

Ascential DataStage™

Teradata Load Stage Guide

Version 1.5



Ascential[®]
Software

This document, and the software described or referenced in it, are confidential and proprietary to Ascential Software Corporation ("Ascential"). They are provided under, and are subject to, the terms and conditions of a license agreement between Ascential and the licensee, and may not be transferred, disclosed, or otherwise provided to third parties, unless otherwise permitted by that agreement. No portion of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Ascential. The specifications and other information contained in this document for some purposes may not be complete, current, or correct, and are subject to change without notice. NO REPRESENTATION OR OTHER AFFIRMATION OF FACT CONTAINED IN THIS DOCUMENT, INCLUDING WITHOUT LIMITATION STATEMENTS REGARDING CAPACITY, PERFORMANCE, OR SUITABILITY FOR USE OF PRODUCTS OR SOFTWARE DESCRIBED HEREIN, SHALL BE DEEMED TO BE A WARRANTY BY ASCENTIAL FOR ANY PURPOSE OR GIVE RISE TO ANY LIABILITY OF ASCENTIAL WHATSOEVER. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL ASCENTIAL BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs.

© 2004, 1999-2003 Ascential Software Corporation. All rights reserved. DataStage®, EasyLogic®, EasyPath®, Enterprise Data Quality Management®, Iterations®, Matchware®, Mercator®, MetaBroker®, Application Integration, Simplified®, Ascential™, Ascential AuditStage™, Ascential DataStage™, Ascential ProfileStage™, Ascential QualityStage™, Ascential Enterprise Integration Suite™, Ascential Real-time Integration Services™, Ascential MetaStage™, and Ascential RTI™ 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. Teradata is a registered trademark of NCR International, Inc. 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.

The software delivered to Licensee may contain third-party software code. See *Legal Notices* ([LegalNotices.pdf](#)) for more information.

How to Use This Guide

The Teradata Load stage loads data into a Teradata database table. It also reads data exported from a Teradata database. Version 1.5 of Teradata Load is compatible with Ascential DataStage Release 7.5.1.

Audience

This guide is intended for DataStage designers who create or modify jobs that use the Teradata Load stage.

How This Book is Organized

The following table lists topics that may be of interest to you and it provides links to these topics.

To learn about	Read...
Functionality	"Functionality" on page 1
Installation	"Installing the Stage" on page 2
Using Teradata Load	"Using Teradata Load" on page 3
Teradata Input link	"Teradata Input Link" on page 4
Teradata output link	"Teradata Output Link" on page 16
Data type support	"Data Type Support" on page 20

Related Documentation

To learn more about documentation from other Ascential products and third-party documentation as they relate to Teradata Load, refer to the following sections/tables.

Ascential Software Documentation

Guide	Description
<i>Ascential DataStage Server Job Developer's Guide</i>	Instructions for using a stage in a DataStage job
<i>Ascential DataStage Designer Guide</i>	General principles for designing jobs
<i>Ascential MetaStage User's Guide</i>	Information about Ascential MetaStage™
<i>Ascential DataStage NLS Guide</i>	Information about NLS and techniques for character-set mapping
<i>Ascential DataStage Plug-In Installation and Configuration Guide</i>	Information required to configure your system and install this stage

Teradata Documentation

Guide	Description
<i>Introduction to Teradata RDBMS</i>	Information about Teradata databases
<i>Teradata BTEQ Reference</i>	Information about the BTEQ program
<i>Teradata FastLoad Reference</i>	Description of the FastLoad utility and of FastLoad errors
<i>Teradata Messages Reference</i>	Description of Teradata errors

Conventions

Convention	Used for...
bold	Field names, button names, menu items, and keystrokes. Also used to indicate filenames, and window and dialog box names.
user input	Information that you need to enter as is.
code	Code examples
<i>variable</i> or <variable>	Placeholders for information that you need to enter. Do not type the greater-/less-than brackets as part of the variable.

Convention	Used for...
>	Indicators used to separate menu options, such as: Start >Programs >Ascential DataStage
[A]	Options in command syntax. Do not type the brackets as part of the option.
B...	Elements that can repeat.
A B	Indicator used to separate mutually-exclusive elements.
{ }	Indicator used to identify sets of choices.

Contacting Support

To reach Customer Care, please refer to the information below:

Call toll-free: 1-866-INFONOW (1-866-463-6669)

Email: support@ascentialsoftware.com

Ascential Developer Net: <http://developernet.ascential.com>

Please consult your support agreement for the location and availability of customer support personnel.

To find the location and telephone number of the nearest Ascential Software office outside of North America, please visit the Ascential Software Corporation website at <http://www.ascentialsoftware.com>.

Contents

Audience	iii
How This Book is Organized	iii
Related Documentation	iii
Ascential Software Documentation	iv
Teradata Documentation	iv
Conventions	iv
Contacting Support	v
Introduction	1
Functionality	1
Installing the Stage	2
Using Teradata Load	3
About the Stage Page	3
The General Tab	4
Defining Character Set Mapping	4
Teradata Input Link	4
Loading Data	4
Loading Modes	5
Troubleshooting	6
Input Data	6
About the Input Page	8
Teradata Output Link	16
About the Output Page	17
Data Type Support	20
DATE Data Type Considerations	21

Introduction

Teradata Load is a passive stage that loads streams of tabular data into tables of a target Teradata database. This stage can have any number of input and output links. (Reference links have no meaning in the context of the Teradata Load stage and are not allowed.)

Input Links

Input links specify the data you are writing, which is a stream of rows to be loaded into a Teradata table. The Teradata Load stage generates a control and data file, which use the Teradata FastLoad utility to load into a single table. The data file contains formatted rows and columns to be loaded into a single table on a Teradata target database.

Output Links

Output links specify the data you are reading, which is a formatted file exported from a Teradata database using the Basic Teradata Query (BTEQ) utility.

Functionality

The Teradata Load stage has the following functionality:

- Support for data files which exceed the 2-GB file size limit for 64-bit file systems.
- Support for NLS (National Language Support). For information, see *Ascential DataStage NLS Guide*.
- Support for Ascential MetaStage™. For more information, see *Ascential MetaStage User's Guide*.
- Generation, and optional automatic execution, of the Teradata commands to load the database with data from input links.
- Support for loading from a stream input link to provide rows of data into the target table.
- Support for FastLoad, the Teradata Load utility for loading data.
- Support for a control file that contains Teradata commands to load or reload a database.
- Two load modes: manual or automatic.
- Load parameters to control the load process.
- Automatically drops and creates specified target tables.

The following functionality is not supported:

- Compatibility with Ascential DataStage releases before 7.0
- Reference output links

- Meta data importing
- Support for stored procedures
- Native data browsing
- Supports reject row handling

Installing the Stage

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

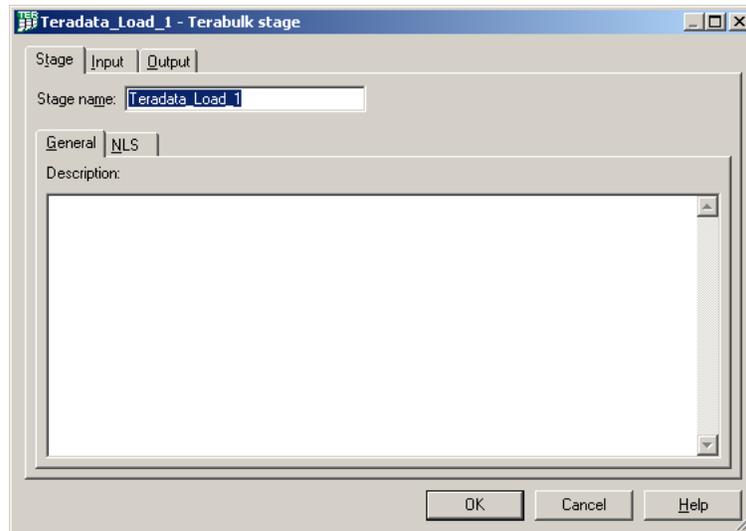
Before installing the stage, consult Teradata documentation for any specific configuration requirements.

Install the Teradata client software on the Ascential DataStage server machine. Use the Teradata client software only for automatic execution of load commands. For manual execution of FastLoad commands in batch mode, no Teradata software is necessary to create control and data files. To execute FastLoad later, Teradata software is required.

Ensure that the Teradata server is running for the automatic execution of FastLoad load commands from the stage or for manual execution of FastLoad commands at the DOS prompt.

Using Teradata Load

Double click the **Teradata Load** icon, right click the **Teradata Load** icon and select **Properties**, or choose **Properties** from the **Edit** menu. The **Terabulk stage** dialog box appears.



This dialog box has up to three pages:

- The Stage page displays the name of the stage you are editing, the General tab and optionally the NLS tab.
- The Input page specifies the information necessary to load data in a Teradata database using FastLoad. This page also specifies the associated column definitions. See ["Teradata Input Link"](#) on page 4.
- The Output page specifies the information necessary to import data from a Teradata database into DataStage. This page also specifies the associated column definitions. See ["Teradata Output Link"](#) on page 16.

About the Stage Page

The **Stage** page consists of the **Stage name**, the **General** tab, and **NLS** tab (optional). The **General** tab on the **Stage** page appears by default.

- **Stage name.** The name of the stage you are editing.

The General Tab

The following field is in the **General** tab:

- **Description.** Optionally, a description of the purpose of the Teradata Load stage.

Defining Character Set Mapping

You can define a character set map for a stage. Do this from the **NLS** tab that appears on the **Stage** page. The **NLS** page appears only if you have installed NLS.

Enter information for the following button and fields, if appropriate:

- **Map name to use with stage.** The default character set map is defined for the project or the job. You can change the map by selecting a map name from the list.
- **Use Job Parameter...** . The parameter values for the job. Use the format *#Param#*, where *Param* is the name of the job parameter. The string *#Param#* is replaced by the job parameter when the job is run.
- **Show all maps.** A list of all the maps that are shipped with Ascential DataStage.
- **Loaded maps only.** A list of only the maps that are currently loaded.

For additional information about NLS, see *Ascential DataStage NLS Guide*, or for additional information about job parameters, see *Ascential DataStage Server Job Developer's Guide*.

Teradata Input Link

When you write data to a Teradata database, Teradata Load has an input link.

Loading Data

The Teradata Load stage uses the FastLoad utility to load data into a Teradata table. FastLoad is a command-driven utility that uses multiple sessions to quickly load large amounts of data into an empty table on a Teradata database.

If Teradata Load has an input link, it generates data to load into a Teradata database table by creating a control and data file for

FastLoad. A DataStage job generates one control file for each input link to each instance of Teradata Load.

The column names and data types of the data are those of the DataStage meta data and must match those of the Teradata table to be loaded or cleared. You can optionally drop and recreate the Teradata table.

The source data for a FastLoad operation comes from the data file generated by the stage. The target table on the Teradata database must be empty, with no defined secondary indexes. Each FastLoad job loads one table into the Teradata database. When using FastLoad to load more than one table, you must submit multiple FastLoad jobs, one for each table.

FastLoad accepts both FastLoad commands and a subset of Teradata SQL statements. The FastLoad commands perform two types of activities:

- Session control commands begin and end FastLoad sessions.
- Data handling commands establish and define a FastLoad operation.

Loading Modes

You can run FastLoad manually in batch mode or automatically from the stage. Set **Load Method** to **Manual** or **Invoke FastLoad** to do this. For additional information, see [page 10](#).

Manual Loading

You may want to run FastLoad manually in batch mode for the following reasons:

- If the FastLoad client software resides on the DataStage server, you can use the **Load Method** to run FastLoad in batch mode after the data and control files have been written. If, however, FastLoad resides on another client or server, you must copy the control and data files to that machine and run FastLoad from there. You can use before- and after-job routines to do this.
- If you need to modify a parameter in the control file before running FastLoad.
- You must also consider the length of time and the system resources FastLoad consumes when you run it using the stage.

Running FastLoad Manually

To run FastLoad manually in batch mode:

- 1 Copy the control and data files to the machine where FastLoad resides.
- 2 Change the directory to the one specified by **Output Files Path** (see [page 11](#)). (If no value is specified in **Output Files Path**, change to the DataStage project directory.)
- 3 Execute the following command:

```
fastload < controlfile.fl
```

controlfile is the value you specify in **Control File** (see [page 11](#)).

Automatic Loading

If you want to run FastLoad from the stage, use the output file from the FastLoad session in the directory path of the control and data files to verify that the load has succeeded. It has a similar name as the control file with the *_floutput.txt* extension appended, for example:

```
controlfile_floutput.txt
```

Note Automatic execution of load commands works only when the Teradata Client resides on the same machine as the DataStage job.

Troubleshooting

Examine the Teradata tables to see why the FastLoad failed. FastLoad stores the input data records related to constraint violations, conversion errors, unavailable Access Module Process (AMP) conditions, and unique primary index violations in the two error tables that you specify in **Error Table 1** and **Error Table 2** (see [page 9](#)).

See *Teradata FastLoad Reference* for information on how to read these error tables and recover or restart the load.

This output contains the end-of-job status report. FastLoad discards all records that produce a duplicate row error, but includes the total number of duplicate rows encountered. It also includes the total records in each error table in the end-of-job status report.

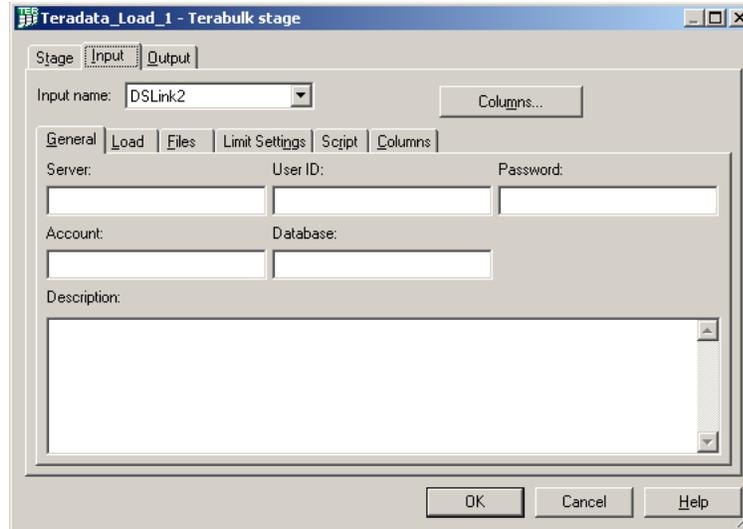
Input Data

Teradata Load generates data files as formatted data, which is input data that conforms to the format of data from a Teradata database source, such as a BTEQ EXPORT file. Each record has a two-byte data length field, a variable-length indicator byte field, a variable-length input data field, and a one-byte end-of-record delimiter field.

SET RECORD Command	<p>The stage uses the SET RECORD command to specify the format of the input data as FORMATTED. Indicators, which let Teradata Load deal with null data (that is, data with no values) are always generated. Thus, null values can be easily loaded into Teradata, and NULLIF qualifiers are not required. This is forced by always adding the INDICATORS keyword to the BEGIN LOADING command. The number of fields in each record determines how many bytes contain null indicators. FastLoad expects the first bytes of the record to contain an indicator bit for each record field. Consult <i>Teradata FastLoad Reference</i> for more information on indicator bits.</p>
DEFINE Command	<p>The DEFINE command describes the fields in a record of input data that are inserted in the FastLoad table. Also, it identifies the name of the input data source. Teradata Load uses the DEFINE command to identify the input data source.</p>
INSERT Statement	<p>To describe the fields in a record of input data the stage uses the INSERT <i>table.*</i> syntax. This retrieves a list of field names from the referenced table. When you use this format of the INSERT statement, FastLoad constructs a list of field names from the table definition. During the insert operation, FastLoad gets the field names and their data types from the CREATE TABLE statement used to define the table and from the table definition. The field name definitions are established in the order in which columns are defined in the CREATE TABLE statement. Therefore, the fields in each data record must be in the same order as the columns in the definition of the table.</p> <p>Note The Teradata INSERT statement for Teradata Load does not support all formats of Unicode data.</p>
Privileges.	<p>The user who is logged in must have the required privileges and rights to run the FastLoad job. Consult the FastLoad documentation for a list of these privileges.</p>

About the Input Page

The Input page has an **Input name** field, a **Columns...** button, and **General**, **Load** (page 9), **Files** (page 11), **Limit Settings** (page 13), **Script** (page 15), and **Columns** tabs (page 16).



- **Input name.** The name of the input link. Choose the link you want to edit from the **Input name** list. This list displays all the input links to the Teradata Load stage.
- **Columns....** A brief list of the columns designated on the input link. As you enter detailed meta data on the **Columns** tab, you can leave this list displayed.

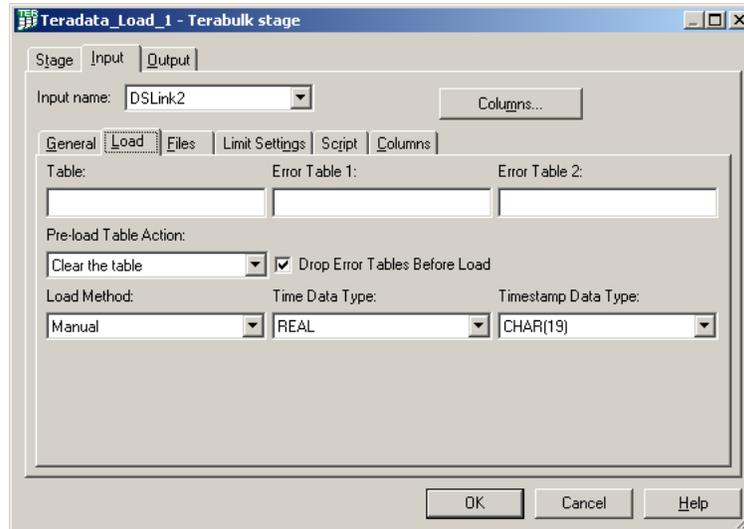
The General Tab

Use the **General** tab on the **Input** page to identify the target of the import into a Teradata table. The **General** tab contains the following fields:

- **Server.** The name of a Teradata Director Program (TDP). Optional.
- **User ID.** A name that identifies the user. The user must have the necessary privileges to write to the database. Required.
- **Password.** The password associated with the Username. Required.
- **Account.** The account associated with the Username. Optional.
- **Database.** The name of the database to be loaded or updated. Optional.
- **Description.** A description of the link. Optional.

The Load Tab

Use the **Load** tab to provide information about the table and the load method.



The **Load** tab contains the following fields:

- **Table.** The name of the FastLoad target table to receive the data from the client system. This name is used in FastLoad statements, such as the `INSERT tablename.*` statement. This statement generates a list of field names from the definition of the table. Required.
- **Error Table 1.** The name of the FastLoad error Table 1 for rows ignored by Teradata because of an error. This table contains records that were rejected because of an error other than unique primary index or duplicate row violation. The stage ensures this is a new table by doing a `DROP TABLE` first. This name is used by the `BEGIN LOADING` statement. Required.
- **Error Table 2.** The name of the FastLoad error Table 2 for rows ignored by Teradata because of an error. This table contains records that violate the unique primary index constraint. The stage ensures this is a new table by doing a `DROP TABLE` first. This name is used in the `BEGIN LOADING` statement. Required.

FastLoad automatically drops error tables that contain no rows when `END LOADING` finishes executing.

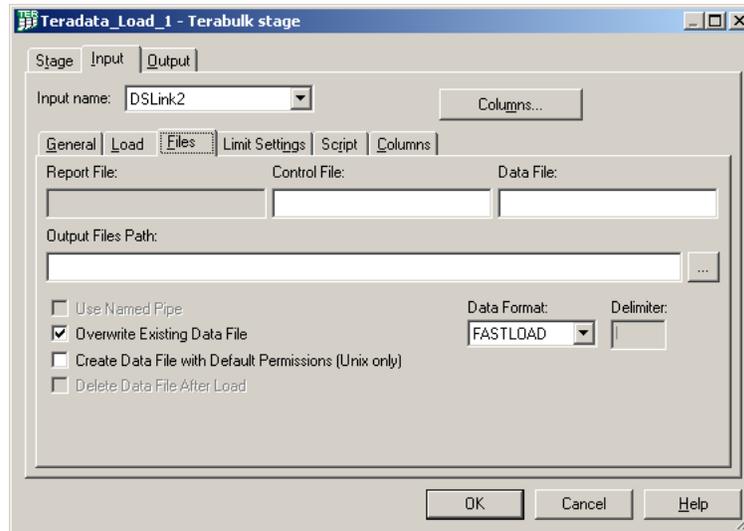
- **Pre-load Table Action.** The action performed on the table prior to loading it. The stage generates the necessary `DROP` and `CREATE TABLE` statements to reestablish the table. The DataStage meta data is used in each of these statements that are created. The options are:

- **Clear the table.** Does not drop the table but deletes all rows from the table.
- **Drop and create the table.** Drops the table and recreates it before loading.
- **User-defined DDL.** Enables the **User-defined DDL** tab on the **Script** tab. The **User-defined DDL** tab allows you to customize the DROP TABLE and CREATE TABLE statements.
- **Drop Error Tables Before Load.** The disposition of the error tables. If **Drop Error Tables Before Load** is selected (which is the default), the generated script drops the error tables before loading begins. If **Drop Error Tables Before Load** is not selected, the stage assumes one of the following:
 - There are no error tables to drop. Therefore, the stage does not put statements to drop the error tables in the generated script. If error tables preexist, FastLoad fails.
 - You are attempting a restart. Therefore the error tables must preexist from a previous run.
- **Load Method.** The method for loading the data. The options are:
 - **Manual.** The files to perform a FastLoad are generated, but no loading is performed by the stage.
 - **Invoke FastLoad.** FastLoad is run after creating the control and data files. Output from the FastLoad sessions is saved in a file.
- **Time Data Type.** The data type of time columns expected by the stage in the Teradata database. The options are:
 - **REAL.** The stage expects time columns to be defined as REAL in the Teradata database. The default value.
 - **Note** Time is encoded for Teradata as (*hour**10000 + *minute**100 + *second*) where *second* may include a fractional value.
 - **TIME(n).** The stage expects time columns to be defined as TIME(n), where n is the Scale value in the range of 0 – 6. 0 – 6 represents the fractional seconds precision. The Length value for the time column must be 8 if Scale equals 0 or the sum of 9 and Scale if Scale is greater than 0.
- **Timestamp Data Type.** The data type of timestamp columns expected by the stage in the Teradata database. The options are:
 - **CHAR(19).** The stage expects timestamp columns to be defined as CHAR(19) in the Teradata database. The default value.

- **TIMESTAMP(n)**. The stage expects timestamp columns to be defined as `TIMESTAMP(n)`, where `n` is the Scale value in the range of 0 – 6, which represents the fractional seconds precision. The Length value for the timestamp column must be 19 if Scale equals 0 or the sum of 20 and Scale if Scale is greater than 0.

The Files Tab

Use the **Files** tab to provide information about the support files.



The **Files** tab contains the following fields:

- **Report File.** The name of the report file. If omitted, the default is `controlfile_floutput.txt`.
- **Control File.** A local name for the FastLoad control file. If the control file does not exist, it is created. If the control file already exists, it is truncated. If omitted, the default is `DATABASE_TABLE.fl`. If **Database** is also omitted, the default is `TABLE.fl` (see [page 8](#)).
- **Data File.** The name of the FastLoad data file, including the `.dat` extension. If omitted, the default is `DATABASE_TABLE.dat`. If **Database** is also omitted, the default is `TABLE.dat` (see [page 8](#)). The value of this property is used in the DEFINE statement. Table definitions are used to define data so that the stage uses the DEFINE statement only with the FILE keyword.
- **Output Files Path.** The directory path for FastLoad control and data files. If omitted, the stage stores the files in the DataStage project directory on the DataStage server.
- **... (ellipsis).** A button that opens a dialog box in which you can browse for the directory path for FastLoad control and data files.

- **Use Named Pipe.** A control that determines whether a named pipe is used for autoloading.
 - If you want rows to be streamed to FastLoad via a named pipe, select **Use Named Pipe**. If **Load Method** is **Manual**, rows are written to a data file. The name and location of the pipe are determined by **Data File** and **Output Files Path**.
 - If you do not want to use a named pipe, clear **Use Named Pipe**. Rows are written to a data file.

This property is available only if **Load Method** is **Invoke FastLoad** (see [page 10](#)). The default is **Use Named Pipe** cleared.

- **Overwrite Existing Data File.** The procedure for handling an existing data file.
 - If you want to replace the current data file each time the job is run, select **Overwrite Existing Data File**.
 - If you want to ignore the input rows and start the load utility, which must load from an existing data file, clear **Overwrite Existing Data File**.

This property is available only if **Use Named Pipe** is cleared. The default is **Overwrite Existing Data File** selected.

- **Create Data File with Default Permissions (Unix only).** The permissions assigned to the data file when it is created by the stage. This property applies only to UNIX.
 - If you want to use the Default file mode creation mask when creating the data file on a UNIX system, select **Create Data File with Default Permissions (Unix only)**.
 - If you want to create the data file with the standard set of permissions that all loaders use, clear **Create Data File with Default Permissions (Unix only)**.

The default is **Create Data File with Default Permissions (Unix only)** cleared.

- **Delete Data File After Load.** The disposition of a data file after a load.
 - If you want to delete the data file after the data is loaded successfully, select **Delete Data File After Load**.
 - If you want to leave the data file intact after loading completes, clear **Delete Data File After Load**.

This property is available only if **Use Named Pipe** is cleared (see [page 12](#)). The default is **Delete Data File After Load** cleared.

- **Data Format.** The format in which rows are written to the data file. The options are:
 - **FASTLOAD.** Rows are written in the Teradata RDBMS standard format.
 - **VARTEXT.** Rows are written in variable-length text record format with each field separated by a delimiter character.

The default is **FASTLOAD**.

- **Delimiter.** The delimiter character that separates fields in variable-length text records. This property is available only if **Data Format** is **VARTEXT**. The default delimiter is the pipe character (|).

The Limit Settings Tab

Use the **Limit Settings** tab to provide parameters to the FastLoad utility. The fields correspond directly to options in the FastLoad utility.

The screenshot shows the 'Teradata_Load_1 - Terabulk stage' dialog box with the 'Limit Settings' tab selected. The 'Input name' is set to 'DSLink2'. The 'Limit Settings' tab contains the following fields:

Field	Value
Sess Max	4
Sess Min	0
Buffer Size	0
Error Limit	0
Checkpoint	0
Start Row	0
End Row	0
Sleep	0
Tenacity	0

Buttons at the bottom: OK, Cancel, Help.

The **Limit Settings** tab contains the following fields:

- **Sess Max.** The number of FastLoad sessions. This property generates a SESSIONS statement, which specifies the number of FastLoad sessions to be logged on when you enter a LOGON statement. The default is 4
- **Sess Min.** The minimum number of sessions required for the job to continue. The default of 0 uses the default value of the FastLoad utility.
- **Buffer Size.** The size of the output buffer, in kilobytes, that is used for FastLoad messages to the Teradata RDBMS. A larger buffer size reduces processing overhead by including more data in each message.

The default buffer size of 0 uses the default value set for the FastLoad utility. If you specify a value greater than the maximum, the FastLoad utility responds with a warning, resets the buffer size to the default value, and continues with the FastLoad job.

- **Error Limit.** The maximum number of rows that can be rejected during a FastLoad run. This generates an ERRLIMIT statement, which limits the number of records that can be rejected while inserting data into the FastLoad table.

The default of 0 does not generate an ERRLIMIT statement and uses the Teradata default.

If you expect no errors in the input data, set the error limit value to 1. The job terminates when any record causes an error.

- **Checkpoint.** The number of rows transmitted to the Teradata RDBMS between checkpoints. This generates a CHECKPOINT option on the BEGIN LOADING command. The default of 0 does not generate a CHECKPOINT option.
- **Start Row.** The row number for FastLoad to start loading from the data file. The default of 0 means processing begins with the first record.

Note This property generates a RECORD statement, which defines the records of the input data source at which FastLoad processing starts and ends.

- **End Row.** The row number for FastLoad to stop loading from the data file. The default of 0 means processing ends with the last record.

The value for this property must be greater than or equal to that of the start record.

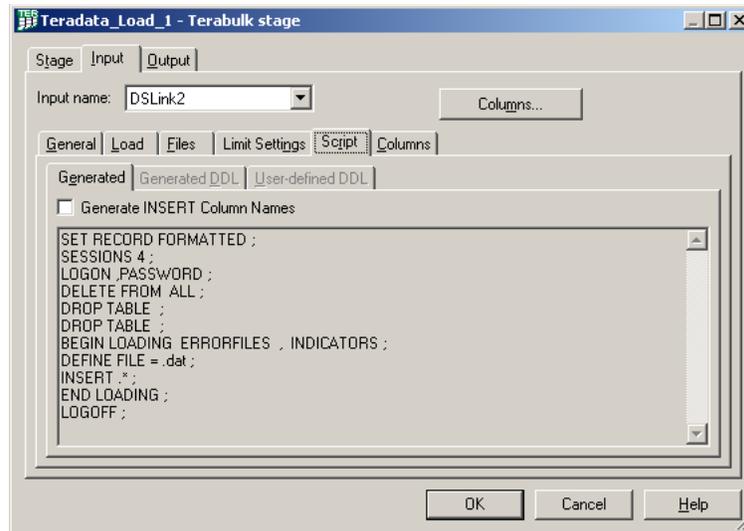
Note This property generates a RECORD statement, which defines the records of the input data source at which FastLoad processing starts and ends.

- **Sleep.** The number of minutes that the FastLoad utility should wait before retrying a logon operation when the maximum number of concurrent load operations are in progress. The default 0 does not generate a SLEEP statement and uses the Teradata default.
- **Tenacity.** The maximum number of hours that the FastLoad utility should continue trying to log on when the maximum number of concurrent load operations are in progress. The default 0 does not generate a TENACITY statement and uses the Teradata default.

See your FastLoad documentation for descriptions of these settings.

The Script Tab

Use the **Script** tab to view the generated FastLoad script.



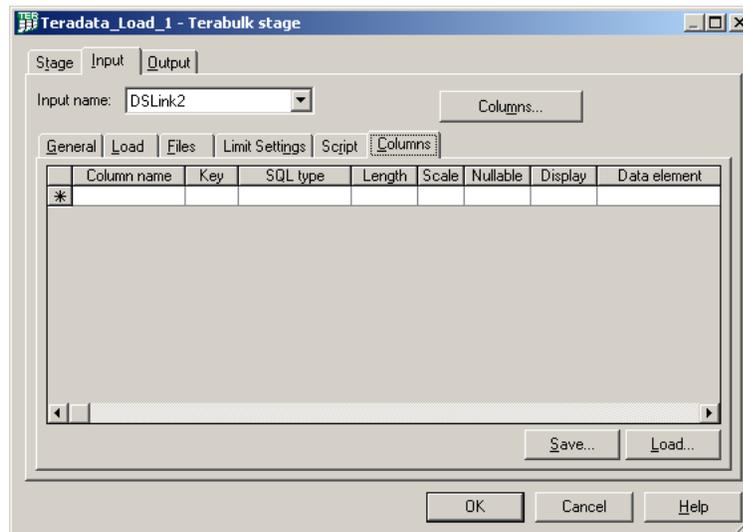
The **Script** tab contains the **Generated**, **Generated DDL**, and **User-defined DDL** tabs.

- **Generated.** The generated FastLoad script.
- **Generate INSERT Column Names.** The disposition of the target column names.
 - If you want the generated INSERT statement to contain the target column names, select **Generate INSERT Column Names**. Select **Generate INSERT Column Names** if the order of the columns on the **Columns** tab does not match the order of the columns in the table or if the data types do not match precisely. The names of the columns on the **Columns** tab must match the names of the columns in the table.
 - If you do not want the generated INSERT statement to contain the target column names, do not select **Generate INSERT Column Names**.
- **Generated DDL.** The generated DROP TABLE and CREATE TABLE statements. **DROP Statement** displays the generated DROP TABLE statement for dropping the target table. **CREATE Statement** displays the generated CREATE TABLE statement that is generated from the column meta data definitions and the information provided in the **Load** tab (see [page 9](#)). The **Generated DDL** tab is disabled if the **Pre-load Table Action** is set to **Clear the Table** (see [page 9](#)).
- **User-defined DDL.** The DROP TABLE and CREATE TABLE statements customized by the user. **DROP Statement** displays the user-defined DROP TABLE statement for dropping the target

table. **CREATE Statement** displays the user-defined CREATE TABLE statement for creating the target table. Use the **User-defined DDL** tab to customize the CREATE TABLE statement. The **User-defined DDL** tab is disabled if the **Pre-load Table Action** is set to **Clear the Table** or **Drop and create the table** (see [page 9](#)).

The Columns Tab

Use the **Columns** tab to define the meta data for the data to be imported to the Teradata database.



You can manually enter the meta data in the grid or load it from the DataStage Repository. The columns listed determine what is imported to the Teradata database.

Click **Save** to save the meta data to the DataStage Repository. Click **Load** to load the meta data from the Repository.

Teradata Output Link

When you read data into Ascential DataStage, Teradata Load has an output link. The output link provides a sequence of rows to read from an export file, which is created by the BTEQ export utility. The meta data for each output column in the Teradata table determines how the data is read.

BTEQ creates this intermediary export file in binary format. Use the output link to read it into Ascential DataStage.

Use the SET RECORDMODE ON command before exporting. The column names and data types of the DataStage meta data must match the Teradata table.

Ascential DataStage can recognize files exported with BTEQ with or without indicators. The values in the columns are set to null if they are used and set.

File Contains Indicator Bytes (see [page 18](#)) controls whether the file was exported using indicator mode and whether indicator bytes are present in the file. If **File Contains Indicator Bytes** is selected and an indicator bit is set to 1, DataStage sets the value of the corresponding field to null when the record is loaded.

Selecting **File Contains Indicator Bytes** means the file was exported with indicators. If selected and indicator bits are not entered at the beginning of the record, Teradata Load assumes that the first field contains indicator bytes and reads the record incorrectly.

If the indicator bit is set to 0, DataStage loads the data specified for that field.

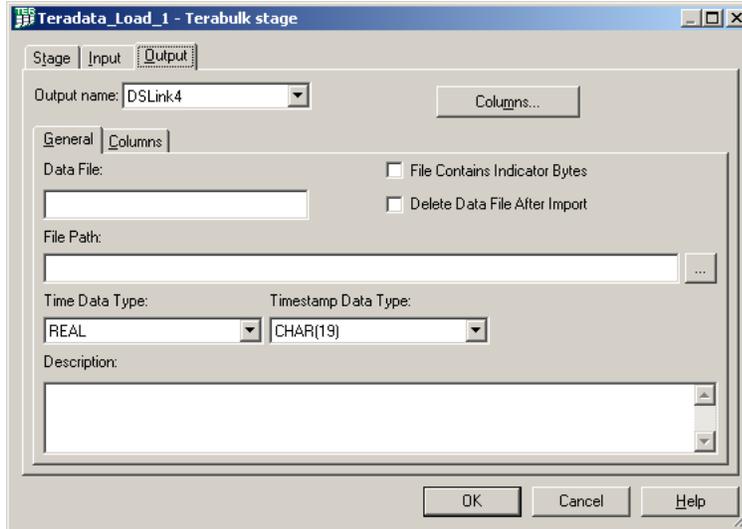
About the Output Page

The Output page has an **Output name** field, a **Columns...** button, and **General** ([page 18](#)) and **Columns** tabs ([page 20](#)).

- **Output name.** The name of the output link. Choose the link you want to edit from the **Output name** list. This list displays all the output links from the Teradata Load stage.
- **Columns....** A brief list of the columns designated on the output link. As you enter detailed meta data on the **Columns** tab, you can leave this list displayed.

The General Tab

Use the **General** tab on the **Output** page to identify the source of the import into DataStage.



The **General** tab contains the following fields:

- **Data File.** The name of the file to be imported, including the .dat extension. Required.
- **File Path.** The directory that contains the file to be imported. Required.
- **File Contains Indicator Bytes.** A control that determines whether the file is exported with indicators and whether indicator bytes are present in the file.

If File Contains Indicator Bytes is	And an indicator bit is	Then DataStage
Selected	Set to 1	Sets the value in the corresponding field to null when the record is loaded, that is, no value
Selected	Set to 0	Loads the data specified for that field
Not selected		Imports the file without indicators

Warning If **File Contains Indicator Bytes** is selected and indicator bits are not entered at the beginning of the record, the stage assumes that the first field contains indicator bytes and reads the record incorrectly.

- **Delete Data File After Import.** The disposition of a data file after an import into DataStage.
 - If you want to delete the data file after the data has been read successfully, select **Delete Data File After Import.**
 - If you want to leave the data file intact after all rows have been read, clear **Delete Data File After Import.**

The default is **Delete Data File After Import** cleared.

- **Time Data Type.** The data type of time columns expected by the stage in the Teradata database. The options are:

- **REAL.** The stage expects time columns to be defined as REAL in the Teradata database. The default value.

Note Time is encoded for Teradata as $(hour*10000 + minute*100 + second)$ where *second* may include a fractional value.

- **TIME(n).** The stage expects time columns to be defined as TIME(n), where n is the Scale value in the range of 0 – 6. 0 – 6 represents the fractional seconds precision. The Length value for the time column must be 8 if Scale equals 0 or the sum of 9 and Scale if Scale is greater than 0.

- **Timestamp Data Type.** The data type of timestamp columns expected by the stage in the Teradata database. The options are:

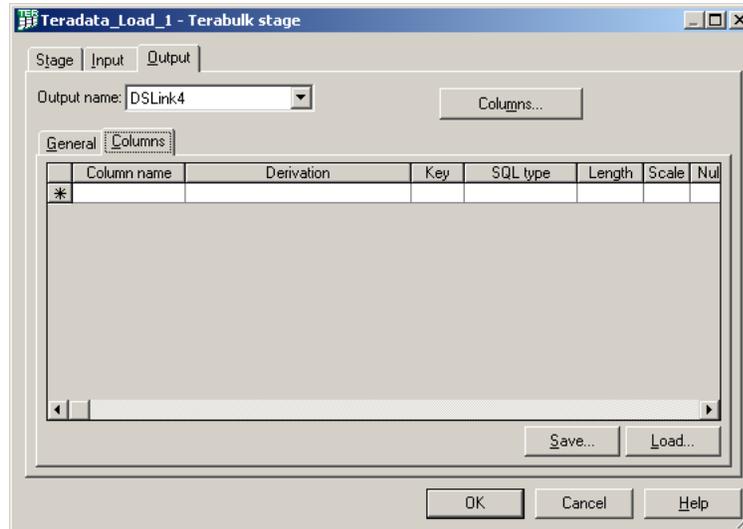
- **CHAR(19).** The stage expects timestamp columns to be defined as CHAR(19) in the Teradata database. The default value.

- **TIMESTAMP(n).** The stage expects timestamp columns to be defined as TIMESTAMP(n), where n is the Scale value in the range of 0 – 6, which represents the fractional seconds precision. The Length value for the timestamp column must be 19 if Scale equals 0 or the sum of 20 and Scale if Scale is greater than 0.

- **Description.** A description of the link. Optional.

The Columns Tab

Use the **Columns** tab to define the meta data for the data to be imported from the Teradata database.



You can manually enter the meta data in the grid or load it from the DataStage Repository. The columns listed determine what is imported to DataStage.

Click **Save** to save the meta data to the DataStage Repository. Click **Load** to load the meta data from the Repository.

Data Type Support

Teradata Load supports mapping to and from DataStage SQL data types and Teradata SQL data types.

All DataStage data types are supported except the following:

- BIGINT
- LONGVARBINARY
- LONGVARCHAR

All Teradata data types are supported except the following:

- Graphic
- Vargraphic
- Long Vargraphic

The following section describes DATE considerations for mapping to and from DataStage SQL data types and Teradata SQL data types.

DATE Data Type Considerations

If **DataFormat** is **FASTLOAD** (see [page 13](#), DATE data is converted from its DataStage internal format to a Teradata internal format and stored as a Teradata DATE data type. This is the default Teradata DATE data type specification for FastLoad jobs if you do not enter a DATEFORM command.

Teradata stores each date value as an integer in a DATE data type using the following formula:

$$(year - 1900) * 1000 + (month * 100) + day$$

If **DataFormat** is **VARTEXT**, DATE is converted from its DataStage internal format to a character date in YYYY-MM-DD format.

