

Technical Bulletin

Part No. 74-0117

DataStage FTP Plug-in

This technical bulletin describes Release 2.3 of the DataStage FTP Plug-in stage, formerly known as the DataStage File Transfer plug-in stage. The FTP Plug-in stage uses remote sequential text files as data sources and data targets for DataStage jobs.

©1998–2003 Ascential Software Corporation. All rights reserved. Ascential, Ascential Software, DataStage, MetaStage, MetaBroker, and Axielle are trademarks of Ascential Software Corporation or its affiliates and may be registered in the United States or other jurisdictions. Adobe Acrobat is a trademark of Adobe Systems, Inc. Microsoft, Windows, Windows NT, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company, Ltd. Other marks mentioned are the property of the owners of those marks. This product may contain or utilize third party components subject to the user documentation previously provided by Ascential Software Corporation or contained herein.

Printing History

First Edition (74-0117) for Release 1.0, February 1998
Second Edition (74-0117) for Release 1.1, April 1998
Third Edition (74-0117) for Release 1.2, April 1998
Fourth Edition (74-0117) for Release 1.3, May 1998
Fifth Edition (74-0117) for Release 2.0, June 1998
Sixth Edition (74-0117) for Release 1.4 for NLS, August 1998
Seventh Edition (74-0117) for Release 2.1, December 1998
Eighth Edition (74-0117) for Release 2.2, March 1999
Ninth Edition (74-0117) for Release 2.3, November 2001
Updated (74-0117) for Release 2.3, August 2002
Updated (74-0117) for Release 2.3, August 2003

How to Order Technical Documents

To order copies of documents, contact your local Ascential subsidiary or distributor, or call our main office at (508) 366-3888.

Documentation Team: Marie E. Hedin

Introduction

This technical bulletin describes the following for Release 2.3 of the FTP Plug-in stage for DataStage Release 7.0:

- [Functionality](#)
- [Terminology](#)
- [Installation](#)
- [Properties](#)
- [ASCII or binary data representation](#)

Like the Sequential File stage, the FTP Plug-in stage extracts data from, or writes data to, a single text file. However, the text files to be accessed by FTP Plug-in reside on another machine (possibly with a different file system and character file storage conventions) over a communications network instead of on a local disk.

FTP Plug-in provides users with rapid and efficient remote file access using existing FTP servers on remote platforms. FTP Plug-in does not require additional installation on the remote platforms.

Additionally, the FTP Plug-in stage provides the option to execute before- and after-commands on the remote machine. This automates the following data flow processes:

- **Before it begins the file transfer.** You can use the before-command to prepare a file to be transferred or to prepare the remote machine to receive it.
- **After it completes the file transfer.** You can use the after-command to delete temporary files or to start a subsequent activity that uses the transferred file.

Each FTP Plug-in stage is a passive stage that can have any number of input and output links:

- Input links specify the data you are writing, which is a stream of rows to be loaded into a single remote file.
- Output links specify the data you are extracting, which is a stream of rows to be read from a single remote file.

Functionality

The FTP Plug-in stage has the following functionality and benefits:

- Shares common properties of the remote host name, user name, password, and directory path to or from which files are transferred for each stage instance.
- Corresponds generally to an independent file transfer session for each link, so that multiple files can be transferred concurrently.
- Acts as an FTP client, using a generic file transfer protocol to initiate sessions with and transfer files to or from any file transfer server. Retains an FTP session long enough to allow the transfer of large amounts of data.
- Supports the STREAM data protocol. If a STREAM transfer connection is closed, the job aborts with an error message.
- Handles job failures appropriately when incomplete files are transferred.

Note: You can specify the number of rows to be processed by a job on the **Limits** page in the DataStage Director. As of Release 1.3, if you perform row limiting, fatal errors may be recorded in the log file at the end of the job because of premature closing of the data connection. However, the data transfer is completed for the number of rows selected.

- Supports a user-specified number of connection retries and retry intervals.
- Provides optional before- and after-commands to be run on the remote machine before and after a file is successfully transferred (requires a telnet server to use all capabilities on Windows NT).
- Provides an optional tracing level to diagnose performance issues.
- Lets you read or write ASCII or binary data.

Note: Binary mode is not supported on the Parallel Server canvas. See the input property **Data Representation Type** and the output property **Data Representation Type**.

- Uses the stage and link properties and column type to determine the format for character strings before the transfer.
- Lets you control which process initiates the connection request for data transfer.
- Provides optional use of meta data definitions for reading a remote file.

- Lets you validate the existence of the remote file within the DataStage Director component (output link only).
- Supports NLS (National Language Support). For information, see *DataStage NLS Guide*.
- Supports MetaStage. For information, see *MetaStage User's Guide*.

The following functionality is not supported:

- Bulk loading for stream input links
- Keyed lookups on a file transfer stage
- Stored procedures

Terminology

The following table explains FTP Plug-in terms used in this document:

Term	Description
after-command	The command to be executed on the remote machine using a telnet session after the transfer is complete.
before-command	The command to be executed on the remote machine using a telnet session before starting the transfer.
FTP	File Transfer Protocol. An interactive file transfer capability often used on TCP/IP networks.
rollback	Cancels all file I/O changes made during a transaction.
telnet	The name of a protocol session that acts as a standard remote terminal emulation with communications to the host over a network.
transaction	A sequence of file I/O operations treated as one logical operation with respect to recovery and visibility to other users.

Installing the Plug-in

For instructions and information supporting the installation, see *DataStage Plug-In Installation and Configuration Guide*.

Note: To specify transaction rollbacks, commits, or after/before processing to the Windows NT server, you must first provide a telnet server other than UniVerse telnet.

Properties

The tables in the following sections include the following column heads:

- **Prompt** is the text that the job designer sees in the stage editor user interface.
- **Type** is the data type of the property.
- **Default** is the text used if the job designer does not supply any value.
- **Description** describes the properties.

Stage Properties

The FTP Plug-in stage supports the following stage properties:

Prompt	Type	Default	Description
Server Name	String	None	Required. The name of the host machine for the FTP server on which the file resides.
Remote FTP Port	Long	21	Required. The port number of the remote machine's FTP server.
Remote Telnet Port	Long	23	Required. The port number of the remote machine's telnet server.
User Name	String	None	Required. The user name to log on to the remote machine.
User Password	String	None	The password for the specified user. Required if the remote machine uses a password for "User Name."
Account Name	String	None	The account name for the remote FTP login. Required only if the remote machine needs user account information during the login process.
Tracing Level	Long	0	Optional. Controls the type of tracing information that is added to the log. Use one of the following tracing levels: 0 No tracing 1 Report stage properties
Retries	Long	3	Optional. The number of retries if the connection fails.
Retry Interval	Long	15	Optional. The number of seconds to wait between retries if the connection fails.

Prompt	Type	Default	Description
Number of Telnet Prompts	String	2	Required if telnet services are being used. The total number of expected prompts that are received during the process of logging on to the telnet server.
Telnet Prompt 1	String	login	Required if telnet services are being used. The literal string (case-insensitive) that is sent by the telnet server, prompting the DataStage process for login data.
Telnet Reply 1	String	None	Required if telnet services are being used. The telnet user name to log on to the telnet session.
Telnet Prompt 2	String	password	Required if telnet services are being used. The literal string (case-insensitive) that is sent by the telnet server, prompting the DataStage process for password data.
Telnet Reply 2	String	None	Required if telnet services are being used. The telnet password for the specified telnet user.
Telnet Prompt <i>n</i>	String	None	Any prompts that are needed to connect to a target system through telnet, in addition to login and password.
Telnet Reply <i>n</i>	String	None	Any replies that are needed to connect to a target system through telnet, in addition to login and password.
Command Timeout	Int	50	The number of milliseconds to wait for the Telnet Before and After Commands to complete.

Input Link Properties

The following table lists the input link properties in the grid editor:

Prompt	Type	Default	Description
Remote Path	String	None	Optional. The pathname of the working directory on the remote machine where the files to be retrieved or sent reside.
Remote File Name	String	None	Required. The name of the file on the remote machine to be retrieved or sent.

Prompt	Type	Default	Description (Cont.)
Data Representation Type	List	ASCII	<p>Required. Controls how the data in the remote file is read or written. For ASCII representation, the data transfer uses standard NVT-ASCII, primarily for text files.</p> <p>For binary representation, the data is transferred in contiguous bits as IMAGE data. You must set "Fixed-width Columns" to Yes.</p> <p>Note: Binary mode is not supported when the stage is run on the Parallel canvas. To transfer data in binary mode, use data types of binary or varbinary with "Data Representation Type" set to ASCII.</p>
Line Termination	List	[CR] [LF] (DOS-Style Termination)	<p>Specifies the row (end-of-line) termination sequence in the remote file.</p> <p>If "Data Representation Type" is set to ASCII, the valid values are no termination and [CR] [LF].</p>
Fixed-width Columns	String	No	Required. Indicates whether the data in the remote file can be with fixed-width columns.
Spaces Between Columns	Long	0	The number of spaces between fixed-width columns in the remote file. Required if "Fixed-width Columns" is set to Yes.
Column Delimiter	Char	, (comma)	Required if "Fixed-width Columns" is set to No. The delimiter that separates the data fields in the remote file. You can enter an unquoted single character or the ASCII value of the character you want to use.
Quote Character	Char	" (double quote)	Optional and only valid if "Fixed-width Columns" is set to No. The single character used to enclose a data value that contains the delimiter character as data. You can also enter the ASCII value for the character you want to use. You can suppress "Quote Character" by entering no value.
Escape Character	Char	\ (back- slash)	Required. The single character entered to be interpreted as the escape character.
Null String	String	None	Optional. Specifies the string that is to be interpreted as the SQL null value.

Prompt	Type	Default	Description (Cont.)
First Line Column Names	String	No	Required if "Data Representation Type" is set to ASCII. Specifies whether to transfer the first line in the remote file (that is, it may contain column names).
Omit Last New Line	String	No	Required. Indicates whether you want to omit the last newline at the end of the data while sending it to the remote machine.
Append to File	String	No	Optional. Indicates whether the data is put into the remote file in append or overwrite mode. Yes indicates append to the existing file. No indicates overwrite the file.
Back Up File	String	No	Optional. Indicates whether "Telnet Backup Command" is executed before proceeding with the job.
Telnet Backup Command	String	None	Optional. Specifies the telnet command to execute on the remote machine before the job writes to the remote file. This telnet command is executed only if "Back Up File" is set to Yes. Use this command to create file backups.
Telnet Before Command	String	None	Optional. The telnet command to execute on the remote machine before starting a job.
Telnet After Command	String	None	Optional. Specifies the telnet command to execute on the remote machine after completing a job.
Transaction Begin Command	String	None	Optional. Specifies the telnet command to execute before starting the file transfer to the remote machine. Use this command to make temporary copies of files.
Transaction Commit Command	String	None	Optional. Specifies the telnet command to execute after a successful file transfer. Use this command to delete any temporary files created.
Transaction Rollback Command	String	None	Optional. Specifies the telnet command to execute if an error occurs while sending the file to the remote machine, or if you use the DataStage Director to reset the job. Use this command to restore any file from the temporary copy in the event of a failure or abort.

Prompt	Type	Default	Description (Cont.)
FTP Data Connection Mode	List	Passive	<p>Specifies which process initiates the connection for the data transfer.</p> <p>If set to Active, connections are initiated by the FTP server.</p> <p>If set to Passive, connections are initiated from the host system where the DataStage server is installed. This lets you store files on remote hosts that are outside router-based firewalls.</p> <p>Digital OpenVMS systems. Set to Active for input links so that the FTP server initiates the connection for data transfer. Otherwise, no data is accepted.</p>
Link Tracing Level	Long	0	<p>Optional. Controls the type of tracing information that is added to the log. The available tracing levels are:</p> <ul style="list-style-type: none"> 0 No tracing 1 Link properties 2 Performance 4 FTP messages 8 Telnet messages 16 Function tracing 32 Telnet data dump <p>You can combine the tracing levels. For example, a tracing level of 3 means that link properties and performance messages are added to the log.</p>
Buffer Length	Long	4096	<p>Required. Sets the length (in chunks greater than 512 bytes) of the FTP send and receive buffers for data rows before they are sent or retrieved.</p>

You can specify any UNIX command for the following link properties: Telnet After Command, Telnet Backup Command, Transaction Begin Command, Transaction Commit Command, or Transaction Rollback Command. For example, the following UNIX command copies a file to another file in a different directory:

```
cp /pathname/filename1 /pathname2/filename2
```

Output Link Properties

The following table lists the input link properties in the grid editor:

Prompt	Type	Default	Description
Remote Path	String	None	Optional. The pathname of the working directory on the remote machine where the files to be retrieved or sent reside.
Remote File Name	String	None	Required. The name of the file on the remote machine to be retrieved or sent.
Data Representation Type	List	ASCII	<p>Required. Controls how the data in the remote file is read or written. For ASCII representation, the data transfer uses standard NVT-ASCII, primarily for text files.</p> <p>For binary representation, the data is transferred in contiguous bits as IMAGE data. You must set "Fixed-width Columns" to Yes.</p> <p>Note: Binary mode is not supported when the stage is run on the Parallel canvas. To transfer data in binary mode, use data types of binary or varbinary with "Data Representation Type" set to ASCII.</p>
Check Data against metadata	List	No	<p>Set to Yes to use meta data definitions to read data from the remote file instead of using a line terminator to identify the end of a row. Data is read until the meta data is exhausted.</p> <p>For fixed-width data, this means the total of the column lengths plus spaces. For delimited data, this means the number of columns.</p> <p>If set to No, end of row is determined by the end-of-line sequence [CR] [LF]</p>
Line Termination	List	[CR] [LF] (DOS-Style Termination)	<p>Specifies the row (end-of-line) termination sequence in the remote file.</p> <p>If "Fixed-width Columns" is set to No, use the [CR] [LF] value. If "Fixed-width Columns" is set to Yes, and "Data Representation Type" is set to ASCII, the valid values are no termination and [CR] [LF] (DOS style terminator).</p> <p>If set to no termination, "Check Data against metadata" must be set to Yes.</p>
Fixed-width Columns	String	No	Required. Indicates whether the data in the remote file can be with fixed-width columns.

Prompt	Type	Default	Description
Spaces Between Columns	Long	0	The number of spaces between fixed-width columns in the remote file. Required if "Fixed-width Columns" is set to Yes.
Column Delimiter	Char	,	(comma) Required if "Fixed-width Columns" is set to No. The delimiter that separates the data fields in the remote file. You can enter an unquoted single character or the ASCII value of the character you want to use.
Quote Character	Char	"	(double quote) Optional and only valid if "Fixed-width Columns" is set to No. The single character used to enclose a data value that contains the delimiter character as data. You can also enter the ASCII value for the character you want to use. You can suppress "Quote Character" by entering no value.
Escape Character	Char	\	(back-slash) Required. The single character entered to be interpreted as the escape character.
Null String	String	None	Optional. Specifies the string that is to be interpreted as the SQL null value.
First Line Column Names	String	No	Required if "Data Representation Type" is set to ASCII. Specifies whether to transfer the first line in the remote file (that is, it may contain column names).
Telnet Before Command	String	None	Optional. The telnet command to execute on the remote machine before starting a job.
Telnet After Command	String	None	Optional. Specifies the telnet command to execute on the remote machine after completing a job.
FTP Data Connection Mode	List	Active	Specifies which process initiates the connection for the data transfer. If set to Active, connections are initiated by the FTP server. If set to Passive, connections are initiated from the host system where the DataStage server is installed. This lets you store files on remote hosts that are outside router-based firewalls.

Prompt	Type	Default	Description
FTP Data Port	List	None	Optional. The unique post number on which to receive the data from the remote machine's FTP server. The remote machine's FTP server connects to this port to transfer the remote file. If you do not specify a value, or the value is 0, the stage automatically configures an available port number for you. If you specify a value, it must be from 1025 to 4999. For more information on the FTP model, see the standard, RFC 959 File Transfer Protocol (FTP).
Link Tracing Level	Long	0	Optional. Controls the type of tracing information that is added to the log. The available tracing levels are: 0 No tracing 1 Link properties 2 Performance 4 FTP messages 8 Telnet messages 16 Function tracing 32 Telnet data dump You can combine the tracing levels. For example, a tracing level of 3 means that link properties and performance messages are added to the log.
Buffer Length	Long	4096	Required. Sets the length (in chunks greater than 512 bytes) of the FTP send and receive buffers for data rows before they are sent or retrieved.

ASCII or Binary Data Representation

ASCII data representation for output links. If you set "Data Representation Type" to ASCII, the following occurs:

1. The FTP service is configured for ASCII representation type. The sender (remote host) converts the data from its internal character representation, that is, ASCII or EBCDIC, to the standard NVT-ASCII representation. For more information on FTP data representation and storage, see the standard, RFC 959 File Transfer Protocol (FTP).
2. The data stream received from the remote host is parsed into rows of data by scanning the data for the end-of-line sequence [CR] [LF].
3. The row of data is further parsed into column data. The parsing method used depends on the setting for "Fixed-width Columns." If set to Yes, the

column meta data determines field sizes. If set to No, the row is parsed into columns by scanning for the column delimiter.

ASCII data representation for input links. If you set “Data Representation Type” to ASCII, the following occurs:

1. The FTP service is configured for ASCII representation type. The receiver (remote host) converts the data from ASCII format to its own internal format.
2. Column data (per row) is put in a formatted row. The format depends on the setting for “Fixed-width Columns.” If set to Yes, the data is put in a character buffer. The column meta data determines the size allotted per column. If the column data is greater than the column width, the data is truncated to the meta data column-width, and a warning message appears. If set to No, the data is put in a character buffer, separated by the delimiter for the configured column.
3. The termination characters [CR] [LF] are appended to each row of data. The data is sent to the remote machine to be stored as a text file.

Binary data representation for output links. If you set “Data Representation Type” to Binary, note the following:

1. “Fixed-width Columns” must be set to Yes.
2. The FTP service is configured for IMAGE representation type. The data is sent from the remote machine as contiguous bits with no character conversions.
3. The data stream received from the remote machine is parsed into rows of data by determining the total length of the row. The row length is calculated by the accumulation of each column’s width and the values associated with “Spaces Between Columns” and “Line Termination.”
4. The row of data is further parsed into column data using the same properties and meta data.

Note: Binary mode is not supported on the Parallel Server canvas. See the input property **Data Representation Type** and the output property **Data Representation Type**.

Binary data representation for input links. If you set “Data Representation Type” to Binary, note the following:

1. “Fixed-width Columns” must be set to Yes.

2. The FTP service is configured for IMAGE representation type. The data is sent as contiguous bits with no character conversions.
3. Column data per row is put in a character buffer. The column meta data determines the size allotted per column. If the column data is greater than the column width, the data is truncated to the meta data column-width, and a warning message appears.
4. The termination characters specified by "Line Termination" are appended to each row of data and are sent to the remote machine.

Note: Binary mode is not supported on the Parallel Server canvas. See the input property **Data Representation Type** and the output property **Data Representation Type**.

