

English



openFT V12.0 for BS2000/OSD

Program Interfaces

Programmer Reference Guide

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:

manuals@ts.fujitsu.com

Certified documentation according to DIN EN ISO 9001:2008

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 9001:2008.

cognitas. Gesellschaft für Technik-Dokumentation mbH

www.cognitas.de

Copyright and Trademarks

Copyright © Fujitsu Technology Solutions GmbH 2012.

All rights reserved.

Delivery subject to availability; right of technical modifications reserved.

All hardware and software names used are trademarks of their respective manufacturers.

Contents

1	Introduction	5
1.1	Brief description of the product openFT	6
1.2	Target group and objectives of this manual	6
1.3	Concept of the openFT for BS2000/OSD manual	7
1.4	Structure of the programming manual	8
1.5	Changes since the last version of the manual	8
1.6	README file	9
2	ASSEMBLER programming interface	11
2.1	Macros for openFT for BS2000	14
2.1.1	NCAN - Cancel file transfer request	15
2.1.2	NCOPY - Transfer file	18
2.1.3	NCOPYSYN - Transfer file	29
2.1.4	NDEL - Delete remote file	38
2.1.5	NLMOD - Modify local FT file attributes	41
2.1.6	NLSHOW - Display local FT file attributes	45
2.1.7	NMOD - Modify remote file attributes	47
2.1.8	NSHOW - Display remote file attributes	52
2.1.9	NSTAT - Query file transfer status	56
2.1.10	Set YNDEQU constants for the openFT macros	60
2.1.11	Return codes	65
2.1.11.1	VERSION parameter	65
2.1.11.2	Format of the return codes	65
2.1.11.3	Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW	68
2.1.11.4	Return codes for NCOPYSYN	81
2.1.11.5	Return codes for NCAN	88
2.1.11.6	Return codes for NSTAT	89

2.2	The openFT-AC programming interface	90
2.2.1	Macros for openFT-AC for BS2000	90
2.2.2	CREFTPRF - Create admission profile	91
2.2.3	DELFTPRF - Delete admission profile	100
2.2.4	EXPFTENV - Export admission sets and profiles	102
2.2.5	IMPFTENV - Import admission sets and profiles	104
2.2.6	MODFTADS - Modify admission sets	106
2.2.7	MODFTPRF - Modify admission profile	109
2.2.8	SHWFTADS - Output admission sets	119
2.2.9	SHWFTENV - List admission sets and profiles	121
2.2.10	SHWFTPRF - Output admission profile	123
2.2.11	SHWFTRGE - List accessible partner systems	125
2.2.12	YFSEQU - Set constants for the openFT-AC macros	127
2.2.13	Example for the use of the programming interface	128
2.2.14	Return information and error displays	130
3	COBOL programming interface	137
<hr/>		
3.1	COBOL macros for openFT for BS2000	137
3.1.1	NCANCEL - Cancel file transfer request	138
3.1.2	NCOPY - Transfer file	142
3.1.3	NDEL - Delete remote file	151
3.1.4	NLMOD - Modify local FT file attribute	156
3.1.5	NLSHOW - Display local FT file attributes	162
3.1.6	NMOD - Modify remote file attributes	167
3.1.7	NSHOW - Display remote file attributes	173
3.1.8	NSTATUS - Query file transfer status	178
3.1.9	Return information and error displays	181
3.1.9.1	Return codes for the CALL macros NCOPY, NCANCEL and NSTAT	183
3.1.9.2	Return code for the file management CALL macros	187
	Related publications	189
<hr/>		
	Index	191
<hr/>		

1 Introduction

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 Technology Solutions offers an extensive range of file transfer products - the openFT product range - for the following system platforms:

- BS2000/OSD[®]
- Solaris[™] (SPARC[®]/Intel[™]), LINUX[®], AIX[®], HP-UX[®]
- Microsoft[®] Windows Vista[™], Windows[™] 7, Windows Server 2008[™] and Windows Server 2008 R2[™]
- z/OS (IBM[®])

1.1 Brief description of the product openFT

openFT for BS2000/OSD is the file transfer product for computers using the operating system BS2000/OSD.

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

openFT allows the use of TCP/IP, ISO TP0/2, ISO TP4, SNA and NEA as transport protocols.

The range of functions made available by openFT can be extended using the add-on products openFT-FTAM, openFT-FTP and openFT-AC:

- openFT also supports the FTAM file transfer protocol (File Transfer Access and Management) standardized by ISO (International Organization for Standardization). This makes it possible to interconnect to systems of other manufacturers whose FT products also support the same standard.
- openFT-FTP supports FTP functionality.
- openFT-AC provides extended system and data access protection. FTAC stands for File Transfer Access Control.

1.2 Target group and objectives of this manual

This manual is aimed at users who wish to program FT applications with the help of the openFT and openFT-AC programming interfaces.

To understand this manual, it is necessary to have a knowledge of the BS2000/OSD operating system as well as of the ASSEMBLER and COBOL programming languages.

The programming manual is intended as a supplement to the user manual. In view of the provided information and references to command descriptions, these two manuals should always be used in conjunction.

1.3 Concept of the openFT for BS2000/OSD manual

The complete description of openFT for BS2000 and its optional components openFT-FTAM for BS2000 and openFT-AC for BS2000 is contained in three manuals. In addition to this Programmer Reference Guide, there is also a User Guide and a System Administrator Guide. The description is divided between the three manuals as follows:

- openFT for BS2000 - Installation and Administration

The System Administrator Guide is intended for FT and FTAC administrators. It describes:

- the installation of openFT and its optional components
- operation, control and monitoring of the FT system and the FTAC environment
- the administration commands for FT and FTAC administrators
- account records

- openFT for BS2000 - Managed File Transfer in the Open World

The User Guide contains the following information:

- an overview of the basic functions of the openFT product family
- a detailed description of the conventions for the file transfer to computers with different operating systems
- information on the implementation of FTAM
- description of the user commands
- messages from openFT and openFT-AC

- openFT for BS2000 - Programming Interfaces

The Programmer Reference Guide describes the openFT and openFT-AC program interfaces.

You will also find current information on the Internet under de.ts.fujitsu.com/openft (german) or ts.fujitsu.com/openft (english).

1.4 Structure of the programming manual

This manual describes the openFT and openFT-AC programming interfaces. It is divided into three chapters.

- The introduction in Chapter 1 provides information on the structure and contents of openFT for BS2000 documentation in general and this manual in particular.
- Chapter 2 describes the Assembler programming interface and contains the return codes both for openFT and openFT-AC functions.
- Chapter 3 discusses the COBOL programming interface.

1.5 Changes since the last version of the manual

This programming manual for openFT V12.0 for BS2000 contains the following new features compared to the manual for openFT V11.0 for BS2000:

Default account number of a user

The account number can be omitted in transfer requests and admission profiles. In this case a user's default account number is used. This results in the following changes:

- In the assembler macro NCOPY you can omit the LTACC and LPACC operands.
- In the assembler macro NCOPYSYN you can omit the LTACC operand.
- In the assembler macros CREFTPRF and MODFTPRF the new value *NONE is provided in the UACC and PACC operands.

Displaying canceled requests

When FT requests are displayed, you can select canceled requests which have not yet been completely terminated. This results in the following changes:

- The assembler macro NSTAT has been extended by the operand value STATE=CAN.
- In the COBOL call NSTATUS the STATE structure has been extended by the CANCELLED field.

1.6 README file

The functional changes to the current product version and revisions to this manual are described in the product-specific Readme file.

Readme files online

Readme files are available to you online in addition to the product manuals under the various products at <http://manuals.ts.fujitsu.com>.

Readme files under BS2000/OSD

On your BS2000 system you will find Readme files for the installed products under the file name:

```
SYSRME.OPENFT.120.E  
SYSRME.OPENFT-FTAM.120.E  
SYSRME.OPENFT-FTP.120.E  
SYSRME.OPENFT-AC.120.E
```

Please refer to your system administrator for the user ID under which the required Readme file can be found. You can also obtain the path name of the Readme file directly by entering the following command:

```
/SHOW-INSTALLATION-PATH INSTALLATION-UNIT=<product>, LOGICAL-ID=SYSRME.E
```

You can view the Readme file on screen with `/SHOW-FILE` or by opening it in an editor, or print it on a standard printer using the following command:

```
/PRINT-DOCUMENT <filename>, LINE-SPACING=*BY-EBCDIC-CONTROL
```

Additional product information

Current information, version and hardware dependencies, and instructions for installing and using a product version are contained in the associated Release Notice. These Release Notices are available at <http://manuals.ts.fujitsu.com>.

2 ASSEMBLER programming interface

An FT request can also be issued from an application program. To do this, openFT for BS2000 has an ASSEMBLER programming interface.

For file transfer with FTAM partners, please note the restrictions described in the user manual.

Macro with EXECUTE format

Name	Operation	Operand
[name]	<macro>	MF=($\left\{ \begin{array}{l} E, adr \\ (r) \end{array} \right\}$)

The symbolic address *adr* or the register *r* refers to the operand list.

The user creates such a list in one of two ways.

Macro with DSECT format

This gives you the layout of the operand list.

Name	Operation	Operand
[name]	<macro>	MF=(D[,xxx])

The prefix *xxx* may contain up to 3 characters and must generate names which are permissible in ASSEMBLER.

To fill the HEADER, the user must generate an initialized header pattern using the following call:

Name	Operation	Operand
[name]	<macro>	HEADER

This macro call generates a default header for the macro parameter field. This default header is to be copied into the corresponding field (e.g. xxxHADS for SHWFTADS) of the operand list using an MVC command (length 4).

The DSECT length fields must contain the exact length of the operand fields (value fields) in the program. "Fillers" such as blanks are considered to be part of the operand and will be evaluated, which will lead to syntax errors. If an operand is not evaluated, the value of the address field and the length field will be set to binary zero.

Macro with LIST format

Name	Operation	Operand
[name]	<macro>	MF=L,Operand

The operand values are addressed using the symbolic addresses *adr*. ASSEMBLER's implicit lengths of these fields must correspond exactly to the length of the value field (no fillers!). If, for example the operand PARTNER has the VALUE C'ABAB', it cannot be entered in the form *adr DC 2C'AB'*, since in this case the implicit length of the fields equals 2 instead of 4.

For this format, the user need not worry about the HEADER, since it is automatically generated internally.

Macro with default format

Name	Operation	Operand
[name]	<macro>	Operand

If the MF parameter is missing for the macro for NCAN, NCOPY, NCOPYSYN and NSTAT, an operand list like the one for MF=L is generated. For all others, the address is also loaded in register 1 and the macro is executed as in MF=E. Please note that macros with default format are not reenterable.

Information on value assignments

If FTAC functionality is used, the user can define a transfer admission for FT requests which must be entered to work with an admission profile.

Please note that any transfer admission which is entered alphanumerically at the command interface must be entered in lower case letters at the programming interface.

If parameters are not explicitly specified, openFT will assign the same default values as for the commands.

Exception



For fields which identify an FTAC security level, the value 0 stands for the security level 0. In other words, it blocks that function! (fields: xxxMOSND, xxxMORCV, xxxMISND, xxxMIRCV, xxxMIPRC for the call MODFTADS and xxxMPLV for the calls CREFTPRF and MODFTPRF).

The value fields must not contain any fillers, i.e. the length fields must indicate the exact length of the value fields.

Enter keywords into the DSECT fields with the help of the above-mentioned equates.

Programs can run in 24 or 31 bit mode. Please note that addresses in 24 bit mode must have an initial 0.

2.1 Macros for openFT for BS2000

The following openFT functions can be used with macros:

Macro	Default prefix	Function	Command
NCAN	YNA	cancel file transfer	CANCEL-FILE-TRANSFER
NCOPY	YNO	transfer file	TRANSFER-FILE
NCOPYSYN	YNY	transfer file synchronous	TRANSFER-FILE-SYNCHRONOUS
NDEL	YNE	delete remote file	DELETE-REMOTE-FILE
NLMOD	YNM	modify local FT file attributes	MODIFY-FILE-FT-ATTRIBUTES
NLSHOW	YNH	show local FT file attributes	SHOW-FILE-FT-ATTRIBUTES
NMOD	YNU	modify remote file attributes	MODIFY-REMOTE-FILE-ATTRIBUTES
NSHOW	YNW	show remote file attributes	SHOW-REMOTE-FILE-ATTRIBUTES
NSTAT	YNS	show file transfer status	SHOW-FILE-TRANSFER

In addition, there is also the CMD macro (for more information see the manual “BS2000/OSD-BC - Executive Macros”).

An overview of the openFT-AC macros can be found on [page 90](#).

Programs that were created for FT-BS2000 (version 2.0 and later) can also be run in openFT V12.0. Of course, if newer parameters are used during macro calls, the macro YNDEQU is to be incorporated into the program. You must remove any value assignments (EQU) from the macro that have not been defined in the macro itself. The macros are located in the file SYSLIB.OPENFT.120.

2.1.1 NCAN - Cancel file transfer request

The macro NCAN can be used to reverse an FT request or cancel the file transfer. openFT deletes the FT request which meets the specified selection criteria from the request file and cancels the corresponding file transfer.. The functionality corresponds to the command described in the user manual under CANCEL-FILE-TRANSFER (NCANCEL).

Format of the macro NCAN (LIST format/default format)

Name	Operation	Operand
[name]	NCAN	[MF=L,] [TRANSID=adr] [,SOWNER=adr] [,PARTNER=adr] [{ ,LFILE=adr { ,LLIB=adr[,LEL=adr][,LELVER=adr][,LELTYP=adr] }] [,JVNAME=adr] [,JVPASS=adr] [,INIT={ BOTH LOCAL REMOTE }] [,FORCE={ NO YES }] [,Version={ 1 2 }]

Format of the macro NCAN (DSECT format)

Name	Operation	Operand
[name]	NCAN	MF=(D[,xxx])

The prefix xxx (default value YNA) may be up to 3 characters long. It must generate names which are permitted in ASSEMBLER. The following ASSEMBLER instructions are generated:

```

xxxNCAN  DSECT
xxxHCAN  FHDR  MF=(C,&P)
xxxTRID  DS    A                A(TRANSFER-ID)
xxxTRIDL DS    XL2             L(TRANSFER-ID)
xxxLNCA1 EQU  *-xxxNCAN       L(MACRO VERSION 1)
*
xxxSOWNL DS    XL2             L(OWNER-ID)
xxxSOWN  DS    AL4             A(OWNER-ID)
xxxPART  DS    AL4             A(PARTNER-SYSTEM)
xxxPARTL DS    XL2             L(PARTNER-SYSTEM)
xxxLFILL DS    XL2             L(FILE-NAME)
xxxLFIL  DS    AL4             A(FILE-NAME)
xxxLLIB  DS    AL4             A(LIBRARY-NAME)
xxxLLIBL DS    XL2             L(LIBRARY-NAME)
xxxLELL  DS    XL2             L(ELEMENT-NAME)
xxxLEL   DS    AL4             A(ELEMENT-NAME)
xxxLEV   DS    AL4             A(ELEMENT-VERSION)
xxxLEVL  DS    XL2             L(ELEMENT-VERSION)
xxxLETYL DS    XL2             L(ELEMENT-TYPE)
xxxLETY  DS    AL4             A(ELEMENT-TYPE)
xxxJVNM  DS    AL4             A(JV-NAME)
xxxJVNML DS    XL2             L(JV-NAME)
xxxJVPSL DS    XL2             L(JV-PASSWORD)
xxxJVPS  DS    AL4             A(JV-PASSWORD)
xxxINIT  DS    XL1             INITIATOR
*
xxxIBOTH EQU    0              INITIATOR = BOTH
xxxILOC  EQU    192            INITIATOR = LOCAL
xxxIREM  EQU    128            INITIATOR = REMOTE
*
&P.LNCA2 EQU  *-&.NCAN       L(MACRO VERSION 2)
*
&P.FORCE DS    XL1             FORCE-CANCELLATION
xxxFNO   EQU    X'00'          FORCE = NO
xxxFYES  EQU    X'01'          FORCE = YES
*
xxxLNCA  EQU    *-xxxNCAN

```


Meaning of the fields

The macro NCAN offers you the same options as the command CANCEL-FILE-TRANSFER (NCANCEL). Please refer to the command description. The following table shows how the parameters in the macro NCAN correspond to the command operands:

Operand (L format)	Address fields (D format)	Meaning in command
TRANSID	TRID	TRANSFER-ID
SOWNER	SOWN	OWNER-IDENTIFICATION
PARTNER	PART	PARTNER-NAME
LFILE	LFIL	FILE-NAME
LLIB	LLIB	LIBRARY
LEL	LEL	ELEMENT
LELVER	LEV	VERSION
LELTYP	LETY	TYPE
JVNAME	JVNM	MONJV
JVPASS	JVPS	JV-PASSWORD
INIT	INIT	INITIATOR
FORCE	FORCE	FORCE-CANCELLATION
VERSION	FCTV	————— (Type of return code output (old/new)) ¹

¹ Depending on the value of the VERSION operand, the FUNCTION INTERFACE VERSION NUMBER is set in the header FHDR

If you do not enter anything in the fields for the OWNER-ID, your own user ID will be used as the value (this corresponds to the default entry *OWN at the command interface. If you intentionally fill the OWNER-ID field with blanks, then all user IDs will be used (this corresponds to the entry *ALL at the command interface).

You may only set FORCE=YES if you have explicitly specified a transfer ID in the TRID field. This request must already have been cancelled with FORCE=NO. Only login names that possess the FDT-ADM privilege may specify FORCE=YES.

2.1.2 NCOPY - Transfer file

The macro NCOPY can be used to transfer a file or library member. The computer in which the program is running can be either sender or receiver of the file. The functionality corresponds to the command TRANSFER-FILE (NCOPY)

Format of the macro NCOPY (LIST format/default format)

Name	Operation	Operand
[name]	NCOPY	<pre> [MF=L,] TRANS={ FROM TO } ,PARTNER=adr [{ ,LFILE=adr [[,LLIB=adr][,LEL=adr][,LELVER=adr][,LLIBTYP=adr]] }] [,LFPASS=adr] [,LTUSER=adr[,LTACC=adr][,LTPASS=adr]] [,LTAD=adr] [,LCCNS=adr] [,LPUSER=adr[,LPACC=adr][,LPPASS=adr]] [,LSUCC=adr] [,LFAIL=adr] [,LIST={ <u>LIST</u> NONE LISTFILE LISTFAIL LISTFFAI }] [,JVNAME=adr] [,JVPASS=adr] [,RSYNTAX={ <u>BS2</u> MSP ANY }] [{ ,RFILE=adr [[,RLIB=adr][,REL=adr][,RELVER=adr][,RLIBTYP=adr]] }] </pre>

Name	Operation	Operand
		[,RFPASS=adr]
		[,RTUSER=adr]
		[,RTACC=adr]
		[,RTPASS=adr]
		[,RTAD=adr]
		[,RPUSER=adr[,RPACC=adr][,RPPASS=adr]]
		[,RSUCC=adr]
		[,RFAIL=adr]
		[,RCCNS=adr]
		[,FAVAIL={ [*] <u>NONE</u> [*]IMMED [*]DEFER }]
		[,STORACC={ <u>*NONE</u> adr }]
		[,ACCMOD={ [*] <u>NONE</u> [*]PAR }]
		[,READF={ [*] <u>NO</u> [*]YES }]
		[,INSDU={ [*] <u>NO</u> [*]YES }]
		[,REPLF={ [*] <u>NO</u> [*]YES }]
		[,EXTDF={ [*] <u>NO</u> [*]YES }]
		[,ERADU={ [*] <u>NO</u> [*]YES }]
		[,READA={ [*] <u>NO</u> [*]YES }]

Name	Operation	Operand
		[,CHNGA={ [<u>*</u>]NO } [<u>*</u>]YES }]
		[,DELF={ [<u>*</u>]NO } [<u>*</u>]YES }]
		[,LQUAL={ <u>*UNCHG</u> adr }]
		[,RCPASS=adr]
		[,COMP={ NONE } BYTE }] ZIP }
		[,WRITE={ <u>REPLACE</u> NEW }] EXT }
		[,DATA={ <u>CHAR</u> BIN }] NOTSPEC USER }
		[,TRANSP={ <u>NO</u> }] ¹ STD }
		[,PRIO={ <u>NORM</u> HIGH }] LOW }
		[,START=adr]
		[,CANCEL=adr]
		[,TRANSID=adr]
		[,DATENCR={ <u>NO</u> YES }] ONLYDI }
		[,RSIZE={ <u>*NOTSPEC</u> <integer 1..32756> }]

Name	Operation	Operand
		$\left[, RFORM = \left\{ \begin{array}{l} \text{STD} \\ \text{VAR} \\ \text{FIX} \\ \text{UNDEF} \end{array} \right\} \right]$ $\left[, Version = \left\{ \begin{array}{l} 1 \\ 2 \end{array} \right\} \right]$

1 The value TRANSP=YES is mapped onto the programming interfaces using TRANSP=STD.

Format of the macro NCOPY (DSECT format)

Name	Operation	Operand
[name]	NCOPY	MF=(D[,xxx])

The prefix xxx (default value YNO) may be up to 3 characters long. It must generate names which are permitted in ASSEMBLER. The following ASSEMBLER instructions are generated:

```

xxxNCOPY DSECT
xxxHCOP FHDR MF=(C,&P)
xxxDMSCO DS A DMS-FEHLERCODE
xxxPART DS A A(PARTNER-NAME)
xxxPARTL DS XL2 L(PARTNER-NAME)
xxxLFILL DS XL2 L(FILE-NAME)
xxxLFIL DS AL4 A(FILE-NAME)
          ORG xxxLFILL
xxxLLIBL DS XL2 L(LIBRARY-NAME)
xxxLLIB DS AL4 A(LIBRARY-NAME)
xxxLEL DS AL4 A(ELEMENT-NAME)
xxxLELL DS XL2 L(ELEMENT-NAME)
xxxLEVL DS XL2 L(ELEMENT-VERSION)
xxxLEV DS AL4 A(ELEMENT-VERSION)
xxxLETY DS AL4 A(ELEMENT-TYPE)
xxxLETYL DS XL2 L(ELEMENT-TYPE)
xxxLFPAL DS XL2 L(FILE-PASSWORD)
xxxLFPA DS AL4 A(FILE-PASSWORD)
xxxLTUS DS AL4 A(USER-ID/TRANS)
xxxLTUSL DS XL2 L(USER-ID/TRANS)
xxxLTACL DS XL2 L(USER-ACC/TRANS)
xxxLTAC DS AL4 A(USER-ACC/TRANS)
xxxLTPA DS AL4 A(USER-PASSWORD/TRANS)

```

xxxLTPAL	DS	XL2	L(USER-PASSWORT/TRANS)
xxxLPUSL	DS	XL2	L(USER-ID/PROC)
xxxLPUS	DS	AL4	A(USER-ID/PROC)
xxxLPAC	DS	AL4	A(USER-ACC/PROC)
xxxLPACL	DS	XL2	L(USER-ACC/PROC)
xxxLPPAL	DS	XL2	L(USER-PASSWORT/PROC)
xxxLPPA	DS	AL4	A(USER-PASSWORT/PROC)
xxxLSUC	DS	AL4	A(SUCCESS-PROCESSING)
xxxLSUCL	DS	XL2	L(SUCCESS-PROCESSING)
xxxLFAIL	DS	XL2	L(FAILURE-PROCESSING)
xxxLFAI	DS	AL4	A(FAILURE-PROCESSING)
*			
xxxRFIL	DS	AL4	A(FILE-NAME)
xxxRFILL	DS	XL2	L(FILE-NAME)
xxxRLIBL	DS	XL2	L(LIBRARY-NAME)
xxxRLIB	DS	AL4	A(LIBRARY-NAME)
xxxREL	DS	AL4	A(ELEMENT-NAME)
xxxRELL	DS	XL2	L(ELEMENT-NAME)
xxxREVL	DS	XL2	L(ELEMENT-VERSION)
xxxREV	DS	AL4	A(ELEMENT-VERSION)
xxxRETY	DS	AL4	A(ELEMENT-TYPE)
xxxRETYL	DS	XL2	L(ELEMENT-TYPE)
xxxRFPAL	DS	XL2	L(FILE-PASSWORT)
xxxRFPA	DS	AL4	A(FILE-PASSWORT)
xxxRTUS	DS	AL4	A(USER-ID/TRANS)
xxxRTUSL	DS	XL2	L(USER-ID/TRANS)
xxxRTACL	DS	XL2	L(USER-ACC/TRANS)
xxxRTAC	DS	AL4	A(USER-ACC/TRANS)
xxxRTPA	DS	AL4	A(USER-PASSWORT/TRANS)
xxxRTPAL	DS	XL2	L(USER-PASSWORT/TRANS)
xxxRPUSL	DS	XL2	L(USER-ID/PROC)
xxxRPUS	DS	AL4	A(USER-ID/PROC)
xxxRPAC	DS	AL4	A(USER-ACC/PROC)
xxxRPACL	DS	XL2	L(USER-ACC/PROC)
xxxRPPAL	DS	XL2	L(USER-PASSWORT/PROC)
xxxRPPA	DS	AL4	A(USER-PASSWORT/PROC)
xxxRSUC	DS	AL4	A(SUCCESS-PROCESSING)
xxxRSUCL	DS	XL2	L(SUCCESS-PROCESSING)
xxxRFAIL	DS	XL2	L(FAILURE-PROCESSING)
xxxRFAI	DS	AL4	A(FAILURE-PROCESSING)
*			
xxxSTAR	DS	AL4	A(START)
xxxSTARL	DS	XL2	L(START)
xxxCANL	DS	XL2	L(CANCEL)
xxxCAN	DS	AL4	A(CANCEL)
xxxTRID	DS	AL4	A(TRANS-ID)
xxxTRIDL	DS	XL2	L(TRANS-ID)

xxxTRDIR	DS	XL1	TRANSFER DIRECTION
xxxCOMP	DS	XL1	COMPRESSION
xxxWRMOD	DS	XL1	WRITE-MODE
xxxDATYP	DS	XL1	DATA-TYPE
xxxPRIO	DS	XL1	PRIORITY
xxxLIST	DS	XL1	LISTING
xxxRSYN	DS	XL1	REMOTE SYNTAX
	DS	0H	
xxxLJVNL	DS	AL2	L(JV-NAME)
xxxLJVN	DS	AL4	A(JV-NAME)
xxxLJVP	DS	AL4	A(JV-PASSW)
xxxLJVPL	DS	AL2	L(JV-PASSW)
xxxLTADL	DS	AL2	L(LOCAL TRANSFER-ADMISSION)
xxxLTAD	DS	AL4	A(LOCAL TRANSFER-ADMISSION)
xxxRTAD	DS	AL4	A(REMOTE TRANSFER-ADMISSION)
xxxRTADL	DS	AL2	L(REMOTE TRANSFER-ADMISSION)
*			
xxxSTACL	DS	XL2	L(STORAGE-ACCOUNT)
xxxSTAC	DS	AL4	A(STORAGE-ACCOUNT)
xxxLQF	DS	AL4	A(LEGAL-QUALIFICATION)
xxxLQFL	DS	XL2	L(LEGAL-QUALIFICATION)
*			
xxxAVAIL	DS	XL1	FILE-AVAIL.: NONE / IMMED / DEFER
xxxISTAC	DS	XL1	STORAGE-ACCOUNT: NONE
xxxILQF	DS	XL1	LEGAL-QUALIFICATION: UNCHG
*			
xxxACCES	DS	XL1	ACCESS-MODE : NONE / PAR
xxxREADF	DS	XL1	READ-FILE: NO / YES
xxxREPLF	DS	XL1	REPLACE-FILE: NO / YES
xxxEXTDF	DS	XL1	EXTEND-FILE: NO / YES
xxxREADA	DS	XL1	READ-ATTRIBUTES: NO / YES
xxxCHNGA	DS	XL1	CHANGE-ATTRIBUTES: NO / YES
xxxDELF	DS	XL1	DELETE-FILE: NO / YES
xxxINSDU	DS	XL1	INSERT-DATA-UNIT: NO / YES
xxxERADU	DS	XL1	ERASE-DATA-UNIT: NO / YES
*			
	DS	0H	
xxxRCPAL	DS	XL2	L(CREATE-PASSWORD)
xxxRCPA	DS	AL4	A(CREATE-PASSWORD)
*			
xxxRSIZE	DS	XL4	RECORD-SIZE: *NOT-SPEC / 1..32756
*			
xxxRID	DS	XL4	REQUEST ID
xxxFUD	DS	AL4	A(FURTHER DETAILS)
xxxLFUD	DS	XL2	L(FURTHER DETAILS)
xxxASYMG	DS	XL1	ASYNCHRONOUS END MESSAGE
xxxSCSET	DS	XL1	CHARACTER SET

```

xxxRFORM DS    XL1          RECORD FORMAT = STD/VAR/FIX/UNDEF
*
xxxTABEX DS    XL1          TABULATOR EXPANSION
xxxLCCSL DS    XL2          L(LOCAL-CODE-CHARACTER-SET)
xxxLCCS  DS    AL4          A(LOCAL-CODE-CHARACTER-SET)
xxxRCCS  DS    AL4          A(REMOTE-CODE-CHARACTER-SET)
xxxRCCSL DS    XL2          L(REMOTE-CODE-CHARACTER-SET)
*
xxxLNCO  EQU    *-xxxNCOPY
*
xxxTRDTO EQU    X'01'      TRANSFER DIRECTION = TO
xxxTRDFR EQU    X'00'      TRANSFER DIRECTION = FROM
xxxCOMBY EQU    X'80'      COMPRESSION = BYTE
xxxCOMZI EQU    X'40'      COMPRESSION = ZIP
xxxCOMNO EQU    X'00'      COMPRESSION = NONE
xxxLISY  EQU    X'80'      LISTING      = SYSLST
xxxLISF  EQU    X'40'      LISTING      = LISTFILE
xxxLISYF EQU    X'20'      LISTING      = SYSLST
*
xxxLISFF EQU    X'10'      LISTING      = LISTFILE
*
xxxLISN  EQU    X'00'      LISTING      = NONE
xxxMSP   EQU    X'03'      REMOTESYNTAX= MSP
xxxBS2   EQU    X'01'      REMOTESYNTAX= BS2
xxxANY   EQU    X'02'      REMOTESYNTAX= ANY
xxxNEW   EQU    X'01'      WRITE-MODE = NEW FILE
xxxEXT   EQU    X'04'      WRITE-MODE = EXTEND
xxxRPL   EQU    X'02'      WRITE-MODE = REPLACE
xxxCHRS  EQU    X'88'      DATA TYPE  = CHARACTER(TRANS=STD)
xxxBINS  EQU    X'84'      DATA TYPE  = BINARY(TRANS=STD)
xxxCHR   EQU    X'08'      DATA TYPE  = CHARACTER
xxxBIN   EQU    X'04'      DATA-TYPE  = BINARY
xxxNOTSP EQU    X'02'      DATA-TYPE  = NOT-SPECIFIED
xxxUSER  EQU    X'01'      DATA-TYPE  = USER
xxxNORM  EQU    X'00'      PRIORITY   = NORMAL
xxxHIGH  EQU    X'01'      PRIORITY   = HIGH
xxxLOW   EQU    X'02'      PRIORITY   = LOW
xxxSTDLC EQU    X'0'       LELVER    = STD BEI BS2/LOCAL
xxxSAME  EQU    X'0'       RELVER    = SAME BEI BS2/REMOTE
xxxSTDRE EQU    X'FF'     RELVER    = STD BEI BS2/REMOTE
xxxNONE  EQU    X'0'       RELVER    = NONE BEI *ANY
xxxNSPEC EQU    X'FF'     RFIL/LPUS/RPUS = NOT-SPECIFIED
xxxCNES  EQU    X'01'     COMPRESSION = NONE/ENCRYPTION = YES
xxxCZES  EQU    X'41'     COMPRESSION = ZIP/ENCRYPTION = YES
xxxCBES  EQU    X'81'     COMPRESSION = BYTE/ENCRYPTION = YES
xxxCNOD  EQU    X'02'     COMPRESSION = NONE/ENCRYPTION = ODI
xxxCZOD  EQU    X'42'     COMPRESSION = ZIP/ENCRYPTION = ODI
xxxCBOD  EQU    X'82'     COMPRESSION = BYTE/ENCRYPTION = ODI

```


xxxRFSTD EQU	X'00'	RFORM	= STD
xxxRFVAR EQU	X'1A'	RFORM	= VAR
xxxRFFIX EQU	X'1B'	RFORM	= FIX
xxxRFUND EQU	X'1C'	RFORM	= UNDEF

If an operand is not to be assigned, the value of the address field and the length field must be set to binary zero.

If the value *NOT-SPECIFIED is entered for the parameter FILE-NAME of the remote system, then the value set in the equate xxxNSPEC is to be written in the value field for the filename (length 1).

If this value is entered for the PROCESSING-ADMISSION of the local or remote system, then it is to be written in the value field for the corresponding USER-ID.

Meaning of the fields

The macro NCOPY offers you the same options as the command TRANSFER-FILE (NCOPY). Please refer to the command description. The following table shows how the parameters in the macro NCOPY correspond to the command operands.

Operand (L format)	Address field (D format)	Meaning in command
TRANS	TRDIR	TRANSFER-DIRECTION
PARTNER	PART	PARTNER-NAME
LFILE	LFIL	FILE-NAME in LOCAL parameter.
LLIB	LLIB	LIBRARY in LOCAL parameter
LEL	LEL	ELEMENT in LOCAL parameter
LELVER	LEV	VERSION in LOCAL parameter
LLIBTYP	LETY	TYPE in LOCAL parameter
LFPASS	LFPA	FILE-PASSWORD in LOCAL parameter
LTUSER	LTUS	USER-IDENTIFICATION in local TRANSFER-ADMISSION
LTACC	LTAC	ACCOUNT in local TRANSFER-ADMISSION ¹
LTPASS	LTPA	PASSWORD in local TRANSFER-ADMISSION
LTAD	LTAD	TRANSFER-ADMISSION (local)
LCCNS	LCCS	CODED-CHARACTER-SET in LOCAL-PARAMETER
LPUSER	LPUS	USER-IDENTIFICATION in local PROCESSING-ADMISSION
LPACC	LPAC	ACCOUNT in local PROCESSING-ADMISSION ¹
LPPASS	LPPA	PASSWORD in local PROCESSING-ADMISSION
LSUCC	LSUC	SUCCESS-PROCESSING in LOCAL parameter

Operand (L format)	Address field (D format)	Meaning in command
LFAIL	LFAI	FAILURE-PROCESSING in LOCAL parameter
LIST	LIST	LISTING in LOCAL parameter
JVNAME	LJVN	MONJV in LOCAL parameter
JVPASS	LJVP	JV-PASSWORD in LOCAL parameter
RSYNTAX	RSYN	REMOTE parameter
RFILE	RFIL	FILE-NAME in REMOTE parameter
RLIB	RLIB	LIBRARY in REMOTE parameter
REL	REL	ELEMENT in REMOTE parameter
RELVER	REV	VERSION in REMOTE parameter
RLIBTYP	RETY	TYPE in REMOTE parameter
RFPASS	RFPA	FILE-PASSWORD in REMOTE parameter
RTUSER	RTUS	USER-IDENTIFICATION in remote TRANSFER-ADMISSION
RTACC	RTAC	ACCOUNT in remote TRANSFER-ADMISSION
RTPASS	RTPA	PASSWORD in remote TRANSFER-ADMISSION
RTAD	RTAD	TRANSFER-ADMISSION (remote)
RPUSER	RPUS	USER-IDENTIFICATION in remote PROCESSING-ADMISSION
RPACC	RPAC	ACCOUNT in remote PROCESSING-ADMISSION
RPPASS	RPPA	PASSWORD in remote PROCESSING-ADMISSION
RSUCC	RSUC	SUCCESS-PROCESSING in REMOTE parameter
RFAIL	RFAI	FAILURE-PROCESSING in REMOTE parameter
RCCNS	RCCS	CODED-CHARACTER-SET in REMOTE-PARAMETER
FAVAIL	AVAIL	FILE-AVAILABILITY
STORACC	STAC	STORAGE-ACCOUNT
ACCMOD	ACCES	ACCESS-MODE
READF	READF	READ-FILE in the structure ACCESS-MODE
INSDU	INSDU	INSERT-DATA-UNIT in the structure ACCESS-MODE
REPLF	REPLF	REPLACE-FILE in the structure ACCESS-MODE
EXTDF	EXTDF	EXTEND-FILE in the structure ACCESS-MODE
ERADU	ERADU	ERASE-DATA-UNIT in the structure ACCESS-MODE
READA	READA	READ-ATTRIBUTES in the structure ACCESS-MODE

Operand (L format)	Address field (D format)	Meaning in command
CHNGA	CHNGA	CHANGE-ATTRIBUTES in the structure ACCESS-MODE
DELF	DELF	DELETE-FILE in the structure ACCESS-MODE
LQUAL	LQF	LEGAL-QUALIFICATION
RCPASS	RCPA	CREATE-PASSWORD in REMOTE parameter
COMP	COMP	COMPRESS
WRITE	WRMOD	WRITE-MODE
DATA	DATYP	DATA-TYPE
TRANSP	DATYP	TRANSPARENT
PRIO	PRIO	PRIORITY
START	STAR	START
CANCEL	CAN	CANCEL
DATENCR	COMP	DATA-ENCRYPTION
TRANSID	TRID	_____
RSIZE	RSIZE	RECORD-SIZE
RFORM	RFORM	RECORD-FORMAT
VERSION	FCTV	_____ (TYpe of return code output (old/new)) ²

¹ If LTACC or LPACC is not specified, this corresponds to an account number not specified in the command, i.e. the user's default account number is used.

² Depending on the value of the VERSION operand, the FUNCTION INTERFACE VERSION NUMBER is set in the header FHDR.

In the field for TRANS-ID or TRANSID, the TRANSFER-ID is entered after a successful NCOPY macro (printable in decimal notation, left justified, filled with blanks on the right). The field must be ≥ 10 bytes or 0, if the TRANSFER-ID is not to be entered.

After a defective NCOPY macro call, additional information (precise cause of the error) concerning the error message itself can be returned in the FUD (Further Details) field. The field must be ≤ 64 bytes or zero if the additional information is not to be stored in the FUD field.

If you run a file transfer with an FTAM or FTP partner from an ASSEMBLER program, the same function restrictions apply as at the command interface. You cannot transfer any library members which are in the remote system, specify follow-up processing for the remote system, or transfer files in compressed form. For more information, see the description of the command TRANSFER-FILE(NCOPY) in the user manual.

Format rules for START and CANCEL

The length field is at least 15 bytes long.

Contents:

y	y	-	m	m	-	d	d			h	h	:	m	m
y	y	-	m	m	-	d	d							
T	O	D	A	Y						h	h	:	m	m
T	O	D	A	Y										
T	O	M	O	R	R	O	W			h	h	:	m	m
T	O	M	O	R	R	O	W							
										h	h	:	m	m

If the first 8 bytes (1 to 8) are blank, DATE= TODAY is taken as the default value.

If the last 5 bytes (11 to 15) are blank, START-TIME=00:00 and CANCEL-TIME=23:59 will be taken as the default values.

Note:

In contrast to the command interface, the specification for year can only be two digits.

2.1.3 NCOPYSYN - Transfer file

The macro NCOPYSYN can be used to synchronously transfer a file or library member. The computer in which the program is running can be either sender or receiver of the file. The functionality corresponds to the command TRANSFER-FILE-SYNCHRONOUS (FTSNCOPY). NCOPYSYN differs from the NCOPY macro in the following points:

- no local follow-up processing
- no result lists
- no job variables
- no priorities and no specification of start and cancellation times.

Format of the macro NCOPYSYN (LIST format/default format)

Name	Operation	Operand
[name]	NCOPY	<pre>[MF=L,] TRANS={ FROM TO } ,PARTNER=adr [{ ,LFILE=adr [[,LLIB=adr][,LEL=adr][,LELVER=adr][,LLIBTYP=adr]] }] [,LFPASS=adr] [,LTUSER=adr[,LTACC=adr],LTPASS=adr] [,LTAD=adr] [,LCCNS=adr] [,RSYNTAX={ BS2 MSP ANY }] [{ ,RFILE=adr [[,RLIB=adr][,REL=adr][,RELVER=adr][,RLIBTYP=adr]] }] [,RFPASS=adr] [,RTUSER=adr] [,RTACC=adr] [,RTPASS=adr] [,RTAD=adr]</pre>

Name	Operation	Operand
		[,RPUSER=adr[,RPACC=adr][,RPPASS=adr]]
		[,RSUCC=adr]
		[,RFAIL=adr]
		[,RCCNS=adr]
		[,FAVAIL={ [*]NONE [*]IMMED [*]DEFER }]
		[,STORACC={ *NONE adr }]
		[,ACCMOD={ [*]NONE [*]PAR }]
		[,READF={ [*]NO [*]YES }]
		[,INSDU={ [*]NO [*]YES }]
		[,REPLF={ [*]NO [*]YES }]
		[,EXTDF={ [*]NO [*]YES }]
		[,ERADU={ [*]NO [*]YES }]
		[,READA={ [*]NO [*]YES }]
		[,CHNGA={ [*]NO [*]YES }]
		[,DELF={ [*]NO [*]YES }]
		[,LOUAL={ *UNCHG adr }]

Name	Operation	Operand
		[,RCPASS=adr]
		[,COMP={ NONE BYTE ZIP }]
		[,WRITE={ REPLACE NEW EXT }]
		[,DATA={ CHAR BIN NOTSPEC USER }]
		[,TRANSP={ NO STD }] ¹
		[,TRANSID=adr]
		[,DATENCR={ NO YES ONLYDI }]
		[,RSIZE={ *NOTSPEC <integer 1..32756> }]
		[,RFORM={ STD VAR FIX UNDEF }]

¹ The value TRANSP=YES is mapped onto the programming interfaces using TRANSP=STD.

Format of the macro NCOPYSYN (DSECT format)

Name	Operation	Operand
[name]	NCOPYSYN	MF=(D[,xxx])

The prefix xxx (default value YNY) may be up to 3 characters long. It must generate names which are permitted in ASSEMBLER. The following ASSEMBLER instructions are generated:

```
xxxNCOPY DSECT
xxxHCOP FHDR MF=(C,&P)
```

xxxDMSCO	DS	A	DMS-FEHLERCODE
xxxPART	DS	A	A(PARTNER-NAME)
xxxPARTL	DS	XL2	L(PARTNER-NAME)
xxxLFILL	DS	XL2	L(FILE-NAME)
xxxLFIL	DS	AL4	A(FILE-NAME)
	ORG	xxxLFILL	
xxxLLIBL	DS	XL2	L(LIBRARY-NAME)
xxxLLIB	DS	AL4	A(LIBRARY-NAME)
xxxLEL	DS	AL4	A(ELEMENT-NAME)
xxxLELL	DS	XL2	L(ELEMENT-NAME)
xxxLEVL	DS	XL2	L(ELEMENT-VERSION)
xxxLEV	DS	AL4	A(ELEMENT-VERSION)
xxxLETY	DS	AL4	A(ELEMENT-TYPE)
xxxLETYL	DS	XL2	L(ELEMENT-TYPE)
xxxLFPAL	DS	XL2	L(FILE-PASSWORT)
xxxLFPA	DS	AL4	A(FILE-PASSWORT)
xxxLTUS	DS	AL4	A(USER-ID/TRANS)
xxxLTUSL	DS	XL2	L(USER-ID/TRANS)
xxxLTACL	DS	XL2	L(USER-ACC/TRANS)
xxxLTAC	DS	AL4	A(USER-ACC/TRANS)
xxxLTPA	DS	AL4	A(USER-PASSWORT/TRANS)
xxxLTPAL	DS	XL2	L(USER-PASSWORT/TRANS)
*			
xxxRFIL	DS	AL4	A(FILE-NAME)
xxxRFILL	DS	XL2	L(FILE-NAME)
xxxRLIBL	DS	XL2	L(LIBRARY-NAME)
xxxRLIB	DS	AL4	A(LIBRARY-NAME)
xxxREL	DS	AL4	A(ELEMENT-NAME)
xxxRELL	DS	XL2	L(ELEMENT-NAME)
xxxREVL	DS	XL2	L(ELEMENT-VERSION)
xxxREV	DS	AL4	A(ELEMENT-VERSION)
xxxRETY	DS	AL4	A(ELEMENT-TYPE)
xxxRETYL	DS	XL2	L(ELEMENT-TYPE)
xxxRFPAL	DS	XL2	L(FILE-PASSWORT)
xxxRFPA	DS	AL4	A(FILE-PASSWORT)
xxxRTUS	DS	AL4	A(USER-ID/TRANS)
xxxRTUSL	DS	XL2	L(USER-ID/TRANS)
xxxRTACL	DS	XL2	L(USER-ACC/TRANS)
xxxRTAC	DS	AL4	A(USER-ACC/TRANS)
xxxRTPA	DS	AL4	A(USER-PASSWORT/TRANS)
xxxRTPAL	DS	XL2	L(USER-PASSWORT/TRANS)
xxxRPUSL	DS	XL2	L(USER-ID/PROC)
xxxRPUS	DS	AL4	A(USER-ID/PROC)
xxxRPAC	DS	AL4	A(USER-ACC/PROC)
xxxRPACL	DS	XL2	L(USER-ACC/PROC)
xxxRPPAL	DS	XL2	L(USER-PASSWORT/PROC)
xxxRPPA	DS	AL4	A(USER-PASSWORT/PROC)

xxxRSUC	DS	AL4	A(SUCCESS-PROCESSING)
xxxRSUCL	DS	XL2	L(SUCCESS-PROCESSING)
xxxRFAIL	DS	XL2	L(FAILURE-PROCESSING)
xxxRFAI	DS	AL4	A(FAILURE-PROCESSING)
*			
xxxTRID	DS	AL4	A(TRANS-ID)
xxxTRIDL	DS	XL2	L(TRANS-ID)
xxxTRDIR	DS	XL1	TRANSFER DIRECTION
xxxCOMP	DS	XL1	COMPRESSION
xxxWRMOD	DS	XL1	WRITE-MODE
xxxDATYP	DS	XL1	DATA-TYPE
xxxRSYN	DS	XL1	REMOTE SYNTAX
	DS	OH	
xxxLTADL	DS	AL2	L(LOCAL TRANSFER-ADMISSION)
xxxLTAD	DS	AL4	A(LOCAL TRANSFER-ADMISSION)
xxxRTAD	DS	AL4	A(REMOTE TRANSFER-ADMISSION)
xxxRTADL	DS	AL2	L(REMOTE TRANSFER-ADMISSION)
*			
xxxSTACL	DS	XL2	L(STORAGE-ACCOUNT)
xxxSTAC	DS	AL4	A(STORAGE-ACCOUNT)
xxxLQF	DS	AL4	A(LEGAL-QUALIFICATION)
xxxLQFL	DS	XL2	L(LEGAL-QUALIFICATION)
*			
xxxAVAIL	DS	XL1	FILE-AVAIL.: NONE / IMMED / DEFER
xxxISTAC	DS	XL1	STORAGE-ACCOUNT: NONE
xxxILQF	DS	XL1	LEGAL-QUALIFICATION: UNCHG
*			
xxxACCES	DS	XL1	ACCESS-MODE : NONE / PAR
xxxREADF	DS	XL1	READ-FILE: NO / YES
xxxREPLF	DS	XL1	REPLACE-FILE: NO / YES
xxxEXTDF	DS	XL1	EXTEND-FILE: NO / YES
xxxREADA	DS	XL1	READ-ATTRIBUTES: NO / YES
xxxCHNGA	DS	XL1	CHANGE-ATTRIBUTES: NO / YES
xxxDELF	DS	XL1	DELETE-FILE: NO / YES
xxxINSDU	DS	XL1	INSERT-DATA-UNIT: NO / YES
xxxERADU	DS	XL1	ERASE-DATA-UNIT: NO / YES
*			
	DS	OH	
xxxRCPAL	DS	XL2	L(CREATE-PASSWORD)
xxxRCPA	DS	AL4	A(CREATE-PASSWORD)
*			
xxxRSIZE	DS	XL4	RECORD-SIZE: *NOT-SPEC / 1..32756
*			
xxxRID	DS	XL4	REQUEST ID
xxxFUD	DS	AL4	A(FURTHER DETAILS)
xxxLFUD	DS	XL2	L(FURTHER DETAILS)
xxxASYMG	DS	XL1	ASYNCHRONOUS END MESSAGE

xxxSCSET DS	XL1	CHARACTER SET
xxxRFORM DS	XL1	RECORD FORMAT = STD/VAR/FIX/UNDEF
*		
xxxTABEX DS	XL1	TABULATOR EXPANSION
xxxLCCSL DS	XL2	L(LOCAL-CODE-CHARACTER-SET)
xxxLCCS DS	AL4	A(LOCAL-CODE-CHARACTER-SET)
xxxRCCS DS	AL4	A(REMOTE-CODE-CHARACTER-SET)
xxxRCCSL DS	XL2	L(REMOTE-CODE-CHARACTER-SET)
*		
xxxLNCO EQU	*-xxxNCOPY	
*		
xxxTRDTO EQU	X'01'	TRANSFER DIRECTION = TO
xxxTRDFR EQU	X'00'	TRANSFER DIRECTION = FROM
xxxCOMBY EQU	X'80'	COMPRESSION = BYTE
xxxCOMZI EQU	X'40'	COMPRESSION = ZIP
xxxCOMNO EQU	X'00'	COMPRESSION = NONE
xxxMSP EQU	X'03'	REMOTESYNTAX= MSP
xxxBS2 EQU	X'01'	REMOTESYNTAX= BS2
xxxANY EQU	X'02'	REMOTESYNTAX= ANY
xxxNEW EQU	X'01'	WRITE-MODE = NEW FILE
xxxEXT EQU	X'04'	WRITE-MODE = EXTEND
xxxRPL EQU	X'02'	WRITE-MODE = REPLACE
xxxCHRS EQU	X'88'	DATA TYPE = CHARACTER(TRANS=STD)
xxxBINS EQU	X'84'	DATA TYPE = BINARY(TRANS=STD)
xxxCHR EQU	X'08'	DATA TYPE = CHARACTER
xxxBIN EQU	X'04'	DATA-TYPE = BINARY
xxxNOTSP EQU	X'02'	DATA-TYPE = NOT-SPECIFIED
xxxUSER EQU	X'01'	DATA-TYPE = USER
xxxSTDLC EQU	X'0'	LELVER = STD BEI BS2/LOCAL
xxxSAME EQU	X'0'	RELVER = SAME BEI BS2/REMOTE
xxxSTDRE EQU	X'FF'	RELVER = STD BEI BS2/REMOTE
xxxNONE EQU	X'0'	RELVER = NONE BEI *ANY
xxxNSPEC EQU	X'FF'	RFIL/LPUS/RPUS = NOT-SPECIFIED
xxxCNES EQU	X'01'	COMPRESSION = NONE/ENCRYPTION = YES
xxxCZES EQU	X'41'	COMPRESSION = ZIP/ENCRYPTION = YES
xxxCBES EQU	X'81'	COMPRESSION = BYTE/ENCRYPTION = YES
xxxCNOD EQU	X'02'	COMPRESSION = NONE/ENCRYPTION = ODI
xxxCZOD EQU	X'42'	COMPRESSION = ZIP/ENCRYPTION = ODI
xxxCBOD EQU	X'82'	COMPRESSION = BYTE/ENCRYPTION = ODI
xxxRFSTD EQU	X'00'	RFORM = STD
xxxRFVAR EQU	X'1A'	RFORM = VAR
xxxRFFIX EQU	X'1B'	RFORM = FIX
xxxRFUND EQU	X'1C'	RFORM = UNDEF

If an operand is not to be assigned, the value of the address field and the length field must be set to binary zero.

If the value *NOT-SPECIFIED is entered for the parameter FILE-NAME of the remote system, then the value set in the equate xxxNSPEC is to be written in the value field for the filename (length 1).

If this value is entered for the PROCESSING-ADMISSION of the local or remote system, then it is to be written in the value field for the corresponding USER-ID.

Meaning of the fields

The macro NCOPYSYN offers you the same options as the command TRANSFER-FILE (NCOPY) apart from the exceptions listed on [page 29](#). The following table shows how the parameters in the macro NCOPYSYN correspond to the command operands.

Operand (L format)	Address field (D format)	Meaning in command
TRANS	TRDIR	TRANSFER-DIRECTION
PARTNER	PART	PARTNER-NAME
LFILE	LFIL	FILE-NAME in LOCAL parameter.
LLIB	LLIB	LIBRARY in LOCAL parameter
LEL	LEL	ELEMENT in LOCAL parameter
LELVER	LEV	VERSION in LOCAL parameter
LLIBTYP	LETY	TYPE in LOCAL parameter
LFPASS	LFPA	FILE-PASSWORD in LOCAL parameter
LTUSER	LTUS	USER-IDENTIFICATION in local TRANSFER-ADMISSION
LTACC	LTAC	ACCOUNT in local TRANSFER-ADMISSION ¹
LTPASS	LTPA	PASSWORD in local TRANSFER-ADMISSION
LTAD	LTAD	TRANSFER-ADMISSION (local)
LCCNS	LCCS	CODED-CHARACTER-SET in LOCAL-PARAMETER
RSYNTAX	RSYN	REMOTE parameter
RFILE	RFIL	FILE-NAME in REMOTE parameter
RLIB	RLIB	LIBRARY in REMOTE parameter
REL	REL	ELEMENT in REMOTE parameter
RELVER	REV	VERSION in REMOTE parameter
RLIBTYP	RETY	TYPE in REMOTE parameter
RFPASS	RFPA	FILE-PASSWORD in REMOTE parameter
RTUSER	RTUS	USER-IDENTIFICATION in remote TRANSFER-ADMISSION
RTACC	RTAC	ACCOUNT in remote TRANSFER-ADMISSION
RTPASS	RTPA	PASSWORD in remote TRANSFER-ADMISSION
RTAD	RTAD	TRANSFER-ADMISSION (remote)

Operand (L format)	Address field (D format)	Meaning in command
RPUSER	RPUS	USER-IDENTIFICATION in remote PROCESSING-ADMISSION
RPACC	RPAC	ACCOUNT in remote PROCESSING-ADMISSION
RPPASS	RPPA	PASSWORD in remote PROCESSING-ADMISSION
RSUCC	RSUC	SUCCESS-PROCESSING in REMOTE parameter
RFAIL	RFAI	FAILURE-PROCESSING in REMOTE parameter
RCCNS	RCCS	CODED-CHARACTER-SET in REMOTE-PARAMETER
FAVAIL	AVAIL	FILE-AVAILABILITY
STORACC	STAC	STORAGE-ACCOUNT
ACCMOD	ACCES	ACCESS-MODE
READF	READF	READ-FILE in the structure ACCESS-MODE
INSDU	INSDU	INSERT-DATA-UNIT in the structure ACCESS-MODE
REPLF	REPLF	REPLACE-FILE in the structure ACCESS-MODE
EXTDF	EXTDF	EXTEND-FILE in the structure ACCESS-MODE
ERADU	ERADU	ERASE-DATA-UNIT in the structure ACCESS-MODE
READA	READA	READ-ATTRIBUTES in the structure ACCESS-MODE
CHNGA	CHNGA	CHANGE-ATTRIBUTES in the structure ACCESS-MODE
DELF	DELF	DELETE-FILE in the structure ACCESS-MODE
LQUAL	LQF	LEGAL-QUALIFICATION
RCPASS	RCPA	CREATE-PASSWORD in REMOTE parameter
COMP	COMP	COMPRESS
WRITE	WRMOD	WRITE-MODE
DATA	DATYP	DATA-TYPE
TRANSP	DATYP	TRANSPARENT
DATENCR	COMP	DATA-ENCRYPTION
TRANSID	TRID	_____
RSIZE	RSIZE	RECORD-SIZE
RFORM	RFORM	RECORD-FORMAT

¹ If LTACC is not specified, this corresponds to an account number not specified in the command, i.e. the user's default account number is used.

In the field for TRANS-ID or TRANSID, the TRANSFER-ID is entered after a successful NCOPYSYN macro (printable in decimal notation, left justified, filled with blanks on the right). The field must be ≥ 10 bytes or 0, if the TRANSFER-ID is not to be entered.

After a defective NCOPYSYN macro call, additional information (precise cause of the error) concerning the error message itself can be returned in the FUD (Further Details) field. The field must be ≤ 64 bytes or zero if the additional information is not to be stored in the FUD field.

If you run a file transfer with an FTAM or FTP partner from an ASSEMBLER program, the same function restrictions apply as at the command interface. You cannot transfer any library members which are in the remote system, specify follow-up processing for the remote system, or transfer files in compressed form. For more information, see the description of the command TRANSFER-FILE(NCOPY) in the user manual.

2.1.4 NDEL - Delete remote file

The macro NDEL is used to delete a file in an FT partner system. Its functionality corresponds to the command DELETE-REMOTE-FILE.

Format of the macro NDEL (LIST format/default format)

Name	Operation	Operand
[name]	NDEL	<pre> [MF=L,] PARTNER =adr [,FILE={*NOTSP} {adr}] [,FPASS={*NONE} {adr}] [,TAD={*NONE} {adr}] / [,UUSER=adr] [,UACC={*NONE} {adr}] [,UPASS={*NONE} {adr}]] [,MSGINS={*NONE} {adr}] [,AFPASS={[*]GRAPH} {[*]OCTET}] [,ATAD={[*]GRAPH} {[*]OCTET}] [,AUPASS={[*]GRAPH} {[*]OCTET}] [,Version={1} {2}] </pre>

The operands AFPASS, ATAD and AUPASS are used to set the coding type of the file password (FPASS), the transfer admission (TAD) and the user password (UPASS). The operands AFPASS, ATAD and AUPASS can have the value GRAPHIC or OCTET.

With GRAPHIC, the entry for the corresponding operand is interpreted as a printable character, and will be converted during transfer to a system which does not use EBCDIC. With OCTET, it is interpreted as binary information and will not be converted.

Format of the macro NDEL (DSECT format)

Name	Operation	Operand
[name]	NDEL	MF=(D[,xxx])

The prefix xxx (default value YNE) may be up to 3 characters long. It must generate names which are permitted in ASSEMBLER. The following ASSEMBLER instructions are generated:

```

xxxNDEL  DSECT
xxxHDEL  FHDR  MF=(C,&P)
xxxPART  DS    A                A(PARTNER-SYSTEM)
xxxPARTL DS    XL2              L(PARTNER-SYSTEM)
xxxFILEL DS    XL2              L(FILE-NAME)
xxxFILE  DS    A                A(FILE-NAME)
xxxFPAS  DS    A                A(FILE-PASSWORD)
xxxFPASL DS    XL2              L(FILE-PASSWORD)
xxxTADL  DS    XL2              L(TRANSFER-ADMISSION)
xxxTAD   DS    A                A(TRANSFER-ADMISSION)
xxxUUS   DS    A                A(USER-IDENTIFICATION)
xxxUUSL  DS    XL2              L(USER-IDENTIFICATION)
xxxUACL  DS    XL2              L(USER-ACCOUNT)
xxxUAC   DS    A                A(USER-ACCOUNT)
xxxUPA   DS    A                A(USER-PASSWORD)
xxxUPAL  DS    XL2              L(USER-PASSWORD)
xxxMSGL  DS    XL2              L(MESSAGE)
xxxMSG   DS    A                A(MESSAGE)
*
xxxIFILE DS    XL1              FILE-NAME:    NOTSP
xxxIFPAS DS    XL1              FILE-PASSWORD: NONE
xxxITAD  DS    XL1              TRANSFER-ADMISSION: NONE / PAR
xxxIUAC  DS    XL1              USER-ACCOUNT:  NONE
*
xxxIUPA  DS    XL1              USER-PASSWORD: NONE
xxxAPAS  DS    XL1              TYPE-FILE-PASSWORD: GRAPH/OCTET
xxxATAD  DS    XL1              TYPE-TRANSFER-ADMS: GRAPH/OCTET
xxxAUPA  DS    XL1              TYPE-USER-PASSWORD: GRAPH/OCTET
*

```

```

xxxFUD   DS   AL4           A(FURTHER DETAILS)
xxxLFUD  DS   XL2           L(FURTHER DETAILS)
*
xxxLNDE  EQU  *-xxx.NDEL

```

Meaning of the fields

The meaning of the parameters in the macro corresponds to the meaning of the parameters in the command DELETE-REMOTE-FILE. Please refer to the corresponding command description in the user manual.

Operand (L format)	Address field (D format)	Meaning in command
PARTNER	PART	PARTNER-NAME
FILE	FILE	FILE-NAME
FPASS	FPAS	PASSWORD to the file
TAD	TAD	TRANSFER-ADMISSION
UUSER	UUS	USER-IDENTIFICATION in the structure TRANSFER-ADMISSION
UACC	UAC	ACCOUNT in the structure TRANSFER-ADMISSION
UPASS	UPA	PASSWORD for identification in the structure TRANSFER-ADMISSION
MSGINS	MSG	Return value of the call
AFPASS	APAS	Coding of the FPASS (GRAPHIC or OCTET)
ATAD	ATAD	Coding of the TAD (GRAPHIC or OCTET)
AUPASS	AUPA	Coding of the UPASS (corresponding to GRAPHIC or OCTET)
VERSION	FCTV	———— (Type of return code output (old/new)) ¹

¹ Depending on the value of the VERSION operand, the FUNCTION INTERFACE VERSION NUMBER is set in the header FHDR.

After a defective NDEL macro call, additional information (precise cause of the error) concerning the error message itself can be returned in the FUD (Further Details) field. The field must be ≤ 64 bytes or zero if the additional information is not to be stored in the FUD field.

2.1.5 NLMOD - Modify local FT file attributes

The macro NLMOD can be used to modify the FTAM attributes of a file in the local system and adapt the attributes to make them available for a file transfer or file management request. The functionality corresponds to that of the command MODIFY-FILE-FT-ATTRIBUTES.

Format of the macro NLMOD (LIST format/default format)

Name	Operation	Operand
[name]	NLMOD	<pre> [MF=L,] FILE=adr [,FPASS={*NONE adr}] [,PERMACT=[*]UNCHG / [,PREADF={[*]NO [*]YES}] [,PINSU={[*]NO [*]YES}] [,PREPLF={[*]NO [*]YES}] [,PEXTDF={[*]NO [*]YES}] [,PERAU={[*]NO [*]YES}] [,PREADA={[*]NO [*]YES}] [,PCHNGA={[*]NO [*]YES}] </pre>

Name	Operation	Operand
		$[,PDELFL= \left\{ \begin{array}{l} [*]NO \\ [*]YES \end{array} \right\}]]$ $[,TRATT=[*]UNCHG /$ $[,DATA= \left\{ \begin{array}{l} [*]UNCHG \\ [*]BIN \\ [*]CHAR \end{array} \right\}]]$ $[,DCHRS= \left\{ \begin{array}{l} [*]GRSTR \\ [*]GENER \\ [*]IA5 \\ [*]VISIB \end{array} \right\}]]$ $[,RECFORM=[*]UNCHG]$ $[,RECSIZE= \left\{ \begin{array}{l} *UNCHG \\ int \end{array} \right\}]]$ $[,MSGINS= \left\{ \begin{array}{l} *NONE \\ adr \end{array} \right\}]]$ $[,AFPASS= \left\{ \begin{array}{l} [*]GRAPH \\ [*]OCTET \end{array} \right\}]]$ $[,Version= \left\{ \begin{array}{l} 1 \\ 2 \end{array} \right\}]]$

Format of the macro NLMOD (DSECT format)

Name	Operation	Operand
[name]	NLMOD	MF=(D[,xxx])

The prefix xxx (default value YNM) may be up to 3 characters long. It must generate names which are permitted in ASSEMBLER. The following ASSEMBLER instructions are generated:

```

xxxNLMO  DSECT
xxxHLMO  FHDR  MF=(C,&P)
xxxFILE  DS    A                A(FILE-NAME)
xxxFILEL DS    XL2             L(FILE-NAME)
xxxFPASL DS    XL2             L(FILE-PASSWORD)
xxxFPAS  DS    A                A(FILE-PASSWORD)
xxxMSG   DS    A                A(MESSAGE)
xxxMSGL  DS    XL2             L(MESSAGE)
*
xxxIFPAS DS    XL1             FILE-PASSWORD: NONE
xxxIREC  DS    XL1             RECORD-SIZE: UNCHG
xxxRECS  DS    XL4             (RECORD-SIZE)
*
xxxTRATT DS    XL1             TRANSFER-ATTRIBUTES: UNCHG / PAR
xxxDATA  DS    XL1             DATA-TYPE: UNCHG / BIN / CHAR
xxxCHRS  DS    XL1             CHAR.SET: GRSTR/GENER/IA5/VISIB
xxxRECF  DS    XL1             REC.FORMAT: UNCHG
*
xxxPACT  DS    XL1             PERMITTED-ACTIONS: UNCHG / PAR
xxxPREAF DS    XL1             READ-FILE:          NO / YES
xxxPINSU DS    XL1             INSERT-DATA-UNIT: NO / YES
xxxPREPF DS    XL1             REPLACE-FILE:       NO / YES
*
xxxPEXTF DS    XL1             EXTEND-FILE:          NO / YES
xxxPERAU DS    XL1             ERASE-DATA-UNIT:     NO / YES
xxxPREAA DS    XL1             READ-ATTRIBUTES:    NO / YES
xxxPCHNA DS    XL1             CHANGE-ATTRIBUTES: NO / YES
*
xxxPDELF DS    XL1             DELETE-FILE:          NO / YES
xxxAPAS  DS    XL1             TYPE-FILE-PASSWORD: GRAPH/OCTET
*
xxxSFPWD DS    XL1             SET / RESET FILE-PWD: NO / UNCHG
xxxRESRV DS    XL1             RESERVED
*
xxxFPWD  DS    A                A(FILE-PASSWORD FOR SET)
xxxFPWDL DS    XL2             L(FILE-PASSWORD FOR SET)
*
xxxRES2  DS    XL2             RESERVED
*
*
xxxLNLM  EQU    *-xxx.NLMO

```

Meaning of the fields

The meaning of the parameters in the macro correspond to the meaning of the parameters in the command MODIFY-FILE-FT-ATTRIBUTES. Please refer to the corresponding command description in the user manual.

Operand (L format)	Address field (D format)	Meaning in command
FILE	FILE	FILE-NAME
FPASS	FPAS	PASSWORD of the file
PERMACT	PACT	PERMITTED-ACTIONS
PREADF	PREAF	READ-FILE in the structure PERMITTED-ACTION
PINSU	PINSU	INSERT-DATA-UNIT in the structure PERMITTED-ACTION
PREPLF	PREPF	REPLACE-FILE in the structure PERMITTED-ACTION
PEXTDF	PEXTF	EXTEND-FILE in the structure PERMITTED-ACTION
PERAU	PERAU	ERASE-DATA-UNIT in the structure PERMITTED-ACTION
PREADA	PREAA	READ-ATTRIBUTES in the structure PERMITTED-ACTION
PCHNGA	PCHNA	CHANGE-ATTRIBUTES in the structure PERMITTED-ACTION
PDELF	PDELF	DELETE-FILE in the structure PERMITTED-ACTION
TRATT	TRATT	TRANSFER-ATTRIBUTES
DATA	DATA	DATA-TYPE in the structure TRANSFER-ATTRIBUTES
DCHRS	CHRS	CHARACTER-SET in DATA-TYPE
RECFORM	RECF	RECORD-FORMAT in the structure TRANSFER-ATTRIBUTES
RECSIZE	RECS	RECORD-SIZE in the structure TRANSFER-ATTRIBUTES
MSGINS	MSG	Return value of the call
AFPASS	APAS	Coding of the FPASS (GRAPHIC or OCTETT)
VERSION	FCTV	———— (Type of return code output (old/new)) ¹

¹ Depending on the value of the VERSION operand, the FUNCTION INTERFACE VERSION NUMBER is set in the header FHDR.

2.1.6 NLSHOW - Display local FT file attributes

The macro NLSHOW can be used to view the FTAM attributes of a file in the local system. The functionality corresponds to that of the command SHOW-FILE-FT-ATTRIBUTES.

Format of the macro NLSHOW (LIST format/default format)

Name	Operation	Operand
[name]	NLSHOW	<pre> [MF=L,] FILE=adr [,INFO={ [*]STD [*]ALL [*]ONLY }] [,OUTPUT={ [*]SYSOUT [*]SYSLST }] [,MSGINS={ *NONE adr }] [,LAYOUT={ [*]STD [*]CSV }] [,Version={ 1 2 }] </pre>

Format of the macro NLSHOW (DSECT format)

Name	Operation	Operand
[name]	NLSHOW	MF=(D[,xxx])

The prefix xxx (default value YNH) may be up to 3 characters long. It must generate names which are permitted in ASSEMBLER. The following ASSEMBLER instructions are generated:

```

xxxNLSH  DSECT
xxxHLSH  FHDR  MF=(C,&P)
xxxFILE  DS    A                      A(FILE-NAME)
xxxFILEL DS    XL2                    L(FILE-NAME)
xxxMSG   DS    XL2                    L(MESSAGE)
xxxMSG   DS    A                      A(MESSAGE)
*
xxxOUTPT DS    XL1                    OUTPUT: SOUT / SLST / SOUTC / SLSTC
xxxINFO  DS    XL1                    INFORMATION: STD / ONLY / ALL
xxxRESRV DS    XL2                    RESERVED
*
xxxLNLS  EQU   *-xxx.NLSH

```

Meaning of the fields

The meaning of the parameters in the macro correspond to the meaning of the parameters in the command SHOW-FILE-FT-ATTRIBUTES. Please refer to the corresponding command description in the user manual.

In the OUTPT address field, SOUT stands for OUTPUT=*SYSOUT in the command, and SLST stands for OUTPUT=*SYSLST. SOUTC and SLSTC stand for their respective outputs in CSV format (LAYOUT=*CSV). For the other two values, output is in the standard format (LAYOUT=*STD).

Operand (L format)	Address field (D format)	Meaning in command
FILE	FILE	FILE-NAME
INFO	INFO	INFORMATION
OUTPUT	OUTPT	OUTPUT
LAYOUT	OUTPT	LAYOUT
MSGINS	MSG	Return value of the call
VERSION	FCTV	———— (Type of return code output (old/new)) ¹

¹ Depending on the value of the VERSION operand, the FUNCTION INTERFACE VERSION NUMBER is set in the header FHDR.

2.1.7 NMOD - Modify remote file attributes

The macro NMOD can be used to modify the attributes of a file in an FT partner system. The functionality corresponds to that of the command MODIFY-REMOTE-FILE-ATTRIBUTES.

Format of the macro NMOD (LIST format/default format)

Name	Operation	Operand
[name]	NMOD	<pre> [MF=L,] PARTNER=adr [,FILE={*NOTSP} {adr}] [,FPASS={*NONE} {adr}] [,TAD= {*NONE} {adr} / [,UUSER=adr] [,UACC={*NONE} {adr}] [,UPASS={*NONE} {adr}]]] [,NEWNAME={*SAME} {adr}] [,FAVAI L={[*]UNCHG} {[*]IMMED} {[*]DEFER}] [,STORACC={*UNCHG} {adr}] [,FFSIZE={*UNCHG} {int}] </pre>

Name	Operation	Operand
		[,LQUAL={ <u>*UNCHG</u> adr }]
		[,MSGINS={ <u>*NONE</u> adr }]
		[,AFPASS={ [<u>*</u>]GRAPH [<u>*</u>]OCTET }]
		[,ATAD={ [<u>*</u>]GRAPH [<u>*</u>]OCTET }]
		[,AUPASS={ [<u>*</u>]GRAPH [<u>*</u>]OCTET }]
		[,ACCMOD={ [<u>*</u>]UNCHG [<u>*</u>]REPL }]
		[,AREADF={ [<u>*</u>]NO [<u>*</u>]YES }]
		[,AINSU={ [<u>*</u>]NO [<u>*</u>]YES }]
		[,AREPLF={ [<u>*</u>]NO [<u>*</u>]YES }]
		[,AEXTDF={ [<u>*</u>]NO [<u>*</u>]YES }]
		[,AERAU={ [<u>*</u>]NO [<u>*</u>]YES }]
		[,AREADA={ [<u>*</u>]NO [<u>*</u>]YES }]
		[,ACHNGA={ [<u>*</u>]NO [<u>*</u>]YES }]
		[,ADELF={ [<u>*</u>]NO [<u>*</u>]YES }]

Name	Operation	Operand
		[,Version= $\left. \begin{array}{l} 1 \\ 2 \end{array} \right\}$]

The operands AFPASS, ATAD and AUPASS are used to set the coding type of the file password (FPASS), the transfer admission (TAD) and the user password (UPASS). The operands AFPASS, ATAD and AUPASS can have the value GRAPHIC or OCTET.

With GRAPHIC, the entry for the corresponding operand is interpreted as a printable character, and will be converted during transfer to a system which does not use EBCDIC. With OCTET, it is interpreted as binary information and will not be converted.

Format of the macro NMOD (DSECT format)

Name	Operation	Operand
[name]	NMOD	MF=(D[,xxx])

The prefix xxx (default value YNU) may be up to 3 characters long. It must generate names which are permitted in ASSEMBLER. The following ASSEMBLER instructions are generated:

```

xxxNMOD  DSECT
xxxHMOD  FHDR  MF=(C,&P)
xxxPART  DS    A                A(PARTNER-SYSTEM)
xxxPARTL DS    XL2             L(PARTNER-SYSTEM)
xxxFILEL DS    XL2             L(FILE-NAME)
xxxFILE  DS    A                A(FILE-NAME)
xxxFPAS  DS    A                A(FILE-PASSWORD)
xxxFPASL DS    XL2             L(FILE-PASSWORD)
xxxTADL  DS    XL2             L(TRANSFER-ADMISSION)
xxxTAD   DS    A                A(TRANSFER-ADMISSION)
xxxUUS   DS    A                A(USER-IDENTIFICATION)
xxxUUSL  DS    XL2             L(USER-IDENTIFICATION)
xxxUACL  DS    XL2             L(USER-ACCOUNT)
xxxUAC   DS    A                A(USER-ACCOUNT)
xxxUPA   DS    A                A(USER-PASSWORD)
xxxUPAL  DS    XL2             L(USER-PASSWORD)
xxxNNAML DS    XL2             L(NEW-NAME)
xxxNNAM  DS    A                A(NEW-NAME)
xxxSTAC  DS    A                A(STORAGE-ACCOUNT)
xxxSTACL DS    XL2             L(STORAGE-ACCOUNT)
xxxLQFL  DS    XL2             L(LEGAL-QUALIFICATION)
xxxLQF   DS    A                A(LEGAL-QUALIFICATION)
xxxMSG   DS    A                A(MESSAGE)

```

xxxMSGL DS	XL2	L(MESSAGE)
*		
xxxIFILE DS	XL1	FILE-NAME: NOTSP
xxxIFFS DS	XL1	FUTURE-FILE-SIZE: UNCHG
xxxFFS DS	XL4	(FUTURE-FILE-SIZE)
*		
xxxIFPAS DS	XL1	FILE-PASSWORD: NONE
xxxITAD DS	XL1	TRANSFER-ADMISSION: NONE / PAR
xxxIUAC DS	XL1	USER-ACCOUNT: NONE
xxxIUPA DS	XL1	USER-PASSWORD: NONE
*		
xxxINNAM DS	XL1	NEW-NAME: SAME
xxxISTAC DS	XL1	STORAGE-ACCOUNT: UNCHG
xxxILQF DS	XL1	LEGAL-QUALIFICATION : UNCHG
xxxAVAIL DS	XL1	FILE-AVAIL.: UNCHG/IMMED/DEFER
*		
xxxACCES DS	XL1	ACCESS-MODE: UNCHG / REPL
xxxAREAF DS	XL1	READ-FILE: NO / YES
xxxAINSU DS	XL1	INSERT-DATA-UNIT: NO / YES
xxxAREPF DS	XL1	REPLACE-FILE: NO / YES
*		
xxxAEXTF DS	XL1	EXTEND-FILE: NO / YES
xxxAERAU DS	XL1	ERASE-DATA-UNIT: NO / YES
xxxAREAA DS	XL1	READ-ATTRIBUTES: NO / YES
xxxACHNA DS	XL1	CHANGE-ATTRIBUTES: NO / YES
*		
xxxADELF DS	XL1	DELETE-FILE: NO / YES
xxxAPAS DS	XL1	TYPE-FILE-PASSWORD: GRAPH/OCTET
xxxATAD DS	XL1	TYPE-TRANSFER-ADMS: GRAPH/OCTET
xxxAUPA DS	XL1	TYPE-USER-PASSWORD: GRAPH/OCTET
*		
xxxFFSZH DS	XL4	FUTURE FILE SIZE HIGH
xxxFFSZL DS	XL4	FUTURE FILE SIZE LOW
xxxLNMO EQU	*-xxx.NMOD	

Meaning of the fields

The meaning of the parameters in the macro correspond to the meaning of the parameters in the command MODIFY-REMOTE-FILE-ATTRIBUTES. Please refer to the corresponding command description in the user manual.

Operand (L format)	Address field (D format)	Meaning in command
PARTNER	PART	PARTNER-NAME
FILE	FILE	FILE-NAME
FPASS	FPAS	PASSWORD
TAD	TAD	TRANSFER-ADMISSION
UUSER	UUS	USER-IDENTIFICATION
UACC	UAC	ACCOUNT
UPASS	UPA	PASSWORD
NEWNAME	NNAM	NEW-NAME
FAVAIL	AVAIL	FILE-AVAILABILITY
STORACC	STAC	STORAGE-ACCOUNT
FFSIZE	FFS	FUTURE-FILE-SIZE
LQUAL	LQF	LEGAL-QUALIFICATION
MSGINS	MSG	Return value of the call
AFPASS	APAS	Coding of the FPASS (GRAPHIC or OCTET)
ATAD	ATAD	Coding of the TAD (GRAPHIC or OCTET)
AUPASS	AUPA	Coding of the UPASS (corresponding to GRAPHIC or OCTET)
ACCMOD	ACCES	ACCESS-MODE
AREADF	AREAF	READ-FILE
AINSU	AINSU	INSERT-DATA-UNIT
AREPLF	AREPF	REPLACE-FILE
AEXTDF	AEXTF	EXTEND-FILE
AERAU	AERAU	ERASE-DATA-UNIT
AREADA	AREAA	READ-ATTRIBUTES
ACHNGA	ACHNA	CHANGE-ATTRIBUTES
ADELFF	ADELFF	DELETE-FILE
VERSION	FCTV	————— (Type of return code output (old/new)) ¹

¹ Depending on the value of the VERSION operand, the FUNCTION INTERFACE VERSION NUMBER is set in the header FHDR.

2.1.8 NSHOW - Display remote file attributes

The macro NSHOW can be used to view the attributes of a file or directory in an FT partner system. The functionality corresponds to that of the command SHOW-REMOTE-FILE-ATTRIBUTES.

Format of the macro NSHOW (LIST format/default format)

Name	Operation	Operand
[name]	NSHOW	<pre> [MF=L,] PARTNER=adr [,FILE={ [*NOTSP] adr [*DIR] }] [,DIR={ [*NOTSP] adr }] [,FPASS={ [*NONE] adr }] [,TAD={ [*NONE] adr } / [,UUSER=adr] [,UACC={ [*NONE] adr }] [,UPASS={ [*NONE] adr }]] [,INFO={ [*]STD [*]ALL [*]ONLY }] [,OUTPUT={ [*]SYSOUT [*]SYSLST }] [,MSGINS={ [*NONE] adr }] [,AFPASS={ [*]GRAPH [*]OCTET }] </pre>

Name	Operation	Operand
		[, ATAD = { [*] GRAPH } [*] OCTET }]
		[, AUPASS = { [*] GRAPH } [*] OCTET }]
		[, LAYOUT = { [*] STD } [*] CSV }]
		[, Version = { 1 } 2 }]

The operands AFPASS, ATAD and AUPASS are used to set the coding type of the file password (FPASS), the transfer admission (TAD) and the user password (UPASS). The operands AFPASS, ATAD and AUPASS can have the value GRAPHIC or OCTET.

With GRAPHIC, the entry for the corresponding operand is interpreted as a printable character, and will be converted during transfer to a system which does not use EBCDIC. With OCTET, it is interpreted as binary information and will not be converted.

Format of the macro NSHOW (DSECT format)

Name	Operation	Operand
[name]	NSHOW	MF=(D[, xxx])

The prefix xxx (default value YNW) may be up to 3 characters long. It must generate names which are permitted in ASSEMBLER. The following ASSEMBLER instructions are generated:

```

xxxNSHW  DSECT
xxxHSHW  FHDR  MF=(C, &P)
xxxPART  DS    A                A(PARTNER-SYSTEM)
xxxPARTL DS    XL2             L(PARTNER-SYSTEM)
xxxFILEL DS    XL2             L(FILE-NAME)
xxxFILE  DS    A                A(FILE-NAME)
xxxDIR   DS    A                A(DIRECTORY)
xxxDIRL  DS    XL2             L(DIRECTORY)
xxxFPASL DS    XL2             L(FILE-PASSWORD)
xxxFPAS  DS    A                A(FILE-PASSWORD)
xxxTAD   DS    A                A(TRANSFER-ADMISSION)
xxxTADL  DS    XL2             L(TRANSFER-ADMISSION)
xxxUUSL  DS    XL2             L(USER-IDENTIFICATION)

```

xxxUUS	DS	A	A(USER-IDENTIFICATION)
xxxUAC	DS	A	A(USER-ACCOUNT)
xxxUACL	DS	XL2	L(USER-ACCOUNT)
xxxUPAL	DS	XL2	L(USER-PASSWORD)
xxxUPA	DS	A	A(USER-PASSWORD)
xxxMSG	DS	A	A(MESSAGE)
xxxMSGL	DS	XL2	L(MESSAGE)
*			
xxxIFILE	DS	XL1	FILE-NAME: NOTSP / DIR
xxxIDIR	DS	XL1	DIRECTORY: NOTSP
*			
xxxIFPAS	DS	XL1	FILE-PASSWORD: NONE
xxxITAD	DS	XL1	TRANSFER-ADMISSION: NONE / PAR
xxxIUAC	DS	XL1	USER-ACCOUNT: NONE
xxxIUPA	DS	XL1	USER-PASSWORD: NONE
*			
xxxOUTPT	DS	XL1	OUTPUT: SOUT / SLST / SOUTC / SLSTC
xxxINFO	DS	XL1	INFORMATION: STD / ONLY / ALL
xxxAPAS	DS	XL1	TYPE-FILE-PASSWORD: GRAPH/OCTET
xxxATAD	DS	XL1	TYPE-TRANSFER-ADMS: GRAPH/OCTET
xxxAUPA	DS	XL1	TYPE-USER-PASSWORD: GRAPH/OCTET
xxxRESRV	DS	XL3	RESERVED
*			
xxxRESRV	DS	XL59	RESERVED
*			
xxxRBUF	DS	A	A(RETURNING OUTPUT)
xxxRLEN	DS	A	IN: A(OUTPUT LEN) OUT: A(INFO LEN)
xxxCALLB	DS	A	A(CALLBACK ROUTINE)
xxxFUD	DS	A	A(FURTHER DETAILS)
xxxLFUD	DS	XL2	L(FURTHER DETAILS)
*			
xxxLNSH	EQU	*-xxx.NSHW	

Meaning of the fields

The meaning of the parameters in the macro correspond to the meaning of the parameters in the command SHOW-REMOTE-FILE-ATTRIBUTES. Please refer to the corresponding command description in the user manual.

In the OUTPT address field, SOUT stands for OUTPUT=*SYSOUT in the command, and SLST stands for OUTPUT=*SYSLST. SOUTC and SLSTC stand for their respective outputs in CSV format (LAYOUT=*CSV). For the other two values, output is in the standard format (LAYOUT=*STD).

Operand (L format)	Address field (D format)	Meaning in command
PARTNER	PART	PARTNER-NAME
FILE	FILE	FILE-NAME
DIR	DIR	DIRECTORY
FPASS	FPAS	PASSWORD
TAD	TAD	TRANSFER-ADMISSION
UUSER	UUS	USER-IDENTIFICATION in the structure TRANSFER-ADMISSION
UACC	UAC	ACCOUNT in the structure TRANSFER-ADMISSION
UPASS	UPA	PASSWORD in the structure TRANSFER-ADMISSION
INFO	INFO	INFORMATION
OUTPUT	OUTPT	OUTPUT
LAYOUT	OUTPT	LAYOUT
MSGINS	MSG	Return value of the call
AFPASS	APAS	Coding of the FPASS (GRAPHIC or OCTET)
ATAD	ATAD	Coding of the TAD (GRAPHIC or OCTET)
AUPASS	AUPA	Coding of the UPASS (corresponding to GRAPHIC or OCTET)
VERSION	FCTV	————— (Type of return code output (old/new)) ¹

¹ Depending on the value of the VERSION operand, the FUNCTION INTERFACE VERSION NUMBER is set in the header FHDR.

2.1.9 NSTAT - Query file transfer status

The macro NSTAT can be used to obtain information on the status of your file transfer request via an ASSEMBLER program. The functionality corresponds to the command SHOW-FILE-TRANSFER (NSTATUS).

Format of the macro NSTAT(LIST format/default format)

Name	Operation	Operand
[name]	NSTAT	[MF=L,] [TRANSID=adr] [,SOWNER=adr] [,PARTNER=adr] { [,LFILE=adr [,LLIB=adr][,LEL=adr][,LELVER=adr][,LELTYP=adr] }] [,JVNAME=adr] [,JVPASS=adr] [,INIT={ BOTH LOCAL REMOTE }] [,STATE={ ALL SUSP LOCK WAIT ACT CAN FIN HOLD }] [,INFO={ STD ALL SUMMARY }] [,OUTPUT={ SYSOUT SYSLST }] [,LAYOUT={ [*]STD [*]CSV }] [,Version={ 1 2 }]

Format of the macro NSTAT (DSECT format)

Name	Operation	Operand
[name]	NSTAT	MF=(D[,xxx])

The prefix xxx (default value YNS) may be up to 3 characters long. It must generate names which are permitted in ASSEMBLER. The following ASSEMBLER instructions are generated:

```

xxxNSTAT DSECT
xxxHSTA FHDR MF=(C,&P)
xxxTRID DS A A(TRANSFER-ID)
xxxTRIDL DS XL2 L(TRANSFER-ID)
xxxSOWNL DS XL2 L(OWNER-ID )
xxxSOWN DS AL4 A(OWNER-ID )
xxxPART DS AL4 A(PARTNER-SYSTEM)
xxxPARTL DS XL2 L(PARTNER-SYSTEM)
xxxLFILL DS XL2 L(FILE-NAME)
xxxLFIL DS AL4 A(FILE-NAME)
xxxLLIB DS AL4 A(LIBRARY-NAME)
xxxLLIBL DS XL2 L(LIBRARY-NAME)
xxxLELL DS XL2 L(ELEMENT-NAME)
xxxLEL DS AL4 A(ELEMENT-NAME)
xxxLEV DS AL4 A(ELEMENT-VERSION)
xxxLEVL DS XL2 L(ELEMENT-VERSION)
xxxLETYL DS XL2 L(ELEMENT-TYPE)
xxxLETY DS AL4 A(ELEMENT-TYPE)
xxxJVNM DS AL4 A(JV-NAME)
xxxJVNML DS XL2 L(JV-NAME)
xxxJVPSL DS XL2 L(JV-PASSWORD)
xxxJVPS DS AL4 A(JV-PASSWORD)
xxxINIT DS XL1 INITIATOR
xxxSTAT DS XL1 TRANSFER-STATUS
xxxINFO DS XL1 INFORMATION
xxxOUTPT DS XL1 OUTPUT /LAYOUT
xxxLNST EQU *-xxxNSTAT
*
xxxIBOTH EQU 0 INITIATOR = BOTH
xxxILOC EQU 192 INITIATOR = LOCAL
xxxIREM EQU 128 INITIATOR = REMOTE
*
xxxSTAL EQU X'00' TRANSFER-STATUS = ALL
xxxSTSU EQU X'01' TRANSFER-STATUS = SUSP
xxxSTLO EQU X'02' TRANSFER-STATUS = LOCK
xxxSTWA EQU X'03' TRANSFER-STATUS = WAIT

```

```

xxxSTAC EQU X'04'          TRANSFER-STATUS = ACT
xxxSTFI EQU X'05'          TRANSFER-STATUS = FIN
xxxSTHO EQU X'06'          TRANSFER-STATUS = HOLD
xxxSTCA EQU X'07'          TRANSFER-STATUS = CANCELLED
*
xxxSTD EQU X'00'           INFORMATION = STD
xxxSUM EQU X'01'           INFORMATION = SUMMARY
xxxALL EQU X'02'           INFORMATION = ALL
*
xxxSOUT EQU X'00'          OUTPUT = SYSOUT
xxxSLST EQU X'02'          OUTPUT = SYSLST
xxxSOUTC EQU X'03'         OUTPUT = SYSOUT LAYOUT=CSV
xxxSLSTC EQU X'04'         OUTPUT = SYSLST LAYOUT=CSV
    
```

Meaning of the fields

The macro NSTAT offers the same options as the command SHOW-FILE-TRANSFER (NSTATUS). Please refer to the command description in the user manual. In the OUTPUT address field, SOUT stands for OUTPUT=*SYSOUT in the command and SLST stands for OUTPUT=*SYSLST. SOUTC and SLSTC stand for their respective output in CSV format (LAYOUT=*CSV). For the other two values, output is in standard format (LAYOUT=*STD). The following table shows the relation between the parameters in the macro NSTAT and the command operands:

Operand (L format)	Address field (D format)	Meaning in command
TRANSID	TRID	TRANSFER-ID
SOWNER	SOWN	OWNER-IDENTIFICATION
PARTNER	PART	PARTNER-NAME
LFILE	LFIL	FILE-NAME
LLIB	LLIB	LIBRARY
LEL	LEL	ELEMENT
LELVER	LEV	VERSION
LELTYP	LETY	TYPE
JVNAME	JVNM	MONJV
JVPASS	JVPS	JV-PASSWORD
INIT	INIT	INITIATOR
STATE	STAT	STATE
INFO	INFO	INFORMATION

Operand (L format)	Address field (D format)	Meaning in command
OUTPUT	OUTPT	OUTPUT
LAYOUT	OUTPT	LAYOUT
VERSION	FCTV	———— (Type of return code output (old/new)) ¹

¹ Depending on the value of the VERSION operand, the FUNCTION INTERFACE VERSION NUMBER is set in the header FHDR.

If you do not enter anything in the fields for the OWNER-ID, your own user ID will be used as the value (this corresponds to the default entry *OWN at the command interface. If you intentionally fill the OWNER-ID field with blanks, then all user IDs will be used (this corresponds to the entry *ALL at the command interface).

2.1.10 Set YNDEQU constants for the openFT macros

The macro YNDEQU is used to generate the constants for the openFT macros. Call the macro as follows:

Name	Operation	Operand
	YNDEQU	

No operands are used in the macro call. The equates are generated with the default prefix YND, the form in which they are used internally by other openFT macros.

```

YNDNOVAL EQU 0      NO VALUE SPECIFIED
YNDPAR   EQU 1      PARAMETER:
YNDNONE  EQU 2      NO PARAMETER: *NONE
YNDNOTSP EQU 3      NOT-SPECIFIED: *NOTSP
YNDOWN   EQU 4      REQUEST TO LOCAL SYSTEM:
YNDDIR   EQU 5      DIRECTORY: *DIR
*
YNDUNCHG EQU 6      UNCHANGED: [*]UNCHG
YNDSAME  EQU 7      FILE-NAME UNCHANGED: *SAME
*
YNDNO    EQU 8      DELETE ATTRIBUTE: [*]NO
YNDYES   EQU 9      SET ATTRIBUTE  : [*]YES
*
YNDGRAPH EQU 10     TYPE OF STRING VALUE = GRAPHIC: [*]GRAPH
YNDOCTET EQU 11     TYPE OF STRING VALUE = OCTET  : [*]OCTET
*
YNDBIN   EQU 20     FILE = BINARY: [*]BIN
YNDCHAR  EQU 21     FILE = TEXT  : [*]CHAR
*
YNDVISIB EQU 22     CODE = ISO 646           : [*]VISIB
YNDIA5   EQU 23     CODE = ISO 646 + CONTR.CHARS: [*]IA5
YNDGRSTR EQU 24     CODE = ISO 8859 (DEFAULT) : [*]GRSTR
YNDGENER EQU 25     CODE = ISO 8859 + CONTR.CHARS: [*]GENER
*
YNDVAR   EQU 26     RECORD-LENGTH = VARIABLE(DEF.): [*]VAR
YNDFIX   EQU 27     RECORD-LENGTH = FIXED           : [*]FIX
YNDUNDEF EQU 28     RECORD-LENGTH = UNDEFINED       : [*]UNDEF
*
YNDIMMED EQU 29     AVAILABILITY = IMMEDIATE: [*]IMMED
YNDDEFER EQU 30     AVAILABILITY = DEFERRED : [*]DEFER
*
YNDADD   EQU 31     ACCESS MODE = MODIFIED: [*]ADD
YNDREPL  EQU 32     ACCESS MODE = NEW      : [*]REPL

```

```

*
YNDSOUT EQU 40      OUTPUT = SYSOUT: [*]SYSOUT
YNDSLST EQU 41      OUTPUT = SYSLST: [*]SYSLST
*
YNDONLY EQU 42      INFORMATION = ONLY-NAMES: [*]ONLY
YNDSTD EQU 43       INFORMATION = STANDARD : [*]STD
YNDALL EQU 44       INFORMATION = ALL      : [*]JALL
*
YNDLOCAL EQU 50     INITIATOR = LOCAL
YNDREMOT EQU 51     INITIATOR = REMOTE
YNDBOTHS EQU 52     INITIATOR = (LOCAL,REMOTE)
*
YNDTODAY EQU 60     DATE = TODAY
YNDTOMOR EQU 61     DATE = TOMORROW
*
YNDTRANF EQU 1      TRANSFER-FILE
YNDREATT EQU 2      READ-FILE-ATTRIBUTES
YNDDELFI EQU 4      DELETE-FILE
YNDCREFI EQU 8      CREATE-FILE
YNDMOATT EQU 16     MODIFY-FILE-ATTRIBUTES
YNDREDIR EQU 32     READ-DIRECTORY
YNDMOVFI EQU 64     MOVE-FILE
*
ERROR CODES
*
YNDOK EQU 0         NO ERROR
*
MAIN CODE VALUES
*
BYTE 1 DEFINES THE ERROR
*
PERMANENT ERRORS
*
YND1INVA EQU 1     INVALID ADDRESS OF PARAMETER VALUE
YND1INC EQU 2     PARAMETER AND INDICATOR INCONSISTENT
YND1PINC EQU 3     INCONSISTENCY WITH OTHER PARAMETER
YND1YERR EQU 4     WRONG SYNTAX IN PARAMETER VALUE
*
REPARABLE ERRORS
*
YND1BUFS EQU 10    BUFFER TOO SMALL
YND1RNGE EQU 11    PARAMETER VALUE OUT OF RANGE
YND1KEYV EQU 12    INVALID KEYWORD VALUE
YND1MAND EQU 13    MANDATORY PARAMETER MISSING
*
SYSTEM ERROR
*
YND1SERR EQU 255   SYSTEM ERROR

```

*

BYTE2 DEFINES THE ERRONEOUS PARAMETER

*

YND2PAR	EQU	1	INVALID ADDRESS OF PARAMETER LIST
YND2PART	EQU	2	PARTNER
YND2FILE	EQU	3	FILENAME
YND2DIR	EQU	4	DIRECTORY
YND2MPW	EQU	5	MANAGEMENT(FILE) PASSWORD
YND2TAD	EQU	6	TRANSFER ADMISSION
YND2TUID	EQU	7	TRANSFER UID
YND2TACC	EQU	8	TRANSFER ACCOUNT
YND2TPW	EQU	9	TRANSFER PASSWORD
YND2NNAM	EQU	10	NEW FILENAME
YND2AVLB	EQU	11	AVAILABILITY
YND2STOR	EQU	12	STORAGE ACCOUNT
YND2CSET	EQU	13	CHARACTER SET
YND2RFMT	EQU	14	RECORD FORMAT
YND2RLEN	EQU	15	RECORD LENGTH
YND2LEGQ	EQU	16	LEGAL QUALIFICATION
YND2FFSZ	EQU	17	FUTURE FILESIZE
YND2PACT	EQU	18	PERMITTED ACTIONS
YND2PRDF	EQU	19	PERM.ACT. READ FILE
YND2PINS	EQU	20	PERM.ACT. INSERT DATA UNIT
YND2PRPF	EQU	21	PERM.ACT. REPLACE FILE
YND2PEXF	EQU	22	PERM.ACT. EXTEND FILE
YND2PERA	EQU	23	PERM.ACT. ERASE DATA UNIT
YND2PRDA	EQU	24	PERM.ACT. READ ATTRIBUTES
YND2PCHA	EQU	25	PERM.ACT. CHANGE ATTRIBUTES
YND2PDEF	EQU	26	PERM.ACT. DELETE FILE
YND2OUTP	EQU	27	OUTPUT MEDIUM
YND2INFO	EQU	28	INFORMATION LEVEL
YND2SYNM	EQU	29	SYNCHRONOUS MESSAGE
YND2ASYM	EQU	30	ASYNCHRONOUS MESSAGE
YND2MSIN	EQU	31	MESSAGE INSERT
YND2LOGI	EQU	32	LOGGING RECORD-ID.
YND2OWNR	EQU	33	OWNER
YND2DATE	EQU	34	DATE
YND2NMBR	EQU	35	NUMBER
YND2SEL	EQU	36	SELECT
YND2INIT	EQU	37	INITIATOR
YND2RTYP	EQU	38	RECORD TYPE
YND2FTFU	EQU	39	FT-FUNCTION
YND2FTAC	EQU	40	FTAC-FUNCTION
YND2RES	EQU	41	RESERVED PARAMETER
YND2TRAT	EQU	42	TRANSFER ATTRIBUTES
YND2REAS	EQU	43	REASON CODS
YND2TID	EQU	44	TRANSFER IDENTIFICATION

YND2MOBF EQU	45	MONITOR OBJECT FILE
YND2PRIO EQU	46	PRIORITY
YND2QPOS EQU	47	QUEUE POSITION
YND2LFPW EQU	48	LOCAL FILE PASSWORD
YND2RSYN EQU	49	REMOTE SYNTAX
YND2TDIR EQU	50	TRANSFER DIRECTION
YND2LLIB EQU	51	LOCAL LIBRARY NAME
YND2LEL EQU	52	LOCAL ELEMENT NAME
YND2LETY EQU	53	LOCAL ELEMENT TYP
YND2LELV EQU	54	LOCAL ELEMENT VERSION
YND2LPUI EQU	55	LOCAL PROCESSION ADMISSION UID
YND2LPAC EQU	56	LOCAL PROCESSION ADMISSION ACCOUNT
YND2LPPW EQU	57	LOCAL PROCESSION ADMISSION PASSWORD
YND2LSUC EQU	58	LOCAL SUCCESS PROCESSING
YND2LFAI EQU	59	LOCAL FAILURE PROCESSING
YND2LIST EQU	60	LISTING
YND2RFIL EQU	61	REMOTE FILENAME
YND2RLIB EQU	62	REMOTE LIBRARY NAME
YND2REL EQU	63	REMOTE ELEMENT NAME
YND2RETY EQU	64	REMOTE ELEMENT TYP
YND2RELV EQU	65	REMOTE ELEMENT VERSION
YND2RFPW EQU	66	REMOTE FILE PASSWORD
YND2RTAD EQU	67	REMOTE TRANSFER ADMISSION
YND2RTUI EQU	68	REMOTE TRANSFER ADMISSION UID
YND2RTAC EQU	69	REMOTE TRANSFER ADMISSION ACCOUNT
YND2RTPW EQU	70	REMOTE TRANSFER ADMISSION PASSWORD
YND2RPUI EQU	71	REMOTE PROCESSION ADMISSION UID
YND2RPAC EQU	72	REMOTE PROCESSION ADMISSION ACCOUNT
YND2RPPW EQU	73	REMOTE PROCESSION ADMISSION PASSWORD
YND2RSUC EQU	74	REMOTE SUCCESS PROCESSING
YND2RFAI EQU	75	REMOTE FAILURE PROCESSING
YND2COMP EQU	76	COMPRESSION
YND2WRMD EQU	77	WRITE MODE
YND2DATT EQU	78	DATA TYPE
YND2STIM EQU	79	START TIME
YND2CTIM EQU	80	CANCEL TIME
YND2MOPV EQU	81	MONITOR OBJECT PASSWORD
YND2STAC EQU	82	STORAGE ACCOUNT
YND2FORC EQU	83	FORCED CANCELLATION
YND2STAT EQU	84	REQUEST STATE OF SELECTION
YND2AMSG EQU	85	ASYNCHRONOUS END MESSAGE
YND2CRYP EQU	86	ENCRYPTION
YND2FUD EQU	87	FURDER DETAIL
YND2LCCS EQU	88	LOCAL CCS NAME
YND2RCCS EQU	89	REMOTE CCS NAME
YND2TBEX EQU	90	TABULATOR EXPANSION

*

SUBCODE1 DEFINES THE ERROR CLASS

*

YNDCOK	EQU	0	NO ERROR
YNDCPERR	EQU	1	PERMANENT ERROR
YNDCSERR	EQU	32	SYSTEM ERROR
YNDCRERR	EQU	64	REPARABLE ERROR
YNDCSHRT	EQU	128	SHORTAGE OF RESOURCES

*

SUBCODE2 DEFINES THE WARNINGS

2.1.11 Return codes

The return codes have been changed in version V10.0 of openFT. As a result, the output message numbers are different from in the past (openFT \leq V9.0). To maintain the compatibility of older programs, a new parameter (VERSION=) has been introduced for the macros NCAN, NCOPY, NDEL, NLMOD, NLSHOW, NMOD, NSHOW and NSTAT.

2.1.11.1 VERSION parameter

The VERSION parameter makes it possible to select either the old or the new return code output. If you select the old return code output (default value), then the compatibility of older programs is maintained, i.e. they can be used without adaptation. If you want to use the new return codes then you must adapt and recompile the programs.

$$[,Version = \left. \begin{array}{l} \{1\} \\ \{ \} \\ \{2\} \end{array} \right\}]$$

Value	Meaning
1 (default value)	Uses conventional return code output.
2	Uses the new return code output.
Incorrect specification	Uses the conventional return code output and generates an MNOTE 'OLD RETURN CODES USED'.

2.1.11.2 Format of the return codes

Each return code consists of a subcode2, a subcode1 and the maincode. The field for these return codes has the following format:

Subcode2	Subcode1	Maincode	
warnings	error class	Parameter error	
1 byte	1 byte	1 byte	1 byte

The field with the return code can be addressed with DSECT after a macro call (field xxxRETC, generated with the macro FHDR).

Subcode1

The error codes are divided into error classes. These are found in subcode .
The following error classes exist:

Value (decimal)	Meaning
0	No error. The call was successfully completed
1	Syntax error. No repeat is possible, there was a syntax error or equivalent parameter error.
32	System error. An internal error occurred during macro processing.
64	Permanent error. Repeat after correction of user input.
128	Temporary error. Wait and repeat.

The corresponding equates are defined in the macro YNDEQU with the prefix YNDC.

Subcode2

This field contains additional information or is zero.

Maincode

This field contains the exact error. It is divided into two bytes.

- If subcode 1 has the value 1 (syntax error, see above), then the flush-right byte (byte 1) specifies the maincode for the cause of error and the flush-left byte (byte 2) specifies the parameter. The flush-right byte can have the following values:

Value of Byte 1 (decimal)	Meaning (if the value of Subcode 1 is 1)
Permanent error	
1	the parameter field is partially or entirely in an invalid address space
2	the address or length field contradicts the keyword field
3	the parameter entry contradicts another parameter
4	the parameter value is too long or does not respect the syntax rules; e.g. FILENAME, PASSWORD

Value of Byte 1 (decimal)	Meaning (if the value of Subcode 1 is 1)
Errors which can be eliminated	
10	the buffer is too small; this error code is reserved for future expansions
11	the arithmetic parameter value (e.g. RECSIZE value) is invalid
12	the keyword entered is not permitted for this parameter
13	a mandatory parameter is missing

The corresponding equates for byte 1 are defined in the macro YNDEQU with the prefix YND1.

The corresponding equates for byte 2 are defined in the macro YNDEQU with the prefix YND2.

- If subcode1 has the value 32, 64 or 128 (decimal), then the maincode contains the message number as the result of the call, see the table on [page 68](#).

Examples

- In the case of a syntax error in the partner name, the subcode1 is X'01' and the maincode is X'0204'.
- If the remote transfer admission is violated, the subcode1 is X'40' and the maincode is X'879'.

Exception

If the maincode contains the value X'FFFF', the request could not be executed for reasons specified uniquely throughout the system. The equates valid in this case for maincode, subcode1 and subcode2 are defined in the macro FHDR, which generates the default header.

Subcode 2 (SC2), subcode 1 (SC1) and maincode (MC) are specified as hexadecimal values in the following tables. In the case of SC1 =X'20', X'40' and X'80', MC corresponds to the message number of the FTR message (decimal).

2.1.11.3 Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW



The table below lists only the return codes for the current version. For the return codes of openFT version V9.0, please see the manual for openFT V9.0.

Return code			Meaning
SC2	SC1	MC	
00	00	0000	The macro call was stored in the local system's request queue or the request was concluded successfully.
00	01	FFFF	In the expansion of the L-form of the macro call, a function is demanded which is not supported by FT.
00	02	FFFF	In the expansion of the L-form of the macro call, a function unit is demanded which is not supported by FT.
00	03	FFFF	In the expansion of the L-form of the macro call, a version is demanded which is not supported by FT.
00	20	0320	The macro call could not be executed because <ul style="list-style-type: none"> – the system was generated without job variables, or – the maximum number of job variables for the user has been reached, or – a temporary job variable was specified, or – the specified job variable for the user is not accessible, or – the specified job variable is monitoring another FT request, or – the specified password for the job variable is incorrect.
00	20	0321	FTAM, NDMS, FJAM, or operating system error other than a DMS or transport system error.
00	20	0322	Following termination of the file transfer request, the job variable monitoring the request has inconsistent contents. Possible cause: During transfer, the job variable was accessed other than for reading. This does not affect the result of the transfer and can be determined from the result list or the asynchronous end message.
00	20	0324	The macro call was not executed due to inconsistent request data.
00	20	0352	The macro call was not executed because <ul style="list-style-type: none"> – the system was generated without job variables, or – the maximum number of job variables for the user has been reached, or – a temporary job variable was specified, or – the specified job variable for the user is not accessible, or – the specified job variable is monitoring another FT request, or – the specified password for the job variable is incorrect.
00	20	0353	The macro call was not executed because of an internal error.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	20	0356	The macro call was not executed because there is insufficient space on the disk/partition on which the logging files are stored.
00	20	0357	The macro call was not executed because there is insufficient space on the disk/partition on which the internal files are stored.
00	20	035B	The macro call was not executed because of teleservice restrictions. Possible cause: Only one partner with the teleservice name is supported.
00	20	035E	The macro call was not executed because the required transfer protocol is not installed.
00	20	03E7	The macro call was not executed because of an abnormal termination.
00	40	0014	The command was not executed because <ul style="list-style-type: none"> – the send file is not located in the catalog or on a data volume in the local system, or – the send file is not or is no longer or the receive file is no longer located in the catalog or on a data volume in the corresponding system.
00	40	00EC	The macro call was rejected because the set instance is not present.
00	40	03FD	The macro call was not executed because the request has not yet been terminated with FORCE=NO.
00	40	040C	The macro call cannot be accepted because it can only be entered by authorized users.
00	40	040E	The macro call could not be executed because the request is terminating and can no longer be deleted.
00	40	0417	The macro call cannot be accepted because no requests were found.
00	40	07E0	The macro call could not be executed because the local directory is not empty.
00	40	07E1	The macro call cannot be accepted because the local send or receive file only permits certain access operations.
00	40	07E2	The macro call cannot be executed because the file properties could not be modified as desired. Possible causes: <ul style="list-style-type: none"> – no access rights for the local file – the desired transfer attributes for the local file are not compatible with the file's BS2000 properties
00	40	07E3	The macro call cannot be executed because the file owner is not the same as the user who requested the setting up of a local receive file.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	40	07E5	The macro call could not be executed because <ul style="list-style-type: none"> – the local CCS name is unknown, or – the CCS names of the send and receive files cannot be mapped to one another or – the partner system does not support the retrieval of files in transparent format.
00	40	07E6	The macro call cannot be accepted because the local file could not be created on a receive request because the specified path does not exist.
00	40	07E7	The macro call cannot be executed because a local receive file which already exists cannot be set up with WRITE-MODE=NEW. WRITE-MODE=NEW may also have been set as a result of a restriction in the employed transfer admission.
00	40	07E8	The macro call was not executed because the FT system only transfers individual file generations.
00	40	07E9	A DMS error occurred during the processing of the macro call. Additional information can be found in the field xxDMSCO (4 bytes).
00	40	07EA	The macro call cannot be executed because the relative file name was specified in the transfer request. The absolute file name as extended by openFT is longer than permitted.
00	40	07EB	The macro call was not executed because the local file name was not specified either explicitly or via the employed TRANSFER-ADMISSION.
00	40	07EC	The macro call was not executed because the management password of the local file is missing or incorrect.
00	40	07EE	The macro call was not executed because the local home directory was not found.
00	40	07F1	The macro call was not executed because <ul style="list-style-type: none"> – the owner of the send or receive file is not defined in the local system, or – the file owner is not the same as the user who requested the creation of a receive file.
00	40	07F2	The macro call was not executed because the password for the local send file or the receive file is missing or incorrect.
00	40	07F4	The macro call was not executed because the retention period which protects the local receive file against being overwritten (RETENTION PERIOD) has not yet expired.
00	40	07F5	The macro call cannot be executed because the local file is write-protected.
00	40	07F6	The macro call was not executed because of a file structure error.
00	40	07F7	The macro call was not executed because it is not possible to access the local file, for example because the absolute file name is too long.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	40	07F8	The macro call was not executed because <ul style="list-style-type: none"> – the CCS name of the send and receive files cannot be mapped to one another or – the partner system does not support the retrieval of files in transparent format.
00	40	07FA	The macro call could not be executed because it is not possible to add an extension to a local file for transparent transfer.
00	40	07FB	The macro was not executed because the local send file or the receive file only permits certain access modes (e.g. read only).
00	40	07FC	The macro call could not be executed because the total of follow-up processing + prefix + suffix (from profile) is too long.
00	40	07FD	The macro call was not executed because the specifications in one of the operands of the local PROCESSING-ADMISSION is incorrect.
00	40	07FE	The macro call was not executed because the specifications in one of the operands of the local TRANSFER-ADMISSION is incorrect.
00	40	07FF	The macro call was not executed because the request was rejected by the product FTAC-BS2000 due to missing admissions.
00	40	0800	The macro call was not executed because the required function for the selected protocol is not available.
00	40	0801	The macro call was not executed because local follow-up processing is only available for the openFT protocol.
00	40	0802	The macro call cannot be executed because the partner machine does not support the data integrity check function.
00	40	0803	The macro call cannot be executed because the partner machine does not support the data encryption function.
02	40	07DE	No file attributes apart from the file name were specified.
00	40	0816	The macro call was not executed because openFT is not authorized to process requests for this user.
00	40	0818	The macro call was cancelled because of the command /CANCEL-FILE-TRANSFER or /NCANCEL or because the time specified in /TRANSFER-FILE or /NCOPY has elapsed. Follow-up processing for the local system has started if no error occurred during this operation. Follow-up processing for the remote system will be started as soon as all the resources have been assigned.
00	40	0819	The macro call cannot be executed because errors occurred during encryption.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	40	081A	The macro call cannot be executed because the file owner is not the same as the user who requested the setting up of a local receive file.
00	40	081B	The macro call cannot be executed because the higher-level local directory could no longer be found.
00	40	081C	The macro call cannot be executed because the local file can no longer be accessed. It may have been deleted during a transfer.
00	40	081D	The macro call cannot be executed because the local send or receive file can no longer be accessed, for example because it was deleted during an interruption at the openFT system.
00	40	0820	The macro call was not executed because the local home directory could no longer be found.
00	40	0822	The macro call was not executed because <ul style="list-style-type: none"> – the owner of the local send file or the receive file is not defined in the corresponding system, or – the file owner is not the same as the user who requested the creation of a receive file.
00	40	0823	The macro call was not executed because the command executed by local preprocessing/postprocessing returned a result other than 'OK'.
00	40	0824	The macro call was not executed because the command executed by local preprocessing/postprocessing returned an exit code.
00	40	0825	The macro call was not executed because the password for the local send file or the receive file is missing or incorrect.
00	40	0826	The macro call was not executed because the request is now write-protected.
00	40	0827	The macro call was not executed because of a file structure error in the local file. File structure errors may be, for example: <ul style="list-style-type: none"> – The attributes of the send file are incomplete. – The data in the send file do not match the structure attributes. – The records in the send file are too long. – The send and receive files have a different structure in WRITE-MODE=EXTEND-FILE or -e (e.g. records of fixed/variable length). – The send or receive file is an element from an old LMS library (not PLAM). – The send file has a non-even blocking factor (e.g. BLKSIZE=(STD,1)) and the receive file is to be stored in an NK4 pubset.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	40	0829	The macro call was not executed because the local restart attempt was unsuccessful. Possible cause: A preprocessing/postprocessing command could not be completed before openFT terminated.
00	40	082A	The macro call was not executed because an error occurred during the final phase of local file transfer. In the case of extended transfers, the recipient should check whether the file was nevertheless transferred correctly. However, error follow-up processing is still performed if it has been specified.
00	40	082C	The command was not executed because <ul style="list-style-type: none"> – the local send file or the receive file only permits certain access modes (e.g. read only), or – a directory was specified as the source or destination for a file transfer.
00	40	082E	The macro call was not executed because the retention period which protects the local receive file against being overwritten (RETENTION PERIOD) has not yet expired.
00	40	082F	The macro call could not be executed because it is not possible to add an extension to a local file for transparent transfer.
00	40	0830	The macro call could not be executed because the file structure is not supported.
00	40	083D	The macro call cannot be executed because the request to establish a connection was rejected by the local transport system.
00	40	083E	The macro call cannot be executed because the data integrity check indicates an error.
00	40	083F	The macro call cannot be executed because <ul style="list-style-type: none"> – no key pair set is present, or – the key length was set to 0. Requests can only be carried out without data encryption or data integrity checking.
00	40	0840	The macro call cannot be executed because the partner machine does not support the data integrity check function.
00	40	0841	The macro call cannot be executed because the partner machine does not support the data encryption function.
00	40	0842	The macro call cannot be executed because the partner has rejected the instance ID of the local system for security reasons or because of an inconsistency. Possible cause: A network description contains both the instance ID and the migration ID <code>%.processor.entity</code> for different partners.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	40	0843	The macro call cannot be executed because the remote system has interrupted the request.
00	40	0844	The macro call cannot be executed because <ul style="list-style-type: none"> – the local application is not defined in the transport system, or – the tnsxd process is not running in the Unix system.
00	40	0846	The macro call cannot be executed because the local system could not be authenticated at the partner.
00	40	0847	The macro call cannot be executed because the local system is not known in the partner system (e.g. BS2000/OSD or z/OS)
00	40	0848	The macro call cannot be executed because the partner specified as the remote system cannot be expanded into an address in the local system.
00	40	0849	The macro call cannot be executed because the remote system could not be authenticated at the local system.
00	40	084A	The macro call cannot be executed because the connection has been rejected or interrupted.
00	40	085C	The macro call could not be executed because openFT is not authorized to process requests for this user.
00	40	085D	The macro call could not be executed because the remote directory is not empty.
00	40	085E	The macro call could not be executed because the file attributes in the remote system do not correspond to the request parameters. Possible cause: A directory was specified instead of a remote file.
00	40	085F	The properties of the remote file could not be modified as desired in the macro call. Possible causes: <ul style="list-style-type: none"> – no access rights for the file – the desired combination of access rights is not supported by the remote system – if the remote system is a BS2000 system: the file is protected by ACL
00	40	0860	The macro call cannot be executed because the file owner is not the same as the user who requested the setting up of a remote receive file.
00	40	0861	The macro call could not be executed because the CCS names of the send and receive files cannot be mapped to one another or because the partner system does not support the retrieval of files in transparent format.
00	40	0862	The macro call cannot be executed because the higher-level local directory in the partner system could not be found.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	40	0863	The macro call was not executed. Possible causes: – The command was not executed because an existing receive file may not be set up again with WRITE-MODE=NEW or the -n switch. WRITE-MODE=NEW or -n may also have been set via a restriction in the employed transfer admission. – ftcredir: The specified directory already exists.
00	40	0864	The macro call was not executed because the FT system only transfers individual file generations.
00	40	0865	A DMS error occurred during the processing of the macro call.
00	40	0866	The macro call was not executed because the resulting remote file name is too long. Possible causes: – value assignments outside of the permitted range of values – invalid operand separator – invalid value assignment character – partially qualified file names. – BS2000: The syntax error which occurred does not relate to a 'missing operand' or 'unknown keyword'.
00	40	0868	The macro call was not executed because the remote file name was not specified either explicitly or via the employed TRANSFER-ADMISSION.
00	40	0869	The macro call was not executed because the management password of the remote file is missing or incorrect.
00	40	086B	The macro call cannot be executed because the send file is not or is no longer located in the catalog or on a volume in the remote system.
00	40	086C	The macro call was not executed because the remote home directory was not found.
00	40	086F	The macro call was not executed because – the owner of the remote send file or the receive file is not defined in the corresponding system, or – the file owner is not the same as the user who requested the creation of a receive file.
00	40	0870	The macro call cannot be executed because the password for the send file or the receive file in the remote system is missing or incorrect.
00	40	0871	The macro call cannot be executed because the retention period which protects the receive file in the remote system against being overwritten (RETENTION PERIOD) has not yet expired.
00	40	0872	The macro call cannot be executed because the remote file or remote directory is write-protected.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	40	0873	The macro call cannot be executed because the file structure is not supported. An attempt was made, for example, to retrieve a PLAM library or ISAM file from the BS2000 system.
00	40	0874	The macro call cannot be executed because of a syntax error other than 'missing operand' or 'unknown keyword' in the remote system. Possible causes: <ul style="list-style-type: none"> – value assignments outside of the permitted range of values – invalid operand separator – invalid value assignment character – partially qualified file names
00	40	0875	The macro call could not be executed because the partner system does not support the transfer of files in transparent format.
00	40	0876	The macro call could not be executed because it is not possible to add an extension to a remote file for transparent transfer.
00	40	0877	The macro call cannot be executed because the remote system only permits certain access modes.
00	40	0878	The macro call cannot be executed because the length of remote follow-up processing has been exceeded.
00	40	0879	The macro call cannot be executed because <ul style="list-style-type: none"> – the specifications in one of the operands of the remote TRANSFER-ADMISSION are incorrect, or – the request was rejected by the FTAC because of missing admissions.
00	40	087A	The macro call cannot be executed because the required function is not supported.
00	40	087B	The macro call was not executed because the specifications in one of the operands of the remote PROCESSING-ADMISSION are incorrect.
00	40	0893	The macro call was not executed because openFT is not authorized to process requests for this user.
00	40	0894	The macro call was cancelled because the request was deleted in the remote system before termination.
00	40	0895	The macro call was not executed because the file owner is not the same as the user who requested the setting up of a remote receive file.
00	40	0896	The macro call was not executed because the higher-level remote directory could no longer be found.
00	40	0897	The macro call was not executed because an error occurred during input/output. Possible cause: The remote send or receive file was deleted during the transfer.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	40	089A	The macro call was not executed because the remote file is not or is no longer located in the catalog or on a volume in the corresponding system (e.g. after a restart).
00	40	089B	The macro call was not executed because the remote home directory could no longer be found.
00	40	089D	The macro call was not executed because <ul style="list-style-type: none"> – the owner of the send or receive file is not defined in the corresponding remote system, or – the file owner is not the same as the user who requested the creation of a receive file.
00	40	089E	The macro call was not executed because the command executed by remote preprocessing/postprocessing returned a result value other than 'OK'.
00	40	089F	The macro call was not executed because the command executed by remote preprocessing/postprocessing returned an exit code.
00	40	08A0	The macro call was not executed because the password for the remote send file or the receive file is missing or incorrect.
00	40	08A1	The macro call was not executed because the remote file or remote directory is write-protected.
00	40	08A2	The macro call was not executed because of a file structure error in the remote system. <ul style="list-style-type: none"> – File structure errors may be, for example: – The attributes of the send file are incomplete. – The data in the send file do not match the structure attributes. – The records in the send file are too long. – The send and receive files have a different structure in WRITE-MODE=EXTEND-FILE or parameter -e (e.g. records of fixed/variable length) – BS2000: The send or receive file is an element from an old LMS library (not PLAM). – BS2000: The send file has a non-even blocking factor (e.g. BLKSIZE=(STD,1)) and the receive file is to be stored in an NK4 pubset.
00	40	08A4	The restart could not be executed. It is possible that a restartable preprocessing/postprocessing operation could not be concluded before termination of the server process (max. waiting time: 10 minutes).
00	40	08A6	The macro call was not executed because <ul style="list-style-type: none"> – the remote send file or the receive file only permits certain access modes (e.g. read only), or – a directory was specified as the source or destination for a file transfer.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	40	08A8	The macro call cannot be executed because the file structure is not supported in the remote system. An attempt was made, for example, to retrieve a PLAM library or ISAM file from the BS2000 system.
00	40	08A9	The macro call was not executed because the retention period which protects the remote receive file against being overwritten (RETENTION PERIOD) has not yet expired.
00	40	08AA	The macro call could not be executed because it is not possible to add an extension to a remote file for transparent transfer.
00	40	08B2	The macro call cannot be accepted because the content of the specified job variable is inconsistent. Possible cause: The user made a non-read access to the job variable while it was monitoring an FT request. The content of the job variable can therefore no longer be used.
00	40	08B3	The macro call cannot be accepted because another process as well as openFT is using the job variable.
00	40	08B4	The macro call was not executed because the referenced job variable is not present.
00	80	0023	The macro call cannot be executed because the local send or receive file is already protected against simultaneous updating by another process.
00	80	0029	The macro call cannot be accepted because the maximum limit for file transfer requests has been reached.
00	80	006C	The macro call could not be accepted because the partner system is not currently available.
00	80	07ED	The macro call cannot be executed because <ul style="list-style-type: none"> – the volume for the local send file or the receive file is not mounted, is unknown or occupied, or – the file extends over more than one private disk, or – an attempt was made to transfer a file migrated with HSMS.
00	80	07F0	The macro call was not executed (or was discontinued) because <ul style="list-style-type: none"> – the permitted storage space in the receive system for the user ID specified in TRANSFER-ADMISSION has been reached, or – the send file contains an excessively long sequence of empty blocks, or – the primary and/or secondary assignment of the password-protected receive file is too small. Once this problem has occurred, the local receive file will no longer be extended or generated.
00	80	081D	The macro call was not executed because the local send or receive file is already protected against simultaneous updating by another process.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	80	081E	The macro call was not executed because <ul style="list-style-type: none"> – the volume for the local send file or the receive file is not mounted, is unknown or occupied, or – the file extends over more than one private disk, or – an attempt was made to transfer a file migrated with HSMS.
00	80	0821	The macro call was not executed (or was discontinued) because <ul style="list-style-type: none"> – the permitted storage space in the local receive system for the user ID specified in TRANSFER-ADMISSION has been reached, or – the local send file contains an excessively long sequence of empty blocks, or – the primary and/or secondary assignment of the password-protected local receive file is too small. <p>Once this problem has occurred, the local receive file will no longer be extended or generated.</p>
00	80	0828	The macro call was rejected because the partner system does not currently possess any resources for the acceptance of requests.
00	80	082D	The macro call was rejected because the request was rejected by FTAM with the specified FTAM diagnostic code.
00	80	084D	The macro call cannot be executed because there was no data exchange due to a line interruption or line procedure error.
00	80	084E	The macro call cannot be executed because no further transfers are currently possible since the maximum number of simultaneous transfers has been reached.
00	80	084F	The macro call cannot be executed because no data was transferred within the specified number of seconds because, for example, the connection is interrupted, the partner is not sending and the local system is waiting for data.
00	80	0867	The macro call was not executed because the remote send or receive file is already protected against simultaneous updating by another process.
00	80	086A	The macro call was not executed because <ul style="list-style-type: none"> – the volume for the remote send file or the receive file is not mounted, is unknown or occupied, or – the file extends over more than one private disk, or – an attempt was made to transfer a file migrated with HSMS.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

Return code			Meaning
SC2	SC1	MC	
00	80	086E	<p>The macro call was not executed (or was discontinued) because</p> <ul style="list-style-type: none"> – the permitted storage space in the receive system for the user ID specified in TRANSFER-ADMISSION has been reached, or – the remote send file contains an excessively long sequence of empty blocks, or – the primary and/or secondary assignment of the password-protected remote receive file is too small. <p>Once this problem has occurred, the remote receive file will no longer be extended or generated.</p>
00	80	087C	The macro call was not executed because the maximum limit for file transfer requests has been reached.
00	80	0898	The macro call was not executed because the remote send or receive file is already protected against simultaneous updating by another process. An attempt was made, for example, to access a library which is open in z/OS.
00	80	0899	<p>The macro call was not executed because</p> <ul style="list-style-type: none"> – the volume for the remote send file or the receive file is not mounted, is unknown or occupied, or – the file extends over more than one private disk, or – an attempt was made to transfer a file migrated with HSMS.
00	80	089C	<p>The macro call was not executed (or was discontinued) because</p> <ul style="list-style-type: none"> – the permitted storage space in the remote receive system for the user ID specified in TRANSFER-ADMISSION has been reached, or – the remote send file contains an excessively long sequence of empty blocks, or – the primary and/or secondary assignment of the password-protected remote receive file is too small. <p>Once this problem has occurred, the remote receive file will no longer be extended or generated.</p>
00	80	08A3	An NDMS error occurred during the processing of the macro call.
00	80	08A5	The macro call was rejected because the partner system does not currently possess any resources for the acceptance of requests.
00	80	08A7	The macro call was rejected by FTAM with the specified FTAM diagnostic code.

Table 1: Return codes for NCOPY, NDEL, NLMOD, NLSHOW, NMOD and NSHOW

2.1.11.4 Return codes for NCOPYSYN

Return code			Meaning
SC2	SC1	MC	
00	00	0000	The macro call was stored in the local system's request queue or the request was concluded successfully.
00	01	FFFF	In the expansion of the L-form of the macro call, a function is demanded which is not supported by FT.
00	02	FFFF	In the expansion of the L-form of the macro call, a function unit is demanded which is not supported by FT.
00	03	FFFF	In the expansion of the L-form of the macro call, a version is demanded which is not supported by FT.
00	20	0320	The macro call could not be executed because <ul style="list-style-type: none"> – the system was generated without job variables, or – the maximum number of job variables for the user has been reached, or – a temporary job variable was specified, or – the specified job variable for the user is not accessible, or – the specified job variable is monitoring another FT request, or – the specified password for the job variable is incorrect.
00	20	0321	FTAM, NDMS, FJAM, or operating system error other than a DMS or transport system error.
00	20	0322	Following termination of the file transfer request, the job variable monitoring the request has inconsistent contents. Possible cause: During transfer, the job variable was accessed other than for reading. This does not affect the result of the transfer and can be determined from the result list or the asynchronous end message.
00	20	0324	The macro call was not executed due to inconsistent request data.
00	20	0352	The macro call was not executed because <ul style="list-style-type: none"> – the system was generated without job variables, or – the maximum number of job variables for the user has been reached, or – a temporary job variable was specified, or – the specified job variable for the user is not accessible, or – the specified job variable is monitoring another FT request, or – the specified password for the job variable is incorrect.
00	20	0353	The macro call was not executed because of an internal error.
00	20	0356	The macro call was not executed because there is insufficient space on the disk/partition on which the logging files are stored.
00	20	0357	The macro call was not executed because there is insufficient space on the disk/partition on which the internal files are stored.

Table 2: Return codes for NCOPYSYN

Return code			Meaning
SC2	SC1	MC	
00	20	035B	The macro call was not executed because of teleservice restrictions. Possible cause: Only one partner with the teleservice name is supported.
00	20	035E	The macro call was not executed because the required transfer protocol is not installed.
00	20	03E7	The macro call was not executed because of an abnormal termination.
00	40	0014	The command was not executed because <ul style="list-style-type: none"> – the send file is not located in the catalog or on a data volume in the local system, or – the send file is not or is no longer or the receive file is no longer located in the catalog or on a data volume in the corresponding system.
00	40	07E0	The macro call could not be executed because the local directory is not empty.
00	40	07E1	The macro call cannot be accepted because the local send or receive file only permits certain access operations.
00	40	07E2	The macro call cannot be executed because the file properties could not be modified as desired. Possible causes: <ul style="list-style-type: none"> – no access rights for the local file – the desired transfer attributes for the local file are not compatible with the file's BS2000 properties
00	40	07E3	The macro call cannot be executed because the file owner is not the same as the user who requested the setting up of a local receive file.
00	40	07E5	The macro call could not be executed because <ul style="list-style-type: none"> – the local CCS name is unknown, or – the CCS name of the send and receive files cannot be mapped to one another or – the partner system does not support the retrieval of files in transparent format.
00	40	07E6	The macro call cannot be accepted because the local file could not be created on a receive request because the specified path does not exist.
00	40	07E7	The macro call cannot be executed because a local receive file which already exists cannot be set up with WRITE-MODE=NEW. WRITE-MODE=NEW may also have been set as a result of a restriction in the employed transfer admission.
00	40	07E8	The macro call was not executed because the FT system only transfers individual file generations.

Table 2: Return codes for NCOPYSYN

Return code			Meaning
SC2	SC1	MC	
00	40	07E9	A DMS error occurred during the processing of the macro call. Additional information can be found in the field xxDMSCO (4 bytes).
00	40	07EA	The macro call cannot be executed because the relative file name was specified in the transfer request. The absolute file name as extended by openFT is longer than permitted.
00	40	07EB	The macro call was not executed because the local file name was not specified either explicitly or via the employed TRANSFER-ADMISSION.
00	40	07EC	The macro call was not executed because the management password of the local file is missing or incorrect.
00	40	07EE	The macro call was not executed because the local home directory was not found.
00	40	07F1	The macro call was not executed because <ul style="list-style-type: none"> – the owner of the send or receive file is not defined in the local system, or – the file owner is not the same as the user who requested the creation of a receive file.
00	40	07F2	The macro call was not executed because the password for the local send file or the receive file is missing or incorrect.
00	40	07F4	The macro call was not executed because the retention period which protects the local receive file against being overwritten (RETENTION PERIOD) has not yet expired.
00	40	07F5	The macro call cannot be executed because the local file is write-protected.
00	40	07F6	The macro call was not executed because of a file structure error.
00	40	07F7	The macro call was not executed because it is not possible to access the local file, for example because the absolute file name is too long.
00	40	07F8	The macro call was not executed because <ul style="list-style-type: none"> – the CCS name of the send and receive files cannot be mapped to one another or – the partner system does not support the retrieval of files in transparent format.
00	40	07FA	The macro call could not be executed because it is not possible to add an extension to a local file for transparent transfer.
00	40	07FB	The macro was not executed because the local send file or the receive file only permits certain access modes (e.g. read only).
00	40	07FC	The macro call could not be executed because the total of follow-up processing + prefix + suffix (from profile) is too long.
00	40	07FD	The macro call was not executed because the specifications in one of the operands of the local PROCESSING-ADMISSION is incorrect.

Table 2: Return codes for NCOPYSYN

Return code			Meaning
SC2	SC1	MC	
00	40	07FE	The macro call was not executed because the specifications in one of the operands of the local TRANSFER-ADMISSION is incorrect.
00	40	07FF	The macro call was not executed because the request was rejected by the product FTAC-BS2000 due to missing admissions.
00	40	0800	The macro call was not executed because the required function for the selected protocol is not available.
00	40	0801	The macro call was not executed because local follow-up processing is only available for the openFT protocol.
00	40	0802	The macro call cannot be executed because the partner machine does not support the data integrity check function.
00	40	0803	The macro call cannot be executed because the partner machine does not support the data encryption function.
00	40	085C	The macro call could not be executed because openFT is not authorized to process requests for this user.
00	40	085D	The macro call could not be executed because the remote directory is not empty.
00	40	085E	The macro call could not be executed because the file attributes in the remote system do not correspond to the request parameters. Possible cause: A directory was specified instead of a remote file.
00	40	085F	The properties of the remote file could not be modified as desired in the macro call. Possible causes: <ul style="list-style-type: none"> – no access rights for the file – the desired combination of access rights is not supported by the remote system – if the remote system is a BS2000 system: the file is protected by ACL
00	40	0860	The macro call cannot be executed because the file owner is not the same as the user who requested the setting up of a remote receive file.
00	40	0861	The macro call could not be executed because the CCS names of the send and receive files cannot be mapped to one another or because the partner system does not support the retrieval of files in transparent format.
00	40	0862	The macro call cannot be executed because the higher-level local directory in the partner system could not be found.

Table 2: Return codes for NCOPYSYN

Return code			Meaning
SC2	SC1	MC	
00	40	0863	The macro call was not executed. Possible causes: – The command was not executed because an existing receive file may not be set up again with WRITE-MODE=NEW or the -n switch. WRITE-MODE=NEW or -n may also have been set via a restriction in the employed transfer admission. – ftcredir: The specified directory already exists.
00	40	0864	The macro call was not executed because the FT system only transfers individual file generations.
00	40	0865	A DMS error occurred during the processing of the macro call.
00	40	0866	The macro call was not executed because the resulting remote file name is too long. Possible causes: – value assignments outside of the permitted range of values – invalid operand separator – invalid value assignment character – partially qualified file names. – BS2000: The syntax error which occurred does not relate to a 'missing operand' or 'unknown keyword'.
00	40	0868	The macro call was not executed because the remote file name was not specified either explicitly or via the employed TRANSFER-ADMISSION.
00	40	0869	The macro call was not executed because the management password of the remote file is missing or incorrect.
00	40	086B	The macro call cannot be executed because the send file is not or is no longer located in the catalog or on a volume in the remote system.
00	40	086C	The macro call was not executed because the remote home directory was not found.
00	40	086F	The macro call was not executed because – the owner of the remote send file or the receive file is not defined in the corresponding system, or – the file owner is not the same as the user who requested the creation of a receive file.
00	40	0870	The macro call cannot be executed because the password for the send file or the receive file in the remote system is missing or incorrect.
00	40	0871	The macro call cannot be executed because the retention period which protects the receive file in the remote system against being overwritten (RETENTION PERIOD) has not yet expired.
00	40	0872	The macro call cannot be executed because the remote file or remote directory is write-protected.

Table 2: Return codes for NCOPYSYN

Return code			Meaning
SC2	SC1	MC	
00	40	0873	The macro call cannot be executed because the file structure is not supported. An attempt was made, for example, to retrieve a PLAM library or ISAM file from the BS2000 system.
00	40	0874	The macro call cannot be executed because of a syntax error other than 'missing operand' or 'unknown keyword' in the remote system. Possible causes: <ul style="list-style-type: none"> – value assignments outside of the permitted range of values – invalid operand separator – invalid value assignment character – partially qualified file names
00	40	0875	The macro call could not be executed because the partner system does not support the transfer of files in transparent format.
00	40	0876	The macro call could not be executed because it is not possible to add an extension to a remote file for transparent transfer.
00	40	0877	The macro call cannot be executed because the remote system only permits certain access modes.
00	40	0878	The macro call cannot be executed because the length of remote follow-up processing has been exceeded.
00	40	0879	The macro call cannot be executed because <ul style="list-style-type: none"> – the specifications in one of the operands of the remote TRANSFER-ADMISSION are incorrect, or – the request was rejected by the FTAC because of missing transfer.
00	40	087A	The macro call cannot be executed because the required function is not supported.
00	40	087B	The macro call was not executed because the specifications in one of the operands of the remote PROCESSING-ADMISSION are incorrect.
00	80	0023	The macro call cannot be executed because the local send or receive file is already protected against simultaneous updating by another process.
00	80	0029	The macro call cannot be accepted because the maximum limit for file transfer requests has been reached.
00	80	07ED	The macro call cannot be executed because <ul style="list-style-type: none"> – the volume for the local send file or the receive file is not mounted, is unknown or occupied, or – the file extends over more than one private disk, or – an attempt was made to transfer a file migrated with HSMS.

Table 2: Return codes for NCOPYSYN

Return code			Meaning
SC2	SC1	MC	
00	80	07F0	<p>The macro call was not executed (or was discontinued) because</p> <ul style="list-style-type: none"> – the permitted storage space in the receive system for the user ID specified in TRANSFER-ADMISSION has been reached, or – the send file contains an excessively long sequence of empty blocks, or – the primary and/or secondary assignment of the password-protected receive file is too small. <p>Once this problem has occurred, the local receive file will no longer be extended or generated.</p>
00	80	0867	<p>The macro call was not executed because the remote send or receive file is already protected against simultaneous updating by another process.</p>
00	80	086A	<p>The macro call was not executed because</p> <ul style="list-style-type: none"> – the volume for the remote send file or the receive file is not mounted, is unknown or occupied, or – the file extends over more than one private disk, or – an attempt was made to transfer a file migrated with HSMS.
00	80	086E	<p>The macro call was not executed (or was discontinued) because</p> <ul style="list-style-type: none"> – the permitted storage space in the receive system for the user ID specified in TRANSFER-ADMISSION has been reached, or – the remote send file contains an excessively long sequence of empty blocks, or – the primary and/or secondary assignment of the password-protected remote receive file is too small. <p>Once this problem has occurred, the remote receive file will no longer be extended or generated.</p>
00	80	087C	<p>The macro was not executed because the maximum limit for file transfer requests has been reached.</p>

Table 2: Return codes for NCOPYSYN

2.1.11.5 Return codes for NCAN



The table below lists only the return codes for the current version. For the return codes of openFT version V9.0, please see the manual for openFT V9.0.

Return code			Meaning
SC2	SC1	MC	
00	00	0000	The request has been terminated successfully.
00	01	FFFF	In the expansion of the L-form of the macro call, a function is demanded which is not supported by FT.
00	02	FFFF	In the expansion of the L-form of the macro call, a function unit is demanded which is not supported by FT.
00	03	FFFF	In the expansion of the L-form of the macro call, a version is demanded which is not supported by FT.
00	20	0352	The macro call was not executed because <ul style="list-style-type: none"> – the system was generated without job variables, or – the maximum number of job variables for the user has been reached, or – a temporary job variable was specified, or – the specified job variable for the user is not accessible, or – the specified job variable is monitoring another FT request, or – the specified password for the job variable is incorrect.
00	20	0353	The macro call was not executed because of an internal error.
00	40	03FD	The macro call was not executed because the request has not yet been terminated with FORCE=NO.
00	40	040B	The macro call was not executed because the command can only be used by FT administrators.
00	40	040C	The macro call cannot be accepted because it can only be entered by authorized users.
00	40	040E	The macro call could not be executed because the request is terminating and can no longer be deleted.
00	40	0417	The macro call cannot be accepted because no requests were found.
00	40	08B2	The macro call cannot be accepted because the content of the specified job variable is inconsistent. Possible cause: The user made a non-read access to the job variable while it was monitoring an FT request. The content of the job variable can therefore no longer be used.
00	40	08B3	The macro call cannot be accepted because another process as well as openFT is using the job variable.
00	40	08B4	The macro call was not executed because the referenced job variable is not present.

Table 3: Return codes for NCAN

2.1.11.6 Return codes for NSTAT



The table below lists only the return codes for the current version. For the return codes of openFT version V9.0, please see the manual for openFT V9.0

Return code			Meaning
SC2	SC1	MC	
00	00	0000	The request has been terminated successfully.
00	01	FFFF	In the expansion of the L-form of the macro call, a function is demanded which is not supported by FT.
00	02	FFFF	In the expansion of the L-form of the macro call, a function unit is demanded which is not supported by FT.
00	03	FFFF	In the expansion of the L-form of the macro call, a version is demanded which is not supported by FT.
00	20	0352	The macro call was not executed because <ul style="list-style-type: none"> – the system was generated without job variables, or – the maximum number of job variables for the user has been reached, or – a temporary job variable was specified, or – the specified job variable for the user is not accessible, or – the specified job variable is monitoring another FT request, or – the specified password for the job variable is incorrect.
00	20	0353	The macro call was not executed because of an internal error.
00	40	040C	The macro call cannot be accepted because it can only be entered by authorized users.
00	40	0417	The macro call cannot be accepted because no requests were found.

Table 4: Return codes for NSTAT

2.2 The openFT-AC programming interface

All functions of openFT-AC for BS2000 can also be used via the ASSEMBLER interface. The following section is divided into three sections:

- a description of the macros
- an example
- a description of the FTAC return codes

There are ten ASSEMBLER macros to execute the functions of the ten FTAC commands. To execute the desired function, the corresponding macro can be called. In addition, there is an eleventh macro, YFSEQU, for the equates for the macro keywords.

Programs can operate in 24 or 31 bit mode. Please note that addresses in 24 bit mode must have an initial 0.

2.2.1 Macros for openFT-AC for BS2000

Macro	Default Prefix	Function	command
CREFTPRF	YCP	create admission profile	CREATE-FT-PROFILE
DELFTPRF	YDP	delete admission profile	DELETE-FT-PROFILE
EXPFTENV	YEE	export admission sets and profiles	EXPORT-FTAC-ENVIRONMENT
IMPPFTENV	YIE	import admission sets and profiles	IMPORT-FTAC-ENVIRONMENT
MODFTADS	YMA	modify admission sets	MODIFY-FT-ADMISSION-SET
MODFTPRF	YMP	modify admission profiles	MODIFY-FT-PROFILE
SHWFTADS	YSA	display admission sets	SHOW-FT-ADMISSION-SET
SHWFTENV	YSE	list admission sets and profiles	SHOW-FTAC-ENVIRONMENT
SHWFTPRF	YSP	display admission profile	SHOW-FT-PROFILE
SHWFTRGE	YSR	list accessible partner systems	SHOW-FT-RANGE

2.2.2 CREFTPRF - Create admission profile

The macro CREFTPRF can be used to create admission profiles. The functionality corresponds to that of the command CREATE-FT-PROFILE.

Format of the macro CREFTPRF (LIST format/default format)

Name	Operation	Operand
[Name]	CREFTPRF	<pre> [MF=L] ,NAME=adr [,PASS= { *NONE { adr }] ,TAD= { *NOTSP { adr [,VALID= { [*]YES { [*]NO }] { [,USAGE= { [*]PRIVATE { [*]PUBLIC }] { [,EXPDATE= { *NOTRS { adr¹ }] }] [,PRIV= { [*]NO } { [*]YES }] [,ILV= { [*]NO } / { [*]YES }] [,IOSND= { [*]NO } { [*]YES }] [,IORCV= { [*]NO } { [*]YES }] [,IISND= { [*]NO } { [*]YES }] </pre>

¹ The date output must be in the form YYYYMMDD. The date must lie between 01.01.1970 and 31.12.2020.

Name	Operation	Operand
		[, IIRCV= { [*]NO [*]YES }]
		[, IIPRC= { [*]NO [*]YES }]
		[, IIMAN= { [*]NO [*]YES }]]
		[, UAD=[*]OWN /
		[, UUSER= { *OWN adr }]
		,UACC= { *OWN *FIRST *NONE *NOTSP adr }
		[, UPASS= { *OWN *NONE *NOTSP adr }]]]
		[, INIT= { [*]REMOTE [*]LOCAL [*]BOTH }]
		[, TRANS= { [*]NOTRS [*]TO [*]FROM }]
		[, PARTNER= { *NOTRS {adr1, ..., adr50} }]
		[, MAXPLV= { *NOTRS integer0-100 }]
		[, FILE= { *NOTRS adr } / PFXFIL=adr /
		[, LIB= { *NOTRS adr } / PFXLIB=adr]

Name	Operation	Operand
		$[,EL= \left\{ \begin{array}{l} *NOTRS \\ \text{adr} \end{array} \right\} / \text{PFXEL}=\text{adr}$
		$[,ELVER= \left\{ \begin{array}{l} *STD \\ \text{adr} \end{array} \right\}]]$
		$[,ELTYP= \left\{ \begin{array}{l} *NOTRS \\ \text{adr} \end{array} \right\}]]$
		$[,FPASS= \left\{ \begin{array}{l} *NOTRS \\ *NONE \\ \text{adr} \end{array} \right\}]$
		$[,PAD= \left\{ \begin{array}{l} [*]SAME \\ [*]NOTRS \end{array} \right\} /$
		$[,PUSER= \left\{ \begin{array}{l} *SAME \\ *NOTRS \\ \text{adr} \end{array} \right\}$
		$,PACC= \left\{ \begin{array}{l} *SAME \\ *NONE \\ *NOTRS \\ \text{adr} \end{array} \right\}$
		$[,PPASS= \left\{ \begin{array}{l} *SAME \\ *NOTRS \\ *NONE \\ \text{adr} \end{array} \right\}]]$
		$[,SUCC= \left\{ \begin{array}{l} *NOTRS \\ *NONE \\ \text{adr} \end{array} \right\} /$
		$[,PFXSUCC= \left\{ \begin{array}{l} *NOTRS \\ \text{adr} \end{array} \right\}]$
		$[,SUXSUCC= \left\{ \begin{array}{l} *NOTRS \\ \text{adr} \end{array} \right\}]]$
		$[,FAIL= \left\{ \begin{array}{l} *NOTRS \\ *NONE \\ \text{adr} \end{array} \right\} /$
		$[,PFXFAIL= \left\{ \begin{array}{l} *NOTRS \\ \text{adr} \end{array} \right\}]$

Name	Operation	Operand
		$[,SUXFAIL= \left\{ \begin{array}{l} *NOTRS \\ \text{adr} \end{array} \right\}]]$
		$[,WRITE= \left\{ \begin{array}{l} [*]NOTRS \\ [*]NEW \\ [*]RPL \\ [*]EXT \end{array} \right\}]$
		$[,FTFUNC= \left\{ \begin{array}{l} [*]NOTRS \\ ([TRANSF] \\ [,MODATT] \\ [,READDIR] \\ [,FILEPROC]) \end{array} \right\}]$
		$[,TEXT= \left\{ \begin{array}{l} *NONE \\ \text{adr} \end{array} \right\}]$
		$[,CHIP= \left\{ \begin{array}{l} [*]NO \\ [*]YES \end{array} \right\}]$
		$[,DENC= \left\{ \begin{array}{l} [*]NOTRS \\ [*]YES \\ [*]NO \end{array} \right\}]$

Format of the macro CREFTPRF (DSECT format)

Name	Operation	Operand
[name]	CREFTPRF	MF=(D[,xxx])

```

XXXFTPRF DSECT
XXXHPRF  FHDR  MF=(C,&P)
XXXNAME  DS    A          A(NAME)
XXXNAMEL DS   XL2        L(NAME)
XXXPASSL DS   XL2        L(PASSWORD)
XXXPASS  DS    A          A(PASSWORD)
XXXTAD   DS    A          A(TRANSFER-ADMISSION)
XXXTADL  DS   XL2        L(TRANSFER-ADMISSION)
XXXUUSL  DS   XL2        L(USER-ID/USER-ADMISSION)
XXXUUS   DS    A          A(USER-ID/USER-ADMISSION)
XXXUAC   DS    A          A(USER-ACC/USER-ADMISSION)
XXXUACL  DS   XL2        L(USER-ACC/USER-ADMISSION)
XXXUPAL  DS   XL2        L(USER-PASSWORD/USER-ADMISSION)
XXXUPA   DS    A          A(USER-PASSWORD/USER-ADMISSION)
XXXPART  DS    A          A(PARTNER-NAME)
XXXPARTL DS   XL2        L(PARTNER-NAME)
XXXFILL  DS   XL2        L(FILE / -PREFIX / LIB / -PREFIX)
XXXFIL   DS    A          A(FILE / -PREFIX / LIB / -PREFIX)
XXXEL    DS    A          A(ELEMENT / -PREFIX)
XXXELL   DS   XL2        L(ELEMENT / -PREFIX)
XXXEVL   DS   XL2        L(ELEMENT-VERSION)
XXXEV    DS    A          A(ELEMENT-VERSION)
XXXETY   DS    A          A(ELEMENT-TYPE)
XXXETYL  DS   XL2        L(ELEMENT-TYPE)
XXXFPAL  DS   XL2        L(FILE-PASSWORD)
XXXFPA   DS    A          A(FILE-PASSWORD)
XXXPUS   DS    A          A(USER-ID/PROCESSING-ADMISSION)
XXXPUSL  DS   XL2        L(USER-ID/PROCESSING-ADMISSION)
XXXPACL  DS   XL2        L(USER-ACC/PROCESSING-ADMISSION)
XXXPAC   DS    A          A(USER-ACC/PROCESSING-ADMISSION)
XXXPPA   DS    A          A(USER-PASSWORD/PROCESSING-ADM)
XXXPPAL  DS   XL2        L(USER-PASSWORD/PROCESSING-ADM)
XXXSUCL  DS   XL2        L(SUCCESS-PROCESSING)
XXXSUC   DS    A          A(SUCCESS-PROCESSING)
XXXFAI   DS    A          A(FAILURE-PROCESSING)
XXXFAIL  DS   XL2        L(FAILURE-PROCESSING)

```

```

*
* 1-BYTE-FIELDS FOR ENTERING KEYWORDS *
*
XXXIPASS DS    XL1          PASSWORD: NONE
*
XXXPRIV  DS    XL1          PRIVILEGED: NO / YES
*
XXXILV   DS    XL1          IGNORE-MAX-LEVELS:  NO / YES
XXXIOSND DS    XL1          OUTBOUND-SEND:          NO / YES
XXXIORCV DS    XL1          OUTBOUND-RECEIVE:       NO / YES
XXXIISND DS    XL1          INBOUND-SEND:           NO / YES
XXXIIRCV DS    XL1          INBOUND-RECEIVE:       NO / YES
XXXIIPRC DS    XL1          INBOUND-PROCESSING:    NO / YES
*
XXXUAD   DS    XL1          USER-ADMISSION:  OWN
XXXIUUS  DS    XL1          USER-ID:        OWN
XXXIUAC  DS    XL1          USER-ACC:       OWN / FIRST / NOTSP
XXXIUAPA DS    XL1          USER-PASSWORD:  OWN / NONE / NOTSP
*
XXXINIT  DS    XL1          INITIATOR:     REM / LOC / BOTH
XXXTRDIR DS    XL1          TRANSFER DIRECTION: FROM / TO / NOTRS
*
XXXIPART DS    XL1          PARTNER-NAME:  NOTRS
XXXMLPV  DS    XL1          MAX-PARTNER-LEVEL: NOTRS / 0-100
*
XXXIFIL  DS    XL1          FILE-NAME:     NOTRS / EXP / LIBEL
XXXILIB  DS    XL1          LIBRARY-NAME:  NOTRS / EXP
XXXIEL   DS    XL1          ELEMENT-NAME:  NOTRS / EXP
XXXIEV   DS    XL1          ELEMENT-VERSION: STD
XXXIETY  DS    XL1          ELEMENT-TYPE:  NOTRS
XXXIFPA  DS    XL1          FILE-PASSWORD: NOTRS / NONE
*
XXXPAD   DS    XL1          PROCESSING-ADMISSION: SAME / NOTRS
XXXIPUS  DS    XL1          USER-ID:      SAME / NOTRS
XXXIPAC  DS    XL1          USER-ACC:     SAME / NOTRS
XXXIPPA  DS    XL1          USER-PASSWORD: SAME / NOTRS / NONE
*
XXXISUC  DS    XL1          SUCCESS-PROCESSING: NOTRS / NONE / EXP
XXXIFAI  DS    XL1          FAILURE-PROCESSING: NOTRS / NONE / EXP
*
XXXWRMOD DS    XL1          WRITE-MODE:   NOTRS / NEW / RPL / EXT
*
XXXCHIP  DS    XL1          CHIP-PROFILE: NO / YES

```



```

*
* VRS_CPRF2
*
XXXEXDA DS A A(EXPIRATION-DATE)
XXXEXDAL DS XL2 L(EXPIRATION-DATE)
XXXPSUCL DS XL2 L(SUCCESS-PROCESSING-PREFIX)
XXXPSUC DS A A(SUCCESS-PROCESSING-PREFIX)
XXXSUC DS A A(SUCCESS-PROCESSING-SUFFIX)
XXXSUCL DS XL2 L(SUCCESS-PROCESSING-SUFFIX)
XXXPFAIL DS XL2 L(FAILURE-PROCESSING-PREFIX)
XXXPFAI DS A A(FAILURE-PROCESSING-PREFIX)
XXXFAI DS A A(FAILURE-PROCESSING-SUFFIX)
XXXFAIL DS XL2 L(FAILURE-PROCESSING-SUFFIX)
XXXTEXTL DS XL2 L(TEXT)
XXXTEXT DS A A(TEXT)
XXXAPRTL DS 50A A(A-PARTNER-NAME)
XXXAPRT DS 50XL2 L(A-PARTNER-NAME)
*
XXXFTFUN DS XL2 FT-FUNCTION: NOTRS /
* [TRANF]+[MODA]+[READ]+[FPRO]
XXXIIMAN DS XL1 INBOUND-MANAGEMENT: NO / YES
XXXITEXT DS XL1 TEXT: NONE
*
XXXITAD DS XL1 TRANSFER-ADMISSION: NOTSP
XXXVALID DS XL1 VALIDATE: YES / NO
XXXUSAGE DS XL1 USAGE: PRIV / PUBL
XXXIEXDA DS XL1 EXPIRATION-DATE: NOTRS
*
XXXIPSUC DS XL1 SUCCESS-PROCESSING-PREFIX: NOTRS
XXXIXSUC DS XL1 SUCCESS-PROCESSING-SUFFIX: NOTRS
XXXIPFAI DS XL1 FAILURE-PROCESSING-PREFIX: NOTRS
XXXIXFAI DS XL1 FAILURE-PROCESSING-SUFFIX: NOTRS
*
* VRS_CPRF4
*
XXXDENC DS XL1 DATA-ENCRYPTION: YES / NO / NOTRS
*
XXXFTPRL EQU *-XXXFTPRF

```

The meaning of the parameters in the macro CREFTPRF corresponds to the meaning of the parameters in the command CREATE-FT-PROFILE as shown in the following table.

Operand (L format)	Address field (D format)	Meaning in command
NAME	NAME	NAME
PASS	PASS	PASSWORD
TAD	TAD	TRANSFER-ADMISSION
PRIV	PRIV	PRIVILEGED
ILV	ILV	IGNORE-MAX-LEVELS
IOSND	IOSND	OUTBOUND-SEND in the structure IGNORE-MAX-LEVELS
IORCV	IORCV	OUTBOUND-RECEIVE in the structure IGNORE-MAX-LEVELS
IISND	IISND	INBOUND-SEND in the structure IGNORE-MAX-LEVELS
IIRCV	IIRCV	INBOUND-RECEIVE in the structure IGNORE-MAX-LEVELS
IIPRC	IIPRC	INBOUND-PROCESSING in the structure IGNORE-MAX-LEVELS
UAD	UAD	USER-ADMISSION
UUSER	UUS	USER-IDENTIFICATION in the USER-ADMISSION
UACC	UAC	USER-ACCOUNT in the USER-ADMISSION
UPASS	UPA	USER-PASSWORD in the USER-ADMISSION
INIT	INIT	INITIATOR
TRANS	TRDIR	TRANSFER-DIRECTION
PARTNER	PART	PARTNER-NAME
MAXPLV	MPLV	MAX-PARTNER-LEVEL
FILE	FIL	FILE-NAME
LIB	LIB	LIBRARY
EL	EL	ELEMENT
ELVER	EV	ELEMENT-VERSION
ELTYP	ETY	ELEMENT-TYPE
FPASS	FPA	FILE-PASSWORD
PAD	PAD	PROCESSING-ADMISSION
PUSER	PUS	USER-IDENTIFICATION in the PROCESSING-ADMISSION
PACC	PAC	USER-ACCOUNT in the PROCESSING-ADMISSION
PPASS	PPA	USER-PASSWORD in the PROCESSING-ADMISSION
SUCC	SUC	SUCCESS-PROCESSING
PFXSUCC	PSUC	PREFIX in the structure SUCCESS-PROCESSING

Operand (L format)	Address field (D format)	Meaning in command
SUCSUCC	XSUC	SUFFIX in the structure SUCCESS-PROCESSING
FAIL	FAI	FAILURE-PROCESSING
PFXFAIL	PFAI	PREFIX in the structure FAILURE-PROCESSING
SUCFAIL	XFAI	SUFFIX in the structure FAILURE-PROCESSING
WRITE	WRMOD	WRITE-MODE
FTFUNC	FTFUN	FT-FUNCTION
TEXT	TEXT	USER-INFORMATION
CHIP	CHIP	CHIPCARD in the structure TRANSFER-ADMISSION
DENC	DENC	DATA-ENCRYPTION

If a chipcard profile is created at the command interface (TRANSFER-ADMISSION=*CHIPCARD(TRANSFER-ADMISSION=,CERTIFICATE=)), then the internally stored transfer admission is formed from the specified transfer admission together with the certificate number and certification authority.

At the programming interface it is not possible to specify these components of the transfer admission in separate fields. Instead, it is the responsibility of the caller of the programming interface to structure the transfer admission correctly. The structure of the transfer admission for specification is as follows:

4 bytes	Number of the Certification Authority configured in the system
4 bytes	Certificate number
0-7 bytes	Transfer admission specified in the partner system. If this component of the transfer admission is empty then this corresponds to the specification TRANSFER-ADMISSION=*NONE at the command interface.

2.2.3 DELFTPRF - Delete admission profile

The macro DELFTPRF can be used to delete admission profiles. The functionality corresponds to that of the command DELETE-FT-PROFILE.

Format of the macro DELFTPRF (LIST format/default format)

Name	Operation	Operand
[Name]	DELFTPRF	<pre> [MF=L] ,NAME= { *ALL { adr } [,PASS= { *NONE { adr }] [,SELECT=[*]OWN / [,TAD= { *ALL { *NOTSP } { adr }] [,OWNER= { *OWN { *ALL } { adr }] </pre>

Format of the macro DELFTPRF (DSECT format)

```

xxxFTPRF DSECT
xxxHPRF FHDR MF=(C,&P)
xxxNAME DS A A(NAME)
xxxNAMEL DS XL2 L(NAME)
xxxPASSL DS XL2 L(PASSWORD)
xxxPASS DS A A(PASSWORD)
xxxTAD DS A A(SELECT-TRANSFER-ADMISSION)
xxxTADL DS XL2 L(SELECT-TRANSFER-ADMISSION)
xxxOWNL DS XL2 L(SELECT-OWNERID)
xxxOWN DS A A(SELECT-OWNERID)
*
* 1-BYTE-FIELDS FOR ENTERING KEYWORDS *
*
xxxINAME DS XL1 NAME: ALL
xxxIPASS DS XL1 PASSWORD: NONE
*
xxxSEL DS XL1 SELECT: OWN
xxxITAD DS XL1 SELECT-TRANSFER-ADMISSION: ALL / NOTSP
xxxIOWN DS XL1 SELECT-OWNERID: OWN / ALL
xxxRESRV DS XL3 RESERVED
*
xxxFTPRL EQU *-xxxFTPRF

```

The meaning of the parameters in the macro DELFTPRF corresponds to the meaning of the parameters in the command DELETE-FT-PROFILE as shown in the following table.

Operand (L format)	Address field (D format)	Meaning in command
NAME	NAME	NAME
PASS	PASS	PASSWORD
SELECT	SEL	SELECT-PARAMETER
TAD	TAD	TRANSFER-ADMISSION in the structure SELECT-PARAMETER
OWNER	OWN	OWNER-IDENTIFICATION in the structure SELECT-PARAMETER

2.2.4 EXPFTENV - Export admission sets and profiles

The macro EXPFTENV can be used to output the admission sets and admission profiles of FTAC to a file. You can use the SHWFTENV macro to display the contents of this file ([page 121](#)). The functionality corresponds to that of the command EXPORT-FTAC-ENVIRONMENT.

Format of the macro EXPFTENV (LIST format/default format)

Name	Operation	Operand
[Name]	EXPFTENV	<pre>[MF=L] ,FILE=adr [,USER= { *ALL {adr1,...,adr100} }] [,SELECT= [*]ALL / [,NAME= { *ALL *NONE {adr1,...,adr100} }]] [,ADMS= { [*]YES [*]NO }]]]</pre>

Format of the macro EXPFTENV (DSECT format)

```

xxxFTENV DSECT
xxxHENV  FHDR  MF=(C,&P)
xxxFILE  DS   A           A(FILE)
xxxFILEL DS  XL2         L(FILE)
xxxRES   DS   XL2         RESERVED
xxxUSERL DS  100XL2      L(A-USER)
xxxUSER  DS   100A        A(A-USER)
xxxNAME  DS   100A        A(A-NAME)
xxxNAMEL DS  100XL2      L(A-NAME)
*
* 1-BYTE FIELDS FOR ENTERING KEYWORDS *
*
xxxSEL   DS   XL1         SELECT: ALL
*
xxxIUSER DS  XL1         USER: ALL
*
xxxINAME DS  XL1         PROFIL-NAME: ALL / NONE
*
xxxADMS  DS   XL1         ADMISSION-SET: YES / NO
*
xxxFTENL EQU  *-xxxFTENV

```

The meaning of the parameters in the macro EXPFTENV corresponds to the meaning of the parameters in the command EXPORT-FTAC-ENVIRONMENT as shown in the following table.

Operand (L format)	Address field (D format)	Meaning in command
FILE	FILE	TO-FILE
USER	USER	USER-IDENTIFICATION
SELECT	SEL	SELECT-PARAMETER
NAME	NAME	PROFILE-NAME in the structure SELECT-PARAMETER
ADMS	ADMS	ADMISSION-SET in the structure SELECT-PARAMETER

2.2.5 IMPFTENV - Import admission sets and profiles

The macro IMPFTENV can be used to import the admission sets and profiles from a file. The functionality corresponds to that of the command IMPORT-FTAC-ENVIRONMENT.

Format of the macro IMPFTENV (LIST format/default format)

Name	Operation	Operand
[Name]	IMPFTENV	<pre>[MF=L] ,FILE=adr [,USER= { *ALL }] { adr1,...,adr100) } [,SELECT= [*]ALL / [,NAME= { *ALL *NONE { adr1,...,adr100) }] [,ADMS= { [*]YES }]] [*]NO)</pre>

Format of the macro IMPFTENV (DSECT format)

```

xxxFTENV DSECT
xxxHENV  FHDR  MF=(C,&P)
xxxFILE  DS    A          A(FILE)
xxxFILEL DS   XL2        L(FILE)
xxxRES   DS   XL2        RESERVED
xxxUSERL DS   100XL2     L(A-USER)
xxxUSER  DS   100A       A(A-USER)
xxxNAME  DS   100A       A(A-NAME)
xxxNAMEL DS   100XL2     L(A-NAME)
*
* 1-BYTE FIELDS FOR ENTERING KEYWORDS *
*
xxxSEL   DS   XL1        SELECT: ALL
*
xxxIUSER DS   XL1        USER: ALL
*
xxxINAME DS   XL1        PROFIL-NAME: ALL / NONE
*
xxxADMS  DS   XL1        ADMISSION-SET: YES / NO
*
xxxFTENL EQU  *-xxxFTENV

```

The meaning of the parameters in the macro IMPFTENV corresponds to the meaning of the parameters in the command IMPORT-FTAC-ENVIRONMENT as shown in the following table.

Operand (L format)	Address field (D format)	Meaning in command
FILE	FILE	TO-FILE
USER	USER	USER-IDENTIFICATION
SELECT	SEL	SELECT-PARAMETER
NAME	NAME	PROFILE-NAME in the structure SELECT-PARAMETER
ADMS	ADMS	ADMISSION-SET in the structure SELECT-PARAMETER

2.2.6 MODFTADS - Modify admission sets

The macro MODFTADS can be used modify admission sets. The functionality corresponds to that of the command MODIFY-FT-ADMISSION-SET.

Format of the macro MODFTADS (LIST format/default format)

Name	Operation	Operand
[Name]	MODFTADS	<pre> [MF=L] [,USER= { *OWN *STD }] [,PASS= { *NONE adr }] [,NEWPASS= { *OLD *NONE } adr] [,PRIV= { *UNCHG *YES }] [,MAXLV= { *UNCHG *STD integer0-100 } / [,MAXOSND= { *UNCHG *STD integer0-100 }] [,MAXORCV= { *UNCHG *STD integer0-100 }] [,MAXISND= { *UNCHG *STD integer0-100 }] [,MAXIRCV= { *UNCHG *STD integer0-100 }] [,MAXIPRC= { *UNCHG *STD integer0-100 }]] [,MAXIMAN= { *UNCHG *STD integer0-100 }]] </pre>

Format of the macro MODFTADS (DSECT format)

```

xxxFTADS DSECT
xxxHADS FHDR MF=(C,&P)
xxxUSER DS A A(USERID)
xxxUSERL DS XL2 L(USERID)
xxxPASSL DS XL2 L(PASS)
xxxPASS DS A A(PASS)
xxxNPA DS A A(NEW-PASS)
xxxNPAL DS XL2 L(NEW-PASS)
*
* 1-BYTE FIELDS FOR ENTERING KEYWORDS *
*
xxxIUSER DS XL1 USERID: OWN / STD
xxxIPASS DS XL1 PASSWORD: NONE
xxxSEL DS XL1 SELECT: ALL
xxxINPA DS XL1 NEW-PASSWORD: NONE / OLD
xxxPRIV DS XL1 PRIVILEGED: UNCHG / YES
*
xxxMOSND DS XL1 OUTBOUND-SEND: UNCHG/STD/0-100
xxxMORCV DS XL1 OUTBOUND-RECEIVE: UNCHG/STD/0-100
xxxMISND DS XL1 INBOUND-SEND: UNCHG/STD/0-100
xxxMIRCV DS XL1 INBOUND-RECEIVE: UNCHG/STD/0-100
xxxMIPRC DS XL1 INBOUND-PROCESSING: UNCHG/STD/0-100
*
* VRS_MADS2
*
xxxMIMAN DS XL1 INBOUND-MANAGEMENT: UNCHG/STD/0-100
xxxRESRV DS XL3 RESERVED
*
xxxFTADL EQU *-xxxFTADS

```

The meaning of the parameters in the macro MODFTADS corresponds to the meaning of the parameters in the command MODIFY-FT-ADMISSION-SET as shown in the following table.

Operand (L format)	Address field (D format)	Meaning in command
USER	USER	USER-IDENTIFICATION
PASS	PASS	PASSWORD
NEWPASS	NPA	NEW-PASSWORD
PRIV	PRIV	PRIVILEGED
MAXLV		MAX-LEVELS
MAXOSND	MOSND	OUTBOUND-SEND in the structure MAX-LEVELS
MAXORCV	MORCV	OUTBOUND-RECEIVE in the structure MAX-LEVELS
MAXISND	MISND	INBOUND-SEND in the structure MAX-LEVELS
MAXIRCV	MIRCV	INBOUND-RECEIVE in the structure MAX-LEVELS
MAXIPRC	MIPRC	INBOUND-PROCESSING in the structure MAX-LEVELS
MAXIMAN	MIMAN	INBOUND-MANAGEMENT in the structure MAX-LEVELS

2.2.7 MODFTPRF - Modify admission profile

The macro MODFTPRF can be used to modify admission profiles. The functionality corresponds to that of the command MODIFY-FT-PROFILE.

Format of the macro MODFTPRF (LIST format/default format)

Name	Operation	Operand
[Name]	MODFTPRF	<pre> [MF=L] ,NAME= { *ALL adr } [,PASS= { *NONE adr }] [,SELECT=[*]OWN / [,TAD= { *ALL *NOTSP adr }] [,OWNER= { *OWN *ALL adr }] [,NEWNAME= { *OLD adr }] [,NEWTAD= { *UNCHG adr }] [,VALID= { [*]UNCHG [*]YES [*]NO }] [,USAGE= { [*]UNCHG [*]PRIVATE [*]PUBLIC }] [,EXPDATE= { *UNCHG *NOTRS adr }] [,VALID= { [*]YES [*]NO }] [,USAGE= { [*]PRIVATE [*]PUBLIC }] [,EXPDATE= { *NOTRS adr¹ }] </pre>

¹ The date output must be in the form YYYYMMDD. The date must lie between 01.01.1970 and 31.12.2020.

Name	Operation	Operand
		[, PRIV= { *UNCHG *NO *YES }]
		[, ILV= { *UNCHG *NO *YES } /
		[, IOSND= { *UNCHG *NO *YES }]
		[, IORCV= { *UNCHG *NO *YES }]
		[, IISND= { *UNCHG *NO *YES }]
		[, IIRCV= { *UNCHG *NO *YES }]
		[, IIPRC= { *UNCHG *NO *YES }]]
		[, IIMAN= { *UNCHG *NO *YES }]]
		[, UAD= { *UNCHG *OWN } /
		[, UUSER= { *OWN adr }]
		, UACC= { *OWN *FIRST *NONE *NOTSP adr }]
		[, UPASS= { *OWN *NONE *NOTSP adr }]]]

Name	Operation	Operand
		[,INIT= { *UNCHG *REMOTE *LOCAL *BOTH }]
		[,TRANS= { *UNCHG *FROM *TO *NOTRS }]
		[,PARTNER= { *UNCHG *NOTRS } adr]
		,ADDPART=(adr1,...,adr50)/
		,REMPART=(adr1,...,adr50)]
		[,MAXPLV= { *UNCHG *NOTRS integer0-100 }]
		[,FILE= { *UNCHG *NOTRS } / PFXFIL=adr / adr]
		[,LIB= { *UNCHG *NOTRS } / PFXLIB=adr adr]
		[,EL= { *UNCHG *NOTRS } / PFXEL=adr adr]
		[,ELVER= { *STD adr }]]
		[,ELTYP= { *UNCHG *NOTRS } adr]]
		[,FPASS= { *UNCHG *NOTRS *NONE adr }]
		[,PAD= { *UNCHG *SAME *NOTRS } /
		[,PUSER= { *SAME *NOTRS adr }]

Name	Operation	Operand
		,PACC= $\left\{ \begin{array}{l} *SAME \\ *NONE \\ *NOTRS \\ \text{adr} \end{array} \right\}$
		[,PPASS= $\left\{ \begin{array}{l} *SAME \\ *NOTRS \\ *NONE \\ \text{adr} \end{array} \right\}]]]$
		[,SUCC= $\left\{ \begin{array}{l} *UNCHG \\ *NOTRS \\ *NONE \\ \text{adr} \end{array} \right\} /$
		[,PFXSUCC= $\left\{ \begin{array}{l} *UNCHG \\ *NOTRS \\ \text{adr} \end{array} \right\}]$
		[,SUXSUCC= $\left\{ \begin{array}{l} *UNCHG \\ *NOTRS \\ \text{adr} \end{array} \right\}]]$
		[,FAIL= $\left\{ \begin{array}{l} *UNCHG \\ *NOTRS \\ *NONE \\ \text{adr} \end{array} \right\} /$
		[,PFXFAIL= $\left\{ \begin{array}{l} *UNCHG \\ *NOTRS \\ \text{adr} \end{array} \right\}]$
		[,SUXFAIL= $\left\{ \begin{array}{l} *UNCHG \\ *NOTRS \\ \text{adr} \end{array} \right\}]]$
		[,WRITE= $\left\{ \begin{array}{l} *UNCHG \\ *NOTRS \\ *NEW \\ *RPL \\ *EXT \end{array} \right\}]$
		[,FTFUNC= $\left\{ \begin{array}{l} [*]UNCHG \\ [*]NOTRS \\ ([TRANSF] \\ [,MODATT] \\ [,READDIR] \\ [,FILEPROC]) \end{array} \right\}]$
		[,TEXT= $\left\{ \begin{array}{l} *UNCHG \\ *NONE \\ \text{adr} \end{array} \right\}]$
		[,CHIP= $\left\{ \begin{array}{l} [*]NO \\ [*]YES \end{array} \right\}$

Name	Operation	Operand
		[,DENC= { [*]UNCHG [*]NOTRS [*]YES [*]NO }]

Format of the macro MODFTPRF (DSECT format)

```

XXXFTPRF DSECT
XXXHPRF FHDR MF=(C,&P)
XXXNAME DS A A(NAME)
XXXNAMEL DS XL2 L(NAME)
XXXPASSL DS XL2 L(PASSWORD)
XXXPASS DS A A(PASSWORD)
XXXTAD DS A A(SELECT-TRANSFER-ADMISSION)
XXXTADL DS XL2 L(SELECT-TRANSFER-ADMISSION)
XXXOWNL DS XL2 L(SELECT-OWNERID)
XXXOWN DS A A(SELECT-OWNERID)
XXXNNAM DS A A(NEW-NAME)
XXXNNAML DS XL2 L(NEW-NAME)
XXXNTADL DS XL2 L(TRANSFER-ADMISSION)
XXXNTAD DS A A(TRANSFER-ADMISSION)
XXXUUS DS A A(USER-ID/USER-ADMISSION)
XXXUUSL DS XL2 L(USER-ID/USER-ADMISSION)
XXXUACL DS XL2 L(USER-ACC/USER-ADMISSION)
XXXUAC DS A A(USER-ACC/USER-ADMISSION)
XXXUPA DS A A(USER-PASSWORD/USER-ADMISSION)
XXXUPAL DS XL2 L(USER-PASSWORD/USER-ADMISSION)
XXXPARTL DS XL2 L(PARTNER-NAME)
XXXPART DS A A(PARTNER-NAME)
XXXFIL DS A A(FILE / -PREFIX / LIB / -PREFIX)
XXXFILL DS XL2 L(FILE / -PREFIX / LIB / -PREFIX)
XXXELL DS XL2 L(ELEMENT / -PREFIX)
XXXEL DS A A(ELEMENT / -PREFIX)
XXXEV DS A A(ELEMENT-VERSION)
XXXEVL DS XL2 L(ELEMENT-VERSION)
XXXETYL DS XL2 L(ELEMENT-TYPE)
XXXETY DS A A(ELEMENT-TYPE)
XXXFPA DS A A(FILE-PASSWORD)
XXXFPAL DS XL2 L(FILE-PASSWORD)
XXXPUSL DS XL2 L(USER-ID/PROCESSING-ADMISSION)
XXXPUS DS A A(USER-ID/PROCESSING-ADMISSION)
XXXPAC DS A A(USER-ACC/PROCESSING-ADMISSION)
XXXPACL DS XL2 L(USER-ACC/PROCESSING-ADMISSION)
XXXPPAL DS XL2 L(USER-PASSWORD/PROCESSING-ADM)
XXXPPA DS A A(USER-PASSWORD/PROCESSING-ADM)

```

```

XXXSUC  DS    A           A(SUCCESS-PROCESSING)
XXXSUCL DS    XL2        L(SUCCESS-PROCESSING)
XXXFAIL DS    XL2        L(FAILURE-PROCESSING)
XXXFAI  DS    A           A(FAILURE-PROCESSING)
*
* 1-BYTE-FIELDS FOR ENTERING KEYWORDS *
*
XXXINAME DS    XL1        NAME: ALL
XXXIPASS DS    XL1        PASSWORD: NONE
*
XXXSEL   DS    XL1        SELECT: OWN
XXXITAD  DS    XL1        TRANSFER-ADMISSION: ALL / NOTSP
XXXIOWN  DS    XL1        OWNERID: OWN / ALL
*
XXXINNAM DS    XL1        NEW NAME: OLD
XXXINTAD DS    XL1        TRANSFER-ADMISSION: UNCHG/NOTSP
*
XXXPRIV  DS    XL1        PRIVILEGED: UNCHG / NO / YES
*
XXXILV   DS    XL1        IGNORE-MAX-LEVELS:  UNCHG / NO / YES
XXXIOSND DS    XL1        OUTBOUND-SEND:      UNCHG / NO / YES
XXXIORCV DS    XL1        OUTBOUND-RECEIVE:    UNCHG / NO / YES
XXXIISND DS    XL1        INBOUND-SEND:       UNCHG / NO / YES
XXXIIRCV DS    XL1        INBOUND-RECEIVE:    UNCHG / NO / YES
XXXIIPRC DS    XL1        INBOUND-PROCESSING: UNCHG / NO / YES
*
XXXUAD   DS    XL1        USER-ADMISSION: UNCHG / OWN
XXXIUUS  DS    XL1        USER-ID: OWN
XXXIUAC  DS    XL1        USER-ACC: OWN / FIRST / NOTSP
XXXIUPA  DS    XL1        USER-PASSWORD: OWN / NONE / NOTSP
*
XXXINIT  DS    XL1        INITIATOR: UNCHG / REM / LOC / BOTH
XXXTRDIR DS    XL1        TRANSFER DIRECTION:UNCHG/FROM/TO/NOTRS
*
XXXIPART DS    XL1        PARTNER-NAME: UNCHG/NOTRS/ADD/REMOV
XXXMLPV  DS    XL1        MAX-PARTNER-LEVEL: UNCHG / NOTRS / 0-100
*
XXXIFIL  DS    XL1        FILE-NAME: UNCHG / NOTRS / EXP / LIBEL
XXXILIB  DS    XL1        LIBRARY-NAME: UNCHG / NOTRS / EXP
XXXIEL   DS    XL1        ELEMENT-NAME: UNCHG / NOTRS / EXP
XXXIEV   DS    XL1        ELEMENT-VERSION: STD
XXXIETY  DS    XL1        ELEMENT-TYPE: UNCHG / NOTRS
XXXIFPA  DS    XL1        FILE-PASSWORD: UNCHG / NOTRS / NONE
*
XXXPAD   DS    XL1        PROCESSING-ADMISSION: UNCHG / SAME / NOTRS
XXXIPUS  DS    XL1        USER-ID: SAME / NOTRS
XXXIPAC  DS    XL1        USER-ACC: SAME / NOTRS
XXXIPPA  DS    XL1        USER-PASSWORD: SAME / NOTRS / NONE

```

```

*
XXXISUC DS XL1 SUCCESS-PROCESSING: UNCHG/NOTRS/NONE/EXP
XXXIFAI DS XL1 FAILURE-PROCESSING: UNCHG/NOTRS/NONE/EXP
*
XXXWRMOD DS XL1 WRITE-MODE: UNCHG/NOTRS/NEW/RPL/EXT
*
XXXCHIP DS XL1 CHIP-PROFILE: NO / YES
*
* VRS_MPRF2
*
XXXEXDA DS A A(EXPIRATION-DATE)
XXXEXDAL DS XL2 L(EXPIRATION-DATE)
XXXPSUCL DS XL2 L(SUCCESS-PROCESSING-PREFIX)
XXXPSUC DS A A(SUCCESS-PROCESSING-PREFIX)
XXXXSUC DS A A(SUCCESS-PROCESSING-SUFFIX)
XXXXSUCL DS XL2 L(SUCCESS-PROCESSING-SUFFIX)
XXXPFAIL DS XL2 L(FAILURE-PROCESSING-PREFIX)
XXXPFAI DS A A(FAILURE-PROCESSING-PREFIX)
XXXXFAI DS A A(FAILURE-PROCESSING-SUFFIX)
XXXXFAIL DS XL2 L(FAILURE-PROCESSING-SUFFIX)
XXXTEXTL DS XL2 L(TEXT)
XXXTEXT DS A A(TEXT)
XXXAPRTL DS 50A A(A-PARTNER-NAME)
XXXAPRT DS 50XL2 L(A-PARTNER-NAME)
*
XXXFTFUN DS XL2 FT-FUNCTION: UNCHG / NOTRS /
* [TRANF]+[MODA]+[READ]+[FPRO]
XXXIIMAN DS XL1 INBOUND-MANAGEMENT: UNCHG / NO / YES
XXXITEXT DS XL1 TEXT: UNCHG / NONE
*
XXXVALID DS XL1 VALIDATE: UNCHG / YES / NO
XXXUSAGE DS XL1 USAGE: UNCHG / PRIV / PUBL
XXXIEXDA DS XL1 EXPIRATION-DATE: UNCHG / NOTRS
XXXIPSUC DS XL1 SUCCESS-PROCESSING-PREFIX: UNCHG / NOTRS
*
XXXIXSUC DS XL1 SUCCESS-PROCESSING-SUFFIX: UNCHG / NOTRS
XXXIPFAI DS XL1 FAILURE-PROCESSING-PREFIX: UNCHG / NOTRS
XXXIXFAI DS XL1 FAILURE-PROCESSING-SUFFIX: UNCHG / NOTRS
*
* VRS_MPRF4
*
XXXDENC DS XL1 DATA-ENCRYPTION: UNCHG / NOTRS / YES / NO
XXXFTPRL EQU *-XXXFTPRF

```

The meaning of the parameters in the macro MODFTPRF corresponds to the meaning of the parameters in the command MODIFY-FT-PROFILE as shown in the following table.

Operand (L format)	Address field (D format)	Meaning in command
NAME	NAME	NAME
PASS	PASS	PASSWORD
SELECT	SEL	SELECT-PARAMETER
TAD	TAD	TRANSFER-ADMISSION in the structure SELECT-PARAMETER
OWNER	OWN	OWNER-IDENTIFICATION in the structure SELECT-PARAMETER
NEWNAME	NNAME	NEWNAME
NEWTAD	NTAD	NEW-TRANSFER-ADMISSION
VALID	VALID	VALID
USAGE	USAGE	INVALID-ON-DETECTION
EXPDATE	EXDA	EXPIRATION-DATE
PRIV	PRIV	PRIVILEGED
ILV	ILV	IGNORE-MAX-LEVELS
IOSND	IOSND	OUTBOUND-SEND in the structure IGNORE-MAX-LEVELS
IORCV	IORCV	OUTBOUND-RECEIVE in the structure IGNORE-MAX-LEVELS
IISND	IISND	INBOUND-SEND in the structure IGNORE-MAX-LEVELS
IIRCV	IIRCV	INBOUND-RECEIVE in the structure IGNORE-MAX-LEVELS
IIPRC	IIPRC	INBOUND-PROCESSING in the structure IGNORE-MAX-LEVELS
IIMAN	IIMAN	INBOUND-MANAGEMENT in the structure IGNORE-MAX-LEVELS
UAD	UAD	USER-ADMISSION
UUSER	UUS	USER-IDENTIFICATION in the structure USER-ADMISSION
UACC	UAC	USER-ACCOUNT in the structure USER-ADMISSION
UPASS	UPA	USER-PASSWORD in the structure USER-ADMISSION
INIT	INIT	INITIATOR
TRANS	TRDIR	TRANSFER-DIRECTION
PARTNER	PART	PARTNER-NAME
MAXPLV	MPLV	MAX-PARTNER-LEVEL
FILE	FILE	FILE-NAME
PFXFIL	FILE	PREFIX of LIBRARY
LIB	FILE	LIBRARY
EL	EL	ELEMENT

Operand (L format)	Address field (D format)	Meaning in command
PFXEL	EL	PREFIX of ELEMENT
ELVER	EV	ELEMENT-VERSION
ELTYP	ETY	ELEMENT-TYPE
FPASS	FPA	FILE-PASSWORD
PAD	PAD	PROCESSING-ADMISSION
PUSER	PUS	USER-IDENTIFICATION in the structure PROCESSING-ADMISSION
PACC	PAC	USER-ACCOUNT in the structure PROCESSING-ADMISSION
PPASS	PPA	USER-PASSWORD in the structure PROCESSING-ADMISSION
SUCC	SUC	SUCCESS-PROCESSING
PFXSUCC	PSUC	PREFIX in the structure SUCCESS-PROCESSING
SUCSUCC	XSUC	SUFFIX in the structure SUCCESS-PROCESSING
FAIL	FAI	FAILURE-PROCESSING
PFXFAIL	PFAI	PREFIX in the structure FAILURE-PROCESSING
SUCFAIL	XFAI	SUFFIX in the structure FAILURE-PROCESSING
WRITE	WRMOD	WRITE-MODE
FTFUNC	FTFUN	FT-FUNCTION
TEXT	TEXT	USER-INFORMATION
CHIP	CHIP	CHIPCARD in the structure TRANSFER-ADMISSION
DENC	DENC	DATA-ENCRYPTION

If a chipcard profile is created at the command interface (TRANSFER-ADMISSION=*CHIPCARD(TRANSFER-ADMISSION=,CERTIFICATE=)), then the internally stored transfer admission is formed from the specified transfer admission together with the certificate number and certification authority.

At the programming interface it is not possible to specify these components of the transfer admission in separate fields. Instead, it is the responsibility of the caller of the programming interface to structure the transfer admission correctly. The structure of the transfer admission for specification is as follows:

4 bytes	Number of the Certification Authority configured in the system
4 bytes	Certificate number
0-7 bytes	Transfer admission specified in the partner system. If this component of the transfer admission is empty then this corresponds to the specification TRANSFER-ADMISSION=*NONE at the command interface.

2.2.8 SHWFTADS - Output admission sets

The macro SHWFTADS can be used to view admission sets. The functionality corresponds to that of the command SHOW-FT-ADMISSION-SET.

Format of the macro SHWFTADS (LIST format/default format)

Name	Operation	Operand
[Name]	SHWFTADS	[MF=L] [, USER= { *OWN *ALL *STD adr }] [, OUTPUT= { [*]SYSQUT [*]SYSLST }] [, LAYOUT= { [*]STD [*]CSV }]

Format of the macro SHWFTADS (DSECT format)

```

xxxFTADS DSECT
xxxHADS FHDR MF=(C,&P)
xxxUSER DS A A(USERID)
xxxUSERL DS XL2 L(USERID)
xxxBUFL DS XL2 RESERVED
xxxBUF DS A RESERVED
*
* 1-BYTE FIELDS FOR ENTERING KEYWORDS *
*
xxxIUSER DS XL1 USERID: OWN/STD/ALL
xxxSEL DS XL1 SELECT: ALL
xxxOUTPT DS XL1 OUTPUT/LAYOUT: SOUT/SLST/SOUTC/SLSTC
xxxRESRV DS XL1 RESERVED
*
xxxFTADL EQU *-xxxFTADS

```

The meaning of the parameters in the macro SHWFTADS corresponds to the meaning of the parameters in the command SHOW-FT-ADMISSION-SET as shown in the following table. In the OUTPT address field, SOUT stands for OUTPUT=*SYSOUT and SLST stands for OUTPUT=*SYSLST in the command. SOUTC and SLSTC stand for their respective output in CSV format (LAYOUT=*CSV). For the other two values, output is in standard format (LAYOUT=*STD).

Operand (L format)	Address field (D format)	Meaning in command
USER	USER	USER-IDENTIFICATION
OUTPUT	OUTPT	OUTPUT
LAYOUT	OUTPT	LAYOUT

2.2.9 SHWFTENV - List admission sets and profiles

You use the SHWFTENV macro to display or output previously logged FTAC admission sets or admission profiles. The output is sent to SYSOUT or SYSLST. To this end, you can also use the EXPFTENV macro to output the FTAC admission sets or admission profiles to a file ([page 102](#)). The functionality corresponds to that of the command SHOW-FTAC-ENVIRONMENT.

Format of the macro SHWFTENV (LIST format/default format)

Name	Operation	Operand
[Name]	SHWFTENV	<pre> [MF=L] , FILE=adr [, USER= { *ALL (adr1, ..., adr100) }] [, SELECT= [*]ALL / [, NAME= { *ALL *NONE (adr1, ..., adr100) }] [, ADMS= { [*]YES [*]NO }]] [, INF= { [*]ONLY [*]ALL }] [, OUTPUT= { [*]SYSOUT [*]SYSLST }] [, LAYOUT= { [*]STD [*]CSV }] </pre>

Format of the macro SHWFTENV (DSECT format)

```

xxxFTENV DSECT
xxxHENV  FHDR  MF=(C,&P)
xxxFILE  DS   A      A(FILE)
xxxFILEL DS   XL2    L(FILE)
xxxRES   DS   XL2    RESERVED
xxxUSERL DS   100XL2 L(A-USER)
xxxUSER  DS   100A   A(A-USER)
xxxNAME  DS   100A   A(A-NAME)
xxxNAMEL DS   100XL2 L(A-NAME)
*
* 1-BYTE FIELDS FOR ENTERING KEYWORDS *
*
xxxSEL   DS   XL1    SELECT: ALL
xxxIUSER DS   XL1    USER: ALL
xxxINAME DS   XL1    PROFIL-NAME: ALL / NONE
xxxADMS  DS   XL1    ADMISSION-SET: YES / NO
xxxINFO  DS   XL1    INFORMATION: ONLY / ALL
xxxOUTPT DS   XL1    OUTPUT/LAYOUT: SOUT/SLST/SOUTC/SLSTC
xxxRESRV DS   XL2    RESERVED
xxxFTENL EQU  *-xxxFTENV

```

The meaning of the parameters in the macro SHWFTENV corresponds to the meaning of the parameters in the command SHOW-FTAC-ENVIRONMENT as shown in the following table. In the OUTPT address field, SOUT stands for OUTPUT=*SYSOUT and SLST stands for OUTPUT=*SYSLST in the command. SOUTC and SLSTC stand for their respective output in CSV format (LAYOUT=*CSV). For the other two values, output is in standard format (LAYOUT=*STD).

Operand (L format)	Address field (D format)	Meaning in command
FILE	FILE	FROM-FILE
USER	USER	USER-IDENTIFICATION
SELECT	SEL	SELECT-PARAMETER
NAME	NAME	PROFILE-NAME in the structure SELECT-PARAMETER
ADMS	ADMS	ADMISSION-SET in the structure SELECT-PARAMETER
INF	INFO	INFORMATION
OUTPUT	OUTPT	OUTPUT
LAYOUT	OUTPT	LAYOUT

2.2.10 SHWFTPRF - Output admission profile

The macro SHWFTPRF can be used to view an admission profile. The functionality corresponds to that of the command SHOW-FT-PROFILE.

Format of the macro SHWFTPRF (LIST format/default format)

Name	Operation	Operand
[Name]	SHWFTPRF	[MF=L] [,NAME= { *ALL adr }] [,SELECT=*OWN / [,TAD= { *ALL *NOTSP adr }] [,OWNER= { *OWN *ALL adr }]] [,INF= { [*]ONLY [*]ALL }] [,OUTPUT= { [*]SYSOUT [*]SYSLST }] [,LAYOUT= { [*]STD [*]CSV }]

Format of the macro SHWFTPRF (DSECT format)

```

xxxFTPRF DSECT
xxxHPRF FHDR MF=(C,&P)
xxxNAME DS A A(NAME)
xxxNAMEL DS XL2 L(NAME)
xxxTADL DS XL2 L(SELECT-TRANSFER-ADMISSION)
xxxTAD DS A A(SELECT-TRANSFER-ADMISSION)
xxxOWN DS A A(SELECT-OWNERID)
xxxOWNL DS XL2 L(SELECT-OWNERID)
xxxBUFL DS XL2 RESERVED
xxxBUF DS A RESERVED
*
* 1-BYTE FIELDS FOR ENTERING KEYWORDS *
*
xxxINAME DS XL1 NAME: ALL/STD
*
xxxINFO DS XL1 INFORMATION: ONLY / ALL
*
xxxSEL DS XL1 SELECT: OWN
xxxITAD DS XL1 TRANSFER-ADMISSION: ALL / NOTSP
xxxIOWN DS XL1 OWNERID: OWN /ALL
*
xxxOUTPT DS XL1 OUTPUT/LAYOUT: SOUT/SLST/SOUTC/SLSTC
*
xxxRESRV DS XL2 RESERVED
*
xxxFTPRL EQU *-xxxFTPRF

```

The meaning of the parameters in the macro SHWFTPRF corresponds to the meaning of the parameters in the command SHOW-FT-PROFILE as shown in the following table. In the OUTPT address field, SOUT stands for OUTPUT=*SYSOUT and SLST stands for OUTPUT=*SYSLST in the command. SOUTC and SLSTC stand for their respective output in CSV format (LAYOUT=*CSV). For the other two values, output is in standard format (LAYOUT=*STD).

Operand (L format)	Address field (D format)	Meaning in command
NAME	NAME	NAME
SELECT	SEL	SELECT-PARAMETER
TAD	TAD	TRANSFER-ADMISSION in the structure SELECT-PARAMETER
OWNER	OWN	OWNER-IDENTIFICATION in the structure SELECT-PARAMETER
INF	INFO	INFORMATION
OUTPUT	OUTPT	OUTPUT
LAYOUT	OUTPT	LAYOUT

2.2.11 SHWFTRGE - List accessible partner systems

The macro SHWFTRGE can be used to list accessible partner systems. The functionality corresponds to that of the command SHOW-FT-RANGE.

Format of the macro SHWFTRGE (LIST format/default format)

Name	Operation	Operand
[Name]	SHWFTRGE	[MF=L] [,USER= { *OWN adr }] [,SELECT= [*]ALL / PARTNER= { *ALL adr }] [,OUTPUT= { [*]SYSOUT [*]SYSLST }] [,LAYOUT= { [*]STD [*]CSV }]

Format of the macro SHWFTRGE (DSECT format)

```

xxxFTRGE DSECT
xxxHRGE FHDR MF=(C,&P)
xxxUSER DS A A(USERID)
xxxUSERL DS XL2 L(USERID)
xxxPTNL DS XL2 L(SELECT-PARTNERNAME)
xxxPTN DS A A(SELECT-PARTNERNAME)
xxxBUF DS A RESERVED
xxxBUFL DS XL2 RESERVED
*
* 1-BYTE FIELDS FOR ENTERING KEYWORDS *
*
xxxIUSER DS XL1 USER-ID: OWN
*
xxxSEL DS XL1 SELECT: ALL
xxxIPTN DS XL1 PARTNER: ALL
*
xxxOUTPT DS XL1 OUTPUT: SOUT / SLST / SOUTC / SLSTC
*
xxxRESRV DS XL2 RESERVED
*
xxxFTRGL EQU *-xxxFTRGE

```

The meaning of the parameters in the macro SHWFTRGE corresponds to the meaning of the parameters in the command SHOW-FT-RANGE. In the OUTPT address field, SOUT stands for OUTPUT=*SYSOUT and SLST stands for OUTPUT=*SYSLST in the command. SOUTC and SLSTC stand for their respective output in CSV format (LAYOUT=*CSV). For the other two values, output is in standard format (LAYOUT=*STD).

Operand (L format)	Address field (D format)	Meaning in command
USER	USER	USER-IDENTIFICATION
SELECT	SEL	SELECT-PARAMETER
PARTNER	PTN	PARTNER-NAME in the structure SELECT-PARAMETER
OUTPUT	OUTPT	OUTPUT
LAYOUT	OUTPT	LAYOUT

2.2.12 YFSEQU - Set constants for the openFT-AC macros

No operands are to be entered for the macro call. The equates are generated with the default prefix YFS, as they are used internally by the other FTAC macros.

Call of the macro YFSEQU

Name	Operation	Operand
	YFSEQU	

YFSTRANF	FT-FUNCTION = TRANSFER-FILE: TRANSF
YFSMODA	FT-FUNCTION = MODIFY-ATTRIBUT: MODATT
YFSREAD	FT-FUNCTION = READ-DIRECTORY: READDIR
YFSFPRO	FT-FUNCTION = FILE-PROCESSING: FILEPROC
YFSREM	INITIATOR = REMOTE: [*]REMOTE
YFSLOC	INITIATOR = LOCAL: [*]LOCAL
YFSBOTH	INITIATOR = (LOCAL,REMOTE): [*]BOTH
YFSFROM	TRANSFER DIRECTION = FROM: [*]FROM
YFSTO	TRANSFER DIRECTION = TO: [*]TO
YFSNEW	WRITE-MODE = NEW FILE: [*]NEW
YFSRPL	WRITE-MODE = REPLACE: [*]RPL
YFSEXT	WRITE-MODE = EXTEND: [*]EXT
YFSONLY	INFORMATION = ONLY-NAMES: [*]ONLY
YFSSOUT	OUTPUT = SYSOUT: [*]SYSOUT
YFSSLST	OUTPUT = SYSLST: [*]SYSLST
YFSSOUTC	OUTPUT = SYSOUT, LAYOUT = CSV
YFSSLSTC	OUTPUT = SYSLST, LAYOUT = CSV
YFSLIBEL	LIBRARY-ELEMENT
YFSEXP	EXPANSION
YFSNO	NO: [*]NO
YFSYES	YES: [*]YES
YFSNONE	NONE: [*]NONE
YFSSAME	SAME: [*]SAME
YFSALL	ALL: [*]ALL
YFSOWN	OWN: [*]OWN
YFSOLD	OLD: [*]OLD
YFSFIRST	FIRST: [*]FIRST

```

YFSADD          PARTNER = ADD
YFSREMOV       PARTNER = REMOVE
YFSPRIV        USAGE = PRIVATE: [*]PRIVATE
YFSPUBL        USAGE = PUBLIC: [*]PUBLIC
YFSNOTSP       NOT-SPECIFIED: *NOTSP

YFSNOTRS       NOT-RESTRICTED: [*]NOTRS
YFSUNCHG       UNCHANGED: [*]UNCHG
YFSSTD         STD: [*]STD

```

2.2.13 Example for the use of the programming interface

The following section contains an example of the use of the programming interface. The example illustrates the macros CREFTPRF, MODFTADS, SHWFTADS and SHWFTPRF.

Example

```

START
BALR  10,0
USING *,10
PRINT NOGEN

*
*           set FTAC password "NURICH",
*           prevent FT by setting security level at 0:
*
MODFTADS NEWPASS=V1,MAXLV=0
BAL   4,CHECKRC          (branch to return code check)
*
*           Display admission set (on SYSOUT):
*
SHWFTADS
BAL   4,CHECKRC          (branch to return code check)
*
*           Create admission profile with the name "MYPROFIL" and
*           transfer admission "accessformeonly";
*           The FT initiative can only be in the local system,
*           the security levels of the admission set should be ignored,
*           the entry for processing admission is not limited:
*
CREFTPRF NAME=V2,
          PASS=V1,
          TAD=V3,
          ILV=*YES,
          INIT=*LOCAL,
          PAD=*NOTRS
BAL   4,CHECKRC          (branch to return code check)

```



```
*
*           Display the created admission profile
*           (macro with EXECUTE and LIST format):
*
SHWFTPRF MF=(E,SHWFTPRF)   EXECUTE format
BAL 4,CHECKRC             (branch to return code check)
*
TERM
*
*           check last byte of return code for 0:
*
CHECKRC CLI 7(1),X'00'
BRE 4                     if 0: continue program
TERM UNIT=STEP,MODE=A    else: abnormal program termination.
*
YFSEQU                    macro with FTAC equates
*
SHWFTPRF SHWFTPRF NAME=V2,INF=*ALL LIST format
*
V1 DC C'NURICH'           FTAC password
V2 DC C'MYPROFIL'        profile name
V3 DC C'accessformeonly' transfer admission
*
END
```

2.2.14 Return information and error displays

The field for the return code has the following format:

Subcode2	Subcode1	Maincode	
warnings	error class	Parameter error	
1 byte	1 byte	1 byte	1 byte

The field with the return code can be addressed by DSECT after a macro call (field xxxRETC, generated by the macro FHDR) or via register 1 (see [section “Example for the use of the programming interface” on page 128](#)).

Maincode

This field contains the exact error. It is divided into two bytes. The right byte shows the cause of error. It can have the following values (in decimal notation):

- 0: call was error-free
- 1: the parameter field is partially or entirely in an invalid address space
- 2: the parameter may not yet be set in this version
- 3: the address or length field contradict the keyword field
- 4: the parameter entry contradicts another parameter
- 10: the buffer is too small; this error code is reserved for future expansions
- 11: the arithmetic parameter value (e.g. RECSIZE value) is invalid
- 12: the parameter value is too long or violates syntax rules; e.g. FILENAME, PASSWORD
- 13: the keyword entered is not permitted for this parameter
- 14: a mandatory parameter is missing
- 70-255: these error codes have the same meaning as the FTAC messages with the related FTC number

The corresponding equates are defined in the macro YFSEQU with the prefix YFS1.

The left byte is only assigned a value other than 0 in the case of a parameter error. It specifies the erroneous parameter. The corresponding equates are defined in the macro YFSEQU with the prefix YFS2.

Subcode1

The error codes are divided into error classes. These are found in subcode 1. The following error classes exist:

Class	Value	Meaning
A	0	the call was successfully completed
B	1	permanent error, no repetition possible, there was a syntax error or equivalent parameter error
C	32	system error an internal error occurred during macro processing
D	64	repeat after correction of user input
E	128	wait and repeat

The corresponding equates are defined in the macro YFSEQU with the prefix YFSC.

Subcode2

This field contains a warning, if the macro was completed successfully (subcode1 = 0 and maincode = 0) but the macro couldn't be executed - for instance, if no files could be found during the SHWFTPRF call. The corresponding equates are defined in the macro YFSEQU with the prefix YFSW.

Exception

If the maincode contains the value X'FFFF', the request could not be executed for reasons specified uniquely throughout the system. The equates valid in this case for maincode, subcode1 and subcode2 are defined in the macro FHDR, which generates the default header.

The following information can be defined in the macro YFSEQU:

```

* ERROR CODES
*
YFSOK    EQU    0          NO ERROR
*
* MAIN CODE VALUES
* PERMANENT ERRORS
* BYTE 1 DEFINES THE ERROR
*
YFS1INVA EQU    1          INVALID ADDRESS OF PARAMETER VALUE
YFS1RES  EQU    2          PARAMETER RESERVED FOR FUTURE USE
YFS1INC  EQU    3          PARAMETER AND INDICATOR INCONSISTENT
YFS1PINC EQU    4          INCONSISTENCY WITH OTHER PARAMETER
* REPARABLE ERRORS
YFS1BUFS EQU    10         BUFFER TOO SMALL
YFS1RNGE EQU    11         PARAMETER VALUE OUT OF RANGE
YFS1YERR EQU    12         WRONG SYNTAX IN PARAMETER VALUE
YFS1KEYV EQU    13         INVALID KEYWORD VALUE
YFS1MAND EQU    14         MANDATORY PARAMETER MISSING
* SHORTAGE OF RESOURCES
YFS1SHRT EQU    70         SHORTAGE OF RESOURCES
YFS1INAC EQU    71         FTAC NOT ACTIVE
* REPARABLE ERRORS
YFS1PROF EQU    100        PROFILE ALREADY EXISTS
YFS1TAD  EQU    101        TRANSFER-ADMISSION ALREADY EXISTS
YFS1FILE EQU    102        FILE-NAME ALREADY EXISTS
YFS1FINV EQU    103        INVALID FILE CONTENT
* PERMANENT ERRORS
YFS1PASS EQU    150        USER NOT AUTH. FOR FTAC COMMANDS
YFS1MOD  EQU    151        USER NOT AUTH. FOR THIS MODIFIC.
YFS1USER EQU    152        USER NOT AUTH. FOR OTHER USERID'S
YFS1OWNR EQU    153        USER NOT AUTH. FOR OTHER OWNERID'S
YFS1DEL  EQU    154        USER NOT AUTH. TO DEL LOGGING RECS
YFS1DIAG EQU    155        USER NOT AUTH. FOR DIAGNOSE
YFS1UPRT EQU    170        GIVEN PARTNERNAME UNKNOWN
YFS1UPRF EQU    171        GIVEN PROFILENAME UNKNOWN
YFS1IUAD EQU    172        INVALID USER-ADMISSION
YFS1IPAD EQU    173        INVALID PROCESSING-ADMISSION
YFS1IMUS EQU    174        INVALID MODIF. FOR NOT UNIQUE SELECTION
YFS1ISTD EQU    175        MODIFICATION INVALID FOR *STD
YFS1IUID EQU    176        GIVEN USERID UNKNOWN
YFS1UFIL EQU    177        FILENAME UNKNOWN
YFS1PANU EQU    178        GIVEN PARTNERNAME NOT UNIQUE
YFS1PAVI EQU    179        VIOLATION OF MAX. NUMBER OF PARTNER
YFS1USNU EQU    180        GIVEN USER IDENTIFICATION NOT UNIQUE
YFS1PRNU EQU    181        GIVEN PROFILE NAME NOT UNIQUE
YFS1LENP EQU    200        SUCCESS AND FAILURE PROC TOO LONG

```

```

YFS1PFLO EQU 206          PARTIALLY QUALIFIED FILENAME TOO LONG
* SYSTEM ERROR
YFS1SERR EQU 255          SYSTEM ERROR

*
* BYTE2 DEFINES THE ERRONEOUS PARAMETER
*
YFS2PAR EQU 1             INVALID ADDRESS OF PARAMETER LIST
YFS2NAME EQU 2            NAME
YFS2NNAM EQU 3            NEWNAME
YFS2OWN EQU 4             OWNER
YFS2USER EQU 5            USER
YFS2SEL EQU 6             SELECT
YFS2PASS EQU 7            PASSWORD
YFS2NPA EQU 8             NEW PASSWORD
YFS2TAD EQU 9             TRANSFER ADMISSION
YFS2NTAD EQU 10           NEW TRANSFER ADMISSION
YFS2MOSN EQU 11           MAX LEVEL OUTBOUND-SEND
YFS2MORC EQU 12           MAX LEVEL OUTBOUND-RECEIVE
YFS2MISN EQU 13           MAX LEVEL INBOUND-SEND
YFS2MIRC EQU 14           MAX LEVEL INBOUND-RECEIVE
YFS2MIPR EQU 15           MAX LEVEL INBOUND-PROCESSING
YFS2MIMA EQU 16           MAX LEVEL INBOUND-MANAGEMENT
YFS2ILV EQU 21            IGNORE MAX LEVEL
YFS2IOSN EQU 22           IGNORE MAX LEVEL OUTBOUND-SEND
YFS2IORC EQU 23           IGNORE MAX LEVEL OUTBOUND-RECEIVE
YFS2IISN EQU 24           IGNORE MAX LEVEL INBOUND-SEND
YFS2IIRC EQU 25           IGNORE MAX LEVEL INBOUND-RECEIVE
YFS2IIPR EQU 26           IGNORE MAX LEVEL INBOUND-PROCESSING
YFS2IIMA EQU 27           IGNORE MAX LEVEL INBOUND-MANAGEMENT
YFS2MPLV EQU 32           MAX PARTNER LEVEL
YFS2PART EQU 33           PARTNER NAME
YFS2UAD EQU 34            USER ADMISSION
YFS2UUS EQU 35            USERID
YFS2UAC EQU 36            ACCOUNT NUMBER
YFS2UPA EQU 37            USER PASSWORD
YFS2PAD EQU 38            PROCESSING ADMISSION
YFS2PUS EQU 39            USERID
YFS2PAC EQU 40            ACCOUNT NUMBER
YFS2PPA EQU 41            PASSWORD
YFS2FIL EQU 42            FILENAME
YFS2PFIL EQU 43           PREFIX OF FILENAME
YFS2LIB EQU 44            LIBRARY NAME
YFS2PLIB EQU 45           PREFIX OF LIBRARY NAME
YFS2EL EQU 46             ELEMENT NAME
YFS2PEL EQU 47           PREFIX OF ELEMENT NAME
YFS2EV EQU 48            ELEMENT VERSION
YFS2ETY EQU 49            ELEMENT TYPE

```

YFS2FPA	EQU	50	FILE PASSWORD
YFS2SUC	EQU	51	SUCCESS PROCESSING
YFS2PSUC	EQU	52	PREFIX OF SUCCESS PROCESSING
YFS2FAI	EQU	53	FAILURE PROCESSING
YFS2PFAI	EQU	54	PREFIX OF FAILURE PROCESSING
YFS2PRIV	EQU	55	PRIVILEGED
YFS2INIT	EQU	56	TRANSFER INITIATOR
YFS2TDIR	EQU	57	TRANSFER DIRECTION
YFS2WMOD	EQU	58	WRITE MODE
YFS2INFO	EQU	59	INFORMATION
YFS2OUTP	EQU	60	OUTPUT
YFS2BUF	EQU	61	BUFFER
YFS2RES	EQU	64	RESERVED
YFS2DATE	EQU	65	DATE
YFS2FTFU	EQU	66	FT-FUNCTION
YFS2ADMS	EQU	67	ADMISSION SET
YFS2XSUC	EQU	68	SUFFIX OF SUCCESS PROCESSING
YFS2XFAI	EQU	69	SUFFIX OF FAILURE PROCESSING
YFS2USAG	EQU	70	USAGE
YFS2VALI	EQU	71	VALID
YFS2TEXT	EQU	72	TEXT
YFS2CHIP	EQU	73	CHIP
YFS2DENC	EQU	74	DENC
*			
* SUBCODE1 DEFINES THE ERROR CLASS			
*			
YFSCOK	EQU	0	NO ERROR
YFSCPERR	EQU	1	PERMANENT ERROR
YFSCSERR	EQU	32	SYSTEM ERROR
YFSCRERR	EQU	64	REPARABLE ERROR
YFSCSHRT	EQU	128	SHORTAGE OF RESOURCES
*			
* SUBCODE2 DEFINES THE WARNINGS			
*			
YFSWLLEV	EQU	50	LOWER ADM-LEVEL REMAINS IN EFFECT
YFSWWARN	EQU	51	TRANSFER-ADMISSION EXISTS AS USERID
YFSWINFI	EQU	52	INFORMATION INCOMPLETE
YFSWNPRF	EQU	53	NO PROFILE FOUND
YFSWNINF	EQU	54	NO INFORMATION AVAILABLE
YFSWPANR	EQU	55	PARTNER RESTRICT. DOES NOT LONGER EXIST
YFSWTADL	EQU	56	TRANSFER ADMISSION LOCKED
*			
FHDR	MF=(C,&P),EQUATES=ONLY		

3 COBOL programming interface

Programs which were created for openFT versions as of V2.0 can also be run with openFT V12.0 for BS2000.

3.1 COBOL macros for openFT for BS2000

The functions of openFT for BS2000 can be used via the following COBOL macros:

COBOL macro	Function	Command
CALL "NCANCEL"	cancel file transfer request	CANCEL-FILE-TRANSFER
CALL "NCOPY"	transfer file	TRANSFER-FILE
CALL "NDEL"	delete remote file	DELETE-REMOTE-FILE
CALL "NLMOD"	modify local FT file attributes	MODIFY-FILE-FT-ATTRIBUTES
CALL "NLSHOW"	display local FT file attributes	SHOW-FILE-FT-ATTRIBUTES
CALL "NMOD"	modify remote file attributes	MODIFY-REMOTE-FILE-ATTRIBUTES
CALL "NSHOW"	display remote file attributes	SHOW-REMOTE-FILE-ATTRIBUTES
CALL "NSTAT"	query file transfer status	SHOW-FILE-TRANSFER

For programs containing calls to openFT for BS2000, the module YNDCOBOL must be either explicitly connected or connected via the AUTOLINK function of the linkage editor. The module YNDCOBOL is in the library \$SYSFJAM.SYSRTC.FT.

At the execution time of the program, the module YNDCOBOL is loaded from the library \$SYSFJAM.SYSRTC.FT.

The COPY elements are in the library SYSLIB.OPENFT.120 under the user ID set by the BS2000 system administrator.



In openFT V9.0 for BS2000, file and link names have been changed. Therefore, you must uninstall the older version of openFT **before** switching to a version of openFT earlier than V9.0, otherwise the YNDCOBOL module of the older version might be used. If you are not installing with IMON, you must, for reasons of compatibility, copy the SYSRTC.OPENFT.120 file under the name SYSRTC.FT on the configuration user ID of the openFT instance (standard \$SYSFJAM).

3.1.1 NCANCEL - Cancel file transfer request

The macro CALL "NCANCEL"... can be used to reverse FT requests or cancel the file transfer. openFT for BS2000 deletes the FT requests from the FT request file which meet the selection criteria specified and cancels any related file transfers. The functionality corresponds to that of the command CANCEL-FILE-TRANSFER (NCANCEL)

Macro

The function can be called as follows:

```
CALL "NCANCEL" USING FT-NCANCEL-LIST FT-RETURN-INFO.
```

FT-NCANCEL-LIST

The range FT-NCANCEL-LIST describes the parameter list for the NCANCEL macro. FT-NCANCEL-LIST must be defined in the WORKING-STORAGE SECTION and can be copied to there with the statement

```
COPY FTNCAN OF linkname.
```

Before executing the macro you must fill the desired fields. If a parameter is not specified or the default value is to be used, this field must be assigned the value LOW-VALUE.

FT-NCANCEL-LIST is defined as follows:

```
01 FT-NCANCEL-LIST.
*
  02 FILLER                                PIC X(4) VALUE "1000".
*
  02 USER-PARAMETERS.
*
    05 TRANSFER-ID                          PIC X(10).
      88 SELECT-ALL                          VALUE LOW-VALUE.
*
    05 SELECT-PARAMETER.
```

```

*
      10 OWNER-IDENTIFICATION PIC X(8).
          88 OWN VALUE LOW-VALUE.
          88 SELECT-ALL VALUE " ".
*
      10 INITIATOR PIC X(1).
          88 SELECT-ALL VALUE LOW-VALUE.
          88 LOCAL VALUE "L".
          88 REMOTE VALUE "R".
*
      10 PARTNER-NAME PIC X(200).
          88 SELECT-ALL VALUE LOW-VALUE.
*
      10 FILE-NAME PIC X(512).
          88 SELECT-ALL VALUE LOW-VALUE.
      10 LIBRARY PIC X(56).
          88 SELECT-ALL VALUE LOW-VALUE.
      10 ELEMENT PIC X(64).
          88 SELECT-ALL VALUE LOW-VALUE.
      10 TYP PIC X(8).
          88 SELECT-ALL VALUE LOW-VALUE.
      10 VERSION PIC X(24).
          88 SELECT-ALL VALUE LOW-VALUE.
*
      10 MONJV PIC X(56).
          88 NONE VALUE LOW-VALUE.
*
      10 JV-PASSWORD PIC X(11).
          88 NONE VALUE LOW-VALUE.
*
      05 FORCE-CANCELLATION PIC X(1).
          88 NO VALUE LOW-VALUE.
          88 YES VALUE "Y".

```

The version specification at the beginning of the structure serves to identify the COPY element and must not be overwritten by the user.

If more than one selection criterion is specified in the NCANCEL macro, a request can be overdefined, for instance, if both TRANSFER-ID and MONJV are specified. If the criteria contradict each other in such a case, the NCANCEL macro is not executed.

You may only set YES if you have explicitly specified a transfer ID in the TRANSFER-ID field. This request must already have been cancelled with FORCE-CANCELLATION=NO and is only possible if the user ID possesses the FT-ADM privilege.

Description of the data fields

The parameters for FT-NCANCEL-LIST have the same name and functions as the corresponding operands for the command CANCEL-FILE-TRANSFER (NCANCEL). Please refer to the corresponding command description in the user manual.

NCANCEL example

```

*****
*   EXAMPLE :                                           *
*                                                     *
*   CANCELLATION OF A FILE TRANSFER REQUEST           *
*   FROM A COBOL PROGRAM                               *
*****
  IDENTIFICATION DIVISION.
  PROGRAM-ID. NCA.
*
  ENVIRONMENT DIVISION.
  CONFIGURATION SECTION.
  SPECIAL-NAMES.
    TERMINAL IS TERM.
*
  DATA DIVISION.
  WORKING-STORAGE SECTION.
  77 MAIN-RCODE-STRING          PIC -ZZZZ9.
  77 SUB-RCODE-STRING          PIC -ZZZZ9.
  COPY FTNCAN OF FTLIB.
  COPY FTRETC OF FTLIB.
*
  PROCEDURE DIVISION.
  CONTROL SECTION.
  ST-01.
*
*TRANS-ID-READING.
  DISPLAY "PLEASE ENTER TRANSFER-ID" AT TERM.
  ACCEPT TRANSFER-ID IN FT-NCANCEL-LIST FROM TERM.
*
*NCANCEL-CALLING.
  CALL "NCANCEL" USING FT-NCANCEL-LIST FT-RETURN-INFO.
*
*RESULT-HANDLING.
  IF OKAY IN MAIN-RETURN-CODE
    DISPLAY "FILE TRANSFER REQUEST CANCELLED"
    UPON TERM

```

```
ELSE
  MOVE MAIN-RETURN-CODE TO MAIN-RCODE-STRING
  MOVE SUB-RETURN-CODE TO SUB-RCODE-STRING
  DISPLAY "TRANSFER CANCELLATION REJECTED" UPON TERM
  DISPLAY "MAIN-RETURN-CODE: " MAIN-RCODE-STRING
    " SUB-RETURN-CODE: " SUB-RCODE-STRING
    UPON TERM.
ST-99.
STOP RUN.
```

3.1.2 NCOPY - Transfer file

The macro CALL "NCOPY"... can be used to transfer a file or library member. The local system can be the sender or receiver of the file. The local system is considered to be the system in which the call is initiated. The partner system is the remote system. The functionality corresponds to that of the command TRANSFER-FILE (NCOPY).

Macro

The function can be called as follows:

```
CALL "NCOPY" USING FT-NCOPY-LIST FT-RETURN-INFO.
```

FT-NCOPY-LIST

The range FT-NCOPY-LIST describes the parameter list for the NCOPY macro. FT-NCOPY-LIST must be defined in the WORKING-STORAGE SECTION and can be copied to there with the statement

```
COPY FTNCOPY [OF linkname].
```

Before the first NCOPY macro, the parameter fields should be deleted with the statement

```
MOVE LOW-VALUE TO USER-PARAMETERS IN FT-NCOPY-LIST.
```

Before executing another NCOPY macro, you must fill the desired fields. If a parameter is not specified or the default value is to be used, this field must be assigned the value LOW-VALUE.

FT-NCOPY-LIST is defined as follows:

```

01 FT-NCOPY-LIST.
*
  02 FILLER                PIC X(4)  VALUE "1000".
*
  02 USER-PARAMETERS.
*
    05 TRANSFER-DIRECTION  PIC X(1).
      88 TO-PARTNER        VALUE "T".
      88 FROM-PARTNER      VALUE "F".
*
    05 PARTNER-NAME        PIC X(200).
*
    05 LOCAL-PARAMETER.
      10 FILE-NAME         PIC X(512).
        88 NOT-SPECIFIED   VALUE LOW-VALUE.
      10 LINK-NAME         PIC X(8).
        88 NOT-SPECIFIED   VALUE LOW-VALUE.
      10 LIBRARY           PIC X(56).
        88 NOT-SPECIFIED   VALUE LOW-VALUE.
      10 ELEMENT           PIC X(64).
        88 NOT-SPECIFIED   VALUE LOW-VALUE.
      10 TYP               PIC X(8).
        88 NOT-SPECIFIED   VALUE LOW-VALUE.
      10 VERSION           PIC X(24).
        88 STD             VALUE LOW-VALUE.
      10 FILE-PASSWORD     PIC X(11).
      10 SUCCESS-PROCESSING PIC X(500).
      10 FAILURE-PROCESSING PIC X(500).
      10 USER-DEF-ADMISSION PIC X(67).
        88 NONE           VALUE LOW-VALUE.
      10 TRANSFER-ADMISSION.
        15 USER-ID        PIC X(8).
        15 ACCOUNT        PIC X(40).
        15 PASSWORD       PIC X(35).
      10 PROCESSING-ADMISSION.
        88 NOT-SPECIFIED   VALUE HIGH-VALUE.
        15 USER-ID        PIC X(8).
        15 ACCOUNT        PIC X(40).
        15 PASSWORD       PIC X(35).
      10 LISTING           PIC X(1).
        88 SYSLST-LST     VALUE LOW-VALUE.
        88 LISTFILE       VALUE "L".
        88 SYSLST-FAIL    VALUE "S".
        88 LISTFILE-FAIL  VALUE "E".
        88 NONE           VALUE "N".

```

```

10 MONJV          PIC X(56).
   88 NONE          VALUE LOW-VALUE.
10 JV-PASSWORD   PIC X(11).
   88 NONE          VALUE LOW-VALUE.
10 CODE-CHARACTER-SET PIC X(8).
   88 STD           VALUE LOW-VALUE.
*
05 REMOTE-PARAMETER.
10 REMOTE-SYNTAX PIC X(1).
   88 BS2000        VALUE LOW-VALUE.
   88 MSP           VALUE "3".
   88 ANY-SYNTAX    VALUE "A".
10 FILE-NAME     PIC X(512).
   88 NOT-SPECIFIED VALUE LOW-VALUE.
10 LINK-NAME     PIC X(8).
   88 NOT-SPECIFIED VALUE LOW-VALUE.
10 LIBRARY       PIC X(59).
   88 NOT-SPECIFIED VALUE LOW-VALUE.
10 ELEMENT      PIC X(67).
   88 NOT-SPECIFIED VALUE LOW-VALUE.
10 TYP          PIC X(11).
   88 NOT-SPECIFIED VALUE LOW-VALUE.
10 VERSION      PIC X(27).
   88 NONE          VALUE LOW-VALUE.
   88 STD           VALUE " ".
10 FILE-PASSWORD PIC X(131).
10 CREATE-PASSWORD PIC X(131).
10 SUCCESS-PROCESSING PIC X(500).
10 FAILURE-PROCESSING PIC X(500).
10 USER-DEF-ADMISSION PIC X(67).
   88 NONE          VALUE HIGH-VALUE.
10 TRANSFER-ADMISSION.
   88 NONE          VALUE HIGH-VALUE.
   15 USER-ID      PIC X(67).
   15 ACCOUNT      PIC X(64).
   15 PASSWORD     PIC X(131).
10 PROCESSING-ADMISSION.
   88 NOT-SPECIFIED VALUE HIGH-VALUE.
   88 NONE          VALUE HIGH-VALUE.
   15 USER-ID      PIC X(67).
   15 ACCOUNT      PIC X(64).
   15 PASSWORD     PIC X(131).
10 FILE-AVAILABLE PIC X(1).
   88 NONE          VALUE LOW-VALUE.
   88 IMMEDIATE    VALUE "I".
   88 DEFERRED     VALUE "D".

```



```

10 STORAGE-ACCOUNT      PIC X(40).
    88 NONE              VALUE LOW-VALUE.
10 ACCESS-MODE          PIC X(1).
    88 NONE              VALUE LOW-VALUE.
    88 PAR               VALUE "P".
10 READ-FILE            PIC X(1).
    88 NO-VALUE         VALUE LOW-VALUE.
    88 NO-PAR           VALUE "N".
    88 YES              VALUE "Y".
10 REPLACE-FILE        PIC X(1).
    88 NO-VALUE         VALUE LOW-VALUE.
    88 NO-PAR           VALUE "N".
    88 YES              VALUE "Y".
10 EXTEND-FILE         PIC X(1).
    88 NO-VALUE         VALUE LOW-VALUE.
    88 NO-PAR           VALUE "N".
    88 YES              VALUE "Y".

10 READ-ATTRIBUTES     PIC X(1).
    88 NO-VALUE         VALUE LOW-VALUE.
    88 NO-PAR           VALUE "N".
    88 YES              VALUE "Y".
10 CHANGE-ATTRIBUTES   PIC X(1).
    88 NO-VALUE         VALUE LOW-VALUE.
    88 NO-PAR           VALUE "N".
    88 YES              VALUE "Y".
10 DELETE-FILE         PIC X(1).
    88 NO-VALUE         VALUE LOW-VALUE.
    88 NO-PAR           VALUE "N".
    88 YES              VALUE "Y".
10 INSERT-DATA-UNIT    PIC X(1).
    88 NO-VALUE         VALUE LOW-VALUE.
    88 NO-PAR           VALUE "N".
    88 YES              VALUE "Y".
10 ERASE-DATA-UNIT     PIC X(1).
    88 NO-VALUE         VALUE LOW-VALUE.
    88 NO-PAR           VALUE "N".
    88 YES              VALUE "Y".
10 LEGAL-QUALIFICATION PIC X(80).
    88 UNCHANGED        VALUE LOW-VALUE.
10 CODE-CHARACTER-SET  PIC X(8).
    88 STD              VALUE LOW-VALUE.

```

*

```

05 COMPRESS            PIC X(1).
    88 NONE            VALUE LOW-VALUE.
    88 BYTE-REPETITION VALUE "B".
    88 ZIP             VALUE "Z".
    88 NONE-ENCRYES   VALUE "E".

```

	88	BYTE-ENCRYES		VALUE	"F".	
	88	ZIP-ENCRYES		VALUE	"G".	
	88	NONE-ENCRODI		VALUE	"O".	
	88	BYTE-ENCRODI		VALUE	"P".	
	88	ZIP-ENCRODI		VALUE	"R".	
*						
	05	WRITE-MODE	PIC	X(1).		
		88	REPLACE-FILE	VALUE	LOW-VALUE.	
		88	NEW-FILE	VALUE	"N".	
		88	EXTEND-FILE	VALUE	"E".	
*						
	05	DATA-TYPE	PIC	X(1).		
		88	CHARACTER-TYPE	VALUE	LOW-VALUE.	
		88	BINARY-TYPE	VALUE	"B".	
		88	USER-TYPE	VALUE	"C".	
		88	CHAR-STD-TYPE	VALUE	"D".	
		88	BIN-STD-TYPE	VALUE	"E".	
		88	NOT-SPECIFIED	VALUE	HIGH-VALUE.	
*						
	05	PRIORITY	PIC	X(1).		
		88	NORMAL	VALUE	LOW-VALUE.	
		88	HIGH	VALUE	"H".	
		88	LOW	VALUE	"L".	
*						
	05	START-TIME.				
		10	EARLIEST-DATE	PIC	X(8).	
			88	TODAY	VALUE	LOW-VALUE.
			88	TOMORROW	VALUE	"T".
		10	EARLIEST-TIME	PIC	X(5).	
*						
	05	CANCEL-PARAMETER.				
		10	CANCEL-DESIRED	PIC	X(1).	
			88	NO-CANCEL	VALUE	LOW-VALUE.
			88	YES	VALUE	"Y".
		10	CANCEL-DATE	PIC	X(8).	
			88	TODAY	VALUE	LOW-VALUE.
			88	TOMORROW	VALUE	"T".
		10	CANCEL-TIME	PIC	X(5).	
*						
	05	RECORD-SIZE	PIC	X(5).		
		88	NOT-SPECIFIED	VALUE	LOW-VALUE.	
*						
	05	RECORD-FORMAT	PIC	X(1).		
		88	STD	VALUE	LOW-VALUE.	
		88	VARIABLE	VALUE	"V".	
		88	FIXED	VALUE	"F".	
		88	UNDEFINED	VALUE	"U".	

An NCOPY macro does not modify any values in the structure FT-NCOPY-LIST.

The fields are to be written with left-justified characters and filled with right-justified blanks (default for the COBOL-MOVE statement for character strings).

Fields which should not contain any specifications are to be assigned LOW-VALUE.

If a parameter is not entered, the default values are generated as in the TRANSFER-FILE(NCOPY) command.

The version specification at the beginning of the structure FT-NCOPY-LIST serves to identify the COPY element and must not be overwritten.

With partners of the type MSP/MVS (REMOTE-SYNTAX:"3"), the filename is to be set in quotation marks if it contains the user ID (prefix).

All other values which should be set in quotation marks at the command interface do not have these quotation marks in the programming interface. For historical reasons, the value is converted to lower-case when USER-DEF-ADMISSION is specified without quotation marks. If the USER-DEF-ADMISSION is to be kept in its original form, it must be enclosed in quotation marks.

Passwords with integer values must be entered in binary form.

Specifications for the remote system which are taken by default from the specifications for the local system using *SAME in the TRANSFER-FILE(NCOPY) command must be explicitly entered at the COBOL interface.

Description of the data fields

Most parameters for FT-NCOPY-LIST have the same names and functions as the corresponding operands for the TRANSFER-FILE(NCOPY) command. Therefore, only the deviations will be explained here. Please refer to the corresponding command description in the user manual for the other explanations.

- In addition to the options offered at the command interface, you can also address filenames and library names in COBOL programs via link names. To do this, use the field LINK-NAME in FT-NCOPY-LIST.
- Only the link name or the filename/library name (not both) may be specified for a system (local or remote).
- The remote filename or library name may only be specified via the link name if the local and remote systems are identical (file transfer with the same computer). If this rule is not observed, openFT will not recognize it and this will lead to undesired results.
- If the file or library is specified using its link name, a missing specification for the remote file will be filled using the local filename. A link name which is not assigned at execution time leads to the message `FILE UNKNOWN`.
- The field USER-DEF-ADMISSION in FT-NCOPY-LIST is only required, if the add-on product openFT-AC is used in the system and an admission profile is being addressed. In this case USER-DEF-ADMISSION can be used to enter the transfer admission required for this admission profile (the TRANSFER-ADMISSION in the command).
- The field REMOTE-SYNTAX controls the syntax check of the entries for the remote system. It corresponds to the entry which you can make in front of the parentheses at the beginning of the REMOTE-PARAMETER in the command. If this field is also assigned the value LOW-VALUE or "3", openFT runs a syntax check of these specifications according to the BS2000 or MVS syntax rules.
- The options for the field START-TIME correspond to those of the command operand START. EARLIEST-DATE sets the earliest date and EARLIEST-TIME sets the earliest time for the start of the file transfer.
- CANCEL-PARAMETER corresponds to the command operand CANCEL. You can use CANCEL-DESIRED to cancel your request while CANCEL-DATE and CANCEL-TIME allow you to specify the required time and date of cancellation.

Example NCOPY

```

*****
*   EXAMPLE:                                                                    *
*   SUBMISSION OF A FILE TRANSFER REQUEST                                       *
*                                                                                   *
*   /TRANSFER=FILE PARTNER=VAR001,TRANS=TO,   -                               *
*   /      LOC=(FILE=LOCFILE,TRANS=(USERID,ACCOUNT,'PASSWORD')), - *
*   /      REM=(FILE=REMFIL,TRANS=(USERID,ACCOUNT,'PASSWORD')) *
*                                                                                   *
*   FROM A COBOL PROGRAM                                                         *
*****
  IDENTIFICATION DIVISION.
  PROGRAM-ID. NCO.
*
  ENVIRONMENT DIVISION.
*
  CONFIGURATION SECTION.
  SPECIAL-NAMES.
    TERMINAL IS TERM.
*
  DATA DIVISION.
  WORKING-STORAGE SECTION.
  77 MAIN-RCODE-STRING                PIC -ZZZZ9.
  77 SUB-RCODE-STRING                 PIC -ZZZZ9.
  COPY FTNCOPY OF FTLIB.
  COPY FTRETC OF FTLIB.
*
  PROCEDURE DIVISION.
  STEUER SECTION.
  ST-01.
*
*SPECIFY PARAMETERS.
  MOVE LOW-VALUE TO USER-PARAMETERS IN FT-NCOPY-LIST.
  MOVE "VAR001" TO PARTNER-NAME.
  MOVE "T" TO TRANSFER-DIRECTION.
  MOVE "LOCFILE" TO FILE-NAME IN LOCAL-PARAMETER.
  MOVE "USERID" TO USER-ID IN TRANSFER-ADMISSION
    IN LOCAL-PARAMETER.
  MOVE "ACCOUNT" TO ACCOUNT IN TRANSFER-ADMISSION
    IN LOCAL-PARAMETER.
  MOVE "'PASSWORD'" TO PASSWORD IN TRANSFER-ADMISSION
    IN LOCAL-PARAMETER.
  MOVE "REMFIL" TO FILE-NAME IN REMOTE-PARAMETER.
  MOVE "USERID" TO USER-ID IN TRANSFER-ADMISSION
    IN REMOTE-PARAMETER.
  MOVE "ACCOUNT" TO ACCOUNT IN TRANSFER-ADMISSION
    IN REMOTE-PARAMETER.

```

```
        MOVE "'PASSWORD'" TO PASSWORD IN TRANSFER-ADMISSION
                                IN REMOTE-PARAMETER.
*
*NCOPY-CALLING.
        CALL "NCOPY" USING FT-NCOPY-LIST FT-RETURN-INFO.
*
*RESULT HANDLING.
        IF OKAY IN MAIN-RETURN-CODE
            DISPLAY "NCOPY-REQUEST ACCEPTED, TID= " TRANSFER-ID
                UPON TERM
        ELSE
            MOVE MAIN-RETURN-CODE TO MAIN-RCODE-STRING
            MOVE SUB-RETURN-CODE TO SUB-RCODE-STRING
            DISPLAY "NCOPY-REQUEST REJECTED" UPON TERM
            DISPLAY "MAIN-RETURN-CODE: " MAIN-RCODE-STRING
                " SUB-RETURN-CODE: " SUB-RCODE-STRING
                UPON TERM.
ST-99.
        STOP RUN.
```

3.1.3 NDEL - Delete remote file

The macro CALL "NDEL"... can be used to delete a file in a partner system. The functionality corresponds to that of the command DELETE-REMOTE-FILE.

Macro

The function can be called as follows:

```
CALL "NDEL" USING FT-NDEL-LIST FT-RETURN-INFO.
```

FT-NDEL-LIST

The range FT-NDEL-LIST describes the parameter list for the NDEL macro. FT-NDEL-LIST must be defined in the WORKING-STORAGE SECTION and can be copied to there with the statement

```
COPY FTNDEL OF linkname.
```

Before the first NDEL macro, the parameter fields should be deleted with the statement

```
MOVE LOW-VALUE TO USER-PARAMETERS IN FT-NDEL-LIST.
```

Before executing another NDEL macro, you must fill the desired fields. If a parameter is not specified or the default value is to be used, this field must be assigned the value LOW-VALUE.

FT-NDEL-LIST is defined as follows:

```

01 FT-NDEL-LIST.
*
  02 FILLER                PIC X(4)  VALUE "1000".
*
  02 USER-PARAMETERS.
*
    05 PARTNER-NAME        PIC X(200).
*
    05 FILE-NAME           PIC X(512).
      88 NOT-SPECIFIED          VALUE LOW-VALUE.
*
    05 FILE-PASSWORD       PIC X(64).
      88 NONE                   VALUE LOW-VALUE.
    05 FILE-PASSWD-ATTR    PIC X(1).
      88 GRAPHIC                 VALUE LOW-VALUE.
      88 OCTET                   VALUE "0".
*
    05 USER-DEF-ADMISSION  PIC X(67).
      88 NONE                   VALUE LOW-VALUE.
    05 USER-DEF-ADM-ATTR   PIC X(1).
      88 GRAPHIC                 VALUE LOW-VALUE.
      88 OCTET                   VALUE "0".
    05 TRANSFER-ADMISSION.
      10 USER-ID             PIC X(67).
      10 ACCOUNT            PIC X(64).
        88 NONE              VALUE LOW-VALUE.
      10 PASSWORD          PIC X(64).
        88 NONE              VALUE LOW-VALUE.
    05 USER-PASSWD-ATTR    PIC X(1).
      88 GRAPHIC                 VALUE LOW-VALUE.
      88 OCTET                   VALUE "0".

```

The version specification at the beginning of the structure FT-NCOPY-LIST serves to identify the COPY element and must not be overwritten.

The NDEL macro does not change any values in the structure FT-NDEL-LIST.

The fields are to be written with left-justified characters and filled with right-justified blanks (default for the COBOL-MOVE statement for character strings).

Fields which should not contain any specifications are to be assigned LOW-VALUE.

If a parameter is not entered, the default values are generated as in the DELETE-REMOTE-FILE command.

All other values which should be set in quotation marks at the command interface do not have these quotation marks in the programming interface.

Passwords with integer values must be entered in binary form.

For the assignment of keywords, e.g. *NOT-SPECIFIED and *NONE, use the values which are set in the corresponding 88 step structure.

Description of the data fields

The parameters for FT-NDEL-LIST have the same names and functions as the operands for the command DELETE-REMOTE-FILE.

The parameter USER-DEF-ADM-ATTR is used to determine how the entry for USER-DEF-ADMISSION is to be interpreted. With the value GRAPHIC, the entry for USER-DEF-ADMISSION is interpreted as a printable character and converted for transfer to a system which does not use EBCDIC. With the value OCTET, the entry for USER-DEF-ADMISSION is interpreted as binary information and not converted.

Example NDEL

```

*****
*   EXAMPLE:                                     *
* /DELETE-REMOTE-FILE PARTNER=VAR001,FILE-NAME=REMOTEFILE,- *
* /                               TRANS=(USID,ACCOUNT,'PASSWORD') *
* FROM A COBOL PROGRAM                                     *
*****
  IDENTIFICATION DIVISION.
  PROGRAM-ID. TESTNDEL.
*
  ENVIRONMENT DIVISION.
*
  CONFIGURATION SECTION.
  SPECIAL-NAMES.
    TERMINAL IS TERM.
*
  DATA DIVISION.
  WORKING-STORAGE SECTION.
  77 MAIN-RCODE-STRING                               PIC -ZZZZ9.
  77 SUB-RCODE-STRING                               PIC -ZZZZ9.
  COPY FTNDEL OF FTLIB.
  COPY FTRETC OF FTLIB.
*
  PROCEDURE DIVISION.
  STEUER SECTION.
  ST-01.
*
  *SPECIFY PARAMETERS.
    MOVE LOW-VALUE TO USER-PARAMETERS.
    MOVE "VAR001" TO PARTNER-NAME.
    MOVE "REMOTEFILE" TO FILE-NAME.
    MOVE "USERID" TO USER-ID.
    MOVE "ACCOUNT" TO ACCOUNT.
    MOVE "'PASSWORD'" TO PASSWORD.
*
  *NDEL-CALLING.
    CALL "NDEL" USING FT-NDEL-LIST FT-RETURN-INFO.
*

```

```
*RESULT-HANDLING.  
  IF OKAY IN MAIN-RETURN-CODE  
  THEN  
    DISPLAY "NDEL OKAY" UPON TERM  
  ELSE  
    MOVE MAIN-RETURN-CODE TO MAIN-RCODE-STRING  
    MOVE SUB-RETURN-CODE TO SUB-RCODE-STRING  
    DISPLAY "NDEL REJECTED" UPON TERM  
    DISPLAY "MAIN-RETURN-CODE: " MAIN-RCODE-STRING  
      " SUB-RETURN-CODE: " SUB-RCODE-STRING  
      UPON TERM.  
ST-99.  
  STOP RUN.
```

3.1.4 NLMOD - Modify local FT file attribute

The macro CALL "NLMOD"... can be used to modify the FTAM attributes of a file in the local system and adapt the attributes to make them available for file transfer or file management request with an FTAM partner. The functionality corresponds to that of the command MODIFY-FILE-FT-ATTRIBUTES.

In addition to the options offered by the openFT for BS2000 command interface, the filenames can also be addressed via their linknames.

Macro

The function can be called as follows:

```
CALL "NLMOD" USING FT-NLMOD-LIST FT-RETURN-INFO.
```

FT-NLMOD-LIST

The range FT-NLMOD-LIST describes the parameter list for the NLMOD macro. FT-NLMOD-LIST must be defined in the WORKING-STORAGE SECTION and can be copied to there with the statement

```
COPY FTNLMOD OF linkname.
```

Before the first NLMOD macro, the parameter fields should be deleted with the statement

```
MOVE LOW-VALUE TO USER-PARAMETERS IN FT-NLMOD-LIST.
```

Before executing another NLMOD macro, you must fill the desired fields. If a parameter is not specified or the default value is to be used, this field must be assigned the value LOW-VALUE.

FT-NLMOD-LIST is defined as follows:

```

01 FT-NLMOD-LIST.
*
  02 FILLER                PIC X(4) VALUE "V520".
*
  02 USER-PARAMETERS.
*
    05 FILE-NAME           PIC X(54).
      88 NOT-SPECIFIED     VALUE LOW-VALUE.
    05 LINK-NAME           PIC X(8).
      88 NOT-SPECIFIED     VALUE LOW-VALUE.
*
    05 FILE-PASSWORD       PIC X(4).
      88 NONE              VALUE LOW-VALUE.
    05 FILE-PASSWD-ATTR    PIC X(1).
      88 GRAPHIC           VALUE LOW-VALUE.
      88 OCTET             VALUE "0".
*
    05 PERMITTED-ACTIONS   PIC X(1).
      88 UNCHANGED         VALUE LOW-VALUE.
      88 PARAMETER         VALUE "P".
    05 READ-FILE           PIC X(1).
      88 NO-VALUE          VALUE LOW-VALUE.
      88 NO-PAR            VALUE "N".
      88 YES               VALUE "Y".
    05 INSERT-DATA-UNIT    PIC X(1).
      88 NO-VALUE          VALUE LOW-VALUE.
      88 NO-PAR            VALUE "N".
      88 YES               VALUE "Y".
    05 REPLACE-FILE        PIC X(1).
      88 NO-VALUE          VALUE LOW-VALUE.
      88 NO-PAR            VALUE "N".
      88 YES               VALUE "Y".
    05 EXTEND-FILE         PIC X(1).
      88 NO-VALUE          VALUE LOW-VALUE.
      88 NO-PAR            VALUE "N".
      88 YES               VALUE "Y".
    05 ERASE-DATA-UNIT     PIC X(1).
      88 NO-VALUE          VALUE LOW-VALUE.
      88 NO-PAR            VALUE "N".
      88 YES               VALUE "Y".
    05 READ-ATTRIBUTES     PIC X(1).
      88 NO-VALUE          VALUE LOW-VALUE.
      88 NO-PAR            VALUE "N".
      88 YES               VALUE "Y".

```

```

05 CHANGE-ATTRIBUTES      PIC X(1).
    88 NO-VALUE            VALUE LOW-VALUE.
    88 NO-PAR              VALUE "N".
    88 YES                  VALUE "Y".
05 DELETE-FILE            PIC X(1).
    88 NO-VALUE            VALUE LOW-VALUE.
    88 NO-PAR              VALUE "N".
    88 YES                  VALUE "Y".

*

05 TRANSFER-ATTRIBUTES.
    10 TR-ATTRIBUTES      PIC X(1).
        88 UNCHANGED      VALUE LOW-VALUE.
        88 PARAMETER      VALUE "P".
    10 DATA-TYPE         PIC X(1).
        88 UNCHANGED      VALUE LOW-VALUE.
        88 BINARY-DATA    VALUE "B".
        88 CHARACTER-TYPE VALUE "C".
    10 CHARACTER-SET      PIC X(1).
        88 NO-VALUE      VALUE LOW-VALUE.
        88 GRAPHIC       VALUE "R".
        88 GENERAL        VALUE "E".
        88 IA5            VALUE "I".
        88 VISIBLE        VALUE "V".
    10 RECORD-FORMAT      PIC X(1).
        88 UNCHANGED      VALUE LOW-VALUE.
    10 RECORD-SIZE        PIC X(5).
        88 UNCHANGED      VALUE LOW-VALUE.

```

The version specification at the beginning of the structure FT-NCOPY-LIST serves to identify the COPY element and must not be overwritten.

The NLMOD macro does not change any values in the structure FT-NLMOD-LIST.

The fields are to be written with left-justified characters and filled with right-justified blanks (default for the COBOL-MOVE statement for character strings).

Fields which should not contain any specifications are to be assigned LOW-VALUE.

If a parameter is not entered, the default values are generated as in the MODIFY-FILE-FT-ATTRIBUTES command.

All other values which should be set in quotation marks at the command interface do not have these quotation marks in the programming interface.

Passwords with integer values must be entered in binary form.

For the assignment of keywords, e.g. *BINARY, *VISIBLE and *YES, use the values which are set in the corresponding 88 step structure.

Description of the data fields

The parameters for FT-NLMOD-LIST have the same names and functions as the corresponding operands for the command MODIFY-FILE-FT-ATTRIBUTES. Please refer to the command descriptions in the user manual.

In addition to the options offered at the command interface, the COBOL program allows you to address files and libraries by their linkname. This is done using the field LINK-NAME in the FT-NLMOD-LIST.

Only the link name or the filename/library name (not both) may be specified.

A link name which is not assigned at execution time leads to the message

FILE UNKNOWN.

FILE-PASSWORD can be used to enter any necessary file passwords.

Example NLMOD

```

*****
*   EXAMPLE:
*   /MODIFY-FILE-FT-ATTRIBUTES FILE-NAME=LOCALFILE,-
*   /                               TRANSFER-ATTRIBUTES=(DATA-TYPE=*BINARY)
*   FROM A COBOL PROGRAM
*****
  IDENTIFICATION DIVISION.
  PROGRAM-ID. TESTNLMOD.
*
  ENVIRONMENT DIVISION.
*
  CONFIGURATION SECTION.
  SPECIAL-NAMES.
    TERMINAL IS TERM.
*
  DATA DIVISION.
  WORKING-STORAGE SECTION.
  77 MAIN-RCODE-STRING                PIC -ZZZZ9.
  77 SUB-RCODE-STRING                 PIC -ZZZZ9.
  COPY FTNLMOD OF FTLIB.
  COPY FTRETC OF FTLIB.
*
  PROCEDURE DIVISION.
  STEUER SECTION.
  ST-01.
*
  *SPECIFY PARAMETERS.
    MOVE LOW-VALUE TO USER-PARAMETERS.
    MOVE "LOCALFILE" TO FILE-NAME.
    SET BINARY-DATA IN DATA-TYPE TO TRUE.
*
  *NLMOD-CALLING.
    CALL "NLMOD" USING FT-NLMOD-LIST FT-RETURN-INFO.
*
  *RESULT-HANDLING.
    IF OKAY IN MAIN-RETURN-CODE
    THEN
      DISPLAY "NLMOD OKAY" UPON TERM

```



```
ELSE
  MOVE MAIN-RETURN-CODE TO MAIN-RCODE-STRING
  MOVE SUB-RETURN-CODE TO SUB-RCODE-STRING
  DISPLAY "NLMOD REJECTED" UPON TERM
  DISPLAY "MAIN-RETURN-CODE: " MAIN-RCODE-STRING
    " SUB-RETURN-CODE: " SUB-RCODE-STRING
    UPON TERM.
ST-99.
  STOP RUN.
```

3.1.5 NLSHOW - Display local FT file attributes

The macro CALL "NLSHOW"... can be used to view the FTAM attributes of a file in the local system. The functionality corresponds to that of the command SHOW-FILE-FT-ATTRIBUTES.

The user can select between three variants for the output of attributes:

- display the filename,
- display a default selection,
- display all file attributes.

The information can be output on the screen or to a file.

In addition to the options offered by the openFT for BS2000 command interface, the files can also be addressed via their linkname.

Macro

The function can be called as follows:

```
CALL "NLSHOW" USING FT-NLSHOW-LIST FT-RETURN-INFO.
```

FT-NLSHOW-LIST

The range FT-NLSHOW-LIST describes the parameter list for the NLSHOW macro. FT-NLSHOW-LIST must be defined in the WORKING-STORAGE SECTION and can be copied to there with the statement

```
COPY FTNLSHOW OF linkname.
```

Before the first NLSHOW macro the parameter fields should be deleted with the statement

```
MOVE LOW-VALUE TO USER-PARAMETERS IN FT-NLSHOW-LIST.
```

Before executing another NLSHOW macro you must fill the desired fields. If a parameter is not specified or the default value is to be used, this field must be assigned the value LOW-VALUE.

FT-NLSHOW-LIST is defined as follows:

```

01 FT-NLSHOW-LIST.
*
  02 FILLER                PIC X(4)  VALUE "V500".
*
  02 USER-PARAMETERS.
*
    05 FILE-NAME           PIC X(54).
      88 NOT-SPECIFIED     VALUE LOW-VALUE.
    05 LINK-NAME          PIC X(8).
      88 NOT-SPECIFIED     VALUE LOW-VALUE.
*
    05 INFORMATION        PIC X(1).
      88 STD                VALUE LOW-VALUE.
      88 ALL-ATTRIBUTES    VALUE "A".
      88 ONLY-NAME         VALUE "O".
*
    05 OUTPUT-PAR         PIC X(1).
      88 SYSOUT             VALUE LOW-VALUE.
88 SYSLST                 VALUE "L".
      88 SYSOUTCSV          VALUE "O".
      88 SYSLSTCSV         VALUE "C".

```

The version specification at the beginning of the structure FT-NCOPY-LIST serves to identify the COPY element and must not be overwritten.

The NLSHOW macro does not change any values in the structure FT-NLSHOW-LIST.

The fields are to be written with left-justified characters and filled with right-justified blanks (default for the COBOL-MOVE statement for character strings).

Fields which should not contain any specifications are to be assigned LOW-VALUE.

If a parameter is not specified, the default values are used as specified in the command SHOW-FILE-FT-ATTRIBUTES.

For the assignment of keywords, e.g. *ONLY-NAME and *SYSLST, use the values which are set in the corresponding 88 step structure.

Description of the data fields

The parameters for FT-NLSHOW-LIST have the same names and functions as the corresponding operands for the command SHOW-FILE-FT-ATTRIBUTES. Please refer to the command descriptions in the user manual.

In addition to the options offered at the command interface, the COBOL program allows you to address files and libraries by their linkname. This is done using the field LINK-NAME in the FT-NLMOD-LIST.

Only the link name or the filename/library name (not both) may be specified.

A link name which is not assigned at execution time leads to the message

FILE UNKNOWN.

Example NLSHOW

```

*****
*   EXAMPLE:                                     *
* /SHOW=FILE=FT-ATTRIBUTES FILE-NAME=LOCALFILE,- *
* /                                     INFORMATION=*ALL-ATTRIBUTES *
* FROM A COBOL PROGRAM                         *
*****
  IDENTIFICATION DIVISION.
  PROGRAM-ID. TESTNLSHOW.
*
  ENVIRONMENT DIVISION.
*
  CONFIGURATION SECTION.
  SPECIAL-NAMES.
    TERMINAL IS TERM.
*
  DATA DIVISION.
  WORKING-STORAGE SECTION.
  77 MAIN-RCODE-STRING                       PIC -ZZZZ9.
  77 SUB-RCODE-STRING                         PIC -ZZZZ9.
  COPY FTNLSHOW OF FTLIB.
  COPY FTRETC OF FTLIB.
*
  PROCEDURE DIVISION.
  STEUER SECTION.
  ST-01.
*
  *SPECIFY-PARAMETERS.
    MOVE LOW-VALUE TO USER-PARAMETERS.
    MOVE "LOCALFILE" TO FILE-NAME.
    SET ALL-ATTRIBUTES IN INFORMATION TO TRUE.
*
  *NLSHOW-CALLING.
    CALL "NLSHOW" USING FT-NLSHOW-LIST FT-RETURN-INFO.
*
  *RESULT-HANDLING.
    IF OKAY IN MAIN-RETURN-CODE
    THEN
      DISPLAY "NLSHOW OKAY" UPON TERM

```

```
ELSE
  MOVE MAIN-RETURN-CODE TO MAIN-RCODE-STRING
  MOVE SUB-RETURN-CODE TO SUB-RCODE-STRING
  DISPLAY "NLSHOW REJECTED" UPON TERM
  DISPLAY "MAIN-RETURN-CODE: " MAIN-RCODE-STRING
    " SUB-RETURN-CODE: " SUB-RCODE-STRING
    UPON TERM.
ST-99.
STOP RUN.
```

3.1.6 NMOD - Modify remote file attributes

The macro CALL "NMOD"... can be used to modify the attributes of a file in an FT partner system. The functionality corresponds to that of the command MODIFY-REMOTE-FILE-ATTRIBUTES.

Macro

The function can be called as follows:

```
CALL "NMOD" USING FT-NMOD-LIST FT-RETURN-INFO.
```

FT-NMOD-LIST

The range FT-NMOD-LIST describes the parameter list for the NMOD macro. FT-NMOD-LIST must be defined in the WORKING-STORAGE SECTION and can be copied to there with the statement

```
COPY FTNMOD OF linkname.
```

Before the first NMOD macro the parameter fields should be deleted with the statement

```
MOVE LOW-VALUE TO USER-PARAMETERS IN FT-NMOD-LIST.
```

Before executing another NMOD macro you must fill the desired fields. If a parameter is not specified or the default value is to be used, this field must be assigned the value LOW-VALUE.

FT-NMOD-LIST is defined as follows:

```

01 FT-NMOD-LIST.
*
  02 FILLER                PIC X(4)  VALUE "1000".
*
  02 USER-PARAMETERS.
*
    05 PARTNER-NAME        PIC X(200).
*
    05 FILE-NAME           PIC X(512).
      88 NOT-SPECIFIED      VALUE LOW-VALUE.
*
    05 FILE-PASSWORD       PIC X(64).
      88 NONE                VALUE LOW-VALUE.
    05 FILE-PASSWD-ATTR    PIC X(1).
      88 GRAPHIC             VALUE LOW-VALUE.
      88 OCTET               VALUE "0".
*
    05 USER-DEF-ADMISSION  PIC X(67).
      88 NONE                VALUE LOW-VALUE.
    05 USER-DEF-ADM-ATTR   PIC X(1).
      88 GRAPHIC             VALUE LOW-VALUE.
      88 OCTET               VALUE "0".
    05 TRANSFER-ADMISSION.
      10 USER-ID            PIC X(67).
      10 ACCOUNT            PIC X(64).
        88 NONE              VALUE LOW-VALUE.
      10 PASSWORD          PIC X(64).
        88 NONE              VALUE LOW-VALUE.
    05 USER-PASSWD-ATTR    PIC X(1).
      88 GRAPHIC             VALUE LOW-VALUE.
      88 OCTET               VALUE "0".
*
    05 NEW-NAME            PIC X(512).
      88 SAME-NAME          VALUE LOW-VALUE.
*
    05 FILE-AVAILABILITY    PIC X(1).
      88 UNCHANGED          VALUE LOW-VALUE.
      88 IMMEDIATE          VALUE "I".
      88 DEFERRED           VALUE "D".
*
    05 STORAGE-ACCOUNT     PIC X(40).
      88 UNCHANGED          VALUE LOW-VALUE.
*
    05 FUTURE-FILE-SIZE    PIC X(4).
      88 UNCHANGED          VALUE LOW-VALUE.

```



```

*
05 ACCESS-MODE          PIC X(1).
    88 UNCHANGED          VALUE LOW-VALUE.
    88 REPLACE-ALL-BY    VALUE "R".
    88 ADD-PAR            VALUE "A".
05 READ-FILE           PIC X(1).
    88 NO-VALUE          VALUE LOW-VALUE.
    88 NO-PAR            VALUE "N".
    88 YES                VALUE "Y".
05 INSERT-DATA-UNIT    PIC X(1).
    88 NO-VALUE          VALUE LOW-VALUE.
    88 NO-PAR            VALUE "N".
    88 YES                VALUE "Y".
05 REPLACE-FILE        PIC X(1).
    88 NO-VALUE          VALUE LOW-VALUE.
    88 NO-PAR            VALUE "N".
    88 YES                VALUE "Y".
05 EXTEND-FILE         PIC X(1).
    88 NO-VALUE          VALUE LOW-VALUE.
    88 NO-PAR            VALUE "N".
    88 YES                VALUE "Y".
05 ERASE-DATA-UNIT     PIC X(1).
    88 NO-VALUE          VALUE LOW-VALUE.
    88 NO-PAR            VALUE "N".
    88 YES                VALUE "Y".
05 READ-ATTRIBUTES     PIC X(1).
    88 NO-VALUE          VALUE LOW-VALUE.
    88 NO-PAR            VALUE "N".
    88 YES                VALUE "Y".
05 CHANGE-ATTRIBUTES  PIC X(1).
    88 NO-VALUE          VALUE LOW-VALUE.
    88 NO-PAR            VALUE "N".
    88 YES                VALUE "Y".
05 DELETE-FILE         PIC X(1).
    88 NO-VALUE          VALUE LOW-VALUE.
    88 NO-PAR            VALUE "N".
    88 YES                VALUE "Y".
*
05 LEGAL-QUALIFICATION PIC X(80).
    88 UNCHANGED          VALUE LOW-VALUE.

```

The version specification at the beginning of the structure FT-NCOPY-LIST serves to identify the COPY element and must not be overwritten.

The NMOD macro does not change any values in the structure FT-NMOD-LIST.

The fields are to be written with left-justified characters and filled with right-justified blanks (default for the COBOL-MOVE statement for character strings).

Fields which should not contain any specifications are to be assigned LOW-VALUE.

If a parameter is not entered, the default values are generated as in the MODIFY-REMOTE-FILE-ATTRIBUTES command.

All other values which should be set in quotation marks at the command interface do not have these quotation marks in the programming interface.

Passwords with integer values must be entered in binary form.

For the assignment of keywords, e.g. *NOT-SPECIFIED and *YES, use the values which are set in the corresponding 88 step structure.

Description of the data fields

The parameters for FT-NMOD-LIST have the same names and functions as the operands for the command MODIFY-REMOTE-FILE-ATTRIBUTES. Please refer to the corresponding command description in the user manual.

The parameter USER-DEF-ADM-ATTR is used to determine how the entry for USER-DEF-ADMISSION is to be interpreted. With the value GRAPHIC, the entry for USER-DEF-ADMISSION is interpreted as a printable character and converted for transfer to a system which does not use EBCDIC. With the value OCTET, the entry for USER-DEF-ADMISSION is interpreted as binary information and not converted.

Example NMOD

```

*****
*   EXAMPLE:                                     *
* /MODIFY-REMOTE-FILE-ATTRIBUTES -             *
* /      PARTNER=VAR001,FILE-NAME=OLDFILE,NEW-NAME=NEWFILE,- *
* /      TRANS=(USID,ACCOUNT,'PASSWORD')      *
* FROM A COBOL PROGRAM                          *
*****
IDENTIFICATION DIVISION.
PROGRAM-ID. TESTNMOD.
*
ENVIRONMENT DIVISION.
*
CONFIGURATION SECTION.
SPECIAL-NAMES.
    TERMINAL IS TERM.
*
DATA DIVISION.
WORKING-STORAGE SECTION.
77 MAIN-RCODE-STRING          PIC -ZZZZ9.
77 SUB-RCODE-STRING          PIC -ZZZZ9.
COPY FTNMOD OF FTLIB.
COPY FTRETC OF FTLIB.
*
PROCEDURE DIVISION.
STEUER SECTION.
ST-01.
*
*SPECIFY-PARAMETERS.
    MOVE LOW-VALUE TO USER-PARAMETERS.
    MOVE "VAR001" TO PARTNER-NAME.
    MOVE "OLDFILE" TO FILE-NAME.
    MOVE "NEWFILE" TO NEW-NAME.
    MOVE "USERID" TO USER-ID.
    MOVE "ACCOUNT" TO ACCOUNT.
    MOVE "'PASSWORD'" TO PASSWORD.
*
*NMOD-CALLING.
    CALL "NMOD" USING FT-NMOD-LIST FT-RETURN-INFO.
*

```

```
*RESULT-HANDLING.  
  IF OKAY IN MAIN-RETURN-CODE  
  THEN  
    DISPLAY "NMOD OKAY" UPON TERM  
  ELSE  
    MOVE MAIN-RETURN-CODE TO MAIN-RCODE-STRING  
    MOVE SUB-RETURN-CODE TO SUB-RCODE-STRING  
    DISPLAY "NMOD REJECTED" UPON TERM  
    DISPLAY "MAIN-RETURN-CODE: " MAIN-RCODE-STRING  
      " SUB-RETURN-CODE: " SUB-RCODE-STRING  
      UPON TERM.  
ST-99.  
  STOP RUN.
```

3.1.7 NSHOW - Display remote file attributes

The macro CALL "NSHOW"... can be used to view the attributes of a file or directory in an FT partner system. The functionality corresponds to that of the command SHOW-REMOTE-FILE-ATTRIBUTES.

The user can select between three variants for the output of attributes:

- display the filename,
- display a default selection,
- display all file attributes.

The information can be output on the screen or to a file.

Macro

The function can be called as follows:

```
CALL "NSHOW" USING FT-NSHOW-LIST FT-RETURN-INFO.
```

FT-NSHOW-LIST

The range FT-NSHOW-LIST describes the parameter list for the NSHOW macro. FT-NSHOW-LIST must be defined in the WORKING-STORAGE SECTION and can be copied to there with the statement

```
COPY FTNSHOW OF linkname.
```

Before the first NSHOW macro the parameter fields should be deleted with the statement

```
MOVE LOW-VALUE TO USER-PARAMETERS IN FT-NSHOW-LIST.
```

Before executing another NSHOW macro you must fill the desired fields. If a parameter is not specified or the default value is to be used, this field must be assigned the value LOW-VALUE.

FT-NSHOW-LIST is defined as follows:

```

01 FT-NSHOW-LIST.
*
  02 FILLER                                PIC X(4)  VALUE "1000".
*
  02 USER-PARAMETERS.
*
    05 PARTNER-NAME                        PIC X(200).
*
    05 FILE-NAME                            PIC X(512).
      88 NOT-SPECIFIED                      VALUE LOW-VALUE.
    05 DIRECTORY                            PIC X(512).
      88 NOT-SPECIFIED                      VALUE LOW-VALUE.
*
    05 FILE-PASSWORD                        PIC X(64).
      88 NONE                               VALUE LOW-VALUE.
    05 FILE-PASSWD-ATTR                     PIC X(1).
      88 GRAPHIC                            VALUE LOW-VALUE.
      88 OCTET                              VALUE "0".
*
    05 USER-DEF-ADMISSION                   PIC X(67).
      88 NONE                               VALUE LOW-VALUE.
    05 USER-DEF-ADM-ATTR                    PIC X(1).
      88 GRAPHIC                            VALUE LOW-VALUE.
      88 OCTET                              VALUE "0".
    05 TRANSFER-ADMISSION.
      10 USER-ID                            PIC X(67).
      10 ACCOUNT                            PIC X(64).
        88 NONE                             VALUE LOW-VALUE.
      10 PASSWORD                           PIC X(64).
        88 NONE                             VALUE LOW-VALUE.
    05 USER-PASSWD-ATTR                     PIC X(1).
      88 GRAPHIC                            VALUE LOW-VALUE.
      88 OCTET                              VALUE "0".
*
    05 INFORMATION                          PIC X(1).
      88 STD                                VALUE LOW-VALUE.
      88 ALL-ATTRIBUTES                     VALUE "A".
      88 ONLY-NAMES                         VALUE "0".
*
    05 OUTPUT-PAR                           PIC X(1).
      88 SYSOUT                             VALUE LOW-VALUE.
      88 SYSLST                             VALUE "L".
      88 SYSOUTCSV                          VALUE "O".
      88 SYSLSTCSV                          VALUE "C".

```

The version specification at the beginning of the structure FT-NCOPY-LIST serves to identify the COPY element and must not be overwritten.

The NSHOW macro does not change any values in the structure FT-NSHOW-LIST.

The fields are to be written with left-justified characters and filled with right-justified blanks (default for the COBOL-MOVE statement for character strings).

Fields which should not contain any specifications are to be assigned LOW-VALUE.

If a parameter is not entered, the default values are generated as in the SHOW-REMOTE-FILE-ATTRIBUTES command.

All other values which should be set in quotation marks at the command interface do not have these quotation marks in the programming interface.

Passwords with integer values must be entered in binary form.

For the assignment of keywords, e.g. *NOT-SPECIFIED and *SYSLST, use the values which are set in the corresponding 88 step structure.

Description of the data fields

The parameters for FT-NSHOW-LIST have the same names and functions as the corresponding operands for the command SHOW-REMOTE-FILE-ATTRIBUTES. Please refer to the command description in the user manual.

The parameter USER-DEF-ADM-ATTR is used to determine how the entry for USER-DEF-ADMISSION is to be interpreted. With the value GRAPHIC, the entry for USER-DEF-ADMISSION is interpreted as a printable character and converted for transfer to a system which does not use EBCDIC. With the value OCTET, the entry for USER-DEF-ADMISSION is interpreted as binary information and not converted.

Example NSHOW

```

*****
*   EXAMPLE:                                                    *
*                                                                 *
* /SHOW-REMOTE-FILE-ATTRIBUTES -                               *
* /      PARTNER=VAR001,FILE-NAME=REMOFILE,-                   *
* /      TRANS=(USID,ACCOUNT,'PASSWORD'),-                     *
* /      INFORMATION=*ALL-ATTRIBUTES                           *
*                                                                 *
* FROM A COBOL PROGRAM                                         *
*****
  IDENTIFICATION DIVISION.
  PROGRAM-ID. TESTNSHOW.
*
  ENVIRONMENT DIVISION.
*
  CONFIGURATION SECTION.
  SPECIAL-NAMES.
    TERMINAL IS TERM.
*
  DATA DIVISION.
  WORKING-STORAGE SECTION.
  77 MAIN-RCODE-STRING          PIC -ZZZZ9.
  77 SUB-RCODE-STRING          PIC -ZZZZ9.
  COPY FTNSHOW OF FTLIB.
  COPY FTRETC OF FTLIB.
*
  PROCEDURE DIVISION.
  STEUER SECTION.
  ST-01.
*
*SPECIFY-PARAMETERS.
  MOVE LOW-VALUE TO USER-PARAMETERS.
  MOVE "VAR001" TO PARTNER-NAME.
  MOVE "REMOFILE" TO FILE-NAME.
  MOVE "USERID" TO USER-ID.
  MOVE "ACCOUNT" TO ACCOUNT.
  MOVE "'PASSWORD'" TO PASSWORD.
  SET ALL-ATTRIBUTES IN INFORMATION TO TRUE.
*
*NSHOW-CALLING.
  CALL "NSHOW" USING FT-NSHOW-LIST FT-RETURN-INFO.
*

```



```
*RESULT-HANDLING.  
  IF OKAY IN MAIN-RETURN-CODE  
  THEN  
    DISPLAY "NSHOW OKAY" UPON TERM  
  ELSE  
    MOVE MAIN-RETURN-CODE TO MAIN-RCODE-STRING  
    MOVE SUB-RETURN-CODE TO SUB-RCODE-STRING  
    DISPLAY "NSHOW REJECTED" UPON TERM  
    DISPLAY "MAIN-RETURN-CODE: " MAIN-RCODE-STRING  
      " SUB-RETURN-CODE: " SUB-RCODE-STRING  
      UPON TERM.  
ST-99.  
  STOP RUN.
```

3.1.8 NSTATUS - Query file transfer status

The macro CALL "NSTATUS"... can be used to obtain information about file transfer request. The functionality corresponds to that of the command SHOW-FILE-TRANSFER (NSTATUS).

Macro

The function can be called as follows:

```
CALL "NSTATUS" USING FT-NSTATUS-LIST FT-RETURN-INFO.
```

FT-NSTATUS-LIST

The range FT-NSTATUS-LIST describes the parameter list for the NSTATUS macro. FT-NSTATUS-LIST must be defined in the WORKING-STORAGE-SECTION and can be copied to there with the statement

```
COPY FTNSTAT of linkname.
```

Before the macro is executed you must fill the desired fields. If a parameter is not specified or the default value is to be used, this field must be assigned the value LOW-VALUE.

FT-NSTATUS-LIST is defined as follows:

```

01 FT-NSTATUS-LIST.
*
  02 FILLER                                PIC X(4)  VALUE "V1000".
*
  02 USER-PARAMETERS.
*
    05 TRANSFER-ID                          PIC X(10).
      88 SELECT-ALL                          VALUE LOW-VALUE.
*
    05 SELECT-PARAMETER.
*
      10 OWNER-IDENTIFICATION PIC X(8).
        88 OWN                               VALUE LOW-VALUE.
        88 SELECT-ALL                         VALUE " ".
*
      10 INITIATOR                          PIC X(1).
        88 SELECT-ALL                         VALUE LOW-VALUE.
        88 LOCAL                              VALUE "L".
        88 REMOTE                              VALUE "R".
*
      10 PARTNER-NAME                          PIC X(200).
        88 SELECT-ALL                         VALUE LOW-VALUE.
*
      10 FILE-NAME                            PIC X(512).
        88 SELECT-ALL                         VALUE LOW-VALUE.
      10 LIBRARY                              PIC X(56).
        88 SELECT-ALL                         VALUE LOW-VALUE.
      10 ELEMENT                              PIC X(64).
        88 SELECT-ALL                         VALUE LOW-VALUE.
      10 TYP                                  PIC X(8).
        88 SELECT-ALL                         VALUE LOW-VALUE.
      10 VERSION                              PIC X(24).
        88 SELECT-ALL                         VALUE LOW-VALUE.
*
      10 MONJV                                PIC X(56).
        88 NONE                               VALUE LOW-VALUE.
*
      10 JV-PASSWORD                          PIC X(11).
        88 NONE                               VALUE LOW-VALUE.
*
      10 STATE                                PIC X(1).
        88 SELECT-ALL                         VALUE LOW-VALUE.
        88 SUSP                               VALUE "S".
        88 LOCKED                             VALUE "L".
        88 WAIT                               VALUE "W".
        88 ACTIVE                             VALUE "A".

```

```

            88 FIN                                VALUE "F".
            88 HOLD                              VALUE "H".
            88 CANCELLED                         VALUE "C".
*
    05 INFORMATION                              PIC X(1).
            88 STD                              VALUE LOW-VALUE.
            88 SUMMARY                          VALUE "U".
            88 SELECT-ALL                       VALUE HIGH-VALUE.
*
    05 OUTPUT-DESTINATION                      PIC X(1).
            88 SYSOUT                           VALUE LOW-VALUE.
            88 SYSLST                          VALUE HIGH-VALUE.
            88 SYSOUTCSV                       VALUE "0".
            88 SYSLSTCSV                       VALUE "L".

```

If more than one selection criterion is specified in the NSTATUS macro, a request can be overdefined, for instance, if both TRANSFER-ID and MONJV are specified. If the criteria contradict each other in such a case, the NSTATUS macro is not executed.

Description of the data fields

The parameters for FT-NSTATUS-LIST have the same names and functions as the operands for the command SHOW-FILE-TRANSFER (NSTATUS). Please refer to the command description in the user manual.

3.1.9 Return information and error displays

After every CALL macro to openFT for BS2000, return information is provided in the structure FT-RETURN-INFO.

FT-RETURN-INFO

The range FT-RETURN-INFO describes the return information returned by openFT for BS2000. FT-RETURN-INFO must be defined in the WORKING-STORAGE SECTION and can be copied to there with the statement

```
COPY FTRETC OF linkname.
```

FT-RETURN-INFO is defined as follows:

```
01 FT-RETURN-INFO.
   05 FILLER                PIC X(4)   VALUE "V300".
   05 TRANSFER-ID           PIC X(10).
*
   05 FT-RETURN-CODE.
      10 MAIN-RETURN-CODE   PIC S9(5) COMP.
         88 OKAY                VALUE 0.
      10 SUB-RETURN-CODE   PIC S9(5) COMP.
*
      10 DMS-RETCODE        PIC X(8).
      10 LINK-RETCODE REDEFINES DMS-RETCODE PIC X(8).
*
```

The version specification at the beginning of the structure FT-NCOPY-LIST serves to identify the COPY element and must not be overwritten.

Description of the data fields

TRANSFER-ID	<p>The fields contains</p> <ul style="list-style-type: none"> – after a successful NCOPY macro, the transfer ID of the generated request (left-justified, filled with right-justified blanks) or – after an unsuccessful NCOPY macro, the character string '#####'. <p>For a NCANCEL-/NSTATUS macro, the fields remains unchanged.</p> <p>For a NMOD-/NSHOW-/NDEL-/NLMOD- or NLSHOW macro, the field has no meaning.</p>
FT-RETURN-CODE	This field contains the return messages.

Set-up of the FT-RETURN-CODE field

The return code field is set up as follows:

Maincode	Subcode2	Subcode1
Parameter error	warnings	error class
1 byte	1 byte	1 byte

The field with the return code meets the conventions for return codes as of BS2000 Version 9.

3.1.9.1 Return codes for the CALL macros NCOPY, NCANCEL and NSTAT

The field MAIN-RETURN-CODE contains the error class and can have the following values:

- 0 No error.
- 1 Error; however, the program can successfully repeat the same call later without further intervention.
- 2 Error; however, the program can repeat the call unchanged after an intervention by a terminal user or operator.
- 3 Error which does not belong to class 1 or 2.

MAIN-RETURN-CODE	SUB-RETURN-CODE	Meaning
0	0	The CALL macro was saved in the request file of the local system.
0	1	Follow-up processing in the local system was not executed, because the entries in the PROCESSING-ADMISSION are erroneous.
0	2	Follow-up processing was not executed for the system whose PROCESSING-ADMISSION is missing.
0	3	The FT request was only started with normal priority.
1	1	The CALL macro cannot be accepted/executed, since the send or receive file is protected via another process from simultaneous updating.
1	2	The CALL macro cannot be accepted, since the maximum permissible limit for file transfer requests has been exceeded.
2	1	The CALL macro cannot be accepted/executed, since the specifications in one of the TRANSFER-ADMISSIONs are erroneous.
2	2	The CALL macro cannot be accepted/executed, since the owner of the send or receive file is not defined in the corresponding system, or since the file owner and the user who wishes to create a receive file do not correspond.
2	3	The CALL macro cannot be accepted, since the remote system is not in the partner list of the local system.
2	4	The CALL macro cannot be accepted/executed, since the password for the send or receive file is missing or wrong.
2	5	The CALL macro cannot be accepted/executed, since the send or receive file does not allow multiple users.
2	6	The CALL macro cannot be accepted/executed, since the send or receive file only permits certain access operations (e.g. read only).
2	7	The CALL macro cannot be accepted/executed, since the protection time for the overwriting of the receive file has not expired.

MAIN-RETURN-CODE	SUB-RETURN-CODE	Meaning
2	8	The CALL macro cannot be (further) executed, since the space which the user (as defined in the TRANSFER-ADMISSION) is permitted to use for saving in the receiving system is full, either because the send file contains too many consecutive blanks or because the primary assignment of the password-protected receive file is too small. The receive file cannot be created or expanded.
2	9	The CALL macro was not executed because <ul style="list-style-type: none"> – the system was generated without job variables or – the user does not have access to the job variable specified or – the job variable specified is monitoring another FT request or – the password specified is not the right one for the job variable.
2	10	The CALL macro was not executed because the job variable in question is not present.
2	11	The CALL macro was not executed because the job variable specified does not monitor an FT request.
2	12	The CALL macro was not executed because the contents of the job variable specified are not consistent.
2	13	The CALL macro was not executed due to an error in the local PROCESSING-ADMISSION.
2	14	The macro was not executed because the file encryption function is not installed.
2	15	The macro was not executed because the file expansion for transparent transfer is not permitted.
2	20	The CALL macro was not executed because the request has not yet been cancelled with FORCE-CANCELLATION=NO.
2	129	The macro cannot be accepted because the local FT system is currently unavailable.
3	1	The macro cannot be accepted because at least one operand is missing.
3	2	The macro cannot be accepted because it contains a syntax error unlike “operand missing” or “keyword unknown”. Such syntax errors include: value assignments outside the permitted value range, wrong operand separators, wrong characters assigned to the value or partially qualified filenames.
3	3	The macro cannot be accepted/executed because the FT system only transfers individual data generations.
3	4	The CALL macro cannot be accepted because the send file is not in the catalog or on a data carrier of the local system. The CALL macro cannot be executed because the send or receive file is no (longer) in the catalog or on a data carrier of the corresponding system.

MAIN-RETURN-CODE	SUB-RETURN-CODE	Meaning
3	5	The CALL macro cannot be accepted/executed because the send file is empty.
3	6	The CALL macro cannot be accepted because no requests could be found.
3	7	The CALL macro cannot be accepted/executed because an existing receive file cannot be recreated.
3	8	The CALL macro cannot be accepted/executed because the file owner and the user requesting the creation of a receive file do not correspond.
3	9	The CALL macro cannot be accepted/executed because the data carrier for the send or receive file is either not mounted, unknown or unnamed or multiple data carriers are not supported.
3	10	The CALL macro cannot be accepted because it can only be input by authorized users.
3	11	The CALL macro cannot be accepted/(further) executed because there is a file structure error. File structure errors include: the attributes of the send file are incomplete. The data of the send file do not correspond to the structure attributes. The send file sentences are too long. The send file and the receive file have a different structure for WRITE-MODE=EXTEND-FILE (e.g.: sentence fixed/variable length).
3	12	During the processing of a CALL macro, a DVS error occurred. The DVS error code is in the DMS-RETCODE field in hexadecimal notation.
3	13	During the processing of a CALL macro, an NDMS error, FJAM error or operating system error occurred which was neither a DVS error nor a transport system error.
3	17	The CALL macro cannot be accepted because the start time is more than 32767 minutes in the future, or because the abort time is either in the past, before the start time or more than 32767 minutes in the future.
3	18	An error occurred while loading the FT runtime module. The return code returned by the LINK macro is in the field LINK-RETCODE in hexadecimal notation.
3	19	The CALL macro was not executed because the local and remote systems are not compatible. Neither system can connect with the other, or at least one of the systems does not support the function called.
3	20	The CALL macro was not executed, because the filename was neither explicitly entered nor indicated by means of the TRANSFER-ADMISSION used.
-1	1	The function NSTAT is not supported by the FT version installed (version is smaller than V3.0).
-1	2	FT runtime module not available.

MAIN-RETURN-CODE	SUB-RETURN-CODE	Meaning
-1	3	The version of the one of the COPY elements used is not supported.
-1	129	The CALL macro cannot be accepted because there is not yet an output for asynchronous end messages.
-1	130	The CALL macro cannot be accepted because the FT system has not yet completed a previous shutdown command (e.g. FTSTOP).

3.1.9.2 Return code for the file management CALL macros

The following descriptions are only valid for the file management CALL macros (NDEL, NLMOD, NLSHOW, NMOD and NSHOW).

Maincode

This field contains the exact error. It is divided into two bytes. The right byte shows the cause of the error. It can have the following values (in decimal notation):

- 0: the call was error-free
- 1: the parameter field is partially or entirely in an invalid address space
- 2: the address or length field contradict the keyword field
- 3: the parameter entry contradicts another parameter
- 4: the parameter value is too long or violates syntax rules;
e.g. FILENAME, PASSWORD
- 10: the buffer is too small; this error code is reserved for future expansions
- 11: the arithmetic parameter value (e.g. RECSIZE value) is invalid
- 12: the keyword entered is not permitted for this parameter
- 13: a mandatory parameter is missing

The left byte is only set to a value other than 0 if there is a parameter error. It specifies the parameter in which the error occurred. The meaning of these values can be found in the description of the Assembler macro YNDEQU on [page 60](#).

If the left byte is set to 0, the right byte contains the message number indicating the result of the macro.



For compatibility reasons, the message numbers valid up to and including openFT V9 are still output at the Cobol interface. You find an conversion table between old and new return codes in section „Maincode conversion table“ in the appendix of the openFT V11.0 User Guide.

Subcode1

The error codes are divided into error classes. These are identified in subcode 1. The following error classes exist:

Class	Value	Meaning
A	0	the call was successfully completed
B	1	permanent error, no repetition possible, there was a syntax error or equivalent parameter error
C	32	system error an internal error occurred during macro processing
D	64	repeat after correction of user input
E	128	wait and repeat

Subcode2

This field contains a warning, if the macro was completed successfully (subcode1 = 0 and maincode = 0) but the macro couldn't be executed - for instance, if no files could be found during the NSHOW call.

Exception

If the maincode contains the value X'FFFF', the request could not be executed for reasons specified uniquely throughout the system.

Related publications

The manuals are available as online manuals, see <http://manuals.ts.fujitsu.com>.

openFT for BS2000/OSD
Managed File Transfer in the Open World
User Guide

openFT for BS2000/OSD
Installation and Administration
System Administrator Guide

BS2000/OSD-BC
Executive Macros
User Guide

Index

A

- admission profile 148
 - create 91
 - delete 100
 - export 102
 - import 104
 - list 121
 - modify 109
- admission set
 - export 102
 - import 104
 - list 121
 - output 119
- ASSEMBLER 16, 39, 43, 46, 49, 53
- ASSEMBLER interface
 - openFT-AC 90
- attribute
 - display 45
 - display (remote file) 52
 - modify (remote file) 47

C

- CALL
 - "NCANCEL" 138
 - "NCOPY" 142
 - "NDEL" 151
 - "NLMOD" 156
 - "NLSHOW" 162
 - "NMOD" 167
 - "NSHOW" 173
 - "NSTATUS" 178
- CALL macro 137
- cancel file transfer 15
- CANCEL-FILE-TRANSFER 15
- COBOL macro 137

CODED-CHARACTER-SET

- local 25, 35
- remote 26, 36
- compatibility 14
- constants
 - set (YNDEQU) 60
- create
 - admission profile 91
- create admission profile 91
- CREATE-FT-PROFILE 98
 - CREFTPRF 91
- CREFTPRF 91, 98
 - DSECT format 95

D

- default format 12
- default header 12
- delete
 - admission profile 100
 - FT request 15, 138
 - remote file 151
- delete remote file NDEL 38
- DELETE-FT-PROFILE 101
 - DELFTPRF 100
- DELETE-REMOTE-FILE 38
- DELFTPRF 100, 101
 - DSECT format 101
 - LIST format 100
- display
 - attributes of a local file 45, 162
 - file attribute (remote file) 52
 - local file attribute (FTAM) 45
 - remote file attributes 173
- DSECT format 11
 - CREFTPRF 95

- DELFTPRF 101
 - EXPFTENV 103
 - IMPFTENV 105
 - MODFTADS 107
 - MODFTPRF 110, 114
 - NCAN 16
 - NCOPY 21
 - NCOPYSYN 31
 - NDEL 39
 - NLMOD 42
 - NLSHOW 46
 - NMOD 49
 - NSHOW 53
 - NSTAT 57
 - SHWFTADS 119
 - SHWFTENV 122
 - SHWFTPRF 124
- E**
- error class 183
 - error code 187
 - error display 65, 181, 182
 - example
 - NCANCEL 140
 - NCOPY 149
 - NDEL 154
 - NLMOD 160
 - NLSHOW 165
 - NMOD 171
 - NSHOW 176
 - EXECUTE format 11
 - EXPFTENV 102, 103
 - DSECT format 103
 - LIST format 102
 - export
 - admission profile 102
 - admission set 102
 - EXPORT-FTAC-ENVIRONMENT 103
 - EXPFTENV 102
- F**
- field length 12
 - file attribute
 - display (remote file) 52
 - display (remote) 52
 - display local (FTAM) 45
 - modify (remote file) 47
 - modify local (FTAM) 41
 - file transfer cancel 15
 - file transfer request
 - cancel 15, 138
 - information 56
 - file transfer status
 - query 56, 178
 - FT request delete 15
 - FTAC
 - macros 90
 - programming interface 90
 - FT-AC function
 - Cobol interface 148
 - FTAM attribute
 - modification of in a local file 41, 156
 - modify local 41
 - FTAM partner 27, 37
 - FT-NCANCEL-LIST 138
 - FT-NCOPY-LIST 142
 - FT-NDEL-LIST 151
 - FT-NLMOD-LIST 156
 - FT-NLSHOW-LIST 162
 - FT-NMOD-LIST 167
 - FT-NSHOW-LIST 173
 - FT-NSTATUS-LIST 178
 - FT-RETURN-CODE 181
 - FT-RETURN-INFO 181
 - FTSNCOPY 29
- I**
- IMPFTENV 104, 105
 - DSECT format 105
 - LIST format 104
 - import
 - admission profile 104
 - admission set 104
 - IMPORT-FTAC-ENVIRONMENT 105
 - IMPFTENV 104
 - information
 - on file transfer requests 56
 - inverted commas 170

L

layout of the operand lists 11

link name 148, 159, 164

list

 accessible partner systems 125

 admission profile 121

 admission set 121

LIST format 12

DELFTPRF 100

EXPFTENV 102

IMPFTENV 104

MODFTADS 106

NCAN 15

NCOPY 18

NCOPYSYN 29

NDEL 38

NLMOD 41

NLSHOW 45

NMOD 47

NSHOW 52

NSTAT 56

SHWFTADS 119

SHWFTENV 121

SHWFTPRF 123

local FT file attributes

 display 45, 162

M

macro

 NCAN 15

 NCOPY 18, 29

 NDEL 38

 NLMOD 41

 NLSHOW 45

 NMOD 47

 NSHOW 52

 NSTAT 56

 YNDEQU 60

macros 90

maincode 66, 130, 187

MODFTADS 106

 DSECT format 107

 LIST format 106

MODFTPRF 109

 DSECT format 110, 114

modify

 admission profile 109

 file attribute (remote file) 47

 FTAM attributes of a local file 41, 156

 local file attributes (FTAM) 41

 remote file attributes 47, 167

MODIFY-FILE-FT-ATTRIBUTES

 NLMOD 41

MODIFY-FT-ADMISSION-SET 108

 MODFTADS 106

MODIFY-FT-PROFILE 117

 MODFTPRF 109

MODIFY-REMOTE-FILE- ATTRIBUTES

 NMOD 47

N

NCAN

 DSECT format 16

 LIST format 15

 macro 15

 return codes 88

NCANCEL 15

 CALL macro 138

 example 140

NCOPY 18

 CALL macro 142

 DSECT format 21

 example 149

 LIST format 18

 macro 18, 29

 return codes 68

NCOPYSYN

 DSECT format 31

 LIST format 29

 return codes 81

NDEL

 CALL macro 151

 delete remote file 38

 DSECT format 39

 example 154

 LIST format 38

 macro 38

 return codes 68

NLMOD

- CALL macro [156](#)
- DSECT format [42](#)
- example [160](#)
- LIST format [41](#)
- macro [41](#)
- return codes [68](#)

NLSHOW

- CALL macro [162](#)
- DSECT format [46](#)
- example [165](#)
- LIST format [45](#)
- macro [45](#)
- return codes [68](#)

NMOD

- CALL macro [167](#)
- DSECT format [49](#)
- example [171](#)
- LIST format [47](#)
- macro [47](#)
- return codes [68](#)

NSHOW

- CALL macro [173](#)
- DSECT format [53](#)
- example [176](#)
- LIST format [52](#)
- macro [52](#)
- return codes [68](#)

NSTAT

- DSECT format [57](#)
- LIST format [56](#)
- macro [56](#)
- return codes [89](#)

NSTATUS [56](#)

- CALL macro [178](#)

O

- openFT-AC [148](#)
 - ASSEMBLER interface [90](#)
- operand list [11](#), [12](#)
- output admission set [119](#)

P

- programming interface [11](#)

Q

- quotation marks [147](#), [152](#), [158](#), [175](#)

R

- Readme file [9](#)
- remote file attributes
 - display [173](#)
 - modify [47](#), [167](#)
- remote file delete [151](#)
- return code [130](#), [182](#), [187](#)
 - NCAN [88](#)
 - NCOPY [68](#)
 - NCOPYSYN [81](#)
 - NDEL [68](#)
 - new variant [65](#)
 - NLMOD [68](#)
 - NLSHOW [68](#)
 - NMOD [68](#)
 - NSHOW [68](#)
 - NSTAT [89](#)
 - old variant [65](#)
- return information [65](#), [181](#)

S

- set constants
 - for macros [60](#)
 - YNDEQU [60](#)
- SHOW-FILE-FT-ATTRIBUTES
 - NLSHOW [45](#)
- SHOW-FILE-TRANSFER
 - NSTAT [56](#)
- SHOW-FTAC-ENVIRONMENT
 - SHWFTENV [121](#)
- SHOW-FT-ADMISSION-SET [120](#)
 - SHWFTADS [119](#)
- SHOW-FT-PROFILE [124](#)
 - SHWFTPRF [123](#)
- SHOW-FT-RANGE [126](#)
 - SHWFTRGE [125](#)
- SHOW-REMOTE-FILE-ATTRIBUTES
 - NSHOW [52](#)
- SHWFTADS [119](#), [120](#)
 - DSECT format [119](#)
 - LIST format [119](#)

SHWFTENV 121, 122
 DSECT format 122
 LIST format 121
SHWFTPRF 123, 124
 DSECT format 124
 LIST format 123
SHWFTRGE 125, 126
subcode 182
symbolic address 12

T

transfer
 file 18
 library member 18
transfer admission 13, 148
transfer file
 NCOPY 18
transfer file synchronous
 NCOPYSYN 29
transfer library member
 NCOPY 18
TRANSFER-ADMISSION 148
TRANSFER-FILE 18
TRANSFER-FILE-SYNCHRONOUS 29
TRANSFER-ID 27, 36

V

VERSION
 parameter 65

Y

YFSEQU 127, 130
YNDEQU 67
 macro 60

