IBM® Sterling Connect:Express® for UNIX

User and Installation Guide

Version 1.5

Copyright

This edition applies to the Version 1.5 of IBM® Sterling Connect:Express® for UNIX and to all subsequent releases and modifications until otherwise indicated in new editions.

Before using this information and the product it supports, read the information in Notices on page N-1.

Contents

Chapter 1 Introduction

Overview 1-1
General Structure of Sterling Connect:Express 1-2
Tom_mon Process 1-2
Tom_apm Process 1-3
Tom_ftp Process 1-4
Tom_api Process 1-4
Tom_opr Process 1-4
L0b8z20 Application Interface 1-4
Sterling Connect:Express Parameters 1-5
Key Terms 1-5

Sterling Connect:Express Directories 1-6
TOM Directory 1-7
Config Directory 1-7
Exit Directory 1-9
GRTF Directory 1-9
ITOM Directory 1-10
sterm Directory 1-11
strf Directory 1-11
Notif Directory 1-11
Httpn Directory 1-12

Sterling Connect:Express Files 1-12
Config Directory 1-12
Using the ch_conf Command 1-16
GTRF and STRF Directories 1-17
Managing RENC Files 1-17

Sterling Connect:Express Environment Variables 1-18

Sterling Connect:Express Transfers 1-20

Chapter 2 Installation

Installation Requirements 2-1
Hardware Requirements 2-1
Software Requirements 2-1
Sterling Connect:Express Installation Materials 2-1

Installing Sterling Connect:Express 2-2
Step 2 Download Archives 2-2
Step 3 Execute the Installation Procedure 2-2
Step 4 Run Customized Procedure 2-3
Step 5 Update the SYSIN File 2-3
Specifying TCP/IP Stack for TCPORT and FTPORT 2-7
Specifying TCP/IP Stack for APPORT 2-7
Step 6 Kernel Parameters 2-7
Step 8 Activate the Monitor 2-7

Implementing Sterling Connect:Express for UNIX 2-8

The sterm Module 2-9
Activating sterm 2-9
sterm Structure 2-10

Chapter 3 Directories and Tables

Accessing the Main Menu 3-1

The Partners Directory 3-3
Managing the Partners Directory 3-3

The Files Directory 3-7
Managing the Files Directory 3-7
Using the Record Length and Record Format Fields 3-12

Monitor Management 3-13
Monitor Status 3-13
Viewing Transfer Requests 3-14
Interrupting a Transfer Request 3-17
Restarting a Transfer Request 3-17
Purging a Transfer Request 3-18
Displaying the Status of a Transfer Request 3-18
Interrogation of Log 3-19
Request Deletion 3-20
Deleting a Group of Transfer Requests 3-20

Tables Management 3-21
Updating Session Tables 3-22
Updating Presentation Tables 3-23

Chapter 4 Transfers

Overview 4-1
The Transfer Process 4-2

Types of Transfer Requests 4-2
Operator Transfer Requests 4-3
Batch Transfer Requests 4-5
The End to End utility 4-7

Acknowledging a Transfer 4-7
Forwarding a Transfer 4-8
P1b8pe2e Reference 4-8

Chapter 5 Application Interface

Overview 5-1

Batch Utilities 5-1
Interrupting a Transfer 5-2
Purging a Transfer 5-2
Restarting a Transfer 5-3
Display Requests from the RENC File 5-3
Display Partners from the RPAR File 5-3
Displaying Files from the RFIC File 5-3
Deleting Partners from the RPAR File 5-3
Deleting Files from the RFIC File 5-3
Error Messages and Results 5-4

Application Program Using L0B8Z20 5-6
Starting a Transfer Request 5-6
Interrupting a Transfer Request 5-8
Purging a Transfer Request 5-9
Restarting a Transfer Request 5-12
Displaying a Record from the RENC File 5-13
Display a Record from the RPAR File 5-14
Displaying a Record from the RFIC File 5-15
Deleting Partners from the RPAR File 5-16
Deleting Files from the RFIC File 5-18
Creating Partner Records in the RPAR File 5-19
Creating a File Record in the RFIC File 5-20
Updating the RPAR File 5-21
Updating the RFIC File 5-22
Starting a Transfer Acknowledgment 5-23
Forwarding a Transfer 5-25
Acknowledging a Transfer Request 5-27
Forwarding a Transfer Request 5-28

Appendix A Sterling Connect:Express Messages

Syntax of Error Messages A-1
SYSLOG Option A-2

Logging Messages A-2

LOG File Rotation A-13

Changing the Size of the LOG File A-13

Appendix B Return Codes

Protocol Return Codes B-1

Sterling Connect:Express Return Codes B-3

Sterling Connect:Express PeSIT SSL Return Codes B-8

Appendix C User Commands and Exits

Overview C-1
User Commands C-2
Standard Error Command (UEXERR) C-3
Standard Signal Error Command (UEXSIG) C-3
Exits C-3

Appendix D Implementing Special Features

Implementing SYSTCP D-1

Implementing a Partner List D-2

Implementing Physical Names D-2
Remote Partner of Type Other D-2
Remote Partner of Type Sterling Connect:Express D-3
Transmission to a Remote Sterling Connect:Express D-3
Reception of a file from a remote Sterling Connect:Express D-3

Translating Data D-4

Implementing a Parameter Cards File D-5

Appendix E Implementing the PeSIT Message Functionality

PeSIT Message E-1
Overview E-1
Sending and Receiving Messages E-2
Store and Forward – End to End Acknowledgment E-5
Overview E-5
End to End Acknowledgment - EERP E-5
Forwarding and Acknowledging Files E-10

Notices 1

Trademarks E-3

Index

Chapter 1

Introduction

This chapter provides an overview of the Sterling Connect:Express for UNIX product and includes information about Sterling Connect:Express files and directories.

Overview

Sterling Connect:Express is a family of software products used for data distribution and management. Sterling Connect:Express distributes, tracks, monitors, and manages information between multiple mainframes, minicomputers, and personal computers using dynamic allocation of files, multi-session control, priority and class management, and store and forward capacity.

Sterling Connect:Express executes on IBM z/OS, UNIX, and Microsoft Windows platforms. It supports IBM SAM, text and binary Microsoft Windows or UNIX files The product also supports different types of networks, including SNA and TCP/IP. In addition, Sterling Connect:Express interfaces with major security packages, such as RACF, ACF2, TOP SECRET.

The network guarantees data delivery to the transport layer of the International Standards Organization (ISO) for Open Systems Interconnection, as shown below. Sterling Connect:Express guarantees data delivery all the way to the application level (disk or tape).

Application	**<----------> **	**Application**
Presentation	<---------->	Presentation
Session	<---------->	Session
Transport	<---------->	Transport
Network	<---------->	Network
Data Link	<---------->	Data Link
Physical	<---------->	Physical

Physical Media (Ethernet, Token Ring)

Sterling Connect:Express for UNIX works with the following operating systems:

- AIX (from 6.1)

- SUN/OS and Sun/Solaris (from 5.10)
- Linux (any 64 bit version with GLIC >= 2.7)
- Linux zSeries (Red Hvat 2.6.9-67)

You can use the French public protocol PeSIT, as well as FTP with Sterling Connect:Express with the TCP protocol.

Note: ETEBAC3 and X25 are not supported anymore.

General Structure of Sterling Connect:Express

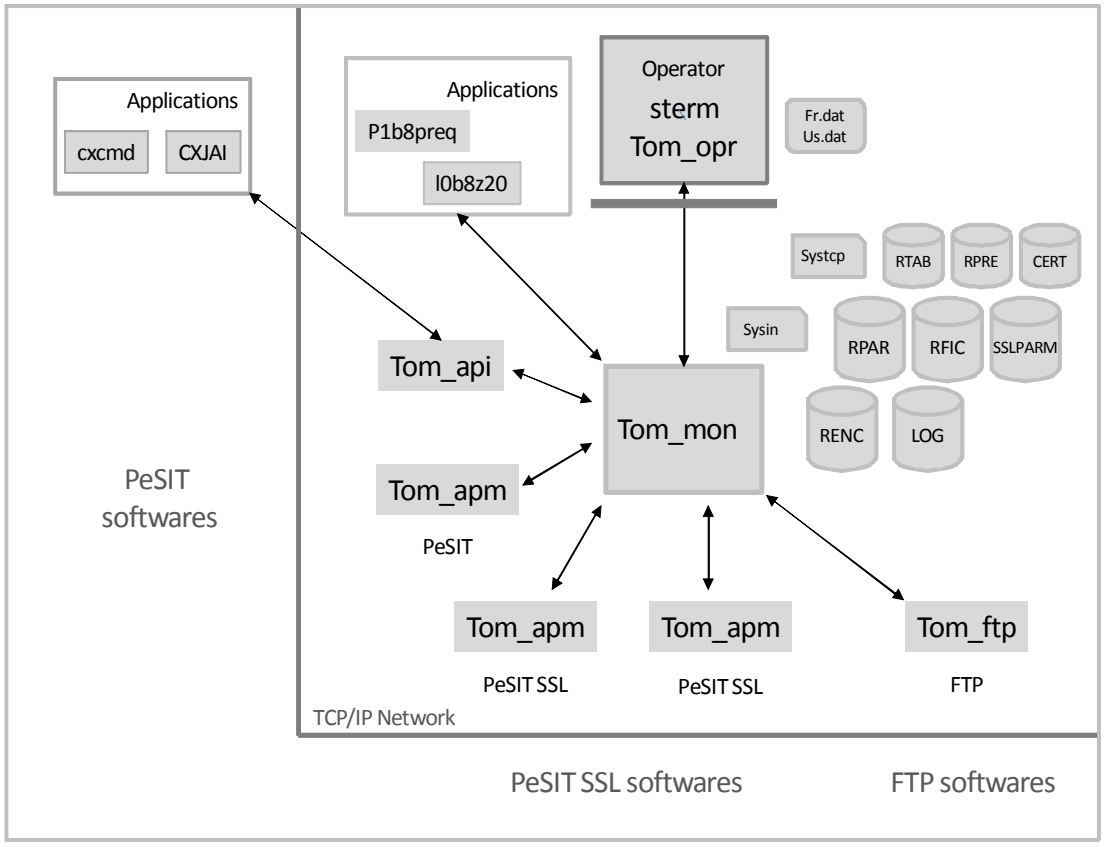
The following figure illustrates the general structure of Sterling Connect:Express for UNIX..

Tom_mon Process

The *tom_mon* process is launched using the *$start_tom* command. This is the monitor process. Tom_mon starts from the *SYSIN* file, where the Sterling Connect:Express configuration is defined. The SYSIN file determines the protocol processes to launch: *tom_apm*, *tom_ftp* and *tom_api*.

Note: The SYSIN file refers to the physical file: *$TOM_DIR/config/sysin*.

Operations are configured in the *RPAR*, *RFIC*, *SSLPARM*, *CERT*, *RTAB* and *RPRE* files. Activity is monitored through the *RENC* and *LOG* files.



The RPAR directory describes the Partners, the RFIC directory describes the Files, the SSLPARM directory describes tls/ssl profiles, the CERT directory manages ssl certificates, the RTAB file defines network session profiles, the RPRE file defines data presentation profiles.

The RENC directory contains the transfer requests submitted locally or from a remote partner. Requests can be executing, ended, interrupted or waiting selection. The LOG file displays the reporting messages issued by tom_mon.

Tom_mon communicates with the processes around through UNIX System V message queues.

Tom_apm Process

The tom_apm process manages the PeSIT protocol, over TCP/IP network. Tom_apm can be a server to accept inbound transmission or reception requests from remote PeSIT partners, or a requester to serve local outbound transmission or reception transfer requests with a PeSIT partner from an application or an operator. During initialization tom_mon launches as many tom_apm servers as defined by the configuration. For each local PeSIT transfer request tom_mon launches a tom_apm requester to serve the request.

The number of tom_apm servers is determined by the SSLPARM directory: one tom_apm manages unsecured PeSIT inbound sessions. The number of PeSIT/SSL tom_apm servers depends on the number of SSL server profiles defines in SSLPARM: an SSL server profile is linked to a SSL certificate, and determines an SSL

handshake policy. When an inbound request is received, the tom_apm server generates a tom_apm child to perform the file transfer.

The total number of tom_apm can be controlled at different levels:

- MAXSES= parameter in the SYSIN file determines the maximum number of simultaneous inbound and outbound file transfers. For example MAXSES=5 means that no more than 5 file transfers can be executed at the same time, in both directions. Should a new inbound request be received, it will be rejected with a return code that will invite to retry later. A new local outbound request will wait until the end of one of the current transfers.
- STRFD= parameter in the SYSIN file determines the maximum number of simultaneous outbound file transfers, thus limiting the number of tom_apm requesters. For example STRFD=2 means that no more than 2 tom_apm requesters can be executed at the same time. A new local outbound request will wait until the end of one of the current outbound transfers.
- RPAR partner definition: for each partner you configure in the RPAR file, you can define the maximum number of inbound, outbound, and total simultaneous file transfers. For example: 3 max inbound + 3 max outbound + 4 max total .

Tom_ftp Process

The tom_ftp process manages the FTP protocol,, over TCP/IP network. Tom_ftp can be a server to accept inbound transmission or reception requests from remote FTP partners, or a requester to serve local outbound transmission or reception transfer requests with a FTP partner from an application or an operator. During initialization tom_mon launches one tom_ftp server if defined by the configuration. SFTP is not supported. For each local FTP transfer request tom_mon launches a tom_ftp requester to serve the request.

When an inbound request is received, the tom_ftp server generates a tom_ftp child to perform the file transfer.

The session control mechanisms described for tom_apm apply to tom_ftp.

Tom_api Process

The tom_api process is a server that manages the Sterling Connect:Express api protocol, over TCP/IP network. Tom_api server accepts inbound requests from remote Sterling Connect:Express Application Interface for Java (CXJAI). The Application Interface for Java of Sterling Connect:Express can be implemented through the *cxcmd* utlity or any java application. One example of java application is *IBM Sterling Control Center*.

Tom_opr Process

The tom_opr process is the native operator interface. Use *$sterm* to launch this interface. Sterm interface enables you to configure Partners, Files, SSL profiles and certifcates, Network session and Data presentation profiles.

$sterm_c and *$sterm_v* commands limit access to sterm options. Include one of the sterm environment variables in the unix user profiles when you want to limit acces to a user.

The sterm interface can be installed in French or in English.

L0b8z20 Application Interface

The application interface is called *l0b8z20*: any C application can link to this interface and submit file transfer requests, or monitor file transfer activity.

One example is the utility *p1b8preq:* use this program in shell scripts to submit file transfer requests.

Sterling Connect:Express Parameters

Sterling Connect:Express for UNIX enables you to interface and communicate with other computers, but you must define parameters to do this. The table below describes the two types of parameters used for system implementation:

Type	Description
Local parameters	Local parameters are entered in the SYSIN file. These parameters define the conditions under which Sterling Connect:Express works.
External parameters	External parameters are entered through operator interface menus. They identify remote Partners and Files to be transferred.

Key Terms

The following table defines key terms used in this manual:

Term	Definition
File	A file is a symbolic name recognized by Sterling Connect:Express that contains transfer characteristics such as direction, partner, physical name, and record length. It is defined in the files directory with the Sterling Connect:Express menus or application interface and can be modified dynamically.
Partner	A Partner is a symbolic name used to define a remote system and its characteristics such as, type of connection, session parameters, and transfer protocol. It is defined in the partners directory with the Sterling Connect:Express menus or application interface and can be modified dynamically. The local computer must be defined in the SYSIN file.
Request	A request is a transfer request sent to Sterling Connect:Express. It is sent to the monitor by a batch program, initiated from the menus, or initiated by a remote Partner. A request number is written in this manual as QQQNNNNN, where QQQ is Julian date (modulo 183) and NNNNNN is a sequential number in hexadecimal.
Requester	A requester is a program which initiates a network session to transmit or receive a file from the server through the network.
Server	A server is a program which is called by the requester to transmit or receive a file through the network.
Transmitter	A transmitter is a program which transmits a file through the network to the receiver.
Receiver	A receiver is a program which receives a file through the network from the transmitter.

Sterling Connect:Express Directories

When you install Sterling Connect:Express for UNIX, the installation process creates directories and files on your system. The following table describes the Sterling Connect:Express main directories.

Directory	Description
tom1	The main directory that contains the .profile procedure. This directory has the name that you provided during installation and cannot exceeds 4 characters long. For example: tom1
config	The config directory contains processing environment definitions.
etc	This directory containes extra commands like the Connect:Express for Unix Data Collector.
exit	All exit examples, user exits, and shell procedure commands are in this directory.
gtrf	This is the Sterling Connect:Express monitor directory. This directory contains executable and trace files for the monitor.
httpn	This directory contains the configuration, executable and trace files of the HTTP Sterling Connect:Express notifications facility. Refer to the *Sterling Connect:Express Adapter for Sterling B2B Integrator* user guide, in *HTTP Notification in Sterling Connect:Express for UNIX* , for installing and configuring this option
ilmt	This directory contains the IBM Licence Metric Tool.
in	This is the default directory for files to be transmitted or received during an incoming session (Server Mode).
itom	This is the Sterling Connect:Express application interface directory. It contains all application utilities like p1b8preq, p1b8ppur and the l0b8z20 interface.
notif	Contains the configuration and command files used by the Sterling Connect:Express notifications facility. Refer to the *Mise en oeuvre de l'Utilitaire de Notification*, for installing and configuring this option
ntfo	Contains the HTTP notifications files.
lib	This directory contains the 3rd party software shared libraries like OpenSSL.
out	This is the default directory for files to be transmitted or received during an outgoing session (Requester Mode).
proc	This directory contains the files used to manage the Connect:Express processes message queues.
rci	This direcory contains the files saved by the Sterling Connect:Express notifications recovery.
scripts	This directory contains sample configuration files and relative sample transfer scripts.
stats	This directory contains statistic files generated when this option is enabled (ISSTAT=1) in the SYSIN file of Sterling Connect:Express.
sterm	This is the directory for the Sterling Connect:Express screen interface.
strf	This directory contains the executable and trace files of the PeSIT and FTP protocols managers.
syn	This directory contains all checkpoint files (.syn).
tmp	This directory is a temporary directory used internally by Sterling Connect:Express.

Work files of processes, such as report files, should be purged periodically. The table below identifies these work files and the directory where they are located.

Directory	Work Files
Exit directory	TOM1_C_* output files for commands or UEXERR shell (2) TOM1_E_* output files for exits (2)
syn directory	*.syn synchronization files (2)
gtrf directory	tom_out.pid output files for monitor process, S*Date_Time_Seq*, and C*Date_Time_Seq* files for the remote application interface tom_api (3).
strf directory	RTqqqnnnnn.pid output files for TCP/IP tom_apm requester process (3) ST.pid output files for TCP/IP tom_apm server process (3) RFqqqnnnnn.pid output files for FTP tom_ftp requester process (3) SF.pid output files for FTP tom_ftp server process (3)
notif directory	Work files used by the notification facility (2)
ntfo directory	Work files used by the HTTP notification facility (2)

Note: Files created while the Sterling Connect:Express monitor is running are referenced with a (2). Files created when the STRACE option is set, are referenced with a (3).

TOM Directory

The TOM directory is the main directory and contains *.profile* which includes the logging procedure to export and define variables that help you use Sterling Connect:Express.

Config Directory

The Config directory includes processing environment definitions, such as the Files and Partners directories. The table below describes the files in the Config directory. Files created when you first start the Sterling Connect:Express monitor are referenced with a (1).

Type	File or Procedure	Description
Command procedures	compact_base reinit_base.sh purge_queue.sh ch_conf.sh start_tom.sh stop_tom.sh p1b8preq.sh rebuild_base.sh check_apm.sh	Procedure to compress/rebuild data base (for compatibility). Procedure to launch the rebuild base program. Procedure to purge remaining IPC queues. Procedure to update configuration parameters. Procedure to start the monitor and test return codes. Procedure to stop the monitor and test return codes. Example of procedure to request a transfer and test return codes. Procedure to reorganize the database and test return codes. Procedure to check the apm status
Data Files	database.p	Definition of database.
	fr.dat	Messages in French.
	LOG	LOG file (1)

Type	File or Procedure	Description
	RENC.*	Requests directory (1)
	RFIC.*	Files directory (1)
	RPAR.*	Partners directory (1)
	CERT.*	SSL certificates
	SSLPARM.*	SSL configuration parameters
	RDN.*	Certificate control parameters
	RPRE.dat	Presentation tables (1)
	RTAB.dat	Session tables (1)
	SYSIN	Launch parameters.
	SYSTCP	Rescue TCP/IP addressing.
	TBL0x.dat	Translation tables. See *Sterling Connect:Express Files* on page 1-7.
	TBL0x.ORG	Translation tables. See *Sterling Connect:Express Files* on page 1-7.
	us.dat	Messages in English.
	parmlftp	File describes specific FTP list records.
	apmftpe	Example of extended FTP identification definition.
Executable programs	build_tbl	Program to construct translation tables.
	stop_tom	Program to stop the Sterling Connect:Express product.
	Info1	Program to display the name and IP addresses of the host.
	rebuild_base	Program to rebuild the base. This program enables you to launch the reorganization of the base manually. To use this option, you can change the parameter RBUILD=1 in the SYSIN file. The following values are valid for this parameter: 0 - No automatic reorganization of the base. 1 - Automatic reorganization of the base when the monitor is launched or stopped. This key word is optional, and the default value is RBUILD=1.
	start_trace	Program to trace beginning and end of transfers events in the LOG file of Sterling Connect:Express.
	stop_trace	Program to stop the start_trace program.
	list_ctx	Program to display active contexts.
	tom_prm	Program to manage the Sterling Connect:Express database. refer to the *Connect:Express UNIX Integration Tools Guide.*
	ch_conf	Program to change configuration parameters.
	Info2	Program to display the key number that the system used to create the queue of the monitor.

Type	File or Procedure	Description
	sslerr	Program to display SSL error labels.
Directory	BASE	Configuration files initialized by the installation procedure, or saved by a user procedure.
	ssl	Files and utilities used by SSL.

Exit Directory

The exit directory contains examples of user exits and shell procedures. The following table describes the files in the Exit directory.

Type	File or Procedure	Description
Command procedures	TRFOK	Sample of command shell procedure.
	UEXERR	Sample of Generalized Error procedure.
	UEXSIG	Sample of Signal Error procedure.
	ROUTAGE UEXROUT	Samples of routing command shell procedures.
	UEXFWRD UEXEERP	Samples of store ans forward and acknowledgment command shell procedures.
	ROUTPI62	Sample of routing command shell procedure based on PI62.
Definition structure (.h)	d1b8ruex.h	Definition of the Parm file given to the user and Etebac3 exit. See *Appendix C, User Commands and Exits* for more information.
Source program	user.c	Sample of exit program.
	EXITETB3.c	**Obsolete**. Exit to program ETEBAC 3 cards. See the ETEBAC 3 Guide for more information.
	Chpi37.c	Sample of exit program to load pi37 and send it to the Partner. Pi37 is the PeSIT file label parameter.
Executable program	Pi37	Exit program to load pi37 and send it to the Partner. Pi37 is the PeSIT file label parameter.
	user	Exit program.

GRTF Directory

The GTRF directory is the central coordinator module and contains an executable program called tom_mon which is the monitor program, and an executable program called tom_api which is the API server program.

ITOM Directory

The ITOM directory contains all Application Utilities and includes the following files.

Type	File or Procedure	Description
Command procedures	samp*.sh	Sample for each Batch program.
	SAMPLES/*	Samples (C sources, Shell, etc)
	Convert_fic.sh	Sample for conversion of file from variable format to fixed format with padding and truncation.
Definition structure (.h)	d0b8z20.h	Description of the communication area with the Application Program Interface.
Linkable object library	libitom.a	
Executable program	p1b8pret	Restart of Transfer Request Program.
	p1b8preq	Request Program.
	p1b8pe2e	End to end pocess Program (Forward and EERP).
	p1b8pcan	Cancel Program.
	p1b8pren	Display RENC Program.
	p1b8ppur	Deletion Program.
	p1b8ppar_c	Creation RPAR Program.
	p1b8ppar_d	Display RPAR Program.
	p1b8ppar_m	Update RPAR Program.
	p1b8ppar_s	Deletion RPAR Program.
	p1b8pfil_c	Creation RFIC Program.
	p1b8pfil_d	Display RFIC Program.
	p1b8pfil_m	Update RFIC Program.
	p1b8pfil_s	Deletion RFIC Program.
	Convert_fic	Convert a file from variable format to fixed format.

The following source programs are also included in the ITOM/SAMPLES directory.

p1b8pret.c	p1b8preq.c	p1b8pcan.c	p1b8pren.c
p1b8ppur.c	display.c	analyse_display_requete.c	partner_enable.c
p1b8ppar_c.c	p1b8ppar_d.c	p1b8ppar_m.c	p1b8ppar_s.c
p1b8pfil_c.c	p1b8pfil_d.c	p1b8pfil_m.c	p1b8pfil_s.c

sterm Directory

The sterm directory is the operator interface module. It contains the executable program *tom_opr* which is the operator program.

strf Directory

The strf directory is the module that processes the transfers. It contains the following executable programs:

- tom_apm, the PeSIT transfer program.
- tom_ftp, the FTP transfer program.

Notif Directory

The Notif Directory contains files used to manage the sending and receiving of notification.

Type	File or Procedure	Description
Executable programs	tom_notify	The program that sends the notification.
	archnotlog	The program that archives the file notification.log.
	displaynot	Utility used for test: to display notification.
	recvnot	Utility used for test: to receive notification from a remote Sterling Connect:Express
	testrebas	Utility used for test: to control that a file name is valid
Command procedures	notcmd	Command procedure provided to manage tom_notify.
	start_archnotlog.sh	Command procedure that Sterling Connect:Express launches during initialization to activate archnotlog process.
	stop_archnotlog.sh	Command procedure that Sterling Connect:Express launches during termination to stop archnotlog process.
	notify.sh	Command procedure that Sterling Connect:Express launches to activate a tom_notify process that sends a notification.
Configuration files	notcfg.xml	Sample of configuration file to customize.
Definition structures	notif.h	Description of the notification record.

Httpn Directory

The Httpn Directory contains files used to manage HTTP notification.

HTTP notifications generated by the monitor are sent to the remote Sterling B2B Integrator by a specific executable tom_httpn. The monitor, tom_httpn and SI can be stopped or restarted independently, without loss of notifications.

Caution: Using HTTP notification with Sterling Connect:Express for UNIX is not compatible with the TCP notification described in « Sterling Connect:Express for UNIX. Agent_notification ».

Refer to the *Sterling Connect:Express Adapter for Sterling B2B Integrator* user guide, in *HTTP Notification in Sterling Connect:Express for UNIX* , for installing and configuring this option.

Type	File or Procedure	Description
Executable programs	tom_httpn	The program that sends the notification.
Command procedures	check_httpn.sh	Command procedure provided to check the tom_httpn process.
	install.sh	Installation procedure.
Configuration files	notformat.xml	Sample of notification format configuration file to customize.
	notformat_template.xml	Reference file for notification format configuration.
	vardef.txt	Environment variable definition (example).

Sterling Connect:Express Files

There are three types of Sterling Connect:Express files: configuration files, reporting files, and trace files. Configuration and reporting files are in the Config directory, and trace files are located in the gtrf and strf directories. This section identifies and describes these files.

Config Directory

To manage file transfers, Sterling Connect:Express uses the configuration files listed in the table below. This directory also includes files that are used for reporting.

File	Description
SYSIN	Sequential file created at installation. It is used for local environment definitions. You must modify the default values before using Sterling Connect:Express. If you modify values other than the STRACE, RBUILD, and NOTIFY fields, you must restart the Monitor. The STRACE, RBUILD, and NOTIFY fields can be dynamically set using the ch_conf utility. The * character at the beginning of a line means that the line is a comment. The keywords of the SYSIN file are described below.

File	Description
SYSTCP	Sequential file created at installation. You can update this file with an editor, and then restart the monitor. This file is used to declare an alternate TCPIP host or address for one partner. For an outgoing call, if the connection failed with the address/host in the partners directory, the address/host given in SYSTCP is used for the next retry. For an incoming call, if the first card of the file is TCPBYPASS, the incoming call is not checked. If the TCPIP address received is different from the one in the partners directory, GTRF searches for it in the SYSTCP file. See *Implementing SYSTCP* on page D-2.

The following table lists the keywords in the SYSIN file.

Keyword	Length	Description
AFMALL	1 character	Y=YES – $$ALL$$ accepted in the FTP list. N=NO – $$ALL$$ not accepted in the FTP list.
ALIASN	optional string	Optional alias name that is displayed in the main menu.
APPORT	5 numeric characters less than or equal to 65635	API Server port to listen for incoming API client calls.
APTIMR	5 numeric characters from 0 to 3600	This is the value of the timer that applies to the API protocol. It determines the number of seconds the API server will wait before cutting the session if no request is received from the client.The default is 60 seconds. If Sterling Connect:Express is administrated by Sterling Control Center, this parameter must match the corresponding parameter in Sterling Control Center
DEFILE	1 to 8 characters	Indicates the global default symbolic file name for FTP connections.
DEVDEF		First parameter: X25 device number available in system configuration (0 to 3). Second parameter: Number of X25 servers to initialize when the monitor starts. Third parameter: Number of X25 applications defined. Fourth and higher parameters: Name of X25 applications (1 to 8 characters, numeric or alphanumeric). Refer to Appendix E for more information about implementing X25.
DPCPSW	1 to 8 characters	Password of Local Monitor.
DPCSID	1 to 8 characters	Symbolic partner name of Local Monitor.
FTPORT	5 numeric characters less than or equal to 65635	FTP Server port to listen for incoming calls. This is only available with the FTP option. FTPORT is used for the commands, FTPORT - 1 for the data.
HTTPNF	1 numeric character	HTTP Notification. 1 - HTTP Notification facility is enabled 0 - HTTP Notification facility is disabled HTTPNF=1 requires that NOTIFY=0.
ISSTAT	1 numeric character	Statistics option. 1 - Statistics facility is enabled 0 - Statistics facility is disabled

Keyword	Length	Description
LAUNCH	1 character	C=Cold start. With a cold start, the monitor ignores the RENC file and all unfinished transfers are considered abnormally ended. H=Hot start. The monitor tries to restart transfers that were in progress when the monitor terminated.
MAXFTP	3 numeric characters from 1 to 128	Defines the maximum number of simultaneous inbound and outbound FTP file transfers at one time. Any new local outbound request will wait until a session becomes available.
MAXSES	3 numeric characters from 1 to 128	Defines the maximum number of simultaneous inbound and outbound PeSIT file transfers (secured and non secured) at one time. Any new local outbound request will wait until a session becomes available.
MAXLOG	3 numeric characters	Specifies the maximum number of rotated LOG files. Default value is 5. Minimum value is 2. Refer to *Appendix A, LOG File Rotation* for more information.
NOTIFY	1 numeric character	This is a notification flag. 0 1 - Notification option is enabled. 0 - Notification option is disabled.
PPDFLT	1 numeric character 1-9	Number of the default presentation table, form 1 to 9, that will be used if no table is specified in both the partner definition and the file definition.
PPFRST	1 character	Specifies which of the partner's presentation table or the file presentation table has priority. - F' indicates that the order is File first, Partner, Sysin. This is the default. - 'P' indicates that the order is Partner first, File, SYSIN. If none of the three are provided, the dummy presentation profile is used: no compression, no multi_record, no transcoding
RBUILD	1 numeric character	This is a rebuild base option flag. 0 - No automatic reorganization of the base. 1 - Automatic reorganization of the base when the monitor is launched or stopped. This key word is optional, and the default value is RBUILD=1.
SIZLOG	5 numeric characters	Defines the number of records in the LOG file when it is first created. Changing the SIZLOG value when the LOG file exists has no effect.
STIMEV	2 fields of 2 numeric value; unit is 1 minute	1st field: Time between connection retries. 2nd field: Time to wait before initiating a transfer request again.
STONER	1 numeric character	Controls the monitor starting process . 1 - The monitor and all the running child processes will be stopped if one of the child process cannot start. 0 -The monitor and all the running child processes will not be stopped if one of the child process cannot start.. This is the default.
STRACE	1 numeric character	This is a trace option flag. 0 means no trace. In this case, work files are not created. 1 means a minimal trace is active. Trace can be activated dynamically. You can use the command kill -USR1 pid to send the SIGUSR1 signal to the monitor and change the STRACE flag.

Keyword	Length	Description
STRFRN	4 numeric characters	This value specifies the number of simultaneous requestor file transfer executions and can be between 1 and 9999. 0 means that there is no control of the number of simultaneous requestor file transfer executions. This keyword is optional, and the default value is STRFRN=0.
SYSLOG	1 numeric character	This is the logging option flag. 0 disables this option. 1 means syslog support is active. In this case, informational or error messages are sent to the syslog daemon. See *Appendix A Sterling Connect:Express Messages* for descriptions of error messages.
TCPORT	5 numeric characters less than or equal to 65635	TCPIP port to listen for incoming calls. This in only available with the TCPIP option.
ZIPCMD	31 characters	Specify the command used to compress the LOG files. Default is “gzip”. If the command to use is not in the PATH then specify its full path name.
ZIPEXT	7 characters	Specify the extension of the compressed LOG files. Default is “gz”.
ZIPOPT	15 characters	Specify the options to use with the compress command. Default is -f gzip option.

The table below lists additional files that are included in the Config directory.

Caution: If one of the first 3 files is altered or suppressed, the monitor cannot be started. In this case, you must delete all 3 files that still exist or restore old ones.

File	Description
RPAR (.DAT,.IDX)	Indexed file formatted when you start the monitor. It is used for partner definitions and is updated using STERM or the application interface. You must match definitions with every partner.
RFIC (.DAT,.IDX)	Indexed file formatted when you start the monitor. It is used for symbolic file definitions and updated using STERM or the application interface. A symbolic file definition represents a pattern under which a physical data set will be transferred.
CERT (.DAT,.IDX)	Indexed file formatted when you start the monitor. It is used for ssl certificates and it is updated using STERM or the application interface.
SSLPARM (.DAT,.IDX)	Indexed file formatted when you start the monitor. It is used for ssl transfer parameters and it is updated using STERM or the application interface.
RDN (.DAT,.IDX)	Indexed file formatted when you start the monitor. It is used for ssl certificates control, and it is updated using STERM or the application interface.
RENC (.DAT,.IDX)	Indexed file formatted when you start the monitor. GTRF writes a record for every accepted transfer request in this file. Since one record is written for each request, the system administrator should purge RENC files through sterm or Itom regularly.
RTAB.DAT	Indexed file formatted when you start the monitor. It is used for session tables and updated using STERM. A session table represents session parameters used during a transfer.
RPRE.DAT	Indexed file formatted when you start the monitor. It is used for presentation tables and updated using STERM. A presentation table represents presentation parameters used during a transfer.

File	Description
LOG	Relative file formatted when you start the monitor. The size may be defined in the SYSIN file with the SIZLOG keyword. It is used for Sterling Connect:Express logging and may be viewed using STERM. This file is a wrapped file. All messages are in English.

Using the ch_conf Command

The operator command ch_conf lets you dynamically modify some configuration parameters from the SYSIN file, and the SYSTCP file. You can pass the command in three ways:

- Using the environment variable, $ch_conf
- Launching the shell procedure $TOM_DIR/config/ch_conf.sh
- Using the command $TOM_DIR/config/ch_conf *argument*
  (argument is one of the parameters shown in the parameter listing that follows.)

Type $ch_conf to display the following list of configuration parameters for the command:

```
************************************************************
*            Update configuration parameters               *
************************************************************
*                                                          *
*   1 --> Activate trace                                   *
*                                                          *
*   2 --> Stop trace                                       *
*                                                          *
*   3 --> Activate the database re-build process at        *
*             next initialization of Sterling Connect:Express        *
*                                                          *
*   4 --> Disable the database re-build process at next    *
*             initialization of Sterling Connect:Express             *
*                                                          *
*   5 --> Reload the SYSTCP file                           *
*                                                          *
*                                                          *
*   x --> Exit                                             *
*                                                          *
************************************************************

Enter your choice :
```

The choices are equivalent to using the command as shown below:

Command	Description
./ch_conf /STRACE=1	Activate trace.
./ch_conf /STRACE=0	Stop trace.
./ch_conf /RBUILD=1	Activate the database rebuild process at next initialization of Sterling Connect:Express.

Command	Description
./ch_conf /RBUILD=0	Disable the database rebuild process at next initialization of Sterling Connect:Express.
./ch_conf /SYSTCP	Reload the SYSTCP file.

GTRF and STRF Directories

Trace files are located in the GTRF and STRF directories.

The gtrf directory contains the file TOM_OUT.pid, which is an output file. A new version is created each time the monitor GTRF is started. The STRACE option in the SYSIN file must be on. The trace files can be deleted without disturbing the monitor.

The gtrf directory also contains the trace files of the tom_api process. To enable traces in this process create a file called ‘apitrace.flag’: SYYYYMMDD_HHMMSS_000 file is the trace file for the tom_api server thread. CYYYYMMDD_HHMMSS_nnn are trace files for tom_api client sessions generated by the server.

The strf directory contains the following files:

- RX_QQQNNNNN.pid
- RT_QQQNNNNN.pid
- RF_QQQNNNNN.pid
- SF.pid
- ST.pid

They are output files of the strf process, identified by #pid. The first character is the session direction, R for Requester or S for Server. The Second character is T for TCPIP or F for FTP. In requester mode only, QQQNNNNNN represents the request number processed. Several requests can be chained together in one strf process. This process executes an image file tom_apm (for TCPIP) or tom_ftp (for FTP). These files can be deleted without disturbing the gtrf monitor.

Managing RENC Files

The RENC file contains one record per request processed by GTRF. When Sterling Connect:Express starts, it processes automatic restarts for eligible requests if LAUNCH is set to H for Hot launch.

The RENC file can get very large, so the administrator should purge this file periodically, using p1b8ppur or the sterm Deletion Screen, to improve response time. Each request must be deleted after 6 months. If not, the GTRF monitor will try to create a new request with the same request number and will fail.

Information in this file can be accessed by applications through the ITOM utilities of the L0B8Z20 display function and p1b8pren.

Sterling Connect:Express Environment Variables

The following Environment variables are available, to manage special configurations.

Environment variable	Description
$CNX_MSG	This variable is used to provids the PeSIT PI99 field to be set in the CONNECT or ACONNECT data units. If set, this is the default.
CNX_PRF	This variable can be used to enable the PeSIT ANY protocol feature. Valid values for this variable are CFT or ANY.
CX_MAX_MSGSND_RETRIES	This variable is used to set number of retries of reading a message from a Connect:Express message queue. Do not use this variable without the guidance of the Sterling Connect:Express support team.
CX_MSGSND_RETRIES_DELAY	This variable is used to set the delay between 2 retries of reading a message from a Connect:Express message queue. Do not use this variable witout the guidance of the Sterling Connect:Express support team.
$EXIT_TRC	This variable is used to change the directory of the user exits and user commands output files. The default is $TOM_DIR/exit.
$GTRF_TRC	This variable is used to change the directory of the tom_mon process trace file. The default is $TOM_DIR/gtrf.
$HTTPN_TRC	This variable is used to change the directory of the tom_httpn process trace file. The default is $TOM_DIR/httpn.
$KEEP_STATS	This variable defines the size of the statistics files table. The default is 7.
$PURGE_STATS_TIME	This variable is used to define the time at which the statistic files are purged. The format of the time is hh:mm:ss. The default is 00:30:00.
$REC_EMPTY	This variable is used to enable or disable sending empty records. 0=Disabled, 1=Enabled. The default is Enabled. Disabled means that an empty record is replaced by a space character.
$RTR_TIMER	This variable is used by the l0b8z20 interface to set the maximum delay to wait before tom_mon sends a reply. The default and minimum value is 30 seconds. The maximum value is 65535 seconds.
$STRF_TRC	This variable is used to change the directory of the tom_apm and tom_ftp processes trace files. The default is $TOM_DIR/strf.
$THR_PI42	This variable is used to define a percentage of error on the file size information received from the remote sender (PeSIT pi42 parameter). Use this parameter if a remote partner is sending unreliable information.
$TOM_API_TCP_TIMEOUT	This variable is used to define a timer value in seconds for the tom_api process. You can also set the APTIMR= parameter in the SYSIN file of Sterling Connect:Express.

Environment variable	Description
TOM_HTTPN_OUT_FILE	This variable is used to change the name of the file to which the standard and error streams of the tom_httpn process are redirected. The default is $TOM_DIR/httpn/tom_httpn.out
TOM_LANG	This variable is used to change the language of the sterm menus. Default is FR (for French). Other possible value is US (for US English).
TOM_LOG_DIR	This variables is used to change the directory of the LOG files. The default is $TOM_DIR/config.
TOM_OPR_DUMP_FILE	If this variable is set with a valid file name, hitting the control key F4 will dump the content of the current sterm module screen in that file.
$TOM_SSL_INSERT_EMPTY_FRAGMENTS	This variable is used to disable the SSL empty fragment option. Some softwares don't support the SSL empty fragment mechanism. If this variable is defined, this disables the mechanism.
$'*partner*'_MSG	This variable provides the PeSIT PI99 field to be set in the CONNECT or ACONNECT data units when establishing a PeSIT connection with '*partner*'. If set, this overrides $CNX_MSG, and this is the default for '*partner*'. $'partner'_MSG=&PI99 returns the PI99 received in the CONNECT data unit back to the remote requestor.
$'*partner*'_IMSG	This variable provides the PeSIT PI99 field to be set in the ACONNECT data units when establishing an inbound PeSIT connection with '*partner*'. If set, this overrides $CNX_MSG, and $'*partner*'_MSG.
$'*partner*'_OMSG	This variable provides the PeSIT PI99 field to be set in the CONNECT data units when establishing an outbound PeSIT connection with '*partner*'. If set, this overrides $CNX_MSG, and $'*partner*'_MSG.

Sterling Connect:Express Transfers

There are three prerequisites for a file transfer using Sterling Connect:Express.

1. The file must be defined in the file directory (RFIC) with a keyword called a symbolic file name. The symbolic file name must match the symbolic file name on the Partner participating in the exchange.
2. Each Partner must be defined in the partner directory (RPAR) with a symbolic Partner name.
3. The transfer direction for this Partner must be authorized in the file directory (RFIC).

Physical data set names and file characteristics can be independent between partners, but you can also exchange file physical names with partners. The physical data set name can be fixed for a file transfer or you can define a file pattern by automatically generating a physical data set name.

Partners can have specific addresses and be specifically identified and under access control. Using SYSTCP enables you to define a TCP/IP partner pattern.

Note: See *Appendix D Implementing Special Features* for information about implementing physical data set names or implementing SYSTCP.

Chapter 2

Installation

This chapter describes the installation requirements and procedure for Sterling Connect:Express for UNIX.

Installation Requirements

The following sections describe the hardware, software, and memory requirements for Sterling Connect:Express for UNIX, and identify installation materials.

Hardware Requirements

The following hardware is needed to run Sterling Connect:Express for UNIX:

- Any supported UNIX computer.
- For TCPIP: a card and IP connection
- A VT320 (or later) emulation or terminal
- At least one other type of computer that supports TCPIP

Approximately 25 Mbytes are needed for Sterling Connect:Express for UNIX. These space requirements are divided among executable programs and command procedures. Sterling Connect:Express for UNIX must be installed on the local file system. It cannot be installed on a Network File System (NFS) partition.

Software Requirements

Software requirements for Sterling Connect:Express are listed below:

- An IBM RISC under AIX 6.1 or later
- A Sun with Solaris starting from version 5.10
- A 64 bit Linux version with GLIBC >= 2.7

Sterling Connect:Express Installation Materials

The Sterling Connect:Express installation kit contains the necessary compressed tar (Tapes Archives) files to install the product. The installation kit is named `CX_`*`OS`*`_150-ffiih.tar.gz`, where '`OS`' indicates the operating system where you are to install the product.

The values for '`OS`' are:

- `aix` for AIX;
- `linux-x86_64` for Linux 64 bit;
- `sunos` for Sun.

The value `ffiih` is decomposed in:

- `ffiih` is the fix pack version;
- `ffiih` is the interim fix version;
- `ffiih` is a hot fix number.

Installing Sterling Connect:Express

The installation process includes the following steps. Each step is described in more detail in the sections below.

1. Check the hardware and sofware requirements are met;
2. Download the Sterling Connect:Express for UNIX installation kit.
3. Execute the install.sh Shell procedure.
4. Modify and run customized command procedures, if needed.
5. Update the SYSIN file.
6. Adjust parameters and activate the monitor.

Caution: If you are planning to use different users for installing and executing a same instance of Sterling Connect:Express it is important that these 2 users belong to the same group.

Step 2 Download Archives

Confirm that you have sufficient space on the file system where you want to install Sterling Connect:Express for UNIX. The product requires 10 MB.

Type the following command from the directory where you want to download installation files:

```
tar xzf CX_OS_150-ffiih.tar.gz
```

Step 3 Execute the Installation Procedure

Log in to your UNIX system as the user that will be the default Sterling Connect:Express user. You may need to create the user ID, if it is a new user. Execute the installation procedure *install.sh -h*. The following screen shows an example of the output.

```
Usage: install.sh [OPTIONS]
Install IBM Sterling Connect:Express Version 150.17000

OPTIONS can be any of the following:
    -h             display this help
    -i             do not stop script execution on warnings
    -d directory   CX installation directory
    -e edition     select the edition to use; must be one of the following:
                     p    Premium Edition
                     pnp  Premium Edition for Non-Production Environment
                     s    Standard Edition
                     snp  Standard Edition for Non-Production Environment
    -n             do not backup the existing installation
    -o             overwrite the default translation tables
    -s             silently install CX; requires -e options for an update,
                   -d and -e for a first install, and possibly -i option
                   if upgrading from an older version than 150.12.
```

You can install Sterling Connect:Express interactively or silently if you specify the *-s* option on the command line. The *-s* option requires you also specify the *-d* and *-e edition* options.

The *-d directory* option determines in which directory you want to install Sterling Connect:Express. The last directory index specifies the name of the monitor and it cannot exceed 4 characters.

Caution: If you are planning to use different users for installing and executing a same instance of Sterling Connect:Express it is important that these 2 users belong to the same group.

Because the same script is used to upgrade an existing instance options *-n* and *-o* are only relevant for an update.

Step 4 Run Customized Procedure

To make the new environment available to your current shell, log out and login. You can also execute the $HOME/.profile file, as shown below. Be sure to include the dots in the command, as needed.

```
. $HOME/.profile
```

Step 5 Update the SYSIN File

Follow the instructions in the letter that came with your Sterling Connect:Express shipment. Complete the following steps to update the SYSIN file.

1. Use a text editor to edit the SYSIN file. Do not modify the number at the end of lines.
2. Replace CETOM1 with your local Sterling Connect:Express symbolic name (up to 8 characters).

```
DPCSID=CETOM1
```

3. Replace PASSWD with your local Sterling Connect:Express password. The password can contain up to 8 characters.

```
DPCPSW=PASSWD
```

4. Replace the first 01 with the number of minutes to retry after a connection attempt. Replace the second 01 with the number of minutes to wait before retrying a transfer.

```
STIMEV=(01,01)
```

5. Replace 5000 with the number of records in your LOG file.

```
SIZLOG=5000
```

6. Enter an H for a hot start or a C for a cold start. This field tells the monitor how to start.

```
LAUNCH=H
```

7. 05000 is the port number of the TCPIP on which Sterling Connect:Express listens for incoming calls. If you do not require TCPIP support, or if the TCPIP option is not valid for your installation, enter a comment for this parameter by typing the * character in the first position.

```
TCPORT=05000
```

8. Leave this field at 1. This field enables or disables the syslog. A 1 means the syslog is active, a 0 disables the syslog. See *Appendix A Sterling Connect:Express Messages* for more information about SYSLOG activity.

```
SYSLOG=1
```

9. This is the TCP/IP port number that listens for the FTP protocol. If you do not need an FTP server or if you will not use the TCP/IP option, enter a comment for this parameter by typing the * character in the first position. The monitor must be authorized to listen to the port number that you specify in this field.

```
FTPORT=06000
```

10. Enter the global default symbolic file name for FTP connections.

```
DEFILE=FTPFILE
```

11. Specify if you want TRACE files to be created for monitor activity. Enter a 0 for No trace, or a 1 to activate this feature.

```
STRACE=0
```

12. Specify if you want automatic reorganization of the base. Enter a 0 for no automatic reorganization, or a 1 to automatically reorganize the base when the monitor is launched and stopped.

```
 RBUILD=1
```

13. Specify the number of simultaneous requestor file transfer executions. Enter a value between 1 and 9999, or a 0 to specify that there is no control.

```
STRFRN=0
```

14. Specify if you want Sterling Connect:Express to include the files accessible to all the partners ($$ALL$$) in the FTP list .

```
AFMALL=N
```

15. Specify if you want Sterling Connect:Express to activate the notification facility (Notify option: 1=ON , 0=OFF).

```
NOTIFY=0
```

16. This is the TCP/IP port number that listens for the API protocol. If you do not need an API server or if you will not use the API option, enter a comment for this parameter by typing the * character in the first position.

```
APPORT=07000
```

17. This is the value of the timer that applies to the API protocol. It determines the number of seconds the API server will wait before cutting the session if no request is received from the client.The default is 60 seconds. If Sterling Connect:Express is administrated by Sterling Control Center, this parameter must match the corresponding parameter in Sterling Control Center.

```
APTIMR=300
```

18. Specify if you want Sterling Connect:Express to activate the http notification facility (http Notification option : 1=ON, 0=OFF). If HTTPNF=1, NOTIFY should be 0.

```
HTTPNF=0
```

19. Specify if you want Sterling Connect:Express to activate the statistics facility (Statistics option : 1=ON, 0=OFF)

```
ISSTAT=0
```

20. Specify the maximum number of simultaneous PeSIT sessions that you want Sterling Connect:Express to support. This parameter enables you to keep control according to the terms of your license. The default maximum number of simutaneous sessions that Sterling Connect:Express supports is 50.)

```
MAXSES=5
```

21. Specify the maximum number of simultaneous FTP sessions that you want Sterling Connect:Express to support. This parameter enables you to keep control according to the terms of your license. The default maximum number of simutaneous FTP sessions that Sterling Connect:Express supports is 50.)

```
MAXFTP=5
```

22. Specify if the limited character set is used for symbolic names and passwords. By default the limited character is used. Set the value to 0 to use the extended character set. The limited character set includes all uppercase letters, digits, space, underscore, hyphen, star and dash while the extended character set includes all printable characters.)

```
CAPSON=0
```

23. Specify if the monitor, and all running child processes, should stop if one of the child processes cannot start during initialization)

```
STONER=1
```

24. Specify the number of the default table, form 1 to 9, that will be used if no table is specified in both the partner definition and the file definition.)

```
PPDFLT=4
```

25. Specify which of the partner's presentation table or the file presentation table has priority: 'F' indicates that the order is File first, Partner, Sysin. 'P' indicates that the order is Partner first, File, sysin. If none of the three are provided, the dummy presentation profile is used: no compression, no multi_record, no transcoding.)

```
PPFRST=P
```

Specifying TCP/IP Stack for TCPORT and FTPORT

When using a port number, Sterling Connect:Express listens to this port on all available TCP/IP addresses on this port. You can, however, specify an IP address so that Sterling Connect:Express only listens on a specific device. For example, if your system has two Ethernet cards with the addresses 111.22.33.44 and 555.66.77.44, and you want Sterling Connect:Express to listen on those 2 devices, enter the following information in your SYSIN file:

```
TCPORT=05000                              5 TCPIP PORT
```

If you want Sterling Connect:Express to listen on 1 of those 2 devices (111.22.33.44 address), enter the following line in your SYSIN file:

```
TCPORT=111.22.33.44:05000                 5 TCPIP PORT
```

When the monitor has started, you can verify that the listener is using the correct device using the netstat -n command.

Specifying TCP/IP Stack for APPORT

Like for TCPORT and FTPORT if you want Sterling Connect:Express API server to listen on a specific device (for example the 111.22.33.44 address), enter the following line in your SYSIN file:

```
APPORT=111.22.33.44:05000:
```

When the monitor has started, you can verify that the api is using the correct device using the netstat -n command.

Step 6 Kernel Parameters

Inter Processes Communication (IPC) in Sterling Connect:Express is based on message queues. Verify the following kernel parameters:

```
MSGMAX >= 8192
MSGMNB >= 8192
MSGMNI >= 1024
```

Note: On Solaris 10 operating system, the names of some of these parameters have been changed. `MSGMNB` is `max-msg-qbytes` and `MSGMNI` is `max-msg-ids`. If you cannot find the above kernel parameters, check with your system administator how to find them and change their values.

Step 8 Activate the Monitor

Ensure that Sterling Connect:Express environment variables such as start_tom and sterm are set. Environment variables are defined in the $HOME/.profile file. Type **print env** at the prompt to check environment variables, then type **$start_tom** at the prompt to activate the monitor.

Implementing Sterling Connect:Express for UNIX

To begin implementing Sterling Connect:Express, complete the following steps:

1. Log onto the Sterling Connect:Express account.
2. Enter $start_tom at the prompt to start Sterling Connect:Express, as shown below.

```
PROMPT> $start_tom
```

start_tom is an environment variable defined in the $HOME/.profile file.

```
start_tom=$TOM_DIR/gtrf/tom_mon
```

To start Sterling Connect:Express outside of this account, you must export the TOM_DIR variable using the following commands:

```
TOM_DIR=(replace with the Sterling Connect:Express home directory)
export TOM_DIR
```

3. Type $sterm, sterm_c, or sterm_v to access the Sterling Connect:Express Main Menu with all or limited functionalities.

```
PROMPT> $sterm or $sterm_c or $sterm_v
```

sterm, sterm_c, and sterm_v are environment variables defined in the $HOME/.profile file.

```
sterm=$TOM_DIR/sterm/tom_opr
sterm_c=$TOM_DIR/sterm/tom_opr C
sterm_v=$TOM_DIR/sterm/tom_opr V
```

Note: See *Activating sterm* on page 2-9 for using the $sterm commands.

4. Stop Sterling Connect:Express. Stop GTRF using one of the following commands.

```
PROMPT> $stop_tom (for immediate stop)
PROMPT> $stop_tom_l (for deferred stop)
```

stop_tom and stop_tom_l are environment variables defined in the $HOME/.profile file.

```
stop_tom=$TOM_DIR/config/stop_tom
stop_tom_l=$TOM_DIR/config/stop_tom L
```

Note: Use stop_tom_l to stop the monitor after all current transfers are completed. All new transfer requests are recorded but they are not scheduled. Inbound transfer requests are rejected. Using stop_tom interrupts all current transfers; results are unpredictable.

The sterm Module

The sterm module is an operator interface that enables communication between an operator and the Sterling Connect:Express monitor. sterm provides input screens for entering or modifying transfer parameters, and display screens for monitoring transfer activity. sterm enables you to:

- Update Sterling Connect:Express session tables and Files and Partners directories.
- Create SSL server, SSL client and X.509 certificate definitions.
- Display and list Sterling Connect:Express configuration information.
- Enter a transfer request.
- Monitor request activity.

Activating sterm

To activate sterm, type $sterm, $sterm_c, or $sterm_v at the prompt. sterm can be activated by any user whose environment contains the TOM_DIR variable.

sterm waits for the monitor to respond. If Sterling Connect:Express has not started or is not initialized, a prompt is displayed. sterm can also be activated with a parameter. For example, $sterm 5 indicates that the LOG display screen will automatically refresh every 5 seconds.

You can configure sterm in three ways, as described in the following table.

Configuration	Functions	Use
Full sterm	All functions are enabled.	Use $sterm or $TOM_DIR/sterm/tom_opr
Client sterm	Display configuration Display activity Use file transfers service	Use $sterm_c or $TOM_DIR/sterm/tom_opr C
Display sterm	Display configuration Display activity	Use $sterm_v or $TOM_DIR/sterm/tom_opr V

sterm Structure

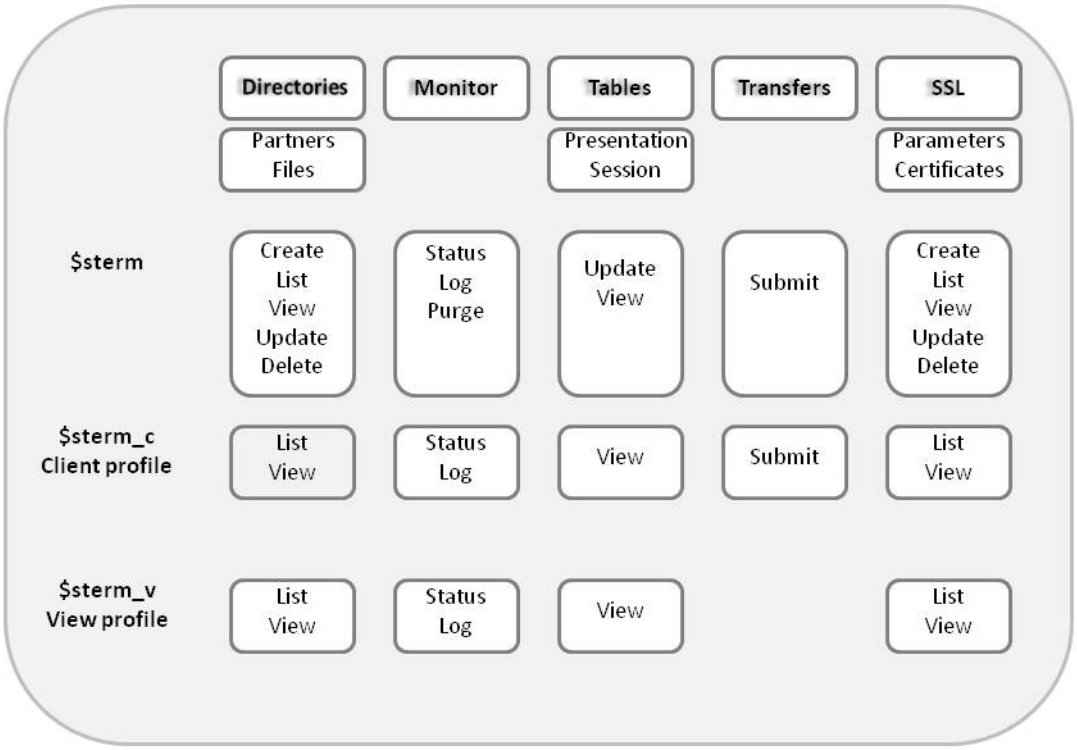
The sterm structure has the following three levels

Level	Description
0	Main menu.
1	Function selection: create, list, view, update, delete
2	Function processing.

The active selected line is displayed in reverse video. You can use the following keys.

Key	Description
<UP> and <DOWN>	Move through fields in a menu.
<LEFT> and <RIGHT>	Move within a field.
<RETURN>	Confirm a field entry.
<TAB> key or <SPACE> bar	Clear a field.
<F3> or <F9>	Quit a function (sometimes 'X').
<F8>	Confirm global menu.

The following figure illustrates the structure of the sterm operator interface and each menu. The client profile ($sterm_c) limits access to view configuration, submit and monitor file transfers. The View profile ($sterm_v) limits access to view configuration and activity.


Directories
Monitor
Tables
Transfers
SSL
Partners
Files
Presentation
Session
Parameters
Certificates
$sterm
Create
List
View
Update
Delete
Status
Log
Purge
Update
View
Submit
Create
List
View
Update
Delete
$sterm_c
Client profile
List
View
Status
Log
View
Submit
List
View
$sterm_v
View profile
List
View
Status
Log
View
List
View

Chapter 3

Directories and Tables

This chapter discusses the Partners and Files directories, Monitor management, and Sterling Connect:Express tables.

Accessing the Main Menu

This manual assumes that Sterling Connect:Express for UNIX is installed in the /home/tom1 directory. To communicate with one monitor with sterm, you must define an environment variable TOM_DIR in your shell in the root directory of installation for example, /home/tom1.

To access the Main Menu, you must establish communication between sterm and gtrf.

1. At the system prompt, type **$sterm** to establish communication and log in directly to the installed monitor. sterm is an environment variable customized at installation and equal to $TOM_DIR/sterm/tom_opr. If

you have more than one monitor, export TOM_DIR to the root directory for each monitor. The following screen displays the Sterling Connect:Express Main Menu.

```
C:X/UNIX 150 --------------- MAIN MENU (GLOBAL) ------------------------ ce01
OPTION ===>                                                        LABS LINUX




                    IBM(R) Sterling Connect:Express(R)
     Licensed Material, Property of IBM. (C) Copyright IBM Corp. 1999, 2012

_ 1  DIRECTORIES     _ 2   MONITOR     _ 3   TABLES           _ 4   REQUEST

     PARTNERS              STATUS            SESSION          _ 5   SSL

     FILES                 LOG               PRESENTATION

                           REQUEST DELETION




  X  EXIT                                                    -F3- END
```

2. Type the number of your selection in the Option field and press <Enter>. The following screen displays the Directories Management menu.

```
C:X/UNIX 150 ------------ DIRECTORIES MANAGEMENT  ----------------------- ce01
OPTION ===>




            1 P-PARTNERS      PARTNERS DIRECTORY MANAGEMENT

            2 F-FILES         FILES DIRECTORY MANAGEMENT









  X  EXIT                                                    -F3- END
```

The Partners Directory

A Partner is any data processing equipment that is linked by media and has file transfer protocols supported by Sterling Connect:Express. A Partner is identified by a symbolic name chosen by the user. The symbolic name can have 1 to 8 characters.

The Partners directory (RPAR) is an indexed file that is initialized by the GTRF process the first time that you start Sterling Connect:Express. If the RPAR file does not exist, Sterling Connect:Express creates it.

Managing the Partners Directory

You can add, list, update, or delete Partners from the Partners Directory Management screen.

1. From the Main Menu, type **1** in the Option field and press <Enter>. Sterling Connect:Express displays the Directories Management screen.
2. Type **1** or **P** in the Option field and press <Enter> to select the Partners Directory Management option. The following screen shows the Partners Directory Management screen.

```
C:X/UNIX 150 --------- PARTNERS DIRECTORY MANAGEMENT -------------------- ce01
OPTION ===> V



            A    ADD
            L    LIST
            U    UPDATE
            D    DELETE
            V    VIEW


                 PARTNER    ===> EXPRESS1









  X  EXIT                                                    -F3- END
```

3. Type a command in the Option field and press <Enter>. The cursor moves to the Partner field. The following table describes each command.

Command	Description
A	Adds a Partner to the Partners directory.
L	Lists all Partners in the Partners directory. Partner information displays on several screens. Use F10 and F11 to scroll horizontally.

Command	Description
U	Updates the Partner record that you specify in the Partner field.
D	Deletes the Partner record that you specify in the Partner field.
V	Displays the Partner record that you specify in the Partner field.

4. In the Partner field, enter the Partner name that you want to add or modify.

The following screen shows the fields for a Partner definition in the Partners directory. The UPD field displays the date and time of the last update.

.Enter information as described in the following table. Press <Enter> to move from field to field.

```
C:X/UNIX 150 --------------- PARTNERS DIRECTORY ------------------------ ce01
OPTION ===>

SYMBOLIC NAME        :    GFIPSR
PASSWORD ........... :    PSR             PASSWORD OF PARTNER
INITIALIZATION STATUS . : E               E:ENABLE     H:DISABLE
PARTNER TYPE ......... : T               T/O
PROTOCOL NUMBER ....... : 3               1:ETEB3,2:FTP,3:PESIT,A:PESIT ANY
SESSION / PRESENTATION  : 6 / 2           TABLES   1->9 / 1->9
MAX. NO. CONNECTIONS .. : 10/10/10        01->64  TOT/IN/OUT
TYPE OF CONNECTION .... : T               T
TCPIP HOST ............ : mvs
PORT .................. : 07000
TCPIP ADDRESS ......... :
DPCSID ALIAS .......... :                 SLL PARM ID ...:
DPCPSW ALIAS .......... : PSR             CERTIFICATE CONTROL   :
DEFAULT FILE .......... :
NUMBER OF RETRIES ..... : 65              INTERV.SESS,TRF: 25, 10 MINUTES


OPTION : VIEW                             UPD : 98/08/04 10:41 root
-ENTER- NEXT FIELD               -F3- CANCEL                  -F8- COMPLETION
```

Fields	Length	Description
SYMBOLIC NAME	1 to 8 characters	This name identifies a trading Partner, and must be unique. The Partner symbolic name is verified before accepting an internal request or accepting an inbound session . You can define a default partner profile in the partners directory, called $$PART$$, that will be used to manage connections with partners that are not defined in the directory. The $$PART$$ is available for accepting any inbound connection, or any outbound request submission. It can be used to manage outbound connections to a single server (one network address) on behalf of several trading partner. Creating $$PART$$ profile means that you accept connnections with any unknown partner. To disable anonymous connections, disable $$PART$$ profile, or delete it from the partners directory. **Note**: $$...$$ syntax is reserved, and a partner symbolic name cannot be equal to character '*' or start with character '#'.
PASSWORD	0 to 8 characters	This string is verified before opening an input session. Setting this field to $$NONE$$ disables the password control.
INITIALIZATION STATUS	1 alphabetic character	E = Enabled. This Partner can execute transfers. H = Held. This Partner cannot execute transfers.
PARTNER TYPE	1 alphabetic character	T = Partner has Sterling Connect:Express software O = Partner has another software Extended features are different depending on the Partner type. See Appendix D for information about managing physical data set names.
PROTOCOL NUMBER	1 numeric character	Identifies the protocol for this Partner. 1= ETEBAC 3 2= FTP 3 = PeSIT A = PeSIT ANY If PeSIT ANY is set, a specific PeSIT protocol feature is enabled to be compatible with a CFT partner that would not be able to change configuration. Normal value is N.
SESSION TABLE NUMBER	1 numeric character	(1 - 9) Indicates the session parameters to use with this partner. See *Updating Session Tables* in this chapter for more information.
PRESENTATION TABLE NUMBER	1 numeric character	(1 - 9) Indicates the presentation parameters to use with this partner. See *Updating Presentation Tables* in this chapter for more information. Depending on the PPFRST parameter of the SYSIN file, the partner presentation table (PPFRST=P) or the file presentation table (PPFRST=F) has priority. If no presentation table is provided in the partner and the file definitions, the default defined in the SYSIN file is used.
MAX. NO. CONNECTIONS (TOT/IN/OUT)	6 numeric characters format XX/YY/ZZ	XX: This is the total number of simultaneous sessions that Sterling Connect:Express can conduct with this Partner. YY: This is the number of simultaneous input sessions that Sterling Connect:Express can conduct with this Partner. ZZ: This is the number of simultaneous output sessions that Sterling Connect:Express can conduct with this Partner. You can use these values to control transfer activity with a remote Partner.
TYPE OF CONNECTION	1 alphabetic character	T = Partner is linked by TCPIP

Fields	Length	Description
TCPIP HOST	1 to 127 characters	The TCPIP host name or TCP/IP address. This field must be completed if the TCPIP Address is blank.
PORT	1 to 5 numeric characters	The TCPIP port of the remote Partner.
TCPIP ADDRESS	7 to 15 dotted numeric characters	The TCP/IPV4 address in the format A.B.C.D. If both the host and address are entered, the address is used. If the address is not entered, an incoming call IP address is not checked.
DPCSID ALIAS	1 to 8 characters	Overwrites the current DPCSID in the SYSIN file. The Remote Host must know your Local Host by its DPCSID name.
SSLPARMID	1 to 8 alphanumeric characters	This field identifies a SSLPARM profile. SSL Transfer Profiles are managed by Option 5 SSL of the main menu. Refer to *Sterling Connect:Express for UNIX SSL User Guide*.
DPCPSW ALIAS	1 to 8 characters	This password overwrites the current DPCPSW in the SYSIN file. The Remote Host will control this DPCPSW password. Setting this field to $$NONE$$ will result in setting the local passsword to blanks
CERTIFICATE CONTROL	1 to 8 alphanumeric characters	This field identifies a Certificate Control profile. Certificate Control Profiles are managed by Option 5 SSL of the main menu. Refer to *Sterling Connect:Express for UNIX SSL User Guide*.
DEFAULT FILE	1 to 8 characters	This is a symbolic file name from the Files Directory. It is the default symbolic file name used with both the FTP and the PeSIT protocols for this Partner. PeSIT : If the file name used for transfer is not defined in the directory, Connect:Express will use the partner's default profile. If no default file name is defined for the partner, Connect:Express will use the general default $$FILE$$ profile if one has been defined in the files directory. FTP : the partner's default file name is used if no file name is provided in the transfer request, or if the file name provided is not defined in the file directory. if no default is defined for the partner, the default file from the SYSIN is used. if no default is defined in the SYSIN, the $$FILE$$ profile is used, if it has been defined in the directory. Any default file must be defined in the file directory. To disable the default file process, disable the default profile, or delete it from the file directory.
NUMBER OF RETRIES	1 to 2 numeric characters	This field indicates the number of authorized transfer restarts or connection retries. It can be a value from 0 to 99. If you enter a value of 0, the Partner is disabled after the first call if the call packet is not accepted.
INTERV.SESS	1 to 2 numeric characters	Time between connection retries. It can be a value from 0 to 99. Units are minutes.
INTERV.TRF	1 to 2 numeric characters	Time between transfer restarts. It can be a value from 0 to 99. Units are minutes.

5. Press <F3> to exit from this screen. The cursor moves to the field **Do you want to go on**. You can also press <F8> to move to this field without pressing <Enter> through each field.
6. Type **Y** or press <Enter> to confirm your entries.

The Files Directory

A File contains the local rules for transferring a collection of data on a disk or tape. This data is organized sequentially and can be either fixed or variable. The file is identified by an 8-character symbolic name. The symbolic file name must be the same on both Sterling Connect:Express systems sending or receiving this data.

The Files directory (RFIC) is an indexed file that is initialized by the GTRF process the first time that you start Sterling Connect:Express. If the RFIC file does not exist, Sterling Connect:Express creates it.

Note: See also *Implementing a Parameter Cards File* in Appendix D.

Managing the Files Directory

You can add, list, update, or delete Files from the Files Directory Management screen.

1. From the Main Menu, type **1** in the Option field and press <Enter>. Sterling Connect:Express displays the Directories Management screen.
2. Type **2** or **F** in the Option field and press <Enter> to select the Files Directory Management option. The following screen shows the Files Directory Management screen.

```
C:X/UNIX 150 ----------- FILES DIRECTORY MANAGEMENT --------------------- ce01
OPTION ===> V



            A    ADD
            L    LIST
            U    UPDATE
            D    DELETE
            V    VIEW



                 FILE       ===> DEFAULT.
  X  EXIT                                                  -F3- END
```

3. Type a command in the Option field and press <Enter>. The cursor moves to the File field. The following table describes each command.

Command	Description
A	Adds a File to the Files directory.
L	Lists all Files in the Files directory. File list information displays on several screens. Use F10 and F11 to scroll horizontally.
U	Updates the File record that you specify in the File field.
D	Deletes the File record that you specify in the File field.
V	Displays the File record that you specify in the File field.

4. Type the symbolic file name of the file you want to create or modify and press <Enter>. The Files Directory screen is displayed. The UPD field displays the date and time of the last update.

```
C:X/UNIX 150 ---------------- FILES DIRECTORY -------------------------- ce01
OPTION ===>

SYMBOLIC NAME        :    DEFAULT

INITIALIZATION STATUS . : E              E:ENABLE     H:DISABLE
DIRECTION (INQUIRY) ... : *  ( N )      T:TRANSMIT R:RECEIVE *:EITHER    (Y)
RECEIVING PARTNER ..... : $$ALL$$        'NAME',#LISTE, $$ALL$$
TRANSMITTING PARTNER .. : $$ALL$$        'NAME',#LISTE, $$ALL$$
PRIORITY .............. : 0              0:URGENT 1:FAST 2:NORMAL
DEFINITION TYPE ....... : D              D:DYNAMIC F:FIXED
PRESENTATION TABLE .... : 5              1 -> 9 PRESENTATION TABLE

PARAMETER CARDS FILE    : Y              Y/N
SPACE TO RESERVE ...... : N              Y/N
ALLOCATION RULE ....... : 0              0:INDIF., 1:PREALL., 2:TO CREATE
RECORD FORMAT ......... : TV             TF, TV, BF, BU, T*, B*, **
RECORD LENGTH ......... : 00255          1-5 NUMERIC CHARAC.
REMOTE DSN (FTP) ...... : /usr/products/update.txt
TYPE/STRUCTURE/MODE FTP : ARS            E/A/I/*,F/R/*,B/S/*
STORE UNIQUE (FTP) .... : N              Y/N         FA :    Y/N NOT: 0  (0-7)

OPTION : VIEW                            UPD : 09/01/26 14:59 mverz1
-ENTER- NEXT FIELD               -F3- CANCEL                 -F8- COMPLETION
```

```
C:X/UNIX 150 ---------------- FILES DIRECTORY -------------------------- ce01
OPTION ===>

SYMBOLIC NAME        :    DEFAULT         DEFINITION : D   DIRECTION : R

PHYSICAL NAME ......... : /tmp/TOM_&REQNUMB.tmp



TRANSMISSION :
START EXIT ............ : ............
START COMMAND ......... : ............
END EXIT ............... : ............
END COMMAND ........... : ............


RECEPTION :
START EXIT ............ : ............
START COMMAND ......... : ............
END EXIT ............... : ............
END COMMAND ........... : ............


DO YOU WANT TO GO ON ?
-ENTER- NEXT FIELD               -F3- CANCEL                 -F8- COMPLETION
```

5. Enter information as described in the following table.

Fields	Length	Description
SYMBOLIC NAME	1 to 8 characters	This name identifies the File and must be unique. The symbolic name is negotiated with the Partners involved in a transfer. You can create a general default profile, in the files directory, called $$FILE$$, to manage transfers of symbolic files not defined in the files directory. This is related to the partners default file, for FTP and PeSIT. It is related to the default SYSIN file as well, for the FTP protocol only. If the symbolic file “UnkFile” is not defined in the directory, transferring this file with a peSIT partner is possible provided that : 1. A default file is defined for the current partner, this file exists in the directory, and is enabled. 2. No default file exists for the current partner, but the $$FILE$$ general profile exists in the files directory, and is enabled. With a FTP partner, the process is: 1. A default file is defined for the current partner, this file exists in the files directory, and is enabled. 2. No default file exists for the current partner, but a default profile is defined in the SYSIN, and exists in the files directory, and is enabled. 2. No default file is defined for the current partner, and no default file is defined in the SYSIN, but the $$FILE$$ general profile exists in the files directory, and is enabled. **Note**: $$...$$ syntax is reserved, and a file symbolic name cannot be equal to character ‘*’ .
INITIALIZATION STATUS	1 alphabetic character	E = Enabled. File can be transferred H = Held. File cannot be transferred.
DIRECTION	1 alphabetic character	This field defines the transfer direction authorized for this File. T = Transmission R = Reception * = Both directions
(INQUIRY)	Y/N	If this field is set to Yes, it enables direct Inquiry from a remote partner, and downloading the file doesn’t require a hold request to be prepared. If this option is set to Y
RECEIVING PARTNER	1 to 8 characters	This is the symbolic name of the Partner receiving the file, or you can enter the $$ALL$$ keyword to authorize all Partners to receive this file. This field is used when the direction is T or *. When the direction is R, this field is ignored. If you enter a Partner list in this field, one request is sent for each partner on the list, up to 17 partners. When a partner calls to receive a file, the monitor verifies that the partner is in the list. The name of a list must begin with the # character, and this physical file must reside in the config directory. See *Implementing a Partner List* on page D-2 for more information.

Fields	Length	Description
TRANSMITTING PARTNER	1 to 8 characters	This is the symbolic name of the transmitting Partner, or you can enter the $$ALL$$ keyword to authorize all Partners to transmit this file. This field is used when the direction is R or *. When the direction is T, this field is ignored. If you enter a Partner list in this field, one request is sent for each partner on the list, up to 17 partners. When a partner calls to transmit a file, the monitor verifies that the partner is in the list. The name of a list must begin with the # character, and this physical file must reside in the config directory. See *Implementing a Partner List* on page D-2 for more information.
PRIORITY	1 numeric character	The following values are valid: 0 = Urgent 1 = Fast 2 = Normal
DEFINITION TYPE	1 alphabetic character	D = Dynamic. The physical name can be supplied at request time, and replaces the name in the directory. F = Fixed. The physical name is always the one in the directory.
PRESENTATION TABLE	1 numeric character	(1 - 9) This table provides presentation parameters used in transfer mode for this file. See *Updating Presentation Tables* in this chapter for more information. Depending on the PPFRST parameter of the SYSIN file, the partner presentation table (PPFRST=P) or the file presentation table (PPFRST=F) has priority. If no presentation table is provided in the partner and the file definitions, the default defined in the SYSIN file is used.
PARAMETER CARDS FILE	Yes/No	This field enables you to use the FICPARAMS.dat file in the config directory. See *Implementing a Parameters Card File* on page D-5.
SPACE TO RESERVE	Yes/No	This fields enables the monitor to reserve file space when it opens or creates a received file. CAUTION: If you set this field to Y, it can overload the monitor.
ALLOCATION RULE	1 numeric character	This value determines how Sterling Connect:Express receives a file. 0 = Indifferent. If the file exists, it is opened and truncated to a length of 0. If the file does not exist, it is created. (Default) 1 = Preallocated. The file must exist. It is opened and truncated to length 0. 2 = Sterling Connect:Express creates the file and it cannot exist.
RECORD FORMAT	2 alphabetic characters	The following values are valid: TF = Text Fixed format TV = Text Variable format BF = Binary Fixed format BU = Binary Undefined format T*, B*, ** = The record format is determined by the sender parameters of from the transfer request. A Text file is a file in which records end with the line feed character, LF or 0A in hexadecimal (or 0D or 0D0A). This character is not interpreted for binary files. A Fixed format file means that all records have the same length. The LF character is suppressed when transmitting text files, and appended when receiving them. The transfer request or a user exit can override this field. If the file can be transferred in both directions, this field can be T*, B* or **, and then the record format is required with the transfer request.

Fields	Length	Description
RECORD LENGTH	5 numeric characters	This field contains the file record length. In transmitting mode, this value cannot be null, except for the BU, TV, T*, B*, and ** record formats which have a default value of 1024 bytes. If this value is not zero in receiving mode, this value is checked with the record length transmitted by the remote partner, if the protocol supports this function. Otherwise, Sterling Connect:Express uses the value received from the remote partner. The transfer request or a user exit can override this field. If the file can be transferred in both directions, this field can be zero, and then the record length is required with the transfer request.
REMOTE DSN	1 to 44 characters	FTP: This field contains the remote physical name of the transferred file. This name must match the physical name rules on the remote operating system. PeSIT: This field is used with Partners of Type other. This field is sent in the Pi99 protocol parameter.
TYPE/STRUCTURE/MODE FTP	3 characters	This field contains the type, structure, and mode parameters for the FTP protocol. Type: Ascii, Ebcdic, Binary, *(unchanged) Structure: File, Record, *(unchanged) Mode: Block, Stream, *(unchanged)
STORE UNIQUE (FTP)	1 character (Y / N)	This field holds the Store Unique option for the FTP protocol. If set, all files are sent and stored on the server. If the file already exists with the same physical name, the FTP server generates a new physical name.
FA	1 character (Y / N)	This field contains the File Agent flag. This flag is effective when using the Connect:Enterprise routing capabilities of the file agent.
NOT	1 numeric character	0 = no notification 1 = Notification sent at beginning of transfer 2 = Notification sent at end of transfer 3 = Notification sent at beginning and end of transfer 4 = Notification sent at error 5 = Notification sent at beginning of transfer or erro 6 = Notification sent at end of transfer or error 7 = Notification sent at beginning and end of transfer or error
PHYSICAL NAME	1 to 127 characters	This field contains the physical name of the file transferred by Sterling Connect:Express. This is an absolute physical name. The physical name can contain environment variables with the $ character. In this case, the limit of the physical translation length becomes the system limit. You can use keywords or variables to build the string at request time. If this field is left blank and the file is received then the physical file name will be &REQNUMB. See *Implementing Physical Names* on page D-3.
START EXIT (RECEPTION)	1 to 12 characters	This field contains the name of an executable user program created by the user in the exit directory. STRF runs the procedure before starting the receiving process.
START COMMAND (TRANSMISSION)	1 to 12 characters	This field contains the name of a Shell user program created by the user in the EXIT directory. GTRF runs the procedure before starting the transmission.
END EXIT (TRANSMISSION)	1 to 12 characters	This field contains the name of an executable user program created by the user in the EXIT directory. The procedure is run by STRF at the end of transmission.

Fields	Length	Description
END COMMAND (TRANSMISSION)	1 to 12 characters	This field contains the name of a Shell user program created by the user in the EXIT directory. GTRF runs the procedure at the end of transmission.
START EXIT (RECEPTION)	1 to 12 characters	This field contains the name of an executable user program created by the user in the EXIT directory. STRF runs the procedure starting the receiving process.
START COMMAND (RECEPTION)	1 to 12 characters	This field contains the name of a Shell user program created by the user in the EXIT directory. GTRF runs the procedure before starting the receiving process.
END EXIT (RECEPTION)	1 to 12 characters	This field contains the name of an executable user program created by the user in the EXIT directory. STRF runs the procedure at the end of reception.
END COMMAND (RECEPTION)	1 to 12 characters	This field contains the name of a Shell user program created by the user in the EXIT directory. GTRF runs the procedure at the end of reception.

6. Press <F3> to exit from this screen. The cursor moves to the field **Do you want to go on**. You can press <F8> to move to this field without pressing <Enter> through each field.
7. Type **Y** or press <Enter> to confirm your entries.

Using the Record Length and Record Format Fields

This section shows two ways to configure the Record Length and Record Format fields. For example, a file contains 162 bytes with the following structure:

```
|.......80..bytes..................|0X0A|.......80.. bytes..................|0X0A|
```

If you use a Text Fixed format, Sterling Connect:Express sends 2 records with 80 bytes (without 0X0A delimiter), as shown below.

Format: Text Fixed Format Field: TF

Length: 80 Record Length Field: 00080

```
|.......80..bytes..................|
|.......80..bytes..................|
```

If you use a Binary Fixed format, Sterling Connect:Express sends 2 records with 81 bytes (with 0X0A delimiter), as shown in the following example.

Format: Binary Fixed Format Field: BF

Length: 81 Record Length Field: 00081

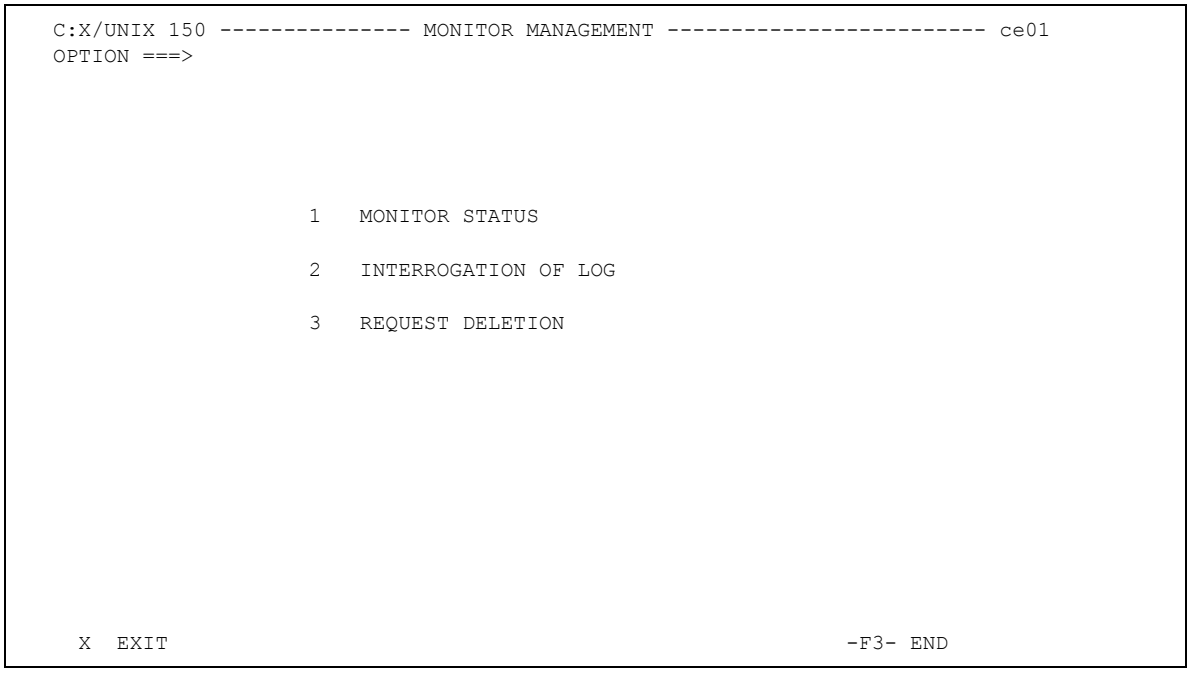
```
|.......80..bytes..................|0X0A|
|.......80..bytes..................|0X0A|
```

With the Text Fixed format, Sterling Connect:Express suppresses the 0X0A delimiters at the end of each record in transmitting mode and appends them in receiving mode.

Monitor Management

From the Monitor Management screen, you can list or update requests in the requests directory (RENC), display a log of initialization information and current monitor activity, and delete a request.

To access the Monitor Management screen, type 2 in the Option field on the Main Menu and press <Enter>.



The following table describes the options on the Monitor Management screen.

Option	Description
Monitor Status	Displays request activity. You enter request criteria on the first Monitor Status screen, and Sterling Connect:Express displays the request records that match your criteria.
Interrogation of Log	Displays Sterling Connect:Express initialization information and current monitor activity. You can display the last page of the log, the first page of the log, or search for a text string in the log.
Request Deletion	Enables you to delete a group of transfer requests.

Monitor Status

You can display a list of transfer requests that meet your criteria using the Monitor Status option. When you display a list of requests, you can interrupt, restart, purge, or display a specific request.

Note: The response time of this function depends on the size of the RENC file. You should purge this file regularly for optimal system performance.

Viewing Transfer Requests

1. From the Main Menu, type **2** in the Option field and press <Enter> to display the Monitor Management screen.
2. Type **1** in the Option field and press <Enter> to display the Monitor Status screen.

```
C:X/UNIX 150 ----------------- MONITOR STATUS -------------------------- ce01
OPTION ===>


          REQUEST .... ---> *              (*, REQUEST NUMBER)

          FILE ....... ---> *              (*, SYMBOLIC NAME)

          PARTNER .... ---> *              (*, SYMBOLIC NAME)

          DIRECTION .. ---> *              (*, T, R)

          STATE ...... ---> *              (*, TRANSFER STATE)

          DATE ....... ---> *              (*, TRANSFER DATE)




DO YOU WANT TO GO ON ?

-ENTER- NEXT FIELD                -F3- CANCEL                  -F8- COMPLETION
```

3. Enter your criteria to select the transfer requests that you want to display. The following table describes each field.

Field	Length	Description
Request	8 numeric characters	Enter a request number to display a specific request. The request number uniquely identifies a transfer. Use the * character to display all request numbers.
File	1 to 8 characters	Enter a symbolic file name to display all transfer requests for that file. Use the * character to display requests for all symbolic file names.
Partner	1 to 8 characters	Enter the symbolic name of the Partner to display all transfer requests for that Partner. Use the * character to display requests for all Partners.
Direction	1 alphabetic character	Enter a transfer direction to display all transfer requests for that direction. The following values are valid: T = Transfer requests in transmission. R = Transfer requests in reception. * = Transfers in both directions.

Field	Length	Description
State	1 alphabetic character	Enter a transfer status to display all requests with a specific status. The following values are valid: A = Search for requests awaiting selection. D = Search for deferred transfers. E = Search for ended transfers. H = Search for hold requests. J = Search for transfers that are waiting to restart. K = Search for restarting requests. O = Search for interrupted requests. C = Search for transfers in progress. X = Seach for transfers acknowledged * = Search for all transfer requests.
Date	1 to 14 numeric characters	Enter the date after which you want to display transfer requests. Sterling Connect:Express displays all transfers that occurred after the specified date and time. Enter the date and time in the format, yyyymmddhhmmss. Use the * character to display transfer requests for all dates.

Sterling Connect:Express displays the requests that meet your criteria. 18 records display at one time and information displays on 3 horizontal screens as shown on the following screens. Press <F10> and <F11> to scroll horizontally. Use <CR> to scroll down, and <BACKSPACE> to scroll up.

```
C:X/UNIX 150 ----------------- MONITOR STATUS --------------------------- ce01
OPTION ===>
    REQ.NUM.  FILE      WITH      DIR. PRI. REQ. TYPE    STATE   STRF ID
    07200001  FICTEST1  EXPRESS1  T    0    N NORMAL     O       0000010408
    07200003  FICTEST1  EXPRESS1  T    0    N NORMAL     C       0000004526
    07200005  FICTEST2  DPX1      T    0    N NORMAL     E       0000011441
    07200006  FICTST    SID1      R    0    N NORMAL     E       0000011698
    07200007  FICTEST2  DPX1      T    0    N NORMAL     E       0000011443
    07200008  DOUDOU    SID1      R    0    N NORMAL     E       0000011700
    07200009  FICSTSN   DPX1      T    0    N NORMAL     E       0000011445
    07200010  FIC22424  SID1      R    0    N NORMAL     E       0000011702
    07200011  FICTEST3  DPX1      T    0    N NORMAL     E       0000011447
    07200012  ARECEVOI  SID1      R    0    N NORMAL     E       0000011704
    07200013  FICTEST3  DPX1      T    0    N NORMAL     E       0000011449
    07200014  ARECEVOI  SID1      R    0    N NORMAL     E       0000011706
    07200015  AENVOYER  DPX1      T    0    N NORMAL     O       0000011451
    07200017  FICTEST4  DPX1      T    0    N NORMAL     E       0000011456
    07200018  FICTST2   SID1      R    0    N NORMAL     E       0000012225
    07200019  FICTEST1  EXPRESS1  T    0    N NORMAL     J       0000011458


    <- -F10-    -F3- END -F7- PREVIOUS SCREEN   -F8- NEXT SCREEN    -F11- ->
```

```
C:X/UNIX 150 ----------------- MONITOR STATUS --------------------------- ce01
OPTION ===>
    REQ.NUM.  TYP.REQ  REQUESTER  EXTERN.NUM.  TYPE-PART.  TRC   PRC   RETRIES
    07200001  STERM    pla        07200001     TOM         2077  3304  0
    07200003  STERM    pla        07200003     TOM         0000  0000  1
    07200005  STERM    pla        07200005     TOM         0000  0000  0
    07200006  STRF     SID1       07200005     TOM         0000  0000  0
    07200007  STERM    pla        07200007     TOM         0000  0000  0
    07200008  STRF     SID1       07200007     TOM         0000  0000  0
    07200009  STERM    pla        07200009     TOM         0000  0000  0
    07200010  STRF     SID1       07200009     TOM         0000  0000  0
    07200011  STERM    pla        07200011     TOM         0000  0000  0
    07200012  STRF     SID1       07200011     TOM         0000  0000  0
    07200013  STERM    pla        07200013     TOM         0000  0000  0
    07200014  STRF     SID1       07200013     TOM         0000  0000  0
    07200015  STERM    pla        07200015     TOM         4091  2299  0
    07200017  STERM    pla        07200017     TOM         0000  0000  0
    07200018  STRF     SID1       07200017     TOM         0000  0000  0
    07200019  STERM    pla        07200019     TOM         2076  3304  0


    <- -F10-    -F3- END -F7- PREVIOUS SCREEN   -F8- NEXT SCREEN    -F11- ->
```

```
C:X/UNIX 150 ----------------- MONITOR STATUS --------------------------- ce01
OPTION ===>
    REQ.NUM.  K.BYTES  RECORDS     PHYSICAL NAME
    07200001  0000000  0000000000  /home/pla/.profile
    07200003  0000000  0000000000  /home/pla/.profile
    07200005  0000001  0000000024  /home/pla/.profile
    07200006  0000001  0000000024  /tmp/TOM_A7200006.tmp
    07200007  0000001  0000000024  /home/pla/.profile
    07200008  0000001  0000000024  /tmp/TOM_A7200008.tmp
    07200009  0000001  0000000024  /home/pla/.profile
    07200010  0000001  0000000024  /tmp/TOM_A7200010.tmp
    07200011  0000001  0000000024  /home/pla/.profile
    07200012  0000001  0000000024  /tmp/TOM_A7200012.tmp
    07200013  0000001  0000000024  /home/pla/.profile
    07200014  0000001  0000000024  /tmp/TOM_A7200014.tmp
    07200015  0000000  0000000000  /home/pla/.profile
    07200017  0000001  0000000024  /home/pla/.profile
    07200018  0000001  0000000024  /tmp/TOM_A7200018.tmp
    07200019  0000000  0000000000  /home/pla/.profile



    <- -F10-    -F3- END -F7- PREVIOUS SCREEN   -F8- NEXT SCREEN    -F11- ->
```

4. Enter any of the following commands next to a request number and press <Enter>.

Command	Description
I	Interrupts the request. Only transfer requests with a status of C can be interrupted. These requests are in progress.
R	Restarts a request that was interrupted. Only transfer requests with a status of O can be restarted. These requests are interrupted.
P	Purges the request.
S	Displays the status of the request.

If you type an **S** to display the status of a request, the following screen is displayed.

```
C:X/UNIX 150 ----------------- MONITOR STATUS ------------------------------ ce01

  REQUEST : 02100018     FROM : PLA       DIRECTION : R  WITH : DPX1      **FILE**
  ORIGIN  : user1        DESTINATION : user2          XFER ID : X:7D0009 08192009
  SENDER  :                                RECEIVER  :
  USERID  : pla          STRF PID    : 0000008586  FA : N    NOT: 0       SSL: 0
  FILE    : ARECEVOI     CMD ORIGIN  : T    LPARM :           TRANSFER STATE : E
  PROTOCOL: PESIT        CRC : -    MULTI : -   TRANSLATION : 0  COMPRESSION : -
  DSNAME  : /home/tmp/ARECEVOI_D980723H124613

  MESSAGE >> :


  BEGIN : 19980723 12:46:13        END : 19980723 12:46:14       RETRIES/MAX 00/00
  NRC : 0000        SRC : 0000   TRC : 0 000     PRC : 0 000     SSLRC : 00000000
  NUMBER OF RECORDS XFERED : 0000000679      FILE BYTES    :     0000025
  RECORD FORMAT ......... : TV               RECORD LENGTH ......... : 00255
  ********** TCPIP **********
  TCPIP HOST (00) :
  TCPIP ADDRESS (12) : 172.17.16.80
  PORT : 01267
```

Interrupting a Transfer Request

1. From the Main Menu, type **2** in the Option field and press <Enter> to display the Monitor Management screen.
2. Type **1** in the Option field and press <Enter> to display the Monitor Status screen.
3. Enter criteria to display the transfer request that you want to interrupt.
4. Type **I** next to the transfer request and press <Enter>. You can only interrupt a request with a status of C in progress.
5. Press <F3> to exit the Monitor Status screen.

Restarting a Transfer Request

1. From the Main Menu, type **2** in the Option field and press <Enter> to display the Monitor Management screen.
2. Type **1** in the Option field and press <Enter> to display the Monitor Status screen.

3. Enter criteria to display the transfer request that you want to restart.
4. Type **R** next to the transfer request and press <Enter>. You can only restart a request with a status of O, or interrupted.
5. Press <F3> to exit the Monitor Status screen.

Purging a Transfer Request

1. From the Main Menu, type **2** in the Option field and press <Enter> to display the Monitor Management screen.
2. Type **1** in the Option field and press <Enter> to display the Monitor Status screen.
3. Enter criteria to display the transfer request that you want to purge.
4. Type **P** next to the transfer request and press <Enter>.
5. Press <F3> to exit the Monitor Status screen.

Displaying the Status of a Transfer Request

1. From the Main Menu, type **2** in the Option field and press <Enter> to display the Monitor Management screen.
2. Type **1** in the Option field and press <Enter> to display the Monitor Status screen.
3. Enter criteria to display the transfer request that you want to display.
4. Type **S** next to the transfer request and press <Enter>.
5. Press <F3> to exit the Monitor Status screen.

Interrogation of Log

The Sterling Connect:Express log file contains Sterling Connect:Express initialization information and current monitor activity. You can use the Interrogation of Log option from the Monitor Management screen to view this information. Complete the following steps to display the log file.

1. From the Main Menu, type **2** in the Option field and press <Enter> to display the Monitor Management screen.
2. Type **2** in the Option field and press <Enter> to display the Interrogation of Log screen. By default, Sterling Connect:Express displays the last 19 records.

```
C:X/UNIX 150 -------------- INTERROGATION OF LOG ------------------------ ce01
OPTION ===>                                          Fri Mar 13 15:03:48 1998
98/03/13 15:03:36 REQUEST 07200018 FICTST   TRANSFER ACCEPTED    STRF 0000012225
98/03/13 15:03:36 REQUEST 07200017 FICTST   TRANSFER ACCEPTED    STRF 0000011456
98/03/13 15:03:36 REQUEST 07200018 FICTST   TRANSFER STARTED     STRF 0000012225
98/03/13 15:03:36 REQUEST 07200018 (R) /tmp/TOM_A7200018.tmp
98/03/13 15:03:36 REQUEST 07200017 FICTST   TRANSFER STARTED     STRF 0000011456
98/03/13 15:03:36 REQUEST 07200017 (T) /home/pla/.profile
98/03/13 15:03:36 REQUEST 07200018 FICTST   TRANSFER ENDED       STRF 0000012225
98/03/13 15:03:36 REQUEST 07200018 RECEIVING     <- SID1    , FILE FICTST   NUMB
98/03/13 15:03:36 REQUEST 07200017 FICTST   TRANSFER ENDED       STRF 0000011456
98/03/13 15:03:36 REQUEST 07200017 TRANSMITTING  -> DPX1    , FILE FICTST   NUMB
98/03/13 15:03:36 COMMUNICATION CLOSED (O) WITH: DPX1
98/03/13 15:03:36 COMMUNICATION CLOSED (I) WITH: SID1
98/03/13 15:03:43 REQUEST 07200019 FICTST   EXPRESS1          SRC=0000 TRC=0000L
98/03/13 15:03:43 REQUEST 07200019 <- pla      ACCEPTED    (N)
98/03/13 15:03:43 REQUEST 07200020          SID1             SRC=0000 TRC=2052L
98/03/13 15:03:43 REQUEST 07200019 - NEW LINK : X25
98/03/13 15:03:43 REQUEST 07200019 REJECTED <- EXPRESS1      SRC=0000 TRC=2076L
98/03/13 15:03:43 COMMUNICATION NOT OBTAINED ->  EXPRESS1 REQ: 07200019       RE
98/03/13 15:03:43 REQUEST INCOMING SESSION ERROR RECEIVE     NRC=0000
98/03/13 15:03:43 REQUEST INCOMING REJECTED <- STRF          SRC=0000 TRC=2052L

    <- -F10-    -F3- END -F7- PREVIOUS SCREEN   -F8- NEXT SCREEN    -F11- ->
```

3. Use any of the following commands to navigate in the log file:
   - Press <F7> to go to the previous page and <F8> to go to the next page. Press <Enter> to refresh the screen and display the last 19 records.
   - Press <F10> to shift right and <F11> to shift left.
   - Type **M** in the Option field and press <F7> to go to the first page of the log. Type **M** in the Option field and press <F8> to go to the last page of the log.
   - To search for a text string, type **FIND <word1> <word2>**... in the Option field and press <Enter>. Sterling Connect:Express searches for the text string from the beginning of the log file. Press <F1> to search for the next occurrence of the text string.
4. Press <F3> to exit the Interrogation of Log file.

Note: For explanations of the fields in the log file, see *Appendix A Sterling Connect:Express Messages*.

Request Deletion

The Request Deletion option enables you to delete a group of transfer requests that meet your criteria. For example, you can delete all requests for a specific Partner.

Deleting a Group of Transfer Requests

1. From the Main Menu, type **2** in the Option field and press <Enter> to display the Monitor Management screen.
2. Type **3** in the Option field and press <Enter> to display the Request Deletion screen.

```
C:X/UNIX 150 --------------- REQUEST DELETION --------------------------- ce01
OPTION ===>


          REQUEST .... ---> *              (*, REQUEST NUMBER)

          FILE ....... ---> *              (*, SYMBOLIC NAME)

          PARTNER .... ---> *              (*, SYMBOLIC NAME)

          DIRECTION .. ---> *              (*, T, R)

          STATE ...... ---> *              (*, TRANSFER STATE)

          DATE ....... ---> *              (*, TRANSFER DATE)




DO YOU WANT TO GO ON ?

-ENTER- NEXT FIELD                -F3- CANCEL                  -F8- COMPLETION
```

3. Enter criteria to select the transfer requests that you want to delete. The following table describes each field.

Field	Length	Description
Request	8 numeric characters	Enter a request number to delete a specific request. The request number uniquely identifies a transfer. Use the * character to search all request numbers.
File	1 to 8 characters	Enter a symbolic file name to delete all transfer requests for that file. Use the * character to search requests for all symbolic file names.
Partner	1 to 8 characters	Enter the symbolic name of the Partner to delete all transfer requests for that Partner. Use the * character to search requests for all Partners.
Direction	1 alphabetic character	Enter a transfer direction to delete all transfer requests for that direction. The following values are valid: T = Transfer requests in transmission. R = Transfer requests in reception. * = Transfers in both directions.

Field	Length	Description
State	1 alphabetic character	Enter a transfer status to delete all requests with a specific status. The following values are valid: A = Search for requests awaiting selection. D = Search for deferred transfers. E = Search for ended transfers. H = Search for hold requests. J = Search for transfers that are waiting to restart. K = Search for restarting requests. O = Search for interrupted requests. C = Search for transfers in progress. X = Seach for transfers acknowledged * = Search for all transfer requests.
Date	1 to 14 numeric characters	Enter the date after which you want to delete transfer requests. Sterling Connect:Express deletes all transfers that occurred after the specified date and time. Enter the date and time in the format, yyyymmddhhmmss. Use the * character to search for all transfer requests.

Caution: You cannot submit two deletion requests at the same time.

If a transfer request is sent to the monitor during the request deletion process, the monitor interrupts deletion processing to process the transfer request, and then returns to the deletion process.

4. Press <F3> to exit from this screen.
5. Type **Y** and press <Enter> to confirm your request.

Tables Management

The Tables option on the Main Menu enables you to update session and presentation tables. Session tables describe the session environment with a remote Partner, and Presentation tables describe how data is presented to a remote Partner.

Updating Session Tables

Session tables describe the session environment with a remote Partner. These tables contain parameters from the RTAB file in the CONFIG directory. RTAB is initialized during installation and must be updated. This is not used with the ETEBAC3 and FTP protocols.

1. From the Main Menu, type **3** in the Option field and press <Enter> to display the Tables Management screen.

```
C:X/UNIX 150 --------------- TABLES MANAGEMENT -------------------------- ce01
OPTION ===> S2




                    S.  SESSION TABLE

                    P.  PRESENTATION TABLE









  X  EXIT                                                  -F3- END
```

2. Type **S** and the table number in the Option field and press <Enter> to select a session table. The table number is a number between 1 and 9 that identifies a specific session table.

```
C:X/UNIX 150 ------------------- SESSION S2 ----------------------------- ce01
OPTION ===>

            U    UPDATE


LINE MESSAGE SIZE ..... : 04096          BYTES < 65536

SYNCHRONIZATION ....... : 32             K. BYTES

WINDOW ................ : 16             0 - 16

LEVEL ................. : 1              PROTOCOL VERSION NUMBER

NUMBER OF RETRIES...... : 05             0 - 99

CRC ................... : Y              Y/N

MAJ : 97/09/16 10:55 pla
-ENTER- NEXT FIELD               -F3- CANCEL                -F8- COMPLETION
```

3. Type **U** in the Option field and press <Enter> to update the table.

4. Enter information as described in the following table. Line 22 displays the date and time of the last update or the date and time that the table was created.

Fields	Length	Description
Line Message Size	1 to 5 numeric characters	This parameter must be less than 65 kbytes and match the network buffers size. This field represents a value negotiated between two PeSIT partners at the beginning of a session. The message size is negotiated to the smaller size, and this applies to any network.
Synchronization	2 numeric characters	This field specifies the number of Kbytes transferred before recording a checkpoint record. “0” indicates that the synchronization facility is not used. During synchronization, Sterling Connect:Express creates a checkpoint record to restart the transfer if it is interrupted. 0 does not allow the transfer to be restarted.
Window	1 to 2 numeric characters	This field indicates the size of the synchronization window or the number of synchronization points transmitted by the sender before the receiver acknowledges it. Valid values are 0 to 16. A value of 0 means that no acknowledgment is sent.
Level	1 numeric character	This field indicates the version level of the protocol. 1 for PeSIT release D 2 for PeSIT release E
Number of Retries	2 numeric characters	This field indicates the number of authorized restarts or connection retries. It can be a value from 0 to 99. If you enter a value of 0, the Partner is disabled after the first call if the call packet is not accepted.
CRC	1 alphabetic character	(Y or N) This field indicates if CRC is used. It is only available with PeSIT level 2.

5. Press <F3> to exit from this screen.
6. Type **Y** or press <Enter> to confirm your entries.

Updating Presentation Tables

Presentation tables are used to describe how to present data to the remote partner. These tables contain parameters from the RPRE file in the CONFIG directory. RPRE.DAT is initialized at installation and must be updated by the user.

1. From the Main Menu, type **3** in the Option field and press <Enter> to display the Tables Management screen.
2. Type **P** and the table number in the Option field and press <Enter> to display a Presentation table screen. The table number is a number between 1 and 9 that identifies a specific presentation table.

You can associate a presentation table number to a partner or a file, or indicate a default presentation table in the SYSIN file of the monitor (PPDFLT=n).

Depending on the PPFRST parameter of the SYSIN file, the partner presentation table (PPFRST=P) or the file presentation table (PPFRST=F) has priority. If no presentation table is provided in the partner and the file definitions, the default defined in the SYSIN file is used.

```
C:X/UNIX 150 ---------------- PRESENTATION P5 -------------------------- ce01
OPTION ===>



            U    UPDATE




COMPRESSION ........... : 3               0/1/2/3

MULTIARTICLE .......... : N               Y/N

TRANSLATION TABLE ..... : 1               0 -> 9









                                          MAJ : 98/01/27 13:00 pla
-ENTER- NEXT FIELD                -F3- CANCEL                  -F8- COMPLETION
```

3. Type **U** in the Option field and press <Enter> to update the table.
4. Enter information as described in the following table. Line 22 displays the date and time of the last update or the date and time that the table was created.

Field	Length	Description
Compression	1 numeric character	0 = No compression. 1 = Horizontal compression. 2 = Vertical compression. 3 = Mixed compression (horiz. and vertical). Compression is negotiated between both Partners when selecting the File. Tests must be carried out according to the type of data transferred. (PeSIT protocol only)
Multiarticle	1 alphabetic character	(Y or N) This field indicates if a multi-article is used with this Partner. For best results, set this option to Yes. (PeSIT protocol only)
Translation Table	1 numeric character	(0 to 9) This field indicates which table TBL0x.DAT is used for ASCII-EBCDIC translation. 0 indicates no translation. See *Translating Data* on page D-4 for more information.

5. Press <F3> to exit from this screen.
6. Type **Y** or press <Enter> to confirm your entries.

Chapter 4

Transfers

This chapter provides information about how transfer requests are processed and explains how to initiate transfer requests.

Overview

When the monitor process GTRF receives a transfer request, it processes the following controls:

Control	Processed
Syntax	Verifies that all necessary parameters are present: • The symbolic File name, a mandatory request parameter. • The symbolic Partner name, given by the request or derived from the File definition. • The direction of transfer, given by the request or derived from the File definition. • The dynamic data set name, given by the request or derived from the File definition. • The priority, given by the request or derived from the File definition. • The link type, given by the request or derived from the Partner definition. • The request type • The date and time
Logic	Checks that parameters are coherent, and validates the data flow to ensure that the Partner is authorized for the file and transfer direction. • Partner is recognized • File is recognized. • Direction is valid • Direction and Request Type are compatible
Network	Verifies the following network parameters: • Number of links customized with this partner. • Network activated.

Note: Default values are supported.

The Transfer Process

When a request is accepted, it is given an 8-digit request number, and a record is created in the RENC file. This request number is unique and has the format *qqqnnnnn*. qqq is the Julian date (modulo 183), and nnnnn is a sequential number from 1 to 61440.

If no session exists with the Partner, GTRF will open one and an STRF process will run detached.

If one or more sessions exist with the Partner and the maximum allowed number of sessions has not been reached, GTRF opens a new session. If the maximum has been reached, GTRF queues the request, then selects it as soon as one of the active transfers is finished.

The process is independent of the transfer direction. GTRF can open a session in both directions and STRF can execute both transmissions and receptions in the same session.

Types of Transfer Requests

A file transfer is always initiated by a command called a request. There are five types of transfer requests, as listed below. The first four types are discussed in this section. The fith type is an external request.

Request Type	Description
Operator Transfer Requests	Online requests entered using the STERM operator interface.
Batch Transfer Requests	Requests entered with the P1B8PREQ utility that works with the LOB8Z20 application interface.
Batch end to end Requests	Requests entered with the P1B8PE2E utility that works with the LOB8Z20 application interface.
Application Program Transfer Requests Using L0B8Z20	Requests entered using the L0B8Z20 application interface. These transfers are discussed in Chapter 5 *Application Interface*.
External Transfer Requests	Transfer requests initiated by any software, including Sterling Connect:Express, that use the PeSIT, or FTP protocol.

Operator Transfer Requests

You can execute transfer requests using the menus of the STERM operator interface. From the Main Menu, type **4** in the Option field and press <Enter>. The Transfer Request screen is displayed.

```
C:X/UNIX 150 -------------- TRANSFER REQUEST ------------------------ ce01
OPTION ===>
FILE ................. : SYMBFILE        DIRECTION ............. : T (T/R)
PARTNER ............... : PARTNER.
DPCSID ALIAS .......... : MYNAME..        DPCPSW ALIAS .......... : MYPASSW.
ORIGIN................. : USER1...        DESTINATION............ : USER2
SENDER................. :                 RECEIVER .............. :
PHYSICAL NAME ......... : /usr/lib/libsock.a.........................

USER DATA ............. : PHYSICAL.FILENAME.TRANSMIT..................
LABEL:.....................................................................

RECORD FORMAT ......... : BU              TF, TV, BF, BU
RECORD LENGTH ......... : 02048
TYPE/STRUCTURE/MODE FTP : ***             E/A/I/*,F/R/*,B/S/*
STORE UNIQUE (FTP) .... : N               Y/N   FA :  O/N   NOT: (0-7)
TYPE .................. : N               (N/I/H/M)
TYPE OF CONNECTION .... : T               (X/P/T)
PRIORITY .............. : 0               (0/1/2)
DATE .................. : 19980728101604  (YYYYMMDDHHMMSS)

DO YOU WANT TO GO ON ?
-ENTER- NEXT FIELD                -F3- CANCEL                  -F8- COMPLETION
```

Each field identifies one characteristic of the user's request for a transfer. The fields are described in the following table.

Field	Length	Description
File	1 to 8 characters	Required. This field contains the symbolic name of the File to be transferred. This name must exist in the files directory (RFIC).
Partner	1 to 8 characters	Optional. The symbolic name of the Partner with whom you are transferring. This name must exist in the Partners directory. The name of the Partner must correspond with the (sender-receiver) definition in the files directory. You can also enter the name of a Partner list. See *Implementing a Partner List* on page D-2.
DPCSID ALIAS	1 to 8 characters	Optional. This name overwrites the current DPCSID in the SYSIN file or RPAR partners directory file. The remote host must know your Local Host ID as DPCSID name.
DPCPSW ALIAS	1 to 8 characters	Optional. This name overwrites the current DPCPSW in the SYSIN file or RPAR partners directory file. The remote host must know your Local Host password as DPCPSW name.
Origin	1 to 8 characters	Optional. Transfer origin.
Destination	1 to 8 characters	Optional. Transfer destination.
Sender	1 to 24 characters	Optional. File sender.
Receiver	1 to 24 characters	Optional. File receiver.

Field	Length	Description
Physical Name	1 to 128 characters	Optional. This is the absolute physical name of the file to be transferred. You can use environment variables but ensure that the monitor knows these variables when it starts. If this field is not filled in, the physical name is taken from the files directory RFIC. If the File is described with a fixed definition type, leave this field blank.
User Data	1 to 44 characters	Optional. This field is used in different ways depending on the protocol. FTP: This field contains the remote physical name of the transferred file. This name must match the physical name rules on the remote operating system. PeSIT, partner type TOM: This is the physical name (IBM format) sent by the transmitter through the PeSIT protocol. By default, the physical file is converted to IBM format, and directories are separated by a dot. PeSIT, partner type other: This field is sent in the Pi99 protocol field. This field is similar to the User Data Field in p1b8preq.
Label	1 to 80 characters	This is the pi37 identifier of the PeSIT protocol. This field is similar to the Label field in p1b8preq.
Record Format	2 alphabetic characters	TF = Text Fixed format. TV = Text Variable format. BF = Binary Fixed format. BU = Binary Undefined format. A Text file is a file which has records that end with a line feed character, LF or 0A in hexadecimal. This character is not interpreted for binary files. A Fixed format file means that all records have the same length. The LF is suppressed when transmitting the text file and appended when receiving. The Record Format field can be overwritten by an optional user exit.
Record Length	5 numeric characters	This field contains the file record length. In transmitting mode, the value cannot be null, except for a BU record format which has a default value of 1024 bytes. If the value is not zero in receiving mode, it is checked with the record length transmitted by the remote partner if the protocol supports this function. Otherwise, the value received from the remote partner is used.
Type/Structure/Mode FTP	3 characters	This field contains the type, structure, and mode parameters for the FTP protocol. Type: Ascii, Ebcdic, Binary, * (unchanged) Structure: File, Record, * (unchanged) Mode: Block, Stream, * (unchanged)
Store Unique (FTP)	1 character (Y / N)	This field contains the Store Unique option for the FTP protocol. If set, all files are sent and stored on the server. If the file already exists with the same physical name, the FTP server generates a new physical name.
FA	1 character (Y / N)	This field contains the File Agent flag. This flag is effective when using the Connect:Enterprise routing capabilities of the file agent.
NOT	1 numeric character (0/1/2/3)	This field contains the notification flag. 0 = No notification 1 = Notification at beginning of transfer 2 = Notification at end of transfer 3 = Notification at beginning and end of transfer 4 = Notification at error 5 = Notification at beginning of transfer or error 6 = Notification at end of transfer or error 7 = Notification at beginning and end of transfer or error

Field	Length	Description
Type	1 alphabetic character	Optional. The type of request. N = Normal request. Note: In receive mode, Type=N requires that the remote software is Sterling Connect:Express. I = Inquiry (in Receive Mode only). This is used to download a file from the remote partner. H = Hold (in server transmitter mode only). This is used when you are waiting for selection with an inquiry request from the remote Partner. M = Message. This is a transmission request, based on the PeSIT Message protocol.
Type of Connection	1 alphabetic character	Optional. The link type of the connection. The following values are valid: X = Partner is linked by X.25 network. P = Partner is linked by PAD (Hold request type only). T = Partner is linked by TCPIP. The link type must be compatible with those given in the partners directory.
Priority	1 numeric character	Optional. The priority of the transfer. This is only used with the PeSIT protocol. 0 = Urgent 1 = Normal 2 = Slow If this value is not entered, the default is taken from the files directory.
Date	14 numeric characters	Optional. The date for a deferred transfer in the format, yyyymmddhhmmss. The current date is entered by default.

If the request is accepted, GTRF returns a request number and a record is created in the RENC file. A request can be rejected for any of the following reasons:

- File is not referenced in the files directory
- File is disabled in the files directory
- Partner is not referenced in the partners directory
- Partner is disabled in the partners directory
- Direction not authorized for this File
- Direction not authorized for this Partner
- Invalid protocol
- Invalid connection type

See *Appendix B Return Codes* for descriptions of Sterling Connect:Express return codes (TRC).

Batch Transfer Requests

You can execute transfer requests using the p1b8preq utility provided with the Application program interface. p1b8preq is in the libitom.a library in the ITOM directory.

The executable program p1b8preq is called by a user Shell procedure and can receive up to 4 arguments separated by spaces.

The first argument has several parameters that define the transfer, the second and third arguments are used to define two of the following three parameters: the local physical name, the remote physical name (User Data field for PeSIT and FTP protocols), and the PeSIT label. The following table describes the batch request fields.

Note: The local and remote physical names can be defined in the file definition, but the PeSIT label cannot. If the three parameters must be used at the same time, one of the physical names must be defined in the directory.

Argument	Field	Length	Description
Transfer Definition (Arg 1)	SYMBOLIC FILE NAME (/SFN=...)	8 characters	Mandatory
	SYMBOLIC PARTNER NAME (/SPN=...)	8 characters	Optional
	PRIORITY (/PRT=...)	0, 1, or 2	Optional
	LINK (/LNK=...)	T, P or X	Optional
	DATE (/DAT=...)	yyyymmddhhmmss	Optional
	DIRECTION (/DIR=...)	T or R	Optional
	REQUEST TYPE (/TYP=...)	N, I or H	Optional
	Dpcsid Alias (/SID=...)	8 characters	Optional
	Dpcpsw Alias (/PSW=...)	8 characters	Optional
	Origin (/ORG=...)	8 characters	Optional
	Destination (/DST=...)	8 characters	Optional
	Sender (/P61=...)	24 characters	Optional
	Receiver (/P62=...)	24 characters	Optional
	Record Format (/RFM=...)	2 alphabetic characters (TV, TF, BU, BF)	Optional
	Record Length (/RLG=...)	5 numeric	Optional
	FTP Format (/TSM=...)	3 alphabetic characters A,E,B,* F,S,* B,R,*	Optional
	FTP STORE FLAG(/STO=...)	1 alphabetic character	(Y,N) Optional
	Notification (/NTF=...)	1 numeric character	(0-7) Optional
Physical Name Definition (Arg 2, 3, or 4)	PHYSICAL NAME (/DSN=...)	127 characters	Optional
User Data Definition (Arg 2, 3, or 4)	USER DATA (/UDF=...)	44 characters	Optional
Label Definition **(Arg 2, 3, or 4)**	LABEL (/LAB=...)	80 characters	Optional

Argument	Field	Length	Description
P99 Field **(Arg 2, 3, or 4)**	P99 FIELD (/P99=...)	Max: 254 characters	Optional. PeSIT User Field.

Syntax Rules

The transfer definition parameter is mandatory. It is made up of different subparameters separated by a slash / and defined by keywords. Blanks are not allowed between subparameters in the first argument. The transfer definition must be the first parameter.

The file symbolic name parameter is the only mandatory transfer definition subparameter. If the other parameters are omitted, default values are taken from the Sterling Connect:Express directories. The following example shows the batch request structure (Shell command file) using p1b8preq.

```
$TOM_DIR/itom/p1b8preq "/SFN=FILE/PRT=1/LNK=T/SPN=PART"  "/DSN=/tmp/TOM.tmp"
```

Note: Application transfer requests are described in Chapter 5 *Application Interface*.

The End to End utility

The end to end utility, called p1b8pe2e, enables you to forward and acknowledge transfers of files and messages.

Acknowledging a Transfer

If the request is present in the RENC file, it is possible to acknowledge it by referencing its number, as shown below:

```
$TOM_DIR/itom/p1b8pe2e "/FUN=E/REQ=10400065/SPN=adjacent" "/ACK='feedback message'"
```

The SPN parameter is necessary if the initial node is not the adjacent partner.

If the request is no longer in the RENC file, all parameters from the initial transfer must be provided:

```
$TOM_DIR/itom/p1b8pe2e"/FUN=E/SPN=adjacent"
                      "/P12=filef/P11=XX/P03=oo/P04=dd/P13=id/p51=dh/p61=cc/p62=bb"
                      "/ACK='feedback message'"
```

Forwarding a Transfer

If the request is present in the RENC file, it is possible to forward it by referencing its number, as shown below:

```
$TOM_DIR/itom/p1b8pe2e "/FUN=F/REQ=10400065/SPN=adjacent"
```

The SPN parameter is required.

If the request is no longer in the RENC file, all parameters from the initial transfer must be provided:

```
$TOM_DIR/itom/p1b8pe2e"/FUN=F/TYP=N/SPN=adjacent"
                      "/P12=filef/P11=XX/P03=oo/P04=dd/P13=id/p51=dh/p61=cc/p62=bb"
```

P1b8pe2e Reference

This section provides the syntax rules and all parameters that apply to p1b8pe2e utility.

P1b8pe2e utility can receive one to five parameters, depending on the type of function used and the way the transfer definition is passed. Parameter #1 can provide general transfer request parameters such as priority, notification options, link , scheduling date etc ….

The tables below list the parameters and sub-parameters and provide a description and rules for each.

EERP - Request

This request refers to the reception initial request, using the /REQ= subparameter.

Argument	Field	Description	Required or default
#1	FUN	Function - E=EERP	Required
	REQ	Request number, 8 numeric characters. Example: /REQ=09800005	Required
	SPN	Remote partner name (adjacent)	Required
	SID	Local name (alias)	RPAR/Sysin
	PSW	Local password (alias)	RPAR/Sysin
	NTF	Notification option	RFIC
	PRT	Priority	RFIC
	LNK	Link type	RPAR
	DAT	Scheduling date	Immediat
	FAG	File agent option	N
#2		Eerp acknowledgment (message or file) default from the $$EERP$$ definition.	RFIC/$$EERP$$
ACK		Eerp acknowledgment (message)	
DSN		Eerp acknowledgment (file)	

EERP - Transfer Definition

This request provides the initial request information. No /REQ= parameter is provided , all transfer information is provided in parameter #2.

Argument	Field	Description	Required or default
#1			
	FUN	Function - E=EERP	Required
	SPN	Remote partner name (adjacent)	Required
	SID	Local name (alias)	RPAR/Sysin
	PSW	Local password (alias)	RPAR/Sysin
	NTF	Notification option	RFIC
	PRT	Priority	RFIC
	LNK	Link type	RPAR
	DAT	Scheduling date	Immediat
	FAG	File agent option	N
#2		Transfer definition	Required
	ORG	Origine of transfer. 1 to 8 characters. (pi3)Example: /ORG=Orgtrf01	Required
	DST	Destination of transfer. 1 to 8 characters. (pi4)Example: /DST=DSTtrf01	Required
	P11	File type. 4 hexadecimal characters. (Pi11) Example: 01FA	Required
	P12	File name. 1 to 8 characters. (pi12) – RFIC definition. Example: /P12=Ftest01	Required
	P13	Transfer identification. 1 to 8 numeric characters. (pi13) Example /P13=18	Required
	P51	File creation date: 12 numeric characters. Example: /P51=040110092503	Required
	P61	Transfer sender: 0 to 24 characters. (pi61) Example: /P61=Client name	Required
	P62	Transfer receiver: 0 to 24 characters. (pi62) Example: /P62=Service name	Required
#3 ACK DSN		Eerp acknowledgment (message or file) default from the $$EERP$$ definition. Eerp acknowledgment (message) Eerp acknowledgment (file)	RFIC/$$EERP$$

Forwarding a Request

This request refers to the reception initial request. Only parameter #1 is provided. /DSN, /P99, /LAB are invalid as these information are retreived in the RENC information for the initial request.

Argument	Field	Description	Required or default
#1	FUN	Function - F=Forward	Required
	REQ	Request number, 8 numeric characters. Example: /REQ=09800005	Required
	SPN	Remote partner name (adjacent)	Required
	SID	Local name (alias)	RPAR/Sysin
	PSW	Local password (alias)	RPAR/Sysin
	NTF	Notification option	RFIC
	PRT	Priority	RFIC
	LNK	Link type	RPAR
	DAT	Scheduling date	Immediat
	FAG	File agent option	N

Forwarding a Transfer Definition

This request provides the initial request information. No /REQ= parameter is provided.

Argument	Field	Description	Required or default
#1			
	FUN	Function - F=Forward	Required
	SPN	Remote partner name (adjacent)	Required
	SID	Local name (alias)	RPAR/Sysin
	PSW	Local password (alias)	RPAR/Sysin
	NTF	Notification option	RFIC
	PRT	Priority	RFIC
	LNK	Link type	RPAR
	DAT	Scheduling date	Immediat
	FAG	File agent option	N
#2		Transfer definition	Required
	ORG	Origine of transfer. 1 to 8 characters. (pi3)Example: /ORG=Orgtrf01	Required
	DST	Destination of transfer. 1 to 8 characters. (pi4)Example: /DST=DSTtrf01	Required
	P11	File type. 4 hexadecimal characters. (Pi11) Example: 01FA	Required
	P12	File name. 1 to 8 characters. (pi12) – RFIC definition. Example: /P12=Ftest01	Required
	P13	Transfer identification. 1 to 8 numeric characters. (pi13) Example /P13=18	Required
	P51	File creation date: 12 numeric characters. Example: /P51=040110092503	Required
	P61	Transfer sender: 0 to 24 characters. (pi61) Example: /P61=Client name	Required
	P62	Transfer receiver: 0 to 24 characters. (pi62) Example: /P62=Service name	Required
#3, #4, #5			
DSN		Physical file name	RFIC
P99		User data	RFIC
LAB		File label	

Error Codes

This section provides the meaning of the return code from p1b8pe2e utility.The return code is a 4 characters field structured as shown below.

Field	Definition
1	1 numeric character: parameter value – from 1 to 5
2	2 numeric characters: sub parameter value. 00 Other 01 Priority 02 Direction 03 Link 04 Partner 05 File 06 Physical Name 07 User Data Field 08 Date 09 Monitor 10 Request Number 11 Alias Name 12 Alias Password 13 Record Format 14 Record Length 15 Api 16 State 17 Request Type 18 Type/Struct/Mode FTP 19 Store/Unique FTP 20 File agent flag Y/N 21 Label 22 Pi99 254 23 User Origin 24 User Destination 25 Pi61 26 Pi62 27 Julian Date 28 Notification 29 Eerp/snf pi11 30 Eerp/snf pi12 31 Eerp/snf pi13 32 Eerp/snf pi51 33 Eerp ACK 34 Eerp or FWD
O	1 numeric character: error code: 1 Invalid Field 2 Duplicate Field 3 Invalid Field Length 4 Missing Required Field

Example: 2331 is for parameter 2, subparameter pi13, invalid length.

Chapter 5

Application Interface

This chapter discusses the application program interface and the utilities provided for integrating transfer operations into your environment.

Overview

This manual assumes that Sterling Connect:Express for UNIX is installed in the /home/tom1 directory. To communicate with one monitor with ITOM, an environment variable TOM_DIR must be defined in your shell in the root directory of installation, for example /home/tom1.

The application interface is based on an object module called l0b8z20. All directories, table management functions, and transfer requests that are accessed through the main menu can be accessed from a user application. There are also utilities programs that you can access from a shell user procedure. These batch utilities work through the l0b8z20 interface to communicate with Sterling Connect:Express.

Batch Utilities

The lob8z20 module enables user applications to call Sterling Connect:Express by communicating with GTRF through a permanent interface. The following modules are provided in the ITOM directory.

Module	Description
libitom.a	Library with LOB8Z20.o object
d0b8z20.h	Standard communication structure between application programs and Sterling Connect:Express. This structure describes the RENC file.
p1b8preq	Initiates a transfer request. This function is described in Chapter 4 Transfers.
p1b8pe2e	Initiates a transfer forward request, or a transfer end to end ackowledgment. This function is described in Chapter 4 Transfers.
p1b8pcan	Interrupts a transfer request.
p1b8ppur	Purges a transfer request.

Module	Description
p1b8pret	Restarts a transfer request.
p1b8pren	Displays requests from the RENC file.
p1b8ppar_c	Creates Partner records in the RPAR file.
p1b8ppar_d	Displays Partner records.
p1b8ppar_m	Updates Partner records.
p1b8ppar_s	Deletes Partner records.
p1b8pfil_c	Creates File records in the RFIC file.
p1b8pfil_d	Displays File records.
p1b8pfil_m	Updates File records.
p1b8pfil_s	Deletes File records.

lob8z20 must be linked with the user program by adding -L/home/tom1/itom -litom to the cc command. Refer to samples in the ITOM/SAMPLES directory.

Interrupting a Transfer

A user Shell procedure can call the executable program p1b8pcan and give the argument to interrupt a request number, /REQ=QQQNNNNN. The following screen shows an example.

```
$TOM_DIR/itom/p1b8pcan /REQ=10400065
```

Purging a Transfer

A user Shell procedure can call the executable program p1b8ppur and give the following arguments:

Argument	Description
/REQ=QQQNNNNN	Request number to delete or to filter by
/DAT=YYYYMMDDHHMMSS	Transfer date
/DIR=	Direction of transfer
/SFN=	Symbolic file name
/SPN=	Symbolic partner name
/QQQ=AAqqq	AA = year, qqq = Julian date
/STA	Status of the request

```
$TOM_DIR/itom/plb8ppur /REQ=10400065
```

Note: If no parameter value is specified, all records are deleted from the RENC file.

Restarting a Transfer

A user Shell procedure can call the executable program p1b8pret and give the argument to restart a request number, /REQ=QQQNNNNN. The following screen shows an example.

```
$TOM_DIR/itom/p1b8pret  /REQ=10400065
```

Display Requests from the RENC File

A user Shell procedure can call the executable program p1b8pren and give the argument to display a request number, /REQ=QQQNNNNN. The following screen shows an example.

```
$TOM_DIR/itom/p1b8pren   /REQ=10400065
```

Display Partners from the RPAR File

A user Shell procedure can call the executable program p1b8ppar_d and give the argument to display the symbolic Partner name, /SPN=PPPPPPPP. The following screen shows an example.

```
$TOM_DIR/itom/p1b8ppar_d   /SPN=PARTNER
```

Displaying Files from the RFIC File

A user Shell procedure can call the executable program p1b8pfil_d and give the argument to display the symbolic file name, /SFN=PPPPPPPP. The following screen shows an example.

```
$TOM_DIR/itom/p1b8pfil_d   /SFN=FILENAME
```

Deleting Partners from the RPAR File

A user Shell procedure can call the executable program p1b8ppar_s and give the argument to delete the symbolic Partner name, /SPN=PPPPPPPP. The following screen shows an example.

```
$TOM_DIR/itom/p1b8ppar_s   /SPN=PARTNER
```

Deleting Files from the RFIC File

A user Shell procedure can call the executable program p1b8pfil_s and give the argument to delete the symbolic file name, /SFN=PPPPPPPP. The following screen shows an example.

```
 $TOM_DIR/itom/p1b8pfil_s   /SFN=FILENAME
```

Error Messages and Results

The executable return values for all utilities are listed below. This return code is found in the variable *$?* of the Shell environment. Refer to the sample procedures in the ITOM directory.

Return Value	Description
0	Return code is OK.
1	Number of arguments is incorrect.
2	An error has been detected. See return code XYYZ.
3	An error has been detected. See *Appendix B Return Codes* for a list of Sterling Connect:Express return codes.

When the shell variable $? = 2, the value returned to the standard error output is in the format XYYZ, where X=argument number (1,2,3), YY=Field that contains the error, and Z=Error type. The following screen shows an example. This is found in the itom/d0b8z20.h file.

```
/* Internal Error Return Code */
#define ERROR_BAD_FUNC  2900
#define ERROR_CRE_QUEUE 2901
#define ERROR_PB_SEND   2902
#define ERROR_PB_RECV   2903
#define ERROR_TIME_OUT  2904
#define ERROR_NOTOM     2912
#define ERROR_OTHER     2999

/* External Error Status (4 digits) : XYYZ */
/* X  : argument number (1,2,3) */
/* YY : Field which contains error */
/* Z  : Error type */

/* YY */
#define Y_OTH  0        /* Other           */
#define Y_PRT  1        /* Priority        */
#define Y_DIR  2        /* Direction       */
#define Y_LNK  3        /* Link            */
#define Y_SPN  4        /* Partner         */
#define Y_SFN  5        /* File            */
#define Y_DSN  6        /* Physical Name   */
#define Y_UDF  7        /* User Data Field */
#define Y_DAT  8        /* Date            */
#define Y_MNM  9        /* Monitor         */
#define Y_REQ 10        /* Request Number  */
#define Y_SID 11        /* Alias Name      */
#define Y_PSW 12        /* Alias Password  */
#define Y_RFM 13        /* Record Format   */
#define Y_RLG 14        /* Record Length   */
#define Y_API 15        /* Api             */
#define Y_STA 16        /* State           */
#define Y_TYP 17        /* Request Type    */
#define Y_TSM 18        /* Type/Struct/Mode FTP */
#define Y_STO 19        /* Store/Unique FTP */
#define Y_FAG 20 /* File agent flag Y/N */
#define Y_LAB 21 /* Label */
#define Y_P99 22 /* Pi99 on 254 */
#define Y_ORG 23 /* User Origin */
#define Y_DST 24 /* User Destination */
#define Y_P61 25 /* PI61 */
#define Y_P62 26 /* PI62 */
#define Y_QQQ 27 /* Julian date */
#define Y_NTF 28 /* Notification */

/* Z */
#define Z_INV_FIELD     1       /* Invalid Field            */
#define Z_DUP_FIELD     2       /* Duplicate Field          */
#define Z_LG_FIELD      3       /* Invalid Field Length     */
#define Z_MIS_FIELD     4       /* Missing Compulsory Field */
```

Application Program Using L0B8Z20

The l0b8z20 module enables user applications to call Sterling Connect:Express by communicating with GTRF through a permanent interface to access Request or Display services. The l0b8z20 module also lets the batch utilities call Sterling Connect:Express and access these services.

The l0b8z20 module called by the application program is in the libitom.a object library and must be included in the link procedure of the program. The following table describes the available modules.

Module	Description
libitom.a	Standard interface library (.o module for applications that are bound into program).
d0b8z20.h	Standard communication structure (text file describing interface record layout).
p1b8pren.c, ... p1b8ppar_d.c	Source code that can be used as examples.

Note: All of the tables in the following sections refer to the module d0b8z20.h.

Starting a Transfer Request

This option allows the user to request a transfer.

Call to L0B8Z20 Module

The following table describes the fields in the header:

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	T	TRANSFER function
ZREQ_TOM_TABN	01	R	File concerned : RENC
ZREQ_TOM_REQN	08	X'0	Request number
ZREQ_TOM_RTCF	01	X'0	Return code
ZREQ_TOM_RSCF	03	X'0	Reason

The following table describes the fields in the structure SCI_ST.

Field	Length	Value	Description
dire	01	T or R	Direction
file	08	Mandatory	Symbolic file name

Field	Length	Value	Description
part	08		Symbolic partner name
dsnam	127		Physical file name
prty	01	0, 1, 2	Priority
dat	08		Date of transfer
Hour	06		Hour of transfer
Lnk	01	X, P, T	Type of link
Udf	44		User Data Field
Typ	01	N, I, H	Request type
Sta	01	IGNORE	State of transfer
Dpcsid	08		Dpcsid alias
Dpcpsw	08		Dpcpsw alias
Format	02	TF, TV, BF, BU	Record format
Lrecl	05		Record length
Api	88		Api field
Tsm	3	A, E, B, * F, S, * B, R, *	Type, Structure, and Mode parameters in FTP protocol
Stou	1	Y, N	Store Unique flag (Yes or No)
Fa	1	Y, N	File Agent flag (Yes or No)
Label	80		Label
S_pi99_254	254		Sender PI99
User_org	8		User origin
User_dst	8		User Destination
User_snd	24		User sender
User_rcv	24		User receiver
Quant_aa	2		Year of Julian calendar
Quant	3		Julian date
Notif	1		Notification flag
Filler (See "Description D0B8Z20.H")			

Positive Return of L0B8Z20 Module

The following table describes the fields in the header:

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	T	TRANSFER function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Negative Return of L0B8Z20 Module

The following table describes the fields in the header:

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	T	TRANSFER function
ZREQ_TOM_TABN	01	R	File concerned : RENC
ZREQ_TOM_REQN	08	X0	Request number
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Interrupting a Transfer Request

This option enables the user to interrupt a transfer.

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	I	INTERRUPTION function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code

Field	Length	Value	Description
ZREQ_TOM_RSCF	03	X0	Reason

Positive Return of L0B8Z20 Module

With a positive return, there is a 0 in the return code field. The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	I	INTERRUPTION function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Negative Return of L0B8Z20 Module

With a negative return, there is a value in the return code field. The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	I	INTERRUPTION function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	X0	Request number
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC.	Reason

Purging a Transfer Request

This option enables you to purge a transfer. You can purge a group of requests, all requests, or a single request by request number. The following table identifies the filters that you can use with the Purge command:

Filter by	Description
Date/time field	Enter the date/time in the format yyyymmddhhmmss to delete all requests older than this date. You can also enter a truncated value, for example yyyymm00000000. A * deletes all requests.

Filter by	Description
Symbolic Partner name	Enter the Partner name or type * to purge transfer requests for all Partners.
Symbolic File name	Enter the File name or type * to purge transfer requests for all Files.
Transfer Direction	Enter any of the following values: T - Purges all transmission requests. R - Purges all reception requests. * - Purges both transmission and reception requests.
Transfer Status	Enter any of the following values: A, D, E, H, J, K, O, C or * for all.

Caution: Deleting a request with a status of C can result in error messages in the log file.

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	P	PURGE function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

The following table describes the parameters for the request using the structure SCI_ST.

Field	Length	Value	Description
Dire	01	T or R	Direction
File	08	Name or *	Symbolic file name
Part	08	Name or *	Symbolic partner name
Dsnam	127	Ignored	Physical file name
Prty	01	Ignored	Priority
Dat	08	YYYYMMDD or *	Date of transfer
Hour	06	HHMMSS or *	Hour of transfer
Lnk	01	Ignored	Type of link
Udf	44	Ignored	User Data Field

Field	Length	Value	Description
Typ	01	Ignored	Request type
Sta	01	A,C,D,E,H,J,K,O	State of transfer
Dpcsid	08	Ignored	Dpcsid alias
Dpcpsw	08	Ignored	Dpcpsw alias
Format	02	Ignored	Record format
Lrecl	05	Ignored	Record length
Api	88	Ignored	Api field
Tsm	3	Ignored	Type, Structure and Mode parameters in FTP protocol.
Stou	1	Ignored	Store Unique flag (Yes or No)
Fa	1	Y, N	File Agent flag (Yes or No)
Label	80	Ignored	Label
S_pi99_254	254	Ignored	Sender PI99
User_org	8	Ignored	User origin
User_dst	8	Ignored	User Destination
User_snd	24	Ignored	User sender
User_rcv	24	Ignored	User receiver
Quant_aa	2	Ignored	Year of Julian date
Quant	3	Ignored	Julian date
Notif	1	Space/0/1/2/3	Notification flag

Positive Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	P	PURGE function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	P	PURGE function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	X0	Request number
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Restarting a Transfer Request

This option enables you to restart a transfer.

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	R	RESTART function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason
User_org	8	Ignored	User Origin
User_dst	8	Ignored	User Destination
User_snd	24	Ignored	User Sender
User_rcv	24	Ignored	User Receiver

Positive Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name

Field	Length	Value	Description
ZREQ_TOM_FUNC	01	R	RESTART function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	R	RESTART function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	X0	Request number
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Displaying a Record from the RENC File

This option enables you to display a transfer record from the RENC file.

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	D	DISPLAY function.
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Positive Return of L0B8Z20 Module

The following table describes the fields in the header, followed by the display structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	D	DISPLAY function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

S_RENC Structure. Refer to the structure for d0b8z20.h.

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	D	DISPLAY function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	X0	Request number
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Display a Record from the RPAR File

This option enables you to display a partner record from the RPAR file.

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	D	DISPLAY function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	X0	Return code

Field	Length	Value	Description
ZREQ_TOM_RSCF	03	X0	Reason

Positive Return of L0B8Z20 Module

The following table describes the fields in the header, followed by the display structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	D	DISPLAY function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Partner Structure. Refer to the structure for d0b8z20.h.

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	D	DISPLAY function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Displaying a Record from the RFIC File

This option enables you to display a file record from the RFIC file.

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name

Field	Length	Value	Description
ZREQ_TOM_FUNC	01	D	DISPLAY function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Positive Return of L0B8Z20 Module

The following table describes the fields in the header, followed by the display structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	D	DISPLAY function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

File Structure. Refer to the structure for d0b8z20.h.

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	D	DISPLAY function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Deleting Partners from the RPAR File

This option enables you to delete a partner record from the RPAR file.

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	S	DELETE function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Positive Return of L0B8Z20 Module

The following table describes the fields in the header, followed by the display structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	S	DELETE function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Partner structure. Refer to the structure for d0b8z20.h.

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	S	DELETE function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Deleting Files from the RFIC File

This option enables you to delete a file record from the RFIC file.

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	S	DELETE function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Positive Return of L0B8Z20 Module

The following table describes the fields in the header, followed by the display structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	S	DELETE function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

File structure. Refer to the structure for d0b8z20.h.

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	S	DELETE function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Creating Partner Records in the RPAR File

This option enables you to create a partner record in the RPAR file.

Call to L0B8Z20 Module

The following table describes the fields in the header, followed by the Partner structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	C	CREATE function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Partner structure. Refer to the structure for d0b8z20.h.

Positive Return of L0B8Z20 Module

The following table describes the fields in the header, followed by the display structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	C	CREATE function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Partner structure. Refer to the structure for d0b8z20.h.

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	C	CREATE function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Creating a File Record in the RFIC File

This option enables you to create a file record in the RFIC file.

Call to L0B8Z20 Module

The following table describes the fields in the header, followed by the file structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	C	CREATE function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

File structure. Refer to the structure for d0b8z20.h.

Positive Return of L0B8Z20 Module

The following table describes the fields in the header, followed by the display structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	C	CREATE function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

File Structure. Refer to the structure for d0b8z20.h.

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	C	CREATE function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Updating the RPAR File

This option enables you to update a partner record in the RPAR file.

Call to L0B8Z20 Module

The following table describes the fields in the header, followed by the Partner structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	M	UPDATE function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Partner Structure. Refer to the structure for d0b8z20.h.

Positive Return of L0B8Z20 Module

The following table describes the fields in the header, followed by the display structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	M	UPDATE function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Partner structure. Refer to the structure for d0b8z20.h.

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	M	UPDATE function
ZREQ_TOM_TABN	01	P	File concerned: RPAR
ZREQ_TOM_REQN	08	PARTNER	Partner name
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Updating the RFIC File

This option enables you to update a file record in the RFIC file.

Call to L0B8Z20 Module

The following table describes the fields in the header, followed by the File structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	M	UPDATE function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

File Structure. Refer to the structure for d0b8z20.h.

Positive Return of L0B8Z20 Module

The following table describes the fields in the header, followed by the display structure.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	M	UPDATE function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

File structure. Refer to the structure for d0b8z20.h.

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	M	UPDATE function
ZREQ_TOM_TABN	01	F	File concerned: RFIC
ZREQ_TOM_REQN	08	FILE	File name
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Starting a Transfer Acknowledgment

This option allows the user to request an end to end acknowledgment of a reception . The initial transfer request does not need to be present in the RENC file. The data received can be a file or a message.

Call to L0B8Z20 Module

The following table describes the fields in the header:

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	E	EERP function
ZREQ_TOM_TABN	01	R	File concerned : RENC
ZREQ_TOM_REQN	08	X'0	Request number
ZREQ_TOM_RTCF	01	X'0	Return code
ZREQ_TOM_RSCF	03	X'0	Reason

The following table describes the fields in the structure SCI_ST.

Field	Length	Value	Description
dire	01	T	Direction
file	08	Required	Symbolic file name
part	08		Symbolic partner name
dsnam	127		Physical file name - can be used to send a feedback message. Pi99_254 is used first
prty	01	0, 1, 2	Priority
dat	08		Date of transfer
Hour	06		Hour of transfer
Lnk	01	T	Type of link
Typ	01	E	Request type EERP
Dpcsid	08		Dpcsid alias
Dpcpsw	08		Dpcpsw alias
Fa	1	Y, N	File Agent flag (Yes or No)
S_pi99_254	254		Feedback message
User_org	8	Required	User origin (Pi3)
User_dst	8	Required	User Destination (Pi4)
User_snd	24	Required	User sender (Pi61)

Field	Length	Value	Description
User_rcv	24	Required	User receiver (Pi62)
Notif	1		Notification flag
dhc	12	Required YYMMDD HHMMSS	File creation date (Pi51)
idt	8	Requirednnnn nnnn	Transfer identification (Pi13)
ftyp	4	Required HHHH	File type (Pi11) (hexadecimal)
Filler (See "Description D0B8Z20.H")			

Positive Return of L0B8Z20 Module

The following table describes the fields in the header:

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	E	EERP function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Negative Return of L0B8Z20 Module

The following table describes the fields in the header:

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	E	EERP function
ZREQ_TOM_TABN	01	R	File concerned : RENC
ZREQ_TOM_REQN	08	X0	Request number
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Forwarding a Transfer

This option allows the user to forward a reception previously completed. The transfer request does not need to be present in the RENC file. The data received can be a file, a message or an eerp.

Call to L0B8Z20 Module

The following table describes the fields in the header:

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	F	FORWARD function
ZREQ_TOM_TABN	01	R	File concerned : RENC
ZREQ_TOM_REQN	08	X'0	Request number
ZREQ_TOM_RTCF	01	X'0	Return code
ZREQ_TOM_RSCF	03	X'0	Reason

The following table describes the fields in the structure SCI_ST.

Field	Length	Value	Description
dire	01	T	Direction
file	08	Required	Symbolic file name
part	08		Symbolic partner name
dsnam	127		Physical file name
prty	01	0, 1, 2	Priority
dat	08		Date of transfer
Hour	06		Hour of transfer
Lnk	01	X, P, T	Type of link
Typ	01	N,E,M	Request type (normal transfer, eerp, message)
Dpcsid	08		Dpcsid alias
Dpcpsw	08		Dpcpsw alias
Fa	1	Y, N	File Agent flag (Yes or No)
label	80		
S_pi99_254	254		
User_org	8	Required	User origin (Pi3)
User_dst	8	Required	User Destination (Pi4)
User_snd	24	Required	User sender (Pi61)

Field	Length	Value	Description
User_rcv	24	Required	User receiver (Pi62)
Notif	1	Required	Notification flag
dhc	12	Required YYMMDD HHMMSS	File creation date (Pi51)
idt	8	Required nnnnnnnn	Transfer identification (Pi13)
ftyp	4	Required HHHH	File type (Pi11) (hexadecimal)
Filler (See "Description D0B8Z20.H")			

Positive Return of L0B8Z20 Module

The following table describes the fields in the header:

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	F	FORWARD function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Negative Return of L0B8Z20 Module

The following table describes the fields in the header:

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	F	FORWARD function
ZREQ_TOM_TABN	01	R	File concerned : RENC
ZREQ_TOM_REQN	08	X0	Request number
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Acknowledging a Transfer Request

This option enables you to send an end to end acknowledgment of a reception request that is recorded in the RENC file. The data received can be a file or a message .

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	E	EERP function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason
dsnam	127		Physical file name - can be used to send a feedback message. Pi99_254 is used first
Typ	01	E	Request type EERP
S_pi99_254	254		Feedback message

Positive Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	E	EERP function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name

Field	Length	Value	Description
ZREQ_TOM_FUNC	01	E	EERP function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	X0	Request number
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Forwarding a Transfer Request

This option enables you to forward a reception request that is recorded in the RENC file. The data received can be a file, a message or an eerp.

Call to L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	F	FORWARD function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason
dsnam	127		Physical file name - can be used to send a feedback message. Pi99_254 is used first
Typ	01	N,E,M	Request type (normal transfer, eerp, message)
S_pi99_254	254		Feedback message

Positive Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	F	FORWARD function

Field	Length	Value	Description
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	QQQNNNNN	Request number
ZREQ_TOM_RTCF	01	X0	Return code
ZREQ_TOM_RSCF	03	X0	Reason

Negative Return of L0B8Z20 Module

The following table describes the fields in the header.

Field	Length	Value	Description
ZREQ_TOM_NAME	04	TOM1	Monitor name
ZREQ_TOM_FUNC	01	F	FORWARD function
ZREQ_TOM_TABN	01	R	File concerned: RENC
ZREQ_TOM_REQN	08	X0	Request number
ZREQ_TOM_RTCF	01	Contains	Return code
ZREQ_TOM_RSCF	03	TRC	Reason

Appendix A

Sterling Connect:Express Messages

This appendix provides a list of error messages in the Sterling Connect:Express log file.

Syntax of Error Messages

The following table defines the codes and syntax that are used in the message descriptions.

Code	Description
SRC	System Return Code, generally refers to the errno UNIX value as described in /usr/include/sys/errno.h
TRC	Sterling Connect:Express Return Code. See *Appendix B Return Codes* for a list of Sterling Connect:Express return codes. In PeSIT level 2, the TRC is followed by an L or an R to indicate that the code is for the Local or the Remote computer. If the TRC = 0, the error is on the remote system.
NRC	Network Return Code. This NRC can have 2 fields. The first one is the decimal value returned by the system errno (see SRC). The second one is a Cause (2 bytes) and Diagnostic (2 bytes) (X25 only).
PRC	Protocol Return Code. See *Appendix B Return Codes* for a list of protocol return codes.
QQQNNNNN	Request number
PPPPPPPP	Symbolic Partner Name
FFFFFFFF	Symbolic File Name
UUUUUUUUUUUU	User Id
SSSSSSSSSS	STRF Process ID
T	Type of request (Normal, Inquiry, Hold).
LLLL	Link for this transfer (X25 or TCPIP).
MM	Time to wait before next retry.

SYSLOG Option

If the SYSLOG option flag equals 1 in the SYSIN file, all messages written in the log file are sent to the syslog daemon based on the following guidelines:

1. All error messages are logged with a LOG_ERR priority.
2. All information messages are logged with a LOG_INFO priority.
3. Messages 124, 201 to 203, 211 to 213, 222, 223, 243, 300 are logged with a LOG_NOTICE priority.

Logging Messages

The following table describes logging messages in Sterling Connect:Express for UNIX. The messages are listed by message number.

Code	Description
007	REQUEST QQQNNNNN LAST CREATED QQQNNNNN ERROR IN SYSTEM DATE Message Type: ERROR Explanation: A problem occurred during the creation of the new request number. The system date may have been updated, so the request numbers may not be sequential. Action: Delete the RTAB file in the config directory and purge the RENC file. The RTAB file contains information about the last request number created.
011	REQUEST QQQNNNNN FFFFFFFF PPPPPPPP SRC=.... TRC=.... PRC=.... Message Type: INFORMATION Explanation: This message gives information about request QQQNNNNN, and is generally followed by an error message.
013	REQUEST QQQNNNNN (D)..... Physical Name..... Message Type: INFORMATION Explanation: This message gives information about request QQQNNNNN, and gives you the physical name of the transferred file. This message generally follows message 131.
014	REQUEST XXXXXXXX - NEW LINK: LLLLLL Message Type: INFORMATION Explanation: This message gives information about request QQQNNNNN, and explains that the monitor is trying another link for this request. This message appears when the monitor failed to contact or transfer the file on one link (X25 or TCP/IP) and is going to try a new link type.
020	REQUEST QQQNNNNN <- UUUUUUUU REJECTED (T) Message Type: ERROR Explanation: The transfer request was rejected by the monitor. Return codes in the previous message explain why. Action: Make corrections based on the codes, and then submit another request.
021	REQUEST QQQNNNNN <- UUUUUUUU ACCEPTED (T) Message Type: INFORMATION Explanation: Transfer request was accepted by monitor.
035	REQUEST QQQNNNNN NOT RESTARTABLE SRC=.... TRC=.... PRC=.... Message Type: ERROR Explanation: The request to restart the transfer was rejected by the monitor. The request must be terminated and in a restartable state. Action: Check request information.

Code	Description
036	REQUEST QQQNNNNN NOT INTERRUPTABLE SRC=.... TRC=.... PRC=.... Message Type: ERROR Explanation: Interruption request was rejected by the monitor. The Request must be running. Action: Wait for request to end.
037	REQUEST QQQNNNNN INTERRUPTED BY UUUUUUUU Message Type: INFORMATION Explanation: Interruption request was accepted by the monitor.
038	INTERRUPT REQUEST TRANSMIT TO STRF SSSSSSSSSS REQ: QQQNNNNN Message Type: INFORMATION Explanation: Interruption request was transmitted to the STRF that processes transfers.
039	INTERRUPT REQUEST NOT TRANSMIT TO STRF SSSSSSSSSS REQ: QQQNNNNN SRC=.... TRC=.... Message Type: ERROR Explanation: The monitor accepted the interruption of transfer, but could not transmit the message to the STRF. Action: Check the existence of STRF with PID SSSSSSSSSS. If there is no STRF, keep as much information as possible and contact technical support.
040	REQUEST QQQNNNNN FFFFFFFF SUSPENDED TRC=.... PRC=.... Message Type: INFORMATION Explanation: Transfer was suspended.
058	REQUEST QQQNNNNN PURGED BY UUUUUUUU Message Type: INFORMATION Explanation: Request was purged by the user listed in the message.
059	REQUEST QQQNNNNN NOT PURGED SRC=.... TRC=.... PRC=.... Message Type: ERROR Explanation: Deletion was rejected by the monitor. The request state is incompatible with deletion. Action: Check state of request.
102	REQUEST QQQNNNNN NOT CREATED SRC=.... TRC=.... PRC=.... Message Type: ERROR Explanation: Monitor is not able to create a transfer process (fork problem). Action: Check the cause based on the SRC, if possible.
103*/	*** ERROR IN CREATING MESSAGE QUEUE *** Message Type: ERROR Explanation: Monitor is not able to create a transfer process (fork problem). Action: Check orphan message queues and purge.
104*/	*** WARNING : DUPLICATION REQUEST NUMBER -> SHOULD PURGE RENC FILE *** Message Type: WARNING Explanation: Monitor is not able to create a request because the request number does exist. Action: Pureg the RENC file.
124	TOM INITIALIZATION COMPLETE Vxxx-r RUN=(T) Message Type: INFORMATION Explanation: This message appears when Sterling Connect:Express is initialized. It gives the version (xxx), the release number (r), and the start type (C: Cold, H: Hot) according to the Launch parameter in the SYSIN file.
125	LARGE FILE (>2Gbytes) SUPPORT AVAILABLE Message Type: INFORMATION Explanation: This message appears when Sterling Connect:Express is initialized.

Code	Description
126	LARGE FILE (>2Gbytes) SUPPORT NOT AVAILABLE Message Type: INFORMATION Explanation: This message appears when Sterling Connect:Express is initialized.
127	X25 DTE ADDRESS: xxxxxxxxxxxxxxxxxx TCPIP HOST ADDRESS: xxxxxxxxxxxxxxxxxx Message Type: ERROR Explanation: This message is logged when the X25 address (or TCPIP) address that the remote Partner used to call the Monitor is bad. Action: Check the remote partner entry in the partners directory.
130	REQUEST QQQNNNNN FFFFFFFF TRANSFER ENDED STRF SSSSSSSSSS Message Type: INFORMATION Explanation: Transfer ended normally.
131	REQUEST QQQNNNNN FFFFFFFF TRANSFER ACCEPTED STRF SSSSSSSSSS Message Type: INFORMATION Explanation: Transfer was accepted by the STRF transfer process.
132	REQUEST QQQNNNNN FFFFFFFF TRANSFER STARTED STRF SSSSSSSSSS Message Type: INFORMATION Explanation: The transfer was stopped. Some data was received (or sent).
133	COMMUNICATION OPENED (O) WITH: PPPPPPPP REQ: QQQNNNNN PESIT TCPIP Message Type: INFORMATION Explanation: X25 (or TCPIP) network connection was opened and the connection phase was acknowledged. An Outgoing session (O) was opened with request number QQQNNNNN in protocol PeSIT (or ETEBAC3 or FTP).
134	COMMUNICATION OPENED (I) WITH: PPPPPPPP REQ: QQQNNNNN PESIT TCPIP Message Type: INFORMATION Explanation: The X25 (or TCPIP) network connection was opened and the connection phase was acknowledged. The Incoming session (I) was opened with request number QQQNNNNN in protocol PeSIT (or ETEBAC3 or FTP).
135	REQUEST QQQNNNNN REJECTED SRC=.... TRC=.... PRC=.... Message Type: ERROR Explanation: Transfer was refused by the transfer process. Action: If TRC is 0, the Protocol Return Code was sent by the emote partner, otherwise correct the problem at the local site based on the local TRC.
136	REQUEST QQQNNNNN RECEIVING <- PPPPPPPP, FILE FFFFFFFF NUMBER OF RECORDS: ..... Message Type: INFORMATION Explanation: Transfer ended normally and the monitor received the number of records specified in the message from the remote partner.
137	REQUEST QQQNNNNN TRANSMITTING -> PPPPPPPP, FILE FFFFFFFF NUMBER OF RECORDS: ..... Message Type: INFORMATION Explanation: Transfer ended normally and the monitor sent the number of records specified in the message to the remote Partner.
138	COMMUNICATION CLOSED (O) WITH: PPPPPPPP Message Type: INFORMATION Explanation: Network Outgoing (O) session was closed on the local site.
139	COMMUNICATION CLOSED (I) WITH: PPPPPPPP Message Type: INFORMATION Explanation: Network Incoming (I) session was closed on the local site.

Code	Description
140	REQUEST QQQNNNNN RETRY WITH PARTNER PPPPPPPP Message Type: INFORMATION Explanation: Restart of request QQQNNNNN is in progress.
141	REQUEST QQQNNNNN POSTPONED WITH PARTNER PPPPPPPP Message Type: INFORMATION Explanation: Request is queued in differed transfer list.
142	REQUEST QQQNNNNN FFFFFFFF ERROR DURING SELECTION TRC=.... PRC=.... Message Type: ERROR Explanation: Transfer was refused by the STRF transfer process during the selection phase. Action: Check the appropriate computer for the cause of the error based on the TRC.
143	COMMUNICATION NOT OBTAINED -> PPPPPPPP REQ: QQQNNNNN RETRY IN: TT MINUTES Message Type: INFORMATION Explanation: The connection attempt with the specified partner failed. Request is queued in the automatic retry transfer list.
144	REQUEST QQQNNNNN SESSION ERROR ..... NRC=.... .... Message Type: ERROR Explanation: An error was detected by the Network Interface. The command name shows which access failed. Network Return Code (NRC) has 2 parts: the first part displays the return code from the system, the second part contains 2 fields of 1 byte displayed in hexadecimal showing the Cause and Diagnostic (for X25). Action: Make corrections based on the values in the NRC. If the transfer does not automatically restart, use the Operator or Application Interface to restart it.
145	REQUEST QQQNNNNN REJECTED <- PPPPPPPP SRC=.... TRC=.... PRC=.... Message Type: ERROR Explanation: Transfer was rejected. Action: Correct the problem based on the TRC value.
146	REQUEST QQQNNNNN ABORT <- PPPPPPPP SRC=.... TRC=.... PRC=.... Message Type: ERROR Explanation: Transfer was aborted by the partner. Action: Check the remote computer for the cause.
147	REQUEST QQQNNNNN ABORT -> PPPPPPPP SRC=.... TRC=.... PRC=.... Message Type: ERROR Explanation: Transfer was aborted by the local site. Action: Correct the problem based on the TRC value.
148	COMMUNICATION ABORTED WITH PPPPPPPP REQ: QQQNNNNN -> PARTNER HELD Message Type: ERROR Explanation: Maximum number of connection tries was reached. The request has an 'O' state, but can be restarted manually. Partner is held. Action: Check call parameters, make another request, and enable the partner.
149	REQUEST QQQNNNNN <- UUUUUUUU Message Type: INFORMATION Explanation: Displays the user who made this request.
150	REQUEST QQQNNNNN FFFFFFFF TRANSFER RESTARTING STRF SSSSSSSSSS Message Type: INFORMATION Explanation: This message is equivalent to the first data transmission (or reception), in a restart session.

Code	Description
151	COMMUNICATION ABORTED WITH PPPPPPPP REQ: QQQNNNNN Message Type: ERROR Explanation: Maximum number of connection tries was reached. The request has an 'O' state, but can be restarted manually. Action: Check call parameters and make another request.
152	REQUEST XXXXXXXX FFFFFFFF FORWARD PROCEEDING ... Message Type: INFORMATION Explanation: Sterling Connect:Express initiated the automatic forward process from the partner alias field **name**.
153	REQUEST XXXXXXXX FORWARDING -> PPPPPPPP. FILE FFFFFFFF Message Type: INFORMATION Explanation: Sterling Connect:Express is executing the automatic forward process from the partner alias field **ROUT**.
201	PARTNER: PPPPPPPP ADDED BY UUUUUUUU Message Type: INFORMATION Explanation: Symbolic name was created by the specified user.
202	PARTNER: PPPPPPPP UPDATED BY UUUUUUUU Message Type: INFORMATION Explanation: Symbolic name was updated by the specified user.
203	PARTNER: PPPPPPPP DELETED BY UUUUUUUU Message Type: INFORMATION Explanation: Symbolic name was deleted by the specified user.
211	FILE : FFFFFFFF ADDED BY UUUUUUUU Message Type: INFORMATION Explanation: Symbolic name was created by the specified user.
212	FILE : FFFFFFFF UPDATED BY UUUUUUUU Message Type: INFORMATION Explanation: Symbolic name was updated by the specified user.
213	FILE : FFFFFFFF DELETED BY UUUUUUUU Message Type: INFORMATION Explanation: Symbolic name was deleted by the specified user.
222	SESSION TABLE #. UPDATED BY UUUUUUUU Message Type: INFORMATION Explanation: Session table #x was updated by the specified user.
223	PRESENTATION TABLE #. UPDATED BY UUUUUUUU Message Type: INFORMATION Explanation: Presentation table #x was updated by the specified user.
224	REQUEST XXXXXXXX NOTIFICATION START TRANSFER Message Type: INFORMATION Explanation: The end transfer notification is processed for request XXXXXXXX.
225	REQUEST XXXXXXXX NOTIFICATION END TRANSFER Message Type: INFORMATION Explanation: The start transfer notification is processed for request XXXXXXXX.
226	REQUEST XXXXXXXX NOTIFICATION TRANSFER ERROR Message Type: INFORMATION Explanation: The transfer error notification is processed for request XXXXXXXX.

Code	Description
227	SSLPARM : FFFFFFFF ADDED BY UUUUUUUUR Message Type: INFORMATION Explanation: The sslparm entry FFFFFFFF has been added by user UUUUUUUU.
228	SSLPARM : FFFFFFFF UPDATED BY UUUUUUUUR Message Type: INFORMATION Explanation: The sslparm entry FFFFFFFF has been updated by user UUUUUUUU.
229	SSLPARM : FFFFFFFF DELETED BY UUUUUUUUR Message Type: INFORMATION Explanation: The sslparm entry FFFFFFFF has been deleted by user UUUUUUUU.
230	CERT : FFFFFFFF ADDED BY UUUUUUUU Message Type: INFORMATION Explanation: The certificate entry FFFFFFFF has been added by user UUUUUUUU.
231	CERT : FFFFFFFF UPDATED BY UUUUUUUU Message Type: INFORMATION Explanation: The certificate entry FFFFFFFF has been updated by user UUUUUUUU.
232	CERT : FFFFFFFF DELETED BY UUUUUUUU Message Type: INFORMATION Explanation: The certificate entry FFFFFFFF has been deleted by user UUUUUUUU.
233	SSL APM (SSSSSSSS) CREATION ERROR (tttt) Message Type: ERROR Explanation: The SSL server SSSSSSSS could not initialize, trc code = tttt. Action: Check the ssl server definition and associated certificate. You can disable it .
234	CERT FFFFFFFF EXPIRED OR NOT YET VALID Message Type: WARNING Explanation: While importing the certificate FFFFFFFF Sterling Connect:Express detected that it is not valid. Action: Import a new certificat .
235	SSL WARNING: (SSSSSSSS) CIPHER LIST READ ERROR. USING DEFAULT Message Type: WARNING Explanation: While initializing ssl server SSSSSSSS, Sterling Connect:Express could not access the cipher list. Action: Check available cipher lists or create one as required. Stop/restart Sterling Connect:Express
236	SSL WARNING: (SSSSSSSS) CA LIST MEMORY ALLOCATION ERROR. NO CA LIST USED Message Type: WARNING Explanation: While initializing ssl server SSSSSSSS, Sterling Connect:Express could not load the CA list. All CA certificates imported in the data base are used Action: Check system resources.
237	SSL WARNING: (SSSSSSSS) CA LIST READ ERROR. NO CA LIST USED Message Type: WARNING Explanation: While initializing ssl server SSSSSSSS, Sterling Connect:Express could not read the CA list.All CA certificates imported in the data base are used Action: Check available CA lists or create one as required. Stop/restart Sterling Connect:Express.
238	CTREE OPEN ERROR. REBUILDING ... Message Type: WARNING Explanation: While initializing , Sterling Connect:Express could not open the database. A rebuild is performed in order to fix the problem.

Code	Description
239	CTREE FATAL OPEN ERROR AFTER REBUILD Message Type: ERROR Explanation: While initializing , Sterling Connect:Express could not open the database. The database has probably been corrupted Action: restore the database from backup. it is advised to backup regularly the database. If this problem is occuring after an upgrade, check the database.p file in $TOM_DIR/config.Contact technical support if this problem persists.
241	ERROR OPENING HTTP NOTIFICATION FILES Message Type: NOTICE Explanation: While initializing , Sterling Connect:Express could not open the http notification files. Action: check the $TOM_DIR/ntfo directory. Contact technical support if this problem persists.
242	ERROR WRITING TOM_HTTPN QUEUE Message Type: NOTICE Explanation: Sterling Connect:Express could not access to the httpn server. Action: start the httpn server using $start_httpn.
300	TOM TERMINATION COMPLETE Message Type: INFORMATION Explanation: Sterling Connect:Express termination was requested and completed by the specified user.
301	ABNORMAL C:E TERMINATION - SIGNAL XX Message Type: ERROR Explanation: Sterling Connect:Express termination occurred due to invalid execution. Signal ss must be kept. Action: Contact technical support if this problem persists.
302	REQUEST XXXXXXXX PROCESSUS ABNORMALLY ENDED STRF PPPPPPPPPP Message Type: ERROR Explanation: The execution of the request XXXXXXXX abended due to abnormal end of strf process PPPPPPPP. Action: Restart the request if this is a local request or contact the remote partner
303	REQUEST XXXXXXXX FFFFFFFF <- PPPPPPPP MESSAGE ACCEPTED STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the request XXXXXXXX has been accepted to receive a message from partner PPPPPPPP, using file name FFFFFFFF.
304	REQUEST XXXXXXXX FFFFFFFF <- PPPPPPPP MESSAGE RECEIVED STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the request XXXXXXXX has been completed, receiving a message from partner PPPPPPPP, using file name FFFFFFFF.
305	REQUEST XXXXXXXX FFFFFFFF -> PPPPPPPP MESSAGE SENT STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the request XXXXXXXX has been completed, sending a message to partner PPPPPPPP, using file name FFFFFFFF.
306	REQUEST XXXXXXXX FFFFFFFF -- PPPPPPPP EERP OOOOOOOO DDDDDDDD TTTTTTTT Message Type: INFORMATION Explanation: the request XXXXXXXX is running, sending/receiving an eerp to/from partner PPPPPPPP, file name FFFFFFFF, origine OOOOOOOO (pi3), Destination DDDDDDDD (pi4) , transfer identification TTTTTTTT (pi13).
307	REQUEST XXXXXXXX FFFFFFFF <- PPPPPPPP EERP RECEIVED STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the request XXXXXXXX is completed, receiving an eerp from partner PPPPPPPP, file name FFFFFFFF, by strf process PPPPPPPP.

Code	Description
308	REQUEST XXXXXXXX FFFFFFFF -> PPPPPPPP EERP SENT STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the request XXXXXXXX is completed, sending an eerp to partner PPPPPPPP, file name FFFFFFFF, by strf process PPPPPPPP.
309	REQUEST XXXXXXXX REJECTED <- PPPPPPPP (M) SRC=.... TRC=.... PRC=.... SSLRC=..... Message Type: ERROR Explanation: the inbound request XXXXXXXX has been rejected, receiving a message from partner PPPPPPPP, Action: check return codes and contact remote partner.
310	REQUEST XXXXXXXX REJECTED -> PPPPPPPP (M) SRC=.... TRC=.... PRC=.... SSLRC=..... Message Type: ERROR Explanation: the outbound request XXXXXXXX has been rejected, sending a message to partner PPPPPPPP, Action: check return codes and contact remote partner.
311	REQUEST XXXXXXXX (D) user data. Message Type: INFORMATION Explanation: the request XXXXXXXX, direction D, was associated the user data shown in the message (50 first characters).
312	CONTROL-DN : CCCCCCCC UPDATED BY UUUUUUUU Message Type: INFORMATION Explanation: the control DN profile CCCCCCCC has been updated by user UUUUUUUU
313	CONTROL-DN : CCCCCCCC ADDED BY UUUUUUUU Message Type: INFORMATION Explanation: the control DN profile CCCCCCCC has been added by user UUUUUUUU
314	CONTROL-DN : CCCCCCCC DELETED BY UUUUUUUU Message Type: INFORMATION Explanation: the control DN profile CCCCCCCC has been deleted by user UUUUUUUU
315	REQUEST XXXXXXXX REJECTED PPPPPPPP/CCCCCCCC TRC=xxxx Message Type: ERROR Explanation: the request XXXXXXXX with partner PPPPPPPP has been rejected due to control dn using profile CCCCCCCC. The trc code xxxx indicates the type of error. Action: check trc and the control dn profile CCCCCCCC contact the remote server or client. Activate the trace for further analysis
316	REQUEST XXXXXXXX FFFFFFFF -- PPPPPPPP FORWARD OOOOOOOO DDDDDDDD TTTTTTTT Message Type: INFORMATION Explanation: the request XXXXXXXX is running, forwarding to/from partner PPPPPPPP, file name FFFFFFFF, origine OOOOOOOO (pi3), Destination DDDDDDDD (pi4) , transfer identification TTTTTTTT (pi13).
317	REQUEST XXXXXXXX PROCESSING COMMAND CCCCCCCCCCCC Message Type: INFORMATION Explanation: the user command CCCCCCCCCCCC has been executed for the request XXXXXXXX.
318	REQUEST XXXXXXXX ERROR COMMAND CCCCCCCCCCCC (ERRNO=NNNN) Message Type: ERROR Explanation: an error occurred with the execution of the user transfer command CCCCCCCCCCCC for the request XXXXXXXX. System error code is NNNN, Action: check the system error code for more information.

Code	Description
319	REQUEST XXXXXXXX PROCESSING EXIT EEEEEEEEEEE Message Type: INFORMATION Explanation: the user exit EEEEEEEEEEE has been executed for the request XXXXXXXX.
320	REQUEST XXXXXXXX ERROR EXIT EEEEEEEEEEE Message Type: ERROR Explanation: an error occurred with the execution of the user transfer exit EEEEEEEEEEE for the request XXXXXXXX, Action: check the return codes in the request details for more information.
321	TCPIP SERVER STARTED - STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the non secured PeSIT server has been successfully started with process ID PPPPPPPPPP.
322	TCPIP SERVER NOT STARTED - (ERRNO=NNNN) Message Type: ERROR Explanation: the non secured PeSIT server failed to start with the system error code NNNN, Action: check the system error code for more information.
323	TCPIP SERVER STOPPED - STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the non secured PeSIT server with process ID PPPPPPPPPP has been successfully stopped.
323	TCPIP SERVER ABNORMALLY STOPPED - STRF PPPPPPPPPP - ERRNO=NNNN Message Type: ERROR Explanation: the non secured PeSIT server with process ID PPPPPPPPPP stopped unexpectedly with the system error code NNNN. Action: check the system error code for more information.
324	TCPIP SERVER NOT STOPPED - STRF PPPPPPPPPP (ERRNO=NNNN) Message Type: ERROR Explanation: the non secured PeSIT server with process ID PPPPPPPPPP failed to stop with the system error code NNNN, Action: check the system error code for more information.
325	TCPIP SERVER DIED UNEXPECTEDLY - STRF PPPPPPPPPP Message Type: ERROR Explanation: the non secured PeSIT server with process ID PPPPPPPPPP has been stopped unexpectedly. Action: if the problem persists contact the Sterling Connect:Express support team.
331	SSL SERVER STARTED - STRF PPPPPPPPPP Message Type: INFORMATION Explanation: a secured PeSIT server has been successfully started with process ID PPPPPPPPPP.
332	SSL SERVER NOT STARTED - (ERRNO=NNNN) Message Type: ERROR Explanation: a non secured PeSIT server failed to start with the system error code NNNN, Action: check the system error code for more information.
333	SSL SERVER STOPPED - STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the secured PeSIT server with process ID PPPPPPPPPP has been successfully stopped.
333	SSL SERVER ABNORMALLY STOPPED - STRF PPPPPPPPPP - ERRNO=NNNN Message Type: ERROR Explanation: the secured PeSIT server with process ID PPPPPPPPPP stopped unexpectedly with the systeme error code NNNN. Action: check the system error code for more information.

Code	Description
334	SSL SERVER NOT STOPPED - STRF PPPPPPPPPP (ERRNO=NNNN) Message Type: ERROR Explanation: the secured PeSIT server with process ID PPPPPPPPPP failed to stop with the system error code NNNN, Action: check the system error code for more information.
335	SSL SERVER DIED UNEXPECTEDLY - STRF PPPPPPPPPP Message Type: ERROR Explanation: the secured PeSIT server with process ID PPPPPPPPPP has been stopped unexpectedly. Action: if the problem persists contact the Sterling Connect:Express support team.
341	FTP SERVER STARTED - STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the FTP server has been successfully started with process ID PPPPPPPPPP.
342	FTP SERVER NOT STARTED - (ERRNO=NNNN) Message Type: ERROR Explanation: the FTP server failed to start with the system error code NNNN, Action: check the system error code for more information.
343	FTP SERVER STOPPED - STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the FTP server with process ID PPPPPPPPPP has been successfully stopped.
344	FTP SERVER NOT STOPPED - STRF PPPPPPPPPP (ERRNO=NNNN) Message Type: ERROR Explanation: the FTP server with process ID PPPPPPPPPP failed to stop with the system error code NNNN, Action: check the system error code for more information.
345	FTP SERVER DIED UNEXPECTEDLY - STRF PPPPPPPPPP Message Type: ERROR Explanation: the FTP server with process ID PPPPPPPPPP has been stopped unexpectedly. Action: if the problem persists contact the Sterling Connect:Express support team.
351	API SERVER STARTED - STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the API server has been successfully started with process ID PPPPPPPPPP.
352	API SERVER NOT STARTED - (ERRNO=NNNN) Message Type: ERROR Explanation: the API server failed to start with the system error code NNNN, Action: check the system error code for more information.
353	API SERVER STOPPED - STRF PPPPPPPPPP Message Type: INFORMATION Explanation: the API server with process ID PPPPPPPPPP has been successfully stopped.
354	API SERVER NOT STOPPED - STRF PPPPPPPPPP (ERRNO=NNNN) Message Type: ERROR Explanation: the API server with process ID PPPPPPPPPP failed to stop with the system error code NNNN, Action: check the system error code for more information.
355	API SERVER DIED UNEXPECTEDLY - STRF PPPPPPPPPP Message Type: ERROR Explanation: the API server with process ID PPPPPPPPPP has been stopped unexpectedly. Action: if the problem persists contact the Sterling Connect:Express support team.

Code	Description
401	CTREE OPEN ERROR XXXXX WITH FILE FFFFFFFF Message Type: ERROR Explanation: opening the Sterling Connect:Express database file FFFFFFFF failed with error XXXXX. Action: try to perform a manual rebuild of the database and try again. If the problem persists contact the Sterling Connect:Express support team.
402	CTREE OPEN ERROR. TRYING A REBUILD... Message Type: INFORMATION Explanation: a rebuild of the Sterling Connect:Express database is performed after a try to open the database failed.
403	CTREE FATAL OPEN ERROR AFTER COMPACTION Message Type: ERROR Explanation: opening the Sterling Connect:Express database failed after it has been rebuilt. Action: contact the Sterling Connect:Express support team.
405	CTREE CLOSE ERROR XXXXX WITH FILE FFFFFFFF Message Type: ERROR Explanation: closing the Sterling Connect:Express database file FFFFFFFF failed with error XXXXX. Action: if the problem persist after after a rebuild of the database, contact the Sterling Connect:Express support team.
406	CTREE CREATE ERROR XXXXX WITH FILE FFFFFFFF Message Type: ERROR Explanation: creating the Sterling Connect:Express database file FFFFFFFF failed with error XXXXX. Action: if the problem persist contact the Sterling Connect:Express support team.
407	CTREE COMPACT ERROR XXXXX WITH FILE FFFFFFFF Message Type: ERROR Explanation: compacting the Sterling Connect:Express database file FFFFFFFF failed with error XXXXX. Action: if the problem persist contact the Sterling Connect:Express support team.
408	CTREE REBUILD ERROR XXXXX WITH FILE FFFFFFFF Message Type: ERROR Explanation: rebuilding the Sterling Connect:Express database file FFFFFFFF failed with error XXXXX. Action: if the problem persist contact the Sterling Connect:Express support team.
409	CTREE FILE FFFFFFFF CREATED Message Type: INFORMATION Explanation: the Sterling Connect:Express database file FFFFFFFF has been successfully created.
410	CTREE FILE FFFFFFFF REBUILT Message Type: INFORMATION Explanation: the Sterling Connect:Express database file FFFFFFFF has been successfully rebuilt.
501	ERROR ASSIGNING MESSAGE QUEUE QQQQQQQQQQ (ERRNO=NNNN) Message Type: ERROR Explanation: the monitor has not been able to connect to the message queue QQQQQQQQQQ. Action: it happens that a process finishes before the monitor send the last message.
502	ERROR SENDING MESSAGE IN QUEUE ..... TRC=.... SRC=.... Message Type: ERROR Explanation: the monitor has not been able to send a message to the message queue QQQQQQQQQQ. Action: if the problem persist contact the Sterling Connect:Express support team.
	ERROR ON FILE (FFFFFFFFFF) SRC=.... TRC=.... PRC=.... Message Type: ERROR Explanation: A severe error occurred with the file FFFFFFFFFF. Action: Check the return codes for more information.

LOG File Rotation

The maximum number of recors in the LOG file is controlled by the SIZLOG parameter of the SYSIN file. When this number is reached the current LOG file is archived. Archiving the LOG file consist in copying it to a file named LOG.0 and reinitializing its content with empty records. If the archive file named LOG.0 already exists then LOG.0 is first archived. Archiving LOG.0 consist in copying and compress its content to a file named LOG.1.gz. This process repeats until MAXLOG archives of the LOG file have been created. When the number of archives has been reached, the content of the older one is simply discarded.

Changing the Size of the LOG File

The LOG file is created when Sterling Connect:Express starts and if it does not exists. The SIZLOG parameter of the SYSIN file is used to determine the number of records the LOG file must contain. Changing the value of the SIZLOG parameter has not effect after a monitor restart unless the LOG file is deleted.

Another way to change the number of records in the LOG file is to use the *$TOM_DIR/config/ch_conf.sh* script. The current LOG file will be rotated ans a new LOG file will be created with the specified number of records.

Appendix B

Return Codes

This appendix lists protocol return codes and Sterling Connect:Express return codes.

Protocol Return Codes

The following protocol error codes are diagnostic codes which appear in error messages in the Sterling Connect:Express log file as PRC. The codes have the format *x xyz* where x represents the severity of the error. For example, PRC 201 is displayed as 2 201.

Code	Description
100	Transmission error.
200	Insufficient file characteristics.
201	System resources temporarily insufficient.
202	User resources temporarily insufficient.
203	Non-priority transfer.
204	File already exists.
205	File not found.
206	Disk quota will be exceeded if the file is received.
207	File occupied.
208	File too old.
209	Message of this type not accepted.
210	Failure of presentation context negotiation.
211	Cannot open file.
212	Cannot routinely close file.

Code	Description
213	Input/output error.
214	Failure of restart point negotiation.
215	System-specific error.
216	Voluntarily premature stop.
217	Too many synchronization points without acknowledgments.
218	Re-synchronization impossible.
219	File space used up.
220	Incorrect record length.
221	End of transmission expiration time.
222	Too much data without synchronization points.
223	Abnormal end-of-transfer.
224	File size larger than expected.
225	Application congested; file deleted.
226	Transfer refused.
233	No transfer restart context available.
299	Miscellaneous.
300	Congested local communication system.
301	Identification of caller Partner unknown.
302	Unauthorized caller Partner.
303	Caller Partner unknown.
304	Identification of called Partner unauthorized.
305	Failure of a SELECT negotiation.
306	Failure of a RESYNC negotiation.
307	Failure of SYNC negotiation.
308	Version number not supported.
309	Too many connections already in progress.
310	Network incident.
311	Remote protocol error code.
312	Closure of service requested by user.
314	Unused connection cut off.
315	Failure of negotiation.
317	Time-out failure.

Code	Description
318	Parameter absent or incorrect value.
319	Number of bytes or records incorrect.
320	Maximum number of re-synchronization reached.
321	Create queue refused.
322	Problem during initialization of strf.
399	Miscellaneous.
817	Time out detected after file transfer was completed.

Sterling Connect:Express Return Codes

TRC codes appear in all Sterling Connect:Express modules. The following table describes Sterling Connect:Express return codes.

Code	Description
1000	Logging rejected by partner.
1001	Logging for this partner (PRECONNECT Field invalid).
11XX	Invalid Protocol: Invalid Length for command XX.
12YY	Invalid Protocol: Unknown YY parameter.
13XX	Invalid Protocol: Invalid structure of command XX.
14YY	Invalid Protocol: Parameter YY not in the right place.
15YY	Invalid Protocol: Invalid value in parameter YY.
1501	Invalid User Parameter.
1502	Invalid Password Parameter.
1509	Invalid Port Parameter.
1511	Invalid Type Parameter.
1512	Invalid Structure Parameter.
1513	Invalid Transfer Mode Parameter.
1514	Invalid Retrieve Parameter.
1515	Invalid Store Parameter.
1516	Invalid Store Unique Parameter.
1517	Invalid Append Parameter.

Code	Description
1520	Invalid Help Parameter.
16XX	Invalid Protocol: Invalid command XX.
17XX	Invalid Protocol: Invalid header for command XX.
18YY	Invalid Protocol: Invalid length for parameter YY.
19YY	Invalid Protocol: Parameter absent or forbidden.
2008	Invalid Request Type.
2010	Invalid File name.
2011	Invalid Partner name.
2012	Invalid direction.
2013	Invalid Physical Name.
2015	Invalid Request Number.
2017	Request table is full.
2018	Invalid Allocation Rule.
2019	Communication Error.
2028	User request not authorized.
2036	Security (RACF…).
2040	Record format between file and directory do not match.
2041	Record length between file and directory do not match.
2042	Request number not found.
2043	Restart impossible.
2044	Virtual Circuit lost.
2045	Network Time-out.
2046	Inactive request.
2047	Request not interrupted.
2048	Context not found.
2049	Context not found in restart.
2050	An EERP or forward request is submitted for a request that is not yet ended.
2051	An EERP or forward request is submitted for a transmission request or an EERP request is received for a reception request.
2052	Sterling Connect:Express is called with an invalid Partner Name (Invalid Server Name).
2053	An EERP request is submitted or received for a previous EERP.
2054	No transfer in queue (HOLD).

Code	Description
2055	An EERP or forward request is submitted for a request that has already been acknowledged.
2056	Invalid direction for queued request (HOLD).
2057	The partner’s subject DN is invalid.
2058	The partner’s issuer DN is invalid.
2059	The control DN definition is not found.
2060	Partner Disabled.
2061	File Disabled.
2065	Maximum for active requests is reached.
2076	Communication with partner not obtained.
2077	Communication with partner not obtained (no more retries).
2078	Invalid link type requested.
2080	Error in Physical Name check.
2081	Allocation rule 1 (pre-allocated) but file does not exist.
2085	Allocation rule 2 (to be created) but file exists.
2086	Not enough disk space to receive file.
2087	A purge is already in progress.
2142	Partner unknown.
2143	File unknown.
2150	Invalid Protocol.
2152	RENC Read Error.
2153	RENC Write Error.
2154	RENC Update Error.
2155	RENC Delete Error.
2161	Network address received do not match network address from partners directory.
2162	RFIC Read Error.
2163	RFIC Write Error.
2164	RFIC Update Error.
2165	RFIC Delete Error.
2172	RPAR Read Error.
2173	RPAR Write Error.
2174	RPAR Update Error.
2175	RPAR Delete Error.

Code	Description
2182	RTAB Read Error.
2183	RTAB Write Error.
2184	RTAB Update Error.
2192	RPRE Read Error.
2193	RPRE Write Error.
2194	RPRE Update Error.
2201	Error on system time.
2208	Invalid password.
2211	SSL parameter profile cannot be added.
2212	SSL parameter profile cannot be read.
2213	SSL parameter profile cannot be updated.
2214	SSL parameter profile cannot be deleted.
2215	SSL parameter profile not found or has been disabled or is not a client profile.
2216	The certificate profile cannot be added.
2217	The certificate profile cannot be read.
2218	The certificate profile cannot be updated.
2219	The certificate profile cannot be deleted.
2220	The certificate file not found or is invalid.
2221	The Subject DN already exists.
2301	File mode Incompatible with Server's Options.
2302	File structure Incompatible with Server's Options.
2303	File type Incompatible with Server's Options.
2304	Phys. name rejected, file resident unique.
2306	Can't open data connection.
2308	File not found
29xx	Return code from L0B8Z20.
3001	ALLO/STOU parameters invalid.
2401	Network API error: Invalid directory
2402	Network API error: Command .. not allowed for directory .. (Internal error)
2403	Network API error: Invalid command. Client not identified (Internal error)
2404	Network API error: Error reading message queue. errno =
2405	Network API error: Error reading message queue. timeout

Code	Description
2406	Network API error: TRC =
2407	Network API error: Error writing message queue. errno =
2408	Network API error: Unable to get thread specific data (Internal error)
2409	Network API error: Invalid length received in message queue (Internal error)
2410	Network API error: Invalid command received (Internal error)
2411	Network API error: Invalid action for SITE command received (Internal error)
2412	Network API error: Invalid limit (Internal error)
2413	Network API error: Size of resulting string exceeds 4096 characters (Internal error)
2414	Network API error: Error getting certificate file
2415	Network API error: Invalid certificate file
3*03	Open Input. (* : 0 -> First transfer, 1 -> Restart of transfer).
3*04	Open Output. (* : 0 -> First transfer, 1 -> Restart of transfer).
3*05	Read Error. (* : 0 -> First transfer, 1 -> Restart of transfer).
3006	File processing error, read failed.
3*07	Write Error. (* : 0 -> First transfer, 1 -> Restart of transfer).
3*08	Open Status Error. (* : 0 -> First transfer, 1 -> Restart of transfer).
3010	Translation Error (Translation File Loading Error).
3020	Translation Error (Translation File Open or Read Error).
3092	LRECL not supported.
4091	The transfer exit returned a code different from 0. SRC contains the last 8 bits of the value of exit return code.
4093	A transfer exit changed the file format or the record length to an invalid value.
4600	Cannot read the transfer command file.
4700	Error loading Start of transfer exit.
4799	Error loading End of transfer exit.
49XX	Return Code XX from End of transfer exit (between 0 and 90).
5003	Re-synchronization point - negotiation error.
5004	Too many bytes without checkpoint.
5005	Null length for a record.
5006	Invalid number of bytes transferred - different for negotiated value.
5007	Invalid number of records transferred - different for negotiated value.
5008	The total number of bytes transferred does match the size of the file.

Code	Description
5010	Unfilled record.
5011	Carriage Return (0x0D) and/or Line Feed character (Ox0A) missing in original file.
6004	File Creation Error.
6005	Communication Error between STRF and GTRF.
6006	Cannot fork the exit.
6007	Cannot open the exit parameters file.
6008	Format of the exit parameters file is invalid.
6010	The maximum number ot retries of reading a message from the mesage queue has been reached.
6011	The delay waiting for a message in the mesage queue has expired.
6012	An error occurred while reading a message from the mesage queue.
6099	Network message size negotiated in PeSIT is more than the system configuration can support.
7*02	Synchronization File OPEN error. (* : 0 -> First transfer, 1 -> Restart of transfer).
7*03	Synchronization File READ error. (* : 0 -> First transfer, 1 -> Restart of transfer).
7*04	Synchronization File WRITE error. (* : 0 -> First transfer, 1 -> Restart of transfer).
8000	STRF Process Abnormally Ended. (* : 0 -> First transfer, 1 -> Restart of transfer).
91xx	CRC error in command XX.

Sterling Connect:Express PeSIT SSL Return Codes

Errors can occur when using SSL with the PeSIT protocol. These errors appear in the SSLRC field of the LOG file as well as in some sterm screens. A 8 digit hexadecimal number (for example: 1408A0C1) is specific to OpenSSL. You can use the command $TOM_DIR/config/sslerr to display a label of this SSL error. Errors starting with letters are specific to Sterling Connect:Express and are described in the following table:

Code	Description
CERT0001	The certificate file defined in the Certificate Definition cannot be found.
CERT0002	Allocation error when loading the certificate in memory.

Code	Description
CERT0003	The certificate file defined in the Certificate Definition cannot be read.
CERT0004	The certificate file defined in the Certificate Definition cannot be decoded.
CERT0005	The before field of the certificate is invalid.
CERT0006	The certificate is not valid yet.
CERT0007	The after field of the certificate is invalid.
CERT0008	The certificate has expired.
INIT0001	The buffer used for the PeSIT message SSL encryption cannot be allocated.
RAND0001	The OpenSSL rand seed file cannot be opened or read.

Appendix C

User Commands and Exits

This appendix provides an overview of user commands, the standard error command, and user exits.

Overview

User commands can be started by the GTRF monitor at the beginning or end of a transfer and are independent of that transfer (asynchronous mode). This means that the request does not wait for the command to finish.

User commands are available for all type of transfers, files, messages and EERP's.

A command is declared as a 12-character name in the file directory in the beginning and end of transfer command fields. It is a Shell procedure file written in the exit directory. The beginning of transfer command is processed before the start of file selection and the end of transfer command is processed after file selection ends. If the transfer is interrupted, the end of transfer command is not executed. When the transfer restarts, the beginning of transfer command is not re-executed.

If the STRACE flag is enabled in the SYSIN file, processing of the command is redirected to the output file Cx_QQQNNNNN in the exit directory. x is replaced by I (Initiator), E (End) or F (Failed., and QQQNNNNN is the request number.

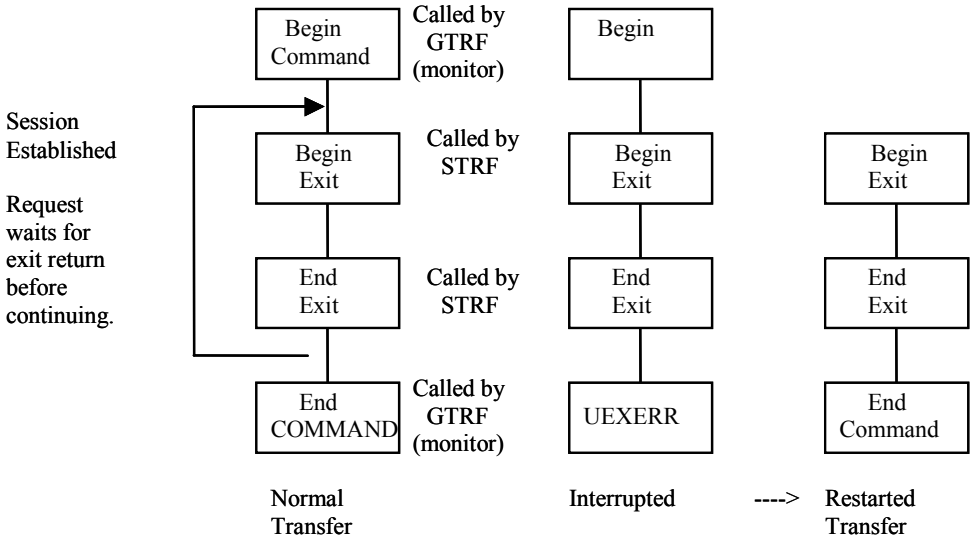
When a transfer is interrupted, a call is made to a special command called UEXERR. This is a dummy call that you can use to code special error handling processing, such as requesting information about this request or purging the request.

User exits can be started by the STRF process at the beginning or end of a transfer and are synchronized with the transfer. The request waits for the exit to finish. This cannot be longer than one minute or the protocol will close the session. An exit is declared as a 12-character name in the file directory in the begining and end of transfer exit fields. It is an executable file written in the exit directory. If the transfer is interrupted, the end of transfer exit is executed. Then when the transfer restarts, the beginning of transfer exit is executed again.

User exits are available for all type of transfers, files, messages and EERP's.

If the STRACE flag is enabled in the SYSIN file, processing of the exit is redirected to the output file Ex_QQQNNNNN in the exit directory. x is replaced by I (Initiator), E (End) or F (Failed), and QQQNNNNN is request number.

The following model illustrates the processing of normal, interrupted, and restarted transfers with user commands and exits.



User Commands

To create start and end of transfer events, a user command must be configured in the files directory and the command must be in the directory exit. The Sterling Connect:Express GTRF process sends nine parameters to the command. See the TRFOK sample in the exit directory. The nine parameters are listed below.

1. Request number of transfer
2. Symbolic file name
3. Symbolic partner name
4. Physical file name (absolute path)
5. Direction of transfer
6. System Return Code (SRC)
7. Sterling Connect:Express Return Code (TRC)
8. Protocol Return Code (PRC)
9. Received Pi99
10. Sent Pi99
11. Transfer Origin
12. Transfer Destination
13. Local Name
14. Label
15. User sender (Pi61)
16. User receiver (Pi62)
17. Request start date

18. Request start time
19. Transfer status
20. Julian date
21. Number of records transferred
22. Number of kilobytes transferred
23. Request end date
24. Request end time
25. Request type
26. File type (pi11)
27. Transfer Id (pi13)
28. File date (pi51)

Standard Error Command (UEXERR)

When requesting a Sterling Connect:Express transfer, the user receives feedback about the request, but not about the result. The standard Shell procedure UEXERR can identify an error that occurred during transfer operations, and Sterling Connect:Express calls UEXERR any time an error occurs. UEXERR is a dummy call that users can modify for their own site-specific error handling procedures. See the UEXERR file in the exit directory. The parameters are the same as for commands.

Standard Signal Error Command (UEXSIG)

In case the monitor received a Signal error and stopped, the UEXSIG command is launched. The default provided gets system information and saves it in /tmp/tom_mon_died.$(date +"%Y%m%d%H%M%S")". You can modify this command with whatever process you want.

Exits

The user can access a number of protocol parameters before the beginning-of-transfer and end-of-transfer. This user-defined exit must be defined in the files directory, and must also be in the exit directory.

STRF runs the exit and sends one parameter, the name of a temporary file. A log file in the exit directory, Ex_QQQNNNN is created if the STRACE flag in the SYSIN file is set to 1. x is replaced by I (Initiator), E (End) or F (Failed), and QQQNNNNN is request number.

Some values can be changed by an exit. Refer to the structure d1b8ruex in exit/d1b8ruex.h. The following table shows the relationship between the PeSIT parameters and Sterling Connect:Express parameters.

Protocol Parameter	Sterling Connect:Express Parameter
pi37	Identified as label[80] in the structure.
pi52	Identified as dhd[12] in the structure.
pi99	Identified as pi99_new[254] for release 2 or pi99_old[64] for release 1 in the structure.

Protocol Parameter	Sterling Connect:Express Parameter
pi31	Identified as recfm[2] in the structure, only when transmitting. Caution: Controls are made on this value returned by the exit. TV, TF, BF and BU are the only accepted values.
pi32	Identified as lrecl[5] in the structure, only when transmitting.
Physical file name	Identified as dsn[127] in the structure.
pi3 bis	Identified as origin[8] in the structure.
pi4 bis	Identified as destination[8] in the structure.
pi61	Identified as sender[24] in the structure.
pi62	Identified as receiver[24] in the structure.

Appendix D

Implementing Special Features

This appendix provides information about implementing SYSTCP, as well as information about translating data, implementing physical names, and implementing parameter cards file.

Implementing SYSTCP

The following screen shows how SYSTCP is implemented:

```
*TCPBYPASS
*
*       SYSTCP File for Sterling Connect:Express product
*
*       '*' to comment line.
*
* Alternate IP address, Alt. port number, Alt. host name, Partner Name.
*111.111.111.111,1234,Host,PARTNER1.
```

SYSTCP is a parameter file located in the config directory. Its format is shown in the second to last record of the SYSTCP file. A line beginning with * is a comment line.

In this file, you can enter the backup TCP/IP host name or address to contact a Partner, or the TCP/IP host name or address that a Partner uses to contact your Sterling Connect:Express monitor. The monitor verifies incoming network information to validate the connection. You can use the keyword TCPBYPASS in the first record to bypass incoming TCP address control. Complete the following steps to bypass incoming TCP address control.

1. Use the UNIX editor to modify the file.
2. Stop and start Sterling Connect:Express for the changes to take effect.

Caution: This file is case sensitive. The Partner name must be in uppercase letters followed by a dot.

Implementing a Partner List

If you regularly receive files from or transmit files to several partners, you can create a Partner list so that one transfer can reference a list of Partners.

A partner list is created as an edit text file with a 7-character physical file name in the config directory. This file can have one symbolic partner name (one to eight characters) per line. A line beginning with an asterisk (*) is a comment line and is ignored. To specify a partner list on a transfer, type the # followed by the name of the Partner list. For example, a symbolic file name called FILE has #LIST1 as the receiving partner name in its definition and #LIST2 as the transmitting partner name. The items below describe what happens with different transfer situations.

- A transfer request from an application with PARTNER1 as the transmitter is accepted if PARTNER1 is in LIST1.
- A transfer request from an application with PARTNER2 as the receiver is accepted if PARTNER2 is in LIST2.
- A transfer request from an application with no partner name specified as the transmitter is accepted and the file will be transmitted to all partners in LIST2.
- PARTNER3 calls Sterling Connect:Express to send the symbolic file named FILE. The transfer is accepted if PARTNER3 is in LIST2.

Implementing Physical Names

A physical name contains up to 128 characters and identifies the data file to transfer. It can contain environment variables as long as they are known by the GTRF monitor. This field can also contain keywords that GTRF will resolve from transfer parameters. Valid keywords depends on the partner type and the protocol used.

Remote Partner of Type Other

With a Partner type of other (Sterling Connect:Express compatible), you can use any of the following keywords:

Keyword	Description
&FILENAM	Replaced by symbolic file name
&PARTNID	Replaced by symbolic partner name
&REQTIM	Replaced by the time of the transfer request in the format HHMMSS. It must begin with H. For example: H120000.
&REQDAT	Replaced by the date of the transfer request in the format YYMMDD. It must begin with D. For example: D930321
&REQNUMB	Replaced by request number (QQQNNNNN). The first Q is replaced with A if Q = 0, or replaced by B if Q=1. For example: The request number 01900034 becomes A1900034, and the request number 11900034 becomes B1900034.
&IDT	Replaced by the contents of pi13, identifier of PeSIT protocol (length is 8 characters).

Keyword	Description
&EXTLAB	Replaced by the label of file, or the pi37 identifier of PeSIT protocol.
&PI99	Replaced by the contents of pi99, identifier of PeSIT protocol (maximum length is 44 characters).
&P99X	When used in the PHYSICAL NAME field of a file definition, the pysical file name of a the received file will be the content of the received PI99.
&ORG	Replaced by transfer origin.
&DST	Replaced by transfer destination.
&P61	Replaced by user sender (Pi61).
&P62	Replaced by user receiver (Pi62).
&QQQ	Replaced by the Julian date; for example: Q122.

Remote Partner of Type Sterling Connect:Express

With a Sterling Connect:Express Partner, you can use any keywords from the table above except &Pi99, and the &EXTDSN, &USRVAR1, and &USRVAR2 keywords.

The physical name of the transmitted file is carried from the sending Sterling Connect:Express product to the receiving Sterling Connect:Express product using the PeSIT protocol field called pi99. This name is carried in ASCII and follows the MVS/IBM physical data set name rules. The maximum length is 44 characters, the name is made of several extensions with up to 8 alphanumeric characters each and is separated by a . (dot). Each extension begins with a letter. The following screen shows an example of an MVS/IBM physical name.

```
extend01.extend02...extend0n
```

Transmission to a Remote Sterling Connect:Express

Sterling Connect:Express for UNIX will build pi99 using the physical file name of the transferred data file. To do this, it removes the first slash in the absolute path of the physical file. Then it substitutes the remaining slashes with a dot. Each extension can contain up to 8 characters, and some extensions may be truncated. For example, if the physical name of the data file is:

```
/usr/applications/appl1/filetobesent
```

The physical name carried in pi99 becomes:

```
USR.APPLICAT.APP11.FILETOBE
```

If the user or application specifies the physical name, the Sterling Connect:Express for UNIX monitor does not build pi99 and uses the user field. The user or application can specify the physical name either with the /UDF option in the application interface (p1b8preq), or with the transfer screen on the Remote Physical Name line. The physical name must be uppercase and have extensions with no more than 8 characters. Each extension must begin with a letter and be separated by a dot.

Reception of a file from a remote Sterling Connect:Express

A remote Sterling Connect:Express sender can transmit the initial physical name using the pi99 protocol field. You can use the physical name received in the pi99 field with the keywords &EXTDSN, &USRVAR1 and &USRVAR2. The following table describes each keyword.

Keyword	Description
&EXTDSN	Replaced by the full physical name from the pi99 field.
&USRVAR1	Replaced by the last extension field.
&USRVAR2	Replaced by the second to last extension field.

WARNING: In all cases, using keywords can generate a physical data file name with a length greater than 44 characters. The monitor returns TRC 2013 when this occurs.

Translating Data

This data translation function converts ASCII characters to EBCDIC in transmitting mode and EBCDIC to ASCII characters in receiving mode. The TBL01.DAT file is provided in the config directory with its editing file, TBL01.ORG.

Caution: Do not change the files TBL01.DAT and TBL01.ORG files. Create a copy of the file and edit the copy.

You must edit the new .ORG file according to its structure, and update it using the config/build_tbl program to update or create the .DAT file. For example:

1. Copy tbl01.org to tbl02.org.
2. Update the tbl02.org file.
3. Execute the build_tbl program as shown in the following example. Tables can have a number between 02 and 09.

```
build_tbl 02 tbl02.org
where 02 indicates the translation table, and tbl02 specifies the input file
```

Tables are dynamically loaded by STRF when each transfer is initialized.

Implementing a Parameter Cards File

The file FICPARAMS.dat in the config directory is used to overwrite specific parameters from the file directory based on the Partner name. This functionality enables the Sterling Connect:Express administrator to define one symbolic file name with different properties for 2 partners. The screen below shows an example.

```
#       This file contains parameter cards to overwrite file directory
#
#       '#' to comment.
#
#       Syntax (case sensitive) :
#       1st Record
#         FILENAME,PARTNER
#       2nd Record
#         TYPE STRUCTURE MODE ALLOC_FLAG ALLOC_RULE FORMAT LENGTH remotedsn
#       3rd Record
#         Empty
FILE2,PART
A R S N 0 BU 54321 file/remote

*,PART2
* * * N 0 BU 54321 file/remote

*,PART
A R S Y 0 BF 12345 test/remote

*,VAX1
A R S N 1 BF 12345 test/remote
```

In the example above, the symbolic filename FILE2 uses the following parameters with the partner PART:

```
FTP Type: Ascii                              Allocation rule is indifferent (value 0)
FTP Structure: Record                        Format is Binary Undefined (value BU)
FTP Mode: Stream                             Record length is 54321
No allocation (value N)                      Default physical name is file/remote
```

All symbolic filenames with the Parameter cards file flag enabled, and transmitted or received from partner PART2 use the following values:

```
No allocation (value N)
Allocation rule is indifferent (value 0)
Format is Binary Undefined (value BU)
Record length is 54321
Default physical name is file/remote
```

Appendix E

Implementing the PeSIT Message Functionality

This appendix provides information about sending and receiving messages instead of files. The store and forward process is described, along with the end to end acknowledgment functionality.

PeSIT Message

This section provides information on what PeSIT Message is, how to use it and how to configure it in Sterling Connect:Express for UNIX.

Overview

PeSIT Message is a protocol feature that enables to send data in one step :

```
Message = data <-> AckMessage
```

Instead of the sequence shown below:

Create <-> AckCreate

Open <-> AckOpen

Write <-> AckWrite

…….

N * Data

Sync <-> Async

…….

DataEnd

TransEnd <-> AckTransEnd

Close <-> AckClose

Deselect <-> AckDeselect.

You can use this feature to send short messages or files, and also to perform end to end acknowledgement either in a standard file transfer process or in a store and forward process. The Store and forward process is described in next section, including the end to end acknowledgment. The batch utility called p1b8pe2e is provided to send end to end acknowledgement or to forward files or messages.

Sending and Receiving Messages

This section describes PeSIT message processes.

- To send a message
- To receive a message

Sending a Message - Type of Request M

The user can send a message without data or with data, using either the “P99” - or “USER DATA” - field or a file to pass them. The type of request ‘M’ indicates that this is a message. If the “P99” field is provided, this is the data to send. If no “P99” field is provided and a “DSN” value is provided, the data is sent from the file. The parameters of the message transfer request provide a symbolic file name: if this name is defined in the directory (RFIC), the definition is used. If this name doesn’t exist, the $$MSGD$$ definition is looked for: if found, and status enabled, it is used. If $$MSGD$$ is not defined, or status disabled, the request is rejected.

The size of the message unit will be determined by the record length if it is provided in the symbolic file definition or in the request parameters, or the session message length, with a maximum of 4096 characters.

In the following the various interfaces are shown: the only parameter to consider is the type of request. All other parameters are similar to any other type of transfer request. The physical file name is not required.

STERM - Set TYPE field to ‘M’ and provide data, if needed, using PHYSICAL NAME or USER DATA.

.

```
C:X/UNIX 150 -------------- TRANSFER REQUEST ------------------------ ce01
OPTION ===>
FILE ................. : SYMBFILE        DIRECTION ............. : T (T/R)
PARTNER ............... : PARTNER.
DPCSID ALIAS .......... : MYNAME..        DPCPSW ALIAS .......... : MYPASSW.
ORIGIN................. : USER1...        DESTINATION............ : USER2
SENDER................. :                 RECEIVER .............. :
PHYSICAL NAME ......... : -.........................................

USER DATA ............. : A short message to send.....................
LABEL:.....................................................................

RECORD FORMAT ......... : ..              TF, TV, BF, BU
RECORD LENGTH ......... : 02048
TYPE/STRUCTURE/MODE FTP :                 E/A/I/*,F/R/*,B/S/*
STORE UNIQUE (FTP) .... : N               Y/N   FA :  O/N   NOT: (0-7)
TYPE .................. : M               (N/I/H/M)
TYPE OF CONNECTION .... : T               (X/P/T)
PRIORITY .............. : 0               (0/1/2)
DATE .................. : 19980728101604  (YYYYMMDDHHMMSS)

DO YOU WANT TO GO ON ?
-ENTER- NEXT FIELD               -F3- CANCEL                -F8- COMPLETION
```

The table below shows how to use interfaces:

p1b8preq	Set parameter /TYP=M and use /P99= or /DSN= to provide message data		
API Z20:	Set field	char typ[1];	/* Request type = M*/
	Provide data in	char s_pi99_254[254];	/* Sender Pi99 */
		char dsnam[127];	/* File physical name */

Receiving a Message –Saving Data

When receiving data with the Message service, Sterling Connect:Express uses the symbolic file name from the PeSIT parameter Pi12: if this name is defined in the directory (RFIC), the definition is used. If this name doesn't exist, the $$MSGD$$ definition is looked for: if found and status enabled it is used, if not found or status disabled the request is rejected.

A message can carry either data, or an end to end acknowledgment of a previous file transfer: the PeSIT parameter Pi11 indicates if this is a data Message (hexadecimal 'FFFF' is for initial message, 'FFFE' is for message acknowledgment) or an end to end acknowledgment Message (Pi11 is the same as the original CREATE parameter).

There are two possibilities for storing the data of the Message: writing it into a file or saving it into the RENC file. Sterling Connect:Express will decide where to store it from the file attributes of the file definition. If a physical file name is provided in the file definition, Sterling Connect:Express will store data in a file. Sterling

Connect:Express will place first 254 characters of data in the r_pi99_254 field of the RENC file , displayed in MESSAGE << field of STERM.

```
10/06/04 16:24:23 REQUEST 07200008 MSGFIC2 <- BOUCLE MESSAGE ACCEPTED  STRF 0000011700
10/06/04 16:24:23 REQUEST 07200008 MSGFIC2 <- BOUCLE MESSAGE RECEIVED  STRF 0000011700
10/06/04 16:24:23 REQUEST 07200008 254 first characters of the file
10/06/04 16:24:23 REQUEST 07200008 $TOM_DIR/msg/MSGFIC2_A7200008
```

If no physical file name is provided (the field must be set to ‘-‘) , the data will be considered as user data and placed in the r_pi99_254 field of the RENC file (254 characters maximum).

```
10/06/04 16:24:23 REQUEST 07200008 MSGFIC2 <- BOUCLE MESSAGE ACCEPTED  STRF 0000011700
10/06/04 16:24:23 REQUEST 07200008 MSGFIC2 <- BOUCLE MESSAGE RECEIVED  STRF 0000011700
10/06/04 16:24:23 REQUEST 07200008 254 first characters of the file
```

STERM monitoring screens show the message information.

```
C:X/UNIX 150 ----------------- MONITOR STATUS --------------------------- ce01
OPTION ===>
    REQ.NUM.  FILE      WITH      DIR. PRI. REQ. TYPE    STATE   STRF ID
    07200001  FICTEST1  EXPRESS1  T    0    N NORMAL     O       0000010408
    07200003  FICTEST1  EXPRESS1  T    0    N MESSAGE    C       0000004526
    07200005  FICTEST2  DPX1      T    0    N NORMAL     E       0000011441
    07200006  FICTST    SID1      R    0    N NORMAL     E       0000011698
    07200007  FICTEST2  DPX1      T    0    N NORMAL     E       0000011443
    07200008  MSGFIC2   SID1      R    0    N MESSAGE    E       0000011700
    07200009  FICSTSN   DPX1      T    0    N NORMAL     E       0000011445
    07200010  FIC22424  SID1      R    0    N NORMAL     E       0000011702
    07200011  FICTEST3  DPX1      T    0    N NORMAL     E       0000011447
    07200012  ARECEVOI  SID1      R    0    N NORMAL     E       0000011704
    07200013  FICTEST3  DPX1      T    0    N NORMAL     E       0000011449
    07200014  ARECEVOI  SID1      R    0    N MESSAGE    E       0000011706
    07200015  AENVOYER  DPX1      T    0    N NORMAL     O       0000011451
    07200017  FICTEST4  DPX1      T    0    N NORMAL     E       0000011456
    07200018  FICTST2   SID1      R    0    N NORMAL     E       0000012225
    07200019  FICTEST1  EXPRESS1  T    0    N NORMAL     J       0000011458


    <- -F10-    -F3- END -F7- PREVIOUS SCREEN   -F8- NEXT SCREEN    -F11- ->
```

```
C:X/UNIX 150 ----------------- MONITOR STATUS ------------------------------ ce01

  REQUEST : 02100018     FROM : PLA          DIRECTION : R     WITH : SID1
  ORIGIN  : user1        DESTINATION : user2       XFER ID : 08192009   *MESSAGE
  SENDER  :                          RECEIVER :
  USERID  : pla          STRF PID    : 0000008586  FA : N    NOT: 0       SSL: 0
  FILE    : ARECEVOI     CMD ORIGIN  : T    LPARM :           TRANSFER STATE : E
  PROTOCOL: PESIT        CRC : -    MULTI : -   TRANSLATION : 0  COMPRESSION : -
  DSNAME  : /home/tmp/ARECEVOI_D980723H124613
  MESSAGE << :254 first characters of the file



  BEGIN : 19980723 12:46:13        END : 19980723 12:46:14       RETRIES/MAX 00/00
  NRC : 0000        SRC : 0000   TRC : 0 000     PRC : 0 000     SSLRC : 00000000
  NUMBER OF RECORDS XFERED : 0000000003      FILE BYTES    :     0000025
  RECORD FORMAT ......... : TV               RECORD LENGTH ......... : 00512
  ********** TCPIP **********
  TCPIP HOST (00) :
  TCPIP ADDRESS (12) : 172.17.16.80
  PORT : 01267
```

Store and Forward – End to End Acknowledgment

This section provides information on the store and forward process (with end to end acknowledgment based on PeSIT Message), how to use it and how to configure it in Sterling Connect:Express for UNIX. First, the end to end acknowledgment is described, as a simple process, then the more complicate store and forward associated with end to end acknowledgment is described.

Overview

In the following we use the acronym “EERP” for “End to End ResPonse”. The end to end response acknowledges that a file (or a message) has been received by the destination application. This can be a simple acknowledgment from receiver to sender, or a store and forward acknowledgment, from final destination to initial origin.

End to End Acknowledgment - EERP

This section describes the different steps of the end to end process. Next section shows how it can be integrated into the store and forward mechanics.

Step one	At end of reception, save end to end context, including the transfer id
Step two	Retreive end to end context and submit an EERP transmission request with it
Step three	Receive the end to end acknowledgment and take appropriate action.

The type of request is provided in the parameter list of the user command ($25), to enable user to take actions specific to a file (Type=N or I), a message (Type=M), or an EERP (Type=E). All parameters required to identify a transfer , the end to end context, are provided in the parameter list of the command. Parameters required to set up the store and forward process (for example to save the EERP context for further acknowledgment) are listed below.

Saving Parameters

Step one is normally done through the RENC file, where all end to end parameters are saved. The end of transfer command enables you to save these parameters, or to use them on line. The table below shows the relationship between Sterling Connect:Express parameters, PeSIT parameters, RENC file fields, the normal transfer request parameters and user command fields.:

Parameter saved	PeSIT-Pi	RENC-trfpar	Normal Xfer request	User command
Partner Identification	3	pi.ident	SPN=	$3
Alias	4	pi.idser	SID=	$13

EERP Context

Parameter saved	PeSIT-Pi	RENC-trfpar	Normal Xfer request	User command
File identification	3bis 4bis 11 12	pi.user_org pi_user_dst pi.tyf pi.nof	ORG= DST= SFN=	$11 $12 $26 $27
Transfer identification	13	pi.idt		$28
File Date-Time	51	pi.dhc		$28
Sender identification	61	pi.user_snd	SND=	$15
Receiver identification	62	pi.user_rcv	RCV=	$16

The EERP transfer process must use access to RENC to build the Message data unit, unless these parameters are provided directly to the end to end utility called p1b8pe2e, that is described in Chapter 4, “Transfers” .

Retreiving Parameters

Step two must build the EERP transfer request parameter list with information required to retreive the end to end context, and an optionnal user message to associate with the context. There are two possibilities: to give the request number or to provide all parameters.

- Giving the request number and an optionnal user message – if the request is ended and recorded in the RENC file.

Information expected	Field	Description	PeSIT parameter
Request Number	REQ=	The request (local) to aknowledge	N/A
Partner Identification	SPN=	Where to send it (default = partner)	Pi3 (Connect)

Local Identification	SID=	My name (default = alias)	Pi4 (Connect)
Notification	NTF=	0-7	
Priority	PRT=	0-2	
Link Type	LNK=		
Scheduling date	DAT=		
User message	ACK=	Provides feedback, lg <= 254 characters	pi91

❖ Giving the end to end parameters and the user message.

Information expected	Field	Description	PeSIT parameter
Partner Identification	SPN=	Where to send it (default = partner)	Pi3 (Connect)
Local Identification	SID=	My name (default = alias)	Pi4 (Connect)
Notification	NTF=	0-7	
Priority	PRT=	0-2	
Link Type	LNK=		
Scheduling date	DAT=		
User message	ACK=	Provides feedback, lg <= 254 characters	pi91

EERP Context.

Information expected	Field	Description	PeSIT parameter
File Identification	ORG= DST= P11= P12=	L <= 24 characters L <= 24 characters L = 2 hexadecimal L <= 14 characters	Pi3bis Pi4bis Pi11 Pi12
Transfer Identification	P13=	L <= 8 numeric	Pi13
File date-Time	P51=	L = 12	Pi51
Sender identification	P61=	L <= 24 characters	Pi61
Receiver identification	P62=	L <= 24 characters	Pi62

Sending End to End Response

To send the acknowledgment, the user must submit an EERP transfer request to Sterling Connect:Express , using the batch utility p1b8pe2e, a program, or the operator interface STERM. Sterling Connect:Express builds the EERP message from the EERP context, either from the parameters provided, or accessing to the RENC file. The initial request must be a reception, a file or a message, with status ended = ‘E’. TRC=2050, 2051, 2053 or 2055 is issued if the request is in the RENC file and it does not meet the conditions.

The EERP process doesn't require a file definition to execute: if symbolic file $$EERP$$ is defined, and status enabled, the process will be executed according to this profile, in any case: for example, exits, commands, physical file name attached to this profile are used. If the file provided in the request is defined and no $$EERP$$ definition exists, or status is disabled, the transfer will be executed according to the file of the request.

Upon reception of an EERP, Sterling Connect:Express searches for the corresponding request. It must be a transmission, of a file or a message, and it must be Ended. TRC code 2050 or 2055 is issued if the request is found and doesn't match these conditions. If the request doesn't exist, the EERP is accepted.

The status of the request in the RENC file is changed from 'E' to 'X' when the corresponding EERP transfer is successfully completed.

Using STERM

You can submit an EERP request without feedback, using STERM. The EERP is built from information retreived in the RENC file. No feedback information is provided is the PeSIT message. in the figure below the user is submitting an EERP for request 07200013.

```
C:X/UNIX 150 ----------------- MONITOR STATUS --------------------------- ce01
OPTION ===>
    REQ.NUM.  FILE      WITH      DIR. PRI. REQ. TYPE    STATE   STRF ID
    07200001  FICTEST1  EXPRESS1  T    0    N NORMAL     O       0000010408
    07200003  FICTEST1  EXPRESS1  T    0    N MESSAGE    C       0000004526
    07200005  FICTEST2  DPX1      T    0    N NORMAL     E       0000011441
    07200006  FICTST    SID1      R    0    N NORMAL     E       0000011698
    07200007  FICTEST2  DPX1      T    0    N NORMAL     E       0000011443
    07200008  MSGFIC2   SID1      R    0    N MESSAGE    E       0000011700
    07200009  FICSTSN   DPX1      T    0    N NORMAL     E       0000011445
    07200010  FIC22424  SID1      R    0    N NORMAL     E       0000011702
    07200011  FICTEST3  DPX1      T    0    N NORMAL     E       0000011447
    07200012  ARECEVOI  SID1      R    0    N NORMAL     E       0000011704
E   07200013  FICTEST3  DPX1      R    0    N NORMAL     E       0000011449
    07200014  ARECEVOI  SID1      R    0    N MESSAGE    E       0000011706
    07200015  AENVOYER  DPX1      T    0    N NORMAL     O       0000011451
    07200017  FICTEST4  DPX1      T    0    N NORMAL     E       0000011456
    07200018  FICTST2   SID1      R    0    N NORMAL     E       0000012225
    07200019  FICTEST1  EXPRESS1  T    0    N NORMAL     J       0000011458


    <- -F10-    -F3- END -F7- PREVIOUS SCREEN   -F8- NEXT SCREEN    -F11- ->
```

Using P1b8pe2e Utility

If you want to send a feedback message with the EERP, use the p1b8pe2e utility, with parameter /ACK=, or /DSN= if you want to place the feedback in a file.

FUN=E for 'send EERP', REQ='request number', ACK='feedback message': this will retreive information from the RENC file, and associate a feedback.:

```
p1b8pe2e  "/FUN=E/SPN=ident/REQ=xxxxxxxx"   "/ACK='User Message'"
```

If the request is no longer in the RENC file, you will have to provide all information.

FUN=E, EERP context (/ORG=/DST=/P11=/P12=/P13=/P51=/P61=/P62=), ACK='feedback message'.:

```
p1b8pe2e  "/FUN=E/SPN=ident"  "/'EERP context'" "/ACK='User Message'"
```

Using API L0b2z20

To submit an EERP request from a program, use d0b8z20.h as you would for a transfer request, and provide the specified information:

```
struct st_sci {
  char dire[1];                    /* Direction                        */
  char file[8];                    /* Symbolic file name    p1b8pe2e */
  char part[8];                    /* Symbolic partner name p1b8pe2e */
  char dsnam[127];                 /* Dsname                           */
  char prty[1];                    /* Priority                         */
  char dat[8];                     /* Date                             */
  char hour[6];                    /* Hour                             */
  char lnk[1];                     /* Link type                        */
  char udf[44];                    /* User data file                   */
  char typ[1];                     /* Request type  = E p1b8pe2e     */
  char sta[1];                     /* State of Request                 */
  char dpcsid[8];                  /* Dpcsid for Alias                 */
  char dpcpsw[8];                  /* Dpcpsw for Alias                 */
  char format[2];                  /* Record Format (TF TV BF BU)    */
  char lrecl[5];                   /* Record Length                    */
  char api[88];                    /* Api Field                        */
  char tsm[3];                     /* Type/Structure/Mode FTP          */
  char stou[1];                    /* Store Unique FTP                 */
  char fa[1];                      /* flag File agent Y/N              */
  char label[80];                  /* Label                            */
  char s_pi99_254[254];            /* Feedback on 254      p1b8pe2e  */
  char user_org[8];                /* User Origin          p1b8pe2e  */
  char user_dst[8];                /* User Destination     p1b8pe2e  */
  char user_snd[24];               /* User Sender   pi61   p1b8pe2e  */
  char user_rcv[24];               /* User Receiver pi62   p1b8pe2e  */
  char quant_aa[2];                /* AA for Julian Date               */
  char quant[3];                   /* Julian Date                      */
  char notif[1];                   /* Notification: space/0-7          */
  char noreq[8];                   /* request number       p1b8pe2e   */
  char dhc[12];                    /* File date Pi51       p1b8pe2e   */
  char idt[8];                     /* Pi13                  p1b8pe2e   */
  char ftype[4];                   /* Pi11                  p1b8pe2e   */
  char filler[SIZE_RENC - 675];
};
```

Receiving End to End Response

Receiving an end to end response means that data is received through the PeSIT message service, Pi11 different from FFFF or FFFE. The file name is provided by Pi12. The process is similar to the PeSIT message process described before.

The EERP process doesn't require a file definition to execute: if symbolic file $$EERP$$ is defined, and status enabled, the process will be executed according to this profile: for example, exits, commands, physical file name are used. If the file is defined and no $$EERP$$ definition exists, or status is disabled, the transfer will be executed according to this profile.

When receiving an EERP, Sterling Connect:Express searches for the corresponding request. The request must be a transmission, a file or a message, with status ended = 'E'. TRC=2050 or 2055 is issued if the request does not meet the condition. If the request is not found, the EERP is accepted. The status of the corresponding request in the RENC file is changed from 'E' to 'X' when the EERP transfer is successfully completed.

```
10/06/04 16:24:23 REQUEST 07200008  FICMSG  <- partner EERP:  org    dest   idt
10/06/04 16:24:23 REQUEST 07200008   FICMSG    <- partner EERP    RECEIVED
10/06/04 16:24:23 REQUEST 07200008 254 first characters of the user feedback
```

254 characters of the feedback message received are shown in STERM monitoring screens.

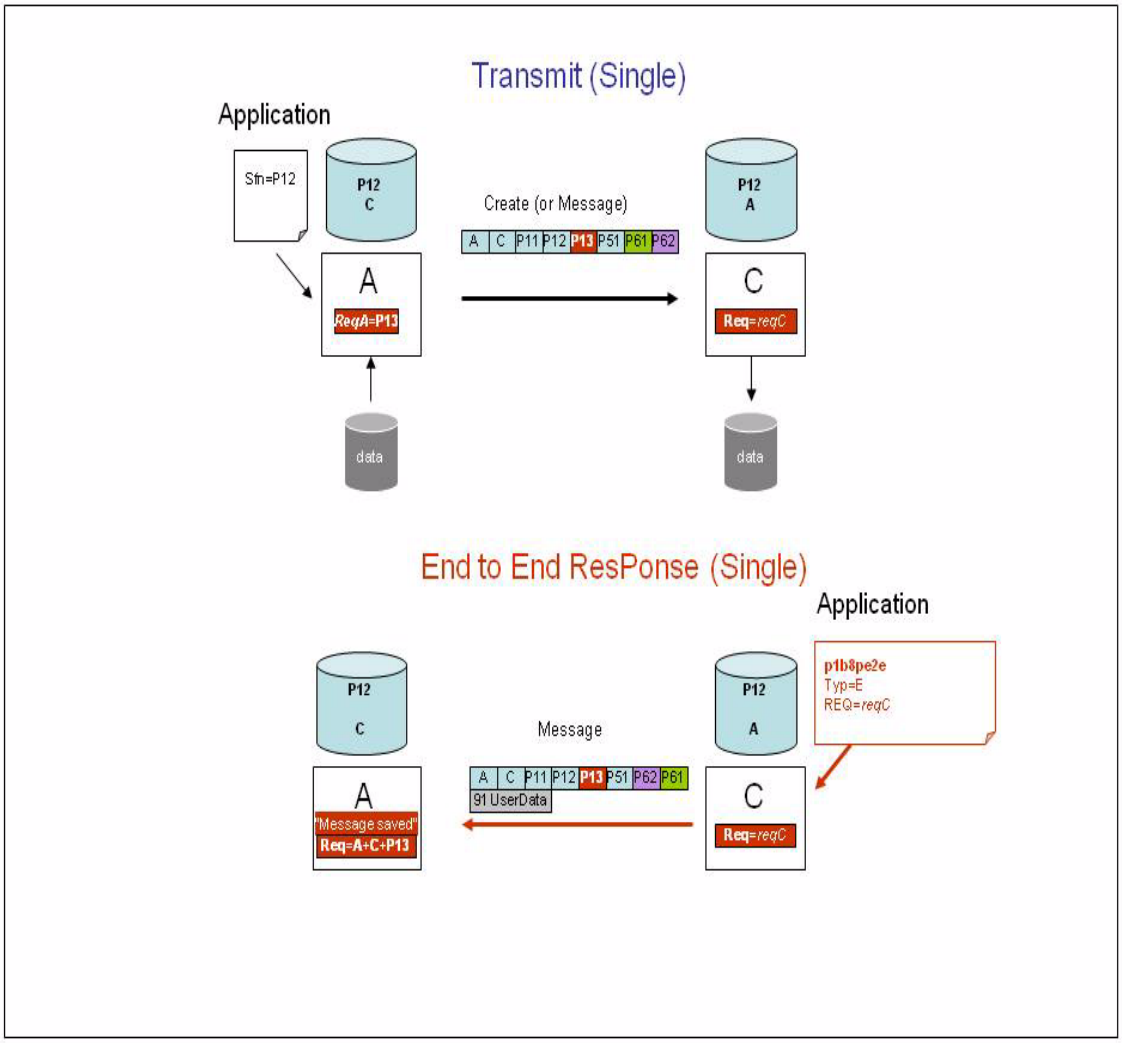
Forwarding and Acknowledging Files

The figure below shows how EERP works, in the most simple process: A sends a file to C, and C sends back an EERP to acknowledge reception.

The A request number, ReqA, is set in Pi13 that is the file transfer identification. C receives the file, with ReqC request number. The local ReqC record is saved. The application ackowledges the file using the ReqC information in which Pi13 has been saved. The end to end response is built from ReqC and sent in a PeSIT message to A. A receives the EERP message and checks in its RENC file the request that is being ackowledged from the information A+C+Pi13.

When the EERP is successfully sent, C changes ReqC status from E to X, and A changes ReqA status from E to X.

Depending if EERP is part of a store and forward process or not, the message must be forwarded or not.


Transmit (Single)
Application
Sfn=P12
P12
C
A
ReqA=P13
data
Create (or Message)
A C P11 P12 P13 P51 P61 P62
P12
A
C
Req=reqC
data
End to End ResPonse (Single)
Application
p1b8pe2e
Typ=E
REQ=reqC
P12
C
A
"Message saved"
Req=A+C+P13
Message
A C P11 P12 P13 P51 P62 P61
91 UserData
P12
A
C
Req=reqC

Store and Forward Overview

You can set up a store and forward process using p1b8pe2e utility. End of transfer commands enable you to save parameters for further use, or to activate automatic forwarding or acknowledgment. The store and forward function is available on Sterling Connect:Express for UNIX. User commands are provided to perform store and forward. Next section describes the automatic store and forward process.

Using Automatic Routing

When the DPCSID ALIAS field of the partner is set to **xxxx** - where ‘xxxx’ is any string composed of A-Z, 0-9, a-z - the UEXxxxx command is launched at end of reception. You can use this mechanism to forward a file, a message or an EERP to the destination. The following store and forward user commands are provided:

exit/UEXFWRD	Uses p1b8pe2e utility to perform a PeSIT forward process.
exit/UEXEERP	Uses p1b8pe2e utility to send an EERP.
exit/UEXROUT	Uses p1b8preq utility to forward the file according to origin/destination (pi3bis/pi4bis).

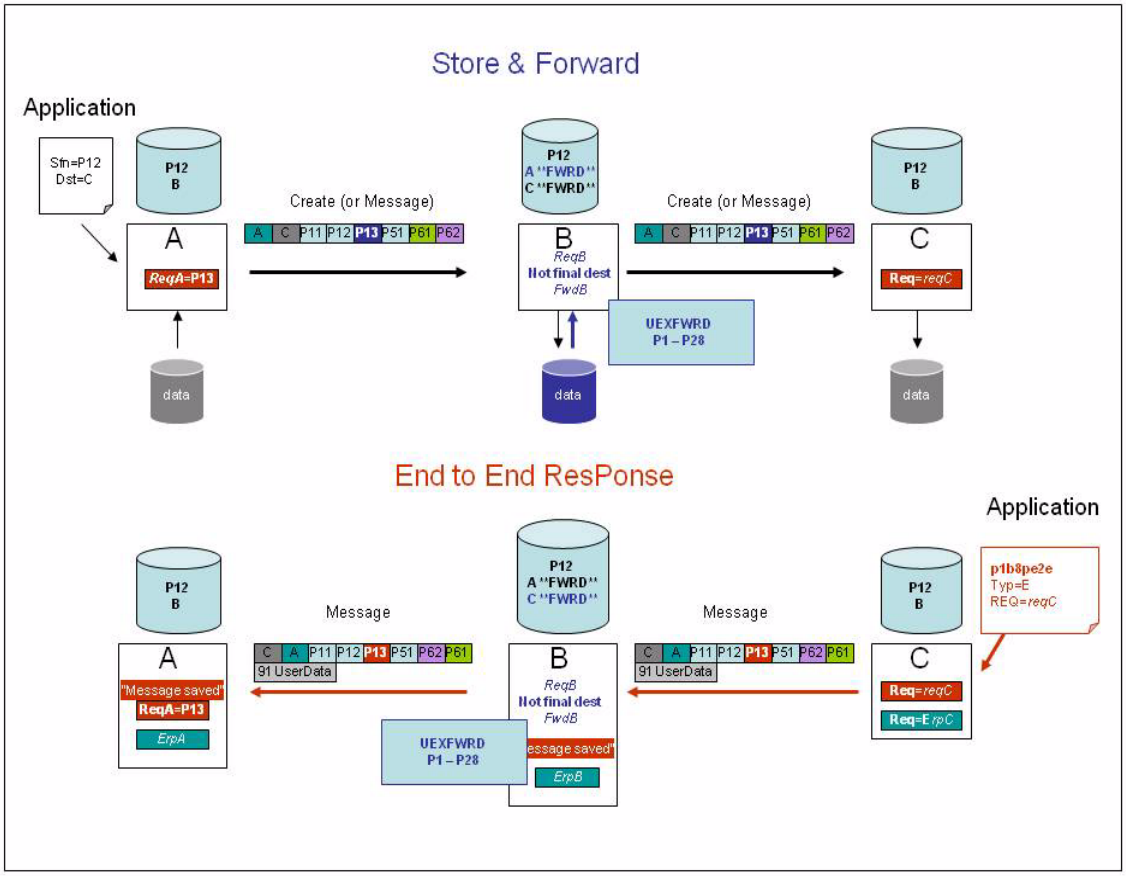
The figure next page shows that A is sending a file or a message to C, via B, and C is sending back the acknowledgment of the reception to A, via B.

Partner A and C are configured in B with DPCSID ALIAS = **FWRD**. B launches UEXFWRD at end of reception of any file or message from A, using p1b8pe2e with FUN=F, type of request = N or M. All end to end transfer PeSIT fields are forwarded in the new transfer. The transfer request number of A is ReqA: it is set in the Pi13. Request numbers on B are ReqB for reception and FwdB for transmission. Request number on C is ReqC.

After data processing, the Application submits a p1b8pe2e request to C, TYP=E, for request ReqC, to acknowledge the file or message received. This is a new request with number ErpC. Sterling Connect:Express prepares the PeSIT Message fpdu from the RENC file record that is accessed with the request number ReqC to retreive original information.

B receives the EERP message for symbolic file PI12, from partner C, and saves it in the RENC file, with request number ErpB. C changes ReqC status to X, and B retreives FwdB from information (Pi13+A+C+sent from A to C) and changes status to X. Partner A and C are configured in B with DPCSID ALIAS = **FWRD**. B knows that this is a end to end message (from the Pi11). B launches UEXFWRD at end of reception, using p1b8pe2e with FUN=F, type of request = E. All end to end transfer PeSIT fields are forwarded in the new transfer. A receives the EERP and saves it in the RENC file. A retreives ReqA from information (Pi13+A+C+sent from A to C) and changes the status to X. B retreives ReqB from information (Pi13+A+C+received from A to C) and changes the status to X.

Note : the difference between UEXROUT and UEXFWRD is that UEXROUT doesn't transmit all the parameters . The forward request is a new request, with a new transfer identification (Pi13).



Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual

Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing

Legal and Intellectual Property Law

IBM Japan Ltd.

1623-14, Shimotsuruma, Yamato-shi

Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT

WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

J46A/G4

555 Bailey Avenue

San Jose, CA__95141-1003

U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available. This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are ficticious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© IBM 2010. Portions of this code are derived from IBM Corp. Sample Programs.
© Copyright IBM Corp. 2010.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Index

A

Application interface 5-1

Application program using L0B8Z20 5-6

B

Batch transfer requests 4-5

Batch utilities 5-1

C

Config directory 1-7

Connect:Express
- Directories 1-6
- Files 1-12
- General structure 1-2
- Implementing 2-8
- Installing 2-2
- Key terms 1-5
- Messages A-1
- Parameters 1-5
- Transfers 1-18, 1-20

Connect:Express return codes B-3

E

Exit directory 1-9

Exits C-3

F

Files Directory 3-7

G

GRTF directory 1-9

H

Hardware requirements 2-1

I

Installation procedure 2-2

Installation requirements 2-1

Interrogation of Log 3-19

ITOM directory 1-10

L

LOB8Z20 5-1

M

Main Menu 3-1

Messages
- Syntax A-1

Monitor management 3-13

Monitor Status 3-13

O

Operator transfer requests 4-3

P

Parameter cards file D-5

Partner lists D-2
Partners directory 3-3
Physical names D-2
Presentation tables
  Updating 3-23
Protocol return codes B-1

R

RENC files 1-17
Request Deletion 3-20
Return codes
  Connect:Express B-3, B-8
  Protocol B-1

S

Session tables
  Updating 3-22
Software requirements 2-1
Standard error command C-3
sterm 2-9
sterm directory 1-11
strf directory 1-11
SYSIN
  Keywords 1-13
SYSLOG option A-2
SYSTCP
  Implementing D-1

T

Tables 3-21
TOM directory 1-7
Transfer process 4-2
Transfer requests
  Displaying the status 3-18
  Interrupting 3-17
  Purging 3-18
  Restarting 3-17
  Types 4-2
  Viewing 3-14
Translating data D-4

U

UEXERR C-3
User commands C-2
User exits C-3