issued the request.

With some systems, the mail cannot be delivered if the login name is longer that 8 bytes.

n

(default value) The result message is not deposited in the mailbox (identical to the *-N* option).

f

The result message is only deposited in the mailbox in the event of errors.

a

The result message is always deposited in the mailbox.

-d

Indicates a directory transfer.

Local and remote file names are interpreted as directory names.

-d is only supported for openFT partners (not for FTAM or FTP partners). Preprocessing and postprocessing are not supported.

If you are using the *-d* option together with other options (e.g overwrite (*-o*) or follow-up processing (*-ls*, *-rs*,...)) then these options apply to the individual files in the directory to be transferred.

file name partner**!**[file name] | partner**!**[file name] file name

specifies the source and destination. The syntax depends on the direction of transfer selected and whether pre- or postprocessing commands are used or whether a directory is transferred. If you are using the option *-d* (directory transfer) then the source and destination file name are considered as directory names.

Sending without pre-/postprocessing

Sending a file

Source	**Destination**
local file name	partner**!**[*remote* file name]

Sending a directory

Source	**Destination**
local directoy name	partner**!** *remote* directoy name

If you transfer a directory (*-d*) then you specify the directory you want to transfer in *local directory name*. For *remote directory name*, you specify the directory under which the transferred directory is stored with identical file names and subdirectory names if applicable. The specification for the remote directory may not be omitted.

Fetching without pre-/postprocessing

Fetching a file

Source	**Destination**
partner**!**[*remote* file name]	*local* file name

Fetching a directory

Source	**Destination**
partner**!**[*remote* directory name]	*local* directory name

Sending and fetching a file with pre- or postprocessing

If you want to perform pre- or postprocessing, then you must enter an operating system command instead of the local or remote file name (in the syntax of the corresponding system).

Sending with preprocessing

Source	**Destination**
"| *local* command**"**	partner**!**[*remote* file name]

Sending with post-processing

Source	Destination
local file name	Partner**!"|** *remote* command**"**

Fetching with preprocessing

Source	Destination
Partner**!"|** *remote* command**"**	*local* file name

Fetching with post-processing

Source	Destination
Partner**!**[*remote* file name]	**"|** *local* command**"**

You can also combine preprocessing and postprocessing in the same request.

A maximum of 712 bytes may be specified both for *source* and *destination* (maximum 512 bytes for the file name and maximum 200 for the partner). Please note that the maximum lengths of file names are system-dependent; for example, in Unix systems it is 512 and in Windows systems a maximum of 256 bytes (for the representation in UTF- 8, see section “Entering commands”).

local file name
local directory name

> **Sending**: Name of the local file or directory (option *-d*). The name may include an absolute or relative path name.
>
> **Fetching**: Name of the receiving local file or directory (option *-d*). The name may include an absolute or relative path name.
>
> However, the *ft* command will not create a directory which does not already exist. If the name ends with % unique or %UNIQUE, this string is replaced by a string which changes for each new call, see section “Entering commands”.

partner

> *partner* is the name of the partner system in the partner list or the address of the partner system. For details concerning address specification, see section “Specifying partner addresses”.

remote file name
remote directory name

> *remote file name* and *remote directory name* (option *-d*) can be either absolute or relative to the remote login authorization. If the file name or directory name in the remote system has been predefined in an FT profile, it must not be specified here. If the name contains blanks, it must be enclosed in double quotes (e. g. "file name").
>
> If the name ends with %unique or %UNIQUE, this string is replaced by a string which changes for each new call, see section “Entering commands”.
>
> If the partner system is running openFT(BS2000), elements from PLAM libraries may also be specified here (Syntax: Libname/Element type/Element name).

|command for *file name*

command is any command on the local or remote system (not permitted in case of directory transfer). The "|" character (pipe character) must always be placed before the command. Since the "|" character is a special character "**|**command" should always be enclosed in double quotes.

Please note that, as of openFT V12, pre- or postprocessing commands are converted to the UTF-8 character set in remote Windows systems and that more characters may therefore be required in the remote system see also section“Entering commands”.

In the case of preprocessing, openFT transfers the data output at the standard output by the command as a file. You can also output the data created by preprocessing in a temporary file created by openFT.

During postprocessing, you can have the transferred data stored in a temporary file created by openFT.

You can find out the name of this temporary file and pass it to preprocessing or postprocessing with the variable %TEMPFILE. See the section “Preprocessing and postprocessing” .

If command execution takes longer than ten minutes, a timeout occurs on partners using versions of openFT prior to V8.1 and command execution is regarded as having failed. This restriction no longer applies to partners using openFT V8.1 or later.

Remote command execution in Unix and Windows systems starts in the user's $HOME directory or home directory respectively.

Note for Unix systems

The PATH variable is used as follows in the search path for preprocessing and postprocessing commands in Unix systems:

- Standard instance: *:/opt/openFT/bin:/bin:/usr/bin:/opt/bin*
- Other instance: *:/var/openFT/instance/openFT/bin:/bin:/usr/bin:/opt/bin* where *instance* is the name of the relevant instance.

This means that the system first searches in the current directory (first ":").

Before calling a "real" preprocessing or postprocessing command you can switch to another directory as follows:

cd *path-name*;*command*

path-name is then used as the current directory. There must not be a blank between the semicolon and the command.

Note for local and remote Windows systems

path-name must not be a directory which is addressed using a UNC name. Exception: The UNC checking is deactivated on the system on which the command is to be executed. To do this, the registry value described under *https://support.microsoft.com/de-de/kb/156276* has to be generated.

If the string "|&" comes before the preprocessing/postprocessing command instead of the character "|", the openFT request is restartable (see the section “Preprocessing and postprocessing” .).

transfer admission | **@d** | **@n** |
user ID[**,**[account][**,**password]]]

To be able to send a file to a remote system or to fetch one from it, you must furnish the remote system with proof of identity. For this purpose, you will need login authorization in the syntax valid for the remote system. You can specify transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization in the syntax used by the remote system (user ID, possibly together with account or password).

For details, see section “Entering the authorization data for the partner system” .

@d for *transfer admission*

Specifying *@d* (blanked transfer admission) causes openFT to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission.

@n for *transfer admission*

By entering *@n*, you specify that the remote system requires no login authorization.

A binary password and a binary transfer admission must be entered in hexadecimal format, see section “Entering commands”.

password not specified

Omitting the password necessary for admission causes openFT to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

Nevertheless, you have to specify the commas, e.g.:

```
ft file partner!file user-id,,
```

or

```
ft file partner!file user-id,account,
```

neither *transfer admission* nor *user ID* specified

causes the same as *@d*, i.e. openFT queries the transfer admission on the screen after the command is entered. Your (blanked) entry is always interpreted as transfer admission and not as user ID.

-p=[password]

If the file in the remote system is protected by a write password, you must enter this password when sending a file. If the file is protected by a read password, then this password must be specified when fetching a file from the remote system.

A binary password must be entered in hexadecimal format, see section “Entering commands”. This is of relevance for links to openFT on a BS2000 system, because BS2000 supports the definition of hexadecimal passwords.

password not specified

Specifying *-p=* causes openFT to query the write or read password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

-di

is specified, if the data integrity of the transferred file is to be checked by cryptographic means. With it, harmful data manipulations on the transmission network are identified. In case of an error openFT performs an error recovery for asynchronous transfer requests.

If the partner system does not support the check of data integrity (e.g. openFT < V8.1), the request is denied.

For requests with data encryption (option *-c*), data integrity is automatically checked. Testing mechanisms of the transfer protocols in use automatically identify transfer errors in the network. For this purpose you do not have to specify the *-di* option.

-lc=CCS name

(local coding) specifies the type of coding (character set) to be used to read or write the local file. *CCS name* must be known in the local system.

The default value is the character set defined by the FT administrator.

Details about the CCS name and the associated code tables can be found in the manual "openFT (Unix and Windows systems) - Installation and Operation".

-rc=CCS name

(remote coding) specifies the type of coding to be used to read or write the remote file. *CCS name* must be known in the remote system.

The default value is the character set defined in the remote system via XHCS (BS2000 system) or the openFT operating parameters (other platforms).

The option *-rc* is supported only by the openFT protocol and partners with openFT V10.0 or higher. Please note that not all partner systems support all the character sets that are possible in the local system.

Details about the CCS name and the associated code tables can be found in the manual "openFT (Unix and Windows systems) - Installation and Operation".

-ls=follow-up processing

Here you can specify a command which will be executed in the local system following a **successful transfer** operation.

Further information is given in the section “Commands for follow-up processing” .

-lf=follow-up processing

Here you can specify a command which will be executed in the local system if a transfer operation is **terminated** as a result of an **error**.

Further information is given in the section “Commands for follow-up processing” .

-rs=follow-up processing

Here you can specify a command in the syntax of the remote system. Following a **successful transfer** operation, this command is executed in the remote system under the specified login.

Further information is given in the section “Commands for follow-up processing” .

-rf=follow-up processing

Here you can specify a command in the syntax of the remote system. This command will be executed in the remote system under the specified login if a transfer operation that has already started is **terminated** as a result of an **error**.

Further information is given in the section “Commands for follow-up processing” .

> If *-d* is specified (directory transfer) the follow-up processing is executed for all files in the directory.

-r=v[record length] | **-r=f**[record length] | -**r=u**[record length] | -**r=**record length

Specifies the record format and the record length. This also enables records that are longer than the default value to be transferred. However, you must bear in mind that not every record length can be processed in all partner systems.

If you have selected the file type *b* (binary), *record length* it the value for all records of the send file.

Maximum value for *record length*: 65535 bytes.

With FTAM partners, the maximum record length specification is not valid unless the file type is set explicitly to *t*, *b* or *u*.

It is also possible to output the record format, see also *ftmodf* command, option *-rf*:

v	variable record length, *record length* defines the maximum value
f	fixed record length, *record length* then applies to all records
u	undefined record length

The combinations *-u -r=frecordlength* and *-u -r=urecordlength* are not permitted.

If *-r* is omitted then the following default values apply for the record format:

Option	Default value	Corresponds to
-b	u (undefined)	-r=u...
-t	v (variable)	-r=v...
-u	v (variable)	-r=v...

-tff=b | **-tff=s**

Specifies the format of the destination file.

b

The destination file is to be saved as a block-structured file. This means, for example, that a file can be transferred to BS2000 and stored there as a PAM file. If you specify *-tff=b*, you must also specify the option *-b* (binary).

s

The destination file is to be saved as a sequential file and the record format is to be retained. This allows an ISAM file or PAM file to be fetched from BS2000, for instance.

-tff=b must not be specified at the same time as *-trf=u*.

-trf=u

Specifies that the file is to be transferred as a sequential file and that the record format of the destination file is to be undefined, i.e. the record structure of the send file is lost. If the file is being transferred to a BS2000 or z /OS system, one block is written per transfer unit.

-trf=u must not be specified at the same time as *-tff=b*.

neither *-tff* nor *-trf* specified

The destination file is to be stored in the same format as the send file.

-tb=n | **-tb=f** | **-tb=a**

Activates/deactivates tabulator expansion and the conversion of blank lines into lines with one character for non-FTAM partners for a single output send request.

The following parameters are provided:

n (on)

Tabulator expansion and blank line conversion are activated.

f (off)

Tabulator expansion and blank line conversion are deactivated.

a (automatic, default value)

Tabulator expansion and blank line conversion are activated if a file is sent to a BS2000, OS/390, or z/OS system.

No tabulator expansion or blank line conversion is performed for outbound receive requests.

If *ft* is used as a preprocessing command, then tabulator expansion and blank line conversion are always deactivated.

The following parameters *-av, -ac*, *-am*, *-lq* and *-cp* are provided exclusively for communication with FTAM partners. openFT thus supports the parameters defined in the FTAM standard. These parameters enable you to define the attributes of the destination file while issuing a file transfer request.

These parameters are ignored for requests involving openFT and FTP partners, but the file transfer is still carried out.

-av=i | **-av=d**

Indicates the availability of the destination file. This parameter can have one of two values: *immediate* or *deferred*. A file may be *deferred* if it has been archived, for example. The partner is responsible for interpreting the term *deferred*. The FTAM partner conventions must therefore be observed here.

The following values are possible:

i

The destination file attribute is set to *immediate*.

d

The destination file attribute is set to *deferred*.

-av is not available for requests involving FTAM partners that do not support the storage group. In this case, the request is executed, but the entry for *-av* is ignored.

-av not specified

The availability file attribute is set to a system-specific default value. In this case, this is the value *immediate*.

-ac=new account

With FTAM partners, this indicates the number of the account to which file storage fees are to be charged. This parameter must be set in accordance with partner system conventions.

-ac is not available for requests involving FTAM partners that do not support the storage group. In this case, the request is executed, but the entry for *-ac* is ignored.

-am=[r][i][p][x][e][a][c][d] | **-am=@rw** | **-am=@ro**

This sets the access rights of the destination file, provided the security group is available. The security group is defined in the manual "openFT (Unix and Windows systems) - Installation and Operation".

The following values can be specified for access mode:

r, *i*, *p*, *x*, *e*, *a*, *c*, *d*, any combination of these values, *@rw*, or *@ro*.

r

means that the file can be read.

r not specified

The file cannot be read.

i

means that data units, such as records, can be inserted in the file.

i not specified

No data units can be inserted in the file.

p

means that the file can be overwritten.

p not specified

The file cannot be overwritten.

x

means that data can be appended to the file.

x not specified

The file cannot be extended.

e

means that data units, such as records, can be deleted from the file.

e not specified

No data units can be deleted from the file.

a

means that the file attributes can be read.

a not specified

The file attributes cannot be read.

c

means that the file attributes can be changed.

c not specified

The file attributes cannot be changed.

d

means that the file can be deleted.

d not specified

The file cannot be deleted.

@rw

is the short form of the common access rights *read-write* (*rpxeacd*), and thus simplifies input.

@ro

is the short form for the common access rights *read-only* (*rac*), and thus simplifies input.

If the partner system is a Windows system, you cannot change the access rights of the destination file.

In Unix systems or in BS2000, the following access rights can be set for a file:

Access mode	Short form	Unix system	BS2000	Access rights
rpxeacd	@rw	rw*	ACCESS=WRITE	read-write
rac	@ro	r-*	ACCESS=READ	read-only

pxeacd		-w*	Only with BASIC-ACL (Access Control List)	write-only
ac		--*	Only with BASIC-ACL (Access Control List)	none

* The x bit is not changed by *ft*.

-am is not available for requests involving FTAM partners that do not support the security group. In this case, the request is executed, but the entry for *-am* is ignored.

-am not specified

The default values of the FTAM partner system apply.

-lq=legal qualification

This specifies a legal qualification for the destination file (similar to a copyright). This may not exceed 80 characters.

-lq is not available for requests involving FTAM partners that do not support the security group. The request is executed, but the entry for *-lq* is ignored.

-cp=[password]

If a password is required in order to create a file on a remote system, this password must be specified here. It can be up to 64 characters long.

A binary password must be specified in hexadecimal format, see section “Entering commands”. If you do not specify a file creation password, but you do enter a file access password for *-p=password*, the file creation password is identical to the file access password. The file creation password is of no significance when retrieving a file.

password not specified

Specifying *-cp=* causes openFT to query the file creation password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

-pr=n | **-pr=l**

indicates the priority of the request:

n (normal)

the request has the priority "normal" (default value).

l (low)

the request has the priority "low".

-sd=start date

indicates the earliest date at which the file transfer is to be started.

Possible values:

yyyymmdd

e.g. 20170331 for the start transfer on March 31, 2017. The largest possible value for the date is 20380119 (January 19, 2038).

+ddddd

e.g. +2 for start of transfer 2 days after issuing the request. You can delay file transfer by 999 days at the most. You can specify at most five figures for the delayed date. The value is limited by the number of days up to 19.01.2038.

-st=start time

specifies the earliest time at which file transfer is to be started (due to the nature of the system, the start time may deviate 5 minutes from the specified time). Possible values:

hhmm

e.g. 1430 for start of transfer at 14:30 hrs.

+hhmm

e.g. +0230 for start of transfer 2 hours and 30 minutes after issue of the request. The maximum delay you may specify is 99 hours and 59 minutes.

The start time must not be specified as relative if the start date has been specified as absolute. For a relative start date and start time, the start time is calculated from the total of the two entries, i.e. if a request is issued at 10.07. at 15:00 hrs. with *-sd=+1* and *-st=+1000*, the request is not started until 12.07. at 01:00 hrs.

If you enter a start date without a start time, transfer is started at 00:00 hrs. on the date specified. If you enter a start time without a start date, the time applies to the current date. If you specify a request with *-st=1000* at 15:00 hrs then the request is run immediately.

-cd=cancel date

Specifies the date on which the request is to be deleted. If the request is active at the time specified, it is aborted. Possible values:

yyyymmdd

e.g. 20170531 for cancellation of the request on May 31, 2017. The specified time must not lie in the past. The largest possible value for the date is 20380119 (January 19, 2038).

+ddddd

e.g. +2 for cancellation of the request 2 days after its issue. The maximum delay you may specify is 999 days. You can specify at most five figures for the delayed date. The value is limited by the number of days up to 19.01.2038.

-ct=cancel time

Specifies the time at which the request is to be deleted (due to the nature of the system, the start time may deviate 5 minutes from the specified time). The specified time must not lie in the past. If the request is active at the time specified, it is aborted. Possible values:

hhmm

e.g. 1430 for cancellation of the request at 14:30 hrs. The specified time must not lie in the past.

+hhmm

e.g. +0230 for cancellation of the request 2 hours and 30 minutes after its issue. The maximum delay you may specify is 99 hours and 59 minutes.

If you enter a cancel date without a cancel time, the file transfer is canceled at 23:59 hrs on the date specified. If you specify a cancel time without a cancel date, the time applies to the current date.

The cancel time must not be specified as relative if the cancel date has been specified as absolute. For a relative delete date and delete time, the delete time is calculated from the total of the two entries, i.e. if a request is issued at 10.07. at 15:00 hrs. with -cd=+1 and -ct=+1000, the request is not deleted until 12.07. at 01:00 hrs.

Requests also have a limited lifetime, even if no values are specified for *-cd* and *-ct*. This lifetime is set by the FT administrator. You may query the value using the command *ftshwo*. The entry stands for MAX-RQ-LIFE. Specifying *-cd* and *-ct* disables the MAX-RQ-LIFE entry.

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for file name and follow-up processing.

t (transparent, default value)

Specification of the remote file name and follow-up processing for the remote system in transparent mode (compatible to the previous versions).

c (character)

Specification of the remote file name and follow-up processing for the remote system in character mode. They are interpreted according to the character code of the remote system, i.e. for Unix partners according to the openFT operating parameter option (*ftmodo -fnccs)* that has been set there.

-fnc=c is only permitted for openFT partners as of openFT V12.1.

-md

(modification date)The modification date of the send file is taken over for the destination file provided that the destination system supports this. If the destination system does not support this function then the request is rejected. The use of this function is only of value for requests via the openFT protocol to BS2000 with OSD V8.0 or higher.

-md not specified

The behavior is the same as in openFT V11.0 or earlier: On Unix and Windows systems as well as under POSIX (BS2000), the modification date of the send file is taken over. On BS2000 with DMS, the current time is taken over as the modification date.

Examples

1. The text file *doc.one* is sent by user *jack* to the BS2000 computer with the symbolic name *bs2r1*. Here, it is stored under the login name *jim* with account number *a1234ft* and password C'*pwd*'. The file should then be printed.

   Unix systems:

```
ft doc.one bs2r1!doc.one jim,a1234ft,C\'pwd\'\
 -rs="/PRINT-FILE %FILENAME,LAYOUT-CONTROL=*PARAMETERS\
(,CONTROL-CHARACTERS=EBCDIC)"
```

Windows systems:

```
ft doc.one bs2r1!doc.one jim,a1234ft,C'pwd'
  -rs="/PRINT-FILE %FILENAME,LAYOUT-CONTROL=*PARAMETERS
(,CONTROL-CHARACTERS=EBCDIC)"
```

2. A file is to be fetched from BS2000, where openFT-AC is running, to the local system. The file name has been predefined in an FT profile, which can be accessed with the access authorization '*fortheRM6*'. In the local system, the file is to be stored in the *test* directory under the name *track.f* as a type *u* file (user format).

   ```
   ft -u bs2! test/track.f 'fortheRM6' (Unix systems)
   ft -u bs2! test\track.f 'fortheRM6' (Windows systems)
   ```

   Note: Windows also accepts '/' in file names.

3. The file *source.lst* is sent to the BS2000 computer *bs2r1*. Here, it is stored under the login name *jim* with account number *a1234ft* and password *C'pwd'* under the file name *lst*. Then, as follow-up processing, the file is to be printed out in BS2000 and then deleted. The source file in the local system is likewise deleted.

   Unix systems:

   ```
   ft source.lst bs2r1!lst jim,a1234ft,C\'pwd\'\
    -ls='rm source.lst'\
    -rs='/PRINT lst,DELETE-FILE=YES'
   ```

   Windows systems (with using the %FILENAME variable):

   ```
   ft D:\home\source.lst bs2r1!lst jim,a1234ft,C'pwd'
     -ls="cmd /c erase %FILENAME"
     -rs="/PRINT %FILENAME,DELETE-FILE=YES"
   ```

4. The text file

   letter is sent to the login name *jim* with the password *jimspass* in the FTAM partner with the symbolic name *ftampart*.

   ```
   ft letter ftampart!letter jim,,jimspass
   ```

5. The file

   data is sent from a Windows computer *pc123* to a Windows computer *pc234* with the transfer admission *topsecret* and stored there under the name *dat.txt* . Then, as follow-up processing, the procedure *evaluate* is started in the remote system if transferred successfully. The procedure contains the file name *dat.txt* , the partner *pc123* and the message number (0 for successful file transfer) as parameters. The parameters are specified using variables. If transfer is successful, the file is to be deleted in the local system.

   ```
   ft data pc234!dat.txt topsecret
   -rs="evaluate.cmd %FILENAME %PARTNER %RESULT"
   -ls=*DELETE
   ```

6. The text file

locfile is to be sent to the Unix computer *ux1*. Here, it is to be stored under the login name *charles* with the password *secret* under the file name *remfile*. Then, as follow-up processing, the file is to be printed out if transferred successfully; if not, the *prog* program is to be started in the remote system. As parameters, the program receives the name of the source file and the message number. The parameters are specified using variables. If the request is not completed after 5 hours, it is deleted from the request queue. If a data connection already existed then error follow-up processing, i.e. the command *prog %FILENAME %RESULT*, is started in the remote system.

Unix systems:

```
ft locfile ux1!remfile charles,,secret -r=100\
 -rs='lpr remfile' \
 -rf='prog %FILENAME %RESULT' \
 -ct=+0500
```

Windows systems:

```
ft locfile ux1!remfile charles,,secret -r=100
 -rs='lpr remfile'
 -rf='prog %FILENAME %RESULT'
 -ct=+0500
```

If file transfer is not successful, e.g. because the record length was greater than 100 bytes, follow-up processing is executed as follows:

```
prog remfile 2210
```

7. The file *locfile* is sent to the z/OS partner *zospart*. Here, the script PT (e.g. with a print job) is to be executed as follow-up processing under the user ID OPUSER.

   Unix systems:

   ```
   ft locfile zospart!remfile OPUSER,account,password \
    -rs="alloc dsname('OPUSER.PT')"
   ```

   Windows systems:

   ```
   ft locfile zospart!remfile OPUSER,account,password
    -rs="alloc dsname('OPUSER.PT')"
   ```

8. Example of specifying UNC names on Windows systems:

   ```
   ft \\Win01\dir\file ux2!file sendfile
   ```

9. Example of specifying domain user IDs in a remote Windows system:

   ```
   ft file2 Win01!file2 mydomain\\miller,,secret (local Unix system)
   ```

   ```
   ft file2 Win01!file2 mydomain\miller,,secret (local Windows system)
   ```

10. This example shows the use of restartable pre- and postprocessing commands. The

    local directory *dir*, along with all its files, is to be transferred to a remote Unix computer using the symbolic name *ftunix.* The current version of openFT should also be running on the remote computer. After the transfer, dir should be available on the remote system under the ID to which the access admission *copydir1* belongs. The directory *dir* must be located on the local computer in the home directory (on Unix systems: value of the *$HOME* variable). Please note that no file name prefix is allowed to be defined in the profile. Details on *ft_tar* are located in the section “ft_tar” .

    ```
    ft "|&ft_tar -cf - dir" ftunix!"|&ft_tar -xf - " copydir1 -b
    ```

11. The directory

    Docs is sent to the Windows system *ftwin*:

    ```
    ft -d Docs ftwin!C:\Software\Docscopy miller,,secret
    ```

    The remote directory *C:\Software* must exist. The directory *Docscopy* is created if it does not yet exist, otherwise it is overwritten (*-o* (overwrite) is default).

3.2 ftaddlic

Note on usage

Function: Add license key

User group: FT administrator

Functional description

You can use *ftaddlic* to add a license key. This officially licenses the associated product or associated product component.

The following types of license key exist:

- Basic key that specified the openFT product in question. There may be additional optional license keys depending on the basic license key.
- One or more optional license keys which make it possible to activate additional product components such as, for example, the FTAM protocol or FTP protocol.

The basic key must always be added first. Only then is it possible to add optional license keys in cases where such optional license keys are available for the licensed openFT product.

If this type of basic key is already present then it is overwritten by the *ftaddlic* command. For example, if the new basic key prohibits the optional extensions FTAM or FTP then any license keys that are present for FTAM and FTP are deleted.

You can display the existing license keys using the *ftshwlic* command.

> ℹ If openFT is installed without a basic key then openFT runs as a demo version with full functionality for 30 days. This demo version may only be used for evaluation purposes!

Format

ftaddlic -h |

<license key>

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

license key

License key consisting of 5 groups of characters each.

Messages of the ftaddlic command

If the license key is not accepted, a self-explanatory message is output. In this case, the exit code is not equal to 0. Check your entry for typing errors.

Example

```
ftaddlic 12345-12345-12345-12345-12345
```

3.3 ftaddptn

Note on usage

Function: Enter a partner in the partner list

User group: FT administrator

Functional description

You use the *ftaddptn* command to enter a partner system in the local system's partner list.

Format

ftaddptn -h |

[<partner name 1..8>]
-pa=<partner address 1..200>
[-id=<identification 1..64> | -id=]
[-ri=<routing info 1..8> | -ri=@i | -ri=]
[-ptc=i | -ptc=a | -ptc=]
[-sl=1..100 | -sl=p | -sl=]
[-pri=l | -pri=n | -pri=h]
[-st=a | -st=d | -st=ad]
[-ist=a | -ist=d]
[-am=y | -am=n]
[-rqp=p | -rqp=s]
[-rco=n | -rco=f | -rco=]
[-tr=n | -tr=f | -tr=]
[-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]] (Linux systems)
[-sif=<0..3>:<0..3>[,<0..3>:<0..3>]..[,<0..3>:<0..3>]] (Windows systems)
[-kl= | -kl=FTOPT | -kl=0 | 768 | 1024 | 2048 | 3072 | 4096]
[-klmin= | -klmin=FTOPT | -klmin=0 | 768 | 1024 | 2048 | 3072 | 4096]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner name

This is the name to be used to enter the partner system in the partner list. The name may consist of 1 to 8 alphanumerical characters. The first character must be a letter and no distinction is made between uppercase and lowercase. The name can be chosen freely and need only be unique within openFT.

partner name not specified

Specifies that the partner is a dynamic partner.

I.e. you assign one or more attributes that are different from the corresponding default values, e.g. *-tr=n* (activate trace).

Please note:

- Security level based on the partner setting (*-sl=p*) is the default setting for free dynamic partners and therefore does not count as a differently set attribute.
- In contrast, security level based on the operating parameter setting (*-sl=*; without parameters, default setting for the *ftaddptn* command) is a differently set attribute.

-pa=partner address

You use *-pa* to enter the address of the partner system in the following form:

[protocol://]host[:[port].[tsel].[ssel].[psel]]

host (= computer name) is mandatory; all other specifications are optional.

For details concerning address specifications, see section “Specifying partner addresses”.

-id=identification | **-id=**

Identification unique in the network of the openFT instance in the partner system. In the case of FTAM partners, it is possible to specify an Application Entity Title in the form n1.n2.n3.n4..mmm as the identification. n1, n2 etc. are positive integer values which describe the "Application Process Title". n1 can only have the values 0, 1 or 2, n2 is restricted to values between 0 and 39 if n1 does not have the value 2. The optional Application Entity Qualifier mmm must be separated from the values of the Application Process Title by two periods. Please refer to the openFT manual "Concepts and Functions", section "Special points for file transfer with FTAM partners" for details.

-id must not be specified for FTP partners!

Identification not specified

The specification of *-id=* means that the *host* (host name) is used for identification for the openFT and FTADM protocol.

Default value: *host* (host name) for the openFT and FTADM protocol, otherwise blank.

-ri=routing info | **-ri=@i** | **-ri=**

If the partner system can only be accessed via an intermediate instance then you specify the address information to be used for routing by the intermediate instance in *routing info.*

@i for *routing info*

The instance identification specified in *-id=* is used as the routing information.

neither *@i* nor *routing info* specified (default value)

The specification of *-ri=* (without parameters) means that the partner system can be accessed directly, i.e. without an intermediate instance.

-ptc=i | **-ptc=a**

You can use *-ptc* to modify the operating parameter setting for sender verification on a partner-specific basis. These settings only affect partners which are connected via the openFT protocol and do not operate with authentication (e.g. partners with openFT V8.0 or earlier).

i (identification)

Deactivates checking of the transport address. Only the partner's identification is checked. The partner's transport address is also not checked even if extended sender verification is globally active (see the *ftmodo* command).

a (address)

Activates checking of the transport address. The partner's transport address is checked even if checking of the transport address is globally deactivated (see the *ftmodo* command).

If the transport address under which the partner logs on is not the same as the entry in the partner list then the request is rejected.

neither *i* nor *a* specified (default value)

-ptc= (without parameters) means that the operating system parameters apply to sender verification.

-sl=1..100 | **-sl=p** | **-sl=**

You use this option to assign a security level to the partner system.

A low security level means that the need for protection towards this partner is low, for instance because the partner's identity has been authenticated using cryptographic methods, which means that you can be certain that the partner is genuinely who they claim to be.

A high security level means that the need for protection towards this partner is high, because the identity of the partner has only been determined on the basis of their address, for instance, and that no authentication has been performed using cryptographic methods.

1..100

Assigns a fixed security level to the partner. 1 is the lowest and 100 the highest security level.

All integers 1 through 100 are permitted.

p

Assigns a security level to the partner on the basis of the partner's attributes, i.e.:

- Security level 10 if the partner has been authenticated.
- Security level 90 if the partner is known in the transport system and is identified by the name it is known by in the transport system.
- Security level 100 if the partner has only been identified by its address.

Security level not specified (default value)

-sl= (without parameters) means that the operating parameter setting for the security level applies (see the *ftmodo* command).

-pri=l | **-pri=n** | **-pri=h**

-pri allows you to specify the priority of a partner in respect of processing requests that have the same request priority. This means that the partner priority only applies in the case of requests that have the same request priority, but that are issued to partners with a different partner priority.

l (low)

The partner is assigned a low priority.

n (normal, default)

The partner is assigned a normal priority.

h (high)

The partner is assigned a high priority.

-st=a | **-st=d** | **-st=ad**

This option allows you to control how locally submitted asynchronous file transfer requests to the specified partner system are processed.

a (active, default value)

Locally submitted asynchronous file transfer requests to this partner system are processed if the asynchronous openFT server is started.

d (deactivated)

Locally submitted asynchronous file transfer requests to this partner system are initially not processed but are stored in the request queue.

ad (automatic deactivation)

Multiple consecutive unsuccessful attempts to establish a connection to this partner system result in its deactivation. The maximum number of unsuccessful attempts is 5. If you want to perform file transfer again with this system, you must explicitly activate it with *ftmodptn -st=a*.

The maximum number of such unsuccessful attempts is 5. After a connection has been established successfully, the counter is reset to 0.

-ist=a | **-ist=d**

This option allows you to control how file transfer requests issued remotely by the specified partner system are processed.

a (active, default value)

File transfer requests issued remotely by this partner system are processed if the asynchronous openFT server is started.

d (deactivated)

Synchronous file transfer requests issued remotely by this partner system are rejected. Asynchronous file transfer requests issued remotely by this partner are stored there and cannot be processed until this partner is activated again with *-ist=a*.

-am=n | **-am=y**

You can use this option to force partner authentication.

n (default value)

Authentication is not forced, i.e. this partner is not restricted with regard to authentication.

y

Authentication is forced, i.e. requests are only processed if the local system is successfully able to authenticate the partner.

-rqp=p | **-rqp=s**

You use this option (rqp = request processing) to control whether asynchronous outbound requests to this partner are always run serially or whether parallel requests are permitted.

p (parallel, default value)

Parallel connections to this partner are permitted.

s (serial)

Parallel connections to this partner are not permitted. If multiple file transfer requests to this partner are pending then they are processed serially. A follow-up request is not started until the preceding request has terminated.

-rco=n | **-rco=f** | **-rco=**

With this option (*rco* = recovery outbound) you can switch on and off the restart function for outbound requests. The parameter has no impact if the implementation of the file transfer protocol (FTP) or type of request (e.g. preprocessing, synchronous orders) does not permit a restart.

n (on)

the restart is always activated for this partner for outbound requests.

f (off)

the restart is deactivated for this partner for outbound requests.

neither *n* nor *f* specified (default value)

-rco= (without parameters) means that the restart operability for outbound requests depends on the setting in the operating parameters, see the *ftmodo* command.

-tr=n | **-tr=f** | **-tr=**

You can use this option to modify the operating parameter settings for the partner selection for the openFT trace function on a partner-specific basis.

n (on)

The trace function is active for this partner. However, a trace is only written if the openFT trace function has been activated via the operating parameters. In this case, this setting for *ftaddptn* takes priority over the partner selection for the trace function in the operating parameters. See the *ftmodo* command, *-tr* and *-trp* options.

f (off)

The trace function is deactivated for this partner.

neither *n* nor *f* specified (default value)

-tr= (without parameters) means that the global setting for partner selection in the openFT trace function applies (see the *ftmodo* command).

You can use the following options to enter an X.25 partner in the local system's partner list. The usage of these options is only allowed if the partner address (option *-pa*) has a valid
X.25 address (beginning with %x25).

-nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] | 2..40

With this option you specify the network address of the X.25 partner system (NSAP = Network Service Access Point). The network address 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
58	1	15	Decimal	0	23	Decimal
59	1	15	Decimal	2	22 (11 x 2)	Hexadecimal

-cl=0/- | **-cl=2/0** | **-cl=2/2**

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 is proposed and is to be accepted as alternative class
2/0	class 2 is proposed and class 0 is to be accepted as alternative class

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

-ws=<1..127>

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

The specification of the window size is optional.

-ps=16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 | 4096

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

The specification of the package size is optional.

-cud=<2..32>

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

The specification of the user data is optional.

-cug=<0..9999>

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

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=75 | 150 | 300 | 600 | 1200 | 2400 | 4800 | 9600 | 9200 | 48000 | 64000 | 128000 | 192000

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

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=y | **-rch=n**

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= <0..3>:<0..3>[,<0..3>:<0..3>]..[,<0..3>:<0..3>] (*Windows systems*)
-sif=[0][,1][,2][,3]..[,15] (*Linux systems*)

Specify the line which is to be used as alternative line if there is a failed connection setup 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. The specification of one or more alternative lines is optional.

A line is uniquely defined via the combination of adapter number: line number under Windows and via the adapter number under Linux.

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

-kl= | **-kl**=FTOPT | **-kl**=0 | 768 | 1024 | 2048 | 3072 | 4096

The parameter can be used to change the length of the RSA key used in encryption. The value of the kl parameter specifies the new RSA key length (RSA-PROPOSED) in bits. The RSA key is only used for the encryption of the AES key agreed between the partners. The configured key length for RSA proposal must be greater than or equal to the specified minimum key length, otherwise a warning will be issued and the proposed key length will be adapted to the minimum key length.

-kl= | **-kl**=FTOPT

Empty string or “FTOPT” option specifies, that key value will be taken from global openFT options displayed via “ftshwo” command. Either both of key values (RSA-PROPOSED and RSA-MINIMUM) need to be set to “FTOPT” or none. Combination of one key having global value and second local partner value (0 … 4096) is not allowed and in such situation’s keys, warning will be issued and keys will be adjusted automatically to “FTOPT value.

-kl=0

-kl=0 explicitly deactivates encryption. If this is set during operation, then any requests with encryption (prior to ftmodo -kl=0) that have been submitted but not yet started are aborted with errors. Any running requests are processed, and their encryption is retained. New requests using encryption are rejected.

-kl=768 | 1024 | 2048 | 3072 | 4096

Standard values for RSA-PROPOSAL encryption. Values from 0 to 4096 take priority over the ones specified in global openFT option visible via ftshwo command.

Default setting following update, export from openFT before version 12.1C70 or not specifying value: -kl=FTOPT.

When only RSA-PROPOSAL is specified during addition of partner (without specifying RSA-MINIMUM), then both parameters will be set to global FTOPT values.

-klmin= | **-klmin**=FTOPT | **-klmin**=0 | 768 | 1024 | 2048 | 3072 | 4096

This option specifies the minimum RSA key length.

-klmin= | **-klmin**=FTOPT

Empty string or “FTOPT” option specifies, that key value will be taken from global openFT options displayed via “ftshwo” command. Either both of key values (RSA-PROPOSED and RSA-MINIMUM) need to be set to “FTOPT” or none. Combination of one key having global value and second local partner value (0 … 4096) is not allowed and in such situation’s keys, warning will be issued and keys will be adjusted automatically to “FTOPT value.

-klmin=0

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

-klmin=768 | 1024 | 2048 | 3072 | 4096

Standard values for RSA-MINIMUM encryption. 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. That means: Since an RSA key set is always created on the open platforms during installation, an RSA key is always sent in the protocol during the subsequent data transfer. If this key is deleted and the partner requests encryption, then the partner rejects the connection with s Session Reject (SRJ) "connection not accepted without encryption".

Values from 0 to 4096 take priority over the ones specified in global openFT option visible via ftshwo command.

Default setting following update, export from openFT before version 12.1C70 or not specifying value: -klmin=FTOPT.

When only RSA-PROPOSAL is specified during addition of partner (without specifying RSA-MINIMUM), then both parameters will be set to global FTOPT values.

During modification of partner, when both keys are set to global “FTOPT” and user modifies only one key to local value, then warning will be prompted and both key values will be adjusted to value specified by user. Additionally, when both keys are set to one of local values and user modifies only one key to global value, then warning will be prompted and both key values will be adjusted to “FTOPT”.

Example for an X.25 partner on Windows systems

```
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
```

Example for an X.25 partner on Linux systems

```
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
```

3.4 ftadm

Note on usage

Function: Execute remote administration command

User group: Users configured as remote administrators on the remote administration server.

A remote administration server must be deployed in order to use this command.

Functional description

The *ftadm* command allows you to act as a remote administrator and administer an openFT instance via a remote administration server. The remote administration server accepts the administration request, checks the authorization and forwards the request to the openFT instance that is to be administered.

In addition, as remote administrator, you can use *ftadm* to query the following information from the remote administration server (see the section “Remote administration commands”):

- You can determine what openFT instances you are authorized to administer and what remote administration permissions you have for these instances.
- You can read the ADM traps that the openFT instances you are administering have sent to the remote administration server. For this to be possible, the remote administration server must also be configured as an ADM trap server for the administered openFT instances. For details, see in the manual "openFT (Unix and Windows systems) - Installation and Operation".

Format

ftadm -h |

[-c]
[-cs=<partner 1..200>]
[-ri=<routing info 1..200>]
[-fnc=t | -fnc=c]
<command 1..8192> | -
[<transfer admission 8..67> | @d]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-c

Specifies whether the user data (i.e. the command and the command output) is to be transferred in encrypted form. It is only possible to specify *-c* if openFT-CR is installed. If openFT-CR is not installed, *-c* is suppressed in the command syntax (*-h*) and a syntax error is generated if *-c* is specified.

-cs=partner

Specifies the name of the remote administration server in the partner list or the address of the remote administration server. The remote administration server must be addressed as an ADM partner. For details, see section “Specifying partner addresses” .

-cs not specified

If you do not specify *-cs*, it is assumed that the local system, i.e. the system at which you logged on, is the remote administration server. This means that you can only omit *-cs* if you enter *ftadm* directly on the remote administration server.

-ri=routing info

Specifies the pathname of the openFT instance that you want to administer. The pathname is configured by the ADM administrator on the remote administration server and is required in order to forward the remote administration request to the openFT instance. You can get the pathname by running the command *ftshwc* on the remote administration server. See the section “Remote administration commands” .

-ri not specified

If you do not specify *-ri*, the command specified under *command* is executed on the remote administration server, e.g. *ftshwc* or *ftshwatp*. See section “Remote administration commands” .

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for the data transferred (administration command and result).

t (transparent, default value)

The data is interpreted on the initiator system and the target system with fixed character codes:

- Unix and Windows systems: ISO8859-1
- BS2000 systems: EBDCDIC DF04-1
- z/OS systems: IBM1047

c (character)

The data is interpreted on the initiator system and the target system with the following character codes:

- Unix systems inbound: Character code set via openFT operating parameter option (*ftmodo -fnccs*). In the ISO646 character set, the coding of this character code must match the coding of ISO8859-1.
- Unix systems outbound: Character code that results from the LOCALE and LANG setting
- Windows systems: UTF-8
- BS2000 systems: EBDCDIC DF04-1
- z/OS systems: IBM1047

-fnc=c is only permitted for administered instances with openFT as of V12.1.

command

The remote administration command to be executed. The maximum command length supported is 8192 characters.

- (dash) for *command*

The dash stands for the standard input *stdin*, i.e. you enter the command at the keyboard. Terminate your input by pressing:

<END> or CTRL+D (*Unix systems*)

STRG+Z at the start of a line, followed by Return (*Windows systems*)

If input is blanked (*@d*) for the *transfer admission*, the system first queries the transfer admission. You can then enter the command.

transfer admission | **@d**

FTAC transfer admission for accessing the remote administration server. Specification of the transfer admission is mandatory if you have specified *-cs*, and must not be specified if you have not specified *-cs*.

@d for *transfer admission*

If you specify *@d* (blanked), the transfer admission is queried on screen after the command has been sent. The entry you make is not displayed, in order to prevent unauthorized persons from seeing the transfer admission.

transfer admission not specified

If you do not specify an FTAC transfer admission, two possible situations arise:

- If you have also specified *-cs*, the transfer admission is queried on screen after the *ftadm* command has been sent.
- If you do not specify *-cs*, i.e. if you enter *ftadm* directly at the remote administration server, your user ID is used as proof that you are authorized to perform remote administration.

3.4.1 Remote administration commands

The following tables list the possible remote administration commands on the individual openFT platforms and on the remote administration server. The Permission column shows the permission required to execute the command as a remote administration command. The following permissions are possible:

FTOP	Read FT access (FT operator)
FT	Read and modify FT access (FT administrator)
FTAC	Read and modify FTAC access (FTAC administrator)

If a number of permissions are specified, e.g. FT | FTAC, it is sufficient if one of these permissions applies, i.e. FT or FTAC.

In the case of a remote administration request, these permissions are compared with the permissions you have on the relevant instance as a remote administrator. The ADM administrator defines the permissions in the configuration data of the remote administration server.

If your permissions are insufficient to execute the remote administration command on a particular instance, the request is rejected, e.g. with:

```
ftadm: Administration request rejected by remote administration server
```

In this event, an ADM log record is written on the remote administration server with a reason code not equal to 0000. The reason code specifies the exact reason for rejection (*fthelp reason-code*).

Commands for openFT partners in BS2000 systems

The commands always have to be prefixed with "/" (slash) before the command name.

BS2000 command	Short forms and aliases	Permission
ADD-FT-PARTNER	ADD-FT-PART FTADDPTN	FT
CANCEL-FILE-TRANSFER	CAN-FILE-T, CNFT NCANCEL, NCAN FTCANREQ	FT
CREATE-FT-KEY-SET	CRE-FT-KEY FTCREKEY	FT
CREATE-FT-PROFILE	CRE-FT-PROF	FTAC
DELETE-FT-KEY-SET	DEL-FT-KEY FTDELKEY	FT
DELETE-FT-LOGGING-RECORDS	DEL-FT-LOG-REC FTDELLOG	FT | FTAC
DELETE-FT-PROFILE	DEL-FT-PROF	FTAC

IMPORT-FT-KEY [2)]	IMP-FT-KEY FTIMPKEY	FT
MODIFY-FILE-TRANSFER	MOD-FILE-T FTMODREQ	FT
MODIFY-FT-ADMISSION-SET	MOD-FT-ADM	FTAC
MODIFY-FT-KEY [2)]	MOD-FT-KEY FTMODKEY	FT
MODIFY-FT-OPTIONS	MOD-FT-OPT FTMODOPT	FT
MODIFY-FT-PARTNER	MOD-FT-PART FTMODPTN	FT
MODIFY-FT-PROFILE	MOD-FT-PROF	FTAC
REMOVE-FT-PARTNER	REM-FT-PART FTREMPTN	FT
SHOW-FILE-TRANSFER	SHOW-FILE-T, SHFT NSTATUS, NSTAT FTSHWREQ	FT | FTOP
SHOW-FT-ADMISSION-SET	SHOW-FT-ADM-S	FTAC
SHOW-FT-DIAGNOSTIC	SHOW-FT-DIAG FTSHWD	FT | FTOP | FTAC
SHOW-FT-INSTANCE	SHOW-FT-INST	FT | FTOP
SHOW-FT-KEY [2)]	FTSHWKEY	FT | FTOP
SHOW-FT-LOGGING-RECORDS	SHOW-FT-LOG-REC FTSHWLOG	FT | FTOP | FTAC
SHOW-FT-MONITOR-VALUES [1)]	SHOW-FT-MON-VAL FTSHWMON	FT | FTOP
SHOW-FT-OPTIONS	SHOW-FT-OPT FTSHWOPT	FT | FTOP
SHOW-FT-PARTNERS	SHOW-FT-PART FTSHWPTN	FT | FTOP
SHOW-FT-PROFILE	SHOW-FT-PROF	FTAC
START-FTTRACE	FTTRACE	FT | FTOP
STOP-FT	FTSTOP	FT

UPDATE-FT-PUBLIC-KEYS	UPD-FT-PUB-KEY FTUPDKEY	FT

[1] As of V11.0

[2] As of V12.0

Commands for openFT partners in z/OS systems

z/OS command	Alias	Permission
FTADDPTN		FT
FTCANREQ	NCANCEL, NCAN	FT
FTCREKEY		FT
FTCREPRF		FTAC
FTDELKEY		FT
FTDELLOG		FT | FTAC
FTDELPRF		FTAC
FTHELP		FT | FTOP | FTAC
FTIMPKEY [2]		FT
FTMODADS		FTAC
FTMODKEY [2]		FT
FTMODOPT		FT
FTMODPRF		FTAC
FTMODPTN		FT
FTMODREQ		FT
FTREMPTN		FT
FTSHWADS		FTAC
FTSHWD		FT | FTOP | FTAC
FTSHWKEY [2]		FT | FTOP
FTSHWINS		FT | FTOP
FTSHWLOG		FT | FTOP | FTAC

FTSHWMON [1)]		FT | FTOP
FTSHWNET		FT | FTOP
FTSHWOPT		FT | FTOP
FTSHWPRF		FTAC
FTSHWPTN		FT | FTOP
FTSHWREQ	NSTATUS, NSTAT	FT | FTOP
FTSTOP		FT
FTTRACE		FT | FTOP
FTUPDKEY		FT

[1)] As of V11.0

[2)]As of V12.0

Commands for openFT partners in Unix and Windows systems

Command	Comment	Permission
fta	up to V10.0	FT
ftaddlic	Windows systems as of V12.0, Unix systems as of V12.1C30	FT
ftaddptn		FT
ftc	up to V10.0	FT
ftcanr		FT
ftcans	openFT-Script command	FT
ftcrek		FT
ftcrep		FTAC
ftdelk		FT
ftdell		FT | FTAC
ftdelp		FTAC
ftdels	openFT-Script command	FT
fthelp		FT | FTOP | FTAC
fti	up to V10.0	FT | FTOP

ftimpk	as of V12.0	FT
ftinfo		FT | FTOP | FTAC
ftmoda		FTAC
ftmodk	as of V12.0	FT
ftmodo		FT
ftmodp		FTAC
ftmodptn		FT
ftmodr		FT
ftremlic	Windows systems as of V12.0, Unix systems as of V12.1C30	FT
ftremptn		FT
ftrs	up to V10.0	FT
ftsetpwd	Windows systems only	FT | FTOP
ftshwa		FTAC
ftshwact	openFT-Script command	FT | FTOP
ftshwd		FT | FTOP | FTAC
ftshwi		FT | FTOP
ftshwk	as of V12.0	FT | FTOP
ftshwl		FT | FTOP | FTAC
ftshwlic	Windows systems as of V12.0, Unix systems as of V12.1C30	FT
ftshwm	as of V11.0	FT | FTOP
ftshwo		FT | FTOP
ftshwp		FTAC
ftshwptn		FT | FTOP
ftshwr		FT | FTOP
ftshws	openFT-Script command	FT | FTOP
ftstop		FT
fttrace		FT | FTOP
ftupdk		FT

Commands on the remote administration server

ftadm allows you to execute the commands *ftshwc* and *ftshwatp* on the remote administration server. When you do so, you must not specify the *-ri* option:

Command	Comment	Permission
ftshwc	Gets the instances that the remote administrator is permitted to administer.	FT | FTOP | FTAC(I.e. all instances are displayed for which the remote administrator has one of these permissions.)
ftshwatp	Outputs the ADM traps of the openFT instances that can be administered.	FT | FTOP(I.e. ADM traps of all instances are displayed for which the remote administrator has one of these permissions.)

3.5 ftalarm

Note on usage

Function: Report failed requests

User group: FT administrator

This command is only available on Unix systems and can be used in multi-user as well as in single-user mode.

Functional description

The *ftalarm* command is used to trigger an alarm if, within two minutes, more FT requests than the number specified by the user fail. The failed FT requests are identified using a return code not equal to 0 for the FTAC log records. *ftalarm* uses the *cron* function.

A separate *ftalarm* call is required for each instance.
Proceed as follows: activate the instance with *ftseti*, and call *ftalarm*.

> If *ftalarm* is started on Solaris systems via SMF then it is inadvisable to start theftalarm command manually since SMF is not informed of any changes. For SMF, *ftalarm* is a so-called transient service, i.e. there is no process that can be monitored.

Format

ftalarm [-h |

-s <number of errors 1..99999999> |
-t | -i]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-s number of errors

starts the *ftalarm* function. When the specified *number of errors* in FTAC log records is exceeded within two minutes, the following message is output on the console and to the *conslog* file:

```
openFTalarm: number or more access control error loggings within 2 minutes
```

The partial string *openFTalarm:* within this message is also guaranteed for future versions of openFT and can be interpreted for automatic processing by system management tools.

The messages are output by the *cron* function at regular intervals and can therefore be delayed by up to one minute when the *ftalarm* function is activated.

conslog is located in the *log* directory of the relevant openFT instance. In the case of the standard instance, the pathname is */var/openFT/std/log/conslog*.

The monitoring function *ftalarm* must be started by the openFT administrator in multiuser mode and by the owner of the invoking instance in single-user mode.

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

-t

terminates the *ftalarm* function.

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.

-i

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

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

Otherwise nothing is output.

3.6 ftbackup

Note on usage

Function: Backing up the openFT configuration

User group: FT administrator

> ℹ Only local system administrators (root on UNIX and users in SYSTEM group on Windows) will be able to execute the command, because all instances need to be backed up, which can have separate FT/FTAC /ADM admins.

Functional description

Since openFT 12.1C80 it is possible to back up the entire openFT configuration with the *ftbackup* command and then restore it with the *ftrestore* command.

The *ftbackup* command saves each instance and its configuration. Private and imported keys as well as CCS files are backed up.

If a user has saved the openFT GUI configuration, this is also backed up.

Licenses are not backed up and must be managed manually by users.

Format

ftbackup -h |

[-s]
<file name 1..512>

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-s

Private RSA keys are not backed up. Normally private RSA keys are also included in the backup. This option excludes them from the backup .

file name

Name of the file or full path of the file where the backed up configuration will be saved. If only the name is specified without specifying the full path, the file will be created where the command was executed. The file format (.zip or .tar) is automatically appended to the end of the file name.

Format of the backup configuration file:

```
Backupfile.tar (or .zip)
         instances
             instance_information
             std
                              ftshwobackup.txt
                  ftshwptnbackup.txt
                  ftprofilesbackup
                  syskpl
                  ftserverbackup
                  sysccs
                     ... (ccs files, if they exist)
                  syskey
                     ... (key files, if they exist)
             instance1
                               ftshwobackup.txt
                  ftshwptnbackup.txt
                  ftprofilesbackup
                  syskpl
                  ftserverbackup
                  sysccs
                     ... (ccs files, if they exist)
                  syskey
                     ... (key files, if they exist)
             ... (more following instances)
         users
             user1
                  <saved_GUI_config>(format depending on system)
             user2
                  <saved_GUI_config>(format depending on system)
         ...(more following users)
             list_users
```

single-user mode on UNIX systems

It is possible to back up the openFT configuration in single-user mode on UNIX systems.

If the configuration was saved in single-user mode, it must be restored in single-user mode. This is not possible in multi-user mode. Likewise, a configuration saved in multi-user mode cannot be restored in single-user mode .

3.7 ftcanr

Note on usage

Function: Cancel asynchronous requests

User group: FT user and FT administrator

Functional description

You can use the *ftcanr* command to cancel asynchronous requests which are in the course of being processed or which are waiting to be processed in the request queue. As an FT user, you can only cancel requests entered under your own login name.

The FT administrator can cancel any requests. In addition, as administrator you can delete requests unconditionally, i.e. without negotiating with the partner system.

If file transfer requests have already been started, the status of the destination file may be undefined.

Format

ftcanr -h |

[-f]
[-ua=<user ID> | @a]
[-ini=l | -ini=r | -ini=lr | -ini=rl]
[-pn=<partner 1..200>]
[-fn=<file name 1..512>]
<request ID 1..2147483647> [<request ID 1..2147483647> ...] | @a

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-f

Allows you to delete the request unconditionally from the local request queue, i.e. without negotiation with the partner system. Note that this can cause requests with an undefined state to arise in the partner's request queue.

You can only call this option as FT administrator. The precondition is that the request was first cancelled with *ftcanr* without the option *-f*.

-ua=user ID | **@a**

You use *-ua* to indicates the user ID for which requests are to be cancelled.

user ID

The user can only specify his/her own login name.

The FT administrator can specify any login name.

@a

This option is only significant for the FT administrator. The FT administrator can specify *@a* to cancel the requests of all the login names.

-ua= not specified

Your login name is used as the selection criterion. Exception: The FT administrator has called the command and specified transfer IDs. In this case, the default is *@a*.

-ini=l | **-ini=r** | **-ini=lr** | **-ini=rl**

You use *-ini* to indicate the initiating party for which you want to cancel requests. You can specify *l*, *r*, *lr*, *rl*:

l

Only requests initiated locally are cancelled.

r

Only requests initiated remotely are cancelled.

lr, rl

Both local and remote requests are cancelled.

-ini not specified

The initiator is not used as a selection criterion (corresponds to *lr* or *rl*).

-pn=partner

You use *-pn* to specify the partner system for which you want to cancel requests. *Partner* is the name or address of the partner system. You should specify the partner in the same form as in the request allocation or as in the output from the *ftshwr* command.

-fn=file name

You use *-fn* to specify the name of the file for which requests are to be cancelled. Requests which access this file in the local system are cancelled.

You must specify the file name which was used when the request was issued and which is output for the *ftshwr* command. Wildcards are not permitted in file names.

request ID1 [request ID2] [request ID3] ... | **@a**

For *request ID*, enter the number of the request to be cancelled. Leading zeros may be omitted. The request identification *request ID* may be obtained from the request receipt acknowledgment displayed on the screen, or using the *ftshwr* command if you have forgotten the *request ID*. You can also specify a number of request identifications at the same time.

If, in addition to *request ID*, you specify other selection criteria, a request with the specified *request ID* is only cancelled if it also satisfies the other conditions.

@a specified as *request ID*

@a selects all requests.

If request IDs were specified and the other selection criteria specified are not satisfied by the requests, the request is not cancelled and the following new error message is issued:

`ftcanr: Request request ID not found`

request ID is the identification of the last unsuitable request.

Examples

1. The asynchronous request with request identification 65546 should be deleted.

   ```
   ftcanr 65546
   ```

2. All local requests to the partner *ux1* which relate to the file *file1* should be deleted.

   ```
   ftcanr -pn=ux1 -fn=file1 -ini=l @a
   ```

3.8 ftcans

Note on usage

Function: Cancelling an openFT-Script request

User group: FT user and FT administrator

Functional description

ftcans allows you to cancel openFT-Script requests that have not yet been concluded. You can cancel either a specific openFT-Script request or all the openFT-Script requests for a user. This also cancels any file transfer requests started by the specified openFT-Script requests which are currently running. This may take a little time. The status of the openFT-Script request is then set to "cancelled" to prevent any restart.

If the openFT-Script request that is to be cancelled is currently being processed then the following message is output at stderr:

`ftcans: Cancellation request for ftscript id` *ftscript id* `started`

If the request has been started but not yet processed then the following message is sent to stderr:

`ftcans: ftscript id` *ftscript id* `cancelled.`

Format

ftcans -h |

[-u=<user ID>] <ftscriptid> | @a

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-u=user ID

User ID under which the search for the openFT-Script request that is to be cancelled is performed.

Only FT administrators may input a user ID.

The default value is the calling party's user ID.

ftscriptid

Identification of the openFT-Script request. This is output if the openFT-Script request is started via an ftscript command.

You can use the wildcard symbols *?* and * in der *ftscriptid*. This cancels all openFT-Script requests that match the wildcard pattern.

?

is interpreted as any single character.

*

is interpreted as any number of characters.

If you use wildcards, enclose the *ftscriptid* specification in single quotes so that the wildcard symbols are not interpreted by the shell.

@a means that all the user's openFT-Script requests are to be cancelled.

3.9 ftcredir

Note on usage

Function: Create remote directories

User group: FT user

Functional description

You use *ftcredir* to create a new directory on a remote system. This is only possible if the remote system supports this function.

Format

ftcredir -h |

<partner 1..200>![<file name 1..512>]
[<transfer admission 8..67> | @n | @d |
<user ID 1..67>[,[<account 1..64>][,[<password 1..64>]]]][-fnc=t | -fnc=c]
[-p=[<management password 1..64>]]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner**!**[file name]

Specifies what directory is to be created on what computer.

partner

Partner is the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses” .

file name

Name of the directory that is to be created. You can specify the name absolutely or relative to the remote login authorization. If the name in the remote system is predefined by an admission profile then it may not be specified here.

If openFT (BS2000) is running on a partner system then an empty PLAM library is created.

transfer admission | **@n** | **@d** |
user ID[**,**[account][**,**[password]]]

Before you can modify the attributes of a file on a remote system, you must first identify yourself at the system. To do this, you need an authorization in the syntax used at the remote system. You can specify this transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization in the syntax used by the remote system (user ID, possibly together with account and/or password).

For details, see section “Entering the authorization data for the partner system” .

@n for *transfer admission*

With *@n* you specify that the remote system does not demand a login authorization.

@d for *transfer admission*

If you specify *@d* (blanked) then the transfer admission is queried on the screen after the command has been sent. Your input is invisible to prevent unauthorized persons from seeing the transfer admission.

A binary password or binary transfer admission must be specified in hexadecimal format, see section “Entering commands”.

password not specified

If you omit a password which is required for authorization then it is queried on the screen after the command has been sent. Your input is invisible to prevent unauthorized persons from seeing the password.

Please note that you still have to enter the commas, e.g.:

```
ftcredir partner!file identification,,
```

or

```
ftcredir partner!file identification,account,
```

neither *transfer admission* nor *user ID* specified

This has the same effect as *@d*, i.e. the transfer admission is queried on the screen after the command has been sent. openFT always interprets your (hidden) input as a transfer admission and not as a user ID.

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for the remote directory name (*file name*).

t (transparent, default value)

Specification of the directory name for the remote system in transparent mode (compatible to the previous versions).

c (character)

Specification of the directory name for the remote system in character mode. The name is interpreted according to the character code of the remote system, i.e. for Unix partners according to the openFT operating parameter option (*ftmodo -fnccs)* that has been set there.

-fnc=c is only permitted for partners as of openFT V12.1.

-p=[management password]

If you want to create a new directory in a password-protected PLAM library then you must specify the password here.

The password can also be specified in hexadecimal format, see section “Entering commands”. This is of relevance in the case of a connection with openFT (BS2000) since it is possible to define hexadecimal passwords in BS2000.

management password not specified

If you specify *-p=* then the password is queried on screen after the command has been sent. Your input is invisible to prevent unauthorized persons from seeing the password.

Examples

1. In the remote Unix system *ux1*, you want to create the directory *dir1*. The identification in *ux1* is protected via the transfer admission *userremote*.

   ```
   ftcredir ux1!dir1 userremote
   ```

2. In the remote Windows system *win1*, you want to create the directories *dir1\dir2* and *dir2* is to be a subdirectory of *dir1*. Neither of these directories exists yet. The directories are to be created in the existing directory *exdir* under the ID *jerry* with the password *secret*.

   To do this, you enter the following commands:

   ```
   ftcredir win1!exdir/dir1 jerry,,secret
   ```

   ```
   ftcredir win1!exdir/dir1/dir2 jerry,,secret
   ```

   The first command is necessary because if you only entered the second command (`ftcredir win1!exdir /dir1/dir2 jerry,,secret`) then the directory *dir1* will not yet exist in the remote system and you will see the error message:

   ```
   Remote system: Higher-level directory not found
   ```

3. In the remote BS2000 system *bs2*, you want to create the PLAM library *user.lib*, the ID is *jimbs2* with the account *j123456* and the password *jimpass*.

   ```
   ftcredir bs2!user.lib jimbs2,j123456,jimpass
   ```

3.10 ftcrei

Note on usage

Function: Create or activate an instance

User group: FT administrator

Functional description

The *ftcrei* command allows you to create a new instance or re-activate a deactivated instance.

If the specified instance file tree does not yet exist, it is created.

When the instance file tree is created, the operating parameters and the profile files are initialized in the same way as for a new installation.

Notes for Unix systems

- When the instance file tree is created, in the case of Solaris with SMF, a manifest is generated and entered in SMF, see the manual "openFT (Unix and Windows systems) - Installation and Operation".

  In all other cases the startup and shutdown files are initialized in the same way as for a new installation.
- When an instance is created, the instance file tree is linked to the */var/openFT* directory with the resources of an instance.

If the instance file tree already exists, *ftcrei* checks the version. If the instance file tree was created using an older version of openFT, it must first be updated using the *ftupdi* command before it can be reactivated.

Important notes for when using multiple instances

- Use of several openFT instances is only possible using the TCP/IP transport system. If you would like to use several instances and are working with CMX with TNS activated (*ftmodo -cmx=y -tns=y*), you must delete all openFT-specific TNS entries that are not TCP/IP compliant (i.e. all except for LANINET and RFC1006).
- You must explicitly assign an individual address to all instances using *-addr*.
- If the instance is to be authenticated in partner systems, it must have a unique instance ID assigned to it (using *fta -id=*). In addition, a public key for the instance must be made available to the partner systems.
- For Windows systems applies: In all newly generated instances, the operating parameter option *Start Asynchronous openFT Server Automatically* is deactivated. You can change this setting in the openFT Explorer by choosing the *Operating Parameters* command in the *Administration* menu and going to the *General* tab.

Format

ftcrei -h |

<instance 1..8> [<directory 1..128>][-addr=<host name>] [-ua=<user ID 1..32>]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

instance

Name of the instance to be created or reactivated. When reactivating an instance in multi-user mode *root* is always the owner of the instance. If until now the owner was not *root*, *root* is made to openFT and FTAC administrator; and if the previous owner was the ADM administrator, this permission is then transferred to *root*. Instance names have a maximum length of 8 characters. The permitted characters are A-Z, a-z and 0-9, and the first character must not be numerical.

The instance name must not be confused with the instance ID (see *ftmodo -id=*).

directory

Directory in which the instance file tree is to be located. The directory must not yet exist.

If you do not specify *directory*, the instance file tree is by default created in:

`/var/openFT/.` *instance* (Unix systems)

%ProgramData%\Fujitsu Technology Solutions\openFT\var\instance (Windows systems)

-addr=host name

Internet host name by which the instance is addressed. If your system has a DNS name, you should specify the full DNS name. openFT then uses the first 8 characters of the first part of the name (the host name qualifier) as the processor name (*ftmodo -p=*) and the entire name as the instance ID (*ftmodo -id=*).

-ua=<user ID 1..32>

With this parameter *root* specifies the owner of a new instance in single-user mode on Unix systems. The parameter is only allowed when creating an instance in single-user mode. The parameter is not allowed in multi-user mode or when reactivating an instance in single-user mode.

Messages of the ftcrei command

If *ftcrei* could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case.

Examples on Unix systems

1. The instance *inst1* is to be newly created in the directory */cluster/inst1.* The DNS name is *hugo.abc.net*. The directory */cluster/inst1* is not allowed to exist.

   ```
   ftcrei inst1 /cluster1/inst1 -addr=hugo.abc.net
   ```

   Where the operational parameter *ftmodo -p=* is *hugo* and *ftmodo -id=* is *hugo.abc.net*.
2. The existing instance *inst2* from the directory */cluster/inst2* is to be re-activated. No host name may be specified.

   ```
   ftcrei inst2 /cluster/inst2
   ```

For examples on Windows systems, please refer to the manual "openFT (Unix and Windows Systems) - Installation and Operation".

3.11 ftcrek

Note on usage

Function: Create key pair set

User group: FT administrator

Functional description

You use this command to create a key pair set for the authentication of your openFT instance in partner systems (RSA procedure). For more information on administering keys, see the manual "openFT (Unix and Windows systems) - Installation and Operation".

If the maximum number of key pair sets is exceeded you get the error message:

```
ftcrek: Maximum number of key pairs exceeded
```

Format

ftcrek [-h]

Description

-h

Displays the command syntax.

3.12 ftcrep

Note on usage

Function: Create an FT profile

User group: FTAC user and FTAC administrator

Functional description

ftcrep stands for "create profile". This command can be used by any user to set up FT profiles for his or her login name.

The FTAC administrator can also set up FT profiles for other login names, either with or without defining a transfer admission.

When it is created, the profile is given a timestamp that is updated each time the profile is modified (e.g. using *ftmodp*).

Note for Windows systems

> Note that the owner of an admission profile can only use their profile if they have stored their user password in openFT. The *ftsetpwd* command is available for this purpose. Alternatively, choose the *User Password...* command from the *Administration* menu of the openFT Explorer.

Format

ftcrep -h |

<profile name 1..8> | @s
<transfer admission> | @n
[-ua=[<user ID>] [,[<password> | @n]]]
[-v=y | -v=n] [-d=yyyymmdd]
[-u=pr | -u=pu]
[-priv=y | -priv=n]
[-iml=y | -iml=n]
[-iis=y | -iis=n] [-iir=y | -iir=n]
[-iip=y | -iip=n] [-iif=y | -iif=n]
[-ff=[t][m][p][r][a][l] | -ff=c]
[-dir=f | -dir=t | -dir=ft]
[-pn=<partner 1..200>,...,<partner(50) 1..200> | -pn=]
[-fn=<file name 1..512> | -fn=]
[-fnp=<file name prefix 1..511>]
[-ls= | -ls=@n | -ls=<command1 1..1000>]
[-lsp=<command2 1..999>] [-lss=<command3 1..999>]
[-lf= | -lf=@n | -lf=<command4 1..1000>]
[-lfp=<command5 1..999>] [-lfs=<command6 1..999>]
[-wm=o | -wm=n | -wm=e | -wm=one]
[-c=y | -c=n]
[-cm=y | -cm=n]
[-txt=<text 1..100>]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

profile name | **@s**

is the name you wish to assign to the FT profile. This name can be used to address the FT profile, for example when it is to be modified or deleted. Be sure not to confuse the profile name with the transfer admission (see below). The profile name must be unique among all the FT profiles under your login name, or FTAC will reject the *ftcrep* command and issue the message *FT profile already exists*. To have the profile names you have already assigned displayed, you can issue the *ftshwp* command (without options).

@s for *profile name*

Creates the standard admission profile for the user ID. You must specify *@n* as the transfer admission, because a standard admission profile in a request is addressed using the user ID and password.

You must not specify the options *-v*, *-d* and *-u* with a standard admission profile.

transfer admission | **@n**

replaces the login authorization for your system otherwise required in inbound requests. When this transfer admission is specified in an FT request, FTAC applies the access rights defined in this FT profile.

transfer admission

The transfer admission must be unique within your system so that there are no conflicts with transfer admissions defined by other FTAC users with other access rights. If the transfer admission you select has already been assigned, FTAC rejects the *ftcrep* command and issues the message:

```
Transfer admission already exists.
```

You can also define a binary admission with any characters, including non-printing characters, see section “Entering commands”.

As the FTAC administrator, you can assign a transfer admission for yourself under your own login name or for any other user.

In this case, if you do not have FT administrator permissions, you must specify the complete login authorization, i.e. the user ID and password.

@n for *transfer admission*

By entering *@n*, you create an FT profile without a transfer admission.

As the FTAC administrator, by specifying *@n*, you can create FT profiles for other login names without having to define transfer admissions.

If the profile is not a standard admission profile, it is locked until you or the owner of the profile assign a valid transfer admission with *ftmodp*.

You must specify *@n* when you create a standard admission profile.

transfer admission not specified

If you do not specify the transfer admission in the command, FTAC prompts you to enter the transfer admission after the command has been sent. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the program expects you to enter the transfer admission a second time as an entry check.

-ua=[user ID][**,**[password | @**n**]]

As the FTAC administrator use *-ua* to specify the user IDs for which you want to set up FT profiles.

user ID

The user without administrator privileges can specify only his own user ID.

As the FTAC administrator, you can specify any user ID.

,password

Specifies the password of the login name. A binary password must be specified in hexadecimal format, see section “Entering commands”. The FT profile for the login name is only valid while the password is valid for the login name. If the password is changed, the profile can no longer be used.

If you want to assign an FT profile for another user and also assign a transfer admission for that profile, you must specify the login name as well as the password for that login name if you do not have FT administrator privileges.

@**n** for *password*

This entry may only be specified by the FTAC administrator. With *@n*, you cannot assign any transfer admission for the FT profile if you do not have FT administrator privileges.

comma only (,) no *password*

Entering comma (,) without *password* causes FTAC to query the password on the screen after the command is entered. The entry is not displayed to prevent unauthorized persons from seeing the transfer admission.

user ID only (without comma and no *password*) specified

the profile is valid for all the passwords for *user ID*.

-ua= specified or *-ua* not specified

the FT profile is created for the individual login name.

-v=y | **-v=n**

defines the status of the transfer admission.

Possible values are:

y (default value)

the transfer admission is not disabled (it is valid).

n

the transfer admission is disabled (it is not valid).

-v must not be specified with a standard admission profile.

-d=yyyymmdd

specifies the period during which the transfer admission can be used. The FT profile is disabled when this period has expired.

You can specify an eight-digit date (e.g. 20170602 for June 2, 2017). The transfer admission can no longer be used after 00:00 hours on the specified day. The largest possible value which can be specified as the date is 20380119 (January 19, 2038).

-d must not be specified with a standard admission profile.

-d not specified (default value)

no period is specified for using the transfer admission.

-u=pr | -**u=pu**

with *-u*, you can control how FTAC reacts when someone attempts to create an FT profile with the same transfer admission. Normally, the transfer admission must be disabled immediately.

Transfer admissions that do not require as much protection are designated as public. This means that they are not disabled, even if a user attempts to assign another transfer admission of the same name.

pr (default value)

the transfer admission is disabled as soon as someone under another login name attempts to specify a transfer admission of the same name (private). In this case, the values for *-u* and *-d* are set to their default values at the same time.

pu

the transfer admission is not disabled, even if someone attempts to specify a transfer admission of the same name (public).

-u must not be specified with a standard admission profile.

-priv=n | **-priv=y**

is used by the FTAC administrator to grant privileged status to FT profiles.

As a user, you can only revoke an existing privileged status, *y* is not permitted.

n (default value)

The FT profile is not privileged (initially).

y

For the FTAC administrator only: The FT profile is privileged.

-iml=y | **-iml=n**

-iml (ignore max. level) is used to specify whether the FT profile is to be restricted by the values in the admission set. You can override your own entries (the MAX. USER LEVELS) for requests using this FT profile.

If the FT profile is also privileged by the FTAC administrator, the values of the FTAC administrator (the MAX. ADM LEVELS) can also be ignored. This FT profile would then allow *inbound* basic functions which are disabled in the admission set to be used. Possible values are:

y

allows the values in the admission set to be ignored.

n (default value)

restricts the functionality of the profile to the values in the admission set.

-iis=y | **-iis=n**

-iis (ignore inbound send) allows the value for the basic function *inbound send* in the admission set to be ignored (see *-iml* for details).

y

allows the basic function *inbound send* to be used even if it is disabled in the admission set. At the same time, the component "display file attributes" of the basic function *inbound file management* can also be used.

Specifying this option is enough as long as the basic function *inbound send* was disabled by the user, but if it was disabled by the FTAC administrator, it is also necessary that he/she grant privileged status to the FT profile.

n (default value)

restricts the profile to the value in the admission set for the basic function *inbound send*.

-iir=y | **-iir=n**

-iir (ignore inbound receive) allows the value for the basic function *inbound receive* in the admission set to be ignored (see *-iml* for details).

y

allows the basic function *inbound receive* to be used even if it is disabled in the admission set. At the same time, components of the basic function *inbound file management* can also be used (see table at *-iif*).

Specifying this option is enough as long as the basic function *inbound receive* was disabled by the user, but if it was disabled by the FTAC administrator, it is also necessary that he/she grant privileged status to the FT profile.

n (default value)

restricts the profile to the value in the admission set for the basic function *inbound receive*.

-iip=y | **-iip=n**

-iip (ignore inbound processing) allows the value for the basic function *inbound followup processing* + *preprocessing* + *postprocessing* in the admission set to be ignored
(see *-iml* for details).

y

allows the basic function *inbound follow-up processing* + *preprocessing* + *postprocessing* to be used even if it is disabled in the admission set.

Specifying *-iip=y* is enough as long as the basic function *inbound follow-up processing* + *preprocessing* + *postprocessing* was disabled by the user. But if it was disabled by the FTAC administrator, it is also necessary that he/she grant privileged status to the FT profile.

n (default value)

restricts the profile to the value in the admission set for the basic function *inbound follow-up processing + preprocessing + postprocessing*.

-iif=y | **-iif=n**

-iif (ignore inbound file management) allows the values for the basic function *inbound file management* in the admission set to be ignored (see *-iml* for details).

y

allows the basic function *inbound file management* to be used even if it is disabled in the admission set. Specifying this option is enough as long as the basic function *inbound file management* was disabled by the user, but if it was disabled by the FTAC administrator, it is also necessary that he/she grant privileged status to the FT profile.

n (default value)

restricts the profile to the value in the admission set for the basic function *inbound file management*.

The following table shows which subcomponents of the file management can be used under which conditions.

Inbound file management function	**Values of the admission set or extension in profile**
Display file attributes	Inbound Send (IBS) enabled
Modify file attributes	Inbound Receive (IBR) **and** Inbound File Management (IBF) enabled
Rename files	Inbound Receive (IBR) **and** Inbound File Management (IBF) enabled
Delete files	Inbound Receive (IBR) enabled **and** Write mode = overwrite in profile
Display directories	Inbound File Management (IBF) enabled
Create, rename and delete directories	Inbound File Management (IBF) enabled **and** direction= from partner in profile

-ff=[**t**][**m**][**p**][**r**][**a**][**l**] | **-ff=c**

-ff defines the FT function for which the FT profile can be used. With the exception of *c*, these letters can be combined in any way (*tm, mt, mr,* ...). *c* must not be combined with other values.

t (transfer)

The FT profile can be used for the file transfer functions "Transfer files", "Display file attributes" and "Delete files".

m (modify file attributes)

The FT profile can be used for the file transfer functions "Display file attributes" and "Modify file attributes".

p (processing)

The FT profile can be used for the file transfer functions "File Preprocessing" or "File Postprocessing". The FT function "Transfer files" must also be permitted.

Specification of *p* has no significance for profiles with a file name prefix (*-fnp=*) or a file name (*-fn=*) since, in this case, the first character of the file name or file name prefix decides whether the profile can only be used for preprocessing and postprocessing ("|") or only for file transfer/file management (no "|").

The use of follow-up processing is not controlled by *-ff=*, but by *-lf=* and *-ls=*.

r (read directory)

The FT profile can be used for the file transfer functions "Display directories" and "Display file attributes".

a (administration)

The admission profile is allowed to be used for the "remote administration" function. This means that it authorizes a remote administration server to access the local openFT instance. To do this, the associated transfer admission must be configured in the remote administration server.

-ff=a may only be specified by the FT administrator or FTAC administrator.

l (logging)

The admission profile is allowed to be used for the "ADM traps" function. This allows another openFT instance to send its ADM traps to the remote administration server via this profile. This specification only makes sense if the local openFT instance is flagged as a remote administration server (*ftmodo -admcs=y* command).

-ff=l may only be specified by the FT administrator.

c (client access)

The admission profile is allowed to be used for the "access to remote administration server" function (ADM profile). This allows a remote administrator on a remote computer to use this profile to access the local remote administration server and issue remote administration requests. The local openFT instance must be flagged as a remote administration server (*ftmodo -admcs=y* command).

The value *c* must not be combined with any other value. *-ff=c* may only be specified by the ADM administrator.

-ff not specified

Corresponds to the specification *-ff=tmr*, i.e. the admission profile can be used for all file transfer functions other than "file processing", but cannot be used for remote administration functions (*a*, *c*) and ADM traps (*l*).

-dir=f | **-dir=t** | **-dir=ft**

specifies for which transfer direction(s) the FT profile may be used.

f

allows data transfer only from a remote system to the local system.

t

allows data transfer only from a local to a remote system. Directories cannot be created, renamed nor deleted.

ft, tf

both transfer directions are allowed.

-dir not specified

transfer direction is not restricted in the FT profile.

-pn=partner[**,**partner2**,** ...] | **-pn=**

You use *-pn* to specify that this admission profile is to be used only for FT requests which are processed by a certain partner system. You can specify the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

You can specify more than one partner system (maximum 50) with a maximum total of 1000 characters.

-pn not specified (or **-pn=**)

means that any remote system can use the FT profile.

-fn=file name | **-fn=**

-fn specifies which file under your login name may be accessed using this FT profile. If you specify a fully qualified file name, only the file with this name can be transferred.

If the file name ends with %unique or %UNIQUE, this string is replaced during the file transfer by a string which changes for each new call, see section “Entering commands”.

If *file name* starts with a "|" (pipe character) then it is interpreted as a preprocessing or postprocessing command.

-fn not specified (or **-fn=**)

omitting *-fn* means that the FT profile allows unrestricted access to all files under the login name (exception see *-fnp*).

-fnp=file name prefix

restricts access to a set of files whose names begin with the same prefix. FTAC adds the character string specified as *file-name-prefix* to the file name in the request and attempts to transfer the file with the expanded name.

Example:

- Unix systems: If this option is specified as *-fnp=scrooge/* and the request contains the file name *stock*, the file is transferred as *scrooge/stock*.
- Windows systems: If this option is specified as *-fnp=scrooge*\ and the request contains the file name *stock* , the file is transferred as *scrooge**stock* .

In this way, you can designate the files you have released for transfer. If the *-fnp* option was used to specify a prefix, the file name specified in the request must not contain a directory separator (Unix systems: "/", Windows systems: "\"). This disables (unintentionally) changing directories specifying ../ or ..\. You should also ensure that there is no chance for a symbolic link to cause a jump to another place in the file tree.

%unique or %UNIQUE cannot be used for a file name prefix. In the case of a file transfer request, the user can use a file name ending with %UNIQUE (or %UNIQUE.*suffix* or %unique or %unique.*suffix*) to generate a unique file name with the prefix specified here.

A file name prefix which starts with the | (pipe) character indicates that the admission profile can only be used for file transfer with preprocessing and postprocessing, since the file name created using the prefix and the name specified for the *ncopy* or *ft* command also starts with the | character. In this case, no follow-up commands may be specified.

Exception on Windows systems: The filename prefix under Windows starts with `|cmd /c` or `|&cmd /c` .

filename prefix can be up to 511 bytes in length (for the representation in UTF-8, see section “Entering commands”).

Notes on profiles with preprocessing or postprocessing

- On Unix systems, the shell metacharacters | ; & < > and "newline" may only be specified if they are enclosed in '...' (single quotes) or "..." (double quotes) or if each of them is escaped with "\" (backslash). The character ` (accent grave) and the string $((dollar+open bracket) may only be specified if they are enclosed in '...' (single quotes) or if they are specified directly after a backslash ("\").
- The following strings may not be specified for the name entered in the *ncopy* or *ft* command:
  - .. (two dots)
  - .\ (dot + backslash)
  - .' (dot + single quote, only for Unix systems)

  This makes it impossible to navigate to higher-level directories.

- Special cases
  - You must specify a file name or file name prefix which starts with the string "|ftexecsv " for admission profiles which are to be used exclusively for the *ftexec* command.

    If a command prefix is also to be defined, you must specify it as follows:

    -fnp="|ftexecsv -p= *command prefix* **"**

    (e.g.: `-fnp="|ftexecsv -p=\"ftshwr \""`)

    The same restrictions apply to the command string of the *ftexec* call as to the filename prefix during preprocessing and postprocessing.
  - For admission profiles that are only to be used for getting monitoring data, specify the filename prefix "|*FTMONITOR ". The functions of the profile must permit File Preprocessing (*-ff=tp*). For details, see Example 3.

-fnp not specified

FTAC adds no prefix to the file name.

-ls= | **-ls=@n** | **-ls=**command1

-ls specifies follow-up processing which is to be performed under your login name in the event that file transfer is successful. If *-ls* is specified, no success follow-up processing may be requested in the FT request. Specifying *-ls* only makes sense if you also make an entry for *-lf* (see below) to preclude the possibility than an intentionally unsuccessful request can circumvent the *-ls* entry. If you have defined a prefix for the file name with *-fnp* and plan follow-up processing for this file, you must specify the complete file name here.

@n for *command1*

If *-ls=@n* is specified, no success follow-up processing is permitted in the event of a successful file transfer.

For details on follow-up processing, please refer to section “Commands for follow-up processing” .

-ls not specified (or **-ls=**)

does not restrict follow-up processing in the local system in the event of successful file transfer (however, see also *-lsp* or *-lss*).

-lsp=command2

-lsp defines a prefix for follow-up processing in the local system in the event of successful file transfer. FTAC then adds the character string *command2* to the follow-up processing specified in the FT request and attempts to execute the resulting command.

Example:

- Unix systems: If this option is specified as *-lsp='lpr '* and the request specifies *file1.txt* as follow-up processing, FTAC executes *lpr file1.txt* as follow-up processing.
- Windows systems: If this option is specified as *-lsp="print "* and the request specifies *file1.txt* as follow-up processing, FTAC executes *print file1.txt* as follow-up processing.

Please also bear in mind the information provided on the *-ls* option!

For details on follow-up processing, please refer to section “Commands for follow-up processing” .

-lsp not specified

FTAC adds no prefix to the follow-up processing specified in the request in the event of successful file transfer.

-lss=command3

-lss defines a suffix for follow-up processing in the local system in the event of successful file transfer. FTAC then appends the character string *command3* to the followup processing specified in the FT request and attempts to execute the resulting command.

Example:

- Unix systems: If this option is specified as *-lss=' file2.txt'* and the request specifies *lpr* as follow-up processing, FTAC executes *lpr file2.txt* as follow-up processing.
- Windows systems: If this option is specified as *-lss=" file2.txt"* and the request specifies *print* as follow-up processing, FTAC executes *print file2.txt* as follow-up processing.

Please also bear in mind the information provided on the *-ls* option!

For details on follow-up processing, please refer to section “Commands for follow-up processing” .

-lss not specified

FTAC adds no suffix to the follow-up processing specified in the request in the event of successful file transfer.

-lf=command4 | **@n**

-lf specifies follow-up processing to be executed under your login name if the file transfer is aborted due to an error. If *-lf* is specified, no failure follow-up processing may be requested in the FT request. Making an *-lf* entry only makes sense if you also make an entry for *-ls* (see above) to preclude the possibility that a successful request can circumvent the *-lf* entry. If you have defined a prefix for the file name with *-fnp* and plan follow-up processing for this file, you must specify the complete file name here.

@n for *command4*

If *-lf=@n* is specified, no failure follow-up processing is then permitted in the event of unsuccessful file transfer.

For details on follow-up processing, please refer to section “Commands for follow-up processing” .

-lf not specified

does not restrict follow-up processing in the local system in the event of unsuccessful file transfer (Exception see *-lfp* or *-lfs*).

-lfp=command5

-lfp defines a prefix for follow-up processing in the local system in the event of unsuccessful file transfer. FTAC then sets the character string *command5* in front of the followup processing specified in the FT request and attempts to execute the resulting command.

Example:

- Unix systems: If this option is specified as *-lfp='lpr '* and the request specifies *error.txt* as follow-up processing, FTAC executes *lpr error.txt* as follow-up processing.
- Windows systems: If this option is specified as *-lfp="print "* and the request specifies *error.txt* as follow-up processing, FTAC executes *print error.txt* as followup processing.

Please also bear in mind the information provided on the *-lf* option!

For details on follow-up processing, please refer to section “Commands for follow-up processing” .

-lfp not specified

FTAC sets no prefix in front of the follow-up processing specified in the request in the event of unsuccessful file transfer.

-lfs=command6

-lfs defines a suffix for follow-up processing in the local system in the event of unsuccessful file transfer. FTAC then sets the character string *command6* after the follow-up processing specified in the FT request and attempts to execute the resulting command.

Example:

- Unix systems: If this option is specified as *-lfs=' error.txt'* and the request specifies *lpr* as follow-up processing, FTAC executes *lpr error.txt* as follow-up processing.
- Windows systems: If this option is specified as *-lfs=" error.txt"* and the request specifies *print* as follow-up processing, FTAC executes *print error.txt* as follow-up processing.

Please also bear in mind the information provided on the *-lf* option!

For details on follow-up processing, please refer to section “Commands for follow-up processing” .

-lfs not specified

FTAC sets no suffix after the follow-up processing specified in the request in the event of unsuccessful file transfer.

-wm=o | **-wm=n** | **-wm=e** | **-wm=one**

-wm specifies which write modes may be used in the file transfer request and what they effect.

o (overwrite)

In the FT request of openFT or FTAM partners, only *-o* or *-e* may be entered for write mode. The receive file is overwritten if it already exists, and is created if it does not yet exist.

With FTP partners, *-n* may also be entered if the file does not yet exist.

n (no overwrite)

In the FT request *-o*, *-n* or *-e* may be entered for write mode. The receive file is created if it does not yet exist. If the receive file already exists, the request is not executed.

e (extend)

In the FT request only *-e* may be entered for write mode, i.e. the receive file is extended by appending the transferred file to the end if the receive file already exists. The receive file is created if it does not yet exist.

one (default value)

means that the FT profile does not restrict the write mode.

-c=y | **-c=n**

Precondition: openFT-CR must be installed.

Using *-c*, you can determine whether data encryption is required or forbidden. If the setting in the profile does not correspond to the setting in the request, the request is denied. The setting is not valid for file management requests, since there is no encryption for these requests.

y

Only requests *with* data encryption may be processed using this profile.

n

Only requests *without* data encryption may be processed using this profile.

-c not specified

Data encryption is neither required nor forbidden.

-cm=y | **-cm=n**

Precondition: openFT-CR must be installed.

Using *-cm*, you can determine whether file(s) and/or directory list attributes encryption is required or forbidden. If the setting in the profile does not correspond to the setting in the request, the request is denied.

y

Only requests *with* file(s) and/or directory list attributes encryption may be processed using this profile.

n

Only requests *without file(s) and/or directory list attributes* encryption may be processed using this profile.

-cm not specified

File(s) and/or directory list attributes encryption is neither required nor forbidden.

-txt=text

enables you to store a comment in the FT profile (up to 100 characters).

-txt not specified

the FT profile is stored without a comment.

> **CAUTION!**
>
> If you use the options *-ff=p*, *-fn*, *-fnp*, *-ls* , *-lsp*, *-lss*, *-lf*, *-lfp* or *-lfs*, you must remember
>
> - that a file-name restriction can be bypassed by renaming the file unless followup processing is also restricted;
> - that follow-up processing must always be restricted for both successful and unsuccessful file transfer and, if necessary, equivalent restrictions must exist for any permitted preprocessing;
> - that prefixes for the file name and follow-up processing must be matched to one another;
> - that no symbolic links should occur in the part of your file tree that is referenced by the file name prefix.
> - that restrictions applied to preprocessing, postprocessing, or follow-up processing can be circumvented if it is possible to replace this command with, for example, a "Trojan horse".

Examples

1. You wish to create an FT profile for the following purpose:

   The Duck Goldmines are to be able to send their monthly reports from their computer *goldmine* to the president at head office via file transfer. The file *monthlyreport_goldmine01* is to be printed out after transfer. The command required to create such an FT profile at head office is:

   Unix systems :

   ```
   ftcrep goldmrep fortheboss -d=20171231 -dir=f\
    -pn=goldmine -fn=monthlyreport_goldmine01\
    -ls='lpr monthlyreport_goldmine01' -lf=@n -wm=o
   ```

   Windows systems:

   ```
   ftcrep goldmrep fortheboss -d=20171231 -dir=f
     -pn=goldmine -fn=monthlyreport_goldmine01
     -ls="print monthlyreport_goldmine01" -lf=@n -wm=o
   ```

   The FT profile has the name *goldmrep* and the transfer admission *fortheboss.* It permits only the *monthlyreport_goldmine01* file to be transferred to the bank. Following successful transfer, the file is printed out in the bank. Follow-up processing after unsuccessful file transfer is, however, prohibited. The transfer admission is only valid until December 30, 2017, the FT profile disabled as of 00:00 hours on December 31, 2017.

2. You want to set up the standard admission profile on your user ID in such a way that only the file transfer and file creation functions are possible. This profile can, for instance, be used by FTAM partners that always have to specify the user ID and the password for inbound access.

   The command is as follows:

   ```
   ftcrep @s @n -wm=n -ff=t
   ```

3. You want to define an admission profile *monitor1* that only allows monitoring data to be output. Assign *onlyftmonitor* as the transfer admission. The command is as follows:

   ```
   ftcrep monitor1 onlyftmonitor -ff=tp -fnp="|*FTMONITOR "
   ```

   The purpose of the blank after *FTMONITOR is to automatically separate any options specified during the call from the command. A profile such as this can be used to call the openFT monitor (e.g. using the *ftmonitor* command) and in the *ncopy* command. The admission profile is only valid for communicating via the openFT protocol.

   You will find further details in the section "Monitoring with openFT" in the manual "openFT (Unix and Windows systems) - Installation and Operation".

3.13 ftdel

Note on usage

Function: Delete a file in a remote system

User group: FT user

Functional description

With *ftdel* you can delete files in the remote system.

Format

ftdel -h |

<partner 1..200>![<file name 1..512>]
[<transfer admission 8..67> | @n | @d |
<user ID 1..67>[,[<account 1..64>][,[<password 1..64>]]]]
[-fnc=t | -fnc=c]
[-p=[<management password 1..64>]]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner**!**file name

Specifies which file in which remote system has to be deleted.

partner

Partner is the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

file name

file name can be either absolute or relative to the remote login authorization. If the file name in the remote system has been predefined in an FT profile, it must not be specified here.

If the partner system is running openFT (BS2000), elements from PLAM libraries may also be specified here (Syntax: Libname/Element type/Element name).

transfer admission | **@n** | **@d** |
user ID[**,**[account][**,**[password]]]

In order to execute file management requests in the remote system, you must furnish the remote system with proof of identity. For this purpose, you will need login authorization in the syntax valid for the remote system. You can specify this transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization in the syntax used by the remote system (user ID, possibly together with account and/or password).

For details, see section “Entering the authorization data for the partner system” .

@n for *transfer admission*

By entering *@n* you specify that the remote system requires no login authorization.

@d for *transfer admission*

Specifying *@d* (blanked transfer admission) causes openFT to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission.

A binary password or binary transfer admission must be specified, see section “Entering commands”.

password not specified

Omitting the password necessary for admission causes openFT to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

Nevertheless, you have to specify the commas, e.g.:

```
ftdel file partner!file user-id,,
```

or

```
ftdel file partner!file user-id,account,
```

neither *transfer admission* nor *user ID* specified

causes the same as *@d*, i.e. openFT queries the transfer admission on the screen after the command is entered. Your (blanked) entry is always interpreted as transfer admission and not as user ID.

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for the remote file name.

t (transparent, default value)

Specification of the remote file name in transparent mode (compatible to the previous versions).

c (character)

Specification of the remote file name in character mode. The name is interpreted according to the character code of the remote system, i.e. for Unix partners according to the openFT operating parameter option (*ftmodo -fnccs)* that has been set there.

-fnc=c is only permitted for partners as of openFT V12.1.

-p=[management-password]

If the file in the remote system is protected by a password, you must enter this password here.

A binary password must be entered in hexadecimal format, see section “Entering commands”. This is of relevance for links to openFT (BS2000), because BS2000 supports the definition of hexadecimal passwords.

management password not specified

Specifying *-p=* causes openFT to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

Example

The file *junk* in the BS2000 computer *bs2r1* under login name *jim* with account number *a1234ft* and password *C'pwd'* is to be deleted from your system. The file is protected by the password *abcd*.

```
ftdel bs2r1!unsinn jim,a1234ft,C\'pwd\' -p=C\'abcd\'
```
(Unix systems)

```
ftdel bs2r1!unsinn jim,a1234ft,C'pwd' -p=C'abcd'
```
(Windows systems)

3.14 ftdeldir

Note on usage

Function: Delete remote directories

User group: FT user

Functional description

You can use *ftdeldir* to delete an empty directory on a remote system. For this to be possible, the remote system must support this function.

You can only delete directories which are empty.

Format

ftdeldir -h |

<partner 1..200>![<file name 1..512>]
[<transfer admission 8..67> | @n | @d |
<user ID 1..67>[,[<account 1..64>][,[<password 1..64>]]]]
[-fnc=t | -fnc=c]
[-p=[<management password 1..64>]]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner**!**[file name]

Specifies what directory is to be deleted on what computer.

partner

Partner is the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

file name

Name of the directory that is to be deleted.

You can specify *file name* absolutely or relative to the remote login authorization. If the file name in the remote system is predefined by an admission profile then it may not be specified here.

If openFT (BS2000) is running on a partner system then an empty PLAM library can be specified here. This deletes the PLAM library.

> If the directory or PLAM library is not empty then you can delete the files or elements with *ftdel* before calling *ftdeldir*.

transfer admission | **@n** | **@d** |
user ID[**,**[account][**,**[password]]]

Before you can modify the attributes of a file on a remote system, you must first identify yourself at the system. To do this, you need an authorization in the syntax used at the remote system. You can specify this transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization in the syntax used by the remote system (user ID, possibly together with account and/or password).

For details, see section “Entering the authorization data for the partner system” .

@n for *transfer admission*

By entering *@n* you specify that the remote system requires no login authorization.

@d for *transfer admission*

If you specify *@d* (blanked) then the transfer admission is queried on the screen after the command has been sent. Your input is invisible to prevent unauthorized persons from seeing the transfer admission.

A binary password or binary transfer admission must be specified in hexadecimal format, see section “Entering commands”.

password not specified

If you omit a password which is required for authorization then it is queried on the screen after the command has been sent. Your input is invisible to prevent unauthorized persons from seeing the password.

Please note that you still have to enter the commas, for example:

```
ftdeldir partner!file identification,,
```

or

```
ftdeldir partner!file identification,account,
```

neither *transfer admission* nor *user ID* specified

This has the same effect as *@d*, i.e. the transfer admission is queried on the screen after the command has been sent. openFT always interprets your (hidden) input as a transfer admission and not as a user ID.

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for the remote directory name (*file name*).

t (transparent, default value)

Specification of the remote directory name in transparent mode (compatible to the previous versions).

c (character)

Specification of the remote directory name in character mode. The name is interpreted according to the character code of the remote system, i.e. for Unix partners according to the openFT operating parameter option (*ftmodo -fnccs)* that has been set there.

-fnc=c is only permitted for partners as of openFT V12.1.

-p=[management password]

If the directory is protected by a password in the remote system then you must specify this here.

A binary password must be specified in hexadecimal format, see section “Entering commands”. This is of relevance in the case of a connection with openFT (BS2000) since it is possible to define hexadecimal passwords in BS2000.

management password not specified

If you specify *-p=* then the password is queried on screen after the command has been sent. Your input is invisible to prevent unauthorized persons from seeing the password.

Example

The empty directory *dir1* on the system *host1* is to be deleted:

```
ftdeldir host1!dir1 transferadm
```

3.15 ftdeli

Note on usage

Function: Deactivate an instance

User group: FT administrator

Functional description

The *ftdeli* command allows you to deactivate an instance. This causes the following:

- Unix systems: Removes only the symbolic link in the local */var/openFT* directory
- Windows systems: The instance is removed from the openFT instance administration.

The instance file tree is not changed. The standard instance *std* and the currently set instance can not be deleted.

The deactivation of an instance should as a matter of principle not be done during ongoing openFT operation because 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.

In single-user mode only *root* is permitted to deactivate an openFT instance with the command *ftdeli*.

Format

ftdeli -h |

<instance 1..8>

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

instance

Name of the instance to be deactivated. Using the *ftshwi @a* command displays the names of all instances.

Messages of the ftdeli command

If *ftdeli* could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case.

Examples

1. The instance *inst1* from the directory */CLUSTER/inst1* (Unix systems) or *S:**CLUSTER**inst1* (Windows systems) is to be deactivated on computer *CLUSTER1*, since it has been switched over to *CLUSTER2.* The directory is retained.

   ```
   ftdeli inst1
   ```

2. Instance *inst2* with the directory */CLUSTER/inst1* (Unix systems) or *S:**CLUSTER**inst1* (Windows systems) is to be deleted along with the instance file tree.

   Unix systems:

   ```
   ftdeli inst2
   ```

```
rm -r /CLUSTER/inst2
```

Windows systems:

```
ftdeli inst2
rmdir /S S:\CLUSTER\inst2
```

3. Using *. ftseti* (Unix systems) or *ftseti* (Windows systems), it was changed to instance *inst3*. There, an attempt is being made to deactivate the instance *inst3*.

```
ftdeli inst3
ftdeli: openFT Instance 'inst3' can not be removed.
```

3.16 ftdelk

Note on usage

Function: Delete key pair set

User group: FT administrator

Functional description

You use this command to delete the key pair sets for a reference. Your system can then no longer be authenticated by partner systems which still use the associated public key. For more information on administering keys, see the manual "openFT (Unix and Windows systems) - Installation and Operation".

A key pair set should always be present in your openFT instance as otherwise all requests are run unencrypted, i.e. neither the request description data nor the file contents are encrypted.

Format

ftdelk [-h]

<key reference 1..9999999>

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

key reference

Used to select the key pair set that is to be deleted. You can find the reference in the name of the public key file, see the manual "openFT (Unix and Windows systems) - Installation and Operation".

3.17 ftdell

Note on usage

Function: Delete log record or offline log file

User group: FT, FTAC or ADM administrator.

Functional description

With *ftdell*, you can delete log records for all login names if you are FT, FTAC or ADM administrator. This function is not permitted for FT users.
You can also delete offline log files that you no longer need. You can delete up to 1024 log files with each *ftdell* command. If you want to delete more files then you must repeat the command.

Store the log records by redirecting the output of *ftshwl* to a file or to the printer.

Deleting log records changes the size of the file since the storage space is freed immediately after deletion.

The time by which the log records are to be deleted can be entered either as a fixed time with date and time or as a relative time; for example: all records before 10 days ago.

> ¡ You can also automate the deletion of log records by using the *ftmodo command* to set the corresponding options (*-ld*, *-lda*, *-ldd*, *-ldt*) in the operating parameters. This is recommended if you only want to retain logging information of up to a given age. However, you should not use this method if you want to maintain a continuous longterm archive of the log records

Format

```
ftdell   -h |
         [ -rg=[[[yyyy]mm]dd]hhmm | -rg=#1..999999999999 | -rg=0..999 ] |
         [ -tlf=yyyymmdd[hh[mm[ss]]] ]
```

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-rg=[[[yyyy]mm]dd]hhmm

You use *-rg* to specify the end of a logging interval.

When selecting the time, this is interpreted as follows:

- a 4-digit specification is interpreted as the time expressed in hours and minutes,
- a 6-digit specification as the day (date) and time in hours and minutes,
- an 8-digit specification as the month, day, and time in hours and minutes,
- a 12-digit specification as the year, month, day, and time in hours and minutes.

The largest possible value that can be specified as the date is 20380119 (January 19, 2038). openFT then deletes all log records which are older that the specified time. The optional data ([...]) is automatically replaced by current values.

-rg=#1..999999999

Here you use *-rg* to specify the end log ID. It is identified by a leading # character, followed by the 1-12-digit ID. openFT then deletes all log records which belong to this log ID or which belong to a smaller log ID.

-rg=0..999

Here you use *-rg* to specify a time interval (relative to the current date and time) as a multiple of 24 hours, i.e. number of days. openFT then deletes all log records which are older than the specified time. This means you are looking back in time. If you specify *-rg=2*, for example, all log records which are older than two days (48 hours) are deleted.

-rg not specified

The range is not a selection criterion, i.e. all log records are to be deleted by 00:00 hours of the current date.

-tlf=yyyymmdd[hh[mm[ss]]]

-tlf deletes all the offline log files that were switched offline on or before the specified time (local time!). This ensures that only log files that are at least as old as the specified time are deleted.

You must always specify the date as an 8-digit value indicating the year month and day. The year must be greater than or equal to 2000. You can specify the time (hhmmss) partially or not at all if you wish. "00" is added to replace any missing specifications.

If you enter the current date or a date in the future then all the existing offline logging records are deleted.

The options *-rg* and *-tlf* must not be specified simultaneously!

Examples

1. As the FT or FTAC administrator, you wish to delete all FT log records written up to 00:00 hours of the current date.

   ```
   ftdell
   ```

2. As the FT or FTAC administrator, you wish to delete all FT log records written up to the current time:

   ```
   ftdell -rg=0
   ```

3. As the FT or FTAC administrator, you wish to delete all log records written before the last 7-day period (7 times 24 hours before the current time):

   ```
   ftdell -rg=7
   ```

4. As the FT or FTAC administrator, you wish to delete all log records from the beginning to the record with the log ID 1450:

   ```
   ftdell -rg=#1450
   ```

5. As the FT or FTAC administrator, you want to delete all the offline log files that were set offline before 01.04.2016:

   ```
   ftdell -tlf=20160331235959
   ```

3.18 ftdelp

Note on usage

Function: Delete FT profiles

User group: FTAC user and FTAC administrator

Functional description

ftdelp stands for "delete profile". You should occasionally thin out the set of profiles (with *ftshwp*) to ensure that no out-of-date admission profiles are retained that could potentially threaten the security of your system.

ftdelp allows the FTAC administrator to delete FT profiles belonging to other login names as well.

ftdelp allows the ADM administrator to delete ADM profiles (i.e. FT profiles with the property "access to remote administration server").

Format

ftdelp -h |

<profile name 1..8> | @s | @a
[-s=[<transfer admission> | @a | @n]

[,<user ID> | @a | @adm]]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

profile name | @**s** | @**a**

is the name of the FT profile you wish to delete.

@**s** for *profile name*

Deletes the standard admission profile for the user ID.

@**a** for *profile name*

profile name is not used as a criterion for selecting the FT profile to be deleted. If you do not identify the profile more closely with *-s* (see below) you will delete all of your FT profiles.

-s=[transfer admission | @**a** | @**n**][**,**user ID | @**a**]

-s is used to specify criteria for selecting the FT profiles to be deleted.

transfer admission

is the transfer admission of the FT profile to be deleted. A binary transfer admission must be specified in the hexadecimal format, see section “Entering commands”.

@**a** for *transfer admission*

deletes either the FT profile specified by *profile name* (see above) or all of your FT profiles.

As the FTAC administrator, you must specify *@a* if you want to delete FT profiles belonging to other login names, since you actually should not know the transfer admission.

@**n** for *transfer admission*

deletes FT profiles with no transfer admissions.

As the FTAC administrator, you can specify *@n* if you only want to delete FT profiles of other login names, which do not have any defined transfer admissions.

transfer admission not specified

causes to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the program prompts you to enter the transfer admission a second time. If you just press <ENTER>, this has the same effect as specifying *@a*.

,user ID

As user, you can enter only your own login name here.

As the FTAC administrator, you can specify any login name.

@**a** for *user ID*

allows you to delete FT profiles belonging your own login name.

If you specify *@a* as the FTAC administrator, FT profiles belonging to all login names are deleted.

@**adm** for *user ID*

For the FTAC and ADM administrator only.

If you specify *@adm* as the FTAC or ADM administrator, ADM profiles are deleted.

user ID not specified

deletes only profiles belonging to the user's own login name, regardless of who issues the command.

-s not specified

if *@a* is specified for *profile name*, all the FT profiles belonging to the login name under which the *ftdelp* command is issued are deleted. Otherwise, the FT profile with the specified name is deleted.

Example

The FT profile *goldmrep* is to be deleted.

```
ftdelp goldmrep
```

3.19 ftdels

Note on usage

Function: Deleting an openFT-Script request

User group: FT user and FT administrator

Functional description

The specified, completed openFT-Script request is deleted from the user's directory or all completed openFT-Script requests are deleted from the user's directory.

No more information is subsequently available for deleted requests. A *ftshws* or *ftshwact* command with the *ftscriptid* of a deleted request is rejected since it no longer exists.

Before an openFT-Script request can be deleted, it must have been completed, i.e. ftshws must indicate the status T, F or C.

> ℹ Since *ftcans* is not a synchronous command, it may be necessary to wait for the status C (cancelled) to arise before a subsequent *ftdels*.

If no *ftdels* is issued for an openFT-Script request then this is automatically deleted when the retention period expires.

Format

```
ftdels   -h |
         [ -u=<user ID> ]
         <ftscriptid> | @a
```

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-u=user ID

User ID under which the search for the openFT-Script request that is to be deleted is performed.

Only FT administrators may input a user ID.

The default value is the calling party's user ID.

ftscriptid

Identification of the openFT-Script request. This is output when the openFT-Script request is started via an *ftscript* command.

You can use the wildcard symbols *?* and * in der *ftscriptid*. This deletes all openFT-Script requests that match the wildcard pattern.

?

is interpreted as any single character.

*

is interpreted as any number of characters.

If you use wildcards, enclose the *ftscriptid* specification in single quotes so that the wildcard symbols are not interpreted by the shell.

@a means that all the user's openFT-Script completed requests are to be deleted.

3.20 ftedit

Note on usage

Function: Load local or remote files in the openFT editor

User group: FT user

Functional description

The command *ftedit* allows you to load local or remote files in the openFT editor.

Note for Unix systems

Please note that you require a graphics-capable terminal in order to use the *ftedit* command.

Note for Windows systems

The *ftedit* command is "Send To"-capable, i.e. you can open a text file directly in *ftedit* by sending it using the context menu command *Send To - ftedit.exe* . To be able to use the "Send To" mechanism, you must first create a shortcut to *ftedit.exe* in the corresponding Windows folder. It is also possible to pass shortcuts to *ftedit* . The Editor then loads the file referred to by the shortcut.

Format

```
ftedit   -h |
         [ -ro ]
         [ -n=<line>]
         [ -t |- b |-u ]
         [ -ccs=<ccs> ]
         [ -tad=<tad> <partner>!]<file>
```

Description

-h

Displays the syntax in a separate window.

-ro

Loads the file in write-protected mode. You can only read the file. This corresponds to the "View" function in the Explorer interface.

-n=line

The editor window is positioned on the specified line after the file is loaded.

-t | **-b** | **-u**

In the case of remote files, the file type to be used when the file is transferred to openFT.

-t (default value for openFT partners)

The file contains text with variable record lengths.

Records end with the linefeed character \n on Unix systems.

On Windows systems, records end with the following characters:

- CRLF (X'0D0A') when sending and/or fetching a file
- LF (X'0A'), only possible when sending a file

-u

The file contains variable record length binary data structured by the user. Every record starts with 2 bytes that specify the length of the record.

-b

The file contains an unstructured sequence of binary data.

-ccs=ccs

Name of the character set that is to be used on opening the file. For more information, see the manual "openFT (Unix and Windows systems) - Installation and Operation".

Default: the character set defined as the default in the local openFT system.

-tad=tad

Transfer admission in the partner system in the case of remote files.

You can specify the transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization using the syntax of the remote system (user ID, where necessary with account and/or password).

You will find further details in the section “Entering the authorization data for the partner system” .

partner

For remote files it is necessary to specify an openFT partner name.

Partner is the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

file

Name of the file to be loaded in the openFT editor.

You can specify an absolute path or a relative path for the file name with a maximum length of 512 characters. Please note that the maximum lengths of file names are system-dependent; for example, in Unix systems it is 512 and in Windows systems a maximum of 256 characters. If the file name contains blanks, you must enclose it in double quotes (e.g. "file name").

Note for Unix systems

If the remote partner requires single quotes around the file name, unlike at the shell level you do not have to invalidate these (e.g. 'file name').

3.21 ftexec

Note on usage

Function: Execute operating system commands in remote system

User group: FT user

Functional description

The *ftexec* command is used to execute operating system commands in the remote system. The resulting output for *stdout* and *stderr* is output in the local system on standard output (*stdout*) or standard error (*stderr*).

ftexec is only available for openFT partners, FTP partners and FTAM partners from Fujitsu Technology Solutions.

The end status, i.e. the result of the command, is also output in the local system as the end status of the *ftexec* command. If the end status received exceeds the value range of the local end status (Unix systems have only a 1-byte end status while Windows systems have a 4-byte end status), only the contents of the least significant byte are output. The significance of the end status is system-specific.

If the command is not executed in the remote system, an end message from the *ftexec* command is output to *stderr*, and *ftexec* terminates with the end status 255.

Notes on character sets

- For commands to be executed in remote Unix or Windows systems, it is possible to set the so called "character mode", i.e. the commands are seen in their character presentation. The same applies to the local system if the commands are entered separately via *stdin*.
- For output operations to *stdout*, it is possible to define character sets (*-lc*, *-rc*).

  In addition, you can define the "system character set" *SYS with *-lc* or *-rc*. *SYS then also applies to *stderr*.
- For output operations to *stderr*, the following character sets are used depending on the system:
  - BS2000 and z/OS systems: character set defined in the system
  - Unix systems: ISO8859-1 or - in character mode - the character set defined via the operating parameter option *-fnccs*.
  - Windows systems: CP850 or UTF-8 in character mode

You will find further information on creating admission profiles for the *ftexec* function in the description of the *ftcrep* command, in particular the *-fnp* option.

If the partner is a Windows system, you can switch to another directory before calling the actual command as follows:

`ftexec WinPart "cd` *path-name*;*command* `"...`

path-name is then used as the current directory. There must not be a blank between the semicolon and the command. *path-name* must not be a directory which is addressed using a UNC name. Exception: The UNC checking is deactivated on the system on which the command is to be executed. To do this, the registry value described under *https://support.microsoft.com/de-de/kb/156276* has to be generated.

Format

ftexec -h |

[-t | -b | -l]
[-c]
[-fnc=t | -fnc=c]
[-lc=<CCS name 1..8>] [-rc=<CCS name 1..8>]
<partner 1..200>
<command> | -
[<transfer admission 8..67> | @n | @d |
<user ID 1..67>[,[<account 1..64>][,[<password 1..64>]]]]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-t

This option indicates the transfer format for *stdout* is text. Tabulator expansion is deactivated.
Default value if a CCS name is specified (*-lc* and/or *-rc*).

-b

This option indicates that the transfer format for *stdout* is binary without conversion.
Default value if no CCS name is specified (neither *-lc* nor *-rc*).

-l

This option indicates that the transfer format for *stdout* is binary with <CRLF> converted to <LF> (transfer of text in binary format). This mode is only of use if both partners use ISO 646 or ISO8859-1 as the text format.

-c

Specifies that the data is also to be encrypted at transfer. The encryption of the request description data is not affected by this option. If the partner system cannot work with encryption, the request is rejected.

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for the commands to be executed in the remote system.

t (transparent, default value)

Specification of the commands to be executed in the remote system in transparent mode (compatible to the previous versions).

c (character)

Specification of the commands to be executed in the remote system in character mode. The commands are interpreted according to the character code of the remote system, i.e. for Unix partners according to the openFT operating parameter option (*ftmodo -fnccs)* that has been set there.

-fnc=c is only permitted for partners as of openFT V12.1.

-lc=CCS name

(local coding) specifies the type of coding (character set) to be used to read the local file. *CCS name* must be known in the local system (exception: *SYS, see below).

The default value is the character set defined by the FT administrator.

***SYS** for *CCS-name*

*-lc=*SYS* causes that the following character set is used for *stdout* and *stderr*:

Unix systems: character set that results from the LOCALE and LANG-setting. Windows systems: UTF-8

-lc may not be combined with *-b* or *-l*.

Details about the CCS name and the associated code tables can be found in the manual "openFT (Unix and Windows systems) - Installation and Operation".

-rc=CCS name

(remote coding) specifies the type of coding to be used to read the data at the standard output from the remote command. *CCS name* must be known in the remote system (exception: *SYS, see below).

The default value is the character set defined in the remote system.

***SYS** for *CCS-name*

*-rc=*SYS* causes that the following character set is used for *stdout* and *stderr*:

Unix systems: Character code set via openFT operating parameter option (*ftmodo -fnccs*).

Windows systems: UTF-8
BS2000 and z/OS systems: same as default value, i.e. the character set defined in the remote system.

The specification *-rc=*SYS* is intended on Unix systems for commands or programs, in which the output depends on local language settings. On Windows systems *-rc=*SYS* is suitable for programs that provide output in UTF-8.

*-rc=*SYS* is permitted only for partners with openFT as of V12.1.

-rc may not be combined with *-b* or *-l*.

The option *-rc* is supported only by the openFT protocol and partners with openFT V10.0 or higher. Please note that not all partner systems support all the character sets that are possible in the local system.

partner

partner is the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

command | -

command is the command to be executed in the remote system. The syntax and the processing of the statements and commands depend on the conventions of the system on which the command is to be executed. A command sequence can only be processed in the remote system if an FT product that supports this function is being used there.

The maximum length of the command is 8191. Special characters count as being two characters (for the representation in UTF-8, see section “Entering commands”).

- (dash) for *command*

You must enter the command after sending the *ftexec* command via *stdin*. You terminate entry by pressing the following keys:

<END> or CTRL+D (Unix systems)

CTRL+Z at the start of a line, followed by Return (Windows systems).

The entry is interpreted with the following character code depending on the encoding mode (*-fnc)*:

- Unix system in transparent mode: ISO8859-1
- Unix system in character mode: the code that corresponds to the set LOCALE and LANG variables.
- Windows system in transparent mode: ISO8859-1
- Windows system in character mode: UTF-8

transfer admission | @**n** | @**d** |
user ID[**,**[account][**,**[password]]s]

If you want to execute a command on a remote system, you must furnish the remote system with proof of identity. For this purpose, you will need login authorization in the syntax valid for the remote system. You can specify this transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization in the syntax used by the remote system (user ID, possibly together with account or password).

For details, see section “Entering the authorization data for the partner system” .

@**n** for *transfer admission*

By entering @*n* you specify that the remote system requires no login authorization.

@**d** for *transfer admission*

Specifying @*d* (blanked transfer admission) causes openFT to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission.

A binary password or binary transfer admission must be specified in hexadecimal format, see section “Entering commands”.

password not specified

Omitting the password necessary for admission causes openFT to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

Nevertheless, you have to specify the commas, e.g.:

```
ftexec system command user-id,,
```

or

```
ftexec system command user-id,account,
```

neither *transfer admission* nor *user ID* specified

causes the same as @*d*, i.e. openFT queries the transfer admission on the screen after the command is entered. Your (blanked) entry is always interpreted as transfer admission and not as user ID.

Examples

1. You want to look at the last 12 log records in the remote Unix system *ux1* using the transfer admission *Transunix1*:

   ```
   ftexec ux1 "ftshwl -nb=12" Transunix1
   ```

2. You want to look at the last 12 log records in the remote BS2000 system *bs2* using the transfer admission *Transbs2*:

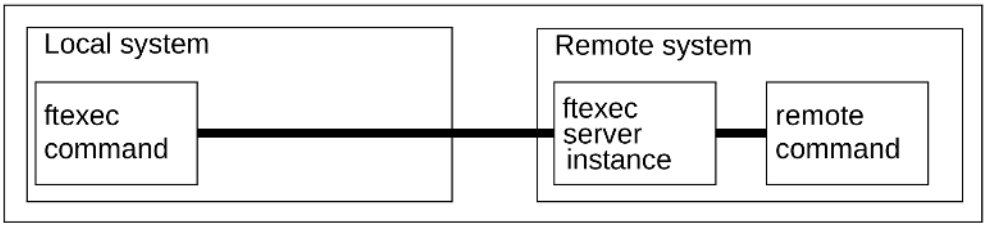
   ```
   ftexec -t bs2 "/SH-FT-LOG ,12" Transbs2
   ```

3. You want to look at the last 12 log records in the remote z/OS system *zos1* using the transfer admission *TranszOS*:

   ```
   ftexec -t zos1 "ftshwlog ,12" TranszOS
   ```

3.21.1 Messages from the ftexec command

Several openFT components in the local and remote systems participate in the execution of an *ftexec* command. Any of these instances can be responsible for the messages issued during execution:



In the local system, these are messages issued locally by the specified *ftexec* command whose execution is very similar to that of the *ncopy* command. Consequently, all the *ncopy* command messages may occur, the only difference being that they start with *ftexec*.

In the remote system, both the remote command itself and the *ftexec* server which monitors the execution of the remote command may handle requests. However, messages from the *ftexec* server are mapped to *ncopy* command messages wherever possible, i.e.:

- If the end status for *ftexec* is not 255, then all *stderr* output originates from the command executed in the remote system (depending on the remote command involved). An end status other than 255 is also the return code of the remote command (at least its last byte).

  > Tip: Avoid return code 255 in the remote command since it is possible that remote command execution may supply an error code 255 which is also passed on. To find out whether a local or remote error has occurred, consult your log files.

- Messages from the other components involved can only have an end status of 255.

- Messages from the *ftexec* command responsible for the transfer of data can have another additional meaning:
  - Request *request ID*: Remote system: Error in pre-/postprocessing
  - Request *request ID*: Remote system: Exitcode *code* from pre-/postprocessing

    Meaning:

    The local preprocessing command could not be executed successfully. The exit code here is the exit code of the *ftexec* server, i.e. 255.
  - Request *request ID*: Remote system: Transfer admission invalid

    Other possible meaning:

    The transfer admission does not permit any command execution.
  - Request

    request ID: Remote system: Syntax error in resulting file name.

    Other possible meaning:

    The command string is too long for the remote partner.
  - Request *request ID*: Remote system: File/directory '*file*' not found

    Other possible meaning:

    The file name prefix in the remote admission profile does not start with "|ftexecsv ".
  - ftexec: Invalid parameter 'c'

    Meaning:

    Encryption of user data is not enabled.

- Messages deriving from *ftexec* server instance messages (these start with "ftexecsv:"):
  - Request *request ID*: Remote system: File/directory does not exist

    Meaning:

    The command specified in *ftexec* does not exist in the remote system - at least not under the explicitly specified or implicitly assumed path. If the partner is a Unix system, this message can also mean that the file exists but cannot be executed as a command or that a resource bottleneck occurred when an attempt was made to start the command.
  - Request *request ID*: Remote system: Access to ... denied

    Meaning:

    The command specified in *ftexec* is not an executable command or includes invalid characters (see *ftcrep* command, *-fnp* option).
  - Request *request ID*: Remote system: Resource bottleneck

    Meaning:

    A resource bottleneck occurred when an attempt was made to start the command specified in *ftexec*.
  - Request *request ID*: Remote system: File structure error

    Meaning:
  - An error occurred while reading the *stdout* or *stderr* data generated when the remote command was executed.
  - A record created by the command specified in *ftexec* cannot be entered in the *ftexec* server buffer. An attempt was probably made to read pure binary output as text.
  - The *ftexec* server received an error flag while forwarding the data from the remote command to the openFT server.
  - Request *request ID*: Internal error. Error code *err_code*

    Meaning:

    An internal error occurred in the remote *ftexec* server.
- Messages from the *ftexec* command itself (these start with "ftexec:"):
  - Request *request ID*: File structure error

    Meaning:

    The data received does not correspond to the *ftexec* format. It may originate from a remote file or from normal preprocessing. Check whether the appropriate transfer admission has been selected.
  - Internal error. Error code *err_code*

    Meaning:

    An internal error *err_code* occurred during the processing of the *ftexec* command.

3.22 ftexpc

Note on usage

Function: Export the XML configuration of the remote administration server

User group: ADM administrator

Functional description

ftexpc stands for "export configuration". If you are the administrator of the remote administration server (= ADM administrator), *ftexpc* allows you to export the configuration data of the remote administration server into an XML file. The content of the XML file with the exported configuration is encoded using UTF-8.

You can use *ftexpc* if you wish to change an existing configuration. To do this, export the existing configuration into an XML file with *ftexpc*, adapt the file (see the manual "openFT (Unix and Windows systems) - Installation and Operation") and then import the changed file again with *ftimpc*.

Format

ftexpc -h |

<file name>

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name

specifies the name of the XML file in which the exported configuration data is to be saved.

The file is created by the *ftexpc* command and must not exist beforehand.

Messages of the ftexpc command

If ftexpc could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case

3.23 ftexpe

Note on usage

Function: Export FT profiles and admission sets

User group: FTAC administrator

Functional description

ftexpe stands for "export environment", i.e. exporting the FTAC environment, or exporting FT profiles and admission sets.

Using *ftexpe* the FTAC administrator can write FT profiles and admission sets of any login names to files, thereby saving them.

However, the standard admission set is not saved and the variable values in an admission set (values marked with an asterisk (*)) that refer to the standard admission set, are saved as variables. This means that there is no fixed value for the relevant basic function in the backup. If an admission set is imported, the relevant basic function receives the value of the standard admission set that is currently valid.

FT profiles and admission sets saved in this way can be re-imported using the *ftimpe* command.

The timestamp of an admission profile is not changed on an export or import operation.

Format

ftexpe -h |

<file name>
[-u=<user ID>[,...,<user ID(100)>]]
[-pr=<profile name 1..8>[,...,<profile name(100) 1..8>] | -pr=@n]
[-as=y | -as=n]
[-adm=y | -adm=n]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name

With *file name*, you specify the name of the file in which the FT profiles and records are to be written. You may access this file only using the *ftimpe* and *ftshwe* commands. No file with the same name must exist in the directory.

-u=user ID1[**,**user ID2][**,**user ID3]...

-u specifies the login names whose FT profiles and admission sets are to be saved to a file. Up to 100 login names can be specified simultaneously.

-u not specified

all FT profiles and admission sets on the system are saved to the specified file.

-pr=profile name1[**,**profile name2][**,**profile name3]... | **@n**

specifies the FT profiles to be saved to the specified file (up to 100).

@n for *profile name*

no FT profiles are saved.

-pr not specified

all FT profiles belonging to the login names specified in the -u parameter, are saved.

-as=y | **-as=n**

specifies whether or not the admission sets should be saved to the specified file. Possible values are:

y (default value)

all admission sets belonging to the login names specified in the *-u* parameter, are saved.

n

no admission sets are saved.

-adm=y | **-adm=n**

specifies whether or not the ADM profiles (i.e. FT profiles with the property "access to remote administration server", corresponding to *ftcrep -ff=c)* should be saved to the specified file. Possible values are:

y (default value)

all ADM profiles are saved.

n

no ADM profiles are saved.

Example

The admission set and the FT profiles belonging to the login name *donald* are to be saved. *ftacsave* is specified for the backup file.

```
ftexpe ftacsave -u=donald
```

3.24 fthelp

Note on usage

Function: Display information on the log record reason codes

User group: FT user and FT administrator

Functional description

With *fthelp*, you can have the meanings of the reason codes for the log function displayed on the screen (RC column in *ftshwl* output).

You can also request the output of the message texts associated with the exit codes of certain FT commands.

Format

fthelp -h | <number 1..ffff>

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

number

This is a four-digit reason code from the log function or the exit code of an FT command belonging to a synchronous FT request. The reason code contains encoded information on an FT request accepted by openFT.

The reason codes and their meanings are listed in the *ftshwl* command in the section “Reason codes of the logging function”.

The exit codes (= message numbers) are listed in section “openFT messages” .

Example

You wish to find out the meaning of reason code 3001.

```
fthelp 3001
```

```
3001 Request rejected. Invalid user identification.
```

Thus, reason code 3001 means that the specified login name or transfer admission is invalid.

3.25 ftimpc

Note on usage

Function: Import the configuration of the remote administration server

User group: ADM administrator

Functional description

ftimpc stands for "import configuration". If you are an ADM administrator, *ftimpc* allows you to import an XML file containing configuration data on the remote administration server. The existing configuration is overwritten on import.

The format of the XML file must match the format in the schema defined in *config.xsd*. *config.xsd* is located in the openFT installation directory under the directory *include*. You will find further details on creating a configuration file in the manual "openFT (Unix and Windows systems) - Installation and Operation".

The XML file is checked for correct syntax and semantics by the XML parser and XML schema validator during import. If errors occur, a message is output to *stderr* indicating the element or the row/column in which the error occurred. The messages generated always appear in English.

In some cases, it is possible that you will receive a message during import indicating that the configuration data cannot be imported and that the asynchronous openFT server must be terminated. In this case, stop the asynchronous openFT server (e.g. using the *ftstop* command), call the *ftimpc* command again and then restart the asynchronous openFT server (e.g. using the *ftstart* command).

You can use *ftimpc* if you wish to change an existing configuration. To do this, export the existing configuration into an XML file with *ftexpc*, adapt the file and then import the changed file again with *ftimpc*.

The content of the XML file exported with *ftexpc* is encoded using UTF-8 (see the *ftexpc* command). You should therefore also encode an import file in UTF-8.

Format

ftimpc -h |

<file name>

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name

specifies the name of the XML file to be imported.

Messages of the ftimpc command

If ftimpc could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case.

3.26 ftimpe

Note on usage

Function: Import profiles and admission sets

User group: FTAC administrator

Functional description

ftimpe stands for "import environment", i.e. importing the FTAC environment or importing FT profiles and admission sets. Using *ftimpe*, the FTAC administrator can import the FT profiles and admission sets of any login names from a file that was created using the *ftexpe* command.

Only those FT profiles whose profile names have not been specified for other FT profiles under the specified login name are imported.

If a profile with the same name is already present, the timestamp (LAST-MODIF with *ftshwp -l*) indicates which has the most recent status.

An FT profile whose transfer admission has already been defined for another FT profile in the system will be imported, but has an undefined transfer admission. It must therefore be assigned a new transfer admission using the *ftmodp* command before it is used. If the existing FT profile in the system is designated as private, it is immediately disabled. It must be assigned a new transfer admission using the *ftmodp* command, before it is used.

The imported FT profiles are automatically locked and must be unlocked before use with the command *ftmodp* and the parameter *-v=y* if the FTAC administrator does not have FT administrator privileges. Privileged FT profiles lose their privileged status when imported. The FTAC administrator can control this behavior with the *-sec* option provided that he has FT administrator privileges.

The standard admission set is not saved when it is exported. Therefore, the standard admission set on the computer at the time of importing remains valid. Variable values in the imported admission sets, that refer to the standard admission set and are therefore marked with an asterisk (*), are assigned the value of the standard admission set that is currently valid.

Format

ftimpe -h |

<file name>
[-u=<user ID>[,...,<user ID(100)>]]
[-pr=<profile name 1..8>[,...,<profile name(100) 1..8>] | -pr=@n]
[-as=y | -as=n]
[-sec=s | -sec=h]
[-adm=y | -adm=n]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name

file name specifies the file from which the FT profiles and admission sets are to be imported.

-u=user ID1[**,**user ID2][**,**user ID3]...

specifies the login names whose FT profiles and admission sets are to be imported. You can specify up to 100 login names simultaneously.

-u not specified

all FT profiles and admission sets are imported.

-pr=profile name1[**,**profile name2][**,**profile name3]...| **-pr=@n**

specifies the FT profiles to be imported (up to 100).

@**n** for *profile name*

no FT profiles are imported.

-pr not specified

all FT profiles belonging to the login names specified in the *-u* parameter are imported. However, the profile is not imported if another FT profile of the same name already exists under this login name.

as=y | **-as=n**

specifies whether or not admission sets are to be imported. Possible values are:

y (default value)

all admission sets belonging to the login names specified in the *-u* parameter are imported.

n

no admission sets are imported.

-sec=s | **-sec=h**

-sec specifies the security level when importing FT profiles. It only makes sense to use the *-sec* option if you, the FTAC administrator, have FT administrator privileges.

s (standard)

If you have FT administrator privileges, the attributes of the FT profile are not changed when it is imported.

If you do not have FT administrator privileges, the effect is the same as *-sec=h*, i.e. the profiles are locked.

-sec=s is the default value.

h (high)

The FT profiles are locked (LOCKED (by import)) and are assigned the attributes *private* and *not privileged*.

-adm=y | **-adm=n**

specifies whether or not the ADM profiles (i.e. FT profiles with the property "access to remote administration server", corresponding to *ftcrep -ff=c)* are to be imported. Possible values are:

y (default value)

all ADM profiles are imported. This option is permissible only if an ADM administrator is configured on the target computer.

n

no ADM profiles are imported.

Example

The admission set and FT profiles of the login name *donald* were saved to the file *ftacsave* with *ftexpe*. They are to be imported to another system under the same login name.

```
ftimpe ftacsave -u=donald
```

As the FTAC administrator you may receive the following messages, for example:

```
OWNER    NAME
donald   secret1     FT profile already exists.
         secret2
```

These messages indicate that *donald* has already created the FT profiles *secret1* and *secret2* on the new system, and these profiles were therefore not imported.

Note

If, after import, you wish to delete an admission set for a login name that does not exist on your computer, enter the command *ftmoda login-name -ml=s*. This situation can occur when you use *ftexpe* to incorporate into your system a file that has been created on a different host.

3.27 ftimpk

Note on usage

Function: Import RSA key

User group: FT administrator

Functional description

You can use the command *ftimpk* (import key) as FT administrator to import a partner's public key or an RSA key pair from a file. The file is made available by the party that generated the key/RSA key pair. On import, the partner key or RSA key pair is saved at the "correct" location in the openFT instance directory and can then be used for authentication.

Importing public keys of a partner

If you want to import the public key of a partner then this partner must be entered in the partner list. The key is stored in the *syskey* subdirectory with the partner ID as file name.

Importing RSA key pairs

You can import an RSA key pair consisting of a public and a private key. The key pair can be used like a key generated by openFT for data encryption and authentication.

The key pair can have been generated using an external tool. Keys must have the length 768, 1024, 2048, 3072 or 4096 bit. The keys may be present in PEM format (native PEM or PKCS#8 format without password phrase or, after v1 / v2, with password phrase) or in PKCS#12 V1.0 format.

If the key pair demands a password phrase (password) then this must be specified during the import.

During import, the same applies as for key pairs generated with *ftcrek*:

- The key pair contains a unique reference number.
- The public key is stored under the name **syspkf.r<key-reference>.l<key-length>** in the *config* directory of the openFT instance's instance file tree.

See also the manual "openFT (Unix and Windows systems) - Installation and Operation".

Format

ftimpk -h |

[-pr=<file name 1..512>]
[-pu=<file name 1..512>]
[-p=<password 1..64> | -p=]
[-p12]

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-pr=file name (private)

indicates that a private and public key are to be imported. *file name* is the absolute or relative path name of the file containing the two keys.

-pu=file-name (public)

indicates that only a public key is to be imported. *file name* is the absolute or relative path name of the file containing the key.

You must always specify either *-pr* or *-pu*!

-p=password | **-p=**

Specifies the password if the key or keys is (are) password-protected.

No password specified

If you specify *-p=* without a password, the password is queried on screen after the command has been sent. The entry you make is not displayed, in order to prevent unauthorized persons from seeing the password.

-p not specified

The key(s) is/are not password-protected, default value.

-p12

The key file contains a certificate and a private key in accordance with the standard PKCS#12 V1.0. The file is searched for a private key and any non-supported elements (e.g. certificates, CRLs) are ignored during the import. The first private key that is found in the file is imported. Any others are ignored.

If the certificate is protected by a signature or hash then openFT does not perform a validity check. The validity of the file must be verified using other means.

-p12 not specified

The private key is not present in PEM format, default value.

Examples

1. You want to import the public key from the file clientkey1 (without a password).

   ```
   ftimpk -pu=clientkey1
   ```

2. You want to import an RSA key in PEM format that was generated using a tool from the file rsakeys20170303. The keys are protected by a password which you must enter invisibly (hidden) at the screen.

   ```
   ftimpk -pr=rsakeys20170303 -p=
   ```

3.28 ftinfo

Note on usage

Function: Output information on the openFT system

User group: FT user

Functional description

ftinfo outputs information about the installed openFT system.

Format

ftinfo -h |

[-csv]

Output

ftinfo always outputs the values in CSV format even if the *-csv* option is not specified:

Name	Type	Values
CmdUiVer	Number	Version of the User Command Interface, e.g. 1210 for V12.1. The User Command Interface provides the user and administrator commands.
CmdTiVer	Number	Version of the Tool Command Interface, e.g. 100 for V1.00.
OsType	String	Name of the operating system: Windows, Unix, BS2000/OSD, z/OS.
UserId	String	Current (calling) user ID.
IsFtAdm	Number	1 for UserId=FT administrator, 0 otherwise
IsFtacAdm	Number	1 for UserId=FTAC administrator, 0 otherwise.
FtLang	String	Set language: de (German), en (English).
CcsName	String	CCS name of the character set currently defined in openFT.
Home	String	Home directory of the calling user ID.
Limited	String	*NO or `yyyy-mm-dd` *NO: The installed openFT product is NOT a limited period evaluation version. `yyyy-mm-dd`: The installed openFT product is a limited period evaluation version that can be used until the specified date. openFT can no longer be used after the date displayed.
IsAdmAdm	Number	1 for UserId=ADM administrator, 0 otherwise

ProdVer	String	openFT product version, e.g. 12.1A00
SrcVer	String	Source version, e.g. 356
Inst	String	Name of the instance, e.g. std
TimeOffset	Number	Time difference between local time and UTC time in seconds, e.g. 3600 corresponds to an hour
FtScriptDir	String	The directory in which openFT-Script applications are stored. It can be specified using the *ftmodsuo* command.
NativeX25	String	*YES / *NO *YES: FarSync X.25 is supported on condition that it is installed on the system.
SingleUser	String	*YES / *NO *YES means Single User Operation (only on Unix systems)
Crypt	String	*YES / *NO *YES: Encryption is supported on condition that openFT-CR is installed on the system.

Example

Unix systems

```
ftinfo
CmdUiVer;CmdTiVer;OsType;UserId;IsFtAdm;IsFtacAdm;FtLang;CcsName;Home;Limited
;IsAdmAdm;ProdVer;SrcVer;Inst;TimeOffset;FtScriptDir;NativeX25;SingleUser;Cry
pt
1212;100;"Unix";"admin";1;1;"de";"ISO88591";"/home/usr/admin";*NO;1;"12.1C00"
;"407";"std";3600;"/home/usr/user1";*NO;*YES;*YES
```

Windows systems

```
ftinfo
CmdUiVer;CmdTiVer;OsType;UserId;IsFtAdm;IsFtacAdm;FtLang;CcsName;Home;Limited
;IsAdmAdm;ProdVer;SrcVer;Inst;TimeOffset;FtScriptDir;NativeX25;SingleUser;Cry
pt
1212;100;"Windows";"admin1";1;1;"de";"CP1252";"C:\Users\admin1";
*NO;0;"12.1C00";"406";"std";3600;"C:\Users\user1";*NO;*NO;*YES
```

3.29 ftlang

Note on usage

Function: Change default language setting

User group: FT administrator

This command is only available on Unix systems.

Functional description

The default language for openFT is determined by evaluating the LANG environment variable during installation (Linux, Solaris, AIX) or, in HP-UX, is set to English by default.

You can switch languages later on using the shell procedure */opt/openFT/bin/ftbin/ftlang*.For more details see the manual "openFT (Unix and Windows systems) - Installation and Operation".

Format

ftlang [-h |

-i |
de |
en]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-i

you can use this switch to query the currently set language variant.

de

openFT is switched to German as the default.

en

openFT is switched to English as the default.

In both cases, the necessary message files, the *fthelp* procedure, the man pages (Solaris, AIX and HP-UX) and the openFT Explorer including the help texts for the selected language are activated.

Example

1. Check which language is selected:

   ```
   /opt/openFT/bin/ftbin/ftlang -i
   ```

   ```
   en
   ```

2. The default language setting is switched from German to English:

   ```
   /opt/openFT/bin/ftbin/ftlang en
   ```

3.30 ftmod

Note on usage

Function: Modify file attributes in a remote system

User group: FT user

Functional description

With *ftmod* you can modify the attributes of a file in a remote system. Depending on the partner (openFT, FTAM or FTP), the following file attributes can be modified:

With openFT partners:

- File name
- Access rights (not if the partner system is a Windows system)

With FTAM partners:

- File name
- Access rights (not if the partner system is a Windows system)
- Availability of the file
- Account for file storage costs
- Legal qualification on using the file
- Future file size

With FTP partners:

- File name

Format

ftmod -h |

<partner 1..200>![<file name 1..512>]
[<transfer admission 8..67> | @n | @d |
<user ID 1..67>[,[<account 1..64>][,[<password 1..64>]]]]
[-fnc=t | -fnc=c]
[-p=[<management password 1..64>]]
[-nf=<new file name 1..512>]
[-av=i | -av=d]
[-ac=<new account 1..64>]
[-fs=<future filesize 1..2**63-1>]
[-am=[+][r][i][p][x][e][a][c][d] | -am=@rw | -am=@ro]
[-lq=<legal qualification 1..80>]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner**!**[file name]

Specifies for which file and on which system the attributes are to be modified.

partner

partner is the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

file name

file name can be either absolute or relative to the remote login authorization. If the file name in the remote system has been predefined in an FT profile, it must not be specified here.

If the partner system is running openFT (BS2000), elements from PLAM libraries may also be specified here (Syntax: Library name/Element type/Element name).

transfer admission | **@n** | **@d** |
user ID[**,**[account][**,**[password]]]

In order to modify the file attributes in the remote system, you must furnish the remote system with proof of identity. For this purpose, you will need login authorization in the syntax valid for the remote system. You can specify this transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization in the syntax used by the remote system (user ID, possibly together with account or password).

For details, see section “Entering the authorization data for the partner system” .

@n for *transfer admission*

By entering *@n* you specify that the remote system requires no login authorization.

@d for *transfer admission*

Specifying *@d* (blanked transfer admission) causes openFT to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission.

A binary password or binary transfer admission must be specified in hexadecimal format, see section “Entering commands”.

password not specified

Omitting the password necessary for admission causes openFT to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

Nevertheless, you have to specify the commas, e.g.:

```
ftmod partner!file user-id,,
```

or

```
ftmod partner!file user-id,account,
```

neither *transfer admission* nor *user ID* specified

causes the same as *@d*, i.e. openFT queries the transfer admission on the screen after the command is entered. Your (blanked) entry is always interpreted as transfer admission and not as user ID.

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for the remote file names (*file name*, *new file name*).

t (transparent, default value)

Specification of the remote file names in transparent mode (compatible to the previous versions).

c (character)

Specification of the remote file names in character mode. The names are interpreted according to the character code of the remote system, i.e. for Unix partners according to the openFT operating parameter option (*ftmodo -fnccs)* that has been set there.

-fnc=c is only permitted for partners as of openFT V12.1.

-p=[management password]

If the file in the remote system is protected by a password, you must enter this password here.

A binary password must be entered in hexadecimal format, see section “Entering commands”. This is of relevance for links to openFT (BS2000), because BS2000 supports the definition of hexadecimal passwords.

management password not specified

Specifying *-p=* causes openFT to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

-nf=new file name

This indicates the new name for the file *file name* in the partner system. The name *file name* is then no longer valid. *new file name* can be either absolute or relative to the remote login authorization.

-nf not specified

The file name remains unchanged.

-av=i | **-av=d**

Indicates the availability of the file in an FTAM partner system. This parameter can have one of two values: *immediate* or *deferred*. A file may be *deferred* if it has been archived, for example. The partner is responsible for interpreting the term *deferred*. The FTAM partner conventions must therefore be observed here.

The following values are possible:

i

In the remote system, the file attribute is set to *immediate*.

d

In the remote system, the file attribute is set to *deferred*. The file on the partner system can then be placed in an archive, for example.

Requests involving openFT or FTAM partners that do not support the storage group are rejected.

-av not specified

The previous value for availability remains unchanged.

-ac=new account

With FTAM partners, this indicates the number of the account to which file storage fees are to be charged. This parameter must be set in accordance with partner system conventions.

Requests involving openFT or FTAM partners that do not support the storage group are rejected.

-ac not specified

The previous account number remains unchanged.

-fs=future filesize

With FTAM partners, this indicates the expected file size. This is used as a guide for system-specific optimization.

Requests involving openFT or FTAM partners that do not support the storage group are rejected.

-fs not specified

The previous file size remains unchanged.

-am=[**+**][**r**][**i**][**p**][**x**][**e**][**a**][**c**][**d**] | **@rw** | **@ro**

This changes the access rights for a file in the remote system. Old access rights can also be replaced with new ones.

The following values can be specified for the *-am* parameter:

+, *r*, *i*, *p*, *x*, *e*, *a*, *c*, *d* or any combination of these values as well as @*rw*, or @*ro*.

+

with FTAM partners means that the file receives a new set of access rights in addition to the existing rights. This entry is only relevant for FTAM partners that support more than one set of access rights.

+ not specified

the existing access rights of the file in the remote system are replaced by the specified access rights.

r

means that the file can be read.

r not specified

The file cannot be read.

i

with FTAM partners means that data units, such as records, can be inserted in the file.

i not specified

No data units can be inserted in the file.

p

means that the file can be overwritten.

The file cannot be overwritten.

x

means that data can be appended to the file.

x not specified

The file cannot be extended.

e

with FTAM partners means that data units, such as records, can be deleted from the file.

e not specified

No data units can be deleted from the file.

a

means that the file attributes can be read.

a not specified

The file attributes cannot be read.

c

means that the file attributes can be changed.

c not specified

The file attributes cannot be changed.

d

means that the file can be deleted.

d not specified

The file cannot be deleted.

@rw

is the short form of the common access rights *read-write* (*rpxeacd*), and thus simplifies input.

@ro

is the short form of the common access rights *read-only* (*rac*), and thus simplifies input.

If the partner system is a Windows system, you cannot change the access rights of the destination file.

With Unix or BS2000 partner systems, only the following access rights can be set for a file:

Access mode	Short form	Unix system	BS2000	Access rights
rpxeacd	@rw	rw*	ACCESS=WRITE	read-write
rac	@ro	r-*	ACCESS=READ	read-only

pxeacd		-w*	only with BASIC-ACL (Access Control List)	write-only
ac		--*	only with BASIC-ACL (Access Control List)	none

* The x bit is not changed by *ftmod*.

Requests involving FTP partners or involving FTAM partners that do not support the security group are rejected.

-am not specified

The current access rights remain unchanged.

-lq=legal qualification

With FTAM partners, this specifies a legal qualification for the file (similar to a copyright). This may not exceed 80 characters.

Requests involving openFT or FTAM partners that do not support the security group are rejected.

-lq not specified

The current legal qualifications remain unchanged.

Example

You wish to reset the access rights of the remote file *junk* from *read-only* to *read-write*. The file is on the BS2000 computer *bs2r1* under login name *jim* with account number *a1234ft* and password *C'pwd'*. The file is protected by the password *abcd*.

Unix systems:

```
ftmod bs2r1!junk jim,a1234ft,C\'pwd\' -p=C'abcd' -am=@rw
```

Windows systems:

```
ftmod bs2r1! junk jim,a1234ft,C'pwd' -p=C'abcd' -am=@rw
```

3.31 ftmoda

Note on usage

Function: Modify admission sets

User group: FTAC user and FTAC administrator

Functional description

ftmoda stands for "modify admission set".

When *ftmoda* is issued by an FTAC user, it modifies one or more of the settings for basic functions in that user's admission set (MAX. USER LEVELS).

As the FTAC administrator, you can use this command to define settings for the standard admission set and for any admission set of any user in the system. The settings made by the administrator for other users are the MAX. ADM LEVELS.

You can assign a security level of between 0 and 100 for each basic function. These values have the following meanings:

0	The basic function is locked, i.e. it is not released for any partner system.
1 to **99**	The basic function is only released for partner systems with the same or a lower security level. You can use the *ftshwptn* command to display the security level of a partner system.
100	The basic function is available for all partner functions.

For basic functions, consult the table in section "Dependencies concerning inbound file management".

The FTAC or ADM administrator can also use *ftmoda* to transfer the FTAC administrator privileges or the ADM administrator privileges to other user IDs. The ADM administrator also has the option of returning the ADM administration permission.

Format

ftmoda -h |

[<user ID> | @s]
[-priv=y]
[-admpriv=y | -admpriv=n]
[-ml=s | -ml=0..100]
[-os=s | -os=0..100]
[-or=s | -or=0..100]
[-is=s | -is=0..100]
[-ir=s | -ir=0..100]
[-ip=s | -ip=0..100]
[-if=s | -if=0..100]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

user ID | **@s**

Users can enter only their own login names here. *@s* is not permitted.

As the FTAC administrator, you can specify any login name desired.

@s for *user ID*

By entering the value @s, the FTAC administrator can modify the standard admission set.

user ID not specified

modifies the admission set of the login name under which *ftmoda* is entered.

-priv=y

can only be used by the FTAC administrator.

As the FTAC administrator, you can assign administrator privileges to the specified *user ID*.

-priv not specified

does not change the FTAC administrator.

-admpriv=y | **-admpriv=n**

The ADM administrator can pass the administration admission to another user ID or return it.

y

Can only be used by the ADM administrator. If you are an ADM administrator, this specification allows you to pass the administration admission for the remote administration server to the *user ID* specified.

In addition, all profiles defined with *-ff=c* are forwarded to the new user ID. If profiles with the same name already exist under the new user ID, the command is rejected.

n

With this option the ADM administrator returns the ADM administration permission.

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 admission 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.

If there does not yet exist an ADM administrator on the remote administration server, the FTAC administrator has to define the ADM administrator **first** using *-admpriv=*. Otherwise the remote administration server cannot be administrated, i.e. the configuration file cannot be imported by means of *ftimpc,* for example.

-admpriv not specified

does not change the ADM administrator.

-ml=s | **-ml=**0..100

sets the same value for all six basic functions.

Possible values are:

s

sets each of the basic functions to the value defined in the standard admission set.

0

disables all of the basic functions.

1 to **99**

All basic functions are released only for partner systems whose security level is equal to or lower than the specified value.

100

All basic functions are released for all partner systems. For outbound file management functions, no check is made.

-ml not specified

leaves the settings in the admission set unchanged if none of the following entries are made.

-os=s | **-os=**0..100

sets the value for the basic function *outbound send,* see below for possible values. *outbound send* means that requests initiated in your local system send data to a remote system.

-or=s | **-or=**0..100

sets the value for the basic function *outbound receive,* see below for possible values. *outbound receive* means that requests initiated in your local system fetch data from a remote system.

-is=s | **-is=**0..100

sets the value for the basic function *inbound send*, see below for possible values. *inbound send* means that a remote partner system fetches data from your local system.

-ir=s | **-ir=**0..100

sets the value for the basic function *inbound receive*, see below for possible values. *inbound receive* means that a remote partner system sends data to your local system.

-ip=s | **-ip=**0..100

sets the value for the basic function *inbound follow-up processing + preprocessing + postprocessing*, see below for possible values. This determines whether or not a remote system may request follow-up, pre- or postprocessing on your local system.

-if=s | -**if=0**..100

sets the value for the basic function *inbound file management*, see below for possible values.

Please note that subcomponents of *inbound file management* are affected by other settings, see section "Dependencies concerning inbound file management".

-os, *-or, -is, -ir, -ip* or *-if* not specified

leaves the setting for the respective basic function unchanged.

Possible values for the basic functions

The following values are possible for the individual basic functions (*-os*, *-or, -is, -ir*, *-ip* and *-if*):

s

The specifications in the default admission record apply to the basic functions.

0

The basic function is locked.

With some basic functions, this can also affect inbound file management components. For details, refer to the table below.

1 to **99**

The basic function is only released for partner systems on which the security level is less than or equal to the specified value.

100

The basic function is released for all partner systems.

Dependencies concerning inbound file management

The subcomponent *Display file attributes* is controlled by the basic *function inbound send*. In addition, the following dependencies on other settings exist for some components:

Inbound file management function	**Values of the admission set or extension in profile**
Display file attributes	Inbound Send (IBS) enabled
Modify file attributes	Inbound Receive (IBR) **and** Inbound File Management(IBF) enabled
Rename files	Inbound Receive (IBR) **and** Inbound File Management (IBF) enabled
Delete files	Inbound Receive (IBR) enabled **and** Write mode = overwrite in profile
Display directories	Inbound File Management (IBF) enabled
Create, rename and delete directories	Inbound File Management (IBF) enabled **and** direction = from partner in profile

Example

The user Donald wishes to change the admission set for his login name *donald* to prevent remote systems accessing his login name, while still allowing to send files. This requires that the *outbound* basic functions be enabled and the *inbound* basic functions disabled. This can be achieved with the following command:

```
ftmoda -os=100 -or=100 -is=0 -ir=0 -ip=s -if=0
```

Donald specifies the value *s* for the basic function *inbound follow-up* + *preprocessing* + *postprocessing* (*-ip* option), which refer to the standard admission set, where this basic function is also disabled.

3.32 ftmoddir

Note on usage

Function: Modify attributes of remote directories

User group: FT user

Functional description

You can use *ftmoddir* to modify the following attributes of a directory in a remote system:

- Directory name
- Access rights (not if the partner system is a Windows system or the partner is an FTP partner)

Format

ftmoddir -h |

<partner 1..200>![<file name 1..512>]
[<transfer admission 8..67> | @n | @d |
<user ID 1..67>[,[<account 1..64>][,[<password 1..64>]]]]
[-fnc=t | -fnc=c]
[-p=[<management password 1..64>]]
-nf=<new file name 1..512> | -am=@rw | -am=@ro

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner**!**[file name]

Specifies the directory and partner system for the attribute modification operation.

partner

partner is the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

file name

Name of the directory whose attributes are to be modified. The name can be either absolute or relative to the remote login authorization. If the file name in the remote system has been predefined in an admission profile, it must not be specified here.

If the partner system is running openFT (BS2000) then the name of a PLAM library can also be specified here.

transfer admission | **@n** | **@d** |
user ID[**,**[account][**,**[password]]]

Before you can modify the attributes of a file on a remote system, you must first identify yourself at the system. To do this, you need an authorization in the syntax used at the remote system. You can specify this transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization in the syntax used by the remote system (user ID, possibly together with account or password).

For details, see section “Entering the authorization data for the partner system” .

@n for *transfer admission*

By entering *@n* you specify that the remote system requires no login authorization.

@d for *transfer admission*

If you specify *@d* (blanked) then the transfer admission is queried on the screen after the command has been sent. Your input is invisible to prevent unauthorized persons from seeing the transfer admission.

A binary password or binary transfer admission must be specified in hexadecimal format, see section “Entering commands”.

password not specified

If you omit a password which is required for authorization then it is queried on the screen after the command has been sent. Your input is invisible to prevent unauthorized persons from seeing the password.

Please note that you still have to enter the commas, for example:

```
ftmoddir partner!file user-id,,
```

or

```
ftmoddir partner!file user-id,account,
```

neither *transfer admission* nor *user ID* specified

This has the same effect as *@d*, i.e. the transfer admission is queried on the screen after the command has been sent. openFT always interprets your (hidden) input as a transfer admission and not as a user ID.

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for the remote directory names (*file name, new file name*).

t (transparent, default value)

Specification of the remote directory names in transparent mode (compatible to the previous versions).

c (character)

Specification of the directory names in character mode. The names are interpreted according to the character code of the remote system, i.e. for Unix partners according to the openFT operating parameter option (*ftmodo -fnccs)* that has been set there.

-fnc=c is only permitted for partners as of openFT V12.1.

-p=[management password]

If the directory is protected by a password in the remote system then you must specify this here.

The password must be specified in hexadecimal format, see section “Entering commands”. This is of relevance in the case of a connection with openFT (BS2000) since it is possible to define hexadecimal passwords in BS2000.

management password not specified

If you specify *-p=* then the password is queried on screen after the command has been sent. Your input is invisible to prevent unauthorized persons from seeing the password.

-nf=new file name

Specifies the new name for the directory *file name* in the partner system. The name *file name* then loses its validity. *New file name* may be specified either absolutely or relative to the remote login authorization.

-nf not specified

The directory name is unchanged.

-am=@rw | **-am=@ro**

Modifies the access rights to the directory *file name* in the remote system.

If the partner system is a Windows system, you cannot change the access rights. For Unix or BS2000 systems you can specify either *@rw* or *@ro*:

@rw

means that the access right is *read-write*.

@ro

means that the access right is *read-only*.

-am not specified

No change is made to the access right definitions.

Examples

1. The directory *d:\dir* in the remote Windows system *win1* is to be moved to *d:\users\dir*, the transfer admission is *ChangeDirwin*:

   ```
   ftmoddir win1!d:\\dir ChangeDirwin -nf=d:\\users\\dir
   ```

2. The directory */home/user1/current* in the remote Unix system *ux1* is to be renamed to */home/user1/previous*, the transfer admission is *ChangeDirux*:

   ```
   ftmoddir ux1!/home/user1/current ChangeDirux -
              -nf=/home/user1/previous
   ```

3.33 ftmodf

Note on usage

Function: Modify the FTAM attributes of a local file

User group: FT user

Functional description

This command is above all useful in connection with FTAM partners.

For openFT partners, files of type *binary-fixed* can be provided (see also example 2). The attributes *file type*, *record format* and *record length* are also evaluated when sending a file to openFT partners, but are not set when creating the receive file.

With *ftmodf*, you can modify the FTAM attributes of a file in the local system for a file transfer or file management request involving an FTAM partner. You can also delete the information in the FTAM catalog without deleting the file itself.

The following attributes can be defined:

- File type
- Character set
- Record format
- Record length
- FTAM access rights for a file that cannot be changed by the FTAM partner (permitted actions).

File attributes for file type, character set and record format may only be changed if you are aware of the file contents. If this is not the case, file inconsistencies occur, with the result that data transfer requests to the affected files are terminated. Consult the table that describes the operands.

Note that you cannot use *ftmodf* to negate file attributes on the local system. This means that a file can be deleted by means of operating-system resources (e.g. command *rm* on Unix systems *or erase* on Windows systems) even if the *permitted actions* do not permit deletion by an FTAM partner.

Format

ftmodf -h |

<file name 1..512> -np=@d |
<file name 1..512>
[-ft=t | -ft=b]
[-cs=g | -cs=c | -cs=i | -cs=v]
[-rf=v | -rf=f | -rf=u]
[-rl=<1..65535>]
[-pa=[n][r][i][p][x][e][a][c][d]]
[-np=<file access password 1..11> | -np=@n]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name **-np=@d**

Deletes all the information on the specified file in the FTAM catalog without deleting the file itself. *-np=@d* should not be specified together with other parameters, as these then have no effect.

file name

file name without *-np=@d* indicates the file in the local system whose attributes are to be modified. The file name can be either absolute or relative.

-ft=t | **-ft=b**

This identifies the type of file in the local system. You can enter either *t* or *b*.

t

The file contains text data.

b

The file contains binary data.

-ft not specified

The previous file type remains unchanged.

-cs=g | **-cs=c** | **-cs=i** | **-cs=v**

This can only be used in conjunction with the *t* (text) file type, and describes the character set for the text file, see also *universal class number* in manual "openFT (Unix and Windows systems) - Installation and Operation". This attribute only has any point in the case of FTAM partners.

g

GraphicString

The file can contain characters from the G0 set defined in ISO646 or ISO8859-1, or from the G1 set defined in ISO8859-1.

c

GeneralString

The file can contain characters from the C0 set defined in ISO646, the G0 set defined in ISO646 or ISO8859-1, or the G1 set defined in ISO8859-1. In the case of transfer with FTAM partners, each set is terminated with a CRLF (Carriage Return Line Feed); in this case, set boundaries do not necessarily correspond to the transfer unit boundaries.

i

IA5String

The file can contain characters from the C0 set and the G0 set defined in ISO646.
In the case of transfer with FTAM partners, each set is terminated with a CRLF (Carriage Return Line Feed); in this case, set boundaries do not necessarily correspond to the transfer unit boundaries.

v

VisibleString

The file can contain characters from the G0 set defined in ISO646.

-cs not specified

The previous character set remains unchanged.

-rf=v | **-rf=f** | **-rf=u**

This indicates the record format of the data to be transferred to a partner.

v (variable)

The data is transferred to a partner in records of variable length. Please note that, in the case of FTAM partners, in accordance with the A/111 profile, only text data from the GraphicString or VisibleString character sets can be transferred in this way. Binary files in a user format (where a record comprises a record length field and the data) can only be transferred to an FTAM partner in records of variable length, if the FTAM partner supports the userformat.

f (fix)

The data is transferred to an partner in records of equal length. Please note that, in the case of FTAM partners, in accordance with the A/111 profile, only text data from the GraphicString or VisibleString character sets can be transferred in this way.

Binary files of fixed record length (the file is made up of records of identical length) can only be transferred to an FTAM partner if the partner supports this fixed length for binary files.

u (undefined)

The record length used to transfer the data is not mapped to the real system. This means that the record length used for the transfer is not identical to that in the real file.

Binary files are stored in a bit string in the real system. Please note that in accordance with the A/111 profile, it is only possible to transfer text data from the GeneralString or IA5String character sets, or binary data with this record format. Any record structure present in text files is also lost unless maintained using other mechanisms (e.g. CRLF line separation for the transfer of IA5 or GeneralString files with FTAM).

-rf not specified

The previous record format remains unchanged.

-rl=record length

Defines the record length in bytes with which the data is to be transferred to an FTAM partner. The maximum record length is 65535 bytes.

-rl not specified

The previous record length remains unchanged.

-pa=[**n**][**r**][**i**][**p**][**x**][**e**][**a**][**c**][**d**]

Defines the "permitted actions" and how an FTAM partner can access a local file. This parameter does not affect the access rights of a file in a local system but instead places additional constraints on the access possibilities for FTAM partners.

The following values can be specified for the *permitted actions* parameter:

n, *r*, *i*, *p*, *x*, *e*, *a*, *c*, *d*, or any combination of these values:

n

means that an FTAM partner cannot access this file. If *n* is specified, all other options are ignored.

r

means that an FTAM partner can read the file.

r not specified

The file cannot be read.

i

with FTAM partners means that the FTAM partner can insert data units, such as records, in the file.

i not specified

No data units can be inserted in the file.

p

means that an FTAM partner can overwrite the file.

p not specified

The file cannot be overwritten.

x

means that an FTAM partner can append data to the file.

x not specified

The file cannot be extended.

e

with FTAM partners means that the FTAM partner can delete data units, such as records, from the file.

e not specified

No data units can be deleted from the file.

a

means that an FTAM partner can read the attributes of the file.

a not specified

The file attributes cannot be read.

c

means that an FTAM partner can change the attributes of the file.

c not specified

The file attributes cannot be changed.

d

means that an FTAM partner can delete the file.

d not specified

The file cannot be deleted.

-pa not specified

The access rights remain unchanged.

-np=file access password | **-np=@n**

This parameter is reserved for special customer applications.

For *file type*, *character set*, and *record format*, you should select combinations that correspond to the file contents:

Entries for	**-ft=**	**-cs=**	**-rf=**
Text files	t	g	f
	t	g	v
	t	v	f
	t	v	v
	t	c	u
	t	i	u
Structured binary files	b	No entry	v
Unstructured binary files	b	No entry	u
Binary files with fixed recorsd length	b	No entry	f

Otherwise, file inconsistencies may occur. File access errors are also possible if the record format is set to *f*, but no record length is specified or the file size is not a multiple of the record length.

Examples

1. FTAM partners:

   You wish to reset the access rights of the local file junk such that no FTAM partner can access the file.

   ```
   ftmodf junk -pa=n
   ```

2. openFT partners

   The combination of *-ft=b* and *-rf=f* is also significant for file transfer with the openFT protocol. In this way, a BS2000 partner, for example, can fetch a file containing binary data from a Unix or Windows system and store it in BS2000 as a SAM file. To do this, the following entries are required in the Unix or Windows system and BS2000 systems.

   Unix or Windows system *ftclient*:

   ```
   ftmodf binfix06 -ft=b -rf=f -rl=14156
   ```

   BS2000:

   ```
   ncopy from,ftclient,(binfix06,l=*n), -
            *a('binfix.06',,'binfixprofile'),data=*bin
   ```

3.34 ftmodi

Note on usage

Function: Modify an instance

User group: FT administrator

Functional description

The *ftmodi* command allows you to assign another Internet host name address to an instance.

Note on using more than one instance

- All instances must be explicitly assigned their own host name(option -addr with ftmodi or ftcrei). This also applies to standard instances.
- Only on Windows systems: Using several openFT instances is only possible with the transport system TCP/IP. If you want to use several instances and you are using the TNS, you must delete all TNS entries specific to openFT which do not relate to TCP/IP.

Format

ftmodi -h | <instance 1..8> [-addr=<host name> | -addr=@n]

[-ua=<user ID 1..32>]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

instance

Name of the instance to be modified. Instance names have a maximum length of 8 characters and must consist of alphanumeric characters. The first character must not be a number.

-addr=host name | **-addr=@n**

Internet host name whose assigned IP address is used to address the instance externally (destination address) and which is used as the sender address with outgoing connections. Changing *-addr* does not affect the instance's operating parameters *instance ID* and *processor*.

host name

A particular or another Internet host name can be assigned to the instance here.

@**n** for *host name*

This specification is only permitted for the standard instance *std*.

The standard instance is not assigned a particular host address anymore, and therefore it signs on for all addresses of the system.

In this manner you can switch from an operation with several instances to a one instance operation.

-ua=user ID

With this parameter *root* allocates an openFT instance to another user in single-user mode on Unix systems. 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.

The changing of the owner of an instance with *ftmodi* should as a matter of principle not be done during ongoing openFT operation because 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.

Examples

1. The host with the name MAPLE is assigned to the standard instance. Local requests to 127.0.0.1 are thus no longer possible. The command is as follows:

   ```
   ftmodi std -addr=MAPLE
   ```

2. The standard instance is to log in with all IP addresses of a system again and listen to all addresses. The command is as follows:

   ```
   ftmodi std -addr=@n
   ```

Messages of the ftmodi command

If *ftmodi* could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case.

3.35 ftmodk

Note on usage

Function: Modify RSA key

User group: FT administrator

Functional description

You can use the *ftmodk* command to modify the expiration date and authentication level of keys that are used for the authentication of partner systems. The changes are stored in the relevant key file.

Once the expiration date of a key has been reached, authentication using this key is rejected. However, you can still modify the expiration date after the key's validity has expired, e.g. in order to temporarily re-enable so that a current key can be transferred securely.

Format

ftmodk -h |

[-id=<identification1..64> | -id=@a] |
[-pn=<partner 1..200> | -pn=@a]
[-al=1 | -al=2]
[-exp=[yyyymmdd]]

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-id=identification | **-id=@a**

identification is the instance identification of the partner whose key is to be modified. *-id* must not be specified in combination with *-pn*.

@a

The installed keys of all partner systems are modified.

-pn=partner | **-pn=@a**

partner is the name of the partner system in the partner list or the address of the partner system whose key is to be modified.

-pn must not be specified in combination with *-id*.

You will find detailed information on address specifications in the section “Specifying partner addresses” .

@a

The installed keys of all partner systems are modified.

Neither *-id* nor *-pn* specified

The installed keys of all partner systems are modified.

-al=1 | **-al=2** (authentication level)

Specifies the authentication level for the key or keys.

1

The authentication level for the partner or partners is set to 1. This corresponds to the possibilities available up to openFT V11.0A.

If the partner system is subsequently authenticated at level 2 then the entry AUTHENTICATION-LEVEL=2 is automatically recorded in its key file.

2

The partner system supports the level 2 authentication procedure introduced in openFT V11.0B. Level 1 authentication attempts are rejected.

-al not specified

The authentication level is unchanged.

-exp=[yyyymmdd]

Specifies the expiration date of the key or keys.

yyyymmdd

Expiration date in the format yyyymmdd, e.g. 20171231 for 31.12.2017. The key or keys can be used for authentication at the latest up until 00:00 on the specified date.

No date specified

exp= without a date specification means that there is no expiration date for the key or keys.

-exp not specified

The expiration date of the key or keys is unchanged.

3.36 ftmodo

Note on usage

Function: Modify operating parameters

User group: FT administrator

Functional description

You can use *ftmodo* to define and modify the following parameters for openFT operation:

- the key length of the RSA key
- the minimum RSA key length
- the minimum AES key length
- the maximum values for file transfer
- the identification and the name of the local system
- the default value for the security level
- the mode for sender verification
- the global setting for sender verification
- the logging scope (file transfer, directory transfer, FTAC, ADM requests)
- the traces scope
- the traps scope
- the automatic deletion of log records
- the switch-over of the log file and trace file
- the scope of measurement data recording
- the variant of the used code table
- the addresses for the individual protocols
- the settings for the remote administration server
- the use of TNS and CMX
- the settings used for user data encryption
- the global deactivation of the restart for outbound and inbound requests
- the configuration of the FarSync X.25 transport system
- the FT administrator

For FTAM operation, you can also activate, deactivate or specify the Application Entity Title (AET).

> You can also use the openFT Explorer to modify the operating parameters (exception: deactivation of the application entity title).

Format

ftmodo -h |

[-kl=0 | -kl=768 | -kl=1024 | -kl=2048 | -kl=3072 | -kl=4096]
[-klmin=0 | -klmin=768 | -klmin=1024 | -klmin=2048 | -klmin=3072 | -klmin=4096]

[-aesmin= | -aesmin=128 | -aesmin=256]
[-tu=<transport unit size 512..65535>]
[-pl=<process limit> | -pl=] (*Windows systems)*
[-pl=1 | -pl=] (*Unix systems*)
[-cl=<connection limit 1..255>]
[-admcl=<connection limit 1..255>]
[-admcs=n | -admcs=y]
[-admpriv=<FT admin name 1..36 (*Windows systems*) or 1..32 (Unix systems)> | -admpriv=system (*Windows systems*) or -admpriv=root (Unix systems)]

[-gadmpriv=<FT admin group name 1..32>]
[-rql=<maximum number of requests 2..32000>]

[-rqt=<request lifetime 1..400> | -rqt=]
[-id=<identification 1..64>]
[-p=<processor name 1..8>][-l=<station name 1..8>]

[-sl=<security level 1..100> | -sl=p]
[-ptc=i | -ptc=a | -ptc=t | -ptc=b]
[-lf=c][-lt=a | lt=f | lt=n] [-lc=a | -lc=m | -lc=r]
[-ltd=a | -ltd=f | ltd=n]
[-la=a | -la=f | -la=m | -la=n]
[-ld=n | -ld=f][-lda=<0..999>][-ldt=hhmm]
[-ldd=@d | Mo | Tu | We | Th | Fr | Sa | Su | <1..31>]
[-mon=n | -mon=f][-monr=[l|r][a|s]]
[-monp=a | -monp=[openft][,][ftam][,][ftp]]
[-tr=n | -tr=f | -tr=c]
[-trp=a | -trp=[openft][,][ftam][,][ftp][,][adm]]
[-trr=[l |r][a | s]][-tro=[b]][-troll=[s | d]
[-atpsv=[<partner 1..200>][,][<transfer admission 8..67> | @d]

[-atp=a | -atp=n | -atp=[[-]fts],[[-]rqs],[[-]rqc],

[[-]rqf],[[-]pts],[[-]ptu]]

[-tpc=a | -tpc=n | -tpc=[[-]sss],[[-]fts],

[[-]rqs],[[-]rqc],[[-]rqf],[[-]pts],[[-]ptu]]

[-ccs=<CCS name 1..8>]
[-fnccs=<csn>] (*Unix systems*)
[-acta=a | -acta=[openft][,][ftam][,][ftp][,][adm]]
[-ftp=<port number 1..65535> | -ftp=@s]
[-openft=[<port number 1..65535>][.<T-Sel 1..8>] |-openft=@s]
[-ftam=[<port number 1..65535>][.<T-Sel>[.<S-Sel>[.<P-Sel>]]] |-ftam=@s]

[-adm=<port number 1..65535> | -adm=@s]
[-ftstd=<port number 1..65535> | -ftstd=@s]
[-tns=y | -tns=n]

[-cmx=y | -cmx=n]
[-rco=n | -rco=f] [-rci=n | -rci=f]
[-ae=y | -ae=n]
[-aet=@n | -aet=@i | -aet=<AET 1..64>]
[-dp=n | -dp=f]
[-c= | -c=i | -c=o | -c=io | -c=oi]
[-x25=[<0..15>[=<DTE 1..15>] .. [,<0..15>[=<DTE 1..15>]]]] (*Linux systems)*

[-x25=[<0..3>:<0..3>[=<DTE 1..15>] ..[,<0..3>:<0..3>[=<DTE 1..15>]]]] (*Windows systems*)

[-openftx25=y | -openftx25=n]
[-openftx25lif=[0][,1][,2][,3]..[,15]] (*Linux systems*)
[-openftx25lif=[0][,1][,2][,3]] (*Windows systems*)
[-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]] (*Linux systems*)
[-ftamx25lif=[0][,1][,2][,3]] (*Windows systems*)
[-ftamx25lnb=<1..99>]
[-ftamx25cl=0/- | -ftamx25cl=2/0 | -ftamx25cl=2/2]
[-ftamx25nsap=<AFI 36 | .. | 59>.[<IDI 0..15>][.<DSP 0..38>] | 2..40]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-kl=0 | **-kl=768** | **-kl=1024** | **-kl=2048 | -kl=3072 | -kl=4096**

The *-kl* parameter can be used to change the length of the RSA key used in encryption. The value of the *kl* parameter specifies the new RSA key length (RSA-PROPOSED) in bits. The RSA key is only used for the encryption of the AES key agreed between the partners. The configured key length for RSA proposal must be greater than or equal to the specified minimum key length, otherwise a warning will be issued and the proposed key length will be adapted to the minimum key length.

openFT uses the AES key for encrypting request description data and any file content present.

The *ftmodo -kl=*... command can be specified in current openFT operation.

-kl=0 explicitly deactivates encryption. If this is set during operation then any requests with encryption (prior to *ftmodo -kl=0*) that have been submitted but not yet started are aborted with errors. Any running requests are processed and their encryption is retained. New requests using encryption are rejected.

After reinstallation, the default value *-kl=768* is used.

Default setting following initial installation: *-kl=2048*.

-klmin=0 | **-klmin=768** | **-klmin=1024** | **-klmin=2048 | -klmin=3072 | -klmin=4096**

This option specifies the minimum RSA key length.

0

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

768 | 1024 | 2048 | 3072 | 4096

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. That means: Since an RSA key set is always created on the open platforms during installation, an RSA key is always sent in the protocol during the subsequent data transfer. If this key is deleted and the partner requests encryption, then the partner rejects the connection with s Session Reject (SRJ) "connection not accepted without encryption".

Default setting following initial installation: *-klmin=0*.

-aesmin= | **aesmin=128** | **-aesmin=256**

This option specifies the minimum AES key length.

aesmin=

If you specify *aesmin=* (without specifying a key length) no minimum AES key length is set. Any AES key length and even requests with DES key will be accepted.

128 | 256

Only AES keys of the specified length or larger ones will be accepted. If the partner cannot fulfill this requirement the request will be rejected.

Default setting following initial installation: *-aesmin=* .

-tu=transport unit size

You use the parameter *-tu* to define the upper limit for message length at transport level (block length). You can choose a value between 512 and 65535.

The block length only applies to requests to openFT partners.

Default setting following initial installation: *-tu=65535*.

-pl=1 | **-pl=** (Unix systems)

Maximum number of processes used for the processing of asynchronous requests.

1

All asynchronous requests are processed by the same process.

No value specified

If you specify *-pl=* without parameters then the number of processes is equal to the number of connections, i.e. each connection is handled by a separate process.

Default setting following initial installation: *-pl=* (i.e. no number specified).

pl= process limit | **-pl=** (Windows systems)

process limit is the maximum number of openFT servers used for the processing of asynchronous requests.

process limit not specified

If you specify *-pl=* without parameters then the number of openFT servers is equal to the number of connections, i.e. each connection is handled by a separate openFT server.

-cl=connection limit

Maximum number of asynchronous requests that are processed simultaneously. Possible values: 1 to 255.

The default value is 16.

Default setting following initial installation: *-cl=16*.

> *-pl=2* means that a maximum of two openFT servers are used to process asynchronous requests. *-cl=16* means that a maximum of 16 requests can be processed simultaneously. However, this means that the second openFT server is not started until the first openFT server has reached its assigned limit of 8 connections! This value is calculated by dividing the value of *-cl* by the value of *-pl*.

-admcl=connection limit

Maximum number of connections provided for remote administration requests. Possible values: 1 through 255.

Read the note under *-admcs*.

Default setting following initial installation: *-admcl=8*.

-admcs=n | **-admcs=y**

Specifies whether the local openFT instance is flagged as a remote administration server.

y

Flags the local openFT instance as a remote administration server. This means that this instance can also be an ADM trap server.

n

The local openFT instance is not (no longer) flagged as a remote administration server. This means that it is not (no longer) possible to receive ADM traps. This is the default after a new installation.

> If you specify *-admcs*, but do not specify *-admcl*, then openFT sets the connection limit (*-admcl)* to the following value:
>
> 64 if *-admcs=y*.
>
> 8 if *-admcs=n*.

Default setting following initial installation: *-admcs=n* .

-admpriv=name of new FT administrator | **-admpriv**=**system** (*Windows systems*) or **root** (*Unix systems*)

The option is used to alter the FT administrator.

name of new FT administrator

The name of any active windows user with or without specifying domain e.g:

ftmodo -admpriv=hugo or *ftmodo -admpriv*=g02\hugo

This will result in giving FT administrator rights to user “hugo”. Since that point, only user “hugo” has access to FT administrator rights with or without local administrator rights. Any other user is not permitted to execute any FT administrator commands, even if utilizing local administrator rights.

system (*Windows systems*) or **root** (*Unix systems*)

This will switch back to the previous functionality, that is the FT administrator needs local administrator rights.

Alteration of FT administrator takes effect immediately, however as for GUI, when user has it opened, then it needs to be restarted in order for change to be applied.

So far until openFT 12.1C70 there was only one openFT administrator (as user or group of users) for whole openFT. Starting from version 12.1C80, every instance will have separate openFT administrator.

The administrator of the STD instance will be at the same time global administrator, who can handle global openFT settings.

The only difference is that this user will be administrator of instance, which is currently set in environment and user will be able to use administrative commands only on that instance.

Globaler Administrator

The user who is set to be the administrator of the STD instance, is the global administrator. The global administrator is the only user that is allowed to run commands that affect openFT globally (ftaddlic), and commands that administer instances (ftcrei, ftmodi, ftdeli).

Additionally, those commands will only work if the currently active instance is STD; Therefore, if global administrator switched to a different instance (using ftseti), those commands will not be able to be executed.

When those command are executed unauthorized, an error message is displayed:

<name_of_command>: Command is only available to the STD administrator.

-gadmpriv=name of FT administrator group (Linux group)

name of FT administrator group (Linux group)

The name of any local Linux group, every user in the group will have FT administrator rights.

The FT administrator cannot be assigned along with the FT administrator group, only one option has to be used at a time.

Alteration of FT administrator takes effect immediately, however as for GUI, when user has it opened, then it needs to be restarted in order for change to be applied.

rql=maximum number of requests

You use *-rql* to specify the maximum number of entries in the request queue. You can choose a value between 2 and 32000.

Default setting following initial installation: *-rql=2000*.

-rqt=request lifetime | **-rqt=**

You use *-rqt* to specify the maximum lifetime of requests in the request queue. The value applies to both inbound and outbound requests and is specified in days. Values between 1 and 400 are permitted. Once the specified period has expired, requests are deleted from the request queue.

request lifetime not specified:

If you specify *-rqt=* without parameters then the maximum lifetime is unlimited.

Default setting following initial installation: *-rqt=30*.

-id=identification

Specifying the instance identification of your instance. Partner systems using openFT Version 8.1 and later, address your system via this string. In return, openFT uses the instance ID as the sender address when addressing the partners. The instance ID must be unique and not case-sensitive (see also section “Instance identification”). If you modify the instance ID, the relevant public key files will be automatically updated.

Default setting following initial installation: *-id= local DNS name or host name*.

-p=processor name

You specify the processor name assigned to your system here.

No processor name is specified after initial installation.

-l=station name

The station name of the openFT application. The default value is $FJAM.

Default setting following initial installation: *-l=$FJAM*.

The specifications for *processor name* and *station name* depend on how your system is connected to the network. Further details can be found in the manual "openFT (Unix and Windows systems) - Installation and Operation".

-sl=security level | **-sl=p**

You use this option to define the default security level. This level applies to partners in the partner list to which no explicit security level value was assigned when they were entered with *ftaddptn*. The effect also depends on the settings for the admission set, see the *ftmoda* command.

security level

Specifies a fixed default security level. Values between 1 and 100 are permitted. 1 indicates a very low and 100 a very high requirement for protection with regard to the partners.

p

The default security level depends on the partner's attributes:

- Security level 10 if the partner has been authenticated.
- Security level 90 if the partner is known in the transport system and is identified by the name it is known by in the transport system.
- Security level 100 otherwise, i.e. if the partner has only been identified by its address.

Default setting following initial installation: *-sl=p*.

-ptc=i | **-ptc=a** | **-ptc=t** | **-ptc=b**

This allows you to modify the global settings for sender verification. This setting only applies to named partners which are connected via the openFT or FTAM protocol and to which the following applies:

- Partners connected via the openFT protocol do not use authentication
- Using the FTAM protocol the partner identifies itself with a Calling Application Entity Title which does not correspond to the nil AP Title

In the case of dynamic partners and FTP partners, this setting has no effect.

i (identification)

openFT partners are checked via the identification. FTAM partners are checked via the transport address; any specified Calling Application Entity Title is ignored.

a (address)

openFT partners are checked via identification and additionally via the transport address. FTAM partners are checked via the transport address; any specified Calling Application Entity Title is ignored.

If the transport address under which the partner logs in does not correspond to the entry in the partner list then the request is rejected.

t (application entity title)

openFT partners are checked via the identification. FTAM partners are checked via the Calling Application Entity Title in case it is specified and does not correspond to the nil AP title; otherwise, the transport address is checked.

b (both)

openFT partners are checked via identification and additionally via the transport address. FTAM partners are checked via the transport address as well as via the Calling Application Entity Title in case it is specified and does not correspond to the nil AP title; otherwise, the transport address is checked.

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 to set the option *-ptc* at this value from the start as existing ambiguities could persist.

Default setting following initial installation: *-ptc=i*.

-lf=c

The log file is changed.

The new log file is created under the name *syslog.Lyymmdd.Lhhmmss*:

- *yymmdd* is the date (year, month, day) on which the file was created,
- *hhmmss* is the time (hour, minute, second for GMT) on which the file was created.

-lt=a | **-lt=f** | **-lt=n**

This option is used to selectively deactivate FT log records. Possible values:

a (all)

Log records are written for all FT requests.

f (failure case)

Log records are written for failed FT requests only.

n (none)

No log records are written.

-lt does not affect the logging of directory transfer, i.e. the options *-lt* and *-ltd* are independent from each other.

Default setting following initial installation: *-lt=a*.

-lc=a | **-lc=m** | **-lc=r**

This option is used to selectively activate/deactivate FTAC log records. Possible values:

a (all)

Log records are written for all FTAC access checks.

m (modifying FM calls)

Log records are written for all modifying file management requests leaving the remote system as well as for all rejected FTAC access checks.

r (reject case)

Log records are written for rejected FTAC access checks only.

Default setting following initial installation: *-lc=a*.

-ltd=a | **-ltd=f** | **-ltd=n**

This option is used to selectively activate the logging of directory transfer (FT-DIR logging). The following parameters are available:

a (all)

Log records are written for all individual FT requests, i.e.:

- one log record for each transferred file
- one log record for each transferred directory or subdirectory

f (failure case)

Log records are written for failed individual FT requests only.

n (none)

No log records are written for individual FT requests. I.e. only the log record of the main FT request and FTAC log records are written.

-ltd does not affect the logging of individual file transfer, i.e. the options *-lt* and *-ltd* are independent from each other.

Default setting following initial installation: *-ltd=n*.

-la=a | **-la=f** | **-la=m** | **-la=n**

This option allows you to selectively activate the logging of administrative requests. The following parameters are available:

a (all)

Log records are written for all administration requests.

f (failure)

Log records are only written for failed administration requests.

m (modifying)

Log records are written for all administration requests that make modifications.

n (none)

No log records are written for administration requests.

Default setting following initial installation: *-la=a*.

-ld=n | **-ld=f**

This option allows you to control whether log records are deleted automatically.

n (on)

Activates the automatic deletion of log records. This activates the criteria specified in *-lda*, *-ldt* and *-ldd* (minimum age and deletion interval).

f (off)

Deactivates the automatic deletion of log records. When this option is set, the settings made for *-lda*, *-ldt* and *-ldd* have no effect.

Default setting following initial installation: *-ld=f*.

-lda=0..999

Minimum age of log records for deletion in days. The days are counted back from the deletion time specified in *-ldt*. The value 0 deletes all the log records that were written before or on the time of the current day specified in *-ldt*.

Default setting following initial installation: *-lda=14*.

-ldt=hhmm

Specifies the (local) time at which the log records are to be deleted. Depending on the system, the delete function may be executed up to 5 minutes before the time specified here.

Default setting following initial installation: *-ldt=0000* (i.e. time = 00:00).

-ldd=@d | **Mo** | **Tu** | **We** | **Th** | **Fr** | **Sa** | **Su** | 1..31

Specifies the day on which the log records are to be deleted.

Mo | **Tu** | **We** | **Th** | **Fr** | **Sa** | **Su**

Delete every week on the selected weekday (Mo=Monday, .. Su=Sunday).

1..31

Delete every month on a specific day of the month (1-31). If the value 29, 30 or 31 is specified for a month that has fewer days than this then deletion is performed on the last day of the month.

@d

The log records are deleted every day.

Default setting following initial installation: *-ldd=@d* (i.e. delete every day).

-mon=n | **-mon=f**

This allows you to activate and deactivate openFT monitoring.

n (on)

openFT monitoring is activated.

f (off)

openFT monitoring is deactivated.

Default setting following initial installation: *-mon=f*.

-monr= | **-monr=**[**l**|**r**][**a**|**s**]

This allows you to select openFT monitoring depending on the request type. The value *l* or *r* can be combined with *a* or *s* (Boolean AND, e.g. *la*, *al*, *ls*, *rs*, ...).

l (local)

Monitoring data is collected for requests issued locally.

r (remote)

Monitoring data is collected for requests issued remotely.

a (asynchronous)

Monitoring data is collected for asynchronous requests. Requests issued remotely are always regarded as asynchronous.

s (synchronous)

Monitoring data is collected for synchronous requests. Synchronous requests are always issued locally.

No request type specified

If you specify *-monr=*, monitoring data is collected for all requests.

Note that *-monr=rs* does not completely deactivate monitoring. *-monr=rs* has the same effect as *-monp=*. See the *ftshwm* command, section “Description of the monitoring values” .

Default setting following initial installation: *-monr=*.

-monp= | **-monp=a** | **-monp=**[**openft**][**,**][**ftam**][**,**][**ftp**]

This allows you to select openFT monitoring depending on the protocol type used for the partners. Combinations are also permitted if you specify the protocols individually (separated by commas).

a

Monitoring data is collected for all partners.

openft

Monitoring data is collected for openFT partners.

ftam

Monitoring data is collected for FTAM partners.

ftp

Monitoring data is collected for FTP partners.

No protocol type specified

If you specify *-monp=* with no parameters, the monitoring is deactivated for partners. In this event, only certain monitoring data values are populated. See the *ftshwm* command, section “Description of the monitoring values” .

Default setting following initial installation: *-monp=a*

-tr=n | **-tr=f** | **-tr=c**

This allows you to activate and deactivate the openFT trace function.

n (on)

The openFT trace function is activated.

f (off)

The openFT trace function is deactivated.

c (change)

The current trace file is closed and a new one is opened.

Default setting following initial installation: *-tr=f*.

-trp=a | **-trp=**[**openft**][**,**][**ftam**][**,**][**ftp**][**,**][**adm**]]

This allows you to select the openFT trace function depending on the type of protocol used for the partners by specifying a comma-separated list of one or more protocol types. All the partners that are addressed via this or these protocol type(s) are then traced.

You can modify the selection made here on a partner-specific basis, see the *-tr* option in the *ftmodptn* command.

a (all)

All protocol types, and consequently all partners, are selected for tracing.

openft

All partners addressed via the openFT protocol are selected for tracing.

ftam

All partners addressed via the FTAM protocol are selected for tracing.

ftp

All partners addressed via the FTP protocol are selected for tracing.

adm

All partners addressed via the FTADM protocol are selected for tracing.

No protocol type selected

If you specify *-trp=* without parameters then no partner is selected for tracing. In this case, only those partners for which tracing has been activated on a partner-specific basis using *ftmodptn ... tr=n* are traced.

Default setting following initial installation: *-trp=a*.

-trr=[**l** | **r**][**a** | **s**]

This option allows you to select the request types that are to be traced. The value *l* or *r* can be combined with *a* or *s* (Boolean AND, e.g. *la*, *al*, *ls*, *rs*, ...).

l (local)

All locally submitted requests are selected for tracing.

r (remote)

All remotely submitted requests are selected for tracing.

a (asynchronous)

All asynchronous requests are selected for tracing. Requests issued remotely are always regarded as asynchronous.

s (synchronous)

All synchronous requests are selected for tracing. Synchronous requests are always issued locally.

No request type specified

If you specify *-trr=* without parameters then all requests are selected for tracing.

Note that *-trr=rs* does not completely deactivate tracing. Interface trace files, for instance, continue to be created (if activated).

Default setting following initial installation: *-trr=*.

-tro=[**b**]

You can use *-tro* to select options for the trace function. These options are only effective if the trace function is active.

b (no bulk data)

Minimum trace. Only protocol elements with no file contents (bulk data) are written to the trace file. In the case of protocol elements with file contents, the trace file simply notes that records have been suppressed at this point. This note is entered only once for a sequence of similar records.

No option specified

If you specify *-tro=* without parameters then the trace is written normally.

Default setting following initial installation: *-tro=*.

-troll=[**s** | **d**]

You use *-troll* to define the scope of the trace for the lower protocol layers. This option is effective only if the trace function is activated.

s (standard)

Additional entries are written in the standard scope for the lower protocol layers. The standard scope comprises comprehensive logging of the calls, their arguments, the content of any options and the user data.

d (detail)

In addition to the standard scope, internal events and transport system information (e.g. system calls) are written for the lower layers.

No option specified

If you specify *-troll=* with no parameters, no trace is performed for the lower protocol layers.

> Note on operation with and without CMX:
>
> - In the case of operation without CMX, the trace entries for the lower protocol layers are written to the openFT trace.
> - In the case of operation with CMX, CMX traces are generated and stored in the *traces* directory of the associated openFT instance. These can then, for example, also be selected and displayed in the openFT Explorer (*Adminis-tration* menu, *Open Trace File* command).
>
> Using this option, it is therefore possible to activate and deactivate CMX traces during active CMX operation.

Default setting following initial installation: *-troll=* .

-atpsv=[partner][**,**][transfer admission | @**d**]

-atpsv= allows you to specify the settings for the ADM trap server. When you enter the ADM trap server for the first time, you must specify both the partner and the transfer admission. You can subsequently change each of the two parameters individually.

partner

Name or address of the partner to which the ADM traps are sent. This must either be a name from the partner list or the address must be specified in the form *ftadm://host...*. See the section “Specifying partner addresses” .

transfer admission

FTAC transfer admission for accessing the ADM trap server.

@**d** for *transfer admission*

If you specify *@d* (blanked), the transfer admission is queried on screen after the command has been sent. Your input is blanked.

neither *partner* nor *transfer admission* specified

If you specify *-atpsv=* without parameters, you remove the ADM trap server. This means that ADM traps are no longer sent.

Default setting following initial installation: *-atpsv=*.

-atp=a | **-atp=n** | **-atp=**ADM trap list (comma-separated)

-atp allows you to activate and deactivate ADM traps. The ADM trap server to which the ADM traps are to be sent is specified with *-atpsv*.

The following specifications are possible with the *-atp* option:

a (all)

All ADM traps are written.

n (none)

No ADM traps are written.

fts

Activates the ADM traps on the status of the asynchronous server.

-fts

Deactivates the ADM traps on the status of the asynchronous server.

rqs

Activates the ADM traps on the status of the request queue.

-rqs

Deactivates the ADM traps on the status of the request queue.

rqc

Activates the ADM traps when a request has been terminated successfully.

-rqc

Deactivates the ADM traps when a request has been terminated successfully.

rqf

Activates the ADM traps when a request has failed.

-rqf

Deactivates the ADM traps when a request has failed.

pts

Activates the ADM traps on the status of the partner system.

-pts

Deactivates the ADM traps on the status of the partner system.

ptu

Activates the ADM traps if a partner system is not available.

-ptu

Deactivates the ADM traps if a partner system is not available.

Default setting following initial installation: *-atp=n*.

-tpc=a | **-tpc=n** | **-tpc=**Console trap list (comma-separated)

You use *-tpc* to activate and deactivate console traps.

In Unix and Windows systems, console traps are written to the openFT file *conslog*. In Unix, BS2000 and z/OS systems they are also output at the console and in Windows systems they are also written to the event log.

For *-tpc* you can enter the following values:

a (all)

All traps are written.

n (none)

No traps are written.

sss

Activates traps relating to the status of the openFT subsystem.

-sss

Deactivates traps relating to the status of the openFT subsystem.

fts

Activates traps relating to the status of the asynchronous server.

-fts

Deactivates traps relating to the status of the asynchronous server.

rqs

Activates traps relating to the status of the request queue.

-rqs

Deactivates traps relating to the status of the request queue.

rqc

Activates traps on the successful termination of a request.

-rqc

Deactivates traps on the successful termination of a request.

rqf

Activates traps on the unsuccessful termination of a request.

-rqf

Deactivates traps on the unsuccessful termination of a request.

pts

Activates traps relating to the status of partner systems.

-pts

Deactivates traps relating to the status of partner systems.

ptu

Activates traps when a partner system is inaccessible.

-ptu

Deactivates traps when a partner system is inaccessible.

Default setting following initial installation: *-tpc=n*.

-ccs=CCS name

You use *CCS name* to define a new character set which is represented by a code table. This character set is then used as the new default value for transfer requests (*ft, ncopy*). The code table specification is only relevant for requests to openFT partners.

Another character set can be explicitly assigned for *ft* and *ncopy* (options *-lc* and *-rc*).

You can also define your own character set. For details concerning CCS names and the associated code tables, see the manual "openFT (Unix and Windows systems) - Installation and Operation".

Default value following initial installation:

-ccs=iso88591 (Unix systems, corresponds to ISO8859-1)

-ccs=CP1252 (Windows systems)

-fnccs=ccsn (only Unix systems)

Specifies the character set, into which the local path names and scripts to be run are converted for inbound openFT requests in character mode, i.e.:

- When listing files and directories in character mode local file names and attributes on the inbound side are interpreted in this character set.
- The output of commands called via *ftadm* in character mode as well as commands called via *ftexec* with *-rc=*SYS* are interpreted in this character set.

Default setting following initial installation: No conversion for inbound requests in character mode.

-acta=a | **-acta=**[**openft**][**,**][**ftam**][**,**][**ftp**][**,**][**adm**]

This option allows you to activate or deactivate the asynchronous inbound server. You can activate the asynchronous inbound server for specific protocols (openFT, FTP, FTAM, ADM), by specifying a comma-delimited list of one or more protocol types.

a

The asynchronous inbound servers are activated for all installed protocol types.

openft

Activates the asynchronous inbound server for requests via the openFT protocol.

ftam

Activates the asynchronous inbound server for requests via the FTAM protocol. A warning is issued if the FTAM protocol is not installed.

ftp

Activates the asynchronous inbound server for requests via the FTP protocol. A warning is issued if the FTP protocol is not installed.

adm

Activates the asynchronous inbound server for administration requests.

No protocol type specified

Specifying *-acta=* without parameters deactivates all asynchronous inbound servers.

> **i** If you specify a list of protocol types then the asynchronous inbound servers of the non-specified protocol types are deactivated!

Default setting following initial installation: *-acta=openft,ftam,adm* .

-ftp=port number | **-ftp=@s**

You use *port number* to specify the port number used by FTP.

Possible values: 1 to 65535.

@s

Sets the port number for FTP server to the default value of 21.

Default setting following initial installation: *-ftp=@s*.

-openft=[port number][**.**T-selector] | **-openft=@s**

port number

You can use *port number* to specify a port number other than the default for the local openFT server.

Possible values for the *port number*: 1 to 65535

T-selector

You can also specify a T-selector of between 1 and 8 characters in length. You can specify the selector in printable or hexadecimal format (0xnnnn...). Alphanumeric characters and the special characters # @ $ are permitted for printable selectors.

A printable selector will be converted to uppercase, coded in EBCDIC in the protocol and may be padded with spaces internally to the length of eight characters. In this case, the port number and T-selector must be separated by a period.

@s

-openft=@s sets the port number and the T-selector for the openFT server to the default value, i.e. 1100 and $FJAM.

> **i** Please use this function carefully because setting a port number or T-selector other than the default makes it difficult for openFT partners to address the local system!

Default setting following initial installation: *-openft=@s* (d.h. *1100* und *$FJAM*).

Notes on operation with TNS

- If you are switching from operation without TNS to operation with TNS (*-tns=y*) and if only the T-selector with no port number had previously been set under *-openft*, you must specify the port number explicitly, even if it matches the default value. This is necessary to ensure that the T-selector cannot be confused with the global name in the TNS.
- For operating with TNS, you can specify a TNS name other than the default for the local openFT server. A period must be placed before the TNS name, e.g. *-openft=.OPNFTSRV.* The TNS name must not contain any period.

In the case of operation with TNS, the default value for the TNS name is $FJAM.

-ftam=[port number][**.**T-selector[**.**S-selector[**.**P-selector]]] | **-ftam=@s**

port number

You can use *port number* to specify a port number other than the default for the local FTAM server.

Possible values for the port number: 1 to 65535

The default value for the port number is 4800.

T-selector**.**S-selector**.**P-selector

You can also specify a T-selector, a session selector and a presentation selector, each of which may have a length of 1 to 16 characters. In this case, the port number, T-selector, S-selector and P-selector must be separated by a period. You can specify the selectors in printable or hexadecimal format (0xnnnn...)

T-selectors that start with $FTAM (default value) are coded in EBCDIC and padded with spaces to the length of 8 characters. In the protocol, all other printable T-selectors as well as all printable session and presentation selectors are converted to uppercase and coded with variable length in ASCII.

The default value for *T-selector* is $FTAM.

S-selectors and *P-selectors* do not have default values because, by default, openFT-FTAM does not use these selectors.

> **i** Make sure that you carefully harmonize the specifications for the port number, the transport selector, the session selector and the presentation selector (in this option or in the relevant TNS entry) with your FTAM partners.

@s

-ftam=@s sets the port number and the TNS name for the FTAM server to the default value, i.e. 4800 and $FTAM.

Default setting following initial installation: *-ftam=@s*.

Notes on operation with TNS

- If you switch to operation with TNS again (*-tns=y*) and if only the T-selector with no port number had previously been set under *-ftam*, you must specify the port number explicitly, even if it matches the default value. This is necessary to ensure that the T-selector cannot be confused with the global name in the TNS.

- For operating with TNS, you can specify a TNS name other than the default for the local FTAM server. A period must be placed before the TNS name, e.g. *-ftam=.FTAMSERV.* The TNS name must not contain any period.

In the case of operation with TNS, the default value for the TNS name is $FTAM.

-adm=port number | **-adm=@s**

port number allows you to specify the port number used for remote administration.

Possible values: 1 to 65535.

@s

-adm=@s resets the remote administration port number to the default value of 11000.

Default setting following initial installation: *-adm=@s*.

-ftstd=port number | **-ftstd=@s**

You use *port number* to define the default port number for the addressing of openFT partners via partner addresses.

Possible values: 1 to 65535

Take care when using this option, because when you change the value of the option, openFT partners that use the default openFT port number 1100 can only be accessed if the port number is specified explicitly.

@s

-ftstd=@s resets the default port number for the addressing of openFT partners via partner addresses. The default port number of 1100 then applies again.

Default setting following initial installation: *-ftstd=@s*.

-tns=y | **-tns=n**

This option allows you to activate or deactivate the use of TNS names. This does not affect the use of TCP/IP host names, IP addresses or partner management, or the explicit specification of the port number and selectors with the *-openft=* and *-ftam=* options.

For operation with TNS to be possible, operation with CMX must be activated (*ftmodo -cmx=y*).

y

This activates the use of TNS names for openFT and FTAM transfer.

This is necessary, for example, if other transport protocols are to be used alongside TCP/IP.

n

This deactivates the use of TNS names. In this case, it is only possible to use the TCP/IP transport protocol. By default, the port numbers set in the operating parameters are used for communications (options *-openft*, *-ftam* and *-ftstd*).

> **Caution!**
>
> This option should not be changed as long as requests are stored or active. Activation and deactivation of the TNS database can cause the conversion of a partner name to a partner address to change, which could in turn lead to requests failing (above all with restart requests) or to unwanted delivery of files. After switchover, temporary partner entries can also appear twice in the partner list for a while (see *ftshwptn*), even if the partner name is converted to the same address in both cases.

Default setting following initial installation: *-tns=n* .

-cmx=y | **-cmx=n**

This option allows you to switch between operation with CMX and operation without CMX. You can only perform this switchover if the asynchronous openFT server has not been started. You may therefore first have to shut down the asynchronous openFT server, e.g. with *ftstop*.

If you want to work with TNS then operation with CMX must be activated.

y

Switches to operation with CMX. This is only possible if CMX is installed in the minimum version required for operation with this openFT version. If CMX is not installed or is not installed in the correct version then the ftmodo command is rejected with an error message.

n

Switches to operation without CMX.

Default setting following initial installation: *-cmx=n*.

-rco=n | **-rco=f**

-rci=n | **-rci=f**

With the options *-rco* (recovery outbound) and *-rci* (recovery inbound) you can globally deactivate the restart function for outbound and inbound requests.

-rco=n (on)

The recovery for outbound jobs is activated for all partners for which there is no own partner-specific setting.

-rco=f (off)

The recovery for outbound jobs is deactivated for all partners for which there is no own partner-specific setting.

-rci=n (on)

The recovery for inbound jobs is activated.

-rci=f (off)

The recovery for inbound jobs is deactivated.

Default setting following initial installation: *-rco=n*, *-rci=n*.

-ae=y | **-ae=n**

This option activates/deactivates the AET (Application Entity Title).

y

A "nil Application Entity Title" is included as the calling or called Application Entity Title (AET) for transfer using the FTAM protocol.

n

The AET is deactivated. The option only has to be reset to *-ae=n* if FTAM partners, as responders, do not expect to receive an AET.

Default setting following initial installation: *-ae=y*.

-aet=@n | **-aet=@i** | **-aet=AET**

With this option you can specify the AET (Application Entity Title). It is sent in the initiator role as "calling AET" and in the responder role as "responding AET".

@n

The setting of *-ae=* is valid: if *-ae=y* is set, the nil APTitle is used. If *-ae=n* is set, no Application Entity Title is sent.

@i

The instance identification is used as Application Entity Title (*ftmodo -id=...*). In this case you should avoid syntactically incorrect specifications in the instance identification for the Application Entity Title.

In most of these situations, the nil APTitle is currently set instead; this can change in future versions.

AET

The explicit indication of a Calling/Responding Application Entity Title.

The format rules apply as described in the manual "Concepts and Functions".

-dp=n | **-dp=f**

You use this option to specify whether or not dynamic partners are permitted.

n (on)

Dynamic partners are permitted. Partners can then be accessed via their address irrespective of whether they are entered in the partner list or not.

f (off)

Dynamic partners are not permitted, i.e. partners cannot be accessed via their address. As a result, it is only possible to use partners that are entered by name in the partner list and are addressed via the partner name.

Default setting following initial installation: *-dp=n*.

-c= | **-c=i** | **-c=o** | **-c=io** | **-c=oi**

You use this data to control system-wide encryption of user data and file and/or directory list attributes. This setting applies to transfer requests file management requests and administration requests.

i

Activates inbound encryption, i.e.:

- Inbound requests must transfer the user data in encrypted form as otherwise they are rejected.
- Inbound FTAM requests and inbound FTP requests are rejected.

o

Activates outbound encryption, i.e.:

- Outbound requests transfer the user data in encrypted form even if no encryption has been specified in the request (e.g. *ft*, *ncopy*, program interface, openFT Explorer). In addition applies:
- Outbound FTAM requests are rejected.
- Outbound FTP requests are, however, permitted.
- File management requests transfer the file and directory list attributes in encrypted form, even if no encryption was called for in the request (*ftshw*).
  However, if the partner system doesn't support encryption, then the file and/or directory list attributes will be transferred without encryption.

io, **oi**

Activates inbound and outbound encryption, i.e. both the statements on inbound encryption and on outbound encryption apply.

No encryption option specified

Specify *-c=* to deactivate system-wide encryption of user data and file and/or directory list attributes. If encryption is required then this must be explicitly specified in the request.

> System-wide encryption may only be activated if openFT-CR is installed.

Default setting following initial installation: *-c=* .

With the following options you configure the FarSync X.25 transport system.

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

-x25= [Adapter **:** Line[**=** DTE address]...] (Windows systems)

-x25=[Adapter[**=**DTE address]...] (Linux systems)

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.

Identification of X.25 lines under Windows:

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

You can specify up to four adapter numbers and line numbers at a time, so that a maximum of 16 lines results.

Identification of X.25 lines under Linux :

A line is uniquely defined via the adapter number. All lines for all adapters are numbered starting with 0.

-x25= Adapter **:** Line **=** DTE address (Windows systems)

-x25=Adapter**=**DTE address (Linux systems)

Assigns a DTE address to an X.25 line.

Default setting following initial installation: a DTE address is not assigned to any line.

-x25= Adapter **:** Line **=** (Windows systems)

-x25=Adapter**=** (Linux systems)

Resets the assigned DTE address to an X.25 line, i.e. the DET address is deleted.

-x25=

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

-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 following initial installation: *-openftx25=n*

-openftx25lif= [**0**][**,1**][**,2**][**,3**] (Windows systems)

-openftx25lif=[**0**][**,1**][**,2**][**,3**]..[**,15**] (Linux systems)

[0],[1] specifies 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.

No adapter number specified:

-openftx25lif= resets the openFT protocol's 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.

Default setting following initial installation: *-openftx25lif=0*

-openftx25lnb=number of list calls

With this option you specify the number of list calls per FarSync X.25 card.

Possible values: 1 to 99

Default setting following initial 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 *-openftx-25lif*.

-openftx25cl=0/- | **-openftx25cl=2/0** | **-openftx25cl=2/2**

Specifies which transport class is accepted for incoming connections for the openFT protocol.

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 following initial installation: *-openftx25cl=2/0*

-openftx25nsap=network address of the local openFT application

You can enter the network address (NSAP) of the local openFT application, 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*.

No network address specified:

-openftx25nsap= resets the openFT protocol's setting for the local NSAP.

Default setting following initial installation: *-openftx25nsap=*

-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 following initial installation: *-ftamx25=n*

-ftamx25lif= [**0**][**,1**][**,2**][**,3**] (Windows systems) **-ftamx25lif=**[**0**][**,1**][**,2**][**,3**]**..**[**,15**] (Linux systems)

[0],[1]...specifies 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.

No adapter number specified:

-ftamx25lif= resets 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.

Default setting following initial installation: *-ftamx25lif=0*

-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 following initial 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 *-ftamx-25lif*.

-ftamx25cl=0/- | **-ftamx25cl=2/0** | **-ftamx25cl=2/2**

Specifies 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 following initial 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*.

No network address specified

-ftamx25nsap= resets the FTAM protocol's setting for the local NSAP.

Default setting following initial installation: *-ftamx25nsap=*

Examples

1. The identification of your own instance is to be set to host.hugo.net:

   ```
   ftmodo -id=host.hugo.net
   ```

2. Only partners from the partner list are to be permitted:

   ```
   ftmodo -dp=f
   ```

3. Flags the local openFT instance as a remote administration server:

   ```
   ftmodo -admcs=y
   ```

4. Only the asynchronous inbound servers for the openFT and FTAM protocols are to be activated.

   ```
   ftmodo -acta=openft,ftam
   ```

5. For directory transfer, the logging of failed transfer requests is to be activated:

   ```
   ftmodo -ltd=f
   ```

6. Examples for the configuration of X.25 lines under Windows:
   - 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
     ```

   - 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
     ```

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

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

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

  ```
  ftmodo -x25=
  ```

7. Examples for the configuration of X.25 lines under Linux:
   - The DTE address 1234 is to be assigned to FarSync X.25 adapter number 0.

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

   - 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
     ```

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

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

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

     ```
     ftmodo -x25=
     ```

3.37 ftmodp

Note on usage

Function: Modify FT profiles

User group: FTAC user and FTAC administrator

Functional description

ftmodp stands for "modify profile".

The FTAC administrator can use this command to change or to privilege FT profiles of other users.

The ADM administrator can use this command to change ADM profiles (i.e. FT profiles which have the property "access to remote administration server", corresponding to ***-ff=c***).

You can use this command to modify your FT profiles. If an FT profile has been privileged, you can use *ftmodp* to remove its privileged status or change the transfer admission.

The timestamp is updated when a profile is modified.

Note for the FTAC administrator

- In the event that the FTAC administrator does not have FT administrator privileges the same time, then admission profiles of other users are blocked after a modification (except after *-priv=y*). This can be by-passed by entering *-ua=user ID,password*. If the user later changes his/her password, the profile will no longer be usable without further modification.
- The FTAC administrator can modify multiple profiles simultaneously specifying *@a* for the profile name and *@a* for the transfer admission in the *-s* option (and *@a* for the user ID if desired).

  Exception: user ID, profile name and transfer admission can be modified only for **one** profile at a time.

Format

ftmodp -h |

<profile name 1..8> | @s | @a
[-s=[<transfer admission> | @a | @n]

[,<user ID> | @a | @adm]]

[-ua=[<user ID>],[<password> | @n]]
[-nn=<profile name 1..8>]
[-tad= | -tad=<transfer admission> | -tad=@n]
[-v=y | -v=n][-d=yyyymmdd | -d=]
[-u=pr | -u=pu][-priv=y | -priv=n]
[-iml=y | -iml=n]
[-iis=y | -iis=n][-iir=y | -iir=n]
[-iip=y | -iip=n][-iif=y | -iif=n]
[-ff= | -ff=[t][m][p][r][a][l] | -ff=c]
[-dir=f | -dir=t | -dir=ft]
[-pn=<partner 1..200>,...,<partner(50) 1..200> | -pn=]
[-pna=<partner 1..200>,...,<partner(50) 1..200>]
[-pnr=<partner 1..200>,...,<partner(50) 1..200>]
[-fn=<file name 1..512> | -fn=] [-fnp=<file name prefix 1..511>]
[-ls= | -ls=@n | -ls=<command1 1..1000>]
[-lsp= | -lsp=<command2 1..999>][-lss= | -lss=command3 1..999>]
[-lf= | -lf=@n | -lf=<command4 1..1000>]
[-lfp= | -lfp=<command5 1..999>][-lfs= | -lfs=<command6 1..999>]
[-wm=o | -wm=n | -wm=e | -wm=one]
[-c= | -c=y | -c=n]
[-cm= | -cm=y | -cm=n]
[-txt=<text 1..100> | -txt=]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

profile name

specifies the name of the FT profile you wish to modify. To see the profile names you have already assigned, you can issue the *ftshwp* command (without options).

@**s** for *profile name*

@s allows you to change the properties of the standard admission profile of the user ID.

The options *-v*, *-d* and *-u* are ignored with a standard admission profile.

@**a** for *profile name*

modifies all FT profiles that come into question at once, unless you select a specific profile with the option *-s*.

> **i** If you specify *ftmodp profile name* without any other parameters, you force the timestamp of the profile to be updated.

-s=[transfer admission | @**n** | @**a**][**,**user ID | @**a** | @**adm**]

is used to specify selection criteria for the FT profile to be modified.

transfer admission

specifies the transfer admission of the FT profile to be modified. You must specify a binary transfer admission in hexadecimal format, see section “Entering commands”.

@**a** for *transfer admission*

modifies either the FT profile specified with *profile name* (see above) or (if no profile name was specified) all the profiles that come into question.

If you specify @*a* as a user, you must specify a login name for *user ID* (not @*a*). Otherwise, an error message is received.

@**n** for *transfer admission*

selects all FT profiles without transfer admission.

transfer admission not specified

causes to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the program prompts you to enter the transfer admission a second time. If you just press <ENTER>, this has the same effect as specifying @*a*.

,user ID

As user, you can only enter your own login name here.

As the FTAC administrator, you can specify any login name here.

@**a** for *user ID*

allows each user to modify only profiles belonging to his or her own login name. If @*a* is specified here, a transfer admission must be specified for *transfer admission* (not @*a*). Otherwise, an error message is received.

If you specify @*a* as the FTAC administrator, you can modify the FT profiles for any login names.

@**adm** for *user ID*

For the FTAC and ADM administrator only.

If you specify @*adm* as the FTAC or ADM administrator, you can modify ADM profiles (corresponding to -*ff*=*c*).
However, you can neither change this property (-*ff*=*c*) nor the user ID (-*ua* option).

user ID not specified

modifies only profiles belonging to the user's own login name, regardless of who issues the command.

-*s* not specified

if @*a* is specified for *profile name*, all the FT profiles belonging to the login name under which the *ftmodp* command is entered are modified. Otherwise, the FT profile with the specified name is modified.

-ua=[user ID]**,**[password | @**n**]

-ua is only meaningful for the FTAC administrator. With *-ua*, the FTAC administrator can assign one FT profile of any login name to another login name, see example 2.

user ID

As user, you can only specify your own login name here.

As the FTAC administrator, you can specify any login name here.

user ID not specified

The user ID is taken from the login authorization under which *ftmodp* is entered.

,password

specifies the password for a login name. A binary password must be specified in hexadecimal format, see section “Entering commands”. The FT profile for the login name is valid only so long as the password *password* is valid for the login name. When the password is changed, the profile can no longer be used (not locked!).

@**n** for *password*

Can only be specified by the FTAC administrator! In this case, the FTAC administrator cannot specify any transfer admission for the FT profile if he/she does not have FT administrator privileges. An existing transfer admission will be automatically deleted in this case.

The password is specified by the owner of the admission profile.

comma only (,) no *password* specified

causes FTAC to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission.

user ID only (without comma and *password*) specified

means that the profile is valid for all passwords of the specified login name *user ID*. When an FT request refers to this admission profile, FTAC uses the password valid at that moment. This prevents you from having to modify the admission profile if the password is changed.

-ua not specified

the login name of this FT profile remains unchanged.

-nn=profile name | **@s**

-nn can be used to assigns a new name to one of your FT profiles.

@**s** for *profile name*

Makes the admission profile the standard admission profile for the user ID. If the admission profile previously had a transfer admission, you must also specify *-tad=@n*.

-nn not specified

leaves the profile name unchanged.

-tad=[transfer admission | @**n**]

allows you as FTAC user to modify the transfer admission of one of your own FT profiles. As the FTAC administrator, you can also modify the transfer admissions for other login names if you have FT administrator privileges.

If the modified admission profile is a standard admission profile (*ftmodp @s* or *-nn=@s*), only *-tad=@n* is permitted.

transfer admission

The transfer admission must be unique within your system so that there are no conflicts with transfer admissions defined by other FTAC users for other access permissions. A binary transfer admission must be specified in hexadecimal format, see section “Entering commands”. If the transfer admission you select has already been assigned, FTAC rejects the *ftmodp* command and issues the message

```
Transfer admission already exists.
```

@**n** for *transfer admission*

disables the old transfer admission.

@n must be specified if you convert an admission profile that has a transfer admission to a standard admission profile using *-nn=@s*.

transfer admission not specified

-tad= causes FTAC to prompt you to enter the transfer admission after the command has been entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. To exclude the possibility of typing errors, the program expects you to enter the transfer admission a second time as an entry check.

The transfer admission is not queried when a standard admission profile is changed. The following message is issued: `Transfer admission of standard`

`profile must be @n.`

-tad not specified

does not modify the transfer admission of the FT profile.

-v=y | **-v=n**

-v defines the status of the transfer admission.

y

the transfer admission is not disabled (it is valid).

n

transfer admission is disabled (it is not valid).

-v is ignored if the modified profile is a standard admission profile.

-v not specified

the transfer admission status remains unchanged.

-d=[yyyymmdd]

-d specifies the period during which the transfer admission can be used. The FT profile is disabled when this period has expired.

You can specify an eight-digit date (e.g. 20170602 for June 2, 2017). The transfer admission can no longer be used after 00:00 hours on the specified day. The largest possible value that can be specified for the date is 20380119 (January 19, 2038).

yyyymmdd not specified

when *-d=* is specified, the previous setting is cancelled, i.e. the time restriction is removed from the transfer admission.

-d is ignored if the modified profile is a standard admission profile.

-d not specified

the previous time restriction defined for the transfer admission remains unchanged.

-u=pr | **-u=pu**

using *-u*, you can control how FTAC reacts when someone attempts to assign an existing transfer admission to an FT profile. Normally, the transfer admission must be disabled immediately, by designating it as private.

Transfer admissions that do not require as much protection, can be designated as public. This means that they are not disabled even when a user attempts to assign another transfer admission of the same name.

Possible values:

pr (default value)

the transfer admission is disabled as soon as someone with another login name attempts to specify a transfer admission of the same name (private).

In this case, the *-u* parameter is set to *no time restriction* at the same time.

pu

the transfer admission is not disabled, even if someone attempts to specify a transfer admission of the same name (public).

-u is ignored if the modified profile is a standard admission profile.

-u not specified

the previous setting remains unchanged.

-priv=y | **-priv=n**

This option is used by the FTAC administrator to grant privileged status to an FT profile.

As a normal FTAC user, you can only withdraw an existing privilege. *y* is not permitted.

y

grants privileged status to the FT profile. The FT administrator's entries in the admission set are ignored for requests executed with a privileged FT profile, i.e., if the user uses the *-iml, -iis, -iir, -iip* or *-iif* options in the FT profile, both the user' s entries (MAX. USER LEVELS) and the administrator' s entries (MAX. ADM LEVELS) are ignored.

n

withdraws the privileged status, if it had been granted, from the FT profile.

-priv not specified

does not modify the privileged status of the FT profile.

-iml=y | **-iml=n**

-iml (ignore max. level) is used to specify whether the FT profile is to be restricted by the values in the admission set. The user can override the entries he/she made himself or herself (the MAX. USER LEVELS) for requests using this FT profile. If the FT profile is also privileged by the FTAC administrator, the entries made by the FTAC administrator (the MAX. ADM LEVELS) can also be ignored. This FT profile would then allow *inbound* basic functions to be used which are disabled in the admission set.

y

allows the values in the admission set to be ignored.

n

restricts the functionality of the profile to the values in the admission set.

-iml not specified

causes the values specified in the profile for the basic functions to apply unchanged.

-iis=y | **-iis=n**

-iis (ignore inbound send) allows the value for the basic function *inbound send* in the admission set to be ignored (see *-iml* for details).

y

allows the basic function *inbound send* to be used even if it is disabled in the admission set. At the same time, component "display file attributes" of the basic function *inbound file management* can be used (see table at *-iif*).

Specifying this option is enough as long as the basic function *inbound send* was disabled by the user, but if it was disabled by the FTAC administrator, it is also necessary that he/she grant privileged status to the FT profile.

n

restricts the profile to the value in the admission set for the basic function *inbound send*.

-iis not specified

causes the values specified in the profile for the basic function *inbound send* to apply unchanged.

-iir=y | **-iir=n**

-iir (ignore inbound receive) allows the value for the basic function *inbound receive* in the admission set to be ignored see *-iml* for details).

y

allows the basic function *inbound receive* to be used even if it is disabled in the admission set. At the same time, subcomponents of the basic function *inbound file management* can also be used (see table at *-iif*).

Specifying this option is enough as long as the basic function *inbound receive* was disabled by the user, but if it was disabled by the FTAC administrator, it is also necessary that he/she grant privileged status to the FT profile.

n

restricts the profile to the value in the admission set for the basic function *inbound receive*.

-iir not specified

causes the values specified in the profile for the basic function *inbound receive* to apply unchanged.

-iip=y | **-iip=n**

-iip (ignore inbound processing) allows the value for the basic function *inbound followup processing* + *preprocessing* + *postprocessing* in the admission set to be ignored
(see *-iml* for details).

y

allows the basic function *inbound follow-up processing* + *preprocessing* + *postprocessing* to be used even if it is disabled in the admission set. Specifying this option is enough as long as the function was disabled by the user, but if it was disabled by the FTAC administrator, it is also necessary that he/she grant privileged status to the FT profile.

n

restricts the profile to the value in the admission set for the basic function *inbound follow-up processing* + *preprocessing* + *postprocessing*.

-iip not specified

causes the values specified in the profile for the basic function *inbound follow-up processing* + *preprocessing* + *postprocessing* to apply unchanged.

-iif=y | **-iif=n**

-iif (ignore inbound file management) allows the values for the basic function *inbound file management* in the admission set to be ignored (see *-iml* for details).

y

allows the basic function *inbound file management* to be used even if it is disabled in the admission set.

Specifying this option is enough as long as the basic function *inbound file management* was disabled by the user, but if it was disabled by the FTAC administrator, it is also necessary that he/she grant privileged status to the FT profile.

n

restricts the profile to the value in the admission set for the basic function *inbound file management*.

The following table shows which subcomponents of the file management can be used under which conditions.

Inbound file management function	**Values of the admission set or extension in profile**
Display file attributes	Inbound Send (IBS) enabled

Modify file attributes	Inbound Receive (IBR) **and** Inbound File Management (IBF) enabled
Rename files	Inbound Receive (IBR) **and** Inbound File Management (IBF) enabled
Delete files	Inbound Receive (IBR) enabled **and** Write mode = overwrite in profile
Display directories	Inbound File Management (IBF) enabled
Create, rename and delete directories	Inbound File Management (IBF) enabled **and** direction = from partner in profile

-iif not specified

causes the values specified in the profile for the basic function *inbound file management* to apply unchanged.

-ff=[**t**][**m**][**p**][**r**][**a**][**l**] | **-ff=c**

-ff defines the FT function for which the FT profile can be used. With the exception of *c*, these letters can be combined in any way (*tm, mt, mr,* ...). *c* must not be combined with other values. Please observe the note concerning the description of *-ff=c*.

t (transfer)

The FT profile can be used for the file transfer functions "Transfer files", "Display file attributes", and "Delete files".

m (modify file attributes)

The FT profile can be used for the file transfer functions "Display file attributes" and "Modify file attributes".

p (processing)

The FT profile can be used for the file transfer functions "File Preprocessing" or "File Postprocessing". The FT function "Transfer files" must also be permitted.

Specification of *p* has no significance for profiles with a file name prefix (*-fnp=*) or a file name (*-fn=*) since, in this case, the first character of the file name or file name prefix decides whether the profile can only be used for preprocessing and postprocessing ("|") or only for file transfer/file management (no "|").

The use of follow-up processing is not controlled by *-ff=*, but by *-lf=* and *-ls=*.

r (read directory)

The FT profile can be used for the file transfer functions "Display directories" and "Display file attributes".

a (administration)

The admission profile is allowed to be used for the "remote administration" function. This means that it authorizes a remote administration server to access the local openFT instance. To do this, the associated transfer admission must be configured in the remote administration server.

-ff=a may only be specified by the FT administrator or FTAC administrator.

l (logging)

The admission profile is allowed to be used for the "Receive ADM traps" function. This allows another openFT instance to send its ADM traps to the remote administration server via this profile. This specification only makes sense if the local openFT instance is flagged as a remote administration server (*ftmodo -admcs=y* command).

-ff=l may only be specified by the FT administrator.

c (client access)

The admission profile is allowed to be used for the "access to remote administration server" function (ADM profile). This allows a remote administrator on a remote computer to use this profile to access the local remote administration server and issue remote administration requests. The local openFT instance must be flagged as a remote administration server (*ftmodo -admcs=y* command).

-ff=c may only be specified by the ADM administrator.

> The value *c* must not be combined with any other value. In addition, an FT profile created with *-ff=c* cannot be changed into a FT profile using the other FT functions (*t, m, p, r, a* or *l*) and vice versa.

No function specified

Specifying *-ff=* allows you to undo any specification with regard to the functions. All file transfer functions are then permitted (corresponds to *tmpr*), but not the remote administration functions (*a, c*) and ADM trap functions (*l*).

-ff not specified

The previous specification with respect to the functions remains unchanged.

-dir=f | **-dir=t** | **-dir=ft**

specifies for which transfer direction(s) the FT profile may be used. Possible values for the direction: *f*, *t*, *ft*, *tf*.

f

allows data transfer only from a partner system to the local system.

t

allows data transfer only from the local system to the remote system. It is thus not possible to create, rename or delete directories.

ft, tf

transfer direction is not restricted in the profile.

-dir not specified

leaves the transfer direction entries in the FT profile unchanged.

-pn=[partner1[**,**partner2, ...]]

You use *-pn* to specify that this admission profile is to be used only for FT requests which are processed by a certain partner system. You can specify the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

You can specify more than one partner system (maximum 50) with a maximum total of 1000 characters.

partner1[,*partner2, ...*] not specified

-pn= cancels a previous restriction defined for partner systems so that the FT profile can be used by every partner system.

-pna=partner1[**,**partner2, ...]

-pna adds one or more partner system(s) to the list of permitted partner systems. Up to 50 partner systems can be entered in the list (max. 1000 characters).

If the list has been empty up to now, then the profile is limited to the specified partner system(s).

-pnr=partner1[**,**partner2, ...]

-pnr deletes one or more partner system(s) from the list of permitted partner systems.

Please note: As soon as you delete the last partner remaining in the list, the profile can be used by every partner system.

-pn, *-pna* and *-pnr* not specified

causes the entries for permitted partner systems to apply unchanged.

-fn=[file name]

-fn specifies which file(s) under your login name may be accessed using this FT profile. If you specify a fully qualified file name, only the file with this name can be transferred.

If the file name ends with %unique or %UNIQUE, this string is replaced by a string which changes for each new call on file transfer or file management requests, see section “Entering commands”.

If *file name* starts with a "|" (pipe character) then it is interpreted as a preprocessing or postprocessing command, see also section “Preprocessing and postprocessing”.

file name not specified

-fn= allows you to cancel a file name entry. This also applies to a prefix assigned with *-fnp*. The FT profile then permits unrestricted access to all files.

-fn not specified

leaves the file name entries in the FT profile unchanged.

-fnp=file name prefix

restricts access to a set of files whose names begin with the same prefix. FTAC adds the character string specified as *file name prefix* to the file name in the request and attempts to transfer the file with the expanded name.

Example:

- Unix systems: If this option is specified as *-fnp=scrooge/* and the request contains the file name *stock*, the file is transferred as *scrooge/stock*.

- Windows systems: If this option is specified as *-fnp=scrooge*\ and the request contains the file name *stock*, the file is transferred as *scrooge**stock*.

In this way, you can designate the files you have released for openFT. If the *-fnp* option was used to specify a prefix, the file name specified in the request must not contain a directory separator (Unix systems: "/", Windows systems: "\").This disables (unintentionally) changing directories specifying ../ or ..\. You should also ensure that there is no chance for a symbolic link to cause a jump to another place in the file tree.

%unique or %UNIQUE cannot be used for a file name prefix. In the case of a file transfer or file management request, the user can use a file name ending with %UNIQUE (or %UNIQUE.*suffix* or %unique or %unique.*suffix*) to generate a unique file name with the prefix specified here.

A file name prefix which starts with the | character indicates that the admission profile can only be used for file transfer with preprocessing and postprocessing, since the file name created using the prefix and the name specified for the ncopy or *ft* command also starts with the | character. In this case, no follow-up commands may be specified.

Exception on Windows systems:The filename prefix under Windows starts with `|cmd /c` or `|&cmd /c`.

file name prefix can be up to 511 bytes in length.

-fn= allows you to cancel a file name prefix entry, see above.

Notes on profiles with preprocessing or postprocessing

- On Unix systems, the shell metacharacters | ; & < > and "newline" may only be specified if they are enclosed in '...' (single quotes) or "..." (double quotes) or if each of them is escaped with "\" (backslash). The character ` (accent grave) and the string $((dollar+open bracket) may only be specified if they are enclosed in '...' (single quotes) or if they are specified directly after a backslash ("\").
- The following strings may not be specified for the name entered in the *ncopy* or *ft* command:
  - .. (two dots)
  - .\ (dot + backslash)
  - .' (dot + single quote, only for Unix systems)

  This makes it impossible to navigate to higher-level directories.
- Special cases
  - You must specify a file name or file name prefix which starts with the string "|ftexecsv " for admission profiles which are to be used exclusively for the *ftexec* command.

  If a command prefix is also to be defined, you must specify it as follows:

  -fnp="|ftexecsv -p= *command prefix* **"**

  (e.g.: `-fnp="|ftexecsv -p=\"ftshwr \""`)

  The same restrictions apply to the command string of the *ftexec* call as to the filename prefix during preprocessing and postprocessing.
  - For admission profiles that are only to be used for getting monitoring data, specify the filename prefix "|*FTMONITOR ". The functions of the profile must permit File Preprocessing (*-ff=tp*). For details, see the *ftcrep* command, Examples.

-fnp not specified

leaves the *file name prefix* entries in the FT profile unchanged.

-ls= | **-ls=@n** | **-ls=**command1

specifies follow-up processing which is to be performed under your login name in the event that file transfer is successful. If *-ls* is specified, no success follow-up processing may be requested in the file transfer request. Specifying *-ls* only makes sense if you also make an entry for *-lf* (see below) to preclude the possibility that an intentionally unsuccessful request can circumvent the *-ls* entry. If you have defined a prefix for the file name with *-fnp* and plan follow-up processing for this file, you must specify the complete file name here.

@n for *command1*

If you enter *-ls=@n*, no follow-up processing is then permitted in the FT profile in the event that file transfer is successful.

command1 not specified

-ls= allows you to cancel a follow-up-processing entry. The FT profile then no longer restricts success follow-up processing in the local system. This is also a way to cancel a prefix for the follow-up processing defined with *-lsp*.

For details on follow-up processing, please refer to section “Commands for follow-up processing”.

-ls not specified

leaves the entries in the FT profile for follow-up processing in the event that file transfer is successful unchanged.

-lsp=[command2]

-lsp defines a prefix for follow-up processing in the local system in the event that file transfer is successful. FTAC then adds the character string *command2* to the follow-up processing specified in the FT request and attempts to execute the resulting command.

Example:

- Unix systems: If this option is specified as *-lsp='lpr '* and the request specifies *file1.txt* as follow-up processing, FTAC executes *lpr file1.txt* as follow-up processing.
- Windows systems: If this option is specified as *-lsp="print "* and the request specifies *file1.txt* as follow-up processing, FTAC executes *print file1.txt* as follow-up processing.

Please also bear in mind the information provided on the *-ls* option!

You can cancel an existing prefix by specifying *-ls=*.

command2 not specified

-lsp= cancels the entry in the FT profile for a follow-up processing prefix after successful file transfer.

For details on follow-up processing, please refer to section “Commands for follow-up processing”.

-lsp not specified

leaves the prefix entries in the FT profiles for follow-up processing in the event that file transfer is successful unchanged.

-lss=[command3]

-lss defines a suffix for follow-up processing in the local system in the event that file transfer is successful. FTAC then appends the character string *command3* to the followup processing specified in the FT request and attempts to execute the resulting command.

Example:

- Unix systems: If this option is specified as *-lss= ' file2.txt '* and the request specifies *lpr* as follow-up processing, FTAC executes *lpr file2.txt* as follow-up processing.
- Windows systems: If this option is specified as *-lss=" file2.txt"* and the request specifies *print* as follow-up processing, FTAC executes *print file2.txt* as follow-up processing.

Please also bear in mind the information provided on the *-ls* option!

For details on follow-up processing, please refer to section “Commands for follow-up processing”.

command3 not specified

-lss= cancels the entry in the FT profile for a follow-up processing suffix after successful file transfer.

-lss not specified

leaves the suffix entries in the FT profiles for follow-up processing in the event that file transfer is successful unchanged.

-lf= | **-lf=@n** | **-lf=**command4

-lf specifies follow-up processing to be executed under your login name if the file transfer is aborted due to an error. If *-lf* is specified, no failure follow-up processing may be requested in the FT request. Making an *-lf* entry only makes sense if you also make an entry for *-ls* (see above) to preclude the possibility that a successful request can circumvent the *-lf* entry. If you have defined a prefix for the file name with *-fnp* and plan follow-up processing for this file, you must specify the complete file name here.

@n for *command4*

-lf=@n is specified, no follow-up processing is then permitted in the FT profile in the event of an unsuccessful file transfer.

command4 not specified (*-lf=*)

-lf= allows you to cancel an entry for follow-up-processing in the event that file transfer is unsuccessful. The FT profile then no longer restricts failure follow-up processing in the local system. This is also a way to cancel a prefix defined with *-lfp*.

For details on follow-up processing, please refer to section “Commands for follow-up processing”.

-lf not specified

leaves the entries in the FT profiles for failure follow-up processing after unsuccessful file transfer unchanged.

-lfp=[command5]

defines a prefix for follow-up processing in the local system in the event that file transfer is unsuccessful. FTAC then adds the character string *command5* to the follow-up processing specified in the FT request and attempts to execute the resulting command.

Example:

- Unix systems: If this option is specified as *-lfp='lpr '* and the request specifies *error.txt* as follow-up processing, FTAC executes *lpr error.txt* as follow-up processing.
- Windows systems: If this option is specified as *-lfp="print "* and the request specifies *error.txt* as follow-up processing, FTAC executes *print error.txt* as followup processing.

Please also bear in mind the information provided on the *-lf* option!

You can cancel an existing prefix by specifying *-lf=*.

For details on follow-up processing, please refer to section “Commands for follow-up processing”.

command5 not specified

-lfp= cancels the follow-up processing prefix in the FT profile in the event of unsuccessful file transfer.

-lfp not specified

leaves the prefix entries in the FT profiles for follow-up processing in the event of unsuccessful file transfer unchanged.

-lfs=[command6]

-lfs defines a suffix for follow-up processing in the local system in the event that file transfer is unsuccessful. FTAC then appends the character string *command6* to the follow-up processing specified in the FT request and attempts to execute the resulting command.

Example:

- Unix systems: If this option is specified as *-lfs=* ' *error.txt* ' and the request specifies *lpr* as follow-up processing, FTAC executes *lpr error.txt* as follow-up processing.
- Windows systems: If this option is specified as *-lfs=" error.txt"* and the request specifies *print* as follow-up processing, FTAC executes *print error.txt* as follow-up processing.

Please also bear in mind the information provided on the *-lf* option!

command6 not specified

-lfs= cancels the follow-up processing suffix in the FT profile in the event of unsuccessful file transfer.

For details on follow-up processing, please refer to section “Commands for follow-up processing”.

-lfs not specified

leaves the suffix entries in the FT profile for a follow-up processing in the event of unsuccessful file transfer unchanged.

-wm=o | **-wm=n** | **-wm=e** | **-wm=one**

-wm specifies which write modes may be used in the file transfer request and what they effect.

o (overwrite)

In the FT request of openFT or FTAM partners, only *-o* or *-e* may be entered for write mode. The receive file is overwritten if it already exists, and is created if it does not yet exist.

With FTP partners, *-n* may also be entered if the file does not yet exist.

n (no overwrite)

In the FT request *-o*, *-n* or *-e* may be entered for write mode. The receive file is created if it does not yet exist. If the receive file already exists, the request is not executed.

e (extend)

In the FT request only *-e* may be entered for write mode, i.e. the receive file is extended by appending the transferred file to the end if the receive file already exists. The receive file is created if it does not yet exist.

one

means that the FT profile does not restrict the write mode.

-wm not specified

leaves the write-mode entries in the FT profile unchanged.

-c= | **-c=y** | **-c=n**

Using *-c*, you can determine whether data encryption is required or forbidden. If the setting in the profile does not correspond to the setting in the request, the request is denied. The setting is not valid for file management requests, since there is no data encryption for these requests.

y

Only requests **with** data encryption may be processed using this profile.

n

Only requests **without** data encryption may be processed using this profile.

neither *y* nor *n* specified

-c= resets the current setting. Requests with and without data encryption are both accepted.

-c not specified

The encryption option remains unchanged.

-cm= | **-cm=y** | **-mc=n**

Using *-cm*, you can determine whether file(s) and/or directory list attributes encryption is required or forbidden. If the setting in the profile does not correspond to the setting in the request, the request is denied.

y

Only requests **with** file(s) and/or directory list attributes encryption may be processed using this profile.

n

Only requests **without** file(s) and/or directory list attributes encryption may be processed using this profile.

neither *y* nor *n* specified

-cm= resets the current setting. Requests with and without file(s) and/or directory list attributes encryption are both accepted.

-cm not specified

The encryption option remains unchanged.

-txt=text | **-txt=**

-txt allows you to enter a new comment in the FT profile (up to 100 characters).

text not specified

-txt= deletes an existing comment.

-txt not specified

an existing comment remains unchanged.

> As soon as you modify an admission profile, the timestamp is also updated. The timestamp is output with *ftshwp -l* (LAST-MODIF). The timestamp is also updated if you do not change the properties of the profile, i.e. if you enter *ftmodp* without any parameters.

> **CAUTION!**If you use the *-ff=p*, *-fn*, *-fnp*, *-ls*, *-lsp*, *-lss*, *-lf*, *-lfp* or *-lfs* options, you must remember
>
> - that a file name restriction can be bypassed by renaming the file unless followup processing is also restricted;
> - that follow-up processing must always be restricted for both successful and unsuccessful file transfer and, if necessary, equivalent restrictions must exist for any permitted preprocessing;
> - that prefixes for the file names and follow-up processing must be matched to one another;
> - that no symbolic links should occur in the part of your file tree that is referenced by the file name prefix;
> - that restrictions applied to preprocessing or follow-up processing can be circumvented if it is possible to replace this command with, for example, a "Trojan horse".

Examples

a. The transfer admission in the *goldmrep* FT profile created in the Examples of the ftcrep command, is to be changed to *forScrooge*. The transfer direction is no longer to be restricted. The profile is to be used to transfer any files with the prefix *mine/* (Unix systems) or *mine*\ (Windows systems). Follow-up processing is to be prohibited entirely.

   The following command has to be entered:

   Unix systems:

   ```
   ftmodp goldmrep -tad=forScrooge -dir=tf\
    -fnp=mine/ -ls=@n -lf=@n
   ```

   Windows systems:

   ```
   ftmodp goldmrep -tad=forScrooge -dir=tf
     -fnp=mine\ -ls=@n -lf=@n
   ```

b. The FTAC administrator has FT administrator privilege and wants to assign the profile *test01* with owner *user1* to the login name *user2*. To do this, there are the following possibilities:

   - `ftmodp test01 -s=@a,user1 -ua=user2`

   The profile can be used immediately.

- `ftmodp test01 -s=@a,user1 -ua=user2,@n`

The profile can be used only after *user2* has activated it via *ftmodp test01 -ua=user2*, for example. Since no password is specified in *-ua* the password currently valid is taken.

3.38 ftmodptn

Note on usage

Function: Modify partner properties

User group: FT administrator

Functional description

You use the *ftmodptn* command to modify the properties of partner systems in the local system's partner list.

Please note that if you modify the partner address, it is no longer possible to convert an openFT partner into an FTP partner or FTAM partner or vice versa.

You can remove an entered dynamic partner from the partner list by setting all the properties to the default values for free dynamic partners by means of the *ftmodptn*
command. The default values are the same as the default values in the *ftaddptn* command with the exception of the security level setting (option *-sl*) which must be set to *-sl=p*.

Similarly, you can add a free dynamic partner to the list by setting at least one of its attributes to a value other than the default. This is possible if *partner* does not reference a partner list entry and *-pa* is not specified.

If a partner name for which there is as yet no partner list entry is specified for *partner* and *-pa* is also specified then a new named entry is created in the partner list. This function is intended for the re-import of exported partner entries. To explicitly create new partner entries, you should use *ftaddptn*.

Format

ftmodptn -h |

<partner 1..200> | @a[-pa=<partner address 1..200>]
[-id=<identification 1..64> | -id=]
[-ri=<routing info 1..8> | -ri=@i | -ri=]
[-ptc=i | -ptc=a | -ptc=]
[-pri=l | -pri=n | -pri=h]
[-sl=1..100 | -sl=p | -sl=]
[-st=a | -st=d | -st=ad]
[-ist=a | -ist=d]
[-am=n | -am=y]
[-rqp=p | -rqp=s]
[-rco=n | -rco=f | -rco=]
[-tr=n | -tr=f | -tr=]
[-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]] (*Linux systems*)
[-sif=<0..3>:<0..3>[,<0..3>:<0..3>]..[,<0..3>:<0..3>]] (*Windows systems*)
[-kl= | -kl=FTOPT | -kl=0 | 768 | 1024 | 2048 | 3072 | 4096]
[-klmin= | -klmin=FTOPT | -klmin=0 | 768 | 1024 | 2048 | 3072 | 4096]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner | @**a**

partner is the name of the partner system in the partner list or the address of the partner system whose properties you want to modify.

@**a** for *partner*

Partner is not a selection criterion, i.e. you modify the properties of all the partner systems present in the partner list. This specification is only possible in combination with the options *-ptc*, *-sl*, *-st*, *-ist*, *-am*, *-rqp* and *-tr*..

Particular care is necessary when using @*a* in combination with *-sl* (security level)!

-pa=partner address

You use *-pa* to enter the address of the partner system in the following form:

[protocol://]host[:[port].[tsel].[ssel].[psel]]

For details concerning address specifications, see section “Specifying partner addresses”.

-pa not specified

The partner address is unchanged.

-id=identification | **-id=**

Identification unique in the network of the openFT instance in the partner system. In the case of FTAM partners, it is possible to specify an Application Entity Title in the form n1.n2.n3.n4..mmm as the identification. n1, n2 etc. are positive integer values which describe the "Application Process Title". n1 can only have the values 0, 1 or 2, n2 is restricted to values between 0 and 39 if n1 does not have the value 2. The optional Application Entity Qualifier mmm must be separated from the values of the Application Process Title by two periods. For details, see the openFT manual "Concepts and Functions".

In the case of FTP partners, *-id* must not be specified!

identification not specified

Specifying *-id=* with no other specification sets the identification to *host* (host name) for partner entries with openFT and FTADM protocol. For FTAM partners, the identification is deleted if *-id=* is entered.

-id not specified

The setting for identification is unchanged.

-ri=routing info | **-ri=@i** | **-ri=**

If the partner system can only be accessed via an intermediate instance then you specify the address information to be used for routing by the intermediate instance in *routing info*.

@i for *routing info*

The instance identification specified in *-id=* is used as the routing information.

neither *@i* nor *routing info* specified

The specification of *-ri=* (without parameters) means that the partner system can be accessed directly, i.e. without an intermediate instance.

-ri not specified

The setting for the routing information is unchanged.

-ptc=i | **-ptc=a** | **-ptc=**

You can use *-ptc* to modify the operating parameter setting for sender verification on a partner-specific basis. These settings only affect partners which are connected via the openFT protocol and do not operate with authentication (e.g. partners with openFT
V8.0 or earlier).

i (identification)

Deactivates checking of the transport address. Only the partner's identification is checked. The partner's transport address is also not checked even if extended sender verification is globally active (see the *ftmodo* command).

a (address)

Activates checking of the transport address. The partner's transport address is checked even if checking of the transport address is globally deactivated (see *ftmodo* command).

If the transport address under which the partner logs on is not the same as the entry in the partner list then the request is rejected.

neither *i* nor *a* specified

-ptc= (without parameters) means that the operating system parameters apply to sender verification.

-ptc not specified

The setting for sender verification is unchanged.

-sl=1..100 | **-sl=p** | **-sl=**

You use this option to assign a security level to the specified partner system or to all the partner systems.

A low security level means that the need for protection towards this partner is low, for instance because the partner's identity has been authenticated using cryptographic methods, which means that you can be certain that the partner is genuinely who they claim to be.

A high security level means that the need for protection towards this partner is high, because the identity of the partner has only been determined on the basis of their address, for instance, and that no authentication has been performed using cryptographic methods.

1..100

Assigns a fixed security level to the partner. 1 is the lowest and 100 the highest security level.

All integers 1 through 100 are permitted.

p

Assigns a security level to the partner depending on the partner's attributes, i.e.:

- Security level 10 if the partner has been authenticated.
- Security level 90 if the partner is known in the transport system and is identified by the name it is known in the transport system.
- Security level 100 if the partner has only been identified by its address.

security level not specified

-sl= (without parameters) means that the operating parameter setting for the security level applies (see command *ftmodo*)

-sl not specified

The setting for the security level is unchanged.

-pri=l | **-pri=n** | **-pri=h**

-pri allows you to specify the priority of a partner in respect of processing requests that have the same request priority. This means that the partner priority only applies in the case of requests that have the same request priority, but that are issued to partners with a different partner priority.

l (low)

The partner is assigned a low priority.

n (normal)

The partner is assigned a normal priority.

h (high)

The partner is assigned a high priority.

-pri not specified

The priority setting remains unchanged.

-st=a | **-st=d** | **-st=ad**

This option allows you to control how locally submitted asynchronous file transfer requests to the specified partner system or systems are processed.

a (active)

Locally submitted asynchronous file transfer requests are processed if the asynchronous openFT server is started.

d (deactivated)

Locally submitted asynchronous file transfer requests are initially not processed but are stored in the request queue.

ad (automatic deactivation)

Multiple consecutive unsuccessful attempts to establish a connection to this partner system result in its deactivation. If you want to perform file transfer again with this system, you must explicitly activate it with *ftmodptn -st=a*.

The maximum number of such unsuccessful attempts is 5. After a connection has been established successfully, the counter is reset to 0.

-st not specified

The processing mode is unchanged.

-ist=a | **-ist=d**

This option allows you to control how file transfer requests issued remotely by the specified partner system or partner systems are processed.

a (active)

File transfer requests issued remotely are processed if the asynchronous openFT server is started.

d (deactivated)

Synchronous file transfer requests issued remotely are rejected. Asynchronous file transfer requests issued remotely by this partner are stored there and cannot be processed until this partner is activated again with *-ist=a*.

-ist not specified

The processing mode is unchanged.

-am=n | **-am=y**

You can use *-am* (authentication mode) to force partner authentication.

n

Authentication is not forced, i.e. this partner is not restricted with regard to authentication.

y

Authentication is forced, i.e. requests are only processed if the local system is successfully able to authenticate the partner.

-am not specified

The authentication mode is unchanged.

-rqp=p | **-rqp=s**

You use this option (rqp = request processing) to control whether asynchronous outbound requests to this partner are always run serially or whether parallel requests are permitted.

p (parallel)

Parallel connections to this partner are permitted.

s (serial)

Parallel connections to this partner are not permitted. If multiple file transfer requests to this partner are pending then they are processed serially. A follow-up request is not started until the preceding request has terminated.

-rqp not specified

The operating mode is unchanged.

-rco=n | **-rco=f** | **-rco=**

With this option (*rco* = recovery outbound) you can switch on and off the restart function for outbound requests. The parameter has no impact if the implementation of the file transfer protocol (FTP) or the type of request (e.g. preprocessing, synchronous orders) does not permit a restart.

n (on)

the restart is always activated for this partner for outbound requests.

f (off)

the restart is deactivated for this partner for outbound requests.

neither *n* nor *f* specified (default value)

-rco= (without parameters) means that the restart operability for outbound requests depends on the setting in the operating parameters, see *ftmodo* command.

-rco not specified

The setting for the restart function remains unchanged.

-tr=n | **-tr=f** | **-tr=**

You can use this option to modify the operating parameter settings for the partner selection for the openFT trace function on a partner-specific basis.

n (on)

The trace function is active for this partner or for all the partners. However, a trace is only written if the openFT trace function has been activated via the operating parameters. In this case, this setting for *ftmodptn* takes priority over the partner selection for the trace function in the operating parameters. See *ftmodo* , *-tr* and *-trp* options.

f (off)

The trace function is deactivated for this partner or for all partners.

neither *n* nor *f* specified

-tr= (without parameters) means that the operating parameter setting for the partner selection in the openFT trace function applies (see the *ftmodo* command).

-tr not specified

The setting for the trace function is unchanged.

The following options are valid for partners with an X.25 transport address.

> **Caution!**
> If, on a Windows system, 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 (Different for Windows and Linux)

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

-sif=

Deletes the setting for the alternative line.

-kl= | **-kl**=FTOPT | **-kl**=0 | 768 | 1024 | 2048 | 3072 | 4096

The parameter can be used to change the length of the RSA key used in encryption. The value of the kl parameter specifies the new RSA key length (RSA-PROPOSED) in bits. The RSA key is only used for the encryption of the AES key agreed between the partners. The configured key length for RSA proposal must be greater than or equal to the specified minimum key length, otherwise a warning will be issued and the proposed key length will be adapted to the minimum key length.

-kl= | **-kl**=FTOPT

Empty string or “FTOPT” option specifies, that key value will be taken from global openFT options displayed via “ftshwo” command. Either both of key values (RSA-PROPOSED and RSA-MINIMUM) need to be set to “FTOPT” or none. Combination of one key having global value and second local partner value (0 … 4096) is not allowed, warning will be issued and keys will be adjusted automatically to “FTOPT value.

-kl=0

-kl=0 explicitly deactivates encryption. If this is set during operation, then any requests with encryption (prior to ftmodo -kl=0) that have been submitted but not yet started are aborted with errors. Any running requests are processed, and their encryption is retained. New requests using encryption are rejected.

-kl=768 | 1024 | 2048 | 3072 | 4096

Standard values for RSA-PROPOSAL encryption. Values from 0 to 4096 take priority over the ones specified in global openFT option visible via ftshwo command.

Default setting following update, export from openFT before version 12.1C70 or not specifying value during creation of partner: -kl=FTOPT. Otherwise default value is "UNCHANGED" meaning that it will stay the same after modification as with most modify commands.

When only RSA-PROPOSAL is specified during addition of partner (without specifying RSA-MINIMUM), then both parameters will be set to global FTOPT values.

-klmin= | **-klmin**=FTOPT | **-klmin**=0 | 768 | 1024 | 2048 | 3072 | 4096

This option specifies the minimum RSA key length.

-klmin= | **-klmin**=FTOPT

Empty string or “FTOPT” option specifies, that key value will be taken from global openFT options displayed via “ftshwo” command. Either both of key values (RSA-PROPOSED and RSA-MINIMUM) need to be set to “FTOPT” or none. Combination of one key having global value and second local partner value (0 … 4096) is not allowed, warning will be issued and keys will be adjusted automatically to “FTOPT value.

-klmin=0

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

-klmin=768 | 1024 | 2048 | 3072 | 4096

Standard values for RSA-MINIMUM encryption. 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. That means: Since an RSA key set is always created on the open platforms during installation, an RSA key is always sent in the protocol during the subsequent data transfer. If this key is deleted and the partner requests encryption, then the partner rejects the connection with s Session Reject (SRJ) "connection not accepted without encryption".

Values from 0 to 4096 take priority over the ones specified in global openFT option visible via ftshwo command.

Default setting following update, export from openFT before version 12.1C70 or not specifying value during creation of partner: -kl=FTOPT. Otherwise default value is "UNCHANGED" meaning that it will stay the same after modification as with most modify commands.

When only RSA-MINIMUML is specified during addition of partner (without specifying RSA-PROPOSAL), then both parameters will be set to global FTOPT values.

Example

X.25 partner on a Windows system:

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

X.25 partner on a Linux system:

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

3.39 ftmodr

Note on usage

Function: Change the property of requests

User group: FT user and FT administrator

Functional description

With the *ftmodr* command, you can change the priority of requests you have issued, or of a group of requests, for example all the requests to a particular partner. Furthermore, you have the option of changing the order of requests within a priority.

As the FT administrator, you can change the priority of all requests in the system.

Format

ftmodr -h |

[-ua=<user ID> | -ua=@a]
[-pn=<partner 1..200>]
[-fn=<file name 1..512>]
[-pr=n | -pr=l][-qp=f | -qp=l]
[<request ID 1..2147483647>]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-ua=user ID | **-ua=@a**

You use *-ua* to specify the user ID for which requests are to be modified. As a user, you can omit this specification since you may only enter your own user ID.

user ID

As FT administrator, you may specify any user ID here.

@a

As FT administrator, you can specify *@a* to modify requests relating to all user IDs.

-ua= not specified

Your own user ID is the selection criterion. Exception: you called the command as FT administrator and also specified a request ID: in this case, the presetting is *@a*.

-pn=partner

You use *-pn* to specify a name or an address for the partner system for which you want to modify requests. The partner should be specified in the same way as in the request or as it is output in the *ftshwr* command without the option *-s*, *-l* or *-csv*. If openFT finds a partner in the partner list that corresponds to the specified partner address then *ftshwr* indicates the name of the partner even if a partner address was specified on request entry.

-fn=file name

You use *-fn* to specify the file name for which requests are to be modified. Requests which access this file in the local system are modified.

You must specify the file name that was used when the request was created. This file name is also output by the *ftshwr* command without the *-fn* option.

Wildcards may not be used in the file name.

-pr=n | **-pr=l**

indicates the new priority. The following values are possible:

n (normal)

the request has the priority "normal".

l (low)

the request has the priority "low".

-qp=f | **-qp=l**

indicates the position of the request within the same priority. The following values are possible:

f (first)

the request is placed at the top of the list of requests with the same priority.

l (last)

the request is placed at the bottom of the list of requests with the same priority.

request ID

request ID is used to specify the identification of a specific request that is to be modified. The request ID is output on the screen when reception of the request is confirmed. It can also be displayed using the *ftshwr* command.

If you have specified a request ID but the other specified selection criteria do not match the request then the request is not modified and the following error message is output:

`ftmodr: Request` *request ID* `not found`

3.40 ftmodsuo

Note on usage

Function: Modifying openFT-Script user options

User group: FT user

Functional description

Functions for the user

Users are able to specify where their openFT-Script requests are to be stored. openFT-Script creates the following subdirectory in the specified working directory:

.openFT/<instance>/script (Unix systems)

.openFT\<instance>\script (Windows systems)

openFT-Script stores openFT-Script requests in it. The user in question then has write permissions for the subdirectory and it cannot be accessed by other users.

You use the *ftmodsuo* command to specify the directory in which the openFT-Script requests are to be stored. However, you can only do this if no openFT-Script is running and there are no current openFT-Script requests for the user. If necessary, you may have to cancel your running openFT-Script requests with *ftcans* and delete terminated openFT-Script requests with *ftdels*. The command is also rejected if another *ftmodsuo* command for the specification of an openFT-Script working directory is currently running under the same user ID.

Functions for the FT administrator

As FT administrator, you can set the following limits for openFT script requests:

- Maximum number of threads which can be used by a user.
- Maximum number of file transfer requests which can be triggered by a user simultaneously.

These settings apply for all users.

Format

ftmodsuo -h |

[-wd=[<directory name 1..128>]]
[-u=@a [-thl=[thread limit 11..10000]]
[-ftl=[file transfer limit 1..500]]]

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-wd

Absolute or relative path name of the working directory in which the subdirectory for the user's openFT-Script requests is to be created.

-wd= resets the working directory to the default value, i.e. the user's home directory.

-u=@a

Only for the FT administrator.

The specified openFT user option should apply for all Ftscript users. This parameter is mandatory when *-thl* and/or *-ftl* is specified, *-u* is not permitted when *-wd* is specified.

-thl=[thread limit]

Only for the FT administrator.

Maximum number of threads which the openFT script simultaneously executes in a Java VM of a user.

If this limit is exceeded during the continuation of an Ftscript run, then the run waits until a sufficient number of threads have been terminated (number of threads <= 3/4 * thread limit + 2). Then the openFT script starts again further threads and the Ftscript run is continued.

Possible values: 11 to 11000.

Default value: 250.

thread limit not specified

If you specify *-thl*= without value then *thread limit* is reset to the default value.

-ftl=[file transfer limit]

Only for the FT administrator.

Maximum number of simultaneous openFT file transfers which the openFT script triggers from a Java VM of a user.

If this limit is exceeded during the continuation of an Ftscript run then the run waits until a sufficient number of file transfer requests have been terminated. Then the openFT script starts again further file transfers and the Ftscript run is continued.

Possible values: 1 to 500.

If the specified value exceeds twice the openFT connection limit (value CONN-LIM in *ftshwo*), the default is then set.

Default value: openFT connection limit *2.

file transfer limit not specified

If you specify *-ftl*= without value then *file transfer timit* is reset to the default value.

ftmodsuo can also be specified without parameters but does nothing.

3.41 ftmonitor

Note on usage

Function: Call the openFT Monitor for displaying measurement data

User group: FT user and FT administrator

Functional description

The *ftmonitor* command calls the openFT Monitor in which the monitoring data collected during openFT operation is displayed. openFT can be running on the local system or on a remote system. The openFT Monitor can only be called if monitoring has been explicitly activated by the administrator on the relevant system (e.g. using the *ftmodo -mon=n* command) and the asynchronous openFT has been started.

Note for Unix systems

Note that you require a graphics-capable terminal to use the *ftmonitor* command.

Format

ftmonitor -h |

[-lay=<monitor layout file name 1..512>]
[-po=<polling interval 1..600>]
[<partner 1..200> [
<transfer admission 8..67> |
<user ID 1..67>[,[<account 1..64>][,[<password 1..64>]]]]]

Description

-h

Outputs the command syntax. Any specifications after *-h* are ignored.

On Windows systems, the command syntax is output in a separate message box.

-lay=monitor layout file name

Name of the Monitor layout file. This file describes what monitoring data is output and how it is presented.

The name of the layout file must be specified with the suffix *.ftmc*. This suffix is automatically assigned by the monitor when the file is saved if it was not explicitly specified there.

The content of the layout file is also generated by the Monitor. You must not change the content of the layout file.

After the default Monitor window has been opened for the first time (without specifying *-lay*), you can create and save your own layout file. To do this, choose a different layout from the *View* menu of the Monitor window, for instance, or set a different value using the selection icon on the top right and store the setting under a name of your choice. Refer to the online Help system of the openFT Monitor window for details.

-lay not specified

If you do not specify *-lay*, the default Monitor window is opened. This contains a chart showing the monitoring value *Networkb/sec of all Requests* (corresponds to the parameter *ThNetbTtl* in the command *ftshwm*).

-po=polling interval

Polling interval in seconds.

Possible values: 1 through 600.

Default value: 1

partner

Name or address of the partner system for which monitoring data is to be shown. The partner must be an openFT partner (i.e. communication via the openFT protocol) and must support the collection of monitoring data, i.e. the openFT version of the partner must be at least V11.

In addition, the partner's asynchronous openFT server must be started and monitoring must be activated in its operating parameters.

partner not specified

If you do not specify a partner, the monitoring data of the openFT instance on the local computer is output.

transfer admission | user ID[**,**[account][**,**[password]]]

Transfer admission for the partner system. File transfer and preprocessing/postprocessing must be permitted under the specified transfer admission.

You can specify this transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system or destination instance. For this purpose, a special admission profile with the filename prefix "|*FTMONITOR " can be set up on the partner system that only permits monitoring data to be collected. You will find an example under *ftcrep* .
- or as a login/LOGON authorization using the syntax of the remote system (*user ID*, where necessary with *account* and/or *password*).

transfer admission not specified

If you do not specify a transfer admission for a remote partner system, the system prompts you for it in a dialog box. The entry made for the password or the FTAC transfer admission remains invisible. Asterisks (*****) are displayed as replacement characters.

Messages from the openFT Monitor

The openFT Monitor issues error messages in the form of a dialog box. It terminates automatically if an error occurs or if monitoring is terminated in the system being monitored.

If the layout of the Monitor window is changed and if openFT is terminated before the changed layout is saved, the openFT Monitor issues a message and queries whether the layout is to be saved.

3.42 ftmsg

Note on usage

Function: Output a message box on a graphical display

User group: FT user

Functional description

The command *ftmsg* allows a message box to be output.

ftmsg can be used to output messages on a graphical display from within local follow-up processing.

Note for Unix systems

The message box is output on the display defined by the DISPLAY variable.

Please note that you require a graphics-capable terminal in order to use the *ftmsg* command.

Note for Windows systems

Under Windows, you have to use the Command Execution Tool.

Format

ftmsg [<window title>:]<message text>

Description

window title

Title of the message box.

Default value for the title is "openFT".

message text

Message text for the message box.

Examples

```
ft file partner!file transadmin -ls="export DISPLAY=$DISPLAY;ftmsg ok"
```

In the case of asynchronous requests, the DISPLAY variable must be set in the environment.

3.43 ftremlic

Note on usage

Function: Remove license key

User group: FT administrator

Functional description

You can use *ftremlic* to remove a license key. You can use *ftshwlic* to output the licenses present on the system.

The following types of license keys exist for the openFT standard product:

SERVER

Basic key for the openFT server function

FTAM

Optional key for the openFT-FTAM function

FTP

Optional key for the openFT-FTP function

Other openFT products have other basic keys and possibly also other optional keys.

If you delete a basic key then any optional license keys that may be present for the openFT product are also deleted. It is then no longer possible to use the functionality of the openFT product.

Format

ftremlic -h |

<license key> | <license type>

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

license key | license type

You can either specify the license key, which consists of 5 groups of 5 characters each, or specify the type of license key (see above). You can display the existing license keys using the *ftshwlic* command.

Messages of the ftremlic command

If the license key or license type is not accepted, a self-explanatory message is output. In this case, the exit code is not equal to 0. Check your entry for typing errors.

If you enter a valid license key which, however, is not present then no message is output.

3.44 ftremptn

Note on usage

Function: Remove a partner from the partner list

User group: FT administrator

Functional description

ftremptn removes a partner from the partner list.

Format

ftremptn [-h] |

<partner 1..200>

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner

Specifies the partner that is to be removed from the partner list. You can specify the name in the partner list or the partner's address. The name and address are displayed using the *ftshwptn* command.

All requests stored for this partner in the request queue are deleted. This is even the case for requests with a status which means that they are known to the partner system. Since this can lead to inconsistencies, you should only remove a partner from the partner list if either there are no more requests for this partner in the request queue or if you can be sure that the partner system will not become active again.

3.45 ftrestore

Note on usage

Function: Restoring the openFT configuration using a previous backup using the *ftbackup* command

User group: FT administrator

> ℹ Only local system administrators (root on UNIX and users in SYSTEM group on Windows) will be able to execute the command, because all instances need to be backed up, which can have separate FT/FTAC /ADM admins.
> The ftrestore command needs to be executed on STD instance by system administrator.

Functional description

Since openFT 12.1C80 it is possible to back up the entire openFT configuration with the *ftbackup* command and then restore it with the *ftrestore* command.

Licenses are not backed up and must be managed manually by users.

Format

ftrestore -h |

<file name 1..512>

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name

Name or full path of the backed up configuration file

Note

On Unix, if multiple instances are going to be restored, firstly user will be prompted with following question for each instance (excluding STD instance):

Instance <instance_name> path: <path_to_instance> already exists, do you agree to overwrite it? (yes/no)

If user answers “yes”, then directory with given path will be removed and replaced with instance directory from backup configuration. In case of “no” answer, directory will be kept and given instance won’t be restored.

On Windows, this feature is not implemented for 12.1C80, so all existing colliding directories will be removed and replaced with instance directories from backup.

If during the restore process it is determined that the backed up user no longer exists, a warning is displayed, the user is skipped, and the restore process continues.

If during the restore process the user who owned an instance no longer exists, a warning is displayed. That instance is not restored and the restore process continues with the next backed up instance.

single-user mode on UNIX systems

It is possible to back up the openFT configuration in single-user mode on UNIX systems.

If the configuration was saved in single-user mode, it must be restored in single-user mode. This is not possible in multi-user mode. Likewise, a configuration saved in multi-user mode cannot be restored in single-user mode.

3.46 ftscript

Note on usage

Function: Starting an openFT-Script request

User group: FT user

Functional description

The *ftscript* command checks the specified script file and executes the statements it includes. The script file must contain a valid XML document which corresponds to the schema for the openFT-Script interface. It must also be possible to read the file using the caller's ID. The maximum number of users who may be owner of openFT-Script requests is 1024. This includes requests that are terminated but not yet deleted.

If errors occur during verification then the script file is not started and the errors are output at stderr.

If the script file starts correctly then the following message is output at stderr:

`ftscript: started successfully. Id:` *ftscript id*

Information about the openFT-Script request is stored in the internal openFT user memory during execution and through to expiry of the retention period. As a consequence, users can view the output *ftscript id* in order to obtain information about the status and success of the operation.

ftscript is restartable, i.e. the processing of the openFT-Script request is ensured even after a system failure.

Format

ftscript -h |

[-t]
<Ftscript file name>

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-t

Diagnostic information (a trace) is created.

Ftscript file name

Name of the script file which contains the XML statements for the openFT-Script request that is to be run.

3.47 ftseti

Note on usage

Function: Set an instance

User group: FT user

Functional description

The *. ftseti* command allows you to select the openFT instance with which you want to work. Using the *ftshwi @a* command displays the names of all instances on your system.

In single-user mode a user may set an instance with the command *ftseti* only in case he is the owner of the instance.

Format

. ftseti -h | <instance 1..8> (Unix systems)

ftseti -h | <instance 1..8> (Windows systems)

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

instance

Name of the instance to be selected.

The command sets the OPENFTINSTANCE environment variable to the instance name. Alternatively, the OPENFTINSTANCE environment variable can also be set manually or in scripts to the desired instance name. On Unix systems, OPENFTINSTANCE can be exported.

Calling ftseti on Unix systems

```
. ftseti
```

Hence, OPENFTINSTANCE is set in the current shell. The *std* instance is set by default.

The first *ftseti* call sets an alias (*ftseti=. ftseti*) in the current shell that allows the preceding period to be dispensed with in subsequent calls.

In some variants of the Bourne shell, the transfer parameters are not forwarded when "." is used in a call.

It may therefore be necessary with a call from a Bourne shell (e.g. under *su*) to switch to the K shell (*ksh*).

Messages of the ftseti command

If *ftseti* could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case.

3.48 ftsetjava

Note on usage

Function: Manage link to the Java executable

User group: FT administrator and FT user (only on Windows systems)

Functional description

ftsetjava is used to set a link to the Java executable.

ftsetjava is used implicitly during installation of openFT. In addition, you can also call *ftsetjava* as administrator in order to

- see what file is referenced by the link to the Java executable used by openFT.
- set the link if Java was not installed or if an incorrect version was installed at the time when openFT was installed or if the installation path of the Java executable has changed.
- see what Java installations are present in the directories searched by openFT.

Format

ftsetjava [@s | @a | <file name 1..512>]

Description

@s

Sets the link to the Java executable.

If the attempt to set a link to the Java executable fails because no suitable Java installation is available, an appropriate message is output to *stdout*. A warning is also issued if this happens during installation of openFT.

The option @s is only supported on Unix systems.

@a

Shows all the Java executables installed in the search path.

Any subsequent call to *ftsetjava @s* is successful if and only if at least one of these installations meets the requirements of openFT with respect to the version. The file whose version is closest to that of the required Java version 1.5 is then used as the source of the link. If multiple Java executables with the same version are installed then the first of these displayed in the list is used.

The option @a is only supported on Unix systems.

file name

Sets the link to the specified Java executable.

You must specify the fully qualified filename of a Java executable that matches the version requirements stipulated by openFT. If the attempt to set a link to the Java executable fails, a message to this effect is issued to standard output.

neither *@s* nor *@a* nor *file name* specified

If *ftsetjava* is called without parameters, it outputs the complete path of the executable used by openFT.

3.49 ftsetmode

Note on usage

Function: Switch on and off single-user mode

User group: FT administrator, ADM administrator, FTAC administrator.

The command is only available on Unix systems.

Functional description

With *ftsetmode* the administrator can switch openFT between multi-user mode and singleuser mode. This is valid for all openFT instances. In single-user mode, openFT completely runs under a specific ID, the so called openFT user ID. However, it is possible to define an individual openFT ID when creating a new instance in single-user mode.

The *ftsetmode* command should not be used during ongoing openFT operation, because if necessary *ftsetmode* ends the openFT processes of all active instances prior to changing to single-user mode (openFT, openFT Explorer, *ftscript* jobs, *ftexec* and *ncopy* commands) respectively prior to changing to multi-user mode (openFT, openFT Explorer, *ftscript* jobs, *ftexec* and *ncopy* commands).

Format

ftsetmode -h |

[-ua=<openFT user ID>]
[-s]
[-m]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-ua=openFT user ID

You use *-ua* to specify the user ID for the single-user mode. openFT completely runs under this ID. Access options, the executing of commands, etc. are consequently restricted to the permissions of this ID, the so called openFT ID.

The standard instance and all other active instances are allocated to the openFT ID.

When switching from multi-user to single-user mode, the following is valid for every active instance:

- FT and FTAC administrators are set to the openFT ID
- If the openFT ID was ADM administrator, it also remains this in single-user mode
- For all other IDs including root, that had been ADM administrator, the ADM administration permission is returned in single-user mode
- The Ftscript user options (including the options of the openFT ID) are deleted. The Ftscript runs of unauthorized IDs can not be accessed by means of openFT resources anymore.

After the switch openFT is restarted for the instances, for which it had been started before the switch.

When booting the system in single-user mode openFT starts under the ID of the instance owner.

-s

Single-user mode. With this option the administrator switches from multi-user to singleuser 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 an ID different from the openFT ID. If required, the openFT ID can then give itself this permission (*ftmoda -admpriv=y*).

-m

Multi-user mode. The administrator switches from single-user to multi-user mode. After invoking this command all active openFT instances work in multi-user mode.

The following applies to 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.
- After the switch openFT is restarted for the instances, for which it had been started before the switch.

Example

Switch to single user mode under the user ID *user01*:

```
ftsetmode -s -ua=user01
```

3.50 ftsetpwd

Note on usage

Function: Store user password

User group: FT administrator

This command is only available on Windows systems.

Functional description

With *ftsetpwd*, you can store the user password of a Windows user ID in openFT. Output is written to standard output. If no user password has been stored for a user, this user cannot use the functions Admission Profiles, Follow-up Processing, Preprocessing/Postprocessing or Asynchronous Requests.

Format

ftsetpwd -h |

[-ua=<user ID 1...36>[,<password 1...64>]]
[-s=<partner 1...15>]
[-c]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-ua=user ID[**,**password]

user ID of the user logged on or of any user whose password is to be stored in openFT or for whom a check is to be performed to determine whether a password is currently stored. In the case of local IDs, you enter the ID without the prefixed host name. In the case of global IDs, you must specify the ID together with the prefixed domain name, e.g. *domain\user1234*

If you specify *-c*, you must not specify a password here.

-s=partner

partner is the name of a different Windows system for which you want to store the user password there. This parameter can be omitted if the user password is to be stored on the local computer.

-c

This parameter allows you to check whether a valid password is stored for a user.

-c must be specified together with *-ua*, and no password may be entered for *-ua*.

Examples

1. You want to store the password *topsecret* for the user ID *Administrator* on the system *Win01*.

   ```
   ftsetpwd -ua=Administrator,topsecret -s=Win01
   ```

2. A check is to be made on the computer *Win02* whether a valid password is stored for the global identifier *dispatch\miller*.

```
ftsetpwd -ua=dispatch\miller -s=Win02 -c
```

3. You want to check if a valid password is stored for the user ID *miller* on the local system.

```
ftsetpwd -ua=miller -c
```

3.51 ftshw

Note on usage

Function: Display the attributes of one or more remote files

User group: FT user

Functional description

With *ftshw* you can display the attributes of a file or files in a directory in the remote system.

There are three options for displaying the attributes:

- List the names of the files in a directory
- Display a default selection of file attributes
- Display all attributes of a file or of files in a directory, as requested from the partner system

A precise description of default output and detailed output can be found in the section “Description of file attribute display”.

Output is written to standard output.

Format

```
ftshw   -h |
        [ -d ]
        [ -c ]
        <partner 1..200>![<file name 1..512>]
        [ <transfer admission 8..67> | @n | @d |
        <user ID 1..67>[,[<account 1..64>][,[<password 1..64>]]] ]
        [ -fnc=t | -fnc=c ]
        [ -sif=n | -sif=l | -sif=m ]
        [ -p=[<management password 1..64>] ]
        [ -s | -l ][ -csv ]
```

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-d

Specifies that the attributes of the files in a remote directory are to be displayed.

-d not specified

The attributes of the file *file name* specified in the command are displayed.

-c

specifies that the file(s) and/or directory list attributes are to be encrypted during transfer. If the partner system doesn’t support encryption, the request is rejected.

-c not specified

The file(s) and/or directory list attributes are not encrypted during transfer.

partner**!**[file name]

specifies the system and the file(s) of which the attributes have to be displayed.

partner

partner is the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

file name

file name can be either absolute or relative to the remote login authorization. If the file name in the remote system has been predefined in an admission profile, it must not be specified here.

If the *-d* option is specified, file name indicates a directory in the remote system.

If the partner system is running openFT (BS2000), elements from PLAM libraries may also be specified here (Syntax: Libname/Element type/Element name).

If openFT (z/OS) is running on the partner system, members from PO libraries can also be output here (syntax: library name/library member).

transfer admission | @**n** | @**d** |
user ID [**,**[account][**,**[password]]]

To enable you to execute file management requests in the remote system, you must furnish the remote system with proof of identity. For this purpose, you will need login authorization in the syntax valid for the remote system. You can specify this transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization in the syntax used by the remote system (user ID, possibly together with account or password).

For details, see section “Entering the authorization data for the partner system”.

@**n** for *transfer admission*

By entering @*n* you specify that the remote system requires no login authorization.

@**d** for *transfer admission*

Specifying @*d* (blanked) causes openFT to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission.

A binary password and binary transfer admission must be specified in hexadecimal format, see section “Entering commands”.

password not specified

Omitting the password necessary for admission causes openFT to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

Nevertheless, you have to specify the commas, e.g.:

```
ftshw partner!file user-id,,
```

or

```
ftshw partner!file user-id,account,
```

neither *transfer admission* nor *user ID* specified

causes the same as *@d*, i.e. openFT queries the transfer admission on the screen after the command is entered. Your (blanked) entry is always interpreted as transfer admission and not as user ID.

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for the remote file names and directory names.

t (transparent, default value)

Specification of the remote file name and directory name in transparent mode (compatible to the previous versions). The file names of the partner system to be displayed are shown as before in older openFT versions. File names in Unix directories are interpreted here on a byte-by-byte basis as ISO8859-1 characters. Only those files that correspond to the ANSI character set in Windows systems are output in Windows systems.

c (character)

Specification of the remote file name and directory name in character mode. I.e., the file and directory name and the file names of the partner system to be displayed are interpreted according to the character code of the remote system, i.e. for Unix partners according to the openFT operating parameter option (*ftmodo -fnccs)* that has been set there.

If the encoding mode on a Unix partner is set to UTF-8 via operating parameter (*ftmodo -fnccs=utf8*), file names that do not correspond to UTF-8 coding are omitted. The file names of the partner system to be displayed are then converted from this character code to the transfer code UTF-8 and then to the output character code, which results on Unix from the setting of the LOCALE and LANG environment variables. File names that cannot be represented in the appropriate output character code are suppressed by default. This can be controlled via the *-sif* option.

-fnc=c is only permitted for partners as of openFT V12.1.

-sif=n | **-sif=l** | **-sif=m** (show improper file names)

Specifies whether non-mappable file names are output (i.e. file names which can not displayed correctly). *-sif* is only permitted together with *-d*, see also examples in section “Output of attributes in directories”.

n (no, default value)

Only mappable file names are output. The output of non-mappable file names is suppressed without feedback.

l (list)

The number of suppressed file names is output in one or more lines at the end of the file list.

m (message)

The number of suppressed file names is output in one or more messages on *stderr*.

The option -sif is ignored for partners with openFT < V12.1 and for FTP partners.

-p=[management password]

If the file in the remote system is protected by a password, you must enter this password here.

A binary password must be entered in hexadecimal format, see section “Entering commands”. This is of relevance for links to openFT (BS2000), because BS2000 supports the definition of hexadecimal passwords.

management password not specified

Specifying *-p=* causes openFT to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

-s

Only the file name or the names of the files in the directory are output (short).

-l

All information available on the remote file in the partner system is requested. However, only attribute values returned by the partner system can be displayed (long).

neither *-s* nor *-l* specified:

A standard scope of information should be displayed.

A precise description of standard output and of detailed output can be found in the following section.

-csv

Specifying *-csv* indicates that the attributes of files on remote systems are to be output in the CSV format. The values in the output are separated by semicolons. If you specify *-csv*, output is always in the long form (analogous to *-l*) regardless of whether you also specify *-l* or *-s*.

-csv not specified

The attributes of files on remote systems are output in the standard format.

3.51.1 Description of file attribute display

The following section describes the output of the commands used to show the attributes of files on the local and remote systems. Both standard output and detailed output are described. The individual fields, their possible values and their meanings are listed.

The standard output is obtained if you do not specify the scope of the output; the detailed output is obtained only with a corresponding specification (see the following examples).

3.51.1.1 Standard output

```
tr-px-acd--- IDENTITY STORAGE-ACCOUNT 1234567890  Apr 30 11:55 FILENAME
|     |          |         |            |           |      |     |
|     |          |         |            |           |      |  file name
|     |          |         |            |          date   /   time
|     |          |         |            |          last modification
|     |          |         |      current file size
|     |          |   account number
|     |   file creator (max. 12 characters)
|  access rights and "permitted actions"
file type
```

Not all information is provided with the FTP protocol. Such missing information is replaced by '-' or by default values.

3.51.1.2 Detailed output, examples

```
ftshw bs2partn!aaa.e42 transbs2 -l
   FILENAME=:6QCA:$HUGO.AAA.E42
   CRE   HUGO DATE=Mar 17 13:01
   MOD   DATE=Mar 17 13:01
   REA   DATE=Mar 17 13:01
   BINARY-FILE
   RECORD-FORMAT=u RECORD-SIZE=32767
   ACCESS-RIGHTS=r-pxeacd---   FILESIZE=32768
ftshw zospart!test.clist transzos -l
   FILENAME=test.clist
   CRE   OPFTWIT
   MOD   DATE=Apr 03 2017
   RECORD-FORMAT=v RECORD-SIZE=648        FILE-AVAILABILITY=i
   ACCESS-RIGHTS=r-pxeacd---   FILESIZE=587860
```

Explanation of fields

file type

specifies the file type. This field can be assigned any of the following values:

t	File contains text
b	File contains binary data
d	Directory
*	No information available on the file structure

The comprehensive output is displayed as follows:

BINARY-FILE	Binary file
DIRECTORY	Directory
CHARACTERSET	Text file

The character set from which the characters in the text file originate is also specified for text files (CHARACTERSET=). The field can be assigned the following values:

g	GraphicString: the file can contain characters from the G0 set of ISO646, or from the G0 set of ISO8859-1 and the G1 set of ISO8859-1.

c	GeneralString: the file can contain characters from the C0 set of ISO646 and either from the G0 set of ISO646 or from ISO8859-1 and from the G1 set of ISO8859-1.
i	IA5String: the file can contain characters from the C0 set and the G0 set of ISO646.
v	VisibleString: the file can contain characters from the G0 set of ISO646.

access rights and permitted actions

contains information on the access rights which can be used for the file or the directory.

For files, this field can be assigned any of the following values:

r	File can be sent.
i	Units of data can be added. [1)]
p	File can be overwritten.
x	File can be extended, i.e., data can be appended to it.
e	Units of data can be deleted from the file.
a	File attributes can be read.
c	File attributes can be modified.
d	File can be deleted.
t	Traversal [1)]
v	Reverse traversal [1)]
r	Random access [1)]

[1)]These values are only relevant for FTAM.

For directories (*-d* is specified), this field can be assigned any of the following values:

r	All files of the directory can be listed.
pxe	Under the directory, files and directories can be created, extended, and deleted.
a	Directory attributes can be read.
c	Directory attributes can be modified.

d	The directory can be deleted.

file creator

identifies the creator of the file. In BS2000, the information refers to the user ID under which the file is created. In the Unix system, this value also identifies the owner of the file.

The field can be up to 12 characters in length.

STORAGE-ACCOUNT

contains the account number used when calculating the cost of storing the file in the remote system.

If the partner returns an account number under FTAM, this is appended to the file owner in the standard output.

FILESIZE - current file size in bytes

contains the current file size in bytes. If the output is followed by a "K", the output is in kilobytes. If it is followed by an "M", the output is in megabytes. This value is only as precise as the value returned by the partner system. Since files are created differently in different systems, different values can be displayed for files of the same size from different systems. Some filestores assign a multiple of a basic unit, e.g. blocks, for file storage. It is therefore advisable not to take this value to be the actual file size; it should be used for guidance only.

date and time of last modification to file contents

contains information on when the file contents were last modified. In the case of modifications made within the last six months, the value is given in the form *month day time* (e.g. Jan 31 15:13); for earlier modifications, the form is *month day year* (e.g. Jan 31 2017).

FILENAME

contains the name of the file.

The following values are part of the comprehensive output:

CRE, MOD, REA, ATM - how the file was last used

contains information on how the file was last accessed. The following types of access are displayed:

CRE	Creating the file
MOD	Modifying the file contents (overwrite, extend)
REA	Reading the file (send), only relevant for FTAM
ATM	Modifying the file attributes, only relevant for FTAM

It is important to remember that it is up to the remote system to determined which information it returns. Therefore, the information line on file use may look different and may contain different information, depending on the partner system. Generally, this section will at least indicate how the file was created.

However, additional information on modifying the file contents or file attributes, or sending a file may not be included. Information on how the file was last used may not be available either.

name of the last file user

identity of the last file user who accessed the file using a particular type of access.

CCS-NAME

Name of the CCS used to encode the file.

RECORD-FORMAT

contains the format of the records transferred. The field can be assigned the following values:

v	Variable length records
f	Fixed length records
u	No defined record length or the record length is hidden in the transmission format, e.g. records are terminated with a CRLF (Carriage Return Line Feed).

RECORD-SIZE

contains the maximum length of the records to be transferred.

FILE-AVAILABILITY

The field can be assigned the following values:

i	File available immediately (immediate).
d	File not available immediately (deferred). The partner is responsible for interpreting the term *deferred*. In the case of openFT partners on BS2000 or z/OS, this means that the file has been migrated.

MAX-FILESIZE

contains the maximum possible file size in bytes (FTAM-specific value). This value is only as precise as the value returned by the partner system. Since files are created differently in different systems, different values can be displayed for files of the same size.

LEGAL-QUALIFICATION

contains a legal qualification for the file (corresponds to a copyright, FTAM specific).

3.51.1.3 Output of attributes in directories

Example for Unix systems

openFT V12.1 is installed on the computer *uxpartn*. Using *-sif=l* you want to check whether there are files with non-mappable file names in the *task* directory.

```
ftshw -d uxpartn!task transunx -fnc=t -sif=l
dr-pxeacd--- user1                          Okt 21 17:10 dtransfer
dr-pxeacd--- user1                          Feb 16 2016  pcmx32
dr-pxeacd--- user1                          Feb 16 2016  pcmx_64
dr-pxeacd--- user1                          Feb 10 2016  unicode
dr-pxeacd--- user1                          Jul 25 12:50 utf16
```

Example for Windows systems

openFT V12.1 is installed on the computer *mt001*. Using *-sif=l* you want to check whether there are files with non-mappable file names in the *Test* directory.

```
ftshw -d mt001!C:\Test smith,,password -fnc=t -sif=l

*r-pxeacd--- mydomain\smith         242   Aug 23 11:39 ftshwk.txt

*r-pxeacd--- mydomain\smith         163   Jan 20 2014   lang.txt

dr-pxeacd--- mydomain\smith              Aug 25 14:48 openFT

*-----------                           0               |*IMPROPER FILE NAMES (D): 2
```

The last line means that there are files with a non-mappable file name which are therefore not output. Instead of attributes, this line contains a string with the following format:

|*IMPROPER FILE NAMES (x): nnn

This string is also output on *stderr* with *-sif=m*.

Explanation:

nnn

is the number of suppressed file names (2 in this example).

(x)

refers to the position that suppresses the file names, possible values:

D

the file access routines in the responder (as in the example). These include file names in Windows, which cannot be presented in the locally set ANSI character set for calls in transparent mode, as well as symbolic links on Unix systems, which refer to a file that does not exist.

R

the encoding of the file names in the responder. These include for example file names on Unix systems for calls in character mode, which do not have a valid UTF-8 byte sequence if UTF-8 is set as the character code for file names

- the decoding of file names in the initiator. These include for example file names for calls from a Unix initiator in character mode, which cannot be mapped to the appropriate character set of the Locale or LANG variables

Up to three of these list entries and messages can be output - one for each of the above "categories".

3.52 ftshwa

Note on usage

Function: Display admission sets

User group: FTAC user and FTAC administrator

Functional description

ftshwa stands for "show admission set", and allows you to examine admission sets.

As a user, you can call *ftshwa* to view your own admission set as well as the standard admission set.

As the FTAC administrator, you can obtain information on all admission sets in your system.

As the FT administrator, you can determine the FTAC administrator and the ADM administrator.

It outputs the following information:

- what limit values the owner of the user ID has set for the individual basic functions,
- what limit values the FTAC administrator has set for the user ID for the individual basic functions,
- whether or not the admission set has the FTAC privilege (i.e. if the owner of the admission set is the FTAC administrator),
- whether or not the admission set has the ADM privilege (i.e. if the owner of the admission set is the ADM administrator).

Format

ftshwa -h |

[<user ID> | @a | @s][-csv]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

user ID | **@a** | **@s**

specifies the user ID for which the admission set is to be displayed.

user ID

You can specify only your own login name here if you are a non-privileged user.

As the FTAC administrator, you can specify any login name desired.

If a login name longer than 8 characters is specified, the first 7 characters are output followed by an asterisk (*).

@**a** for *user ID*

displays information on your admission set and the standard admission set.

When entered by the FTAC administrator, *@a* displays information on the standard admission set and all admission sets that differ from it.

When entered by the FT administrator (who is not the FTAC administrator), *@a* displays information on the own admission set, the standard admission set and the admission set of the FTAC administrator.

@s for *user ID*

returns information only on the standard admission set.

If you specify a non-existent login name, the current standard admission set is displayed for this login name.

user ID not specified

FTAC displays information on the admission set of the login name under which *ftshwa* was entered.

-csv

Specifying *-csv* indicates that the FT admission sets are to be output in the CSV format. The values in the output are separated by semicolons.

-csv not specified

The FT admission sets are output in the standard format.

3.52.1 Output format of ftshwa

Example for outputting all admission sets:

Unix systems:

```
ftshwa @a
               MAX. USER LEVELS              MAX. ADM LEVELS         ATTR
 USER-ID  OBS  OBR  IBS  IBR  IBP  IBF  OBS  OBR  IBS  IBR  IBP  IBF
 *STD     100  100  100  100  100  100  100  100  100  100  100  100
 root      50   50    1    1    1   1   100* 100* 100* 100* 100* 100* PRIV,ADMPR
 smith     90   90    0    0    0   90  100* 100* 100* 100* 100* 100*
```

Windows systems:

```
 ftshwa @a
                  MAX. USER LEVELS               MAX. ADM LEVELS          ATTR

  USER-ID   OBS   OBR   IBS   IBR   IBP   IBF   OBS   OBR   IBS   IBR   IBP   IBF

  *STD      100   100   100   100   100   100   100   100   100   100   100   100

  admin      50    50     1     1     1    1    100* 100* 100* 100* 100* 100* PRIV,
ADMPR

  smith      90    90     0     0     0    90   100* 100* 100* 100* 100* 100*
```

Explanation

USER-ID

The USER-ID column contains the login names to which the respective admission sets belong. If a login name longer than 8 characters is specified, the first 7 characters are output followed by an asterisk (*).

MAX. USER LEVELS / MAX. ADM LEVELS

The six columns under MAX. USER LEVELS show the values specified by each of these FTAC users for their respective admission sets. The six columns under MAX. ADM LEVELS contain the values set by the FTAC administrator.

The lower of the two values determines whether or not the owner of this admission set may use the basic function specified.

The names of the basic functions are abbreviated as follows:

OBS = **O**UT**B**OUND-**S**END

OBR = **O**UT**B**OUND-**R**ECEIVE
IBS = **I**N**B**OUND-**S**END

IBR = **I**N**B**OUND-**R**ECEIVE

IBP = **I**N**B**OUND-**P**ROCESSING
IBF = **I**N**B**OUND-**F**ILE-MANAGEMENT

The values in the admission set have the following meaning:

0	The basic function is disabled.
1..99	The basic function is only released for partner systems with the same or a lower security level. You can use the *ftshwptn* command to display a partner system's security level.
100	The inbound basic function is enabled for all partner systems.

An asterisk '*' after the value indicates that this entry was taken from the standard admission set and will automatically be modified if the value in the standard admission set is changed.

ATTR

This column indicates administrator privileges and is empty for non-privileged users.

PRIV in the ATTR column indicates the privileged admission set, i.e. the FTAC administrator. *root* (Unix systems) or *admin* (Windows systems) is the FTAC administrator in this example.

ADMPR in the ATTR column indicates the ADM administrator. This means that *root* or *admin* is also the administrator of the remote administration server.

3.53 ftshwact

Note on usage

Function: Displaying the activity associated with an openFT-Script request

User group: FT user and FT administrator

Functional description

Outputs information about the individual openFT-Script requests.

Format

ftshwact -h |

[-csv]
[-a=<ID of the activity> | -d=<Level depth 1...> | -c=<Chapter>]
[-st=[W]|[R]|[T]|[F]|[K]|[D]|[C]]
[-u=<user ID>]
<ftscriptid>

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-csv

The information is output in CSV format. If you do not specify *-csv* then the information is output in table format.

-a=ID of the activity

Only the specified activity is displayed.

You may also indicate a specific instruction in a request.

An activity's ID can be determined using a preceding *ftshwact* command (without the *-a* option). This means that you can view the status of the activity later.

-d=Level depth

Depth of the levels to be displayed.

All activities whose *activity ID* is not greater than the specified level number are displayed. The level number is the number of index numbers separated by dots.

Examples: from a request with activity IDs 1, 1.2, 1.2(1).1, 1.2(1).2, 1.2(2).1, 1.2(2).2 and1.3 the option *-d*=2 selects the activities with the activity IDs 1, 1.2 and 1.3.

-c=Chapter

Chapter corresponding to the activities to be displayed.

Those activities are output that are a level below the activity with the activity ID specified as the chapter.

In the above example, these are -c=1: 1.2 and 1.3; for -c=1.2: 1.2(1).1, 1.2(1).2, 1.2(2).1 and 1.2(2).2.

-st=[**W**][**R**][**T**][**F**][**K**][**D**][**C**]

Display activities with the specified status. You can specify multiple statuses one after the other, e.g. *-st=WRT*.

Activity 1 is always output since it displays the execution status of the entire script.

-u=user ID

User ID under which the specified request is searched for.

Only FT administrators may input a user ID.

The default value is the calling party's user ID.

ftscriptid

Identification of the openFT-Script request. This is output if the openFT-Script request is started via an *ftscript* command.

You must specify precisely one openFT-Script request. Wildcard syntax is not supported.

3.53.1 Description of the output

Output is possible in tabular form and in CSV format.

It should be noted that for activities which have not yet been started, the output from the *ftshwact* command is usually incomplete since the references present in the request have not yet been resolved and it is not therefore possible to enter all the desired output values. In particular, file and directory names in reference specifications are not fixed until runtime since they may be dependent on the operating system.

Output in table format

The processing level of the activities is displayed in four columns:

Id

Unique identification of the activity within the request. This can be converted into an Xpath which mirrors the position of the activity in the tree which is statically predefined by the XML script.

Dynamic information is simply added for the *foreach* nodes (sequence number in the *foreach* loop).

For more detailed information, see the description of the XML statements for the openFT-Script interface.

Sta

Status of the statement. The following status identifiers are possible:

W (waiting)	The activity has not yet been started.
R (running)	The activity has been started but has not yet been terminated.
T (terminated)	The activity has been terminated without errors.
F (failure)	The activity has been terminated with an error.
K (killed)	The activity was cancelled by means of a faulthandler or an *ftcans* command.
D (dead)	The activity no longer starts due to a previous error.

In the case of the *ftscript* activity (first activity in an openFT-Script request), a distinction is made between the following statuses:

I (interrupted)	The request was interrupted, e.g. due to a system crash.
C (cancelled)	The request was cancelled with *ftcans*.
X (cancelling)	The request is currently being cancelled due to an *ftcans* command.
F (failure)	Is only displayed for the *ftscript* activity if the error was not handled by a *faulthandler*.

In the case of activities with the status F and *faulthandler* activities, the cause of the error is output in clear text in an additional line.

Activity

Activity name. The names are based on the openFT-Script language but may be truncated in some cases, e.g. *faulthdlr* instead of *faulthandler*.

foreach is designated in accordance with the value of the execute attribute as *foreach_P* (parallel) or *foreach_S* (sequential).

TransferFile is designated as *sendFile* or *rcvFile* (=receive File) depending on the direction of transfer.

ActivityObject

The content of this column depends on the activity in question, see the table below.

Activity	**ActivityObject**	**Meaning**
ftscript	<scriptPath>	Complete path name of the original file with the XML statements.
empty	-	
foreach_P	<contextObject>	context object which assumes the value of the current list element
foreach_S	as foreach_P	
parallel	-	
sequence	-	
sendFile	Specifies the remote file in the following form:	
	<partner>!<file name>	Partner with file name if both are known.
	*unknown!<file name>	if the partner is not yet known.
	*unknown!*unknown	if both are not yet known.
	<partner>!*ref(<contextId>)	if *contextId = foreach contextObject* and the resolution is not yet known because it has not yet been passed through.

	<file name>	in the case of requests which have already been started, this is the name specified in the FT request. In the case of requests which have not yet been started, this name is derived from the operating system-specific name specified in the XML file (e.g. unixName) and extended by the directory specifications.
rcvFile	as sendFile	
deleteFile	specifies the remote file as in sendFile (with partner), if the file is local, without partner:	
	<file name>	like *sendFile*, is determined from the FT request in the case of requests that have already been started, and from the XML file in the case of requests that have not yet started. A local file name would be output as an absolute file name in the case of a started request and as a relative path name in the case of an as yet unstarted request.
	*unknown!<file name>	if it is not known if the file is local when a file object is referenced.
createDir deleteDir	<partner>!<directory-name>	Partner with directory name if both are known.
	*unknown!<directory-name>	if the partner is not yet known.
	*unknown!*unknown	if both are not yet known.
	<partner>!*ref(<contextID>)	if *contexId* = *foreach contextObject* and the resolution is not yet known because it has not yet been passed through.
	<directory-name>	if the directory is local. In this case, as for *sendFile*, the name for already started requests is determined from the FT request and for requests which have not yet been started, from the specifications in the XML file. A local file name would be output as an absolute file name in the case of a started request and as a relative path name in the case of an as yet unstarted request.
	as createDir.	as createDir.

listDir	as createDir.	as createDir.
execScript	32 characters	Contains the first 32 characters of the command that is to be executed. For security reasons, the user should make sure that the first 32 characters do not contain any confidential parameters.
fault	<faultcode>	Error code specified by the user.
faulthdl	<triggering activity id>: <special faultcode>;<general faultcode>	

3.54 ftshwatp

Note on usage

Function: Display ADM traps

User group: FT administrator, ADM administrator, and users configured as remote administrators on the remote administration server.

Functional description

If you are the FT administrator of the ADM trap server, *ftshwatp* allows you to obtain information on the ADM traps sent to the ADM trap server and stored in the ADM trap log file there.

If the ADM trap server is also used as remote administration server, both the ADM administrator and the remote administrators can view traps.

- If you are the ADM administrator of the remote administration server, you can view all ADM traps.
- If you are a remote administrator, you can view "your" ADM traps (locally or with *ftadm*). This means that you only see the ADM traps of those openFT instances for which you have at least FTOP permission. See the *ftshwc* command.

The ADM traps are identified by trap IDs. The trap IDs are assigned in ascending sequence. For technical reasons, the numbering sequence is not always unbroken. If no other specifications are made, openFT always outputs the most recent ADM trap. When requested, openFT outputs all the ADM traps up to the number specified in the command.

The ADM traps are stored in the ADM trap log file. The maximum number of stored ADM traps depends on the maximum possible size of the ADM trap log file. If the maximum number of ADM traps is exceeded, the records with the lowest trap ID are overwritten by the current records.

You can choose between three output formats, short output format, detailed output format and CSV output format (**C**haracter **S**eparated **V**alue).

The ADM traps are output to standard output.

Format

ftshwatp -h |

[-rg=[[[[yyyy]mm]dd]hhmm |

#1..999999999999999999][- [[[[yyyy]mm]dd]hhmm |
[#1..999999999999999999]]

[-src=<partner 1..200>]
[-tt=[fts][,][pts][,][ptu][,][rqc][,][rqf][,][rqs]][-nb=1.. 9999999 | -nb=@a]
[-l | -csv]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-rg=[[[[yyyy]mm]dd]hhmm][-[[[[yyyy]mm]dd]hhmm]

With *-rg*, you can optionally specify the start or end of a time period.

[[[yyyy]mm]dd]hhmm

If you specify a time as 4 digits, this is interpreted as hours and minutes. 6 digits are interpreted as day (date) and time in hours and minutes, 8 digits as month, day and time in hours and minutes and 12 digits as year, month, day and time in hours and minutes. The largest possible value that can be entered for the date is 20380119 (19th January 2038).

openFT then outputs the ADM traps that lie between the specified limits.

-rg=[[[yyyy]mm]dd]hhmm

The ADM traps that occurred at the specified time are output.

-rg=[[[yyyy]mm]dd]hhmm-[[[yyyy]mm]dd]hhmm

The time period begins with the start time and ends with the second time specified.

If a number is specified with *-nb* that is smaller than the number of ADM traps in the period, the required number of ADM traps up to the end time is output.

-rg=[[[yyyy]mm]dd]hhmm-

The time period begins at the start time and ends with the most recent ADM trap entry.

If a number is specified with *-nb* that is smaller than the number of ADM traps in the period, the most recent ADM traps are output.

-rg=-[[[yyyy]mm]dd]hhmm

The time period ends at the specified time.

If a number is specified with *-nb* that is smaller than the number of ADM traps in the period, the required number of ADM traps up to the end time is output.

-rg=[#1..9999999999999999][-[#1..9999999999999999]]

With *-rg*, you can optionally specify the start or end of a trap ID range.

#1..999999999999999999

Selection of a trap ID is indicated by the leading # sign. openFT outputs those ADM traps that lie within the specified range.

-rg=#1..999999999999999999

The ADM trap with exactly this trap ID is output. If this ID does not exist (gaps in the numbering are possible), the following message is output: `No ADM traps available for the selected criteria.`

-rg=#1..99999999999999999-#1..999999999999999999

The range starts with the ADM trap with the first specified trap ID and ends with the second specified trap ID.

If a number is specified with *-nb* that is smaller than the number of ADM traps in the range, the required number of records up to the end ID is output.

-rg=#1..999999999999999999-

The range starts with the ADM trap for the specified trap ID and ends with the most recent ADM trap.

If a number is specified with *-nb* that is smaller than the number of ADM traps in the period, the most recent ADM traps are output.

-rg=-#1..999999999999999999

The range ends with the ADM trap with the specified trap ID.

If a number is specified with *-nb* that is smaller than the number of ADM traps in the range, the required number of ADM traps up to the end ID is output.

-rg not specified

The trap ID range or the time period is not used as a selection criterion, in other words, output starts with the current (most recent) ADM trap.

-src=partner

-src allows you to specify that only those ADM traps are to be displayed that originate from a specific partner. You can specify the name from the partner list or specify the partner address.

-src not specified

The partner name is not used as a selection criterion.

-tt=[**fts**][**,**][**pts**][**,**][**ptu**][**,**][**rqc**][**,**][**rqf**][**,**][**rqs**]

-tt allows you to specify the type of ADM traps to be output. You can specify several values separated by commas:

fts

All ADM traps are output that indicate that the asynchronous openFT has started (*FT-START) or stopped (*FT-STOP).

pts

All ADM traps are output that indicate a status change of a partner system (*PART-STATE).

ptu

All ADM traps are output that indicate that a partner system may not be reachable (*PART-UNREA).

rqs

All ADM traps are output that indicate that the amount of requests in the request queue has reached a limit of at least 85% (*RQ-LIM-HIGH) or has fallen below a value of 80% (*RQ-LIM-LOW).

rqf

All ADM traps are output that indicate failed transfer (*TRANS-FAIL).

rqc

All ADM traps are output that indicate successful transfer (*TRANS-SUCC).

-tt not specified

The ADM trap type is not used as a selection criterion.

-nb=1.. 9999999 | @**a**

-nb allows you to specify the number of ADM traps to be output.

@**a** for *number*

-nb=@*a* outputs all ADM traps that meet the specified selection criteria.

-nb not specified

If *-nb* is not specified, the output will depend on whether *-rg* has also been specified or not:

- If *-rg* is specified, all ADM traps that meet the specified selection criteria are output (corresponds to *-nb*=@*a*).
- If *-rg* is not specified, then only one ADM trap is output (corresponds to *-nb=1*).

-l

-l specifies that the ADM traps are to be output in detailed format.

-csv

-csv specifies that the ADM traps are to be output in CSV format. The values in the output are separated by semicolons.

-csv must not be specified at the same time as *-l*.

Neither *-l* nor *-csv* specified

The ADM traps are output in the default short format.

3.54.1 Description of the output of ADM traps

When you output ADM traps using the *ftshwatp* command, you can select between a short, concise output format, a long, detailed output and finally, output in CSV format for further processing in external programs.

The ADM traps are identified by trap IDs. These IDs are assigned in ascending sequence. For technical reasons, the numbering sequence may contain gaps. The sequence of entries in the ADM trap log file does not always correspond to the temporal sequence in which the ADM traps occurred on the system concerned. Searching for records according to particular selection criteria can therefore take a long time, because it is in principle necessary to read in all the entries.

3.54.1.1 Short output format of an ADM trap

The last three ADM traps are output in this example:

```
$ftshwatp -nb=3
        TRAP-ID TYPE         DATE        TIME      SOURCE
             52 RQ-LIM-HIGH  2016-12-02  10:36:56  fileserv
             51 TRANS-FAIL   2016-12-02  10:36:48  FTSERV01
             50 PART-UNREA   2016-12-02  10:32:01  FTSERV01
```

Explanation

TRAP-ID

Number of the ADM trap in the ADM trap log file, up to 18 digits.

TYPE

Trap type, possible values:

FT-START

Asynchronous openFT has started

FT-STOP

Asynchronous openFT has stopped

PART-STATE

Status change on a partner system

PART-UNREA

Partner system possibly not reachable

RQ-LIM-HIGH

Request queue has reached a filling level of at least 85%

RQ-LIM-LOW

Request queue has fallen below a filling level of 80%

TRANS-SUCC

Successful file transfer

TRANS-FAIL

Failed file transfer

DATE

Date on which the trap occurred.

TIME

Time at which the trap occurred.

SOURCE

Name of the partner on which the trap occurred.

3.54.1.2 Long output format of an ADM trap

Example for outputting the last two ADM traps in detailed format:

```
$ftshwatp -nb=2 -l

TRAP-ID = 52 TYPE = RQ-LIM-HIGH TIME = 2016-12-02 10:36:56

   SOURCE    = fileserv
   PARTNER   =                        PTN-STATE =
   TRANS-ID  =    RC   =              INITIATOR =
   FILENAME  =
   ERROR-MSG =
TRAP-ID      = 51 TYPE = TRANS-FAIL   TIME = 2016-12-02 10:36:48
   SOURCE    = FTSERV01
   PARTNER   = PARTLINU               PTN-STATE =
   TRANS-ID  = 11 RC   = 2169         INITIATOR = user
   FILENAME  = order.txt
   ERROR-MSG = Request 11. Remote System: Transfer admission invalid
```

Explanation

TRAP-ID

Number of the ADM trap in the ADM trap log file, up to 18 digits.

TYPE

Trap type.

The possible values are the same as for the short output format, see the description in section “Short output format of an ADM trap”.

TIME

Date and time at which the trap occurred.

SOURCE

Name of the partner on which the trap occurred.

TRANS-ID

Transfer ID of the transfer that triggered the trap.

RC

Reason code of the transfer that triggered the trap.

INITIATOR

User ID or location of the transfer that triggered the trap.

PARTNER

Partner name of the transfer or partner that triggered the trap.

PTN-STATE

Partner state of the partner that triggered the trap.

FILENAME

Filename of the transfer that triggered the trap.

ERROR-MSG

Message text of the transfer that triggered the trap.

3.55 ftshwc

Note on usage

Function: Show openFT instances that can be remotely administered

User group: Users configured as remote administrators on the remote administration server.

Functional description

ftshwc allows you to show the openFT instances that you are permitted to administer as remote administrator.

You can enter *ftshwc* both locally on the remote administration server and by remote administration using *ftadm* :

- If you enter *ftshwc* locally on the remote administration server, the openFT instances are determined on the basis of the user ID under which you issue the *ftshwc* command.
- If you enter *ftshwc* via a remote administration request using *ftadm*, you must specify an FTAC transfer admission. The openFT instances are determined on the basis of the admission profile that belongs to this transfer admission.

ftshwc searches the configuration data on the remote administration server for openFT instances that are allowed to be remotely administered with the user ID or using this admission profile and outputs them.

If you are not permitted to remotely administer any instances, the following message is issued:

```
ftshwc: No instances available
```

Format

ftshwc -h |

[-rt=i | -rt=gi | -rt=ig]
[-csv]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-rt=i | **-rt=gi** | **-rt=ig**

-rt specifies what information is to be displayed.

You can specify the following: *i*, *gi* (default), *ig*

i

Only information on instances is shown.

gi, ig

Information on groups and instances is shown.

-csv

-csv specifies that the data is to be output in CSV format.

-csv not specified

The data is output in default format.

3.55.1 Output format of ftshwc

Example of output in default format:

```
ftshwc
TYPE    = *GROUP        ACCESS =                MODE =
   NAME = Hamburg
   DESC = Computing center north in Hamburg Wandsbek
TYPE    = *GROUP        ACCESS =                MODE =
   NAME = Hamburg/HH1
   DESC = QA computing center
TYPE    = *INSTANCE     ACCESS = FT+FTOP+FTAC   MODE = FTADM / *CHAR
   NAME = Hamburg/HH1/HHWSRV01
   DESC = Solaris 10
TYPE    = *INSTANCE     ACCESS = FT+FTOP+FTAC   MODE = FTADM
   NAME = Hamburg/HH1/HHWSRV02
   DESC = HP-11
TYPE    = *INSTANCE     ACCESS = FT+FTOP        MODE = LEGACY / *TRANSPARENT
   NAME = Hamburg/HH1/HHWSRV11
   DESC = Solaris 9
TYPE    = *GROUP        ACCESS =                MODE =
   NAME = Hamburg/HH2
   DESC = Human resources department
TYPE    = *INSTANCE     ACCESS = FTOP           MODE = FTADM
   NAME = Hamburg/HH2/HHWSRV99
   DESC = Mainframe system (BS2000)
```

Explanation

TYPE

Specifies whether the item is a group or an openFT instance:

*GROUP

Group

*INSTANCE

openFT instance

ACCESS

Only contains a value if *TYPE=*INSTANCE* and specifies what remote administration privileges the remote administrator has on this instance:

FTOP

Read FT access only (FT operator)

FT

Read and modify FT access. Corresponds to the permissions of an FT administrator.

FTAC

Read and modify FTAC access. Corresponds to the permissions of an FTAC administrator.

MODE

Only contains a value if *TYPE=*INSTANCE* and specifies the protocol that is used to administer this instance and whether there is configured a recommendation for the encoding mode:

FTADM

The instance is administered using the FTADM protocol.

LEGACY

The instance is administered using *ftexec*.

The recommended encoding mode is output as a supplement to the protocol:

/ *CHAR

The character mode is recommended with *ftadm* commands for this instance.

/ *TRANSPARENT

The transparent mode is recommended with *ftadm* commands for this instance.

In case of groups no encoding mode is output.

NAME

Pathname of the group or of the openFT instance.

In remote administration requests, you must always specify the name of the openFT instance as it is displayed here, i.e. as a complete pathname.

DESC

Description of the group or openFT instance.

3.56 ftshwd

Note on usage

Function: Display diagnostic information

User group: FT administrator

Functional description

With the *ftshwd* command, you can display diagnostic information.

The diagnostic documents are used by the Maintenance and Diagnostic Service for error diagnosis.

Format

ftshwd

Description

The command has a number of options, but these are only significant for the Customer Service team.

Example

```
ftshwd
DATE     TIME    SSID  COMPONENT    LOCATION-ID       INFO
20160617 100921  FT    251/yfysequ  46/SwinsLwrite    ffffffff
20160617 100923  FTAC  39/yfslogg   1/WriteErr        ffffffff
```

Explanation of output:

DATE

Date when the error occurred

TIME

Time at which the error occurred

SSID

Subsystem ID. Name of the subsystem that generated the diagnostic record.

COMPONENT

Module number/name

LOCATION-ID

Location in the code at which the error occurred.

INFO

Error code

3.57 ftshwe

Note on usage

Function: Display FT profiles and admission sets from a file

User group: FTAC administrator

Functional description

ftshwe stands for "show environment", i.e. display FT profiles and admission sets from a file. Using *ftshwe*, the FTAC administrator can display FT profiles and admission sets that were saved using the *ftexpe* command.

Format

ftshwe -h |

<file name>
[-u=<user ID>[,...,<user ID(100)>]]
[-pr=<profile name 1..8>[,...,<profile name(100) 1..8>] | -pr=@n]
[-as=y | -as=n]
[-l][-csv]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name

file name specifies the file from which the FT profiles and admission sets are to be displayed.

-u=user ID1[**,**user ID2][**,**user ID3]..

specifies the user IDs whose FT profiles and admission sets are to be displayed. You can specify up to 100 login names simultaneously.

If the specified user ID has no admission sets, only the standard admission set is displayed.

If you specify a non-existent login name for *user ID1*, the current standard admission set is displayed.

-u not specified

all FT profiles and admission sets are displayed.

-pr=profile name1[**,**profile name2][**,**profile name3]... | **-pr=@n**

specifies the FT profiles to be displayed (up to 100).

@**n** for *profile name*

no FT profiles are displayed.

-pr not specified

all FT profiles belonging to the user IDs specified in the -u parameter are displayed.

-as=y | **-as=n**

specifies whether or not admission sets are to be displayed.

y (default value)

all admission sets belonging to the login names specified in the *-u* parameter are displayed.

n

no admission sets are displayed.

-l

specifies that you wish to see the contents of the selected FT profiles.

-l not specified

displays only the names of the FT profiles. Markings also indicate whether or not an FT profile is privileged (*) and whether or not it is disabled (!).

-csv

-csv specifies that the FT profiles and admission sets are to be output in CSV format. The values are output separated by semicolons. When *-csv* is specified, the output is always detailed (analogous to -l), regardless of whether or not -l is specified at the same time.

For details, see section “ftshwp” and section “ftshwa”.

-csv not specified

The FT profiles and admission sets are output in the standard format.

3.58 ftshwf

Note on usage

Function: Display the attributes of a local file

User group: FT user

Functional description

The command is above all useful in connection with FTAM partners. For openFT partners, information about *binary-fixed* file can be displayed.

With *ftshwf*, you can display the FTAM attributes of a file in the local system. Thus, you can define the file attribute values for file transfer and file management requests involving FTAM partners.

There are three options for outputting the attributes:

- Display the file name
- Display a default selection of file attributes
- Display all attributes of the file

Output is written to standard output.

A precise description of standard output and detailed output can be found in the section “Description of file attribute display”.

Format

ftshwf -h |

<file name 1..512>
[-s | -l][-csv]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

file name

Indicates the file whose attributes are to be displayed. Some of the attributes displayed only apply for FTAM partners who wish to transfer files with openFT-FTAM.

-s

Only the file name is output (short).

-l

All information available on the file in the partner system is output.

neither *-s* nor *-l* specified:

The standard information is displayed. The amount of information and the layout of the output are described in the section “Description of file attribute display”.

-csv

You use *-csv* to specify that the file attributes are to be output in CSV format. The values are output separated by semicolons. If *-csv* is specified then output is always complete (in the same way as for *-l*) irrespectively of whether *-l* is specified simultaneously or not.

Examples

1. You wish to output the standard scope of information on the *locfile* file on the local system.

   `/home/john/locfile (Unix systems)`

   `ftshwf locfile`

   Output on Unix systems:

   ```
   *ripxeacd--- john 214 Apr 30 11:55
   ```

   Output on Windows systems:

   `C:\john\locfile (Windows systems)`

   ```
   *ripxeacd--- john 214 Apr 30 11:55
   ```

2. You wish to output detailed information on the FTAM attributes of the *locfile* file on the local system.

   `ftshwf locfile -l`

   Output on Unix systems:

   ```
   FILENAME=/home/john/locfile
   CRE   otto
   MOD   DATE=Apr 28 15:54
   REA   DATE=Apr 30 09:01
   ATM   DATE=Apr 28 15:54
   FILE-AVAILABILITY=i
   ACCESS-RIGHTS=ripxeacd---     FILESIZE=214
   ```

   Output on Windows systems:

   ```
   FILENAME=C:\john\locfile
   CRE otto DATE=Apr 28 15:54
   MOD DATE=Apr 28 15:54
   REA DATE=Apr 30 09:01
   FILE-AVAILABILITY=i
   ACCESS-RIGHTS=ripxeacd---      FILESIZE=214
   ```

3. Example of a file with the attribute *binary fixed* that is evaluated for openFT partners, see the command *ftmodf* :

   `ftshwf binfix.06 -l`

   Output on Unix systems:

```
FILENAME=/home/special/binfix.06
CRE   special
MOD   DATE=Nov 28 15:54
REA   DATE=Dez 05 10:01
ATM   DATE=Dez 05 15:54
BINARY-FILE  RECORD-FORMAT=f  RECORD-SIZE=14156
FILE-AVAILABILITY=i
ACCESS-RIGHTS=ripxeacd---     FILESIZE=42468
```

Output on Windows systems:

```
FILENAME=C:\special\binfix.06
CRE    special DATE=NOV 26 16:31
MOD    DATE=Nov 28 15:54
REA    DATE=Dez 05 10:01
BINARY-FILE   RECORD-FORMAT=f   RECORD-SIZE=14156
FILE-AVAILABILITY=i
ACCESS-RIGHTS=ripxeacd---      FILESIZE=42468
```

3.59 ftshwi

Note on usage

Function: Display information on instances

User group: FT user

Functional description

The *ftshwi* command allows you to display information on the openFT instances.

Format

ftshwi -h | [-l | -d] [<instance 1..8> | @a]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-l

(long) Detailed information is output, consisting of the instance name, the host name and the instance directory. In single-user mode additionally the owner of the instance is output.

-d

Displays only the instance directory.

If neither *-l* nor *-d* are set, only the instance name is displayed.

instance | @**a**

Name of the instance on which you want information to be displayed.Instance names have a maximum length of 8 characters and must consist of alphanumeric characters. The first character must not be a number.

@**a** for *instance*

Information on all instances is output. If neither an instance name nor @*a* is specified, information is displayed on the instance that is currently set.

Examples

1. You enter *ftshwi* immediately after installation:

   `ftshwi -l @a`

   Output on Unix systems:

```
Instance Address           Directory
-------  -------           -----------
std      -                 /var/openFT/std
```

   Output on Windows systems:

```
Instance Address            Directory
-------   -------            -----------
std
```

- ```
  C:\ProgramData\Fujitsu
  Technology
  ```

```
Solutions\openFT\var\std
```

The output "-" under *Address* means that the standard instance logs into all addresses of the system and only accepts inbound connections for all the addresses.

2. You enter *ftshwi* after the FT administrator has assigned the standard instance the address MAPLE using the *ftmodi* command:

   ```
   ftshwi -l @a
   ```

   Output on Unix systems:

   ```
   Instance Address           Directory
   -------  -------           -----------
   std      MAPLE             /var/openFT/std
   ```

   Output on Windows systems:

   ```
   Instance Address              Directory
   -------   -------              -----------
   cluster   OPENFT.XYZ.NET       S:\openFT\cluster
   std       MAPLE                C:\Program Files\openFT\var\std
   ```

   The standard instance only logs into the address MAPLE and only accepts inbound connections for all the address MAPLE.

3. You enter

   ftshwi in a cluster configuration with several instances:

   ```
   ftshwi -l @a
   ```

   Output on Unix systems:

   ```
   Instance Address            Directory
   -------  -------            -----------
   maple    CL_MAPLE           /sha_MAPLE/oFT
   beech    CL_BEECH           /sha_BEECH/oFT
   std      MAPLE              /var/openFT/std
   ```

   Output on Windows systems:

   ```
   Instance Address              Directory
   -------   -------              -----------
   cluster   OPENFT.XYZ.NET       S:\openFT\cluster
   std    P870_DDM       C:\ProgramData\Fujitsu Technology Solutions
   \openFT\var\std
   ```

4. You enter *ftshwi* in single-user mode on Unix systems.
   ```

```
ftshwi -l @a
Instance  Addresse             Owner
                              Directory
-------  -------              -----------
aba      host0                aba
                              /var/openFT/aba
std      MAPLE                fts
                              /var/openFT/std
```

Messages of the ftshwi command

If *ftshwi* could not be executed properly, a self-explaining message is output. The exit code is not equal zero in this case.

3.60 ftshwk

Note on usage

Function: Show properties of RSA keys

User group: FT administrator

Functional description

You can use the *ftshwk* command to output the properties of RSA keys. You can display the RSA keys of your own instance as well as the RSA keys of partners.

Format

ftshwk -h

[-own]
[-id=<identification 1..64> | -id=@a]
[-pn=<partner 1..200> | -pn=@a] |
[-exp=n | -exp=e | -exp=yyyymmdd | -exp=1..999]
[-csv]

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-own

Displays the key for your own instance.

-own must not be specified in combination with *-pn* or *-id*.

-id=identification | **-id=@a**

identification is the instance identification of the partner whose key is to be displayed. *-id* must not be specified in combination with *-pn* and *-own*.

@a

Displays the installed keys of all partner systems.

-pn=partner | **-pn=@a**

partner is the name of the partner system in the partner list or the address of the partner system whose key is to be displayed.

-pn must not be specified in combination with *-id* and *-own*.

You will find detailed information on address specifications in the section “Specifying partner addresses”.

@a

Displays the installed keys of all partner systems.

Neither *-id* nor *-pn* nor *-own* specified

Displays the keys of your own instance and the installed keys of all the partner systems.

-exp=n | **-exp=e** | **-exp=**yyyymmdd | **-exp=**1..999

Selects the keys on the basis of their expiration date.

n (no)

Displays all partner keys that do not have an expiration date.

e (expired)

Displays all partner keys that have already expired.

yyyymmdd

Displays all partner keys that expire at the latest at 00:00 local time on the specified date. For example, 20171201 displays all the keys that will become invalid by 00:00 on 01.12.2017.

1..999

Displays all partner keys that will expire within the specified number of days.

-exp not specified

The expiration date is not a selection criterion.

-csv

-csv specifies that the key properties are to be output in CSV format. The values in the output are separated by semicolons.

-csv not specified

The key properties are output in the default format.

Example

You want to output the properties of all the keys:

```
ftshwk
CRE-DATE   EXP-DATE   KEY-LEN KEY-REF   AUTHL PARTNER  IDENTIFICATION
2021-07-20            768     5         2
2021-07-20            1024    5         2
```

```
2021-07-20            2048    5         2
2021-07-20            3072    5         2
2021-07-20            4096    5         2
2015-01-31            1024    6         2
2015-02-29            2048    7         2
2015-03-28 2017-12-24 2048    7         2     MYOWN    MYOWNID.DOMAIN.NET
2015-07-12 EXPIRED    768     12        2     PC17QD   PC17QD.DOMAIN.NET
2017-01-14            2048    1036      1     PC27ABC  PC27ABC.DOMAIN.NET
```

Explanation:

CRE-DATE

Date on which the key was generated.

EXP-DATE

Date on which the key expires, i.e. 00:00 on the specified day. EXPIRED means that the key has already expired.

If there is no specification here then there is no expiration date.

KEY-LEN

Key length in bit: 768, 1024, 2048, 3072 or 4096

KEY-REF

Key reference

AUTHL

Authentication level: 1 or 2

PARTNER

Partner's name. This field is left empty for keys belonging to your own instance.

IDENTIFICATION

Partner's instance ID. This field is left empty for keys belonging to your own instance.

3.61 ftshwl

Note on usage

Function: Display log records and offline log files

User group: FT user and FT, FTAC and ADM administrator

Functional description

With *ftshwl*, you can obtain information on all openFT requests logged up to now by openFT. In addition, you can output the names of the current log file and the offline log files.

You can display all log records entered under your own login name.

If you are the FT, FTAC or ADM administrator, you can view log records of all user IDs. The log records are stored in the file *syslog.Lyymmdd.Lhhmmss*. This file is located in the *log* directory of the relevant openFT instance, see also section “Instance identification”. *yymmdd* is the date (year, month, day) and *hhmmss* the time (hour, minute, second for GMT) at which the file was created.

In the case of the standard instance, the pathname of the *log* directory is */var/openFT/std/log/syslog* (Unix systems) and *C:\ProgramData\Fujitsu Technology Solutions\openFT\var\std\log* (Windows systems), respectively. For details on newly created instances, see the command *ftcrei* .

The log records are marked as FT, FTAC and ADM log records respectively, which means that you can determine the type of log record from the output.

For every request, there is an FTAC log record in which you can find the result of the FTAC admission check. For transfer requests, openFT logs whether it was actually able to execute this request in FT log records and for remote administration requests in ADM log records.

If no options are specified, openFT outputs the current log record. If options are specified, openFT outputs all log records up to the time specified in the command in reverse chronological order, i.e. starting from the most recent record to the oldest record.

The polling options allow you to specify that the output of new log records is to be repeated at regular intervals.

There are three types of output: short output, long output and CSV output (**C**haracter **S**eparated **V**alue).

Output is written to standard output, see also the note on Unix systems at the description of the *-fn* option.

Format

ftshwl -h |

[<user ID> | @a]
[-lf=<file name1..512> | -tlf=yyyymmdd[hh[mm[ss]]]]
[-plf=<0..3>]
[-rg=[[[[yyyy]mm]dd]hhmm|#1..999999999999|0..999|:0..999][-

[[[[yyyy]mm]dd]hhmm|#1..999999999999|0..999|:0..999]]]

[-rt=[t][c][a]]
[-ff=[t][T][m][r][d][a][C][D][M][l][f]]
[-ini=l | -ini=r | -ini=lr | -ini=rl]
[-pn=<partner 1..200>]
[-fn=<file name 1..512>]
[-rc=0..ffff | -rc=@f]
[-tid=1..2147483647]
[-gid=<global request ID 1..4294967295>]
[-adm=<administrator ID 1..32>]
[-ri=<routing info 1..200>]
[-llf]
[-nb=1..99999999 | -nb=@a]
[-po=<polling interval 1..600>

[-pnr=<polling number 1..3600>]]

[-l][-csv]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

user ID | **@a**

is used to specify the login name(s) for which log records are to be displayed. As ordinary user, you can only specify your own login name. As the administrator, you can specify any login name.

@a for *user ID*

This also displays information, but only on the log records that refer to your own login name.

FT or FTAC or ADM administrators can display the log records for all login names.

user ID not specified

Only the log records for the login name under which the command was entered are displayed.

-lf=file name | **-tlf=**yyyymmdd[hh[mm[ss]]]

Selects the log file(s) whose log records or name are to be used. This means that you can also view offline log records.

-lf=file name

The log file is selected based on its file name. You must specify the full relative or absolute path name. If no log file exists with the specified file name then an error message is output.

-tlf=yyyymmdd[hh[mm[ss]]]

The log file is selected based on its creation time (local time!). The log file created at or before the specified time is selected. If more than one log file corresponds to the specified time then the next oldest log file is selected.

You must at least specify the date as an 8-digit value indicating the year month and day. The year must be greater than or equal to 2000.

You can specify the time (hhmmss) partially or not at all if you wish. "00" is added to replace any missing specifications. See also Examples.

Neither *-lf* nor *-tlf* specified

The current log file is used.

-plf=number

Specifies the number of preceding log files (0 to 3) that are to be selected in addition to the current file or the file specified with *-lf* or *-tlf*.

-plf not specified

Selects only the current log file or the log file specified with *-lf* or *-tlf*.

> If you omit the options *-plf* and *-lf* or *-tlf* then this corresponds to the behavior up to openFT V11.0.

-rg=[[[[yyyy]mm]dd]hhmm]-[[[[yyyy]mm]dd]hhmm]

With *-rg* you can specify the start and/or end of a logging interval.

[[[yyyy]mm]dd]hhmm

A 4-digit specification is interpreted as the time expressed in hours and minutes, a 6-digit specification as the day (date) and time in hours and minutes, an 8-digit specification as the month, day, and time in hours and minutes, and a 12-digit specification as the year, month, day, and time in hours and minutes. The largest possible value that can be specified as the date is 20380119 (January 19, 2038).

openFT then displays all the log records written during the specified time period. The older time is taken to be the start time and the earlier time as the end time.

If optional data (*[[[yyyy]mm]dd]*) is omitted, then it is automatically replaced by current values.

If you omit the limit after the dash, the current time is taken. If you omit the limit before the dash, the time of the first log record written is taken.

-rg=-

Displays everything (same meaning as *-nb=@a*)

-rg=[[[yyyy]mm]dd]hhmm

If the minus sign is missing, the range is the exact minute specified. The largest possible value that can be specified as the date is 20380119 (January 19, 2038). If optional data (*[[[yyyy]mm]dd]*) is omitted, then it is automatically replaced by current values.

-rg=[#1..999999999999]-[#1..999999999999]

-rg is used to specify the start and/or end of a range of log IDs.

#1..999999999999

The selection of a log ID is indicated by the leading # character. openFT then displays all the log records which lie within the specified range.

If the log ID limit before the dash is omitted, the current ID is taken, and if the log ID limit after the dash is omitted, the ID of the first log record written is taken.

-rg=#1..999999999999

If the minus sign is omitted, the range is restricted to the specified log ID only.

-rg=[0..999][-[0..999]]

Here you specify with *-rg* a relative time period as a multiple of 24 hours (i.e. as a number of days). Note that the relative time period is calculated with an accuracy of one second from the current time. You have the following options (*d1* and *d2* 1 through 3 digits):

- *-rg=d1-d2* outputs all log records that are between *d1* and *d2* days old, irrespective of whether *d1* is larger or smaller than *d2*.
- *-rg=d1-* outputs all log records that are no more than *d1* days old.
- *-rg=-d2* outputs all log records that are at least *d2* days old.

-rg=[**:**0..999][-[**:**0..999]]

Here you specify with *-rg* a relative time period in minutes. You have the following options in this case (*m1* and *m2* 1 through 3 digits):

- *-rg=m1-:m2* outputs all log records that are between *m1* and *m2* minutes old, irrespective of whether *m1* is larger or smaller than *m2*.
- *-rg=:m1* (or *-rg=:m1-*) outputs all log records that are no more than *m1* minutes old.
- *-rg=-:m2* outputs all log records that are at least *m2* minutes old.

-rg not specified

The range is not a selection criterion.

-rt=[**t**][**c**][**a**]

Defines which type of log record is to be displayed.

You may specify *t*, *c*, *a* and any combination of these values:

t

The FT log records are displayed.

c

The FTAC log records are displayed.

a

The ADM log records are displayed. For further details, refer to the manual "openFT (Unix and Windows systems) - Installation and Operation".

-rt not specified

The record type is not a selection criterion.

-ff=[**t**][**T**][**m**][**r**][**d**][**a**][**C**][**D**][**M**][**l**][**f**]

Defines the FT function for which log records are to be output. Possible values are: *t*, *T, m*, *r*, *d*, *a*, *C*, *D*, *M*. *l*, *f* or any combination of these values.

The entries *m*, *r*, *d*, *a*, *C*, *D*, *M* and *l* are only reasonable for FTAC log records. The entry *f* is only reasonable for ADM log records. *t* and *T* are reasonable for all log records.

t

All log records for the function "transfer files" are output.

T

All log records for the function "transfer directories" are output.

m

All log records for the function "modify file attributes" are output.

r

All log records for the function "read directories" are output.

d

All log records for the function "delete files" are output.

a

All log records for the function "read file attributes" are output.

C

All log records for the function "Create directory" are output.

D

All log records for the function "Delete directory" are output.

M

All log records for the function "Modify directory" are output.

l

All log records for the function "inbound FTP access" are output. These log records are written if incorrect admission data (FTAC transfer admission or user ID/password) was specified for inbound FTP access.

f

All ADM log records of the "Routing" function are output on the remote administration server. Output can be further restricted with the *-adm* and *-ri* options. This specification is only of significance to the administrator of the remote administration server.

-ff not specified

The FT function is not a selection criterion.

-ini=l | **-ini=r** | **-ini=lr** | **-ini=rl**

Defines the initiator for which log records are to be output. Possible values are: *l*, *r*, *lr*, *rl*.

l (local)

Only log records belonging to openFT requests issued locally are output.

r (remote)

Only log records belonging to openFT requests issued remotely are output.

lr, rl

The log records belonging to openFT requests issued locally and remotely are output.

-ini not specified

The initiator is not a selection criterion.

-pn=partner

Defines the partner system to which the log records are to be output. Partner is the name of the partner in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

For the partner name, you can also use the wildcard symbols '*' (asterisk) and '?' (question mark). * stands for any string and ? stands for any single character.

-pn not specified

The partner system is not a selection criterion.

-fn=file name

Defines the file to which the log records are to be output. You can specify wildcards such as "*" (asterisk, i.e. any character string) and "?" (question mark, i.e. single character).

File names may consist of Unicode characters.

Note on Unix systems

The following table shows the encoding settings for which search via file name and the display of the log records is working correctly:

Input file name	File name in log file	Terminal	System
ISO88591	ISO88591	ISO88591	UTF-8
ISO88591	ISO88591	UTF-8	UTF-8
UTF-8	UTF-8	UTF-8	UTF-8
UTF-8	UTF-8	ISO88591	UTF-8

-fn not specified

The file name is not a selection criterion.

-rc=0..ffff | @**f**

Defines the reason code as a selection criterion for log record output.

0 .. ffff

All log records with a specified reason code are output.

@f

All log records with reason codes other than 0000 are output. This criterion yields a list of log records for all requests terminated with error messages.

-rc not specified

The reason code is not a selection criterion.

-tid=request ID

-tid specifies the request ID for which you want to output the log records.

-tid not specified

The request ID is not a selection criterion.

-gid=global request ID

With the *-gid*, you specify the global request ID for which you want to display log records. The global request ID is only relevant for inbound requests from openFT and FTAM partners. It is assigned by the initiator of the request (transfer ID) and is sent to the local system.

-gid= not specified

The global request ID is not used as a selection criterion.

-adm=administrator ID

-adm specifies the administrator ID for which you want to output the ADM log records.

-adm not specified

The administrator ID is not a selection criterion.

-ri=routing info

-ri specifies the routing information for which you want to output the ADM log records.

-ri not specified

The routing info is not a selection criterion.

-llf

outputs the names of log files. *-llf* is only permitted on its own or in combination with the options *-lf*, *-tlf*, *-plf*, *-csv* or *-h*. If any other combination is used then the command is rejected.

-llf without *-lf*, *-plf* or *-tlf* outputs the names of all the log files (current log file together with all the offline log files (up to a maximum of 1024)). To restrict the output, you can also specify *-lf*, *-plf* or *-tlf*, see also Examples.

-llf not specified

Log records that correspond to the current selection criteria are displayed.

-nb=number | **@a**

Defines the number of log records to be output.

@a for *number*

All log records are output.

-nb not specified

If *-rg* is specified simultaneously, *-nb* is replaced by the value *-nb=@a*.

If *-rg* is also not specified, *-nb* is replaced by the value *-nb=1*.

-po=polling interval

The *polling interval* indicates the time between repetitions in seconds. On each repetition, all the new log records are filtered in accordance with the specified selection criteria and the detected records are output.

If you also specify *-pnr*, you can limit the number of times the data is output. If you specify *-po* without *-pnr*, output is repeated an unlimited number of times.

If repeated output has been started with the *-po* option (with or without *-pnr*), it can be canceled by an interrupt signal (e.g. Ctrl+C). In addition, the operation is canceled if an error occurs. When the asynchronous server is stopped, output is not interrupted but continues to be issued.

-po must not be specified in combination with *-lf*, *-llf*, *-plf*, *-tlf*, *-tid*, *-gid*, *-nb* or *-rg*.

Possible values: 1 through 600.

> **i** No log records should be deleted during polling as otherwise discontinuities in the output may appear!

-po not specified

The log records are output immediately and once only.

-pnr=polling number

-pnr specifies the number of repetitions.

-pnr can only be specified in conjunction with *-po*.

Possible values: 1 through 3600.

-pnr not specified

The output is repeated without restriction.

-l

Defines that the log records are to be output in long form.

-l not specified

The log records are output in short form if *-csv* has not been specified.

-csv

You can use *-csv* to specify that the log records are to be output in the CSV format. The values in the output are separated by semicolons.

If *-csv* is specified, output is always in long form (analogous to *-l*) regardless of whether or not *-l* has also been specified.

-csv not specified

The log records are output in the standard format, i.e. in abbreviated form if *-l* is not specified and in detailed form if *-l* is specified.

Examples

The following examples each output the log records for the user's own ID. If you are an FT, FTAC or ADM administrator and want to output the log records for all user IDs, you must also specify *@a*.

1. All log records that are more than two days (48 hours) old are output:

   ```
   ftshwl -rg=-2
   ```

2. All log records that are more than 15 minutes old but less than 30 minutes old are output:

   ```
   ftshwl -rg=:15-:30
   ```

3. All log records that are less than 30 minutes old are output:

   ```
   ftshwl -rg=:30
   ```

4. All log records that are more than 30 minutes old are output:

   ```
   ftshwl -rg=-:30
   ```

5. The last 10 log records where FTAC checks failed (reason code not equal to 0) are output:

   ```
   ftshwl -rc=@f -rt=c -nb=10
   ```

6. The name of the current log file and the names of the two preceding offline log files are to be output:

   ```
   ftshwl -llf -plf=2
   ```

7. Output of 100 log records from the log file that was created on or before 24.02.2017 00:00:

   ```
   ftshwl -tlf=20170224 -nb=100
   ```

   Note

   -tlf=20170224 is extended to *-tlf=20170224000000.* If, for example, there are three log files with the creation dates 20170224 13:30:00, 20170217 10:00:00 and 20170210 08:00:00, then the file with the date 20170217 10:00:00 is taken as the next oldest file.

8. All records that were created for the filename remote-file. The short output will only show the name of the local file.

   ```
   ftshwl -fn=remote-file
   TYP LOG-ID TIME     RC    PARTNER  INITIAT. PROFILE  USER-ADM FILENAME
   2021-03-19
   T    62779 12:45:20 0000 >locBS    root             root     local-file
   ```

   To see the remote filename the long output format has to be used:

```
# ftshwl -fn=remote-file -l
LOGGING-ID  = 62779     RC      = 0000        TIME     = 2021-03-19 12:45:20
   TRANS    = TO        REC-TYPE= FT          FUNCTION = TRANSFER-FILE
   PROFILE  =           PCMD    = NONE        STARTTIME= 2021-03-19 12:45:20
   TRANS-ID = 65551     WRITE   = REPLACE     REQUESTED= 2021-03-19 12:45:19
   TRANSFER =        10 kB                    CCS-NAME = ISO88591
   INITIATOR= root
   USER-ADM = root
   PARTNER  = locBS
   FILENAME = local-file
   FNC-MODE = *TRANSPARENT
   REMOTE-FN= remote-file
```

3.61.1 Description of log record output

Log records can be displayed using the openFT Explorer or by using the *ftshwl* command. You can choose between a short overview, detailed information or, if further processing is to be performed with external programs, output in the CSV format.

The log records are identified by log IDs. The log IDs are assigned in ascending order, but for technical reasons the numbering is not contiguous (i.e. there may be gaps).

3.61.1.1 Logging requests with preprocessing/postprocessing

For security reasons, only the first 32 characters (or 42 characters in the case of *ftexecsv* preprocessing) of a preprocessing or postprocessing command are transferred to the log record. By arranging the call parameters appropriately or by inserting blanks, you can influence which command parameters do not appear in the log.

3.61.1.2 Short output format of a FT or FTAC log records

Example: The option *-rt=tc* causes only FT and FTAC log records to be output.

Unix systems:

```
$ftshwl -rt=tc -nb=11
TYP LOG-ID TIME     RC    PARTNER  INITIAT. PROFILE  USER-ADM FILENAME
2017-02-02
TTD   8276 09:20:33 0000 <%ip10.1* user2*            user2    trans/dir1
CTD   8274 09:20:32 0000 <%ip10.1* user2*            user2    trans/dir1
CA    8273 09:16:07 0000 >PARTLINU *REMOTE  pr1      user1    file.10
CA    8272 09:16:07 0000 >PARTLINU user1             user1
CD    8271 09:15:30 0000 <PARTLINU *REMOTE  pr1      user1    file.new
CD    8270 09:15:30 0000 <PARTLINU user1             user1
CM    8269 09:15:03 0000 <PARTLINU *REMOTE  pr1      user1    file.rem
CM    8268 09:15:03 0000 <PARTLINU user1             user1    file.new
CR    8267 09:14:14 0000 >PARTLINU *REMOTE  pr1      user1    .
CR    8266 09:14:14 0000 >PARTLINU user1             user1
T     8265 09:13:50 0000 >PARTLINU user1             user1    file.10
```

Windows systems:

```
ftshwl -rt=tc -nb=12
TYP LOG-ID TIME      RC     PARTNER   INITIAT. PROFILE   USER-ADM FILENAME
2017-02-02
TTD    3305 14:43:33 0000 <%ip10.1* miller             miller    H:/transdir
CTD    3303 14:43:32 0000 <%ip10.1* miller             miller    H:/transdir
T      3302 14:42:27 0000 <pitti     *REMOTE            DOMAIN1* Trace1.txt
C      3301 14:42:27 0000 <pitti     *REMOTE   profil01 DOMAIN1* Trace1.txt
CCD    3300 14:16:41 0000 <pitti     *REMOTE            thomasw   D:/current
T      3299 14:03:48 0000 <pitti     *REMOTE            peter     readme.txt
T      3296 14:02:32 0000 >pitti     smith              smith     C:/f01.txt
C      3294 14:02:07 0000 >pitti     miller             miller    C:/rme.txt
T      3292 13:56:07 0000 >pitti     *REMOTE            DOMAIN1* |ftexecsv
ftshwo -b -a -u
T      3289 09:09:10 2072 >cog2-te* miller             miller    tw.txt
T      3287 08:51:29 2072 >cog2-te* DOMAIN1*           DOMAIN1* tw.txt
T      3286 09:46:34 0000 <servus.* DOMAIN1*           DOMAIN1* *CMDOUT
```

Explanation

TYP

Comprises three columns. The first column specifies whether the log record is an FT or FTAC log record:

T

FT log record

C

FTAC log record

The second and third column identify the FT function:

_

(empty): transfer file

A

read file attributes (only in the FTAC log record)

D

delete file (only in the FTAC log record)

C

create file (only in the FTAC log record)

possible only for transfer requests issued in the remote partner system

M

modify file attributes (only in the FTAC log record)

R

read directory (only in the FTAC log record)

CD

create directory (only in FTAC log record)

DD

delete directory (only in FTAC log record)

MD

modify directory attributes (only in FTAC log record)

TD

transfer directory (FT main log record or FTAC log record)

SD

transfer directory (FT log record for creating a subdirectory)

SF

transfer directory (FT log record for transferring a file)

L

Login: Failed inbound FTP access (only in FTAC log record)

LOG-ID

Log record number

TIME

specifies time when the log record was written

RC

Reason code. Specifies whether a request was successful (RC=0) or if not, why it was rejected or cancelled. Additional information on the reason code is available using the *fthelp* command.

PARTNER

Provides information about the partner system involved. The name in the partner list or the address of the partner system, possibly truncated to 8 characters, or the name under which the partner system is entered in the TNS is output.

The name or address of the partner system is preceded by an identifier to indicate the direction of the request.

>

The request is sent to the partner system. This transfer direction is specified for:

- a send request
- a request to display file attributes
- a request to display directories

<

The request is sent to the local system. This transfer direction is specified for:

- a receive request
- a request to modify file attributes

(When a FTAM partner modifies the access rights of a local file, two log records are written. No direction is specified in front of PARTNER in this case.)

a request to delete files

INITIAT.

Request initiator. If initiated in the remote system: *REMOTE.

PROFILE

Name of the profile used for file transfer (only in FTAC log record).

USER-ADM

Login name to which the requests in the local system refer.

If a login name longer than 8 bytes was specified, the first seven bytes are output, followed by an asterisk (*).

FILENAME

Local file name

3.61.1.3 Short output format of an ADM log record

In the following examples, the option *-rt=a* causes only ADM log records to be output.

1. Output ADM log records on a client:

```
ftshwl ftadmin -rt=a -nb=5
TYP LOG-ID TIME     RC    PARTNER  INITIAT. PROFILE  USER-ADM FILENAME
2017-01-16
A       39 04:30:35 0000 <flexthom ftadmin           ftadmin
A       36 04:30:15 0000 <flexthom ftadmin           ftadmin
A       33 04:29:49 0000 <flexthom ftadmin           ftadmin
A       30 04:28:15 0000 <flexthom ftadmin           ftadmin
A       27 04:22:56 0000 <flexthom ftadmin           ftadmin
```

2. Output ADM log record on the administered openFT instance:

```
ftshwl -rt=a
TYP LOG-ID TIME     RC    PARTNER  INITIAT. PROFILE  USER-ADM FILENAME
2017-01-16
A     2575 13:30:15 0000 >ftadm:/* *REMOTE  adminrem admin001
```

3. Output routing ADM log record on the remote administration server:

```
ftshwl -rt=a -ff=f
TYP LOG-ID TIME     RC    PARTNER  INITIAT. PROFILE  USER-ADM FILENAME
2017-01-16
AF     396 13:22:54 0000 >Testrech *REMOTE  adminacc admin002
```

Explanation

The following differences apply to ADM log records compared with FT or FTAC log records:

- The value *A* is output for an ADM log record in the TYP column. In the case of ADM log records with routing information on the remote administration server (*ftshwl -ff=f*), the value *F* is also shown in column 2.
- The FILENAME column is empty for ADM log records.

3.61.1.4 Long output format of an FT log record

The log records with the numbers 175, 193 and 405 are to be output in long form:

```
ftshwl @a -rg=#175 -l
LOGGING-ID  = 175       RC      = 0000        TIME     = 2017-02-02 13:14:30

   TRANS    = FROM      REC-TYPE= FT          FUNCTION = TRANSFER-FILE
   PROFILE  =           PCMD    = NONE        STARTTIME= 2017-02-02 13:14:30

   TRANS-ID = 65554     WRITE   = REPLACE     STORETIME= 2017-02-02 13:14:30

   TRANSFER =         1 kB                    CCS-NAME = ISO88591 (1)
                                               CHG-DATE = SAME
   SEC-OPTS = ENCR+DICHK+RAUTH2, RSA-2048 / AES-256
   INITIATOR= *REMOTE                         GLOB-ID  = 92183
   USER-ADM = user004
   PARTNER  = mn122
   PTNR-ADDR= %ip192.168.0.133
   FILENAME = example
   FNC-MODE = *TRANSPARENT
   REMOTE-FN= remote-example (4)
ftshwl @a -rg=#193 -l
LOGGING-ID  = 193       RC      = 2164        TIME     = 2017-02-02 13:31:16

   TRANS    = TO        REC-TYPE= FT          FUNCTION = TRANSFER-FILE
   PROFILE  =           PCMD    = NONE        STARTTIME= 2017-02-02 13:31:16

   TRANS-ID = 65568     WRITE   = REPLACE     REQUESTED= 2017-02-02 13:31:15

   TRANSFER =         0 kB                    CCS-NAME = ISO88591 1
   SEC-OPTS = RAUTH
   INITIATOR= smith (2)
   USER-ADM = smith (2)
   PARTNER  = mn122
   FILENAME = text.txt
   FNC-MODE = *CHAR, FNCCS=utf8
   REMOTE-FN= text.remote.txt (4)
   ERRINFO  = No unicode filename support
ftshwl @a -rg=#405 -l
```

```
 LOGGING-ID  = 405       RC      = 0000        TIME     = 2017-02-03 08:33:02

    TRANS    = FROM      REC-TYPE= FT          FUNCTION = TRANSFER-DIR
    PROFILE  =           PCMD    = NONE        STARTTIME= 2017-02-03 08:33:02

    TRANS-ID = 67867     WRITE   = REPLACE     REQUESTED= 2017-02-03 08:33:01

    TRANSFER =       358 kB                    CCS-NAME = ISO88591
    SEC-OPTS = ENCR+DICHK+LAUTH2, RSA-2048 / AES-256
    TRANSFILE= 4/14
    INITIATOR= smith  (2)
    USER-ADM = smith  (2)
    PARTNER  = %ip192.168.0.144
    FILENAME = trans/test/file1.c  (3)
    FNC-MODE = *TRANSPARENT
    REMOTE-FN= remote-file.c (4)
```

(1) On Windows systems: CP1252, for example

(2) On Windows systems: COG\smith, for example

(3) On Windows systems: D:/trans/test/file1.c, for example

(4) for outbound requests and openFT version >= V12.1C10

Explanation

LOGGING-ID

Log record number; up to twelve characters in length

TRANS

Transfer direction

TO

Transfer direction to the partner system. This transfer direction is specified for

- a send request
- a request to display the file attributes
- a request to display the directories

FROM

Transfer direction to the local system. This transfer direction is specified for

- a receive request
- a request to modify the file attributes
- a request to delete files

PROFILE

Name of profile used

TRANS-ID

Request number

TRANSFER

Number of bytes transferred

SEC-OPTS

Security options used during transfer

ENCR

Encryption of the request description

DICHK

Data integrity check of the request description

DENCR

Encryption of the transferred file content

DDICHK

Data integrity check of the transferred file content

LAUTH

Authentication of the local system in the remote system (authentication level 1)

LAUTH2

Authentication level of the local system in the remote system (authentication level 2)

RAUTH

Authentication of the remote system in the local system (authentication level 1)

RAUTH2

Authentication level of the remote system in the local system (authentication level 2)

RSA-nnn

Length of the RSA key used for the encryption

AES-128 / AES-256 / DES

The encryption algorithm used

TRANSFILE

In case of directory transfer this field shows the number of completed subrequests and the total number of subrequests. 4/14 means 4 subrequests completed and 14 subrequests in total, for example.

INITIATOR

Request initiator. If initiated in the local system: login name. If initiated in the remote system: *REMOTE

USER-ADM

Login name to which the requests in the local system refer

PARTNER

Identifies the partner system in question.

The name in the partner list or the address of the partner system, possibly truncated to 8 characters, or the name under which the partner system is entered in the TNS is output.

In the case of requests issued from a remote computer, it is also possible for *%strange* to be output followed by a part of the address of the partner system if the partner system is not entered in the TNS and TCP/IP-RFC1006 was not used as the transport system.

Note for Windows systems: In this case, *%strange* followed by the DTE address of the partner system is shown for X.25 links in Windows, for example.

PTNR-ADDR

Address of the partner system, only output for inbound requests.

FILENAME

Local file name

FNC-MODE

Encoding mode for file names and follow-up processing:

*TRANSPARENT

File names and follow-up processing are represented in a fixed binary code, independent of local character code settings (transparent mode).

*CHAR, FNCCS=ccs

File names and follow-up processing are seen in their character presentation (character mode). *ccs* specifies the character set that was relevant for the creation of the FT request, e.g. *utf8* on Windows systems.

REMOTE-FN

Filename in remote system (outbound, openFT version >= V12.1C10).

ERRINFO

Additional information on the error message if an error occurred during a transfer.

RC

Reason code. Specifies whether a request was successful (RC=0) or if not, why it was rejected or cancelled. You can obtain further information with the *fthelp* command.

REC-TYPE

Specifies whether the log record is an FT log record.

PCMD

Indicates whether follow-up processing was specified and started. Possible values:

NONE

No follow-up processing specified.

STARTED

Follow-up processing was started (contains no information about the successful completion of follow-up processing!).

NOT-STARTED

Follow-up processing could not be started.

> In case of directory transfer a follow-up processing is executed for all files of a directory, but not for creating sub-directories. The main log record which is also written when the FT-DIR logging is deactivated always indicates the status of the last individual file transfer.

WRITE

Write mode. The field is assigned a value only for outbound requests; for inbound requests, it contains a blank. Possible values:

NEW

A new file is created. If a file with this name already exists, file transfer is aborted.

EXT

An existing file is extended, otherwise a new is created.

REPLACE

An existing file is overwritten. If it does not already exist, it is created.

TIME

Specifies time when log record was written

FUNCTION

FT function. Possible values:

TRANSFER-FILE

Transfer file

TRANSFER-DIR

Transfer directory

STARTTIME

Indicates the start time of the request.

STORETIME

If the request was submitted in the remote system then the time of the entry in the request queue is displayed here.

REQUESTED

When initiative in the local system, the time of issue of the request is shown here.

> Depending on the initiator of the request (local or remote), either STORETIME or REQUESTED is output but never both together.

CCS-NAME

Name of the character set used to code the local file.

CHG-DATE

Specifies whether the change date of the send file is taken over for the receive file.

SAME

The modification date of the send file is taken over.

GLOB-ID

Global request identification, displayed in the case of inbound requests from openFT and FTAM partners (INITIATOR=REMOTE). This corresponds to the request identification (=TRANSFER-ID) on the initiator system.

3.61.1.5 Long output format of an FTAC log record

The log record with log record number 5172 or 947 is to be output in long form:

Unix systems:

```
ftshwl @a -rg=#5172 -l
LOGGING-ID  = 00005172  RC      = 0000        TIME     = 2016-11-16 09:38:06
   TRANS    = TO        REC-TYPE= FTAC        FUNCTION = TRANSFER-FILE
   PROFILE  = remadmin  PRIV    = NO
   INITIATOR= *REMOTE
   USER-ADM = thomasw
   PARTNER  = angel.domain1.de
   FILENAME = |ftexecsv ftshwo -tn -a -u -ccs=ISO88591
```

Windows systems:

```
ftshwl @a -rg=#947 -l

LOGGING-ID  = 947       RC      = 0000        TIME     = 2016-11-16 10:42:45

   TRANS    = TO        REC-TYPE= FTAC        FUNCTION = TRANSFER-FILE

   PROFILE  =           PRIV    =
   INITIATOR= DOMAIN1\thomasw

   USER-ADM = DOMAIN1\thomasw

   PARTNER  = servus
   FILENAME = test2.txt
```

Explanation

LOGGING-ID

Log record number, up to twelve characters in length

TRANS

Transfer direction

TO

Transfer direction to the partner system. This transfer direction is specified for

- a send request
- a request to display the file attributes
- a request to display the directories

FROM

Transfer direction to the local system. This transfer direction is specified for

- a receive request
- a request to modify the file attributes
- a request to delete files

BOTH

The request direction is to the partner system and to the local system. When an FTAM partner modifies the access rights of a local file, two log records are written. The direction BOTH is specified in each.

PROFILE

Name of the profile used

INITIATOR

Request initiator. If initiated in the local system: login name. If initiated in the remote system: *REMOTE

USER-ADM

Login name to which the requests in the local system refer

PARTNER

Identifies the partner system in question.

The name in the partner list or the address of the partner system, possibly truncated to 8 characters, or the name under which the partner system is entered in the TNS is output.

In the case of requests issued from a remote computer, it is also possible for *%strange* to be output followed by a part of the address of the partner system if the partner system is not entered in the TNS and TCP/IP-RFC1006 was not used as the transport system.

Note for Windows systems: In this case, *%strange* followed by the DTE address of the partner system is shown for X.25 links in Windows, for example.

FILENAME

Local file name

RC

Reason code. Specifies whether a request was successful (RC=0) or if not, why it was rejected or cancelled. You can use the *fthelp* command to obtain further information.

REC-TYPE

Specifies whether the log record is an FTAC log record.

PRIV

Specifies whether or not the FT profile being used is privileged

TIME

Specifies time when the log record was written

FUNCTION

FT function

TRANSFER-FILE

Transfer file

TRANSFER-DIR

Transfer directory

READ-FILE-ATTR

Read file attributes

DELETE-FILE

Delete file

CREATE-FILE

Create file (possible only in requests submitted in the remote partner system)

MODIFY-FILE-ATTR

Modify file attributes

READ-FILE-DIR

Read directories

CREATE-FILE-DIR

Create file directory

DELETE-FILE-DIR

Delete file directory

MODIFY-FILE-DIR

Modify file directory

LOGIN

Login: Inbound FTP access.

This log record is written if incorrect admission data was specified for inbound FTP access.

3.61.1.6 Long output format of an ADM log record

In the following examples, the option *-rt=a* causes only ADM log records to be output.

1. ADM log record on a client:

```
ftshwl -rt=a -l
LOGGING-ID  = 27        RC      = 0000        TIME     = 2016-11-16 04:22:56
   TRANS    = FROM      REC-TYPE= ADM         FUNCTION = REM-ADMIN
   TRANS-ID = 190845    PROFILE =
   SEC-OPTS = ENCR+DICHK, RSA-768 / AES-256
   INITIATOR= ftadmin
   USER-ADM = ftadmin
   PARTNER  = flexthom
   ADM-CMD  = ftshwo
   ADMIN-ID =
   ROUTING  = Muenchen/Jonny
```

2. ADM log records on the remote administration server:

```
ftshwl -rt=a -l -nb=3
LOGGING-ID  = 400       RC      = 0000        TIME     = 2016-11-16 13:22:56
   TRANS    = TO        REC-TYPE= ADM         FUNCTION = REM-ADMIN
   TRANS-ID = 65608     PROFILE = adminacc
   SEC-OPTS = ENCR+DICHK, RSA-2048 / AES-256
   INITIATOR= *REMOTE
   USER-ADM = admin002
   PARTNER  = ftadm://cog2-test-eng.homenet.de
   ADM-CMD  = ftshwo
   ADMIN-ID = Hugo
   ROUTING  = Munich/Jonny
LOGGING-ID  = 399       RC      = 0000        TIME     = 2016-11-16 13:22:55
   TRANS    = FROM      REC-TYPE= ADM         FUNCTION = REM-ADMIN
   TRANS-ID = 152973    PROFILE =
   SEC-OPTS = ENCR+DICHK, RSA-2048 / AES-256
   INITIATOR= admin002
   USER-ADM = admin002
   PARTNER  = Test0001
   ADM-CMD  = ftshwo
   ADMIN-ID =
   ROUTING  =
LOGGING-ID  = 396       RC      = 0000        TIME     = 2016-11-16 13:22:54
   TRANS    = TO        REC-TYPE= ADM         FUNCTION = REM-ADMIN-ROUT
   TRANS-ID =           PROFILE = adminacc
   SEC-OPTS =
   INITIATOR= *REMOTE
   USER-ADM = admin002
   PARTNER  = Test0001
   ADM-CMD  = ftshwo
   ADMIN-ID = Hugo
   ROUTING  = Munich/Jonny
```

3. ADM log record on the administered openFT instance:

```
ftshwl -rt=a -l
LOGGING-ID  = 2571      RC      = 0000        TIME     = 2016-11-16 13:29:49
   TRANS    = TO        REC-TYPE= ADM         FUNCTION = REM-ADMIN
   TRANS-ID = 334030    PROFILE = adminrem
   SEC-OPTS = ENCR+DICHK, RSA-2048 / AES-256
   INITIATOR= *REMOTE
   USER-ADM = admin001
   PARTNER  = ftadm://flexthom.homenet.de
   ADM-CMD  = ftshwl
   ADMIN-ID =
   ROUTING  =
```

Explanation

LOGGING-ID

Log record number, up to twelve characters in length

RC

Reason code of the request.

TIME

Specifies time when the log record was written

REC-TYPE

ADM is always output here for ADM log records

FUNCTION

Administration function executed:

REM-ADMIN

Execute remote administration request

REM-ADMIN-ROUT

Check admission for remote administration request and forward remote administration request to the openFT instance to be administered if the admission check is successful

TRANS-ID

Number of the administration request

PROFILE

Name of the profile used

SEC-OPTS

Security options used during transfer:

ENCR

Encryption of the request description

DICHK

Data integrity check of the request description

DENCR

Encryption of the transferred file content

DDICHK

Data integrity check of the transferred file content

LAUTH

Authentication of the local system in the remote system (authentication level 1)

LAUTH2

Authentication level of the local system in the remote system (authentication level 2)

RAUTH

Authentication of the remote system in the local system (authentication level 1)

RAUTH2

Authentication level of the remote system in the local system (authentication level 2)

RSA-nnn

Length of the RSA key used for the encryption

AES-128 / AES-256 / DES

The encryption algorithm used

INITIATOR

Request initiator. If initiated in the local system: login name. If initiated in the remote system: *REMOTE

USER-ADM

User ID to which the remote administration request refers in the local system

PARTNER

Partner system involved. Depending on the location to which the ADM log record was written, the following is output:

- Client: Name/address of the remote administration server
- Remote administration server (inbound): Name/address of the client
- Remote administration server (outbound): Name/address of the openFT instance to be administered
- Administered openFT instance: Name/address of the remote administration server

ADM-CMD

Administration command without parameters

ADMIN-ID

Administrator ID under which the request is processed on the remote administration server. In the case of ADM log records on a client, this field is empty.

ROUTING

Routing information on the openFT instance to be administered

3.61.2 Reason codes of the logging function

The FTAC log records contain a reason code which indicates whether a request was accepted after the admission check successfully and if not, why it was rejected.

In ADM log records, the reason code specifies why a remote administration request was not executed.

You can use the *fthelp* command to output the message text associated with the code number:

`fthelp` *code-number*

In many codes, the last three digits correspond to the number of the associated openFT message.

In addition, there are a certain number of codes which do not correspond to openFT messages (see chapter “Messages” (openFT messages)). These are listed in the tables below:

RC	Reason
0000	Request successfully completed.
1001	Request rejected. Invalid transfer admission
1003	Request rejected. Transfer direction not permissible
1004	Request rejected. Illegal partner
1006	Request rejected. Violation of file name restriction
100f	Request rejected. Violation of success processing restriction
1010	Request rejected. Violation of failure processing restriction
1011	Request rejected. Violation of write mode restriction
1012	Request rejected. Violation of FT function restriction
1014	Request rejected. Violation of data encryption restriction
2001	Request rejected. Syntax error on file name extension
2004	Request rejected. Overall length of follow-up processing exceeds 1000 characters
3001	Request rejected. Invalid user identification
3003	Request rejected. Invalid password
3004	Request rejected. Transfer admission locked
3011	Request rejected. Violation of user outbound send level
3012	Request rejected. Violation of user outbound receive level
3013	Request rejected. Violation of user inbound send level
3014	Request rejected. Violation of user inbound receive level
3015	Request rejected. Violation of user inbound processing level

3016	Request rejected. Violation of user inbound file management level
3021	Request rejected. Violation of ADM outbound send level
3022	Request rejected. Violation of ADM outbound receive level
3023	Request rejected. Violation of ADM inbound send level
3024	Request rejected. Violation of ADM inbound receive level
3025	Request rejected. Violation of ADM inbound processing level
3026	Request rejected. Violation of ADM inbound file management level

RC	**Reason**
7001	The administrator ID is invalid. It was not possible to determine a valid administrator ID from the user ID or the profile name in the configuration data of the remote administration server.
7002	The routing information is invalid. The specified openFT instance specified in the routing information could not be found in the configuration data of the remote administration server.
7003	The specified remote administration command is invalid. The remote administration server rejects the specified command because it is not a supported remote administration command.
7101	Infringement against the access rights list. On checking the access rights, the system identified that the administrator ID has not been assigned the necessary rights in the configuration data of the remote administration server to be able to execute the valid remote administration command on the specified openFT instance.
7201	Infringement against the maximum command length. In particular in the case of BS2000 commands, the remote administration server replaces the shortest command names, which are guaranteed by openFT, by the full command names. If this replacement of the command name causes the entire remote administration command to become longer than the maximum command length of 8192 characters, the command is rejected.

3.62 ftshwlic

Note on usage

Function: Show license keys

User group: FT administrator

Functional description

You can use *ftshwlic* to display all the available license keys.

> If openFT is installed without a basic key then openFT runs as a demo version with full functionality for 30 days. This demo version may only be used for evaluation purposes!

Format

ftshwlic -h |

[-csv]

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-csv

The data is output in CSV format. The individual values are separated by semi-colons.

Messages of the ftshwlic command

Starting from version 12.1C80 *ftshwlic* command can display how many days left until openFT demo expires when there is no license added.

Example

ftshwlic: Your evaluation version of openFT will expire in '28' days.

3.62.1 Output format of ftshwlic

ftshwlic outputs information on all the installed license keys.

Example

```
ftshwlic
Type    Serial No. Performance class

SERVER 001234        0-24 CPUs
FTAM    000032        0-24 CPUs
FTP     000029        0-24 CPUs
```

Explanation:

Type

License type. The following standard types are possible:

SERVER

Basic key for the openFT server- product

FTAM

Optional key for the openFT FTAM component

FTP

Optional key for the openFT FTP component

The first key (here SERVER) is always the basic key. All further keys are optional. In the case of customers with special licenses, a number may also be output for *Type*. In the case of special applications or products, there may be additional types of license keys.

Serial No.

Serial number

Performance class

Range for the maximum permitted number of CPUs. Optional keys may also have a higher performance class than the basic key.

unlimited

The number of CPUs is unlimited.

3.63 ftshwm

Note on usage

Function: Display monitoring values of openFT operation

User group: FT user and FT administrator

Functional description

The *ftshwm* command allows you to output the current monitoring values from openFT operation. In order to do this, the FT administrator must have activated monitoring (*ftmodo -mon=n* command) and the asynchronous openFT server must be running.

Format

ftshwm -h |

[-ty]
[-raw]
[-po=<polling interval 1..600> [-pnr=<polling number 1..3600>]]
[-csv]
[<name 1..12> [... <name(100) 1..12>]| @a]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-ty

The types and scaling factors are to be output in place of the monitoring values and metadata.

The metadata type can be *TIME (timestamp) or *STRING (text output of the chosen selection).

A monitoring value can have one of the following types:

INT, BOOL or PERCENT (integer, on/off value or percentage). In the case of integer values, the scaling factor may be specified in brackets: INT(<scaling factor>).

The scaling factor of a monitoring value is only significant for output in CSV format. In this case, it is the number by which the value shown must be divided in order to obtain the real value.

-raw must not be specified at the same time.

-raw

Monitoring values are to be output as unedited raw data. This option is intended to be used in conjunction with external programs for further processing. The option must not be specified in conjunction with *-ty*. Monitoring values of the object *Duration* are not output.

If the specification is not used, the data is output in print-edited form.

The section “Description of the monitoring values” contains a table with notes that show what values are output when the *-raw* option is specified or is not specified and how the values are to be interpreted depending on this option.

-po=polling interval

Data is to be output initially after the specified polling interval in seconds has elapsed and then repeated at this interval.

If you also specify *-pnr*, you can limit the number of times the data is output. If you specify *-po* without *-pnr*, output is repeated an unlimited number of times.

If repeated output has been started with the *-po* option (with or without *-pnr*), it can be cancelled by an interrupt signal. Output is also cancelled in the event of an error, when the asynchronous openFT is terminated, or when monitoring is terminated.

Possible values: 1 through 600.

-po not specified

The monitoring values are output immediately and once only.

-pnr=polling number

-pnr specifies the number of times data is output. *-pnr* can only be specified in conjunction with *-po*.

Possible values: 1 through 3600.

-csv

The information is to be output in CSV format. First, the short names of the monitoring values are output in one row as the field names. This is followed by a row containing the monitoring values or their types and scaling factors as decimal numbers.

You can limit the scope of the output by specifying individual monitoring values that are significant for you.

name [name ...] | **@a**

The specified monitoring value or, if *-ty* is specified, the type and scaling factor associated with the named value is to be output.

name must be one of the short names of the monitoring values as they appear in the CSV header. You can specify up to 100 names separated by blanks.

@a for *name*

All openFT monitoring values or the types and scaling factors of all openFT monitoring values are to be output.

name not specified

A predefined default set of monitoring values is output (see the section “Description of the monitoring values”).

3.63.1 Description of the monitoring values

The table below shows all the monitoring values output with the option @*a*. You can instead specify a list of any of the monitoring values shown in the table.

You can use the openFT Monitor to display the monitoring values for openFT operation. You call the openFT Monitor by means of the *ftmonitor* command.

Note for Windows systems:

> You can output the listed monitoring values via the Windows Performance Monitor, see section "Outputting monitoring values using the Windows Performance Monitor".

The first two letters of the name indicate the data object that the monitoring value belongs to:

- Th = Throughput
- Du = Duration
- St = State

The second component of the name indicates the performance indicator, e.g. *Netb* for net bytes. In the case of monitoring values for the *Throughput* or *Duration* data object, the last 3 letters of the name indicate the types of requests from which the monitoring value originates, e.g.

- Ttl = FT Total
- Snd = FT Send requests
- Rcv = FT Receive requests
- Txt = Transfer of text files
- Bin = Transfer of binary files
- Out = FT Outbound
- Inb = FT Inbound

> If monitoring is deactivated for all partners (*ftmodo -monp=*), only the following values are populated:
>
> Status: StCLim, StCAct, StRqLim, StRqAct, StOftr, StFtmr, StFtpr, StTrcr
>
> All the other values are set to 0.

Name	Meaning	Output prepared (formatted)	Output not prepared (raw)
ThNetbTtl	Throughput in net bytes: Number of bytes transferred	Number of bytes per second	Bytes, accumulated
ThNetbSnd	Throughput in net bytes (send requests): Number of bytes transferred with send requests7	Number of bytes per second	Bytes, accumulated

ThNetbRcv	Throughput in net bytes (receive requests): Number of bytes transferred with receive requests	Number of bytes per second	Bytes, accumulated
ThNetbTxt [1)]	Throughput in net bytes (text files): Number of bytes transferred when transferring text files	Number of bytes per second	Bytes, accumulated
ThNetbBin [1)]	Throughput in net bytes (binary files): Number of bytes transferred when transferring binary files	Number of bytes per second	Bytes, accumulated
ThDiskTtl	Throughput in disk bytes: Number of bytes read from files or written to files with transfer requests	Number of bytes per second	Bytes, accumulated
ThDiskSnd	Throughput in disk bytes (send requests): Number of bytes read from files with send requests	Number of bytes per second	Bytes, accumulated
ThDiskRcv	Throughput in disk bytes (receive requests): Number of bytes written to files with receive requests	Number of bytes per second	Bytes, accumulated
ThDiskTxt [1)]	Throughput in disk bytes (text files): Number of bytes read from text files or written to text files with transfer requests	Number of bytes per second	Bytes, accumulated
ThDiskBin [1)]	Throughput in disk bytes (binary files): Number of bytes read from binary files or written to binary files with transfer requests	Number of bytes per second	Bytes, accumulated
ThRqto	openFT requests: Number of openFT requests received	Number per second	Number, accumulated
ThRqft [1)]	File transfer requests: Number of file transfer requests received	Number per second	Number, accumulated
ThRqfm [1)]	File management requests: Number of file management requests received	Number per second	Number, accumulated
ThSuct	Successful requests: Number of successfully completed openFT requests	Number per second	Number, accumulated
ThAbrt	Aborted requests: Number of aborted openFT requests	Number per second	Number, accumulated
ThIntr	Interrupted requests: Number of interrupted openFT requests	Number per second	Number, accumulated
ThUsrf	Requests from non-authorized users: Number of openFT requests in which the user check was terminated with errors	Number per second	Number, accumulated

ThFoll [1)]	Follow-up processing operations started: Number of followup processing operations started	Number per second	Number, accumulated
ThCosu [1)]	Connections established: Number of connections successfully established	Number per second	Number, accumulated
ThCofl	Failed connection attempts: Number of attempts to establish a connection that failed with errors	Number persecond	Number, accumulated
ThCobr	Disconnections: Number of disconnections as a result of connection errors	Number per second	Number, accumulated
DuRqtlOut [1)]	Maximum request duration Outbound: Maximum request duration of an outbound request	Milliseconds [2)]	-
DuRqtlInb [1)]	Maximum request duration Inbound: Maximum request duration of an inbound request	Milliseconds [2)]	-
DuRqftOut [1)]	Maximum request duration Outbound transfer: Maximum duration of an outbound file transfer request	Milliseconds [2)]	-
DuRqftInb [1)]	Maximum request duration Inbound transfer: Maximum duration of an inbound file transfer request	Milliseconds [2)]	-
DuRqfmOut [1)]	Maximum request duration Outbound file management: Maximum duration of an outbound file management request	Milliseconds [2)]	-
DuRqfmInb [1)]	Maximum request duration Inbound file management: Maximum duration of an inbound file management request	Milliseconds [2)]	-
DuRqesOut [1)]	Maximum outbound request waiting time: Maximum waiting time before an outbound request is processed (for requests without a specific start time)	Milliseconds [2)]	-
DuDnscOut [1)]	Maximum duration of an outbound DNS request: Maximum time an outbound openFT request was waiting for partner checking	Milliseconds [2))]	-
DuDnscInb [1)]	Maximum duration of an inbound DNS request: Maximum time an inbound openFT request was waiting for partner checking	Milliseconds [2)]	-
DuConnOut [1)]	Maximum duration of establishment of a connection: Maximum time between requesting a connection and receiving confirmation of a connection for an outbound openFT request	Milliseconds [2)]	-

DuOpenOut [1)]	Maximum file open time (outbound): Maximum time an outbound openFT request required to open the local file	Milliseconds [2)]	-
DuOpenInb [1)]	Maximum file open time (inbound): Maximum time an inbound openFT request required to open the local file	Milliseconds [2)]	-
DuClosOut [1)]	Maximum file close time (outbound): Maximum time an outbound openFT request required to close the local file	Milliseconds [2)]	-
DuClosInb [1)]	Maximum file close time (inbound): Maximum time an inbound openFT request required to close the local file	Milliseconds [2)]	-
DuUsrcOut [1)]	Maximum user check time (outbound): Maximum time an outbound openFT request required to check the user ID and transfer admission	Milliseconds [2)]	-
DuUsrcInb [1)]	Maximum user check time (inbound): Maximum time an inbound openFT request required to check the user ID and transfer admission	Milliseconds [2)]	-
StRqas	Number of synchronous requests in the ACTIVE state	Average value [3)]	Current number
StRqaa	Number of asynchronous requests in the ACTIVE state	Average value [3)]	Current number
StRqwt	Number of requests in the WAIT state	Average value [3)]	Current number
StRqhd	Number of requests in the HOLD state	Average value [3)]	Current number
StRqsp	Number of requests in the SUSPEND state	Average value [3)]	Current number
StRqlk	Number of requests in the LOCKED state	Average value [3)]	Current number
StRqfi [1)]	Number of requests in the FINISHED state	Average value [3)]	Current number
StCLim	Maximum number of connections: Upper limit for the number of connections established for asynchronous requests.	Value currently set	
StCAct	Number of occupied connections for asynchronous requests	Share of StCLim in % [4)]	Current number
StRqLim	Maximum number of requests: Maximum number of asynchronous requests in request management	Value currently set	
StRqAct	Entries occupied in request management	Share of StRqLim in % [4)]	Current number
StOftr	openFT Protocol activated/deactivated	ON (activated), OFF (deactivated)	
StFtmr	FTAM protocol activated/deactivated	ON (activated), OFF (deactivated)	

StFtpr	FTP protocol activated/deactivated	ON (activated), OFF (deactivated)
StTrcr [1)]	Trace activated/deactivated	ON (activated), OFF (deactivated)

[1)]Output only if @a is specified.

[2)]Maximum value of the monitoring interval (= time elapsed since the last time the monitoring values were queried or since the start of monitoring).

[3)]Average value of the monitoring interval (= time elapsed since the last time the monitoring values were queried or since the start of monitoring). Format: n.mm, where n is an integer and mm are to be interpreted as decimal places.

[4)]If the reference value is reduced in live operation, it is possible for the value output to lie above 100 (%) temporarily.

Outputting monitoring values using the Windows Performance Monitor

Before you can output the monitoring values via the Windows Performance Monitor, you must first configure the Windows Performance Monitor accordingly, see section on recording monitoring values in the manual "openFT (Unix and Windows systems) - Installation and Operation".

The monitoring values that are output using the Windows Performance Monitor are sometimes scaled differently:

- All the specifications in the table that output the *Number of bytes per second* (*ThNetbTtl* to *ThDiskBin*) are displayed in millions of bytes per second in the Windows Performance Monitor, i.e. 1 byte/sec corresponds to the value 0.000001.
- The value for StRqLim (maximum request number) is displayed in units of 1000 in the Windows Performance Monitor, i.e. StRqLim=1 corresponds to the value 0.001.

Example

```
ftshwm
openFT(std)   Monitoring (formatted)
MonOn=2017-02-17 15:36:12 PartnerSel=OPENFT  RequestSel=ONLY-ASYNC,ONLY-LOCAL
2017-02-17 15:40:01
 Name      Value
 ----------------
 ThNetbTtl 38728
 ThNetbSnd 38728
 ThNetbRcv 0
 ThDiskTtl 16384
 ThDiskSnd 16384
 ThDiskRcv 0
 ThRqto    1
 ThSuct    0
 ThAbrt    0
 ThIntr    0
 ThUsrf    0
 ThCofl    0
 ThCobr    0
 StRqas    0.00
 StRqaa    8.66
 StRqwt    1.66
 StRqhd    0.00
 StRqsp    0.00
 StRqlk    0.00
 StCLim    16
 StCAct    37
 StRqLim   1000
```

```
StRqAct   1
StOftr    ON
StFtmr    OFF
StFtpr    OFF
```

Explanation of output:

The default output format begins with a header containing the following specifications:

- Name of the openFT instance and selected data format (*raw* or *formatted*)
- Monitoring start time and partner and request selection
- Current timestamp

This is followed by the list of default values.

3.64 ftshwo

Note on usage

Function: Display operating parameters

User group: FT user and FT administrator

Functional description

The *ftshwo* command outputs the operating parameters of the local openFT system. Alongside the standard output and output in CSV format, output may also be specified as a platform-specific command sequence. In this way, it is possible to save the settings and then load them onto another computer with the selected operating system.

The FT administrator can set or modify the operating parameters with the *ftmodo* command.

> The transfer admission of the ADM trap server is not output with the default output format and CSV output format. It only appears as a command sequence in the output (*-px*, *-pw*, *-p2*, *-pz*) for the FT administrator.

Format

ftshwo -h |

[-csv | -px | -pw | -p2 | -pz | -ae | -x25]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-csv

The operating parameters are output in CSV format. The individual values are separated by semicolons.

-px

The operating parameters are output as a command string. This can be called as a shell procedure on Unix systems in order to regenerate the identical operating parameters.

-pw

The operating parameters are output as a command string. This can be called as a batch procedure on Windows systems in order to regenerate the identical operating parameters.

-p2

The operating parameters are output as a command string. This can be called as an SDF procedure on BS2000 systems in order to regenerate the identical operating parameters.

-pz

The operating parameters are output as a command string. This can be called as a Clist procedure on z/OS systems in order to regenerate the identical operating parameters.

-ae

The operating parameters of the Application Entity Titles (AET) are output.

-x25

The operating parameters of the FarSync X.25 transport system are output.

No option specified

The operating parameters are output in standard format.

3.64.1 Output format of ftshwo

- Standard output format
- Output format for X.25
- Output format for AET

3.64.1.1 Standard output format

Example for Unix systems

```
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 FT-DIR-LOG ADM-LOG     USE TNS USE CMX
  STD       ON   B-P-ATTR    ALL    ALL       FAIL     ALL          NO      NO
OPENFT-APPL      FTAM-APPL         FTP-PORT           ADM-PORT        ADM-CS
*STD             *STD              21                 11000             NO
ACTIVE           ACTIVE            ACTIVE             ACTIVE
RSA-PROP RSA-MIN AES-MIN ENC-MAND
2048         0     NONE     NO
HOST-NAME        IDENTIFICATION / LOCAL SYSTEM NAME
*NONE            mc011.mynet.local / $FJAM,MC011
FT-ADMIN         FTAC-ADMIN       FT-ADMIN-GROUP
ROOT             *STD             *NONE
FN-CSS-NAME    DEL-LOG  ON  AT     RETPD    RECOVERY   ADM-TRAP-SERVER
  ISO88591      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    OPTIONS-LL
MONITOR  OFF   ALL                  ALL
TRACE    OFF   ALL                  ALL                  NONE       OFF
```

Example for Linux systems with FT administrator group

```
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 FT-DIR-LOG ADM-LOG     USE TNS USE CMX
  STD       ON   B-P-ATTR    ALL    ALL       FAIL     ALL          NO      NO
OPENFT-APPL      FTAM-APPL         FTP-PORT           ADM-PORT        ADM-CS
*STD             *STD              21                 11000             NO
ACTIVE           ACTIVE            ACTIVE             ACTIVE
RSA-PROP RSA-MIN AES-MIN ENC-MAND
2048         0     NONE     NO
HOST-NAME        IDENTIFICATION / LOCAL SYSTEM NAME
*NONE            mc011.mynet.local / $FJAM,MC011
FT-ADMIN         FTAC-ADMIN       FT-ADMIN-GROUP
*NONE            *STD             ftgroup
FN-CSS-NAME    DEL-LOG  ON  AT     RETPD    RECOVERY   ADM-TRAP-SERVER
  ISO88591      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    OPTIONS-LL
MONITOR  OFF   ALL                  ALL
TRACE    OFF   ALL                  ALL                  NONE       OFF
```

Example for Windows systems

```
ftshwo
STARTED PROC-LIM CONN-LIM ADM-CLIM RQ-LIM MAX-RQ-LIFE TU-SIZE  CCS-NAME
  YES       2       16       8      2000      30       65535    CP1252
PTN-CHK DYN-PART SEC-LEV  FTAC-LOG FT-LOG FT-DIR-LOG ADM-LOG     USE TNS USE CMX
  STD       ON   B-P-ATTR    ALL    ALL     FAIL       ALL          NO      NO
OPENFT-APPL      FTAM-APPL         FTP-PORT           ADM-PORT        ADM-CS
*STD             *STD              21                 11000             NO
ACTIVE           ACTIVE            ACTIVE             ACTIVE
RSA-PROP RSA-MIN AES-MIN ENC-MAND
2048        0     NONE     NO
HOST-NAME        IDENTIFICATION / LOCAL SYSTEM NAME
*NONE            mc011.mynet.local / $FJAM,MC011
FT-ADMIN         FTAC-ADMIN       FT-ADMIN-GROUP
SYSTEM           *STD             *NONE
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    OPTIONS-LL
MONITOR  OFF   ALL                  ALL
TRACE    OFF   ALL                  ALL                  NONE       OFF
```

Meaning of the output together with the associated command options:

Field name	**Meaning and values**	**Command/ option**
STARTED	Specifies whether the asynchronous openFT server has started (YES) or not (NO).	*ftstart* *ftstop*
PROC-LIM	Maximum number of openFT servers available for the processing of asynchronous requests.	*ftmodo -pl=*
CONN-LIM	Maximum number of asynchronous requests that can be processed simultaneously.	*ftmodo -cl=*
ADM-CLIM	Maximum number of asynchronous administration requests including ADM traps that can be processed simultaneously.	*ftmodo -admcl=*
RQ-LIM	Maximum number of file transfer requests that can simultaneously be present in the local system's request queue.	*ftmodo -rql=*
MAX-RQ-LIFE	Maximum lifetime of requests in the request queue (in days).	*ftmodo -rqt=*
TU-SIZE	Upper limit for message length at transport level (in bytes).	*ftmodo -tu=*
CCS-NAME	Name of the character set used by default for file transfer requests.	*ftmodo -ccs=*

PTN-CHK	Setting for sender verification: ADDR: addressSTD: identification AET: Application Entity Title (for FTAM partner) T+AE: AET + address	*ftmodo -ptc=*
DYN-PART	Setting for dynamic partner entries: ON (activated) OFF (deactivated)	*ftmodo -dp=*
SEC-LEV	Default security level for partners in the partner list for which no security level has been set:	*ftmodo -sl=*
	1..100: Fixed security level. 1 is the lowest and 100 the highest security level.	
	B-P-ATTR: The security level is depending on the partner's attributes, i.e.: 10 if the partner has been authenticated. 90 if the partner is known in the transport system. 100 otherwise, i.e. if the partner has only been identified by its address.	*ftmodo -sl=p*
FTAC-LOG	Scope of FTAC logging:	*ftmodo -lc=*
	ALL: All FTAC access checks	
	MODIFY: Modifying file management requests and rejected FTAC access checks	
	REJECTED: Only rejected FTAC access checks	
FT-LOG	Scope of FT logging:	*ftmodo -lt=*
	ALL: All requests	
	FAIL: Only errored FT requests	
	NONE: FT Logging deactivated	
FT-DIR-LOG	Scope of logging of directory transfer:	*ftmodo -ltd=*
	ALL: All invidual requests	
	FAIL: Only errored invidual requests	
	NONE: No invidual requests	
ADM-LOG	Scope of ADM logging:	*ftmodo -la=*
	ALL: All requests	

	FAIL: Only errored ADM requests	
	MODIFY: only modifying ADM requests	
	NONE: ADM Logging deactivated	
USE TNS	Specifies whether the TNS is to be used (YES) or not (NO) during operation with CMX	*ftmodo -tns=*
USE CMX	Specifies whether operation with CMX is activate (YES) or not (NO)	*ftmodo -cmx=*
OPENFT-APPL	Port number of the local openFT server, possibly extended by the transport selector. *STD means that the default value is used i.e. 1100 and $FJAM in Transdata format (EBCDIC, 8 characters long, padded with blanks).	*ftmodo -openft=*
	Line 2: ACTIVE: openFT protocol activated DISABLED: openFT protocol (inbound) deactivated INACT: openFT protocol (inbound) not available NAVAIL: openFT protocol not licensed (only on Windows systems)	*ftmodo -acta=*
FTAM-APPL	Port number of the local FTAM server, possibly extended by the transport selector, the session selector and the presentation selector. *STD means that the default value is used i.e. 4800 and $FTAM in Transdata format (EBCDIC, 8 characters long, padded with blanks)	*ftmodo -ftam=*
	Line 2: ACTIVE: FTAM protocol activated DISABLED: FTAM protocol (inbound) deactivated INACT: FTAM protocol (inbound) not available NAVAIL: FTAM not installed (Unix systems) NAVAIL: FTAM protocol not licensed (Windows systems)	*ftmodo -acta=*
FTP-PORT	Port number used by local FTP server. Default port: 21	*ftmodo -ftp=*
	Line 2: ACTIVE: FTP protocol activated DISABLED: FTP protocol (inbound) deactivated INACT: FTP protocol (inbound) not available NAVAIL: FTP not installed (Unix systems) NAVAIL: FTP protocol not licensed (Windows systems)	*ftmodo -acta=*

ADM-PORT	Port number used by remote administration. Default port: 11000	*ftmodo -adm=*
	Line 2: ACTIVE: remote administration activated DISABLED: remote administration (inbound) deactivated INACT: remote administration (inbound) not available	*ftmodo -acta=*
ADM-CS	Specifies whether the local openFT instance is flagged as a remote administration server (YES) or not (NO).	*ftmodo -admcs=*
RSA-PROP	RSA key length to encrypt the AES/DES key. Values: 0 | 768 | 1024 | 2048 | 3072 | 4096.	*ftmodo -kl=*
RSA-MIN	RSA minimum key length. Values: 0 | 768 | 1024 | 2048 | 3072 | 4096.	*ftmodo -klmin=*
AES-MIN	AES minimum key length. Values: NONE | 128 | 256.	*ftmodo -aesmin=*
ENC-MAND	Specifies whether the inbound and/or outbound encryption is activated (YES) or not (NO).	*ftmodo -c=*
HOST-NAME	Host name of the local computer, *NONE means that no host name has been assigned.	*ftcrei -addr=* *ftmodi -addr=*
IDENTIFICATION	Instance identification of the local openFT instance.	*ftmodo -id=*
FT-ADMIN	Name of the FT administrator or SYSTEM (on Windows systems) or ROOT (on Unix systems).	*ftmodo -admpriv=*
FTAC-ADMIN	Name of the FTAC administrator or *STD (only on Windows systems).	*ftmoda ... -priv=*
FT-ADMIN-GROUP	Name of the FT administrator group (Linux systems)	*ftmodo -gadmpriv=*
LOCAL-SYSTEM-NAME	Name of the local system.	*ftmodo -p= -l=*
DEL-LOG	Automatic deletion of log records activated (ON) or deactivated (OFF)	*ftmodo -ld=*
FN-CCS-NAME	Character set used for displaying file names in the case of inbound requests in character mode (only on Unix systems).	*ftmodo -fnccs=*
ON	Day on which the log records are to be deleted: MON, TUE, ... SUN (day of the week) or 1...31 (day of the month) or DAILY (every day)	*ftmodo -ldd=*

AT	Time at which the log records are to be deleted (hh:mm)	*ftmodo -ldt=*
RETPD	Minimum age of log records for deletion in days. 0 means the current day.	*ftmodo -lda=*
RECOVERY	Activate and deactivate the restart function for inbound and outbound requests: IN+OUT: restart function is activated for inbound as well as for outbound requests. IN: restart function is only activated for inbound requests OUT: restart function is only activated for outbound requests NO: restart function is deactivated for inbound as well as for outbound requests.	
ADM-TRAP-SERVER	Name or address of the partner to which the ADM traps are sent. *NONE means that the sending of ADM traps is deactivated.	*ftmodo -atpsv=*
TRAP	The TRAP settings are output here. The possible values are ON and OFF. The row CONS indicates the console traps and the row ADM the ADM traps. The columns designate the events for which traps may be generated:	*ftmodo -tpc= -atp=*
	SS-STATE: Change of the status of the openFT subsystem (row CONS only)	
	FT-STATE: Change of the status of the asynchronous server	
	PART-STATE: Change of the status of partner systems	
	PART-UNREA: Partner systems unreachable	
	RQ-STATE: Change of the status of request administration	
	TRANS-SUCCRequests completed successfully	
	TRANS-FAIL: Failed requests	
FUNCT	The settings for monitoring (MONITOR row) and tracing (TRACE row) are output in this section. The individual columns have the following meanings:	
	SWITCH: Function (monitoring or tracing) activated (ON) or deactivated (OFF)	*ftmodo -mon= -tr=*

	PARTNER-SELECTION: Selection based on the partner system's protocol type. Possible protocol types: OPENFT, FTP, FTAM. ADM (administration partner) can also be output under TRACE. ALL means that all protocol types have been selected, i.e. tracing/monitoring is possible for all partner systems. NONE means that no protocol type has been selected.	*ftmodo -monp= -trp=*
	REQUEST-SELECTION: Selection based on the request type. The following are possible: ONLY-SYNC/ONLY-ASYNC (only synchronous or only asynchronous requests) ONLY-LOCAL/ONLY-REMOTE (only locally or only remotely submitted requests). ALL means no restriction, i.e. all requests.	*ftmodo -monr= -trr=*
	OPTIONS (only in the TRACE row)NONE means no options (trace in default format) NO-BULK-DATA means minimum trace, i.e. bulk data (file contents) is not logged. In addition, no repetitions of data log elements are logged.	*ftmodo -tro=*
	OPTIONS-LL Scope of tracing for lower protocol layers: OFF: Deactivated STD: Default DETAIL: Details	*ftmodo -troll=*

3.64.1.2 Output format for X.25

Example for Windows systems

```
ftshwo -x25

ADAPTER LINE DTE

0 0 12345

0 1 54321

1 0 22222

1 1 33333

OPENFT-APPL

USE X.25 NUM-LISTS CLASS ADAPTER

  NO         3       0/-  0,1
NSAP = 430000000012345678901

       AFI = 43
       IDI = 123
       DSP = 45678901

FTAM-APPL

USE X.25 NUM-LISTS CLASS ADAPTER

  YES        4       2/0  1
NSAP = 430000000032110987654

       AFI = 43
       IDI = 321
       DSP = 10987654
```

Example for a Linux system

```
ftshwo -x25
ADAPTER LINE DTE
0 12345
0 54321
1 22222
1 33333
OPENFT-APPL
USE X.25 NUM-LISTS CLASS ADAPTER
  NO         3       0/-  0,1
NSAP = 4300000000012345678901
       AFI = 43
       IDI = 123
       DSP = 45678901
FTAM-APPL
USE X.25 NUM-LISTS CLASS ADAPTER
  YES        4       2/2  2
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

(Different for Windows and Linux)

DTE address which is allocated to the line. A line is uniquely defined via the combination of adapter number and line number under Windows and via the adapter number under Linux.

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 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, then it is 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 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 an OSI network address, then it is 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.

3.64.1.3 Output format for AET

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

Explanation

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*).

AETitle

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 (Directory form/transparent)

The Calling AET is defined in the transparent format

AETitle format 2 (numeric)

The Calling AET is defined in the numeric format

Application Process Title and Application Entity Qualifier are optional outputs.

3.65 ftshwp

Note on usage

Function: Display FT profiles

User group: FTAC user and FTAC administrator

Functional description

ftshwp stands for "show profile" and allows you to obtain information about FT profiles. In short form, it displays the names of the selected FT profiles, as well as the following information:

- whether or not the FT profile is privileged: asterisk (*) before the profile name
- whether or not the transfer admission is disabled: exclamation mark (!) before the profile name.

You can only obtain information about your own FT profiles.

As the ADM administrator, you may also obtain information about ADM profiles (i.e. FT profiles with the property "access to remote administration server").

As the FTAC administrator, you may obtain information about all FT profiles in the system.

Format

ftshwp -h |

[<profile name 1..8> | @s]
[-s=[<transfer admission> | @a | @n]

[,<user ID> | @a | @adm]]

[-l][-csv]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

profile name | @**s**

Is the name of the FT profile you wish to see.

@**s** for *profile name*

Provides information on the standard admission profile for the user ID if this has been set up. Otherwise you see a corresponding message.

profile name not specified

Profile name is not used as a criterion for selecting the FT profile to be displayed. If you do not specify the profile with -s (see below), FTAC will display information on all of your FT profiles.

-s=[transfer admission | @**a** | @**n**][**,**user ID | @**a**]

-s is used to specify criteria for selecting the FT profiles to be displayed.

If you wish to view standard admission profile, you can only specify @*n* or @*a*.

Transfer admission

Is the transfer admission of the FT profile to be displayed. A binary transfer admission must be specified in hexadecimal format, see section “Entering commands”.

@**a** for *transfer admission*

Displays information either on the FT profile specified with *profile name* (see above) or (if no *profile name* was specified) on all of your FT profiles.

As the FTAC administrator, you can specify *@a* if you want to obtain information on FT profiles belonging to other login names, since even you should not know the transfer admission.

@**n** for *transfer admission*

displays information on FT profiles that do not have a defined transfer admission.

As the FTAC administrator, you can specify *@n* if you want to obtain information on FT profiles belonging to other login names which do not have a defined transfer admission.

transfer admission not specified

causes FTAC to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission. If you just press <ENTER>, this has the same effect as specifying *@a*.

,user ID

must be your own login name if you are a normal user.

As the FTAC administrator, you can specify any login name here.

@**a** for *user ID*

allows you to display only profiles belonging to your own login name.

As the FTAC administrator, you can obtain information on the FT profiles of all login names.

As the ADM administrator, you can obtain information on the own FT profiles and the ADM profiles.

@**adm** for *user ID*

For the FTAC and ADM administrator only.

As the FTAC or ADM administrator, you obtain information on ADM profiles.

user ID not specified

displays only profiles belonging to the user's own login name, regardless of who issues the command.

-*s* not specified

if no profile name is specified, displays all the FT profiles belonging to the login name under which the *ftshwp* command is issued. Otherwise, displays information on the FT profile with the specified name.

-l

displays the contents of the selected FT profiles.

In long form, the entire contents of the selected FT profiles are displayed. The USER-ADM parameter contains the following information:

- the login name for which an admission profile is valid or if it is an ADM profile
- whether or not it is valid for a specific password of the login name
- whether or not it is valid for any password of the login name
- whether or not it has an undefined password and is thus disabled.

Please note that ADM profiles always are indicated by the value *ADM under the USER-ADM parameter.

USER-ADM=	**Meaning**
(user ID,,OWN)	Profile is valid for all passwords of the login name.
(user ID,,YES)	The profile is valid only for a specific password of the login name (specified in *-ua=user ID, password* with an *ftcrep* or *ftmodp* command). The profile is deactivated (not disabled) if the password is changed. You can activate it again, for example, by resetting the password.
(user ID,, NOT-SPECIFIED)	The FTAC administrator created or modified the FT profile knowing only the login name. As a result, the profile was disabled. You must enable the profile with *ftmodp* and the *-v=y* parameter.

If an FT profile is disabled, the *TRANS-ADM* parameter indicates the reasons why the profile was disabled. The following table shows the possible parameter values, as well as their meanings:

TRANS-ADM=	**Possible cause and action**
NOT-SPECIFIED	The FTAC administrator created the FT profile without transfer admission, or the FTAC user did not specify transfer admission. Measure: specify transfer admission
DUPLICATED	An attempt was made to create an FT profile with the same transfer admission. Measure: specify new transfer admission
LOCKED (by_adm)	The FTAC administrator modified the FT profile by login name only. The transfer admission remained unchanged but was disabled. Measure: enable the profile using the *ftmodp* command and the *-v=y* parameter
LOCKED (by_import)	The FT profile was created using the *ftimpe* command. The transfer admission remains unchanged, but is marked as disabled. Measure: enable the profile using the *ftmodp* command and the *-v=y* parameter.

LOCKED (by_user)	The FTAC user disabled his/her own FT profile. Measure: enable profile using the *ftmodp* command and the *-v=y* parameter.
EXPIRED	The time up to which the transfer admission can be used has expired. Measure: enable profile using the *ftmodp* command and the *-d* parameter, by removing the temporal restriction using the *-d* entry and defining a new time span with *-d=date*.

ftshwp does not provide a means of displaying a transfer admission. If you have forgotten a transfer admission, you have to define a new one using *ftmodp*.

-l not specified

displays only the names of your FT profiles. Markings also indicate whether or not an FT profile is privileged (*) and whether or not it is disabled (!).

-csv

You can use *-csv* to specify that the FT profiles are to be output in the CSV format. The values in the output are separated by semicolons. If *-csv* is specified, output is always in long form (analogous to *-l*) regardless of whether or not *-l* has also been specified.

-csv not specified

The FT profiles are output in the standard format, i.e. in abbreviated form if *-l* is not specified and in detailed form if *-l* is specified.

Examples

1. Scrooge McDuck wishes to see the FT profile *goldmrep* under his login name. This profile was created in the command *ftcrep* (see Examples).

   ```
   ftshwp goldmrep -l
   ```

   The output is as follows:

   Unix systems:

   ```
   goldmrep
    EXP-DATE    = 20173112
    TRANS-DIR   = FROM
    PARTNER     = goldmine
    FILE-NAME   = monthlyreport_goldmine01
    WRITE       = REPLACE-FILE
    USER-ADM    = (scrooge,,OWN)
    FT-FUNCTION = (TRANSFER-FILE, FILE-PROCESSING)
    SUCC-PROC   = 'lpr monthlyreport_goldmine01'
    FAIL-PROC   = NONE
    DATA-ENC    = YES
    FILE-AT-ENC = YES
    LAST-MODIF  = 2016-03-27 14:55:23
   ```

   Windows systems:

```
goldmrep
  EXP-DATE     = 20173112
  TRANS-DIR    = FROM
  PARTNER      = goldmine
  FILE-NAME    = monthlyreport_goldmine01
  WRITE        = REPLACE-FILE
  USER-ADM     = (scrooge,,OWN)
  FT-FUNCTION = (TRANSFER-FILE, FILE-PROCESSING)
  SUCC-PROC    = 'lpr monthlyreport_goldmine01'
  FAIL-PROC    = NONE
  DATA-ENC     = YES
  FILE-AT-ENC = YES
  LAST-MODIF   = 2016-03-27 14:55:23
```

The timestamp of the most recent change is shown under LAST-MODIF.

If you specify *ftmodp goldmrep* without any further parameters, you can force the timestamp to be updated without changing the profile properties.

2. Scrooge McDuck wishes to see the standard FT profile:

```
ftshwp @s -l
*STD
 TRANS-ADM   = (NOT-SPECIFIED)
```

```
WRITE       = NEW-FILE
USER-ADM    = (scrooge,,OWN)
FT-FUNCTION = (TRANSFER-FILE)
LAST-MODIF  = 2016-03-22 16:06:55
```

3. You are an FTAC administrator and want to view all the standard admission profiles on

   your system.

   `ftshwp @s -s=@n,@a -l`

   Output takes the following form:

```
*STD
 TRANS-ADM   = (NOT-SPECIFIED)
 USER-ADM    = (john,,OWN)
 FT-FUNCTION = (TRANSFER-FILE, MODIFY-FILE-ATTRIBUTES, READ-FILE-
DIRECTORY)
 LAST-MODIF  = 2016-03-23 17:12:25
*STD
 TRANS-ADM   = (NOT-SPECIFIED)
 WRITE       = NEW-FILE
 USER-ADM    = (dagobert,,OWN)
 FT-FUNCTION = (TRANSFER-FILE)
 LAST-MODIF  = 2016-03-22 16:06:55
```

4. You are the FT administrator and wish to view the profile *acctrap1* on the ADM trap server.

   ```
   ftshwp acctrap1 -l
   ```

   Output takes the following form:

   ```
   acctrap1
    USER-ADM    = (ADMIN002,,OWN)
    FT-FUNCTION = (ADM-TRAP-LOG)
    LAST-MODIF  = 2016-01-23 18:24:42
   ```

   The value ADM-TRAP-LOG under FT-FUNCTION in the *acctrap1* profile means that the remote administration server can receive ADM traps with this profile.

5. You are the ADM administrator and wish to view the ADM profiles on the remote administration server.

   ```
   ftshwp -s=@a,@adm -l
   ```

   Output takes the following form:

   ```
   acccentr
    USER-ADM    = (*ADM,,OWN)
    FT-FUNCTION = (ACCESS-TO-ADMINISTRATION)
    LAST-MODIF  = 2016-01-23 18:21:08
   ```

   The profile *acccentr* is a ADM profile. This is indicated by the value ACCESS-TO-ADMINISTRATION under FT-FUNCTION and the value *ADM for user ID under USER-ADM.

6. You are the FT administrator and would like to view the profile

   remadmin that has been set up for remote administration.

   ```
   ftshwp remadmin -l
   ```

   Output takes the following form:

   ```
   remadmin
    USER-ADM    = (ADMIN001,,OWN)
    FT-FUNCTION = (REMOTE-ADMINISTRATION)
    LAST-MODIF  = 2016-02-27 16:20:38
   ```

3.66 ftshwptn

Note on usage

Function: Display partner properties

User group: FT user and FT administrator

Functional description

You use the *ftshwptn* command to call up the following information about the partner systems entered in the partner list:

- The name of the partner system
- The status of the partner system (activated, deactivated)
- The security level that was assigned to the partner system
- The priority that was assigned to the partner system
- The setting for the openFT trace function for the partner system
- The number of file transfer requests to the partner system issued in the local system that have not yet been completed
- The number of file transfer requests for the local system that have been issued in the partner system
- The mode for sender verification and authentication
- The partner system's transport address, possibly with the port number if this is different from the default
- The identification of the partner system
- The routing information if the partner system can only be accessed via an intermediate instance
- The RSA-PROPOSAL key of the partner system
- The RSA-MINIMUM key of the partner system

You can also output the partners in the partner list as a platform-specific command sequence. In this way, it is possible to save the partner list and load it at another computer which may possibly be running a different operating system.

Format

ftshwptn -h |

[<partner 1..200> | @a]
[-st=a | -st=na | -st=d | -st=ie | -st=nc | -st=ad | -st=da]
[-l | -csv | -px | -pw | -p2 | -pz | -pa]

[-kl= | -kl=FTOPT | -kl=0 | 768 | 1024 | 2048 | 3072 | 4096]
[-klmin= | -klmin=FTOPT | -klmin=0 | 768 | 1024 | 2048 | 3072 | 4096]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

partner | **@a**

Specifies the partner whose properties you want to display. You can specify the name of the partner in the partner list or the address of the partner system. For details in address specifications, see section “Specifying partner addresses”.

@a for *partner*

The properties of all the partners in the partner list are displayed.

partner not specified

The properties of all the partners in the partner list are displayed.

-st=a | **-st=na** | **-st=d** | **-st=ie** | **-st=nc** | **-st=ad** | **-st=da**

This operand enables you to display the properties of partner systems which have a specific status. You can specify the following values:

a (active)

All the partner systems with the status ACTIVE are displayed.

na (not active)

All the partner systems which do **not** have the status ACTIVE are displayed.

d (deactivated)

All the partner systems with the status DEACTIVE are displayed.

ie (installation error)

All the partner systems with the status LUNK, RUNK, LAUTH, RAUTH, NOKEY or IDREJ are displayed.

nc (not connected)

All the partner systems with the status NOCON or DIERR are displayed.

ad (active + automatic deactivation)

All the partner systems for which the option

AUTOMATIC-DEACTIVATION is set (see the option *-ad* in the *ftaddptn* and *ftmodptn* commands) but are still active are displayed.

da (deactivated + automatic deactivation)

All the partner systems which have actually been deactivated because of the AUTOMATIC-DEACTIVATION option are displayed.

-st not specified

The output is not restricted to partner systems with a specific status.

-l | **-csv** | **-px** | **-pw** | **-p2** | **-pz** | **-pa**

These options determine the scope and format of the output.

-l

The properties of the partner systems are output in full as a table.

-csv

The properties of the partner systems are output in CSV format. The individual values are separated by semicolons.

-px

The properties of the partner systems are output as a command sequence. This can be called in Unix systems as a shell procedure in order to generate partner entries with identical properties.

-pw

The properties of the partner systems are output as a command sequence. This can be called in Windows systems as a batch procedure in order to generate partner entries with identical properties.

-p2

The properties of the partner systems are output as a command sequence. This can be called in BS2000 systems as an SDF procedure in order to generate partner entries with identical properties.

-pz

The properties of the partner systems are output as a command sequence. This can be called in z/OS systems as a CLIST procedure in order to generate partner entries with identical properties.

-pa

Additional address-specific properties of an X.25 partner are output on Linux and Windows systems.

-l, *-csv*, *-px*, *-pw*, *-p2*, *-pz*, *-pa* not specified

If you do not specify any of these options then the partners' properties are output in their abbreviated form.

-kl= | **-kl**=FTOPT | **-kl**=0 | 768 | 1024 | 2048 | 3072 | 4096

This operand enables you to display the properties of partner systems which have a specific RSA-PROPOSAL key (-kl). You can specify following values:.

-kl= | **-kl**=FTOPT

All partner systems with RSA-PROPOSAL set to global option "FTOPT" will be displayed.

-kl=0 | 768 | 1024 | 2048 | 3072 | 4096

All partner systems with RSA-PROPOSAL set to one of partner local RSA key value will be displayed.

If you want to display partners with any RSA-PROPOSAL, then do not set this value.

-klmin= | **-klmin**=FTOPT | **-klmin**=0 | 768 | 1024 | 2048 | 3072 | 4096

This operand enables you to display the properties of partner systems which have a specific RSA-MINIMUM key (-klmin). You can specify following values.

-klmin= | **-klmin**=FTOPT

All partner systems with RSA-MINIMUM set to global option "FTOPT" will be displayed.

-klmin=0 | 768 | 1024 | 2048 | 3072 | 4096

All partner systems with RSA-MINIMUM set to one of partner local RSA key value will be displayed.

If you want to display partners with any RSA-MINIMUM, then do not set this value.

3.66.1 Output format of ftshwptn

- Standard output
- Output in X.25 address format

3.66.1.1 Standard output

Example for the output in abbreviated form and in long output format:

```
ftshwptn
NAME     STATE SECLEV   PRI  TRACE  LOC  REM P-CHK RSA-PROP RSA-MIN ADDRESS
pingftam ACT   50       NORM FTOPT    0    0       2048     0       ftam://PING.homenet.de
PINGO    ACT   STD      NORM FTOPT    0    0 FTOPT FTOPT    FTOPT   PINGPONG.homenet.de:1234
rout0001 ACT   STD      HIGH FTOPT    0    0 FTOPT FTOPT    FTOPT   INCOGNITO
servftp  ACT   B-P-ATTR LOW  ON       0    0       4096     1024    ftp://ftp.homenet.de
ftshwptn -l
NAME     STATE SECLEV   PRI  TRACE  LOC  REM P-CHK RSA-PROP RSA-MIN ADDRESS
         INBND REQU-P        RECOV                ROUTING  IDENTIFICATION
pingftam ACT    50      NORM FTOPT    0    0       2048     0       ftam://PING.homenet.de
         DEACT STD           ON
PINGO    ACT   STD      NORM FTOPT    0    0 FTOPT FTOPT    FTOPT   PINGPONG.homenet.de:1234
         ACT   SERIAL        OFF
PINGPONG.homenet.de
rout0001 ACT   STD      HIGH FTOPT    0    0 FTOPT FTOPT    FTOPT   INCOGNITO
         ACT   STD           FTOPT                ROUT01   INCOGNITO.id.new
servftp  ACT   B-P-ATTR LOW  ON       0    0       4096     1024    ftp://ftp.homenet.de
         ACT   STD           OFF
```

Example for the output of an FTAM partner in long output format

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

Explanation

NAME

Name of the entry in the partner list.

STATE

Specifies how file transfer requests issued locally to the specified partner system are processed.

ACT

File transfer requests issued locally to this partner system are processed with *ftstart*.

DEACT

File transfer requests issued locally to this partner system are initially not processed, but are only placed in the request queue.

ADEAC

Failed attempts at establishing a connection lead to this partner system being deactivated. The maximum number of consecutive failed attempts is 5. In order to perform file transfers with this partner system again, it must be explicitly reactivated with *ftmodptn -st=a*.

NOCON

Attempt to establish a transport connection failed.

LUNK

Local system is not known in the remote FT system.

RUNK

Partner system is not known in the local transport system.

AINAC

Partner system has been deactivated after a number of unsuccessful attempts to establish a connection.

LAUTH

Local system could not be authenticated in the partner system. A valid public key for the local openFT instance must be made available to the partner system.

RAUTH

Partner system could not be authenticated in the local system. A valid public key for the partner system must be stored in the folder *syskey* of the openFT instance, see also section “Instance identification”.

In the case of the standard instance, *syskey* is in the directory */var/openFT/std* (Unix systems) and *%ProgramData%\Fujitsu Technology Solutions\openFT\var\std* (Windows systems), respectively.

DIERR

A data integrity error has been detected on the connection to the partner system. This can be the result of attempts at manipulation on the data transfer path or of an error in the transport system. The connection has been interrupted, but the affected request is still live (if it has the capability of being restarted).

NOKEY

The partner does not accept unencrypted connections, but no key is available in the local system. A new key must be generated.

IDREJ

The partner or an intermediate instance has not accepted the instance ID sent by the local system. Check whether the local instance ID matches the entry for the partner in the partner list.

SHORT

A resource bottleneck has occurred on the partner.

SECLEV

Security level assigned to the partner system.

1..100

A fixed security level is assigned to the partner system:

1 is the lowest security level (partner is extremely trusted) and 100 is the highest security level (partner is not trusted).

STD

The global setting for the security level applies.

B-P-ATTR

The security level is assigned to the partner on the basis of the partner's attributes, i.e.:

- Security level 10 if the partner has been authenticated.
- Security level 90 if the partner is known in the transport system and is identified by the name it is known by in the transport system.
- Security level 100 otherwise, i.e. if the partner has only been identified by its address.

PRI

Priority of a partner with respect to the processing of requests:

NORM

Normal priority.

LOW

Low priority.

HIGH

High priority.

TRACE

The global settings for partner selection in the openFT trace function apply.

FTOPT

The global setting for partner selection in the openFT trace function applies.

ON

The trace function is activated for this partner. However, a trace is only written if the global openFT trace function is also activated.

OFF

The trace function is deactivated for this partner.

LOC

Shows the number of file transfer requests addressed to the partner system entered in the local system.

REM

Shows the number of file transfer requests issued by the remote FT system and addressed to the local FT system.

P-CHK

Shows the settings for sender verification and authentication.

FTOPT

The global setting for sender verification applies.

STD

Checking of the transport address is deactivated. Only the identification of the partner is checked. The transport address of the partner is not checked even if extended sender verification is activated globally.

T-A

Checking of the transport address is activated. The transport address of the partner is checked even if checking of the transport address is deactivated globally. If the transport address used by the partner to log in does not correspond to the entry in the partner list, the request is rejected.

AUTH

The partner is subjected to a cryptographic identity check on the basis of its public key in the *syskey* directory (for authentication). The partner supports authentication level 2.

!AUTH

The partner is subjected to a cryptographic identity check on the basis of its public key in the *syskey* directory (for authentication). The partner supports authentication level 1.

AUTHM

Authentication must be used.

NOKEY

No valid key is available from the partner system although authentication is required.

RSA-PROP

Shows the properties of partner systems which have a specific RSA-PROPOSAL key.

FTOPT

The global setting for keys during transfers.

0 | 768 | 1024 | 2048 | 3072 | 4096

Local values for specifing partner for keys during transfers, which takes priority over globally set keys.

RSA-MIN

Shows the properties of partner systems which have a specific RSA-MINIMUM key.

FTOPT

The global setting for keys during transfers.

0 | 768 | 1024 | 2048 | 3072 | 4096

Local values for specifing partner for keys during transfers, which takes priority over globally set keys.

ADDRESS

Address of the partner system.

The following parameters are only output with *ftshwptn -l*.

ROUTING

Routing info of the partner system if specified.

IDENTIFICATION

Identification of the partner system if specified.

INBND

State of the partner for inbound requests:

ACT

Inbound function is activated, i.e. requests issued remotely are processed.

DEACT

Inbound function is deactivated, i.e. requests issued remotely are rejected.

REQU-P

Operating mode for asynchronous outbound requests:

STD

Requests to this partner can be processed in parallel.

SERIAL

Requests to this partner are always processed serially.

RECOV

Restart function (recovery) for outbound requests.

FTOPT

The global setting for the restart function in the operating parameters is valid.

ON

The restart function is activated.

OFF

The restart function is deactivated.

AETitle

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 Directory form (transparent).

AETitle format 2 (numeric)

The Calling AET is defined in numeric format.

Application Process Title and Application Entity Qualifier are optional outputs.

3.66.1.2 Output in X.25 address format

Example for the output of X.25 address parameters (Windows systems)

```
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 for the output of X.25 address parameters (Linux)

```
ftshwptn mchx25 -pa
NAME = mchx25
       TYPE = X.25 [FarSync] ID = 0
       DTE = 123456789012345
       NSAP = 430000000012345678901
          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 TCP/IP-RFC1006 transport system.

HOST/TNS Host or TNS name.

ID

Index of the address extension. 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 an OSI network address, then it is 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 case of a failed connection setup.

3.67 ftshwr

Note on usage

Function: Display request properties and status

User group: FT user and FT administrator

Functional description

The *ftshwr* ("show requests") command allows you to request information about FT requests. You can specify selection criteria in order to obtain information about specific FT requests.

Users can only obtain information about the requests they own.

The FT administrator can obtain information about the requests of any owner.

Format

ftshwr -h |

[-ua=<user ID> | -ua=@a]
[-ini=l | -ini=r | -ini=lr | -ini=rl]
[-st=a | -st=w | -st=l | -st=c | -st=f | -st=h | st=s][-pn=<partner 1..200>]
[-fn=<file name 1..512>]
[-gid=<global request identification 1..4294967295>][-s | -l][-csv]
[<request ID 1..2147483647>]

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-ua=user ID | **-ua=@a**

You use *-ua* to specify the user ID for which requests are to be displayed.

user ID

As a user, you can only specify your own user ID.

As an FT administrator, you may specify any user ID here.

@a

As an FT administrator, you can specify @a to display requests for all user IDs.

-ua= not specified

Your own user ID is the selection criterion.

Exception: The FT administrator has called the command and also specified a request ID: in this case, the presetting is *@a*.

-ini=l | **-ini=r** | **-ini=lr** | **-ini=rl**

You use *-ini* to specify the initiator for which you want to display requests. The following specifications are possible:

l (local)

Only locally submitted requests are displayed.

r (remote)

Only remotely submitted requests are displayed.

lr, rl (local + remote)

Both locally and remotely submitted requests are displayed.

-ini not specified

The initiator is not the selection criterion (corresponds to *lr* or *rl*).

-st=a | **-st=w** | **-st=l** | **-st=c** | **-st=f** | **-st=h** | **-st=s**

If you specify *-st* then only information on requests with the corresponding status is output.

The following specifications are possible:

a (active)

The request is currently being executed.

w (wait)

The request is waiting to be executed.

l (locked)

The request is locked.

c (cancelled)

The request has been deleted.

f (finished)

The request has already been executed.

h (hold)

The starting time specified on the issue of the request has not yet been reached.

s (suspend)

The request was interrupted, i.e. it is currently in the SUSPEND status.

-pn=partner

You use *-pn* to specify a name or an address for the partner system for which you want to display requests. The partner should be specified as on request submission or as output by the *ftshwr* command without the *-s*, *-l* or *-csv* option. If openFT finds a partner in the partner list for a specified partner address then *ftshwr* displays the name of the partner even if a partner address was specified at the time the request was entered.

-fn=file name

You use *-fn* to specify the file name for which requests are to be displayed. Requests that access this file in the local system are displayed.

You must specify the file name that was used when the request was issued. This file name is also output by the *ftshwr* command without the *-fn* option.

Wildcards are not permitted in the file name.

-gid=global request identification

With *-gid*, you specify the global request ID for a specific request that is to be displayed. The global request ID is only relevant for inbound requests from openFT and FTAM partners. It is assigned by the initiator of the request (transfer ID) and is sent to the local system.

-gid= not specified

The global request ID is not used as a selection criterion.

-s (sum)

specifies that a summary overview of requests is to be output. For each possible request status (see the *-st* option), this overview indicates the number of requests that have this status.

-l (long form)

specifies that the request properties are to be output in full.

-csv

Specifies that the request properties are to be output in CSV format. If you also specify *-s* then the summary overview is output in CSV format. The values in the overview are output separated by semicolons.

-s, *-l* and *-csv* not specified

The request attributes are output in standard form.

request ID

request ID specifies the identification of a specific request that is to be output. The request ID is output on the screen on acknowledgment of receipt of the request. It can also be viewed, for example, using the *ftshwr -l* command.

If you have specified a request ID and the other specified criteria do not correspond to the request then the request is not displayed and the following error message is output:

`ftshwr: Request` *request ID* `not found`

3.67.1 Output format of ftshwr

- Standard ftshwr output
- Totaled ftshwr output
- Detailed ftshwr output

3.67.1.1 Standard ftshwr output

Unix systems:

```
$ftshwr
TRANS-ID   INI STATE PARTNER  DIR  BYTE-COUNT  PROGRESS  FILE-NAME
65558      LOC WAIT  *PINGO   TO   0           NA        /home1/september.pdf
196610     LOC WAIT  servus.* FROM 0           NA        /home2/mails/memo02.txt
262146     LOC WAIT  servus.* TO   0           NA        /home3/pic/picture10.gif
196694     LOC ACT   win01    TO   0           2/11      dir/file.gr
65558      LOC ACT   %ip172.* FROM 154740      15/150    /home/hcl5
```

Windows systems:

```
ftshwr

TRANS-ID    INI STATE PARTNER   DIR   BYTE-COUNT   PROGRESS   FILE-NAME
65558       LOC WAIT   *PINGO    TO    0            NA         D:\september.pdf
196610      LOC WAIT   servus.* FROM 0            NA         D:\memo02.txt
262146      LOC WAIT   servus.* TO    0            NA         E:\pic\picture10.gif

327688      LOC ACT    %ip172.* FROM 174720       8/107      dir1/D1/ab1.txt

262190      LOC ACT    %ip172.* TO    145600       7/2166     dir2/D1
```

Explanation of the output

TRANS-ID

The TRANS-ID column (transfer identification) contains the request numbers used by openFT to identify the file transfer requests. The TRANS-ID can be used to cancel requests with the *ftcanr* command.

INI

The INI column indicates the initiator:

LOC: The request was submitted in the local system.

REM: The request was submitted in the remote system.

STATE

The STATE column indicates the state of the request.

The following states are possible:

ACT (active)

The request is currently being processed.

WAIT (wait)

The request is waiting.

In this case, the partner system (PARTNER) may be indicated. This indication shows the cause of the *WAIT* state.

LOCK (locked)

The request is temporarily excluded from processing.

This state may occur both for openFT and for FTAM partners.

With openFT partners, e.g. when a resource bottleneck is encountered or when external data media must be made available.

With FTAM partners, when one of the partners proposes a waiting period until the next start or recovery attempt via the FTAM protocol, and this period exceeds the delay normally permitted.

In this case, the partner system (PARTNER) may be indicated. This indication shows the cause of the *LOCKED* state.

CANC (cancelled)

The request was cancelled in the local system.

However, the remote system is aware of its existence, e.g. because it was previously active. Therefore, the request cannot be removed from the request queue until a connection to the partner has been re-established.

FIN (finished)

This status arises for requests involving FTAM partners when the request has been either completed or cancelled, but the user has not yet been informed of the fact.

HOLD (hold)

The start time specified when the request was issued has not been reached.

SUSP (suspend)

The request was interrupted.

PARTNER

Name or address of the partner, see also section “Specifying partner addresses”. If the partner address is more than 8 characters in length then it is truncated to 7 characters and suffixed with an asterisk (*).

If the request is in a WAIT or LOCKED state, the following indicators before PARTNER are also entered in the request queue:

(empty) No resources free at present (e.g. no memory).

*

The local FT administrator has locked the resource, e.g. deactivating the partner.

!

Connection setup to the partner system failed. The partner is currently inactive, or he can currently accept no further connections, or a network node has crashed.

Other possibilities: The connection to the partner system has been lost; a data integrity error has been detected.

?

An installation or configuration error has occurred (e.g. the local system is not known to the partner), authentication of one of the partners has failed, or the encryption is local, or not available to the partner system.

DIR

The DIR column specifies the direction of transfer.

TO

Send to the remote system.

FROM

Fetch from the remote system.

BYTE-COUNT

This column indicates the number of bytes transferred and saved up to now. The BYTE-COUNT counter is only updated at certain intervals.

PROGRESS

In case of directory transfer this column indicates the progress of directory transfer in the format *mm*/*nn* where *mm* is the number of subdirectories and files already transferred and *nn* is the total number of subdirectories and files to be transferred.

NA (not applicable)

indicates that either the directory transfer has not yet been started or a single file is to be transferred.

FILE-NAME

Name of the file in the local system.

3.67.1.2 Totaled ftshwr output

In the case of totaled output, a table showing the number of requests in the various request states is output (refer to the *State* column under the standard output for the meanings of the states):

```
ftshwr -s
 ACT   WAIT   LOCK   SUSP   HOLD    FIN      TOTAL
   3      2      0      0      0      0          5
```

3.67.1.3 Detailed ftshwr output

Example for the detailed output of the request with request ID 131074:

Unix systems:

```
ftshwr -l 131074
TRANSFER-ID =131074      STORE  =17-01-25 11:45:27  FILESIZE=514610
   STATE    =WAIT        BYTECNT=0                  PROGRESS=NA
   INITIATOR=LOCAL       TRANS  =FROM
   WRITE    =REPLACE     START  =SOON               CANCEL  =NO
   COMPRESS =NONE        DATA   =CHAR               PRIO    =NORM
   TRANSP   =NO          ENCRYPT=NO
   TARGFORM =BLOCK       TRECFRM=STD
   OWNER    =maier       DICHECK=NO
   FNC-MODE =CHAR                                   RECFORM =VARIABLE
   PARTNER  =ftserv01.mycompany.net
   PARTNER-STATE = ACT
   PARTNER-PRIO  = NORM
   LOC: FILE     =/home2/memo02.txt
        TRANS-ADM=(maier)
        CCSN     =ISO88591
   REM: FILE     =/home/save/memo02.txt
        TRANS-ADM=(servelog)
```

Windows systems:

```
TRANSFER-ID =131074      STORE  =17-01-25 11:49:11  FILESIZE=514610
   STATE    =WAIT        BYTECNT=0                  PROGRESS=NA
   INITIATOR=LOCAL       TRANS  =FROM
   WRITE    =REPLACE     START  =SOON               CANCEL  =NO
   COMPRESS =NONE        DATA   =CHAR               PRIO    =NORM
   TRANSP   =NO          ENCRYPT=NO
   TARGFORM =BLOCK       TRECFRM=STD
   OWNER    =maier       DICHECK=NO
   FNC-MODE =TRANSPARENT                            RECFORM =VARIABLE
   PARTNER  =ftserv01.mycompany.net
   PARTNER-STATE = ACT
   PARTNER-PRIO  = NORM
   LOC: FILE     =E:\memo02.txt
        TRANS-ADM=(mydomain\maier)
        CCSN     =CP1252
   REM: FILE     =memo02.txt
        TRANS-ADM=(servelog)
```

Example of detailed output of inbound request with request ID 524410:

```
ftshwr -l 524410
   TRANSFER-ID =524410  STORE =17-01-25 14:33:24   FILESIZE=10485760
   STATE =ACTIVE        BYTECNT=0                   PROGRESS=NA
   INITIATOR=REMOTE     TRANS =FROM                 RECSIZE =1024
   WRITE =REPLACE       START =SOON                 CANCEL =NO
   COMPRESS =NONE       DATA =CHAR                  PRIO =
   TRANSP =NO           ENCRYPT=NO                  GLOB-ID =852520
   OWNER =user1         DICHECK=NO                  TABEXP =NO
   FNC-MODE =CHAR                                   RECFORM =VARIABLE
   PARTNER =ftserv.mycompany.net
   PARTNER-STATE =ACT
   PARTNER-PRIO =NORM
   FILE =par.file.S3.C31
   TRANS-ADM=(serv,)
```

Explanation of the output

TRANSFER-ID (transfer identification)

Request ID which openFT uses to identify file transfer requests. Requests can be canceled using the *ftcanr* and the request ID.

STATE

State of the request. Possible values:

ACTIVE

The request is currently being processed.

WAIT

The request is waiting. If the cause of the WAIT state is known, further information is indicated in the PARTNER-STATE field.

LOCKED

The request is temporarily excluded from processing. This status can also occur at openFT and at FTAM partners.

With openFT partners, when a resource bottleneck is encountered or if external data media must first be made available for example.

With FTAM partners, when one of the partners proposes a waiting period until the next start or recovery attempt via the FTAM protocol, and this period exceeds the delay normally permitted.

If the cause of the LOCKED state is known, further information is indicated in the PARTNER-STATE field.

CANCELLED

The request was cancelled in the local system. However, the remote system is aware of its existence because, for example, it was previously active. Therefore, the request cannot be removed from the request queue until the connection to the partner has been re-established.

FINISHED

This status occurs for requests involving FTAM partners when the request has either been completed or cancelled, but the user has not yet been informed of this.

HOLD

The start time specified when the request was issued has not yet been reached.

SUSPENDED

The request was interrupted.

INITIATOR

This specifies where the request was issued. Possible values:

LOCAL

The request was issued in the local system.

REMOTE

The request was issued in the remote system.

WRITE

This specifies whether the destination file is to be overwritten, extended or created. Possible values:

OVERWRITE (default value)

If the destination file already exists, it is overwritten; otherwise, it is created.

EXTEND

If the destination file already exists, the file sent is appended to the destination file.

If the destination file did not exist, it is created.

NEW

A new destination file is created and written.

COMPRESS

This specifies whether the file should be transferred with data compression.

Possible values: BYTE, ZIP, NONE.

TRANSP

Indicated whether the file is to be sent in transparent file format. Possible values: YES, NO

TARGFORM

Format of the file in the target system.

Possible values:

STD (default value)

The file is saved in the same format as in the sending system.

BLOCK

The file is saved in block format.

SEQ

The file is saved as a sequential file.

OWNER

Local login name.

FNC-MODE

Encoding mode for remote file names and follow-up processing. Possible values:

TRANSPARENT

Encoding in transparent mode.

CHAR

Encoding in character mode.

PARTNER

Name or address of the partner, see also section “Specifying partner addresses”.

PARTNER-STATE

Status of the partner. Possible values:

ACT

Activated

DEACT

Deactivated

NOCON

No connection, for example because the openFT server has not been started in the remote system.

INSTERR

An installation or configuration error has occurred (the local system is not known to the partner, for instance), authentication of one of the partners has failed, or the encryption is local, or the encryption is not available at the partner system.

SHORT

A resource bottleneck has occurred on the partner.

PARTNER-PRIO

Prioritization of the partner when processing requests.

Possible values:

LOW

The partner has low priority.

NORM

The partner has normal priority.

HIGH

The partner has high priority.

LOC

Properties in the local system:

FILE

File name in the local system

TRANS-ADM

Transfer admission for the local system

CCSN

CCS name used in the local system. The CCSN is only output for text files.

SUCC-PROC

Local follow-up processing commands if successful (if specified in the request).

FAIL-PROC

Local follow-up processing commands if unsuccessful (if specified in the request).

REM

Properties in the remote system:

FILE

File name in the remote system

TRANS-ADM

Transfer admission in the remote system. Possible values:

REMOTE-PROFILE

request with FTAC transfer admission

TRANS-ADM=(*user ID*)

request with *user ID,,password*

CCSN

CCS name used in the remote system

SUCC-PROC

Remote follow-up processing commands if successful (if specified in the request).

FAIL-PROC

Remote follow-up processing commands if unsuccessful (if specified in the request).

STORE

Indicates the time at which the request was entered in the request queue.

BYTECNT

This value is output only if the request is currently active or if it was already active and the file transfer has been interrupted. BYTECNT indicates the number of bytes transferred and saved up to now. The counter is updated regularly.

TRANS

This shows the direction of transfer. Possible values are:

TO

The file is sent.

FROM

The file is received.

START

Indicates the time at which the request is to be started. Possible values:

Date / Time

The date and time at which the request is to be started is output.

SOON

The request should be started as soon as possible.

No entry

The request was issued in the remote system.

DATA

Indicates the file type. Possible values:

CHAR (default value for openFT partners)

The file contains text with variable record lengths.

BIN

The file contains an unstructured sequence of binary data.

USER

The file contains structured binary data with variable record length.

ENCRYPT

Indicates whether data encryption was specified.

Possible values: NO, YES.

TRECFRM

Record format of the file in the target system

Possible values:

STD (default value)

The file is saved with the same record format as in the sending system.

UNDEFINED

The file is saved with an undefined record format.

DICHECK

Specifies whether the integrity of the data is to be checked.

Possible values: NO, YES.

FILESIZE

Size of the file in bytes. If the output is followed by a "K", the output is in kilobytes. If it is followed by an "M", the output is in megabytes. The size is indicated here only if the request was already active. For receive requests, a value is indicated here only if the partner has sent one with the request.

PROGRESS

In case of directory transfer this column indicates the progress of directory transfer in the format *mm*/*nn* where *mm* is the number of subdirectories and files already transferred and *nn* is the total number of subdirectories and files to be transferred.

NA (not applicable)

indicates that either the directory transfer has not yet been started or a single file is to be transferred.

RECSIZE

Maximum record size, if specified.

CANCEL

If the "Cancel-Timer" was set, the time at which the request is deleted from the request queue is indicated here. If no cancel time was specified in the request, NO is output.

PRIO

Request priority. Possible values:

NORM

The request has normal priority

LOW

The request has low priority

No entry

The request was issued in the remote system.

GLOB-ID

Global request identification, displayed only in the case of inbound requests from openFT and FTAM partners (INITIATOR=REMOTE). This corresponds to the request identification (=TRANSFER-ID) on the initiator system.

TABEXP

Tabulator expansion and the conversion of blank lines into lines with one character activated/deactivated, only relevant for outbound send requests.

Possible values: YES, NO

RECFORM

Record format.

Possible values: UNDEFINED, VARIABLE, FIX.

DIAGCODE

This column is usually empty. Otherwise, it provides further diagnostic information on operational states in the form of a CMX return code or an FTAM or openFT diagnostic code. FTNEA diagnostic codes have the format NEBFnnnn (NEABF) or NEBDnnnn (NEABD). The following openFT diagnostic codes have been defined:

Value	Meaning
0	No cause specified.
1	Connection setup normal.
2	There is a resource bottleneck.
3	There is a resource bottleneck; the connection will be set up later by the rejecting entity.
4	Initialization is not yet complete.
5	SHUTDOWN is in progress.
6	The requesting entity is unknown.
7	A protocol error has occurred.
8	A transport error has occurred.
9	A system error has occurred.
10	This code is reserved (for SN77309 part 5).
11	The connection is not accepted without encryption.

FTAM diagnostic codes have the format FTAMnnnn and values from the ISO 8571-3 standard. An extract of possible diagnostic codes taken from the standard can be found in the section “FTAM diagnostic codes as per ISO 8571-3”.

The following values are only output for FTAM partners:

STOR-ACCOUNT

Account number; is output only if specified by the user.

AVAILABILITY

Possible values: IMMEDIATE, DEFERRED.

Is output only if specified by the user.

ACCESS-RIGHTS

Access mode

Possible values: combinations of r, i, p, x, e, a, c, d.

Is output only if specified by the user.

LEGAL-QUAL

Legal qualification

Is output only if the local system is the initiator and the value is specified by the user.

3.68 ftshws

Note on usage

Function: Displaying openFT-Script requests

User group: FT user and FT administrator

Functional description

Outputs information about the status of a user's openFT-Script requests. You can also specify a *ftscriptid* in order to select a specific openFT-Script request.

Format

ftshws -h |

[-csv]
[-t]
[-v]
[-st=[W]|[R]|[T]|[F]|[I]|[C]|[X]]
[-u=<user ID> | @a]
[<ftscriptid>]

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-csv

The information is output in CSV format. If you do not specify *-csv* then the information is output in table format.

-t

The openFT-Script requests are displayed sorted on generation time, beginning with the last request.

By default, the requests are displayed in alphabetical order.

-v

Diagnostic information is also output (verbose).

If *-v* is specified then, in the case of openFT-Script requests which terminate with an error, the cause of the error is output in a second line after the tabular information.

In CSV format, the *-v* option is ignored.

-st=[**W**][**R**][**T**][**F**][**I**][**C**][**X**]

displays openFT-Script requests with the specified status, see *Sta* field below.

You can specify multiple statuses one after the other, e.g. *-st=WRT*.

-u=user ID | **@a**

User ID for which openFT-Script requests are output or under which the specified request is searched for.

Only administrators may specify a user ID or *@a* (all user IDs).

The default value is the calling party's user ID.

ftscriptid

Identification of the openFT-Script request. This is output if the openFT-Script request is started via an *ftscript* command.

You can use the wildcard symbols *?* and * in der *ftscriptid*. This outputs all openFT-Script requests that match the wildcard pattern.

?

is interpreted as any single character.

*

is interpreted as any number of characters.

If you use wildcards, enclose the *ftscriptid* specification in single quotes so that the wildcard symbols are not interpreted by the shell.

By default, if you do not specify *ftscriptid,* all the user's openFT-Script requests are displayed.

Output in table format

The processing level of the openFT-Script requests is displayed in four columns:

User

User ID under which the request was started.

Ftscriptid

Unique identification of the request. The identification is returned by the *ftscript* command.

Sta

Indicates the processing status, where:

W (waiting)	The request has not yet been started.
R (running)	The request has been started but has not yet been terminated.
T (terminated)	The request has been terminated without errors.
F (failure)	The request has been terminated with errors.
I (interrupted)	The request was interrupted, e.g. due to a system crash.
C (cancelled)	The request was cancelled with an *ftcans* command.
X (cancelling)	The request is currently being cancelled due to an *ftcans* command.

FtscriptFileName

Path name of the script file.

If the status F and the *-v* option are specified then the cause of the error is output in clear text in another column.

3.69 ftshwsuo

Note on usage

Function: Displaying openFT-Script user options

User group: FT user and FT administrator

Functional description

You use the *ftshwsuo* command to display the directory in which the openFT-Script requests are to be stored. In addition, *ftshwsuo* displays the limits currently set.

Format

ftshwsuo -h |

[-csv]
[-u=<user ID> | @a]

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-csv

The information is output in CSV format. If you do not specify *-csv* then the information is output in table format.

-u=user ID| **@a**

User ID whose openFT-Script options are to be displayed: *@a* means that the openFT-Script options of all active openFT-Script users as well as of all openFT-Script users who have a working directory other than the default openFT-Script working directory are to be displayed.

Only administrators may specify a user ID or *@a* (all user IDs).

The default value is the calling party's user ID.

Output in table format

```
Common Ftscript User Limits:    Threads: <mmm>  File Transfers:  <nnn>
User              FtscriptWorkdir
---------------------------------
<user>            <path name>
```

Explanation

Threads: <mmm>

mmm is the maximum number of threads which openFT-Script simultaneously executes in a Java VM of a user.

File Transfers: <nnn>

nnn is the maximum number of simultaneous openFT file transfers which openFT-Script triggers from a Java VM of a user.

*CLIM for *nnn* means that the default file transfer limit applies, i.e. 2 * CONN-LIM where CONN-LIM is the connection limit set via operating parameters, see output of the *ftshwo* command.

<user>

User ID

<path name>

Designates the name of the openFT-Script working directory that the user has set with *ftmodsuo* without the subdirectory names created by openFT-Script.

If the user has not set any special working directory then the name of his or her home directory is output since this is the openFT-Script directory by default and is used to store the openFT-Script requests.

3.70 ftstart

Note on usage

Function: Start asynchronous openFT server

User group: FT administrator

Functional description

This command starts the asynchronous openFT server. This processes all the requests stored in the request queue as well as all the inbound requests. It is necessary to shut down and restart the asynchronous openFT server, for example, if you want to switch between operation with and without CMX.

Notes for Unix systems

When the asynchronous openFT server is started, the protection bit settings for files that are created on inbound requests are set implicitly. The settings for the shell under which you entered *ftstart* apply. For more details, see the manual "openFT (Unix and Windows systems) - Installation and Operation".

In the case of openFT on Solaris, please refer to the section "Solaris SMF" in the manual "openFT (Unix and Windows systems) - Installation and Operation".

Notes for Windows systems

It is necessary to shut down and restart the asynchronous openFT server, for example, if you have changed the file access permissions for newly created files. See the notes on setting file access rights for newly created files in the manual "openFT (Unix and Windows systems) - Installation and Operation".

Format

ftstart [-h]

Description

-h

Displays the command syntax on the screen.

3.71 ftstop

Note on usage

Function: Stop asynchronous openFT server

User group: FT administrator

Functional description

This command shuts down the asynchronous openFT server. After this, no further inbound requests and no locally submitted asynchronous requests are processed:

- Inbound requests are rejected
- Locally submitted asynchronous requests are stored in the request queue

Once the *ftstop* command has been issued, the asynchronous openFT server is not stopped until all the server processes have been terminated. This may take a few minutes if, for example, disconnection is delayed due to line problems.

When the asynchronous openFT server is restarted, the requests present in the request queue are processed normally. Requests that were cancelled due to the shutdown of the asynchronous openFT server are relaunched provided that the partner supports this function.

In the case of openFT on Solaris, please refer to section "Solaris SMF" in the manual "openFT (Unix and Windows systems) - Installation and Operation".

Format

ftstop [-h]

Description

-h

Displays the command syntax on the screen.

3.72 fttrace

Note on usage

Function: Evaluating trace files

User group: FT user and FT administrator

Functional description

Trace files for all protocols (openFT, FTAM and ftp protocol) are evaluated with the *fttrace* command.

Format

fttrace -h |

[-d]
[-sl=n | -sl=l | -sl=m | -sl=h]
[-cxid=<context id>]
[-f=hh:mm:ss]
[-t=hh:mm:ss]
<tracefile> [<tracefile> ...]

Description

-h

Outputs the command syntax on screen. Any specifications after *-h* are ignored.

-d

Specifies that the trace files are to be output in hexadecimal format (dump format). However, this does not function with the FTP protocol.
If you do not specify *-d* then the files are output in printable form, default value.

-sl=n | -**sl**=l | -**sl**=m | -**sl**=h

Specifies the security level for the output if the files are output in printable format (also see the note):

n (no)

No security requirements, i.e. all the data is output. This includes IDs, passwords, transfer admissions, file names etc.

l (low)

Passwords are overwritten with XXX.

m (medium)

Passwords, user IDs, transfer admissions, account numbers and follow-up processing commands are overwritten with XXX.
Default value if *-sl* is not specified.

h (high)

Passwords, user IDs, transfer admissions, account numbers, follow-up processing commands and file names are overwritten with XXX.

> **i** If the files are output in dump format (*-d*) then, irrespective of the value specified in *-sl*, the lowest security level (*-sl=n*) is always used since the trace files are output without any further interpretation or evaluation and may therefore also contain user IDs and passwords in clear text.

-cxid=context id

Selects the trace entries on the basis of the context ID. If you omit *-cxid* or specify *-cxid=* without a context ID then all the trace entries are output.

-f=hh:mm:ss

(from) Specifies the time as of which trace entries in the trace file are to be evaluated. You enter the time in the format hours:minutes:seconds (2 digits each).
If you do not specify a start time then trace entries are output from the start of the file.

-t=hh:mm:ss

(to) Specifies the time up to which trace entries in the trace file are to be evaluated. You enter the time in the format hours:minutes:seconds (2 digits each).
If you do not specify an end time then trace entries are output up to the end of the file.

tracefiles

Name(s) of the trace file(s) that you want to evaluate. You can specify multiple trace files and wildcards can be used.

3.73 ftupdi

Note on usage

Function: Update the instance directory

User group: FT administrator

Functional description

Using *ftupdi*, you can update an instance file tree that was made using openFT V10.0, V11.0 or V12.0 so that it can continue to be used with openFT V12.1. The settings of the operating parameters, FTAC admission sets, FTAC admission profiles and log records are retained.

Any requests for this instance which are still present will be lost.

Single-user mode on Unix systems

Only the owner of an 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*.

Format

ftupdi -h | <directory 1..128>

Description

-h

Displays the command syntax on the screen. Any entries after *-h* are ignored.

directory

Here, you enter the directory which contains the instance file tree of the instance to be updated.

Messages of the ftupdi command

If *ftupdi* could not be carried out as specified, an explanatory message is displayed; the exit code will then be "not equal to zero".

Example

The FT administrator wants to update the directory of the instance *hugo*.

```
ftupdi /var/openFT/.hugo (Unix systems)

ftupdi "C:\ProgramData\Fujitsu Technology Solutions\var\hugo" (Windows  systems)
```

3.74 ftupdk

Note on usage

Function: Update public keys

User group: FT administrator

Functional description

Using *ftupdk*, you can update the public key files of existing key pair sets.

For example, you can use it to insert updated comments from the *syspkf.comment* file into existing public key files or replace accidentally deleted public key files of a key pair set.

Format

ftupdk [-h]

Description

-h

Displays the command syntax on the screen.

Example

The name of the FT administrator is to be imported into the public key files. First, the file *syspkf.comment* is edited using an editor. This file is located in the *config* subdirectory of the instance directory, see the *ftcrei* command.

The file might, for example, contain only the following line:

```
FT administrator: John Smith, Tel. 12345
```

The command is:

```
ftupdk
```

The command is executed without an error message. Following this, the information will be placed at the beginning of all *syspkf...* public key files as a comment line.

3.75 install.ftam

> As of openFT Version V12.1C30, install.ftam and install.ftp do not exist anymore, because they are replaced with the license system. Running install.ftp and install.ftam on openFT V12.1C30 or later will have no effect.

Note on usage

Function: Install openFT-FTAM

User group: FT administrator

This command is only available on Unix systems.

Functional description

The *install.ftam* command allows you to install and uninstall openFT-FTAM. Installation is only permitted if you have an openFT-FTAM license.

The *install.ftam* command is located in the */opt/openFT/bin/ftbin* directory.

Format

install.ftam -h | -i | -d

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-i

openFT-FTAM is installed.

-d

openFT-FTAM is uninstalled.

3.76 install.ftp

> As of openFT Version V12.1C30, install.ftam and install.ftp do not exist anymore, because they are replaced with the license system. Running install.ftp and install.ftam on openFT V12.1C30 or later will have no effect.

Note on usage

Function: Install openFT-FTP

User group: FT administrator

This command is only available on Unix systems.

Functional description

You use the *install.ftp* command to install and uninstall openFT-FTP. Installation is only permitted if you have an openFT-FTP license.

The *install.ftp* command is located in the */opt/openFT/bin/ftbin* directory.

Format

install.ftp -h | -i | -d

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

-i

openFT-FTP is installed.

-d

openFT-FTP is uninstalled.

3.77 ncopy

Note on usage

Function: Synchronous file transfer

User group: FT user

Alias name: *ftscopy*

Functional description

The *ncopy* command is used to issue synchronous requests for sending one or several files or directories to a remote system or for fetching a file or a directory from a remote system or for executing an operating system command in the local or remote system. The *ncopy* command is executed even if the asynchronous openFT server has not been started.

Instead of a local file, you can also use standard input (*stdin*) when sending a file, and standard output (*stdout*) when receiving a file.

If openFT rejects your request, an error message will be displayed explaining why it was rejected (see chapter “Messages”).

openFT transfers the file synchronously to the user process or executes the remote command.

Notes on transferring multiple files

- For the openFT protocol and the FTP protocol applies: Only **one** file can be fetched from a remote system for each *ncopy* command (without *-d* option). If you want to fetch several files synchronously, use the *ft_mget* command.
- For the FTAM protocol applies: It is also possible to fetch or send multiple files for each *ncopy* command. This is controlled via using a file name starting with two commas. Please refer to the openFT manual "Concepts and Functions", section "Special points for file transfer with FTAM partners" for details.

Status message

openFT displays a status message while file transfer is in progress. The syntax of this message is as follows:

*b*KB [*p*%; [*hh*:]*mm*:*ss*]

The variables are:

b

Number of bytes (in KB) already transferred

p

Percentage of file already transferred

hh:mm:ss

estimated time to completion of transfer in hours, minutes and seconds. The hours are not displayed unless the time to completion is longer than sixty minutes. If the size of a file for a receive request is unknown, only the counter for the number of bytes transferred is active.

The status message is updated every three seconds. The first message does not include the anticipated time to completion of transfer. You receive status information only if

- the file is correspondingly large,
- the *-S* or *-s* switch was not set to suppress messages,
- the request is not running as a background process (*ncopy .... &*, only on Unix systems),
- the standard error output (*stderr*) is not redirected to a file,
- a file was specified as source file or the data was input via a pipe (dash (-) for source file), i.e. not input via keyboard.

If the size of the send file is unknown, the status message merely shows the number of bytes already transferred. This is the case if the data is input via a pipe or when a file is received.

When the transfer has been successfully completed, openFT outputs a result message on the screen (*stderr*) of the user with the following format:

`ncopy: request` *request ID.* `File` *file name* `transferred` (without *-d* option)

`ncopy: request` *request ID.* `Directory` *directory name* `transferred` (with *-d* option)

If openFT was not able to execute your request successfully, an error message will be displayed on the screen (see chapter “Messages”).

> A number of special considerations apply for transfer requests with FTP partners. See the section “Notes on FTP partners”.

Format

ncopy -h |

[-t | -u | -b][-x]
[-o | -e | -n]
[-k | -z][-c][-S | -s] [-m=n | -m=f | -m=a] *)
[-d]
[<file name 1..512> [<file name 1..512>...<file name 1..512>] | -

<partner 1..200>![<file name 1..512> | <prefix 0..511>%]] |

[<partner 1..200>![<file name 1..512>]

<file name 1..512> | <prefix 0..511>% | -]

[<transfer admission 8..67> | @n | @d |
<user ID 1..67> [,[<account 1..64>] [,[<password 1..64>]]]][-p=[<password 1..64>]][-di]
[-lc=<CCS name 1..8>][-rc=<CCS name 1..8>][-rs=<follow-up processing 1..1000>]
[-rf=<follow-up processing 1..1000>]
[-r=v[<1..65535>] | -r=f[<1..65535>] | -r=u[<1..65535>] |-r=<1..65535>]
[-tff=b | -tff=s][-trf=u]
[-tb=n | -tb=f | -tb=a]
[-av=i | -av=d] [-ac=<new account 1..64>]
[-am=[r][i][p][x][e][a][c][d] | -am=@rw | -am=@ro] [-lq=<legal qualification 1..80>]
[-cp=[<password 1..64>]]
[-fnc=t | -fnc=c][-md]

*)The option *-m* is only available on Unix systems.

Description

-h

Displays the command syntax on the screen. Entries after the *-h* are ignored.

[**-t** | **-u** | **-b**][**-x**]

Identifies the type of file in the local system.

If you send a file to an FTAM partner without specifying a file type, the file type is determined by the structure entries of the send file. The structure entries can be displayed by outputting the local openFT attributes (*ftshwf file name -l*). If there are no structure entries, the default value is *-t*. If you fetch a file from an FTAM partner without specifying a file type, the file type is determined by the file attributes in the FTAM partner. For more detailed information about file types when dealing with FTAM partners, see the section “FTAM diagnostic codes as per ISO 8571-3”.

-t (default value with openFT partners)

The file contains text with variable-length records.

Records end with the linefeed character \n on Unix systems.

On Windows systems, records end with the following characters:

- CRLF (X'0D0A') when sending and/or fetching a file
- LF (X'0A'), only possible when sending a file

Maximum record length = 32767 bytes.

-u

The file contains binary data with variable record length structured by the user. Each record starts with 2 bytes which contain the length data for the record.

-b

The file contains user-structured binary data with variable-length records.

-x

The send file is transferred in a transparent file format and is stored in the destination system, i.e. this is a file whose attributes are transparent for the local system.
The local system here acts as a storage and/or transport medium.

If a file is transparently retrieved with *-x* for local buffering, then it must be sent again to the remote system in binary form (i.e. with *-b*).

-o | **-e** | **-n**

Indicates whether a destination file is to be newly created, overwritten or extended.

-o (default value)

The destination file will be overwritten or newly created if it does not already exist.

In case of directory transfer (*-d* option), the target files are overwritten if the specified directory and the files in this directory already exist. Otherwise, the target directory, subdirectories (if they may exist) and the files are newly created. Files and subdirectories which only exist in the target directory remain unchanged.

-e

The transferred file will be appended to an existing destination file. If this destination file does not exist, it will be newly created.

-e is not permitted in case of directory transfer (*-d* option).

-n

The destination file will be newly created and written. If the destination file already exists, the request will be rejected. In this way, you can protect a file from being overwritten inadvertently.

In case of directory transfer (*-d* option), the target directory and the files are newly created. If the target directory already exists, the request is rejected.

-k

Indicates that identical characters repeated consecutively are to be transferred in compressed form (byte compression). In the case of connections to partners which do not support this type of compression, no compression is used automatically.

-z

Indicates that zip compression is used. In the case of connections to partners which do not support this type of compression, byte compression (corresponds to the option *-k*) or no compression are used automatically.

-c

Indicates that the transfer data are encrypted during file transfer. Encryption of the request description data is not affected by this option. If the partner system does not support data encryption, the request is rejected.

[**-S** | **-s**]

Suppresses file transfer messages to *stderr*.

-S

All messages are suppressed.

-s

The status message and the end messages are suppressed; error messages are output.

-m=n | **-m=f** | **-m=a** (only on Unix systems)

This indicates whether the result message is to be deposited in the mail box of the user who issued the request.

n (default value)

The result message is not deposited in the mailbox.

f

The result message is only deposited in the mailbox in the event of errors.

a

The result message is always deposited in the mailbox.

-d

indicates a directory transfer.

Local and remote file names are interpreted as directory names.

-d is only supported for openFT partners (not for FTAM or FTP partners). Preprocessing and postprocessing are not supported.

If you are using the *-d* option together with other options (e.g overwrite (*-o*) or follow-up processing (*-rs*,...)) then these options apply to the individual files in the directory to be transferred.

file name1 [file name2.. [file name]] | - partner**!**[file name | [prefix]**%**] |

partner**!**[file name] file name | - | [prefix]**%**

specifies the source and destination. The syntax depends on the direction of transfer selected and whether pre- or postprocessing commands are used or whether a directory is transferred. The character "-" (hyphen) stands for the standard input or output (*stdin*/*stdout*).

If you are using the option *-d* (directory transfer) then the source and destination file name are considered as directory names.

Sending without pre/postprocessing

Sending a file

Source	**Destination**
local file1 [*local* file2 ..] | -	partner**!**[*remote* file | [prefix]**%**]

Sending directories

Source	**Destination**
local directory name1 [*local* directory name2 ..]	partner**!** *remote* directoy name

If you transfer one or more directories (*-d*) then you specify the directories you want to transfer in *local directory name1, local diretory name2 ...*. For *remote directory name*, you specify the directory under which the transferred directories are stored with identical file names and subdirectory names if applicable. The specification for the remote directory may not be omitted.

Fetching without pre/postprocessing

Fetching a file

Source	**Destination**
partner**!**[*remote* file]	*local* file | - | [prefix]**%**

Fetching a directory

Source	Destination
partner**!**[*remote* directory name]	*local* directory name | [prefix]**%**

Sending and fetching a file with pre- or postprocessing

If you want to perform pre- or postprocessing, then you must enter an operating system command instead of the local or remote file name (in the syntax of the corresponding system):

Sending with preprocessing

Source	Destination
"| *local* command**"**	Partner**!**[*remote* file]

Sending with postprocessing

Source	Destination
local file1 [*local* file2 ..] | -	Partner**!"|** *remote* command**"**

Fetching with preprocessing

Source	Destination
Partner**!"|** *remote* command**"**	*local* file | - [1)]

[1)]- stands for the standard output

Fetching with postprocessing

Source	Destination
Partner**!**[*remote* file]	**"|** *local* command**"**

You can also combine preprocessing and postprocessing in the same request.

A maximum of 712 bytes may be specified for *source* and *destination* (maximum 512 bytes for the file name and maximum 200 for the partner). Please note that the maximum lengths of file names are system-dependent; for example, in Unix systems it is 512 and in Windows systems a maximum of 256 bytes (for the representation in UTF-8, see section “Entering commands”).

local file1 [local file2 ..] local directory name1 [local directory name2]

> **Sending:** The name(s) of the local file(s) or directory(s) have to be entered here. If you send several files, you have either to specify %, %BASENAME or %FILENAME for the remote file name, see below, or you specify one remote file name and use option *-e*. With *-e*, the transferred files are concatenated and written in the specified remote file.
>
> The specification of UNC names is also possible.
>
> Wildcards

If you wish to send several files to a remote system and the files should have the same names in the remote system, you may use wildcards. Do this using the asterisk (*) commonly used for example. The file name must not contain exclamation marks (!). If you specify commands for follow-up processing, follow-up processing is carried out for each file.

If you want to send diretories to the remote system you can use wildcards too.

Note on Windows systems

If you call the *ncopy()* function from the DLL *ncpdll32.dll* from a program, you can only specify a filename. Wildcards and prefixes are not supported.

Fetching: Enter the name of the receive file or the local directory (*-d* option).

The name may be an absolute or relative path name.

If the name ends with %unique or %UNIQUE, this string is replaced by a string which changes for each new call, see section “Entering commands”.

If the directory in the specified path name does not yet exist the following applies:

- in case of transferring a file the directory is not created
- in case of transferring a directory the directory is always created

\- (dash) for *local file*

Sending: The dash for local file stands for standard input *stdin*.

You can also enter data directly via keyboard, in which case you send the *ncopy* command with a dash for the local file, before processing to enter data.

You terminate entry by pressing the following keys:

<END> or CTRL+D (Unix systems)

CTRL+Z at the start of a line, followed by Return (Windows systems).

See Examples for more details.

Fetching: The dash stands for standard output *stdout*. The dash directs output to the screen.

Note

- The dash is only allowed for transferring individual files but not with option *-d*.
- On Unix systems apply: You can use the dash if you want to link the *ncopy* output with a command on the Unix system, see Examples for more details.

[prefix]**%** for *local file*

Fetching files: For the receive file name, you may specify *%*, *%BASENAME*, *%FILENAME* or, in addition, a prefix. These variables are substituted as follows:

% and **%BASENAME**

are substituted by the last part of the name of the remote file. The last part of the name starts after the slash (/) or backslash (\), or a corresponding character in the remote system.

%FILENAME

is overwritten by the full name of the remote file specified in the command.

prefix

You may also specify a prefix for the local file name, e.g. *save.%FILENAME*. This prefix must end with a dot (.), a slash (/) or a backslash (\).

Fetching directories: When you specify *%*, *%BASENAME* or *%FILENAME*, the name and the structure of the fetched directory are taken over. The specification of prefix% acts analogously as for fetching files.

remote fileremote directory name

remote file and *remote directory name* can be either absolute or relative to the remote transfer admission (when sending or fetching). If the name in the remote system has been predefined in an admission profile, it must not be specified here.

If the name contains blanks, they must be enclosed in double quotes (e.g. "file name").

If the partner system is running openFT (BS2000), elements from PLAM libraries may also be specified here (syntax: Libname/Element type/Element name).

If the name ends with %unique or %UNIQUE, this string is replaced by a string which changes for each new call, see section “Entering commands”.

If the file name of a receive request starts with an pipe character ("|"), the file name is executed on the remote system as a command if the remote system supports the preprocessing function.

[prefix]**%** for *remote file* or *remote directory name*

Sending files: If you are sending several files, you have to specify *%*, *%BASENAME*, *%FILENAME* for the remote file name. In addition, you can specify a prefix. These variables are substituted as follows:

% and **%BASENAME**

are substituted by the last part of the name of the send file. The last part of the name starts after the slash (/) or backslash (\), or a corresponding character in the send system.

Please note that when you use % and %BASENAME with wildcards, files with the same names can be produced during substitution and that these are mutually overwritten.

Example

Unix systems:

```
ncopy file/test1.c test/test1.c\
      partner!destination/% transadm
```

Windows systems:

```
ncopy file\test1.c test\test1.c
       partner!destination\% transadm
```

Both files are copied to *destination / test1.c and destination \ test1.c,* respectively.

%FILENAME

is overwritten by the full name of the send file specified in the command.

prefix

You may also specify a prefix for the remote file name. This name must end with a dot (.), a slash (/) or a backslash (\).

Example

```
ncopy *.c *.txt test partner!prob.% profile01
```

All files which end with *.c* and *.txt* and the *test* file are transferred to the remote system and stored there under the name *prob*.<*local filename*>. Here, *profile01* is the transfer admission.

Sending directories: When you specify *%*, *%BASENAME*, or *%FILENAME*, the name and the structure of the sended directory(s) are taken over. The specification of prefix% acts analogously as for sending several files.

|command for *file name*

command is any command on the local or remote system (not permitted in case of directory transfer). The "|" character (vertical bar or pipe character) must always be placed before the command. Since the "|" character is a special character
"**|**command" should always be enclosed in double quotes.

Please note that, as of openFT V12, pre- or postprocessing commands are converted to the UTF-8 character set in remote Windows systems and that more characters may therefore be required in the remote system, see also section “Entering commands”.

In the case of preprocessing openFT transfers the data output by the command to standard output as a file.

In the case of postprocessing openFT reads the transferred data from the standards input.

In the case of preprocessing, you can also pass the data to the %TEMPFILE variable and, in the case of postprocessing, read the data from the %TEMPFILE variable, see section “Preprocessing and postprocessing”.

If command execution takes longer than ten minutes, a timeout occurs on partners using openFT prior to V8.1 and command execution is regarded as having failed. On partners using openFT V8.1 and later, this restriction no longer applies.

The remote command processing in Unix or Windows systems is starting in the $HOME or Home directory of the user.

Note for Unix systems

The PATH variable is used as follows in the search path for preprocessing and postprocessing commands in Unix systems:

- Standard instance: *:/opt/openFT/bin:/bin:/usr/bin:/opt/bin*
- Other instance: *:/var/openFT/instance/openFT/bin:/bin:/usr/bin:/opt/bin* where *instance* is the name of the relevant instance.

This means that the system first searches in the current directory (first ":").

Before calling a "real" preprocessing or postprocessing command you can switch to another directory as follows:

cd *path-name*;*command*

path-name is then used as the current directory. There must not be a blank between the semicolon and the command.

Note for local and remote Windows systems

path-name must not be a directory which is addressed using a UNC name. Exception: The UNC checking is deactivated on the system on which the command is to be executed. To do this, the registry value described under *https://support.microsoft.com/de-de/kb/156276* has to be generated.

partner

partner is the name of the partner system in the partner list or the address of the partner system. For details on address specifications, see section “Specifying partner addresses”.

transfer admission | **@n** | **@d** |

user ID [**,**[account][**,**password]]]

In order to be able to send a file to a remote system or to fetch one from it, you must furnish the remote system with proof of identity. For this purpose, you will need login authorization in the syntax valid for the remote system. You can specify this transfer admission

- as an FTAC transfer admission if FTAC is used in the remote system,
- or as a login/LOGON authorization in the syntax used by the remote system (user ID, possibly together with account or password).

For details, see section “Entering the authorization data for the partner system”.

@n for *transfer admission*

By entering *@n* you specify that the remote system requires no login authorization.

@d for *transfer admission*

Specifying *@d* (blanked transfer admission) causes openFT to query the transfer admission on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the transfer admission.

A binary password or binary transfer admission must be specified in hexadecimal format, see section “Entering commands”.

password not specified

Omitting the password necessary for admission causes openFT to query the password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

Nevertheless, you have to specify the commas, e.g.:

```
ncopy file partner!file user-id,,
```

or

```
ncopy file partner!file user-id,account,
```

neither *transfer admission* nor *user ID* specified

causes the same as *@d*, i.e. openFT queries the transfer admission on the screen after the command is entered. Your (blanked) entry is always interpreted as transfer admission and not as user ID.

-p=[password]

If the file in the remote system is protected by a write password, you must enter this password when sending a file. If the file is protected by a read password, then this password must be specified when fetching a file from the remote system.

A binary password must be entered in hexadecimal format, see section “Entering commands”. This is of relevance for links to openFT (BS2000), because BS2000 supports the definition of hexadecimal passwords.

password not specified

Specifying *-p=* causes openFT to query the write or read password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

-di

is specified, if the data integrity of the transferred file is to be checked by cryptographic means. With it, harmful data manipulations on the transmission network are identified. In case of an error openFT performs an error recovery for asynchronous transfer requests.

If the partner system does not support the check of data integrity (e.g. openFT < V8.1), the request is denied.

For requests with data encryption (option *-c*), data integrity is automatically checked. Testing mechanisms of the transfer protocols in use automatically identify transfer errors in the network. For this purpose you do not have to specify the *-di* option.

-lc=CCS name

(local coding) specifies the type of coding (character set) to be used to read or write the local file. *CCS name* must be known in the local system.

The default value is the character set defined by the FT administrator.

Details about the CCS name and the associated code tables can be found in the manual "openFT (Unix and Windows systems) - Installation and Operation".

-rc=CCS name

(remote coding) specifies the type of coding to be used to read or write the remote file. *CCS name* must be known in the remote system.

The default value is the character set defined in the remote system by means of XHCS (BS2000 systems) or the openFT operating parameters.

The option *-rc* is supported only by the openFT protocol and partners with openFT V10.0 or higher. Please note that not all partner systems support all the character sets that are possible in the local system. For details on CCS names and the associated code tables, see the manual "openFT (Unix and Windows systems) - Installation and Operation".

-rs='follow-up processing**'**

Here you can specify a command in the syntax of the remote system. Following a **successful transfer** operation, this command is executed in the remote system under the specified login.

Further information is given in the section “Commands for follow-up processing”.

-rf='follow-up processing**'**

Here you can specify a command in the syntax of the remote system. This command will be executed in the remote system under the specified login if a **transfer** operation that has already started is **cancelled**.

Further information is given in the section “Commands for follow-up processing”.

> ¡ If *-d* is specified (directory transfer) the follow-up processing is executed for all files in the directory.

-r=v[record length] | **-r=f**[record length] | **-r=u**[record length] | **-r=**record length

indicates the record format and the record length. This also enables records that are longer than the default value to be transferred. However, you must bear in mind that not every record length can be processed in all partner systems.

If you have selected file type *b* (binary), *record length* is the value for all records of the send file.

Maximum value: 65535 bytes.

With FTAM partners, the maximum record length specification is not valid unless the file type is set explicitly to *t*, *u* or *b*.

It is also possible to specify the record format, see command *ftmodf* , option *-rf*:

v	variable record length, *record length* determines the maximum value
f	fixed record length, *record length* then applies to all records
u	undefined record length

The combinations *-u -r=frecordlength* and *-u -r=urecordlength* are not permitted.

If *-r* is omitted then the following default values apply for the record format:

Option	Default value	Corresponds to
-b	u (undefined)	-r=u...
-t	v (variable)	-r=v...
-u	v (variable)	-r=v...

-tff=b | **-tff=s**

Specifies the format of the destination file.

b

The destination file is to be saved as a block-structured file. This means, for example, that a file can be transferred to BS2000 and stored there as a PAM file. If you specify *-tff=b*, you must also specify the option *-b* (binary).

s

The destination file is to be saved as a sequential file and the record format is to be retained. This allows an ISAM file or PAM file to be fetched from BS2000, for instance.

-tff=b must not be specified at the same time as *-trf=u*.

-trf=u

Specifies that the file is to be transferred as a sequential file and that the record format of the destination file is to be undefined, i.e. any existing record format of the send file is lost. If the file is being transferred to a BS2000 or z/OS system, one block is written per transfer unit.

-trf=u must not be specified at the same time as *-tff=b*.

Neither *-tff* nor *-trf* specified

The destination file is to be stored in the same format as the send file.

-tb=n | **-tb=f** | **-tb=a**

Activates/deactivates tabulator expansion and the conversion of blank lines into lines with one character for a single output send request.

The following parameters are provided:

n (on)

Tabulator expansion and blank line conversion are activated.

f (off)

Tabulator expansion and blank line conversion are deactivated.

a (automatic, default value)

Tabulator expansion and blank line conversion are activated if a file is sent to a BS2000, OS/390, or z/OS system.

No tabulator expansion or blank line conversion is performed for outbound receive requests. If *ncopy* is used as a preprocessing command, then tabulator expansion and blank line conversion are always deactivated.

The following parameters *-av, -ac*, *-am,* and *-lq* are provided exclusively for communication with FTAM partners. openFT thus supports the parameters defined in the FTAM standard. These parameters enable you to define the attributes of the destination file while issuing a file transfer request.

These parameters are ignored for requests involving openFT partners, but the file transfer is still carried out.

-av=i | **-av=d**

Indicates the availability of the destination file. This parameter can have one of two values: *immediate* or *deferred*. A file may be *deferred* if it has been archived, for example. The partner is responsible for interpreting the term *deferred*. The FTAM partner conventions must therefore be observed here.

The following values are possible:

i

The destination file attribute is set to *immediate*.

d

The destination file attribute is set to *deferred*.

av is not available for requests involving FTAM partners that do not support the storage group. In this case, the request is executed, but the entry for *av* is ignored.

-*av* not specified

The availability file attribute is set to a system-specific default value. In this case, this is the value *immediate*.

-ac=new account

With FTAM partners, this indicates the number of the account to which file storage fees are to be charged. This parameter must be set in accordance with partner system conventions.

ac is not available for requests involving FTAM partners that do not support the storage group. In this case, the request is executed, but the entry for ac *is* ignored.

-am=[r][i][p][x][e][a][c][d] | -am=@rw | -am=@ro

This sets the access rights of the destination file, provided the security group is available.

The following values can be specified:

r, *i*, *p*, *x*, *e*, *a*, *c*, *d*, any combination of these values, *@rw*, or *@ro*.

r

means that the file can be read.

r not specified

The file cannot be read.

i

means that data units, such as records, can be inserted in the file.

i not specified

No data units can be inserted in the file.

p

means that the file can be overwritten.

p not specified

The file cannot be overwritten.

x

means that data can be appended to the file.

x not specified

The file cannot be extended.

e

means that data units, such as records, can be deleted from the file.

e not specified

No data units can be deleted from the file.

a

means that the file attributes can be read.

a not specified

The file attributes cannot be read.

c

means that the file attributes can be changed.

c not specified

The file attributes cannot be changed.

d

means that the file can be deleted.

d not specified

The file cannot be deleted.

@rw

is the short form of the common access rights *read-write* (*rpxeacd*), and thus simplifies input.

@ro

is the short form for the common access rights *read-only* (*rac*), and thus simplifies input.

In Unix systems or in BS2000, only the following access rights can be set for a file:

Access mode	Short form	Unix system	BS2000	Access rights
rpxeacd	@rw	rw*	ACCESS=WRITE	read-write
rac	@ro	r-*	ACCESS=READ	read-only
pxeacd		-w*	only with BASIC-ACL (Access Control List)	write-only
ac		--*	only with BASIC-ACL (Access Control List)	none

* The x bit is not changed by *ncopy*.

am is not available for requests involving FTAM partners that do not support the security group. In this case, the request is executed, but the entry for *am* is ignored.

-am not specified

The default values of the FTAM partner system apply.

-lq=legal qualification

This specifies a legal qualification for the destination file (similar to a copyright). This may not exceed 80 characters.

lq is not available for requests involving FTAM partners that do not support the security group. The request is executed, but the entry for *lq* is ignored.

-cp=[password]

If a password is required in order to create a file on a remote system, this password must be specified here. It can be up to 64 characters long. A binary password must be specified in hexadecimal format, see section “Entering commands”.

If you do not specify a file creation password, but you do enter a file access password for *-p=password*, the file creation password is identical to the file access password. The file creation password is of no significance when retrieving a file.

password not specified

Specifying *-cp=* causes openFT to query the file creation password on the screen after the command is entered. Your entry is not displayed to prevent unauthorized persons from seeing the password.

-fnc=t | -fnc=c (file name coding)

specifies the encoding mode for file name and follow-up processing.

t (transparent, default value)

Specification of the remote file name and follow-up processing for the remote system in transparent mode (compatible to the previous versions).

c (character)

Specification of the remote file name and follow-up processing for the remote system in character mode. They are interpreted according to the character code of the remote system, i.e. for Unix partners according to the openFT operating parameter option (*ftmodo -fnccs)* that has been set there.

-fnc=c is only permitted for openFT partners as of openFT V12.1.

-md (modification date)

The modification date of the send file is taken over for the destination file provided that the destination system supports this. If the destination system does not support this function then the request is rejected.

-md not specified

The behavior is the same as in openFT V11.0 or earlier: On Unix and Windows systems as well as under POSIX (BS2000), the modification date of the send file is taken over. On BS2000 with DMS, the current time is taken over as the modification date.

Examples

1. The text file *airplane* is sent to the login name *bill* with account number *a1234ft* and password *C'pwd'* in the BS2000 computer with the partner name *bs2r1*, where it is to be printed out.

   Local Unix system:

   ```
   ncopy airplane bs2r1!% bill,a1234ft,C\'pwd\' \
    -rs="/PRINT-FILE airplane,LAYOUT-CONTROL=*PARAMETERS\
   (,CONTROL-CHARACTERS=EBCDIC)"
   ```

   Local Windows system:

   ```
   ncopy airplane bs2r1!% bill,a1234ft,C'pwd'
     -rs="/PRINT-FILE airplane,LAYOUT-CONTROL=*PARAMETERS
   (,CONTROL-CHARACTERS=EBCDIC)"
   ```

2. A file is to be fetched from BS2000, where openFT-AC is running, to the local Unix or Windows system. The file name has been predefined in an FT profile, which can be accessed with the authorization *'onlyforme'.* In the local system the file is to be stored under the name *stat.b*. It is to be transferred as an unstructured binary stream. The data is to be compressed for file transfer.

   ```
   ncopy -b bs2! stat.b 'onlyforme' -k
   ```

3. The text file

   letter is sent to the login name *joe* with the password *pass* in the Unix system with the host name *xserver*. The file should then be printed out in the remote Unix system.

   ```
   ncopy letter xserver!letter joe,,pass -rs="lpr letter"
   ```

4. The text file *letter* is sent to the login name *jim* with the password *jimfun* in the FTAM partner with the host name *ftampart.*

   ```
   ncopy letter ftam://ftampart:102.FTAM.FTAM.FTAM!letter jim,,jimfun
   ```

   The FT administrator can use *ftaddptn* to enter the partner in the partner list in order to shorten the command, e. g.

   ```
   ftaddptn ftamp1 ftam://ftampart:102.FTAM.FTAM.FTAM
   ```

   The *ncopy* command is then:

   ```
   ncopy letter ftamp1!letter jim,,jimfun
   ```

5. The file *data* is sent from a Windows system *pc123* to a Windows system *pc234* with the transfer admission *topsecret* and stored there under the name *dat.txt*. Then, as follow-up processing, the procedure *evaluate* is started in the remote system if transferred successfully. The procedure contains the file name *dat.txt*, the partner *pc234* and the message number as parameters. The parameters are specified using variables.

   ```
   ncopy data pc234!dat.txt topsecret
     -rs="cmd.exe /c evaluate.bat %FILENAME %PARTNER %RESULT"
   ```

6. The text file *locfile* is sent to the login name *charles* with the password *secret* in the Unix system *ux1*. There, it is stored under the file name *remfile*. As follow-up processing, the file is printed if transferred successfully; if not, the *prog* program is started in the remote system. This program receives the name of the source file and the message number as parameters. The parameters are specified using variables.

   Local Unix system:

```
ncopy locfile ux1!remfile charles,,secret -r=100 \
 -rs='lpr remdfile' \
 -rf='prog %FILENAME %RESULT'
```

Local Windows system:

```
ncopy  locfile  ux1!remfile charles ,,secret -
r=100
  -rs='lpr remdfile'
  -rf='prog %FILENAME %RESULT'
```

If file transfer is not successful, e.g. because the record length was greater than 100 bytes, follow-up processing is executed as follows:

```
prog remfile 2210
```

7. In the local Unix system, the *ls* command enables you to view a list of files in a directory on the screen. You want to store this information as a text file in the remote system *wx1* and give this file the name *unix.dir.* The userid is *smith* and the password *any.*

   ```
   ls | ncopy - wx1!unix.dir smith,,any
   ```

8. Data is sent from the keyboard to the user *smith* whose computer is *wx1* with the password *any*. The data is stored in the file *MEMO*.

   ```
   ncopy - wx1!memo smith,,any
   ```

   Then you enter via the keyboard:

   ```
   Will be in headquarters at 4 p.m.
   Regards, Johnson
   ```

   You terminate entry by pressing the following keys:

   <END> or CTRL+D (Unix systems)

   CTRL+Z at the start of a line, followed by Return (Windows systems).

   The successful transfer is indicated by the message:

   ```
   ncopy: request 65786. File '*STDIN' transferred
   ```

9. This example for Unix systems shows how to bypass the restriction of follow-up processing commands to 1000 characters in total.

   The text file *finalreport* is sent to the central system *ux1* for storage under the login name *branch1* with password *a-to-z* under the file name *helpfile*. After successful transfer, the file is stored in the directory */home /branch1/file.smith* under the file name *finalreport*, printed out, and appended to the file *file.smith*. The file *file. smith* is then sent to the boss's computer *bosscomp*. In the event of errors, a detailed entry is to be written to the log file *errlog* in the remote system *ux1*.

   The restriction is bypassed here by placing the follow-up processing commands in procedures. *succproc* is the procedure for remote follow-up processing if the transfer is successful, and *failproc* is the procedure for remote follow-up processing if the transfer fails.

   ```
   ncopy finalreport ux1!helpfile branch1,,a-to-z\
    -rs='succproc' \
    -rf='failproc'
   ```

If file transfer is successful, the procedure *succproc* is executed in the remote system under the login name *branch1*. This contains the following commands:

```
cp helpfile /home/branch1/file.smith/finalreport
lpr -ws=G005 -  pb3 /home/branch1/file.smith/finalreport
cat helpfile >> /home/branch1/file.smith/file.smith
    ncopy /home/branch1/file.smith/file.smith bosscomp!file.smith \
secretary,,secret
rm helpfile
```

If file transfer is not successful, the procedure *failproc* is executed in the remote system under the login name *branch1*. This contains the following commands:

```
echo "In the event of an error, a detailed message  " >> errlog
echo "should be written to the log file.            " >> errlog
echo "In this case, you can assume that the file    " >> errlog
echo "transfer failed.                              " >> errlog
```

Please note here that the *succproc* and *failproc* procedures must be executable (*rwx*------) in the remote system, or called with *sh* (e.g. *-rs='sh succproc'*).

10. The file

    filex on a Windows system is transferred to the user *joe* in the Windows domain

    dom1.

    ```
    ncopy filex system!filex dom1\joe,,passw
    ```

11. Example of specifying UNC names on Windows systems:

    ```
    ncopy \\Win01\dir\file ux2!file remoteaccess
    ```

12. Example of the use of preprocessing commands:

    The remote Unix system *ux1* possesses a tar archive *tar.all* under the ID *karlotto* with the password *secret*. The file *file.1* is to be retrieved from this tar archive into the local

    system and saved in the local file *file.loc*.

    ```
    ncopy ux1!"|ft_tar -xOf tar.all file.1" file.loc \ karlotto,,secret
    ```

    ft_tar -xOf retrieves the file from the archive and writes it to *stdout*. The file *file.1* is then therefore not available under the remote ID.

13. Example of the use of preprocessing commands on a Windows system:

    - All the error messages of the remote file *log.txt* on the Unix system *ux1* are appended to the local file l*ogux1.txt*.

      ```
      ncopy ux1!"|fgrep ERROR log.txt" logux1.txt geterrorux1 -e
      ```

    - The last five log records (return code not equal 0) from the Windows computer *Win01* are displayed on the screen.

      ```
      ncopy win01!"|ftshwl -nb=5 -rc=@f" - charles,,secret
      ```

14. Example of the use of postprocessing commands on a Unix system:

    The local file *file* is to be entered in the tar archive *tar.all* under the name *file.x*. The tar archive *tar.all* is located on the remote computer *win1* under the transfer admission *tarremote*. After being entered in the tar archive, the file is to be deleted in the remote system.

```
ncopy file win1!"|cp %TEMPFILE file.x;ft_tar -uf tar.all \
      file.x --remove_files" tarremote
```

15. Example for illustrating the use of preprocessing and postprocessing commands.

    The local directory *dir* and all its files are to be transferred to the remote Unix host with the symbolic name *ftunix* . The current openFT version is also running on the remote host. After the transfer, *dir* should be available on the remote system under the user ID that owns the *copydire* transfer admission.

    ```
    ncopy "|ft_tar -cf - dir" ftunix!"|ft_tar -xf - " copydire -b
    ```

    The *dir* directory must be located on the local computer in the home directory (on Unix systems: value of the *$HOME* variable). Please note that no file name prefixes may be defined in the profile.

16. Example of the use of preprocessing and postprocessing commands on a Unix system:

    At the remote computer *ux1*, you first want to compress the remote file *remfile* under the user ID *karlotto* with the password *secret* (using the command *compress -c remfile*). The result is transferred and written to the local system's standard output (-). Here, the output is transferred via a pipe to the *uncompress -c* command and saved in the local file *locfile*.

    ```
    ncopy -b ux1!"|compress -c remdate" "\
    |uncompress -c>locfile" karlotto,,secret
    ```

    If the command is rejected with `Remote System: Exitcode 2` in the case of preprocessing/postprocessing then the cause may lie in the remote system's *compress* command. In some Unix systems, the command supplies return code 2 even though it was successful.

    You can avoid this problem by extending the preprocessing command with 'exit 0':

    ```
    ncopy -b ux1!"|compress -c remdate;exit 0" "\
    |uncompress -c>locfile karlotto,,secret
    ```

17. Example for FTP connection on a Unix system:

    In the remote system with the host name *wini2* there is only one FTP server. The file *all_files* under the ID *user1* with the password *usrpass* is to be fetched into the local system. Here, it is to be stored in the directory *save_files* under the partner-specific name *wini2.all_files*.

    ```
    ncopy ftp://wini2!all_files save_files/%PARTNER.all_files \
    user1,,usrpass
    ```

18. Directory transfer

    The current directory contains two subdirectories. These are to be sent to the Windows system *ftwin*:

    ```
    ncopy -d * ftwin!C:\Software\% smith,,secret
    ```

    The remote directory *C:\Software* must exist. The directories are stored under the directory *Software* with identical structure and file names. The following message is output after successful transfer:

    ```
    ncopy: Requests carried out; 2 Directories transferred
    ```

4 Messages

The openFT messages are sent to you as a result code (Unix systems: shell variable $?, Windows systems: shell variable *errorlevel* or *%errorlevel%*) and as text to the screen *stderr*.

The messages appear in the language that is set for openFT (English or German). Please refer to the section "Switching language interfaces" in the manual "openFT (Unix and Windows systems) - Installation and Operation" for a detailed description how you can switch the language.

If multiple file transfers are running in parallel, you can use the request ID to assign the error message to the correct file transfer.

<local file> or <remote file> specifies the file name.

<Request id> specifies the number of the file transfer request. openFT informs you of this number on confirmation of request receipt.

There follows a description of the error messages output by openFT together with the associated exit codes, meanings and measures where appropriate.

The description has the following format:

exit code Message text
meanings and measures as appropriate

4.1 openFT messages

4.1.1 Messages applying to all commands

0 The command was successful

3 The command was cancelled as the result of a response to a query

4 A syntax error occurred during command processing

225 Information output canceled

Meaning:
A show command was interrupted, for example.

Measure:
Repeat the command.

229 ftaddptn/ftmodo: License infringement:
The following causes are possible:
Maximum number of partners reached or exceeded
Dynamic partners not permitted

Meaning:
There are already more partners entered in the partner file than are permitted by the license or the current license does not permit dynamic partners.

Measures:
Delete partners from the partner file (see *ftremptn*) or install another license (see *ftaddlic*)
Install another license (see *ftaddlic*)

236 Current instance '<instance>' no longer found

Meaning:
The command was rejected. The instance '<instance>' could not be found.

250 An internal error occurred during command processing

251 Command aborted with core dump

Measure on Windows systems:
In the Application section of the Event Viewer there is an error event for openFT that specifies the file name and directory under which the dump has been saved. Where necessary, contact Customer Service and send the dump for further analysis.

253 Current openFT instance is invalid

Meaning:
During command processing a defined instance was found to be invalid

254 Command client error

(only on Windows systems)

Meaning:
An error occurred while connecting a command to the openFT service

255 ftexec/ftadm command failed;
This can also be issued as an exit code from remote command execution.

Meaning:
Remote execution of the command with *ftexec* or *ftadm* failed.
ftexec: Protocol stack '<openFT|FTAM|FTP>' not licensed or not installed

Measure:
Install license key (see *ftaddlic*)

Messages applying to file transfer, file management and remote administration commands

4.1.2 Messages for file transfer and file management commands

All the messages listed below, with the exception of the message with exit code 5, can also be output during logging. In this case, however, the specified code is increased by 2000, e.g. 2169 instead of 169.

5 Request <Request id>. File '<local file>' transferred

Meaning:
The file transfer request <Request id> has been successfully completed.
Follow-up processing has been started for both the local system and the remote system, as requested, provided no error occurred. Local errors are indicated as a message.

6 Request <Request id>. Directory '<local file>' transferred

Meaning:
The directory transfer request <Request id> has been successfully completed.
Follow-up processing has been started for both the local system and the remote system, as requested, provided no error occurred. Local errors are indicated as a message.

14 No file attribute changes requested

Meaning:
No further file attributes besides the file name were specified.

Measure:
Enter the desired file attributes in addition to the file name.

15 openFT is not authorized to execute requests for this user

Meaning:
The user has not informed openFT of his or her logon password or an openFT command has been called by a user other than the user under which the openFT service is running when a service has been started under user rights.

Measure:
Store the password or call the command from the ID under which the openFT service is running in another operating mode.

16 Directory '<local file>' is not empty

17 File attributes do not match request parameters

Meaning:
The specified attribute combination is not permissible.

Measure:
Specify a permissible combination.

18 Attributes could not be modified

Meaning:
The properties of the file could not be changed as specified in the command. The following reasons are possible:

For the remote file:

- No access rights to the file.
- The required combination of access rights is not supported by the remote system.
- If the remote system is a BS2000: the file is protected by ACL.

For the local file:

- No access rights to the file.
- The requested transfer attributes are not compatible with the BS2000 properties of the file.

19 '<local file>' could not be created

Meaning:
The command was not executed because the file owner and user requesting the creation of a receive file are not the same.

Measure:
Match the user ID in the receiving system's transfer admission to the ID of the receive file's owner.
Repeat the command.

20 '<local file>' not found

Meaning:
The command was not executed because the send file is not in the catalog or on a volume of the local system. The command was not executed because either the send file is not/no longer or the receive file is no longer in the catalog or on a volume of the relevant system.

Measure:
Correct the file name, read in file from tape or restore the send file.
Repeat the command.

21 CCS name unknown

Meaning:
The request could not be carried out because the CCS names of the send and receive files could not be mapped to each other or because the partner system does not support the transparent receipt of files.

22 Higher-level directory not found

Meaning:
In the case of a receive request, the local file could not be created because the specified path does not exist.

Measure:
Create or correct the path for the receive file and repeat the command.

23 '<local file>' already exists

Meaning:
The command was not executed because an existing receive file cannot be created again with option *-n*.
Option *-n* may also have been set due to a restriction in the access authorization used.

Measure:
Either delete the receive file and repeat the command, or repeat the command specifying option *-o* or using different access authorization.

24 Transfer of file generation groups not supported

Meaning:
The command was not executed because the FT system only transfers single file generations.

Measure:
Repeat the command using the name of a single file generation.

25 Error accessing '<local file>' <2>

Meaning:
<2>: DMS error, possibly the transfer ID.
The FT system continues to run after the message has been issued.

Measure:
Take the appropriate action in accordance with the error code.

26 Resulting file name'<local file>' too long

Meaning:
The relative file name was specified in the transfer request.
The absolute file name completed by openFT is longer than permitted.

Measure:
Shorten the file name or path and repeat the command.

27 No file or directory name specified

Meaning:
The command was not executed because the file name was neither specified explicitly nor by the 'transfer admission' used.

Measure:
Repeat the command, specifying the file ID explicitly or a transfer admission that defines the file ID.

28 Invalid management password

29 '<local file>' not available

Meaning:
The command was not executed because the volume for either the send file or the receive file is not mounted, unknown or reserved, or the file extends over more than one private disk.

Measure:
Inform the operator if necessary. Repeat the command.

30 Home directory not found

31 Renaming not possible

32 Not enough space for '<local file>'

Meaning:
The command was not (fully) executed because

- the permissible storage space on the receive system is used up for the user ID specified in transfer admission, or
- the send file contains too long a sequence of empty blocks, or
- the primary and/or secondary allocation of the password-protected receive file is too small.

The receive file can not be created/extended after the problem occurs.

Measure:
Take the appropriate action depending on the cause of the error:

- delete all files no longer required on the receive system, or
- ask the system administrator to allocate more storage space, or
- remove empty blocks from the send file, or
- increase the receive file's primary/secondary allocation.

If option *-e* is specified, restore the receive file.
Repeat the command.

33 File owner unknown

Meaning:
The command was not executed because the owner of either the send file or the receive file was not defined in the local system or because the file owner and the user requesting the creation of a receive file are not the same.

Measure:
Define the file owner, correct transfer admission or file name.
Repeat the command.

34 Invalid file password

Meaning:
The command was not executed because the password for the send file or the receive file is missing or incorrect.

Measure:
Correct the password in the file description or the command.
Repeat the command.

35 File locked to prevent multiple access

Meaning:
The command was not executed because either the send file or the receive file is already locked by another process to prevent it from being updated simultaneously.

Measure:
Repeat the command later or unlock the file.
After a system crash you may need to verify files that are not closed correctly.
If the lock is caused by an FT request, it will be canceled automatically when the request is finished.

36 Retention period of file not yet expired

Meaning:
The command was not executed because the retention period protecting the receive file against overwriting has not yet expired (RETENTION PERIOD).

Measure:
Correct the transfer direction, retention period or file name.
Repeat the command.

37 '<local file>' is read only

38 File structure not supported

39 Syntax error in resulting file name '<local file>'

Meaning:
The local file cannot be accessed because, for example, the absolute file name is too long.

Measure:
Shorten the path or file name. Repeat the command.

40 Transparent file transfer not supported

Meaning:
The request could not be carried out because the CCS names of the send and receive files cannot be mapped to each other or because the partner system does not support the receipt of files in a transparent format.

41 Request queue full

Meaning:
The command was not executed because the maximum number of permissible file transfer requests has been reached.

Measure:
Notify the FT administrator. Repeat the command later.

42 Extension of file not possible for transparent transfer

Meaning:
The command could not be executed because it is not possible to add to a file in a transparent transfer.

Measure:
Start transfer without option *-e*.

43 Access to '<local file>' denied

Meaning:
The command was not executed because either the send file or the receive file only permits certain access modes (e.g. read only).

Measure:
Correct the file name or file protection attributes. Repeat the command.

44 Follow-up processing exceeds length limit

Meaning:
Prefix + suffix (from prof) + local follow-up processing together are too long.

Measure:
Shorten the follow-up processing, or use procedures.
Repeat the command.

45 Processing admission invalid

Meaning:
The command was not executed because the specifications in one of the PROCESSING-ADMISSION operands were incorrect.

Measure:
Define the required PROCESSING ADMISSION or correct it.
Repeat the command if necessary.

46 Local transfer admission invalid

Meaning:
The command was not executed because the specifications in one of the transfer admission operands were incorrect.

Measure:
Define the required transfer admission or correct it.
Repeat the command if necessary.

47 Request rejected by local FTAC

Meaning:
The command was not executed because the request was rejected by the FTAC due to a lack of authorization.

Measure:
Use the return code in the log record to determine and remove the cause.
Repeat the command.

48 Function not supported for protocol '<partner protocol type>'

Meaning:
The desired function is not available for the selected protocol.

Measure:
Select a different protocol.

49 Remote follow-up processing not supported

Meaning:
Remote follow-up processing is only available for the openFT protocol.

Measure:
Select a different protocol, or specify follow-up processing by means of an admission profile.

50 Data integrity check not supported

Meaning:
The partner system does not support the data integrity check function.

Measure:
Repeat the request without a file integrity check.

51 User data encryption not possible for this request

Meaning:
The partner system does not support the data encryption function.

Measure:
Repeat the request without data encryption or install openFT-CR (or have it installed) on the remote system.

52 Administration request rejected by remote administration server

Meaning:
The administration request was rejected by the remote administration server because it clashes with the settings in the configuration file of the remote administration server.

The ADM administrator can determine the precise reason for rejection from the associated ADM log record on the remote administration server.

Possible reason codes:

7001 The administrator ID is invalid. It was not possible to determine a valid administrator ID from the user ID or the profile name in the configuration data of the remote administration server.

7002 The routing information is invalid. The specified openFT instance specified in the routing information could not be found in the configuration data of the remote administration server.

7003 The specified remote administration command is invalid. The remote administration server rejects the specified command because it is not a supported remote administration command.

7101 Infringement against the access rights list. On checking the access rights, the system identified that the administrator ID has not been assigned the necessary rights in the configuration data of the remote administration server to be able to execute the valid remote administration command on the specified openFT instance.

7201 Infringement against the maximum command length. In particular in the case of BS2000 commands, the remote administration server replaces the shortest command names, which are guaranteed by openFT, by the full command names. If this replacement of the command name causes the entire remote administration command to become longer than the maximum command length of 8192 characters, the command is rejected.

Measure:
Have the ADM administrator carry out the necessary adjustments to the configuration data or check the command. Repeat the changed command if necessary.

54 Invalid command

Meaning:
The specified command is not a command that is permitted to be executed on the specified system using the remote administration facility.

Measure:
Specify an admissible command or add the missing routing information.
Repeat the command.

55 Transfer of protection attributes not supported

56 Syntax error in partner name '<partner>'

57 openFT is not authorized to execute administration requests

Meaning:
openFT is not (no longer) authorized to process administration requests. This is, for example, the case if a remote administration server has been demoted to a normal server (*ftmodo -admcs=n*) or if commands that are only allowed to be executed on a remote administration server are processed by an openFT instance that has not been configured as a remote administration server.

70 Request <Request id>. openFT is no longer authorized to execute requests for this user

Meaning:
The user has not informed openFT of his or her logon password or an openFT command has been called by a user other than the user under which the openFT service is running when a service has been started under user rights.

Measure:
Store the password or call the command from the ID under which the openFT service is running in another operating mode.

71 Request <Request id>. User data encryption not installed

Meaning:
The user data encryption function cannot be used unless openFT-CR is installed.

Measure:
Use openFT-CR.

72 Request <Request id> has been canceled

Meaning:
The FT request was canceled because the *ftcanr* command was specified, or the time specified in the transfer request has been reached.

Follow-up processing has been started for the local system, provided no error occurred. Follow-up processing is started for the remote system once all the resources are allocated. Local errors are indicated by the message FTR0050 at the start of follow-up processing.

73 Request <Request id>. Encryption error

Meaning:
Encryption not possible.

74 Request <Request id>. '<local file>' could not be created

Meaning:
The command was not executed because the file owner and user requesting the creation of a receive file are not the same.

Measure:
Match the user ID in the receive system's transfer admission to the ID of the receive file owner.
Repeat the command.

75 Request <Request id>. Higher-level directory no longer found

76 Request <Request id>. I/O error for '<local file>'

Meaning:
The file can no longer be accessed. It may have been deleted during a transfer.

Measure:
Repeat the request.

77 Request <Request id>. File now locked to prevent multiple access

Meaning:
The command was not executed because the send file or the receive file is already locked by another process so that it cannot be simultaneously updated.

Measure:
Repeat the command later or unlock the file.
After a system crash you may need to verify files that are not closed correctly.
If the lock is caused by an FT request, it will be released automatically when the request is finished.

78 Request <Request id>. '<local file>' no longer available

Meaning:
The command was not executed because the volume for either the send file or the receive file is not mounted, unknown or reserved, or the file extends over more than one private disk.

Measure:
Inform the operator if necessary. Repeat the command.

79 Request <Request id>. '<local file>' no longer found

Meaning:
The local send or receive file can no longer be accessed because, for example, it was deleted during an interruption of the openFT system.

Measure:
Restore the file.
Repeat the command.

80 Request <Request id>. Home directory no longer found

81 Request <Request id>. '<local file>' gets no more space

Meaning:
The command was not (any further) executed because

- the permissible storage space on the receive system for the user ID specified in transfer admission has been used up, or
- the send file contains too long a sequence of empty blocks, or
- the primary and/or secondary allocation of the password-protected receive file is too small.

The receive file can not be created/extended once this problem occurs.

Measure:
Take the appropriate action depending on the cause of the error:

- delete all files no longer required on the receive system, or
- ask the system administrator to allocate more storage space, or
- remove empty blocks from the send file, or
- increase the receive file's primary/secondary allocation.

If option *-e* is specified, restore the receive file.
Repeat the command.

82 Request <Request id>. File owner no longer known

Meaning:
The command was not executed because the owner of the send file or receive file is not defined on the relevant system or because the file owner and the user who wants to create a receive file are not the same.

Measure:
Define the file owner, or correct transfer admission or file name.
Repeat the command.

83 Request <Request id>. Pre-/post-processing error

Meaning:
The command executed as part of local pre-/post-processing returned an exit code other than 0.

Measure:
Correct and repeat the command.

84 Request <Request id>. Exit code <2> for pre-/post-processing

Meaning:
The command executed as part of local pre-/post-processing returned the exit code <2>.

Measure:
Correct the command using the exit code <2> and issue it again.

85 Request <Request id>. File password no longer valid

Meaning:
The command was not executed because the password for send file or the receive file is missing or incorrect.

Measure:
Correct the password in the file description or the command.
Repeat the command.

86 Request <Request id>. '<local file>' is now read only

87 Request <Request id>. File structure error

Meaning:
The command was not executed due to a file structure error.
File structure errors include:

- The attributes of the send file are incomplete.
- The data of the send file is incompatible with its structure attributes.
- The records of the send file are too long.
- If *-e* is specified, the send file and receive file have different structures (e.g. fixed-/variable-length records).
- The send file or receive file is a member of an old LMS library (not PLAM).
- The source file has an odd block factor (e.g. BLKSIZE=(STD,1)) and the receive file is to be stored on an NK4 pubset.

Measure:
Correct the file or file attributes. If option *-e* is specified, restore the receive file.
Repeat the command.

88 Request <Request id>. NDMS error <2>

Meaning:
The request was rejected because the partner system currently does not have the resources available to accept requests.

Measure:
Repeat the request a little later.

89 Request <Request id>. Recovery failed

Meaning:
The restart attempts were unsuccessful (for example, a pre-/post-processing command could not be completed before the termination of openFT).

Measure:
Repeat the command.

90 Request <Request id>. Error in file transfer completion

Meaning:
An error occurred during the final phase of the file transfer.
If it was a long transfer, the recipient is advised to check if the file has still been transferred correctly.
However, error follow-up processing will be started if it was specified.

Measure:
Repeat the request, if necessary.

91 Requests only partially completed; <1> of <2> files were transferred

Meaning:
In the case of a synchronous send request with wildcards, not all files were successfully transferred.

Measure:
Transfer unsuccessfully transferred files again.

92 Request <Request id>. Access to '<local file>' no longer permissible

93 Request <Request id>. FTAM error <2>

94 Request <Request id>. Retention period of file not yet expired

95 Request <Request id>. Extension of file not possible for transparent transfer

96 Request <Request id>. File structure not supported

97 Request <Request id>. Resulting file name '<local file>' too long

99 Request <Request id>. Transfer of protection attributes not supported

108 Request <Request id>. Remote system not accessible

Meaning:
The command could not be accepted because the partner system is currently not available.

Measure:
Repeat the command later. If the error persists, contact the system or network administrator.

109 Request <Request id>. Connection setup rejected by local transport system

110 Request <Request id>. Data integrity check indicates an error

Meaning:
The integrity of the data was violated.

111 Encryption/data integrity check not possible. Encryption switched off

Meaning:
There is no key pair set or the key length was set to 0. Requests can only be carried out without data encryption or a data integrity check.

Measure:
Repeat the request without data encryption, create a key or set a key length >0.

112 Request <Request id>. Data integrity check not supported by partner

Meaning:
The partner system does not support the data integrity check.

Measure:
Repeat the request without a data integrity check.

113 Request <Request id>. User data encryption not possible for this request

Meaning:
The partner system does not support the data encryption function.

Measure:
Repeat the request without data encryption or install openFT-CR (or have it installed) on the remote system.

114 Request <Request id>. Identification of local system rejected by remote system '<partner>'

Meaning:
For security reasons or because of an inconsistency, the partner did not accept the instance identification of the local system (for example, because in the partner list both the instance identification and migration identification %.processor.entity occur for different partners).

Measure:
Ensure that the local identification has been entered correctly on the partner system and has not been assigned to a different partner.

115 Request <Request id>. Interrupted by remote system

116 Local application '<1>' not defined

Meaning:
The local application is not defined in the transport system, or the tnsxd process is not running in the Unix system.

Measure:
Make the local application known to the local transport system, or start the tnsxd process.

117 Local application '<1>' not available

118 Request <Request id>. Authentication of local system failed

Meaning:
The local system could not be authenticated by the partner system.

Measure:
Give the current public key file to the partner and name it correctly there. Repeat the command.

119 Request <Request id>. Local system unknown in remote system

Meaning:
The local system is not known on the partner system (e.g. BS2000 or z/OS system).

Measure:
Make the local system known on the partner system and repeat the command.

120 Remote system '<partner>' unknown

Meaning:
The partner specified as the remote system cannot be expanded to an address on the local system.

Measure:
Correct the specification for the partner or add the partner to the partner list and repeat the command.

121 Request <Request id>. Authentication of partner failed

Meaning:
The remote system could not be authenticated by the local system.

Measure:
Get the current public key file from the partner and name it correctly.

122 Request <Request id>. FT session rejected or disconnected.
Reason <2>

123 Request <Request id>. OSS call error <2>

Meaning:
The command was not executed because the session instance detected a communication error.
<2>: error code.

Measure:
Take the appropriate action in accordance with the error code.

124 Request <Request id>. No free connection

Meaning:
No more transfers are possible because the maximum number of simultaneous transfers has been reached.

Measure:
Check whether the transport system is working (or have it checked).

125 Request <Request id>. Connection lost

Meaning:
No data transfer took place because of a line interrupt or a line protocol error.

Measure:
Repeat the request.

126 Request <Request id>. Transport system error. Error code <2>

Meaning:
An error occurred in the transport system during processing of a /START-FT command or ftstart or a file transfer or file management request.

Measure:
Take the appropriate action in accordance with the error code. Most often the occurrence of this message indicates that the partner addressed is not known to the transport system. Contact system administrator to make sure there is an entry for the partner system.

127 Request <Request id>. No data traffic within <2> seconds

Meaning:
No data transfer took place within the period of seconds specified because, for example, the connection is interrupted, the partner is not sending and the local system is waiting for data.

Measure:
Repeat the request.

140 Request <Request id>. Remote system: openFT is not authorized to execute requests for this user

Measure:
If the remote system is a Windows system: Declare the password of the user to openFT (e.g. using the *ftsetpwd* command).

141 Request <Request id>. Remote system: Directory '<remote file>' is not empty

Meaning:
The command could not be executed because there are files in the specified directory of the partner system.

Measure:
Delete all the files in the directory first and repeat the command.

142 Request <Request id>. Remote system: File attributes do not match the request parameters

Meaning:
The command could not be executed because the file attributes on the remote system do not agree with the request parameters (e.g. a directory was specified instead of a remote file).

Measure:
Check the file name on the remote system and correct it.
Repeat the command.

143 Request <Request id>. Remote system: Attributes could not be modified

Meaning:
The properties of the file could not be modified as desired in the command. Possible reasons are:

For the remote file:

- No access rights to the file.
- The combination of access rights required is not supported by the remote system.
- If the remote system is a BS2000: the file is protected by ACL.

144 Request <Request id>. Remote system: File/directory '<remote file>' could not be created

Meaning:
The command was not executed because the file owner and user requesting the creation of a receive file are not the same.

Measure:
Match the user ID in the receive system's transfer admission to the ID of the receive file owner.
Repeat the command.

145 Request <Request id>. Remote system: CCS name unknown or not supported

Meaning:
The request could not be carried out because the CCS names of the send and receive files cannot be mapped to each other or because the partner system does not support the receipt of files in a transparent format.

146 Request <Request id>. Remote system: Higher-level directory not found

Meaning:
The command was not executed because the higher-level directory could not be found on the partner system.

Measure:
Create the directory on the remote system or correct the remote directory name and repeat the command.

147 Request <Request id>. Remote system: File/directory '<remote file>' already exists

Meaning:
The command was not executed. Possible reasons:

- The command was not executed because an existing receive file cannot be created with the *-n* option. *-n* may also have been set by a restriction in the access authorization used.
- *ftcredir*: The specified directory already exists.

Measure:
Either delete the receive file before repeating the command or reenter the command specifying option *-o* or using different access authorization.

148 Request <Request id>. Remote system: Transfer of file generation groups not supported

Meaning:
The command was not executed because the FT system can only transfer single file generations.

Measure:
Repeat the command using the name of a single file generation.

149 Request <Request id>. Remote system: Access error for ’<remote file>’ <3>

Meaning:
<3>: DMS error, possibly the transfer ID
The FT system continues to run after output of the message.

Measure:
Take the appropriate action in accordance with the error code.

150 Request <Request id>. Remote system: Resulting file name too long

Meaning:
A syntax error other than 'Mandatory parameter missing' (703) or 'keyword unknown' has been detected.

Possible reasons:

- Values assigned outside the valid range
- Invalid operand separators
- Invalid value assignment characters
- Partially qualified file names

Measure:
Repeat the command using the correct syntax.

151 Request <Request id>. Remote system: File locked to prevent multiple access

Meaning:
The command was not executed because either the send file or the receive file is already locked by another process to prevent it from being updated simultaneously.

Measure:
Repeat the command later or unlock the file on the remote system.
After a system crash in BS2000 you may need to call VERIFY for files not closed correctly.

If the lock is caused by an FT request, it will be released automatically when the request is finished.

152 Request <Request id>. Remote system: No file or directory name specified

Meaning:
The command was not executed because the file ID was neither specified explicitly nor by the transfer admission used.

Measure:
Repeat the command, specifying the file ID explicitly or using a transfer admission that defines the file ID.

153 Request <Request id>. Remote system: Invalid management password

154 Request <Request id>. Remote system: File/directory '<remote file>' not available

Meaning:
The command was not executed because the volume for either the send file or the receive file is not mounted, unknown or reserved, the file extends over more than one private disk, or an attempt has been made to transfer a file migrated by HSMS.

Measure:
Inform the operator if necessary or carry out an HSMS recall for the file. Repeat the command.

155 Request <Request id>. Remote system: File/directory '<remote file>' not found

Meaning:
The command was not executed because the send file is not or no longer in the catalog or on a volume of the remote system.

Measure:
Correct the remote file name, read the file in from tape or restore the send file.
Repeat the command.

156 Request <Request id>. Remote system: Home directory not found

157 Request <Request id>. Remote system: Renaming not possible

158 Request <Request id>. Remote system: Not enough space for '<remote file>'

Meaning:
The command was not executed (any further) because

- the permissible storage space on the receive system for the user ID specified in transfer admission has been used up, or
- the send file contains too long a sequence of empty blocks, or
- the primary and/or secondary allocation of the password-protected receive file is too small.

The receive file is no longer created/extended after the problem has occurred.

Measure:
Take the appropriate action depending on the cause of the error:

- delete all files no longer required on the receive system, or
- ask the system administrator to allocate more storage space, or
- remove empty blocks from the send file, or
- increase the receive file's primary/secondary allocation.

If option *-e* is specified, restore the receive file.
Repeat the command.

159 Request <Request id>. Remote system: File owner unknown

Meaning:
The command was not executed because the owner of either the send file or the receive file was not defined on the relevant system or because the file owner and the user requesting the creation of a receive file are not the same.

Measure:
Define the file owner, correct transfer admission or file name.
Repeat the command.

160 Request <Request id>. Remote system: Invalid file password

Meaning:
The command was not executed because the password for the send file or the receive file is missing or incorrect.

Measure:
Correct the password in the file description or the command.
Repeat the command.

161 Request <Request id>. Remote system: Retention period of file not yet expired

Meaning:
The command was not executed because the retention period protecting the receive file against overwriting has not yet expired.

Measure:
Correct the transfer direction, retention period or file name.
Repeat the command.

162 Request <Request id>. Remote system: File/directory '<remote file>' is read only

Meaning:
The file or directory is write-protected.

Measure:
Correct the remote file name or remove the write protection of the remote file.
Repeat the command.

163 Request <Request id>. Remote system: File structure not supported

Meaning:
The request cannot be carried out because the file structure is not supported. For example, an attempt was made to get a PLAM library or ISAM file from the BS2000 system.

Measure:
Transfer the file transparently.

164 Request <Request id>. Remote system: Syntax error in resulting file name

Meaning:
A syntax error other than 'Mandatory parameter missing' (703) or 'keyword unknown' has been detected.

Possible reasons:

- Values assigned outside the valid range
- Invalid operand separators
- Invalid value assignment characters
- Partially qualified file names

Measure:
Repeat the command using the correct syntax.

165 Request <Request id>. Remote system: Transparent file transfer not supported

Meaning:
The request could not be carried out because the partner system does not support the transfer of files in a transparent format.

166 Request <Request id>. Remote system: Extension of file not possible for transparent transfer

Meaning:
The command could not be executed because it is not possible to add to a file in a transparent transfer.

167 Request <Request id>. Remote system: Access to '<remote file>' denied

Meaning:
The command was not executed because the remote file only permits certain access modes.

Measure:
Correct the transfer direction, file name or file protection attributes on the remote system.
Repeat the command.

168 Request <Request id>. Remote system: Follow-up processing exceeds length limit

Meaning:
The length of follow-up processing was exceeded; see the command syntax description.

Measure:
Shorten the follow-up processing, or use procedures.
Repeat the command.

169 Request <Request id>. Remote system: Transfer admission invalid

Meaning:
The command was not executed because the specifications in one of the transfer admission operands are incorrect or the request was rejected by FTAC because of insufficient authorization.

Measure:
Define the requisite transfer admission or correct it or check the authorization entered in FTAC. Repeat the command if necessary.

170 Request <Request id>. Remote system: Function not supported

171 Request <Request id>. Remote system: Processing admission invalid

172 Request <Request id>. Remote system: Request queue full

195 Request <Request id>. Remote system: openFT is no longer authorized to execute requests for this user

Measure:
If the remote system is a Windows system: Declare the password of the user to openFT (e.g. using the *ftsetpwd* command).

196 Request <Request id> has been canceled in the remote system

Meaning:
The request was deleted on the remote system before termination.

197 Request <Request id>. Remote system: File/directory '<remote file>' could not be created

Meaning:
The command was not executed because the file owner and user requesting the creation of a receive file are not the same.

Measure:
Match the user ID in the receive system's transfer admission to the ID of the receive file owner. Repeat the command.

198 Request <Request id>. Remote system: Higher-level directory no longer found

199 Request <Request id>. Remote system: I/O error for '<remote file>

Meaning:
An error occurred at input/output. Possible cause:

- BS2000: DMS error, possibly the transfer ID.
- The send or receive files was deleted during transfer.

The FT system continues to run after the message has been issued.

Measure:
Take the appropriate action in accordance with the error code.

200 Request <Request id>. Remote system: File now locked to prevent multiple access

Meaning:
The command was not executed because either the send file or the receive file is already locked by another process to prevent it from being updated simultaneously.
An attempt is made, for example, to access a library opened in z/OS.

Measure:
Repeat the command later or unlock the file.
After a system crash you may need to verify files not closed correctly.
If a lock is caused by an FT request, it will be released automatically when the request is finished.

201 Request <Request id>. Remote system: File/directory '<remote file>' no longer available

Meaning:
The command was not executed because the volume for either the send file or the receive file is not mounted, unknown or reserved, or because the file extends over more than one private disk or an attempt has been made to transfer a file migrated by HSMS.

Measure:
Inform the operator if necessary or carry out an HSMS recall for the file. Repeat the command.

202 Request <Request id>. Remote system: File/directory '<remote file>' no longer found

Meaning:
The command was not executed because the remote file is not or no longer in the catalog or on a volume of the corresponding system (e.g. after a restart).

Measure:
Restore the remote file. Repeat the command.

203 Request <Request id>. Remote system: Home directory no longer found

204 Request <Request id>. Remote system: File/directory '<remote file>' gets no more space

Meaning:
The command was not executed (any further) because

- the permissible storage space on the receive system for the user ID specified in transfer admission has been used up, or
- the send file contains too long a sequence of empty blocks, or
- the primary and/or secondary allocation of the password-protected receive file is too small.

The receive file can no longer be created/extended after the problem occurs.

Measure:
Take the appropriate action depending on the cause of the error:

- delete all files no longer required on the receive system, or
- ask the system administrator to allocate more storage space, or
- remove empty blocks from the send file, or
- increase the receive file's primary/secondary allocation.

If option *-e* is specified, restore the receive file. Repeat the command.

205 Request <Request id>. Remote system: File owner no longer known

Meaning:
The command was not executed because the owner of either the send file or the receive file is not defined on the relevant system, or because the file owner and the user requesting the creation of the receive file are not the same.

Measure:
Define the file owner, correct transfer admission or file name.
Repeat the command.

206 Request <Request id>. Remote system: Pre-/post-processing error

Meaning:
The command executed in local pre-/postprocessing returned an exit code other than 0.

Measure:
Correct the pre-/post-processing command and issue it again.

207 Request <Request id>. Remote system: Exit code <2> during pre-/post-processing

Meaning:
The command executed in local pre-/postprocessing returned the exit code <2>.

Measure:
Correct the pre-/post-processing command in accordance with the exit code and issue it again.

208 Request <Request id>. Remote system: File password no longer valid

Meaning:
The command was not executed because the password for the send file or receive file is missing or incorrect.

Measure:
Correct the password in the file description or the command.
Repeat the command.

209 Request <Request id>. Remote system: File/directory '<remote file>' is now read only

210 Request <Request id>. Remote system: File structure error

Meaning:
The command was not executed due to a file structure error.
File structure errors include:

- The attributes of the send file are incomplete.
- The data of the send file is incompatible with its structure attributes.
- The records of the send file are too long.
- If the *-e* option is specified, the send file and receive file have different structures (e.g. fixed-/variable-length records).
- BS2000: The send or receive file is a member of an old LMS library (not PLAM).
- BS2000: The send file has an odd block factor (e.g. BLKSIZE=(STD,1)), and the receive file is stored on an NK4 pubset.

Measure:
Correct the file or file attributes. If *-e* option is specified, restore the receive file.
Repeat the command.

211 Request <Request id>. Remote system: NDMS error <2>

Meaning:
Repeat the request a little later.

212 Request <Request id>. Recovery failed

Meaning:
The restart could not be carried out. It may not have been possible to complete restart-capable pre-/post-processing before termination of the server process (waiting time: max. minutes).

Measure:
Repeat the command.

213 Request <Request id>. Remote system: Resource bottleneck

Meaning:
The order was rejected because the partner system currently does not have the resources available to accept requests.

Measure:
Repeat the request a little later.

214 Request <Request id>. Remote system: Access to '<remote file>' is no longer permissible

215 Request <Request id>. FTAM error <2>

216 Request <Request id>. Remote system: File structure not supported

217 Request <Request id>. Remote system: Retention period of file not yet expired

218 Request <Request id>. Remote system: Extension of file not possible for transparent transfer

510 Requests carried out; <1> directories were transferred

Meaning:
In the case of a synchronous send request all directories were successfully transferred.

2100 Requests only partially completed; <1> of <2> directories were transferred

In the case of a synchronous send request with wildcards, not all directories were successfully transferred.

Measure:
Transfer unsuccessfully transferred directories again.

4.1.3 Messages for administration commands and measurement data recording

20 openFT already started

Meaning:
openFT can only be started once in each instance.

Measure:
Terminate openFT if necessary.

21 Request must be canceled without FORCE option first

Meaning:
Before the FORCE option is used, the command must be called without the FORCE option.

Measure:
Issue the command without the FORCE option first.

29 Maximum number of key pairs exceeded

Measure:
Before a new key pair set can be created, an older key pair set must be deleted.

30 Warning: last key pair deleted

Meaning:
The last key pair set has been deleted. Without a key pair set, encrypted transfer, authentication and data integrity checking are not possible.

Measure:
Create a new key pair set.

31 No key pair available

Meaning:
All transfers are carried out without encryption.

Measure:
Create a new key pair set, if necessary.

32 Last key pair must not be deleted

33 The public key files could not be updated

Meaning:
The contents of the *syspkf* file could not be fully updated.
Possible reasons:

- The *syspkf* file is locked.
- There is not enough disk space to allow the file to be created.

Measure:
Take the appropriate action depending on the cause of the error:

- Unlock the file.
- Allocate disk space or have your system administrator do it.

Update the key with *ftupdk*.

34 Command only permissible for FT, FTAC or ADM administrator

Meaning:
Only the FT, FTAC or ADM administrator is permitted to use the command.

Measure:
Have the command executed by the FT, FTAC or ADM administrator.

35 Command only permissible for FT administrator

Meaning:
Only the FT administrator is permitted to use the command.

Measure:
Have the command executed by the FT administrator.

36 User not authorized for other user IDs

Meaning:
The user is not authorized to use a different user ID in the command.

Measure:
Specify your own ID, or have the command executed by the FT or FTAC administrator.

37 Key reference unknown

Meaning:
The specified key reference is unknown.

Measure:
Repeat the command with an existing key reference.

38 Request <Request id> is in the termination phase and can no longer be canceled

39 openFT not active

Meaning:
openFT is not started.

Measure:
Start openFT, if necessary.

40 Config user ID unknown or not enough space

Meaning:
The Config user ID of the current instance is unknown or the disk space allocated is insufficient to allow creation of the request file, the file for storing trace data, or the key files.

Measure:
Either create the Config user ID or increase its disk space allocation or have your system administrator do it.

41 Specified file is not a valid trace file

42 openFT could not be started

43 Partner with same attribute <attribute> already exists in partner list

Meaning:
There is already a partner entry with the same attribute <attribute> in the partner list.

Measure:
The attribute <attribute> in partner entries must be unique. Correct the command accordingly and try again.

44 Maximum number of partners exceeded

Meaning:
The partner list already contains the maximum permissible number of partner entries.

Measure:
Delete partners that are no longer required.

45 No partner found in partner list

Meaning:
A partner for the specified selection could not be found in the partner list.

Measure:
Check if the specified partner name or address was correct.
If necessary, repeat the command using the correct name or address.

46 Modification of partner protocol type not possible

Meaning:
The protocol type of the partner entry cannot be changed subsequently.

Measure:
Delete the partner from the partner list, if necessary, and enter it again with a new protocol type.

47 Request <Request id> not found

Meaning:
The request with the transfer ID <Request id> could not be found.

Measure:
Specify the existing transfer ID and repeat the command.

48 Active requests could not yet be deleted

49 CCS name '<1>' unknown

57 Inbound requests cannot be modified

58 The ADM trap server configuration is invalid

59 monitoring is not active

Meaning:
The command is only supported if monitoring is activated.

Measure:
Ask the FT administrator to activate monitoring in the operating parameters and repeat the command.

60 File could not be created <2>

Meaning:
The command was not executed because the local file could not be created.

Measure:
Check the directory and access rights. Repeat the command.

61 Higher-level directory not found

Meaning:
The local file could not be created because the specified path does not exist.

Measure:
Create or correct the path for the file and repeat the command.

62 File already exists

Meaning:
The command was not executed because the specified file already exists.

Measure:
Either delete the existing file or choose a different name and repeat the command.

63 Resulting file name too long

Meaning:
The filename has the wrong syntax or is too long. Specifying a partially qualified filename may be the cause of the error.

Measure:
Repeat the command using the correct syntax.

64 File locked to prevent multiple access

Meaning:
The command was not executed because the file is already locked by another process.

Measure:
Repeat the command later.

65 File not found

Meaning:
The command was not executed because the specified file was not found.

Measure:
Correct the file name and repeat the command.

66 Not enough space for file

Meaning:
The command was not executed because the permitted storage space on the local volume is exhausted.

Measure:
Take appropriate measures depending on the cause of the error.

- Delete any files that are no longer required or
- Request the system administrator to assign more storage space.

67 Syntax error in resulting file name

Meaning:
The file cannot be accessed because the absolute file name has become too long, for instance.

Measure:
Shorten the path or the file name. Repeat the command.

68 Access to file denied<2>

Meaning:
The command was not executed because the file only permits certain access modes (e.g. read-only).

Measure:
Correct the file name or the file protection attributes.
Repeat the command.

69 Error accessing file<2>

Meaning:
<2>: DMS error

Measure:
Take appropriate measures depending on the error code.

70 Configuration data invalid

Meaning:
The configuration data is syntactically or semantically incorrect and can therefore not be imported.

Measure:
Correct the error on the basis of the additional diagnostic output and then repeat import of the configuration data.

71 Import of configuration data not possible while remote administration server is started

Meaning:
The changes to the configuration data are so extensive that they can only be imported when the remote administration server has been terminated.

Measure:
Terminate openFT using the *ftstop* command and then attempt to import the configuration data again.

73 Command aborted

Meaning:
The user has cancelled the command.

74 Command only permissible for ADM administrator on a remote administration server

Meaning:
The command is only permitted for the ADM administrator.

Measure:
Have the ADM administrator execute the command if necessary.

77 Not possible to change transport system access. Cause: <1>

Meaning:
The operating mode with and without CMX could not be changed using the *ftmodo* command. Possible causes could be:

openFT is started

CMX not installed

> ¡ On Windows systems, the cause "CMX not installed" can also occur if PCMX is installed but the installed version of PCMX is too old. Please refer to the release notice for openFT (Windows) for information on the minimum required version of PCMX-32.

78 Interval too short since last log file change

Meaning:
Log file cannot be changed at present because the timestamp-dependent name part does not differ from the name part of the current log file.

Measure:
Wait for a short time and repeat the command (if necessary).

4.1.4 Messages for openFT-Script commands

In the case of the messages listed below, the value for *fthelp* must be increased by 1000, e.g. 1052 instead of 52.

15 openFT is not authorized to process requests for this user (e.g. password not set on access to home directory)

50 ftscript process could not be started

51 Error displaying an ftscript user

52 ftscript user number limit exceeded

53 ftscript chapter not found

54 ftscript ID not found

55 ftscript file not found

56 ftscript request is still running

69 File access error (*Prelock.lck/UserLock.lck* in *FtscriptWorkdir*)

79 openFT-Script interpreter or other *ftmodsuo* is running. Command aborted

80 Current openFT-Script requests are present. Command aborted

81 Old openFT-Script request not accessible

88 Subdirectories cannot be created in the openFT-Script working directory.

Meaning: The directory <*wd*>*/.openFT/*<*instance name*>*/script* (Unix systems) or <*wd*>*.openFT*\<*instance name*>*script* (Windows systems) could not be created, for example due to the absence of write access permission or because a physical error occurred.

90 Working directory does not exist. Command aborted

91 Warning: The previous working directory could not be checked

4.1.5 Messages for remote administration

With the following messages, the value for *fthelp* must be increased by 2000, e.g. 2052 instead of 52.

52 Administration request rejected by remote administration server

Meaning:
The administration request was rejected by the remote administration server because it clashes with the settings in the configuration file of the remote administration server.

The ADM administrator can determine the precise reason for rejection from the associated ADM log record on the remote administration server.

Possible reason codes:

7001 The administrator ID is invalid. It was not possible to determine a valid administrator ID from the user ID or the profile name in the configuration data of the remote administration server.

7002 The routing information is invalid. The specified openFT instance specified in the routing information could not be found in the configuration data of the remote administration server.

7003 The specified remote administration command is invalid. The remote administration server rejects the specified command because it is not a supported remote administration command.

7101 Infringement against the access rights list. On checking the access rights, the system identified that the administrator ID has not been assigned the necessary rights in the configuration data of the remote administration server to be able to execute the valid remote administration command on the specified openFT instance.

7201 Infringement against the maximum command length. In particular in the case of BS2000 commands, the remote administration server replaces the shortest command names, which are guaranteed by openFT, by the full command names. If this replacement of the command name causes the entire remote administration command to become longer than the maximum command length of 8192 characters, the command is rejected.

Measure:
Have the ADM administrator carry out the necessary adjustments to the configuration data or check the command. Repeat the changed command if necessary.

54 Invalid command

Meaning:
The specified command is not a command that is permitted to be executed on the specified system using the remote administration facility.

Measure:
Specifying an admissible command or add the missing routing information. Repeat the command.

57 openFT is not authorized to execute administration requests

Meaning:
openFT is not (no longer) authorized to process administration requests. This is, for example, the case if a remote administration server has been demoted to a normal server (*ftmodo -admcs=n*) or if commands that are only allowed to be executed on a remote administration server are processed by an openFT instance that has not been configured as a remote administration server.

4.2 FTAC messages

001 FTAC version $VERSION active

003 $NUMBER logging records deleted

050 Lower ADM-level remains in effect

051 Transfer admission exists as user ID

052 Information incomplete

053 No FT profile found

054 No information available

055 Partner restriction does no longer exist

056 Transfer admission locked

057 Attributes of transfer admission are ignored

070 Shortage of resources

071 openFT not active

100 FT profile already exists

101 Transfer admission already exists

102 File already exists

103 Invalid file content or access to file denied

104 Access to directory denied

105 Access to file denied

106 Access to temporary file denied

107 No space available

108 The version of export file is not compatible with current version

109 File is no FTAC export file

Meaning:
A *ftshwe* or *ftimpe* command was issued for a file which is not a FTAC backup file.

110 File name too long

111 Syntax error in file name

112 Expiration date not valid

150 User not authorized for FTAC commands

151 User not authorized for this modification

152 User not authorized for other user IDs

153 User not authorized for other owner IDs

154 No authorization for deletion of log records

155 User not authorized for diagnose

156 Command allowed for FTAC administrator only

157 No authorization for this set of parameters

170 Given partner unknown

171 Given FT profile name unknown

172 Invalid user admission

173 Invalid processing admission

174 Modification invalid for not unique selection criteria

175 Modification invalid for standard authorization record

176 Given user ID unknown

177 File unknown

178 Multiple partner specified

179 Violation of maximal number of partners

180 Multiple user ID specified

181 Multiple FT profile name specified

182 Total maximum partner length exceeded

183 Partner not supported

184 Transfer admission of standard profile must be @n

185 Combination of these transfer functions not allowed

200 Follow-up processing too long

201 User ID too long

202 Profile name too long

203 Transfer admission too long

204 Partner too long

205 Fully qualified file name too long

Meaning:
By extension with absolute path name, the maximum value of 512 characters (Unix systems) or 256 characters (Windows systems) was exceeded.

206 Partially qualified file name too long

207 Processing command too long

208 Invalid date specified

209 Invalid time specified

210 Transfer admission too short

211 Parameters $PAR1 and $PAR2 must not be specified together

212 License check error $NUMBER for FTAC

213 Mandatory parameter profile name is missing

214 Mandatory parameter file name is missing

215 Syntax error in parameter $PARAMETER

216 Password too long

217 Text too long

218 Too many partners

219 Too many users

220 Too many profiles

250 Initialization of FTAC failed

251 FTAC not available

252 FTAC version incompatible

253 FTAC command not found in syntaxfile

254 System error. Errorcode $NUMBER

255 System error

If message 254 or 255 is displayed, please follow the instructions given in the chapter “What if ...”.

4.3 FTAM diagnostic codes as per ISO 8571-3

The following excerpt from ISO FTAM standard ISO 8571-3 describes the possible diagnostic codes that can appear in the DIAGCODE column or in the messages 2093 or 2215 as $NUMBER when displaying the request queue for requests to FTAM partners:

Identifier	Reason
0	No reason
1	Responder error (unspecific)
2	System shutdown
3	FTAM management problem (unspecific)
4	FTAM management, bad account
5	FTAM management, security not passed
6	Delay may be encountered
7	Initiator error (unspecific)
8	Subsequent error
9	Temporal insufficiency of resources
10	Access request violates VFS security
11	Access request violates local security
1000	Conflicting parameter values
1001	Unsupported parameter values
1002	Mandatory parameter not set
1003	Unsupported parameter
1004	Duplicated parameter
1005	Illegal parameter type
1006	Unsupported parameter types
1007	FTAM protocol error (unspecific)
1008	FTAM protocol error, procedure error
1009	FTAM protocol error, functional unit error
1010	FTAM protocol error, corruption error

1011	Lower layer failure
1012	Lower layer addressing error
1013	Timeout
1014	System shutdown
1015	Illegal grouping sequence
1016	Grouping threshold violation
1017	Specific PDU request inconsistent with the current requested access
2000	Association with user not allowed
2001	(not assigned)
2002	Unsupported service class
2003	Unsupported functional unit
2004	Attribute group error (unspecific)
2005	Attribute group not supported
2006	Attribute group not allowed
2007	Bad account
2008	Association management (unspecific)
2009	Association management - bad address
2010	Association management - bad account
2011	Checkpoint window error - too large
2012	Checkpoint window error - too small
2013	Checkpoint window error - unsupported
2014	Communications QoS not supported
2015	Initiator identity unacceptable
2016	Context management refused
2017	Rollback not available
2018	Contents type list cut by responder
2019	Contents type list by Presentation service
2020	Invalid filestore password

2021	Incompatible service classes
3000	Filename not found
3001	Selection attributes not matched
3002	Initial attributes not possible
3003	Bad attribute name
3004	Non-existent file
3005	File already exists
3006	File cannot be created
3007	File cannot be deleted
3008	Concurrence control not available
3009	Concurrence control not supported
3010	Concurrence control not possible
3011	More restrictive lock
3012	File busy
3013	File not available
3014	Access control not available
3015	Access control not supported
3016	Access control inconsistent
3017	Filename truncated
3018	Initial attributes altered
3019	Bad account
3020	Override selected existing file
3021	Override deleted and recreated file with old attributes
3022	Create override deleted and recreate file with new attributes
3023	Create override - not possible
3024	Ambiguous file specification
3025	Invalid create password
3026	Invalid delete password on override

3027	Bad attribute value
3028	Requested access violates permitted actions
3029	Functional unit not available for requested access
3030	File created but not selected
4000	Attribute non-existent
4001	Attribute cannot be read
4002	Attribute cannot be changed
4003	Attribute not supported
4004	Bad attribute name
4005	Bad attribute value
4006	Attribute partially supported
4007	Additional set attribute value not distinct
5000	Bad FADU (unspecific)
5001	Bad FADU - size error
5002	Bad FADU - type error
5003	Bad FADU - poorly specified
5004	Bad FADU - bad location
5004	FADU does not exist
5006	FADU not available (unspecific)
5007	FADU not available for reading
5008	FADU not available for writing
5009	FADU not available for location
5010	FADU not available for erasure
5011	FADU cannot be inserted
5012	FADU cannot be replaced
5013	FADU cannot be located
5014	Bad data element type
5015	Operation not available

5016	Operation not supported
5017	Operation inconsistent
5018	Concurrence control not available
5019	Concurrence control not supported
5020	Concurrence control inconsistent
5021	Processing mode not available
5022	Processing mode not supported
5023	Processing mode inconsistent
5024	Access context not available
5025	Access context not supported
5026	Bad write (unspecific)
5027	Bad read (unspecific)
5028	Local failure (unspecific)
5029	Local failure - filespace exhausted
5030	Local failure - data corrupted
5031	Local failure - device failure
5032	Future file size exceeded
5034	Future file size increased
5035	Functional unit invalid in processing mode
5036	Contents type inconsistent
5037	Contents type simplified
5038	Duplicate FADU name
5039	Damage to select/open regime
5040	FADU locking not available on file
5041	FADU locked by another user
6000	Bad checkpoint (unspecific)
6001	Activity not unique
6002	Checkpoint outside window

6003	Activity no longer exists
6004	Activity not recognized
6005	No docket
6006	Corrupt docket
6007	File waiting restart
6008	Bad recovery point
6009	Non-existent recovery point
6010	Recovery mode not available
6011	Recovery mode inconsistent
6012	Recovery mode reduced
6013	Access control not available
6014	Access control not supported
6015	Access control inconsistent
6016	Contents type inconsistent
6017	Contents type simplified

5 What if ...

Locked transfer admissions - possible causes and remedies

If FTAC rejects a file transfer request on account of an invalid transfer admission, the cause may be one of several:

- No transfer admission was defined when the FT profile was created or modified.
- A user wished to create an FT profile with a transfer admission which was already assigned to a different FT profile on the computer. If the relevant FT profile is marked as private, the transfer admission becomes invalid. At the same time, the values for date, scope (public/private) and validity (*-d*, *-u* and *-v*) are set to the default values.
- The FTAC administrator modifies an FT profile for a user without knowledge of the complete login authorization. In this case, the transfer admission remains valid, but is locked.
- The FT profile was imported by an FTAC administrator who is not the FT administrator. It is therefore locked automatically.
- The FT profile was locked explicitly.
- The period during which the transfer admission may be used has expired.

The detailed output of the *ftshwp* command displays the cause of an invalid transfer admission using the additional output parameter *TRANS-ADM*. The possible values for this output parameter, the meanings and counteractions are shown in the table "TRANS-ADM" in the *ftshwp* command.

6 Structure of CSV outputs

The output format for all commands corresponds to the following rules:

- Each record is output in a separate line. A record contains all the information to be displayed on an object.
- The first line is a header and contains the field names of the respective columns. **Only the field names are guaranteed, not the order of fields in the record.** In other words, the order of columns is determined by the order of the field names in the header line.
- Two tables, with their own respective headers, are output sequentially for the command *ftshwe*. If one of the tables is empty, the corresponding header is also dropped.
- Individual fields within an output line are delimited by a semicolon “;”.

The following data types are differentiated in the output:

- Number

  Integer
- String

  Since ";" is a metacharacter in the CSV output, any text that contains ";" is enclosed in double quotes ("). Double quotes within a text field are doubled in order to differentiate them from text delimiters. When imported into a program, the doubled quotes are automatically removed and the text delimiters removed. Keywords are output in uppercase with a leading asterisk (*) and are not enclosed in double quotes.
- Date

  The date and time are output in the form yyyy-mm-dd hh:mm:ss. In some cases, only the short form yyyy-mm-dd is ouput, i.e. the date alone.
- Time

  The time is output in the form yyyy-mm-dd hh:mm:ss or only hh:mm:ss.

6.1 ftshw/ftshwf

The following table indicates the CSV output format for file attributes.

The **Std** column is not relevant on Unix and Windows systems.

The **Parameter** column indicates the name of the output parameter in the case of detailed output, see Description of file attribute display.

Column	Type	Values and Meaning	Parameter	Std
FileName	String	File name or directory name enclosed in double quotes / *NSPEC	FILENAME	
StorageAccount	String	Account number enclosed in double quotes / *NSPEC	STORAGE-ACCOUNT	
CreIdentity	String	Identity of the last user of the file (creator) enclosed in double quotes / *NSPEC	CRE name	
CreTime	Date	Time at which the file was created / *NSPEC	CRE DATE	
ModIdentity	String	Identity of the last user of the file (modification of file content) enclosed in double quotes / *NSPEC	MOD name	
ModTime	Date	Time at which the file was last modified / *NSPEC	MOD DATE	
ReaIdentity	String	Identity of the last user of the file (file read access) enclosed in double quotes / *NSPEC	REA name	
ReaTime	Date	Time at which the file was last read / *NSPEC	REA DATE	
AtmIdentity	String	Identity of the last user of the file (modification of file attributes) enclosed in double quotes / *NSPEC	ATM name	
AtmTime	Date	Time at which the file attributes were last modified / *NSPEC	ATM DATE	
FileType	String	*BIN / *DIR / *TEXT / *NONE / *NSPEC File type	file type	
CharSet	String	*VISIBLE / *IA5 / *GRAPHIC / *GENERAL / *NONE / *NSPEC Character set for the text file if FileType=*TEXT, in the case of another FileType, this is *NONE or *NSPEC	CHARACTERSET	
RecFormat	String	*VAR / *FIX / *NSIG / *NSPEC Record format	RECORD-FORMAT	
RecSize	Number	1... 65535 / *NSPEC Maximum length of the records	RECORD-SIZE	

FileAvail	String	*IMMEDIATE / *DEFERRED / *NSPEC File avaliability	FILE-AVAILABILITY	
AccessRights	String	nnnnnnnnnnn / *NSPEC Access rights, n = p, x, e, a, c, d, t, v, r, -	ACCESS-RIGHTS	
FileSize	Number	Current file size in bytes / *NSPEC	FILESIZE	
MaxFileSize	Number	Maximum file size in bytes / *NSPEC	MAX-FILESIZE	
LegalQualif	String	Legal qualification enclosed in double quotes / *NSPEC	LEGAL-QUALIFICATION	
CcsName	String	Name of the character set / *NSPEC	CCS-NAME	

Example

```
$ ftshw bs2partn!aaa.e42 transbs2 -csv
FileName;StorageAccount;CreIdentity;CreTime;ModIdentity;
ModTime;ReaIdentity;ReaTime;AtmIdentity;AtmTime;FileType;
CharSet;RecFormat;RecSize;FileAvail;AccessRights;FileSize;
MaxFileSize;LegalQualif;CcsName
"aaa.e42";*NSPEC;"maier";*NSPEC;*NSPEC;2017-01-17 13:01:34;
*NSPEC;*NSPEC;*NSPEC;*NSPEC;*NSPEC;*NSPEC;*NSIG;*NSPEC;
*NSPEC;r-pxeacd---;174;*NSPEC;*NSPEC;*NSPEC
```

Note: In case of not mapped file names (*sif=l*) for *FileName* the string
`|*IMPROPER FILE NAMES (x): nnn` and for *MaxFileSize* the value nnn is output.
x= D,I or R, *nnn* = number of non-mappable file names of category *x*. Other attributes are not supplied (*NSPEC).

6.2 ftshwa

The following table indicates the CSV output format of an admission set.

The **Parameter** column contains the name of the output parameter during normal output, see Output format of ftshwa.

Column	Type	Values and Meaning	Parameter
UserId	String	User ID, enclosed in double quotes / *STD *STD means standard admission set	USER-ID
UserMaxObs	Number	0 ... 100 Maximum user level for OUTBOUND-SEND	MAX. USER LEVELS OBS
UserMaxObsStd	String	*YES / *NO *YES means same value as standard admission set [1]	
UserMaxObr	Number	0 ... 100 Maximum user level for OUTBOUND-RECEIVE	MAX. USER LEVELS OBR
UserMaxObrStd	String	*YES / *NO *YES means same value as standard admission set[1]	
UserMaxIbs	Number	0 ... 100 Maximum user level for INBOUND-SEND	MAX. USER LEVELS IBS
UserMaxIbsStd	String	*YES / *NO *YES means same value as standard admission set [1]	
UserMaxIbr	Number	0 ... 100 Maximum user level for INBOUND-RECEIVE	MAX. USER LEVELS IBR
UserMaxIbrStd	String	*YES / *NO *YES means same value as standard admission set [1]	
UserMaxIbp	Number	0 ... 100 Maximum user level for INBOUND-PROCESSING	MAX. USER LEVELS IBP
UserMaxIbpStd	String	*YES / *NO *YES means same value as standard admission set [1]	
UserMaxIbf	Number	0 ... 100 Maximum user level for INBOUND-FILE-MANAGEMENT	MAX. USER LEVELS IBF
UserMaxIbfStd	String	*YES / *NO *YES means same value as standard admission set [1]	

AdmMaxObs	Number	0 ... 100 Maximum level of FTAC administrator for OUTBOUND-SEND	MAX. ADM LEVELS OBS
AdmMaxObsStd	String	*YES / *NO *YES means same value as standard admission set [1]	
AdmMaxObr	Number	0 ... 100 Maximum level of FTAC administrator for OUTBOUND-RECEIVE	MAX. ADM LEVELS OBR
AdmMaxObrStd	String	*YES / *NO *YES means same value as standard admission set [1]	
AdmMaxIbs	Number	0 ... 100 Maximum level of FTAC administrator for INBOUND-SEND	MAX. ADM LEVELS IBS
AdmMaxIbsStd	String	*YES / *NO *YES means same value as standard admission set [1]	
AdmMaxIbr	Number	0 ... 100 Maximum level of FTAC administrator for INBOUND-RECEIVE	MAX. ADM LEVELS IBR
AdmMaxIbrStd	String	*YES / *NO *YES means same value as standard admission set [1]	
AdmMaxIbp	Number	0 ... 100 Maximum level of FTAC administrator for INBOUND-PROCESSING	MAX. ADM LEVELS IBP
AdmMaxIbpStd	String	*YES / *NO *YES means same value as standard admission set [1]	
AdmMaxIbf	Number	0 ... 100 Maximum level of FTAC administrator for INBOUND-FILE-MANAGEMENT	MAX. ADM LEVELS IBF
AdmMaxIbfStd	String	*YES / *NO *YES means same value as standard admission set [1]	
Priv	String	*YES / *NO *YES means admission set of FTAC administrator	ATTR
Password	String	*NO	ATTR
AdmPriv	String	*YES / *NO *YES means admission set of the ADM administrator	ATTR

[1]Relevant only if UserId is not *STD, *NO is always output in the case of the standard admission set. In the normal output, *YES corresponds to an asterisk (*) after the value.

6.3 ftshwact

The following table indicates the CSV output format of the activities of the individual openFT-Script request. For details on *Id*, *State*, *Activity* and *ActivityObject*, please refer to the *ftshwact* command, section „Output in table format “.

Column	Type	Values and Meaning
Id	String	Unique identification of the activity within the request.
State	String	W / R / T / F / K / D and, in addition, I / C / X / F for the ftscript activity. Status of the statement.
Activity	String	Activity name.
ActivityObject	String	Activity, see table format, enclosed in double quotes, otherwise: - the path name is output without partner specifications - only the *faultcodes* are output for the *faulthandler* activity.
Partner	String	In the case of path-related activities, the partner or partner specification that would be present in front of the path name in table format, enclosed in double quotes. Otherwise empty.
AddInfo	String	For *sendFile* and *rcvFile*: TID, enclosed in double quotes if the activity has already started. Otherwise empty. For *faulthdl*, the triggering *activity-Id* enclosed in double quotes. Otherwise empty.
NrElements	String	In the case of a started *foreach*: number of loop passes. In the case of a started *parallel* or *sequence*: number of sub-activities.
StartTime	String	Start time in the format yyyy-mm-dd hh:mm:ss
Error	String	In the case of requests with the status F, case of error in clear text enclosed by double quotes. Otherwise empty.

6.4 ftshwatp

The following table indicates the CSV output format of an ADM trap.

The **Parameter** column indicates the name of the output parameter in the long output from *ftshwatp*, see Long output format of an ADM trap.

Column	Type	Values and Meaning	Parameter
TrapId	Number	Number of the ADM trap, up to 18 digits	TRAP-ID
Source	String	Name of the partner that triggered the trap enclosed in double quotes	SOURCE
TrapTime	Date	Date and time at which the trap occurred	DATE, TIME
TrapType	String	Type of the trap	TYPE
PartnerState	String	Partner state of the partner that triggered the trap	PTN-STATE
TransId	Number	Transfer ID [1]	TRANS-ID
RqInitiator	String	User ID or location [1] enclosed in double quotes / *REM	INITIATOR
PartnerName	String	Partner name [1] enclosed in double quotes	PARTNER
FileName	String	File name [1] enclosed in double quotes	FILENAME
RqError	String	Reason code [1] enclosed in double quotes	RC
RqErrorMsg	String	Message text [1] enclosed in double quotes	ERROR-MSG

[1] of the transfer that triggered the trap

6.5 ftshwc

The following table indicates the CSV output format of instances that can be remote administrated.

The **Parameter** column indicates the name of the output parameter in the normal output from *ftshwc*, see Output format of ftshwc.

Column	Type	Values and Meaning	Parameter
Name	String	Name enclosed in double quotes	NAME
Description	String	Description enclosed in double quotes	DESCRIPTION
Type	String	*GROUP / *INSTANCE Type (group or openFT instance)	TYPE
AccessFtAdm	String	*YES / *NO / *NONE Reading and modifying FT accesses are allowed (corresponds to FT administrator rights) / not allowed / not relevant (for Type = *GROUP)	ACCESS
AccessFtacAdm	String	*YES / *NO / *NONE Reading and modifying FTAC accesses are allowed (corresponds to FTAC administrator rights) / not allowed / not relevant (for Type = *GROUP)	ACCESS
AccessFtOp	String	*YES / *NO / *NONE Reading FT accesses are allowed / not allowed / not relevant (for Type = *GROUP)	ACCESS
Mode	String	*FTADM / *LEGACY / *NONE The instance is administered using the FTADM protocol / via ftexec / not relevant (if Type = *GROUP)	MODE
CmdMode	String	*CHAR / *TRANSPARENT / empty Encoding mode with *ftadm* commands: character mode / transparent mode / not relevant	MODE

Example

```
ftshwc -csv
Name;Description;Type;AccessFtAdm;AccessFtacAdm;AccessFtOp;Mode;CmdMode
"Hamburg";"Northern Computer Center in Hamburg Wandsbek";*GROUP;*NONE;*NONE;
*NONE;;
"Hamburg/HH1";"QA Computer Center";*GROUP;*NONE;*NONE;*NONE;;
"Hamburg/HH1/HHWSRV01";"Solaris 10";*INSTANCE;*YES;*YES;*YES;*FTADM;*CHAR
"Hamburg/HH1/HHWSRV02";"HP-11";*INSTANCE;*YES;*YES;*YES;*FTADM;
"Hamburg/HH1/HHWSRV11";"Solaris 9";*INSTANCE;*YES;*NO;*YES;*LEGACY;*TRANSPARENT
"Hamburg/HH2";"HR department";*GROUP;*NONE;*NONE;*NONE;;
"Hamburg/HH2/HHWSRV99";"Mainframe system
(BS2000/OSD)";*INSTANCE;*NO;*NO;*YES;*FTADM;
```

6.6 ftshwe

The command *ftshwe* sequentially displays the objects contained in an FTAC export file in a format that corresponds to the output of the *ftshwa* and *ftshwp* commands.

6.7 ftshwk

The table below indicates the CSV format for the output of the properties of the RSA keys.

The **Parameter** column contains the name of the output parameter during normal output, see *ftshwk*.

Column	Type	Values and Meaning	Parameter
Reference	Number	Key reference	KEY-REF
Identification	String	Identification of the partner enclosed in double quotes / *OWN *OWN means the private key for the user's own instance	IDENTIFICATION
PartnerName	String	Name of partner / *OWN *OWN means the private key for the user's own instance	PARTNER
CreDate	Date	Date on which the key was generated	CRE-DATE
ExpDate	String	Date on which the key expires / *NONE	EXP-DATE
Expired	String	*YES / *NO Key has expired / not expired	EXP-DATE (EXPIRED)
KeyLen	Number	768 / 1024 / 2048 / 3072 / 4096 Key length in bits	KEY-LEN
AuthLev	Number	1 / 2 Authentication level	AUTHL

6.8 ftshwl

The following table indicates the CSV output format of a log record if the option *-llf* has not been specified. If the option *-llf* is specified then the output has a different format, see "ftshwl".

A format template in Microsoft Excel format is present in the following file as an example of a possible evaluation procedure:

/opt/openFT/samples/ftaccnt.xlt (Unix systems)

openFT-installation-directory\samples\msexcel\ftaccnt.xlt (Windows systems)

The **Std** column is not relevant on Unix and Windows systems.

The **Parameter** column contains the name of the output parameter during long output, see Description of log record output.

Column	Type	Values and Meaning	Parameter	Std
LogId	Number	Number of the log record (up to twelve digits)	LOGGING-ID	
ReasonCode	String	Reason code enclosed in double quotes to prevent interpretation as a number. FTAC Reason Codes are output as hexadecimal strings	RC	
LogTime	Date	Time at which the log record was written	TIME	
InitUserId	String	Initiator of the request enclosed in double quotes / *REM	INITIATOR	
InitTsn	String	*NONE	---	
PartnerName	String	Partner name enclosed in double quotes (name or address)	PARTNER	
TransDir	String	*TO / *FROM / *NSPEC Transfer direction	TRANS	
RecType	String	*FT / *FTAC / *ADM Type of log record	REC-TYPE	
Func	String	*TRANS-FILE / *TRANS-DIR / *READ-FILE-ATTR / *DEL-FILE / *CRE-FILE / *MOD-FILE-ATTR / *READ-DIR / *MOVE-FILE / *CRE-FILE-DIR / *DEL-FILE-DIR / *LOGIN / *MOD-FILE-DIR / *REM-ADMIN / *REM-ADMIN-ROUT FT function	FUNCTION	
UserAdmisId	String	User ID to which the requests in the local system relate, enclosed in double quotes	USER-ADM	
FileName	String	Local file name enclosed in double quotes	FILENAME	

Priv	String	*YES / *NO / *NONE Profile is privileged / not privileged / not relevant because no profile was used or no FTAC log record is present	PRIV	
ProfName	String	Name of the admission profile enclosed in double quotes / *NONE	PROFILE	
ResultProcess	String	*STARTED / *NOT-STARTED / *NONE Status of follow-up processing	PCMD	
StartTime	Date	Start time of transfer	STARTTIME	
TransId	Number	Number of transfer request	TRANS-ID	
Write	String	*REPL / *EXT / *NEW / *NONE Write rules	WRITE	
StoreTime	Date	Acceptance time of request • If initiated in the local system: time the request was issued • If initiated in the remote system: time of entry in the request queue	REQUESTED STORETIME	
ByteNum	Number	Number of bytes transferred	TRANSFER	
DiagInf	String	Diagnostic information / *NONE	---	
ErrInfo	String	Additional information on the error message, enclosed in double quotes / *NONE	ERRINFO	
Protection	String	*SAME / *STD Protection attributes are transferred / not transferred	PROTECTION ---	
ChangeDate	String	*SAME / *STD Take over modification date of send file for receive file / do not take over modification date	CHG-DATE	
SecEncr	String	*YES / *NO Encryption of request description activated / deactivated	SEC-OPTS	
SecDichk	String	*YES / *NO Data integrity check of request description activated / deactivated	SEC-OPTS	
SecDencr	String	*YES / *NO Encryption of transferred file content activated / deactivated	SEC-OPTS	
SecDdichk	String	*YES / *NO Data integrity check of transferred file content activated / deactivated	SEC-OPTS	

SecLauth	String	*YES / *NO Authentication of the local system in the remote system activated / deactivated	SEC-OPTS	
SecRauth	String	*YES / *NO Authentication of the remote system in the local system activated / deactivated	SEC-OPTS	
RsaKeyLen	Number	768 / 1024 / 2048 / 3072 / 4096 / empty Length of the RSA key used for the encryption in bit or empty if SecEncr does not have the value *YES	SEC-OPTS	
SymEncrAlg	String	*DES / *AES-128 / *AES-256 / empty The encryption algorithm used or empty if SecEncr does not have the value *YES	SEC-OPTS	
CcsName	String	Name of the character set enclosed in double quotes / empty	CCS-NAME	
AdminId	String	Administrator ID on the remote administration server, enclosed in double quotes / empty	ADMIN-ID	
Routing	String	Routing information enclosed in double quotes / empty	ROUTING	
AdmCmd	String	Administration command enclosed in double quotes / empty	ADM-CMD	
As3Type	String	empty (internal function)	---	
As3MsgTid	String	empty (internal function)	---	
As3RcpStat	String	empty (internal function)	---	
AuthLev	Number	1 / 2 / empty Authentication level	SEC-OPTS	
GlobReqId	Number	Global request identification (requests issued remotely) / empty (requests issued locally)	GLOB-ID	
FileNameCMode	String	*TRANSPARENT / *CHAR Encoding mode for file names	FNC-CODE	
FileNameCcs	String	Name of the character set enclosed in double quotes (FileNameCMode=*CHAR) / empty	FNCCS	
PtnrAddr	String	Address of the partner system in the case of inbound FT requests	PTNR-ADDR	
TrnsFiles	String	In case of directory transfer: Number of completed subrequests and total number of subrequests	TRANSFILE	
RemoteFileName	String	Remote file name enclosed in double quotes (openFT version >= V12.1C10)	REMOTE-FN	

CSV output on ftshwl -llf

If the option *-llf* is specified then only the following columns are output:

Column	Type	Values and Meaning	Parameter
TimeStamp	Date	Creation time of the log file	---
LoggingFileName	String	Fully qualified name of the log file	(file name)

6.9 ftshwlic

The table below indicates the CSV output format for a license key. The **Parameter** column contains the name of the output parameter during normal output from *ftshwlic*, see Output format of ftshwlic.

Column	Type	Values and Meaning	Parameter
Type	String	License type enclosed in double quotes. The standard types are SERVER, FTAM and FTP	type
SerialNr	Number	Serial number	serial number
PerfClass	String	Performance class / unlimited, each enclosed in double quotes	performance class

Example

```
ftshwlic -csv
Type;SerialNr;PerfClass
"SERVER";002000;"1-4 CPUs"
"FTAM";000030;"1-4 CPUs"
"FTP";0000050;"1-4 CPUs"
```

6.10 ftshwm

The following table shows the CSV output format for the monitoring values for openFT operation if all the monitoring values are output (*ftshwm -csv @a*).

If the *-raw* option is specified, the duration values are not output (*Duxxx*, see footnote).

The default values are marked with "x" in the **Std** column. These are output if *ftshwm -csv* is specified without *@a* and without names being specified explicitly.

For a detailed description of the monitoring values, refer to the Description of the monitoring values.

The individual monitoring values (ThNetbTtl ... StTrcr) have the same names in all the output formats (normal output, long output and CSV output).

Column	Type	Values prepared	Values not prepared	Meaning	Std
CurrTime	Date	Time	Time	Current time	x
MonOn	Date	Time	Time	Start time of measurement date recording or last change of configuration (a modification of PartnerSel/ReqSel has the same effect as a new start)	x
PartnerSel	String	*ALL / *NONE / OPENFT / FTAM / FTP		Partner type selected	x
ReqSel	String	*ALL / ONLY-SYNC / ONLY-ASYNC / ONLY-LOCAL / ONLY-REMOTE		Request type selected	x
Data	String	FORM	RAW	Output format (prepared / not prepared)	x
ThNetbTtl	Number	Number of bytes per second	Bytes, accumulated	Throughput in net bytes	x
ThNetbSnd	Number	Number of bytes per second	Bytes, accumulated	Throughput in net bytes, send requests	x
ThNetbRcv	Number	Number of bytes per second	Bytes, accumulated	Throughput in net bytes, receive requests	x
ThNetbTxt	Number	Number of bytes per second	Bytes, accumulated	Throughput in net bytes, text files	
ThNetbBin	Number	Number of bytes per second	Bytes, accumulated	Throughput in net bytes, binary files	
ThDiskTtl	Number	Number of bytes per second	Bytes, accumulated	Throughput in disk bytes	x

ThDiskSnd	Number	Number of bytes per second	Bytes, accumulated	Throughput in disk bytes, send requests	x
ThDiskRcv	Number	Number of bytes per second	Bytes, accumulated	Throughput in disk bytes, receive requests	x
ThDiskTxt	Number	Number of bytes per second	Bytes, accumulated	Throughput in disk bytes, text files	
ThDiskBin	Number	Number of bytes per second	Bytes, accumulated	Throughput in disk bytes, binary files	
ThRqto	Number	Number per second	Number, accumulated	openFT requests received	x
ThRqft	Number	Number per second	Number, accumulated	File transfer requests received	
ThRqfm	Number	Number per second	Number, accumulated	file management requests received	
ThSuct	Number	Number per second	Number, accumulated	Successfully completed openFT requests	x
ThAbrt	Number	Number per second	Number, accumulated	Aborted openFT requests	x
ThIntr	Number	Number per second	Number, accumulated	Interrupted openFT requests	x
ThUsrf	Number	Number per second	Number, accumulated	Requests from non-authorized users	x
ThFoll	Number	Number per second	Number, accumulated	Follow-up processing operations started	
ThCosu	Number	Number per second	Number, accumulated	Connections established	
ThCofl	Number	Number per second	Number, accumulated	Failed connection attempts	x
ThCobr	Number	Number per second	Number, accumulated	Disconnections as a result of connection errors	x
DuRqtlOut[1]	Number	Milliseconds	---	Maximum request duration Outbound	
DuRqtlInb[1]	Number	Milliseconds	---	Maximum request duration Inbound	
DuRqftOut[1]	Number	Milliseconds	---	Maximum request duration Outbound transfer	

DuRqftInb[1]	Number	Milliseconds	---	Maximum request duration Inbound transfer	
DuRqfmOut[1]	Number	Milliseconds	---	Maximum request duration Outbound file management	
DuRqfmInb[1]	Number	Milliseconds	---	Maximum request duration Inbound file management	
DuRqesOut[1]	Number	Milliseconds	---	Maximum outbound request waiting time	
DuDnscOut[1]	Number	Milliseconds	---	Maximum time an outbound openFT request was waiting for partner checking	
DuDnscInb[1]	Number	Milliseconds	---	Maximum time an inbound openFT request was waiting for partner checking	
DuConnOut[1]	Number	Milliseconds	---	Maximum duration time of establishment of a connection for an outbound openFT request	
DuOpenOut[1]	Number	Milliseconds	---	Maximum file open time (outbound)	
DuOpenInb[1]	Number	Milliseconds	---	Maximum file open time (inbound)	
DuClosOut[1]	Number	Milliseconds	---	Maximum file close time (outbound)	
DuClosInb[1]	Number	Milliseconds	---	Maximum file close time (inbound)	
DuUsrcOut[1]	Number	Milliseconds	---	Maximum user check time (outbound)	
DuUsrcInb[1]	Number	Milliseconds	---	Maximum user check time (ínbound)	
StRqas	Number (100)[2]	Average value	Current number	Number of synchronous requests in the ACTIVE state	x
StRqaa	Number (100)[2]	Average value	Current number	Number of asynchronous requests in the ACTIVE state	x
StRqwt	Number (100)[2]	Average value	Current number	Number of requests in the WAIT state	x
StRqhd	Number (100)[2]	Average value	Current number	Number of requests in the HOLD state	x

StRqsp	Number (100)[2]	Average value	Current number	Number of requests in the SUSPEND state	x
StRqlk	Number (100)[2]	Average value	Current number	Number of requests in the LOCKED state	x
StRqfi	Number (100)[2]	Average value	Current number	Number of requests in the FINISHED state	
StCLim	Number	Value currently set		Maximum number of connections established for asynchronous requests.	x
StCAct	Percent	Share of StCLim in %	Current number	Number of occupied connections for asynchronous requests	x
StRqLim	Number	Value currently set		Maximum number of asynchronous requests in request management	x
StRqAct	Percent	Share of StRqLim in %	Current number	Entries occupied in request management	x
StOftr	BOOL	1 / 0		openFT Protocol activated / deactivated	x
StFtmr	BOOL	1 / 0		FTAM Protocol activated / deactivated	x
StFtpr	BOOL	1 / 0		FTP Protocol activated / deactivated	x
StTrcr	BOOL	1 / 0		Trace activated / deactivated	

[1]is not output with option *-raw*

[2]number is the measured value multiplied by 100 (e.g. output 225 corresponds to value 2.25)

Examples

```
ftshwm -ty -csv @a
CurrTime;MonOn;PartnerSel;ReqSel;Data;ThNetbTtl;ThNetbSnd;ThNetbRcv;ThNetbTxt
;ThNetbBin;ThDiskTtl;ThDiskSnd;ThDiskRcv;ThDiskTxt;ThDiskBin;ThRqto;ThRqft;Th
Rqfm;ThSuct;ThAbrt;ThIntr;ThUsrf;ThFoll;ThCosu;ThCofl;ThCobr;DuRqtlOut;DuRqtl
Inb;DuRqftOut;DuRqftInb;DuRqfmOut;DuRqfmInb;DuRqesOut;DuDnscOut;DuDnscInb;DuC
onnOut;DuOpenOut;DuOpenInb;DuClosOut;DuClosInb;DuUsrcOut;DuUsrcInb;StRqas;StR
qaa;StRqwt;StRqhd;StRqsp;StRqlk;StRqfi;StCLim;StCAct;StRqLim;StRqAct;StOftr;S
tFtmr;StFtpr;StTrcr
*TIME;*TIME;*STRING;*STRING;*STRING;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*
INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*IN
T;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT;*INT(100);
*INT(100);*INT(100);*INT(100);*INT(100);*INT(100);*INT(100);*INT;*PERCENT;*IN
T;*PERCENT;*BOOL;*BOOL;*BOOL;*BOOL
ftshwm -csv ThNetbTtl ThDiskTtl
CurrTime;MonOn;PartnerSel;ReqSel;Data;ThNetbTtl;ThDiskTtl
2017-02-28 15:40:01;2017-02-28 15:36:12;OPENFT,FTAM;ONLY-ASYNC,ONLY-
REMOTE;FORM;2681262;524064
```

6.11 ftshwo

The following table indicates the CSV output format of the operating parameters.

The **Parameter** column contains the name of the output parameter during normal output, see Output format of ftshwo. Some parameters have fixed values because they are supported only for reasons of compatibility or have been replaced by other parameters.

Column	Type	Values and Meaning	Parameter
PartnerLim	Number	0	---
ReqLim	Number	Maximum number of requests	RQ-LIM
TaskLim	Number	Maximum number of processes	PROC-LIM
ConnLim	Number	Maximum number of connections	CONN-LIM
ReqWaitLev	Number	1	---
TransportUnitSize	Number	Maximum length of a transport unit	TU-SIZE
PartnerCheck	String	*STD / *TRANSP-ADDR / *AET / *BOTH Partner check: Identification / transport address / Application Entity Title (AET) / AET and transport address	PTN-CHK
SecLev	Number	0... 100 / *B-P-ATTR Default value for the security level of partners	SEC-LEV
TraceOpenft	String	*STD / *OFF Trace function for openFT partner activated / deactivated	FUNCT, line TRACE PARTNER-SELECTION
TraceOut	String	*FILE / empty Trace function activated / deactivated	FUNCT, line TRACE SWITCH
TraceSession	String	*OFF	---
TraceFtam	String	*STD / *OFF Trace function for FTAM partner activated / deactivated	FUNCT, line TRACE PARTNER-SELECTION
LogTransFile	String	*ON / *OFF FT logging activated / deactivated	FT-LOG
MaxInboundReq	Number	Maximum number of requests	(same as RQ-LIM)
MaxReqLifetime	String	Maximum lifetime of requests in the request queue / *UNLIMITED	MAX-RQ-LIFE

SnmpTrapsSubsystemState	String	*ON / *OFF SNMP traps on subsystem status change activated / deactivated	TRAP, line SNMP SS-STATE
SnmpTrapsFtState	String	*ON / *OFF SNMP traps on asynchronous server status change activated / deactivated	TRAP, line SNMP FT-STATE
SnmpTrapsPartnerState	String	*ON / *OFF SNMP traps on partner status change activated / deactivated	TRAP, line SNMP PART-STATE
SnmpTrapsPartnerUnreach	String	*ON / *OFF SNMP traps on unreachable partner systems activated / deactivated	TRAP, line SNMP PART-UNREA
SnmpTrapsReqQueueState	String	*ON / *OFF SNMP traps on request management status change activated / deactivated	TRAP, line SNMP RQ-STATE
SnmpTrapsTransSucc	String	*ON / *OFF SNMP traps on successfully terminated requests activated / deactivated	TRAP, line SNMP TRANS-SUCC
SnmpTrapsTransFail	String	*ON / *OFF SNMP traps on failed requests activated / deactivated	TRAP, line SNMP TRANS-FAIL
ConsoleTraps	String	*ON / *OFF Console traps (for at least one criterion) activated / deactivated	TRAP, line CONS
TeleService	String	empty	
HostName	String	Host name of the local computer / *NONE	HOST-NAME
Identification	String	Instance identification enclosed in double quotes	IDENTIFICATION
UseTns	String	*YES / *NO Use / do not use TNS in operation with CMX	USE TNS
ConsTrapsSubsystemState	String	*ON / *OFF Console traps on subsystem status change activated / deactivated	TRAP, line CONS SS-STATE
ConsTrapsFtState	String	*ON / *OFF Console traps on asynchronous server status change activated / deactivated	TRAP, line CONS FT-STATE

ConsTrapsPartnerState	String	*ON / *OFF Console traps on partner status change activated / deactivated	TRAP, line CONS PART-STATE
ConsTrapsPartnerUnreach	String	*ON / *OFF Console traps on unreachable partner systems activated / deactivated	TRAP, line CONS PART-UNREA
ConsTrapsReqQueueState	String	*ON / *OFF Console traps on request management status change activated / deactivated	TRAP, line CONS RQ-STATE
ConsTrapsTransSucc	String	*ON / *OFF Console traps on successfully terminated requests activated / deactivated	TRAP, line CONS TRANS-SUCC
ConsTrapsTransFail	String	*ON / *OFF Console traps on failed requests activated / deactivated	TRAP, line CONS TRANS-FAIL
FtLog	String	*ALL / *FAIL / *NONE Scope of FT logging	FT-LOG
FtacLog	String	*ALL / *FAIL / *MODIFY Scope of FTAC logging	FTAC-LOG
Trace	String	*ON / *OFF Trace function activated / deactivated	FUNCT, line TRACE SWITCH
TraceSelp	String	*ALL / OPENFT / FTP / FTAM / ADM / empty [1] Trace selection based on partner type	FUNCT, line TRACE PARTNER-SELECTION
TraceSelr	String	*ALL / ONLY-SYNC / ONLY-ASYNC / ONLY-LOCAL / ONLY-REMOTE [1] Trace selection based on request type	FUNCT, line TRACE REQUEST-SELECTION
TraceOpt	String	*NO-BULK-DATA / *NONE Minimum trace / no trace options	FUNCT, line TRACE OPTIONS
KeyLen	Number	768 / 1024 / 2048 / 3072 / 4096 RSA key length in bit	RSA-PROP
CcsName	String	Character set enclosed in double quotes	CCS-NAME
AppEntTitle	String	*YES / *NO In the case of FTAM, "nil-Application Entity Title" is sent / not sent	---
StatName	String	Name of the local openFT application$FJAM	LOCAL-SYSTEM-NAME

SysName	String	Name of the local system / empty	LOCAL-SYSTEM-NAME
FtStarted	String	*YES / *NO Asynchronous openFT server started / not started	STARTED
openftAppl	String	*STD / port number Port number of the local openFT server	OPENFT-APPL
ftamAppl	String	*STD / port number Port number of the local FTAM server	FTAM-APPL
FtpPort	Number	Port number Port number of the local FTP server	FTP-PORT
ftpDPort	Number	Value / empty (internal function)	---
ftstdPort	String	*STD / port number Default port for dynamic partners	---
DynPartner	String	*ON / *OFF Dynamic partner entries activated / deactivated	DYN-PART
ConTimeout	Number	Value (internal function)	---
ChkpTime	Number	Value (internal function)	---
Monitoring	String	*ON / *OFF Monitoring data activated / deactivated	FUNCT, line MONITOR SWITCH
MonSelp	String	*ALL / OPENFT / FTP / FTAM / empty [1] Selection based on type of partner system	FUNCT, line MONITOR PARTNER-SELECTION
MonSelr	String	*ALL / ONLY-SYNC / ONLY-ASYNC / ONLY-LOCAL / ONLY-REMOTE [1] Selection based on type of request	FUNCT, line MONITOR REQUEST-SELECTION
AdmTrapServer	String	Name of the ADM-TRAP server / *NONE	ADM-TRAP-SERVER
AdmTrapsFtState	String	*ON / *OFF ADM traps on asynchronous server status change activated / deactivated	TRAP, line ADM FT-STATE
AdmTrapsPartnerState	String	*ON / *OFF ADM traps on partner status change activated / deactivated	TRAP, line ADM PART-STATE

AdmTrapsPartnerUnreach	String	*ON / *OFF ADM traps on unreachable partner systems activated / deactivated	TRAP, line ADM PART-UNREA
AdmTrapsReqQueueState	String	*ON / *OFF ADM traps on request management status change activated / deactivated	TRAP, line ADM RQ-STATE
AdmTrapsTransSucc	String	*ON / *OFF ADM traps on successfully terminated requests activated / deactivated	TRAP, line ADM TRANS-SUCC
AdmTrapsTransFail	String	*ON / *OFF ADM traps on failed requests activated / deactivated	TRAP, line ADM TRANS-FAIL
AdminConnLim	String	Maximum number of administration connections	ADM-CLIM
AdmPort	String	Port number / *NONE Port number for remote administration	ADM-PORT
OpenftApplState	String	*ACTIVE / *INACT / *DISABLED / *NAVAIL Status of the openFT server	OPENFT-APPL, 2nd line
FtamApplState	String	*ACTIVE / *INACT / *DISABLED / *NAVAIL Status of the FTAM server	FTAM-APPL, 2nd line
FtpState	String	*ACTIVE / *INACT / *DISABLED / *NAVAIL Status of the FTP server	FTP-PORT, 2nd line
AdmState	String	*ACTIVE / *INACT / *DISABLED Status for inbound remote administration, on ADM trap server also status for receiving ADM traps	ADM-PORT, 2nd line
AdminLog	String	*ALL / *FAIL / *MODIFY / *NONE Scope of ADM logging	ADM-LOG
CentralAdminServer	String	*YES / *NO Local computer is remote administration server / not remote administration server	ADM-CS
ActiveAppl	String	*ALL / *NONE / OPENFT / FTAM / FTP / ADM[1] active servers	see 2nd line of OPENFT-APPL, FTAM-APPL, FTP-PORT, ADM-PORT
UseCmx	String	*YES / *NO Operation with CMX / without CMX	USE CMX
TraceOptLowerLayers	String	*DETAIL / *STD / *OFF Trace scope for lower protocol layers	OPTIONS-LL

EncMandIn	String	*YES / *NO Inbound encryption activated / deactivated	ENC-MAND (IN)
EncMandOut	String	*YES / *NO Outbound encryption activated / deactivated	ENC-MAND (OUT)
DelLog	String	*ON / *OFF Automatic deletion of log records activated / deactivated	DEL-LOG
DelLogRetpd	Number	Minimum age, in days, of the log records to be deleted. 0 means current day.	RETPD
DelLogRepeat	String	*MONTHLY / *WEEKLY / *DAILY Repeat interval for deletion of log records.	DEL-LOG ON
DelLogDay	Number	1..31 / 1..7 / 0 Day on which deletion is to be repeated. In case of DelLogRepeat = *MONTHLY then this is the day of the month, if DelLogRepeat = *WEEKLY then it is the day of the week (1 = Monday), if DelLogRepeat = *DAILY then 0 is output	DEL-LOG ON
DelLogTime	Time	Time of deletion	DEL-LOG AT
OutboundRecovery	String	*ON / *OFF Restart function for outbound requests	RECOVERY
InboundRecovery	String	*ON / *OFF Restart function for inbound requests	RECOVERY
ApplicationEntityTitle	String	"<AET>" (explicit output of the AET included in quotation marks) / *NSPEC / *IDENTIFICATION Setting of the Calling Application Entity Title	localAET (only on Unix /Windows systems)
RSAMinimum	Number	0 / 768 / 1024 / 2048 / 3072 / 4096 Minimum length of the RSA key	RSA-MIN
FileNameCcs	String	Unix systems: Character set used for displaying file names in the case of inbound request in character mode / empty	FN-CCS-NAME (only Unix systems) ---
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 of the lines
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	Attach of the FTAM protocol to the FarSync X.25 transport system. 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
FtDirLog	String	*ALL / *FAIL / *NONE Scope of logging in case of directory transfer	FT-DIR-LOG
AESMinimum	String	*NONE / 128 / 256 Minimum length of AES key	AES-MIN
FtAdmin	String	Windows systems: Name of the FT administrator / SYSTEM	FT-ADMIN (only Windows systems)

FtacAdmin	String	Windows systems: Name of the FTAC administrator / *STD	FTAC-ADMIN (only Windows systems)
FtAdmGroup	String	Linux systems: Name of the FT administrator group / *NONE	FT-ADMIN-GROUP (only Linux systems)

[1]Combinations of multiple values are also possible (not with *ALL or *NONE)

Example for Windows (X.25 parameters)

```
ftshwo –csv
...
X25Config;openftApplUseX25;openftApplX25ListInt;openftApplX25ListNum;openftApplX25
Class;openftApplX25Nsap;ftamApplUseX25;ftamApplX25ListInt;ftamApplX25ListNum;ftamA
pplX25Class;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";...
```

Example for Linux (X.25 parameters)

```
ftshwo –csv
...
X25Config;openftApplUseX25;openftApplX25ListInt;openftApplX25ListNum;openftApplX25
Nsap;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 on Linux is identical to the Windows output except the columns *openftApplX- 25ListInt* and *ftamApplX25ListInt*. The difference only lies in the values which are output because the notation of lines is different in Windows and Linux.

Example for Linux (FtAdmGroup) for a FT administrator group:

```
ftshwo –csv
...; FtAdmin;FtacAdmin;FtAdmGroup
...; "*NONE";*STD;"ftgroup"
```

Example for Linux (FtAdmGroup) for a single FT administrator:

```
ftshwo –csv
...; FtAdmin;FtacAdmin;FtAdmGroup
...; "hugo";*STD;"*NONE"
```

6.12 ftshwp

The following table indicates the CSV output format of an admission profile.

The **Std** column is not relevant on Unix and Windows systems.

The **Parameter** column contains the name of the output parameter during long output, see also *ftshwp*.

Column	Type	Values and Meaning	Parameter	Std
ProfName	String	Name of the profile enclosed in double quotes	(Profile name)	
Priv	String	*YES / *NO Profile is privileged / not privileged	PRIVILEGED	
TransAdm	String	*SECRET / *NSPEC Transfer admission has been assigned / not assigned	TRANS-ADM NOT-SPECIFIED	
Duplicated	String	*YES / *NO *YES means: profile is locked due to attempt to assign the transfer admission twice	TRANS-ADM DUPLICATED	
LockedByImport	String	*YES / *NO *YES means: profile is locked because it was imported	TRANS-ADM LOCKED (by_import)	
LockedByAdm	String	*YES / *NO *YES means: profile locked by FTAC administrator	TRANS-ADM LOCKED (by_adm)	
LockedByUser	String	*YES / *NO *YES means: profile locked by user	TRANS-ADM LOCKED (by_user)	
Expired	String	*YES / *NO *YES means: profile locked because period expired	TRANS-ADM EXPIRED	
ExpDate	String	Expiration date in short format yyyy-mm-dd / *NRES (no expiration date)	EXP-DATE	
Usage	String	*PUBLIC / *PRIVATE / *NSPEC Usage	USAGE	
IgnObs	String	*YES / *NO Ignore / do not ignore predefined value for Outbound Send	IGN-MAX-LEVELS OBS	
IgnObr	String	*YES / *NO Ignore / do not ignore predefined value for Outbound Receive	IGN-MAX-LEVELS OBR	

IgnIbs	String	*YES / *NO Ignore / do not ignore predefined value for Inbound Send	IGN-MAX-LEVELS IBS	
IgnIbr	String	*YES / *NO Ignore / do not ignore predefined value for Inbound Receive	IGN-MAX-LEVELS IBR	
IgnIbp	String	*YES / *NO Ignore / do not ignore predefined value for Inbound Processing	IGN-MAX-LEVELS IBP	
IgnIbf	String	*YES / *NO Ignore / do not ignore predefined value for Inbound File Management	IGN-MAX-LEVELS IBF	
Initiator	String	*LOC / *REM / *NRES Initiator: only local / only remote / unrestricted	INITIATOR	
TransDir	String	*FROM / *TO / *NRES Permitted transfer direction: from partner / to partner / unrestricted	TRANS-DIR	
MaxPartLev	Number	0... 100 / *NRES Maximum security level / security level unrestricted	MAX-PART-LEV	
Partners	String	One or more FT partners, delimited by commas and enclosed in double quotes / *NRES (no restriction)	PARTNER	
FileName	String	File name or file name prefix enclosed in double quotes / *NRES Restricts access to this file or files with this prefix. *NRES means there is no restriction	FILE-NAME	
Library	String	*NRES not relevant on Unix and Windows systems	LIBRARY	
FileNamePrefix	String	*YES / *NO The file name in FileName is a prefix / is not a prefix	FILE-NAME = (PREFIX=..)	
ElemName	String	*NRES	---	
ElemPrefix	String	*NO	---	
ElemVersion	String	*NRES	---	
ElemType	String	*NRES	---	

FilePass	String	*NRES	---	
Write	String	*NEW / *EXT / *REPL / *NRES Write rules	WRITE	
UserAdmId	String	User ID enclosed in double quotes	USER-ADM (user-id,...)	
UserAdmAcc	String	Account number enclosed in double quotes / *NRES	USER-ADM (..,account,...)	
UserAdmPass	String	*OWN / *YES / *NSPEC / *NONE Password is taken over / was specified / was not specified / is not required	USER-ADM (...,...,password)	
ProcAdmId	String	*NRES	---	
ProcAdmAcc	String	*NRES	---	
ProcAdmPass	String	*NRES	---	
SuccProc	String	Follow-up processing on success, enclosed in double quotes / *NONE / *NRES / *EXPANSION	SUCC-PROC	
SuccPrefix	String	Follow-up processing prefix on success, enclosed in double quotes / *NONE	SUCC-PREFIX	
SuccSuffix	String	Follow-up processing suffix on success, enclosed in double quotes / *NONE	SUCC-SUFFIX	
FailProc	String	Follow-up processing on error, enclosed in double quotes / *NONE / *NRES / *EXPANSION	FAIL-PROC	
FailPrefix	String	Follow-up processing prefix on error, enclosed in double quotes / *NONE	FAIL-PREFIX	
FailSuffix	String	Follow-up processing suffix on error, enclosed in double quotes / *NONE	FAIL-SUFFIX	
TransFile	String	*ALLOWED / *NOT-ALLOWED Transfer and delete files permitted / not permitted	FT-FUNCTION = (TRANSFER-FILE)	
ModFileAttr	String	*ALLOWED / *NOT-ALLOWED Modify file attributes permitted / not permitted	FT-FUNCTION = (MODIFY-FILE-ATTRIBUTES)	
ReadDir	String	*ALLOWED / *NOT-ALLOWED View directories permitted / not permitted	FT-FUNCTION = (READ-DIRECTORY)	
FileProc	String	*ALLOWED / *NOT-ALLOWED Preprocessing/postprocessing permitted / not permitted	FT-FUNCTION = (FILE-PROCESSING)	

AccAdm	String	*ALLOWED / *NOT-ALLOWED Access to remote administration server permitted / not permitted	FT-FUNCTION = (ACCESS-TO-ADMINISTRATION)	
RemAdm	String	*ALLOWED / *NOT-ALLOWED Remote administration via remote administration server permitted / not permitted	FT-FUNCTION = (REMOTE-ADMINISTRATION)	
Text	String	Text enclosed in double quotes / *NONE	TEXT	
DataEnc	String	*YES / *NO / *NRES Data encryption is mandatory / prohibited / neither mandatory nor prohibited	DATA-ENC	
ModDate	Date	Time of last modification	LAST-MODIF	
AdmTrapLog	String	*ALLOWED / *NOT-ALLOWED Reception of ADM traps permitted / not permitted	FT-FUNCTION = (ADM-TRAP-LOG)	
FileAtEnc	String	*YES / *NO / *NRES File management encryption is mandatory / prohibited / neither mandatory nor prohibited	FILE-ATTR-ENC	

6.13 ftshwptn

The following table indicates the CSV output format of a partner in the partner list.

The **Parameter** column contains the name of the output parameter during long output, see Output format of ftshwptn .

Column	Type	Values and Meaning	Parameter
PartnerName	String	Partner name enclosed in double quotes	NAME
Sta	String	*ACT / *DEACT / *NOCON / *LUNK / *RUNK / *ADEAC / *AINAC / *LAUTH / *RAUTH / *NOKEY / *DIERR / *IDREJ Partner status	STATE
SecLev	String	*STD / *B-P-ATTR / 1...100 Global security level / attribute-specific security level / fixed security level	SECLEV
Trace	String	*FTOPT / *STD / *ON / *OFF Trace setting	TRACE
Loc	Number	Number of locally issued file transfer requests to this partner	LOC
Rem	Number	Number of file transfer requests issued by this partner	REM
Processor	String	Processor name enclosed in double quotes / empty	ADDRESS
Entity	String	Entity name enclosed in double quotes / empty	ADDRESS
NetworkAddr	String	Partner address (network address without port number/selectors) enclosed in double quotes	ADDRESS
Port	Number	Port number	ADDRESS (port number)
PartnerCheck	String	*FTOPT / *STD / *TRANSP-ADDR / *AUTH / *AUTHM / *NOKEY Sender verification	P-CHK
TransportSel	String	Transport selector enclosed in double quotes / empty	ADDRESS (transport selector)
LastAccessDate	Date	Time of last access in short format yyyy-mm-dd	---
SessionSel	String	Session selector enclosed in double quotes / empty	ADDRESS (session selector)
PresentationSel	String	Presentation selector enclosed in double quotes / empty	ADDRESS (presentation selector)

Identification	String	Identification enclosed in double quotes / empty	IDENTIFICATION
SessRout	String	Routing information enclosed in double quotes / *ID / empty *ID means routing information same as identification	ROUTING
PartnerAddr	String	Partner address (including port number und selectors) enclosed in double quotesn	ADDRESS
Check	String	*FTOPT / *STD / *TRANSP-ADDR Partner check	P-CHK
AuthMand	String	*YES / *NO Authentication is mandatory / not mandatory	P-CHK
Priority	String	*LOW / *NORM / *HIGH Priority	PRI
AS3	String	*NO (internal function)	---
AuthLev	Number	1 / 2 / empty Authentication level	P-CHK
InboundSta	String	*ACT / *DEACT Inbound function activated / deactivated	INBND
RequProc	String	*STD / *SERIAL The processing mode for asynchronous outbound requests is parallel / is serial	REQU-P
OutboundRecovery	String	*FTOPT / *ON / *OFF Partner-specific restart function for asynchronous outbound requests	RECOV
ForeignPartner	String	*YES / *NO (internal function)	---
ScopeId	String	Scope ID (for IPv6) or line number (on Windows adapter number:line number) (X.25) / empty	ADDRESS
AddrType	String	*X25/*TCP/IP/*TNS Type of the adress	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 formatat	NSAP or AFI.IDI.DSP
CallUserData	String	User data for the connection setup (e.g. CUD transport protocol identification)on)	CUD

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
RSAProposal	String	*FTOPT | 0 | 768 | 1024 | 2048 | 3072 | 4096	RSA-PROP
RSAMinimum	String	*FTOPT | 0 | 768 | 1024 | 2048 | 3072 | 4096	RSA-MIN

Example for an X.25-Partner:

```
ftshwptn mchx25 –csv
PartnerName;Sta;SecLev;Trace;Loc;Rem;Processor;Entity;NetworkAddr;Port;Partne
rCheck;TransportSel;LastAccessDate;SessionSel;PresentationSel;Identification;
SessRout;PartnerAddr;Check;AuthMand;Priority;AS3;AuthLev;InboundSta;RequProc;
OutboundRecovery;ForeignPartner;ScopeId;AddrType;ExtensionID;DteAddress;NsapA
ddress;CallUserData;TransportClass;WindowSize;PacketSize;ClosedUserGroup;Thro
ughputClass;ReverseCharging;SpareInterface;RSAProposal;RSAMinimum
"mchx25";*ACT;*STD;*FTOPT;0;0;;;"%x25[123456789012345]";;*FTOPT;"$fjam";2016-
04-01;;;"%x25[123456789012345]";;"%x25[123456789012345%0:0]";*FTOPT;*NO;*NORM
;*NO;;*ACT;*STD;*FTOPT;*NO;"0:0";"*X25";3;"123456789012345";"43.123.45678901"
;"12345678901234567890123456789012";"2/2";7;4096;9999;192000;*NO;"1:0,2:0";1024;0
```

Example for an openFT-Partner:

```
ftshwptn TW01 -csv
PartnerName;Sta;SecLev;Trace;Loc;Rem;Processor;Entity;NetworkAddr;Port;Partne
rCheck;TransportSel;LastAccessDate;SessionSel;PresentationSel;Identification;
SessRout;PartnerAddr;Check;AuthMand;Priority;AS3;AuthLev;InboundSta;RequProc;
OutboundRecovery;ForeignPartner;ScopeId;AddrType;ExtensionID;DteAddress;NsapA
ddress;CallUserData;TransportClass;WindowSize;PacketSize;ClosedUserGroup;Thro
ughputClass;ReverseCharging;SpareInterface;RSAProposal;RSAMinimum
"TW01";*ACT;*STD;*FTOPT;0;0s;;;"MN122.cognitas.local";1100;*FTOPT;"$fjam";201
6-07-17;;;"MN122.cognitas.local";;"MN122.cognitas.local";*FTOPT;*NO;*NORM;*NO
;;*ACT;*STD;*FTOPT;*NO;"";"*TNS";;"";"";"";"";;;;;;"";1024;0
```

Example for an FTAM-Partner:

```
ftshwptn ftampar1 -csv
PartnerName;Sta;SecLev;Trace;Loc;Rem;Processor;Entity;NetworkAddr;Port;Partne
rCheck;TransportSel;LastAccessDate;SessionSel;PresentationSel;Identification;
SessRout;PartnerAddr;Check;AuthMand;Priority;AS3;AuthLev;InboundSta;RequProc;
OutboundRecovery;ForeignPartner;ScopeId;AddrType;ExtensionID;DteAddress;NsapA
ddress;CallUserData;TransportClass;WindowSize;PacketSize;ClosedUserGroup;Thro
ughputClass;ReverseCharging;SpareInterface;RSAProposal;RSAMinimum
"ftampar1";*ACT;*STD;*FTOPT;0;0;;;"mc122";4800;;"$ftam";2016-11-18;;;;;"ftam:
//mc122";;*NO;*NORM;*NO;;*ACT;*STD;*FTOPT;*NO;"";"*TNS";;"";"";"";"";;;;;;"";1024;0
```

6.14 ftshwr

The following table indicates the CSV output format of a request.

Short output is also possible with *ftshwr*, see "ftshwr".

The **Parameter** column contains the name of the output parameter during long output, see Output format of ftshwr.

Column	Type	Values and Meaning	Parameter
TransId	Number	Request ID	TRANSFER-ID
Initiator	String	*LOC / *REM Initiator is local / remote	INITIATOR
State	String	*LOCK / *WAIT / *HOLD / *FIN / *ACT / *CANC / *SUSP Request status	STATE
PartnerName	String	Name or address of the partner enclosed in double quotes	PARTNER
PartnerState	String	*ACT / *INACT / *NOCON / *INSTERR Partner status	PARTNER-STATE
TransDir	String	*TO / *FROM Transfer direction	TRANS
ByteNum	Number	Number of bytes transferred / empty	BYTECNT
LocFileName	String	File name in the local system enclosed in double quotes	LOC: FILE
LocElemName	String	empty	---
LocElemType	String	empty	---
LocElemVersion	String	empty	---
Prio	String	*NORM / *LOW Priority of the request	PRIO
Compress	String	*NONE / *BYTE / *ZIP Compressed transfer	COMPRESS
DataEnc	String	*YES / *NO User data is transferred encrypted / unencrypted	ENCRYPT
DiCheck	String	*YES / *NO Data integrity is checked / is not checked	DICHECK
Write	String	*REPL / *EXT / *NEW Write rules	WRITE

StartTime	String	Time at which the request is started (format yymm-dd hh:mm:ss) / *SOON (request is started as soon as possible)	START
CancelTime	String	Time at which the request is deleted from the request queue (format yy-mm-dd hh:mm:ss) / *NO (no delete time)	CANCEL
Owner	String	Local user ID enclosed in double quotes	OWNER
DataType	String	*CHAR / *BIN / *USER Data type	DATA
Transp	String	*YES / *NO Transfer transparent / not transparent	TRANSP
LocTransAdmId	String	User ID for accessing the local system, enclosed in double quotes / *NONE	LOC: TRANS-ADM (USER)
LocTransAdmAcc	String	empty	---
LocProfile	String	empty	---
LocProcAdmId	String	empty	---
LocProcAdmAcc	String	empty	---
LocSuccProc	String	Local follow-up processing on success, enclosed in double quotes / *NONE / empty	LOC: SUCC-PROC
LocFailProc	String	Local follow-up processing on error, enclosed in double quotes / *NONE / empty	LOC: FAIL-PROC
LocListing	String	empty	---
LocMonjv	String	empty	---
LocCcsn	String	Name of the character set in the local system enclosed in double quotes / *STD	LOC: CCSN
RemFileName	String	File name in the remote system enclosed in double quotes / *NSPEC / *NONE / empty	REM: FILE
RemElemName	String	empty	---
RemElemType	String	empty	---
RemElemVersion	String	empty	---
RemTransAdmId	String	User ID in the remote system enclosed in double quotes / *NONE	REM: TRANS-ADM=(user-id,...)

RemTransAdmAcc	String	Account number in the remote system enclosed in double quotes / empty	REM: TRANS-ADM=(..., account)
RemTransAdmAccount[1]	String	Account number in the remote system enclosed in double quotes / empty	REM: TRANS-ADM=(..., account)
RemProfile	String	*YES / *NONE *YES means access via FTAC admission profile	REM: TRANS-ADM=REMOTE-PROFILE
RemProcAdmId	String	empty	---
RemProcAdmAcc	String	empty	---
RemSuccProc	String	Remote follow-up processing on success, enclosed in double quotes / *NONE / empty	REM: SUCC-PROC
RemFailProc	String	Remote follow-up processing on error, enclosed in double quotes / *NONE / empty	REM: FAIL-PROC
RemCcsn	String	Name of the character set used in the remote system, enclosed in double quotes / *STD	REM: CCSN
FileSize	Number	Size of the file in bytes / empty	FILESIZE
RecSize	Number	Maximum record size in bytes / empty	RECSIZE
RecFormat	String	*STD / *VARIABLE / *FIX / *UNDEFINED Record format	RECFORM
StoreTime	Date	Time at which the request was entered in the request queue	STORE
ExpEndTime	Date	empty	---
TranspMode	String	*YES / *NO Transfer transparent / not transparent	TRANSP
DataEncrypt	String	*YES / *NO User data transferred encrypted / unencrypted	ENCRYPT
TabExp	String	*AUTO / *YES / *NO Tabulator expansion	TABEXP
Mail	String	*ALL / *FAIL / *NO Result messages	LOC: MAIL
DiagCode	String	Diagnostic information / empty	DIAGCODE
FileAvail	String	*IMMEDIATE / *DEFERRED / *NSPEC Availability (for FTAM only)	AVAILABILITY

StorageAccount	String	Account number (for FTAM only) / empty	STOR-ACCOUNT
AccessRights	String	FTAM access rights / empty Possible values are @r, @w or combinations of r, i, p, x, e, a, c, d	ACCESS-RIGHTS
LegalQualif	String	Legal qualification (for FTAM only) / empty	LEGAL-QUAL
PartnerPrio	String	*LOW / *NORM / *HIGH Partner priority	PARTNER-PRIO
TargetFileForm	String	*STD / *BLOCK / *SEQ File format in the target system	TARGFORM
TargetRecForm	String	*STD / *UNDEFINED Record format in the target system	TRECFRM
Protection	String	*STD / *SAME Transfer of protection attributes	PROTECT
GlobReqId	Number	Global request identification For locally issued requests, same as request ID; for globally issued requests, same as the request ID in the initiating system	TRANSFER-ID or GLOB-ID
FNCMode	String	*TRANSPARENT / *CHAR Encoding mode for remote file names and followup processing	FNC-MODE
Progress	Number	number1/number2 / *NSPEC Number of completed/total subrequestes	PROGRESS

[1]RemTransAdmAcc and RemTransAdmAccount have the same meaning and the same content. For reasons of compatibility, both parameters are present in the CSV output.

Short output from ftshwr in CSV format

ftshwr -s -csv outputs a table with two rows indicating the number of requests that have the corresponding status, see also "Standard ftshwr output".

Column	Type	Values
Act	Number	Number of requests with the status ACTIVE
Wait	Number	Number of requests with the status WAIT
Lock	Number	Number of requests with the status LOCK
Susp	Number	Number of requests with the status SUSPEND
Hold	Number	Number of requests with the status HOLD

Fin	Number	Number of requests with the status FINISHED
Total	Number	Total number of requests

Example

```
ftshwr -s -csv
Act;Wait;Lock;Susp;Hold;Fin;Total
0;1;0;0;2;0;3
```

6.15 ftshws

The following table indicates the CSV output format of the status of the user’s openFT-Script requests.

Column	Type	Values and Meaning
User	String	User ID under which the request was started, enclosed in double quotes.
Ftscriptid	String	Unique identification of the request, enclosed in double quotes. The identification is returned by the *ftscript* command.
State	String	W / R / T / F / I / C / X Processing status, see table format (Sta).
CreationTime	String	Time at which the openFT-Script request was created, in the format yyyy-mm-dd hh:mm:ss.
FtscriptFileName	String	Path name of the script file, enclosed in double quotes.
Error	String	Cause of error in clear text enclosed in double quotes in the case of openFT-Script requests with status F, otherwise empty.

6.16 ftshwsuo

The following table indicates the CSV output format of the directory in which the openFT-Script requests are to be stored.

Column	Type	Values
User	String	User ID
FtscriptWorkdir	String	Name of the openFT-Script working directory
ThreadLimit	String	Thread limit for a user
TransferFileLimit	String	File transfer limit for a user or *CLIM

7 Appendix

This chapter lists the commands in the tool command library, describes the samples delivered with openFT and the CSV outputs from the openFT commands.

7.1 Tool Command Library

The following tool commands are supplied with openFT:

- ft_tar
- ft_gzip
- ft_b2u and ft_u2b
- ft_mget
- ft_cexsv and ft_cexcl (only for Windows systems)

ft_tar and ft_gzip are the Gnu tar and Gnu zip tools subject to the Gnu Public License (GPL). These tools are supplied with openFT but are not subject to the openFT license, which means that you can copy and distribute them as long as you abide by the GPL. Fujitsu Technology Solutions reserves the right to stop supplying these tools in following versions or corrections versions of openFT or to supply them although they are not fully compatible with these versions. Renaming the tools to *ft_tar* and *ft_gzip* serves only to prevent collisions of installations on the various platforms.

An openFT user can therefore use these functions in procedures, preprocessing, postprocessing or follow-up processing with a defined scope of functions. You can call up a short description of the functionality available using the "--help" option. You should only use the subset of functions described below if possible to minimize the possibility of encountering incompatibilities in later versions.

7.1.1 ft_tar

GNU 'tar' saves many files together into a single tape or disk archive, and can restore individual files from the archive.

Usage

ft_tar [OPTION]... [FILE]...

If a long option shows an argument as mandatory, then it is mandatory for the equivalent short option also. Similarly for optional arguments.

Main operation mode:

-t, --list list the contents of an archive

-x, --extract, --get extract files from an archive

-c, --create create a new archive

-r, --append append files to the end of an archive

-u, --update only append files newer than copy in archive

Operation modifiers:

-k, --keep-old-files don't overwrite existing files when extracting

-U, --unlink-first remove each file prior to extracting over it

--recursive-unlink empty hierarchies prior to extracting directory

-O, --to-stdout extract files to standard output

Device selection and switching:

-f, --file=ARCHIVE use archive file or device ARCHIVE

Archive format selection:

-z, --gzip, --ungzip filter the archive through gzip

Informative output:

--help print this help, then exit

--version print tar program version number, then exit

-v, --verbose verbosely list files processed

FILE may be a file or a device.

This `tar' defaults to `-f- -b20'.

Report bugs to <tar-bugs@gnu.org>.

7.1.2 ft_gzip

Usage

ft_gzip [-OPTION] [file ...]

-c --stdout write on standard output, keep original files unchanged

-d --decompress decompress

file... files to (de)compress. If none given, use standard input.

7.1.3 ft_b2u and ft_u2b

These two commands are used to convert data between binary format and user format (record length fields).

- The *ft_b2u* command converts binary data into data in user format (records with record length fields). It reads the data from *stdin* and outputs it at *stdout*.
- The *ft_u2b* command converts data in user format (records with record length fields) into binary data.

Format

ft_b2u -r=<1...32000> [-rf=1...32000>] [-rl=<1...32000>]

ft_u2b <inputfile> [<outputfile>]

Description

-r Length of the records into which the byte stream is to be converted.

-rf Optional: Length of the first record.

-rl Optional: Length of the last record.

inputfile

Name of the file in user format or '-' (hyphen) for *stdin*.

outputfile

Name of the binary file.

Default value: *stdout*

Example

```
cat file.in|ft_b2u -r=100 > file.out (Unix systems)

type file.in|ft_b2u -r=100 > file.out (Windows systems)
```

7.1.4 ft_mget

Note on usage

Function: Fetching multiple files

User group: FT user

Functional description

ft_mget allows you to fetch synchronously or asynchronously multiple files from a remote partner computer. You specify the files using wildcards. To do this, *ft_mget* uses the *ncopy* (synchronous) or the *ft* (asynchronous) command internally. The transfer mode (synchronous or asynchronous) is controlled via the *-async* option.

Format

ft_mget -h |

[-async]
[-case=y | -case=n]
[-t | -u | -b][-x]
[-o | -e | -n]
[-k | -z][-c][-S | -s][-m=n | -m=f | -m=a] *)
<partner 1..200>!<file name with wildcard 1..512>
<prefix 0..511>%s
<transfer admission 8..67> | @n |

<user ID 1..67>[,[<account 1..64>][,<password 1..64>]]

[-p=[<password 1..64>]][-di]
[-lc=<CCS name 1..8>][-rc=<CCS name 1..8>]
[-ls=<follow-up proc 1..1000>][-lf=<follow-up proc 1..1000>]
[-rs=<follow-up proc 1..1000>][-rf=<follow-up proc 1..1000>]
[-r=v[<1..65535>] | -r=f[<1..65535>] | -r=u[<1..65535>] |
-r=<1..65535>]
[-tff=b | -tff=s][-trf=u]
[-av=i | -av=d] [-ac=<new account number 1..64>]
[-am=[r][i][p][x][e][a][c][d] | -am=@rw | -am=@ro]
[-lq=<legal qualification 1..80>]
[-pr=n | -pr=l]
[-sd=yyyymmdd | +<start date 0..ddddd>]
[-st=[+]<start time hhmm>]
[-cd=yyyymmdd | +<cancel date 0..ddddd>]
[-ct=[+]<cancel time hhmm>]
[-md]

*) The option *-m* is only available on Unix systems

Description

Only the differences compared with the *ncopy* and *ft* command are described below. The other parameters have the same meanings as in the *ncopy* command and the *ft* command.

Note that the same conditions apply to the *-c* option (encryption of user data) as for the *ft* or *ncopy* command, i.e. openFT-CR must be installed and the partner system must support encryption.

-async

The files are fetched asynchronously. In this event, you must not specify the *-s* option. All other parameters are permitted.

> **i** In the case of asynchronous transfer, the number of transfer requests that can be processed simultaneously is limited by the size of the request queue. If you wish to fetch a large number of files asynchronously using ft_mget, the FT administrator may have to increase the maximum size of the request queue. For further details, refer to the manual "openFT (Unix and Windows systems) - Installation and Operation".

-async not specified

If you omit -async, the files are fetched synchronously. In this event, you must not specify the following options:

- *-ls* and *-lf* (local follow-up processing)
- *-pr* (priority)
- *-sd* and *-st* (start date and time)
- *-cd* and *-ct* (deletion date and time)

All other parameters are permitted.

-case=y | **-case=n**

The option *-case* sets the consideration or non-consideration of upper case / lower case in the file name pattern. It does not influence the determination of the directory name.

y

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.

n

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.

-case not specified

If -case is not specified, then the following is valid: file names are case-sensitive with Unix and POSIX systems. Other partner systems are not case-sensitive.

-c

Controls the encryption option for user data and file/directory list attributes. I.e. the specifications also apply to file management requests (unlike file transfer commands).

transfer-admission | **@n** | userid[,[account][,password]]

Specification of the transfer admission is mandatory. Blanking of your entry is not supported. You are therefore not permitted to specify either the value *@d* or a user ID without password in the form *userid*,[*account*].

filename with wildcard

Specifies which files are to be fetched from the remote system.

You can only use wildcard characters in the final part of the name following the last slash (/) or backslash (\), not in the directory name. A BS2000 partner is regarded as a POSIX system if the specified file name is a POSIX pathname, i.e. starts with / or ./.

If the *-async* option has not been specified then all files that match the pattern specified under *file name with wildcard* are transferred to the local computer synchronously by *ft_mget* in a loop of *ncopy* commands. Otherwise asynchronous transfer requests are issued in the loop by means of *ft* commands.

The following characters can be used to define a wildcard pattern:

*

as a wildcard for any string (including an empty string).

?

as a wildcard for any single character.

[chars]

as a wildcard for a single character from the set specified by *chars*. In *chars*, you can list individual characters or specify one or more character ranges in the form a-z.
This selects all characters a through z (inclusive).
Example: [aeiX-Z] stands for one of the characters a e i X Y Z.

x as a wildcard for one only of the following characters: * ? [ ] \

The backslash is used to cancel the special meaning of these characters in the specified wildcard pattern.

> **i** On Unix systems, steps must be taken to ensure that wildcard characters and the exclamation mark (!) are not interpreted or resolved by the local shell. For this reason, we strongly recommend that you enclose the expression *<partner 1..200>!<file name with wildcard 1..512>* in single quotes, i.e. enter it in the form*'<partner 1..200>!<file name with wildcard 1..512>'*, e.g. `ft_mget 'server01!*.pdf'` ....

prefix%

Determines the names of the receive files in the local system.

You can specify %, %BASENAME, prefix%, or prefix%BASENAME:

% or %BASENAME

Each of these are replaced by the last part of the name of the remote file. The last part of the name starts after the last slash (/) or backslash (\) or a corresponding character in the remote system.

prefix% or prefix%BASENAME

You can also specify an optional prefix, e.g. *saved*.%BASENAME.
This prefix must end with a dot (.), a slash (/) or a backslash (\). The prefix can also contain the absolute or relative path of a directory that exists on the local computer. If the specified directory does not exist, *ft_mget* is not executed.

Note that the resulting file name must comply with the rules of the local system, otherwise the files will not be transferred.

If, for instance, the last part of the name of a file that matches the search pattern contains double quotes (") when fetching files from a Unix system to a Windows system, transfer of this file fails, because, unlike Unix systems, Windows systems do not permit quotes in file names.

Result messages and return codes

On success, *ft_mget* issues one of the following messages:

`<n> files successfully transferred` (synchronous transfer)

`Transfer of <n> files successfully initiated` (asynchronous transfer)

Where <*n*> stands for the number of files transferred synchronously or the number of asynchronous file transfer requests initiated. If no files that match the specified pattern were found on the remote system, the following message appears instead:

`No files corresponding to specified pattern found`

ft_mget normally terminates with the return code 0. If an error occurs during execution, the command terminates and returns one of the following return codes (*RC*):

RC	Output to stderr	Meaning
1	Invalid source parameter '<par>'. Source expected as <partner 1..200>!<file name with wildcard 1..512>.	The specification of the parameter used to specify the files to be transferred does not match the required format.
1	ft_mget syntax help	One of the mandatory parameters for *ft_mget* was not specified.
1	Invalid transfer admission specified.	*@d* or *userid*,[*account*], was specified in place of a transfer admission.
1	Parameter(s) '<par>' only allowed together with '-async'	The parameters <par> are only allowed for asynchronous file transfer.
1	Parameter(s) '<par>' must not be specified together with '-async'	The parameters <par> are not allowed for asynchronous file transfer.
2	Given target directory '<dir>' does not exist.	The target directory specified does not exist on the local system.

3	Given target path must contain %, %BASENAME, or %FILENAME.	The parameter specified for the target of *ft_mget* does not end with one of the specified placeholders.
4	openFtCmd <ftshw> failed	The openFT command *ftshw* for determining the files in the specified remote directory failed.
5	ft::isAbort after openFtCmd <ftshw>	The openFT command *ftshw* for determining the files in the specified remote directory failed.
6	Remote directory <dir> on host <partner> could not be accessed (return code= '<rc>', exit code='<code>').	It is not possible to access the specified directory on the remote partner system.
6	Reading content of remote directory <dir> on host <partner> failed (return code='<rc>', exit code='<code>').	It was not possible to read the specified directory on the remote partner system.
7	Not all files successfully transferred	At least one source file could not be transferred to the local system. The previous message(s) indicate(s) the file(s) concerned: Transfer of file '<file>' failed. Reason: '<rc>'

Example

You want to fetch synchronously all files on the remote computer located in a specific subdirectory and whose names start with *cfg* onto the local computer and store them there in the *config* subdirectory of the current directory. *mytad001* is a valid FTAC transfer admission for the remote computer.

- The local system and the remote system *MCH0001X* are Unix systems.

  If the files are located in the *tmp/config* directory, the command is as follows:

  ```
  ft_mget  'MCH0001X!/tmp/config/cfg*'
           config/copy.%BASENAME mytad001
  ```

  If, for instance, the source directory contains the files *cfg001*, *cfg002* and *cfg003*, *ft_mget* creates the local receive files *config/copy.cfg001*, *config/copy.cfg002* and *config/copy.cfg003*.

- The local system and the remote system *MCH0001W* are Windows systems.

  If the files are located in the *D:\tmp\config* directory, the command is as follows:

  ```
  ft_mget  MCH0001W!D:\tmp\config\cfg*
           config\copy.%BASENAME mytad001
  ```

  If, for instance, the source directory contains the files *cfg001*, *cfg002* and *cfg003*, *ft_mget* creates the local receive files *config\copy.cfg001*, *config\copy.cfg002* and *config\copy.cfg003*.

7.1.5 Command Execution Tool

The Command Execution Tool is only available on Windows systems.

The Command Execution Tool allows you to start applications which run interactively in the logon session of the user from within follow-up processing. The tool comprises the Command Execution Server and the Command Execution Client.

The Command Execution Server (**CES**) must be started in the logon session of the user for whom openFT is to start an application interactively. The Command Execution Client (CEC) is called during follow-up processing.

Format

Server: ft_cexsv.exe [<ID 1..16>]

Client: ft_cexcl.exe [–sv=<ID 1..16>] <command 1..2000>

Description

ID

is an optional identifier for the CES with a maximum length of 16 characters. It serves to distinguish concurrent instances of CES. The identifier is defined when CES is started and is read by the client with the parameter *-sv*.

command

designates the commands to be executed.

Notes

- A CES can be started automatically, for instance by making an entry in the Startup group. After the CES is started, a small icon (globe) appears in the task bar.
- The CES listens for commands that the Command Execution Client (CEC) sends to it for execution. The CES and CEC together ensure that only those CECs and CESs communicate with each other that belong to the same user context. This prevents User A from executing processing in an interactive logon session of User B with the rights of the latter.
- The CES terminates automatically when the user logs off or when the computer is shut down. It can be terminated manually by right-clicking on the CES icon and choosing Exit from the context menu that appears.
- The CEC is simply called during follow-up processing. The command string to be executed is passed as an argument when the application is started. If the optional parameter *–sv* is to be specified when the CEC is started in order to address a particular CES, *–sv* must be the first parameter passed when CEC is started.

Examples

1. User A receives files from a partner via openFT and these are to be edited using the WordPad (*write.exe*) word processing program after the files have been transferred successfully. In order to achieve this, User A can, for instance, set up an FTAC protocol in which follow-up processing is defined which starts the Command Execution Client and passes the command to be executed to CES.
   - Starting the CES:

     ```
     ft_cexsv.exe
     ```
   - Follow-up processing command for starting the CEC:

     ```
     ft_cexcl.exe write.exe %FILENAME
     ```

2. User A has opened two sessions with a terminal server. Files received over the admission profile *profile1* are to be opened with *write.exe* in the first terminal session. Files received over the admission profile *profile2* are to be opened with Microsoft's *Excel.exe* in the second terminal session.
   - Starting the CES in the first terminal session:

     `ft_cexsv.exe` *Session1*
   - Starting the CES in the second terminal session:

     `ft_cexsv.exe` *Session2*
   - Follow-up processing command for starting the CEC for admission profile *profile1:*

     `ft_cexcl.exe –sv=` *Session1* `write.exe %FILENAME`
   - Follow-up processing command for starting the CEC for admission profile *profile2*:

     `ft_cexcl.exe –sv=` *Session2* `Excel.exe %FILENAME`

7.2 Sample files

openFT is supplied with a range of sample files that you can use for various purposes. Once openFT has been installed, you will find these files under

/opt/openFT/samples (Unix systems)

openFT-installation-directory\samples (Windows systems)

ftadm

The file *config.xml* contains a simple sample configuration for remote administration. You can use this sample as a template and adapt it according to your needs.

ftscript

This directory contains examples for the openFT-Script interface. You will find a description of the interface in the manual "openFT (Unix and Windows systems) - openFT-Script Interface".

filedist.ftsc

Distribute files to several different partner systems.

transsuc.ftsc

Transfer a file to a partner system with follow-up processing.

treecopy.ftsc

Transfer a complete directory tree to a partner system.

msexcel

This directory contains the Microsoft Excel file *ftaccnt.xlt* and *openft32.xls*.

ftaccnt.xlt

Microsoft Excel template (Microsoft Excel 2003 and 2007). The template demonstrates how to evaluate the CSV output format of the logging commands and how to use them in Microsoft Excel for accounting purposes.

openft32.xls

Adds the menu *openFT* containing the commands *Transfer this File* and *Cancel Transmission* to the menu bar of Microsoft Excel. The file contains the associated Microsoft Visual Basic macros. The description of the macros is displayed when you open the file.

sample1.c, sample2.c, sample3.c, sample4.c, sample5.c, sample6.c, sample7.c

These examples are located in the *ftapi* subdirectory.

The examples illustrate various options for using the C programming interface of openFT. You will find a description of the examples in the manual "openFT (Unix and Windows systems) - Program Interface".

sample1.c

Transfer a file asynchronously

sample2.c

Transfer several files with follow-up processing.

sample3.c

Show the contents of a remote directory.

sample4.c

Execute a command on the partner system.

sample5.c

Run a loop that reads in, in quantities equivalent to the size of the buffer, the file attributes of all the files in a remote directory.

sample6.c

Create a directory on partner system.

sample7.c

Delete an empty directory on partner system.

Sample1.java, Sample2.java, Sample3.java, Sample4.java, Sample5.java, Sample6java, Sample7.java

These examples are located in the *java* subdirectory.

The examples illustrate the Java programming interface of openFT. How to compile and run the examples is described in the manual "openFT (Unix and Windows systems) - Program Interface".

Sample1.java

Transfer a file asynchronously

Sample2.java

Transfer several files with follow-up processing.

Sample3.java

Show the contents of a remote directory.

Sample4.java

Execute a command on the partner system.

Sample5.java

Run a loop that reads in, in quantities equivalent to the size of the buffer, the file attributes of all the files in a remote directory.

Sample6.java

Create a new directory on partner system.

Sample7.java

Delete an empty directory on partner system.

treecopy-get, treecopy-send, treecopy-send-unique (Unix systems)

These shell scripts illustrate various ways of transferring a complete directory to Unix or Windows partner systems.

treecopy-get

Fetch all files of a directory from a partner system using preprocessing. In this example, preprocessing is used in the remote system without an intermediate file being specified.

treecopy-send

Pack all files of a directory in a tar archive using preprocessing, transfer them to a partner system and unpack them there using postprocessing.

treecopy-send-unique

Pack all files of a directory in a tar archive using preprocessing, transfer them to a partner system and unpack them there using follow-up processing.

The use of %UNIQUE in the receive file name allows several scripts to be executed concurrently.

msword (Windows systems)

The file *openFT32.dot* is a Microsoft Word template (Microsoft Word 2003 and 2007). Documents created with this template can transfer themselves to a partner system where they can be printed if necessary. To do this, the template must be installed in the folder for Microsoft Word templates locally on your computer and on the partner system.

openFT32.dot adds the menu *openFT* with the commands *Transfer Document*, *Cancel Transfer* and *Send Clipboard* to the menu bar of Microsoft Word. The file contains the associated Microsoft Visual Basic macros. The description of the macros is displayed when you open the file.

ocxdemo (Windows systems)

This sample illustrates the use of the OCX control *fttrans.ocx*. The directory contains the relevant Microsoft Visual Basic application and the associated source code. You will find a description of OCX control in openFT fin the *Readme* file in the *ocxdemo* directory and in the manual "openFT (Unix and Windows systems) - Program Interface".

The Microsoft Visual Basic application is started and the OCX control for openFT is loaded by calling *ocxdemo.exe*.

shellext (Windows systems)

This directory contains the program library and the install file and uninstall file for the openFT shell extension. The openFT shell extension allows you to create predefined send patterns on the desktop in order to start a file transfer to a defined partner by dragging and dropping the file onto the send pattern from the Windows Explorer.

After installation, you create a new send pattern as follows:

- Choose *New* from the desktop context menu and then choose the entry *openFT Send Pattern*.

  A new icon whose name you can change is created on the desktop. The file name extension *.openFTst* must not be deleted from the name. You can then configure the openFT-specific parameters.

- Right-click the send pattern icon and choose *Properties* from the menu. The *Properties* dialog box is displayed.

  This contains the following additional openFT-specific tabs:

  openFT General
  Specifications on the partner system

  openFT Options
  Specifications on the send request options

simple (Windows systems)

The file *ncopy.c* illustrates how to call the commands from a program using the associated DLLs using the *ncopy* command as an example.

www (Windows systems)

The sample programs in this directory show how you can use openFT for downloading in the Internet or in an intranet. The example uses a Windows system as a client and a Unix system as the server platform. You will find a detailed description of the concept and the way in which it is implemented in the *Readme* file in the *www* directory.