Choose a Data Source

Use this dialog box to specify the source of the data to be copied. Not all the options defined below are available for all providers. A subset of the options below will be shown, depending on the provider chosen.

Options

Data Source

Choose the data-specific driver that matches the data storage format of the source data.

File Name

Specify the database path and file name holding the data to be imported (for example, C:\MyData.xls, \\Sales\Database\Northwind.mdb).

Username

Specify a user name for the database connection.

Password

Specify a password for the database connection.

Advanced

Display the **Advanced Properties** dialog box, where you can enter custom settings. For more information about the OLE DB provider properties, search in the Platform SDK section in the MSDN® Library at Microsoft Web site.

UDL Filename

Specify the name of the data link (.udl) file that contains the connection string.

Always read properties from UDL file

Request that the package search for and read the connection string from the specified data link (.udl) file each time the package is executed. Changes made to the data link file between different executions of the package will be incorporated on the next run. If you select this check box, the .udl file must be deployed with the package so the package can find it and read from it. If you do not select this check box, the connection string is copied from the .udl into the package, and the file is not referenced again. Connection changes then can be modified only by editing the Data Transformation Services (DTS) package directly.

Properties

Display the **Data Link Properties** dialog box, where you configure the data link connection. Changes made in the dialog box will be incorporated into the package created during the current session and will not change the data link file.

User/System DSN

Specify the name of the existing user or system data source name (DSN) that points to the data source.

New

Display the **Create New Data Source** dialog box, where you can create an ODBC DSN. For more information about creating a new ODBC data source, search in the Platform SDK section in the MSDN Library at Microsoft Web site.

File DSN

Specify the name of the existing file DSN that points to the data source.

Server

Specify the name of the server holding the data source.

Use Windows Authentication

Specify that the package use Windows Authentication for login to the Microsoft® SQL Server™ database.

Use SQL Server Authentication

Specify that the package use SQL Server Authentication for login to the Microsoft SQL Server database.

Database

List databases on the specified instance of SQL Server.

Refresh

Cause the database list to populate on Microsoft Windows® 98 computers.

See Also

DTS Connections

Choose a Destination

Use this dialog box to specify the destination of the data being copied. Not all the options defined below are available for all providers. A subset of the options below will be shown, depending on the provider chosen.

Options

Destination

Choose the data-specific driver that matches the data storage format of the destination.

File Name

Specify the path and file name of where the data being imported is to be stored (for example, C:\MyData.xls, \\Sales\Database\Northwind.mdb).

User name

Specify a user name for the database connection.

Password

Specify a password for the database connection.

Advanced

Display the **Advanced Properties** dialog box, where you can enter custom settings. For more information about the OLE DB provider properties, search in the Platform SDK section in the MSDN® Library at Microsoft Web site.

UDL Filename

Specify the name of the data link (.udl) file that contains the connection string.

Always read properties from UDL file

Request that the package search for and read the connection string from the specified data link (.udl) file each time the package is executed. Changes made to the data link file between different executions of the package will be incorporated on the next run. If you select this check box, the .udl file must be deployed with the package so the package can find it and read from it. If you do not select this check box, the connection string is copied from the .udl into the package, and the file is not referenced again. Connection changes then can be modified only by editing the Data Transformation Services (DTS) package directly.

Properties

Display the **Data Link Properties** dialog box, where you configure the data link connection. Changes made in the dialog box will be incorporated into the package created during the current session and will not change the data link file.

User/System DSN

Specify the name of the existing user or system data source name (DSN) that points to the data source.

New

Display the **Create New Data Source** dialog box, where you can create an ODBC DSN. For more information about creating a new ODBC data source, search in the Platform SDK section in the MSDN Library at Microsoft Web site.

File DSN

Specify the name of the existing file DSN that points to the data source.

Server

Specify the name of the server to hold the data.

Use Windows Authentication

Specify that the package use Windows Authentication for login to the Microsoft® SQL Server™ database.

Use SQL Server Authentication

Specify that the package use SQL Server Authentication for login to the Microsoft SQL Server database.

Database

List databases on the specified SQL Server.

Refresh

Cause the database list to populate on Microsoft Windows® 98 computers.

See Also

DTS Connections

Advanced Connection Properties

Use this dialog box to set custom values for certain OLE DB initialization properties. The OLE DB advanced properties available for customization varies depending on the properties supported by a particular provider. For more information, see the programming documentation for the individual OLE DB provider.

To change an OLE DB property value, in **Value**, click a cell, and then enter the new value.

Note If you need to work more closely at the OLE DB level, the OLE DB Rowset Viewer, available with the OLE DB Software Development Kit (SDK), offers a simple way to view and manipulate OLE DB rowsets. It also gives you the added ability to call and manipulate other OLE DB methods from the data source, session, command, rowset, transaction, and notification objects supported by any OLE DB provider.

See Also

DTS Connections

Create Database

Use this dialog box to define a new database for a data source connection. You can define a new database only when connecting to an instance of Microsoft® SQL Server™ through the Microsoft OLE DB Provider for SQL Server or through the Microsoft OLE DB Provider for ODBC. The database files will be put into the same location as your **master** database files. If you are unable to create a new database, make sure your login has the appropriate permissions.

This dialog box offers a subset of the available options for creating a database. For access to all the options when creating a database, in SQL Server Enterprise Manager, right-click the **Databases** node and then click **New Database**.

Make sure to follow SQL Server naming conventions for the database and appropriately set the data file size and log file size for the new database. For more information, see Naming Conventions for Instances of SQL Server 2000.

See Also

Creating a Database

Select File Format

Use this dialog box to specify the formatting of a source or destination text file. Columns and rows in the text file may be of fixed length or delimited with special characters.

Note If importing from an instance of Microsoft® SQL Server™ to a text file, column sizes may default to 2 gigabytes (GB). If you need to adjust the column size setting, click **Transform** in the **Select destination file format** dialog box to modify columns. When copying data from a text file to a new SQL Server table, the column sizes may default to **varchar** (255).

Options

Delimited

Specify that data within the file is aligned into fields and that each field is delimited with the same terminating character.

Fixed Field

Specify that data within the file is aligned into fields of equal width. A field within the file has the same width for all rows of data. However, each field can have a width different from other fields within the same row.

File type

Specify file type by clicking **ANSI**, **OEM**, or **Unicode**, depending on the type of data in the file. If you click **ANSI** or **OEM**, the data will be interpreted as belonging to the code page that is current on the computer executing the package.

Row delimiter

Specify that each row in the file is separated from the next with a character sequence. Click one of the following: **{CR}** (carriage return); **{LF}** (line feed); **Semicolon**; **Comma**; **Tab**; **Vertical Bar**. You also can type in a character to use as the row delimiter. You can also click <**none**> if the file is a fixed field file.

Text qualifier

Specify which character marks were used in the delimited data file to qualify text. Click one of the following: **Double Quote {"}**; **Single Quote {'}**; **<none>**. You also can type in a character to use as the text qualifier. If the text was not delimited, but the file is not a **Fixed field** file, then this property can be left to the default of **Double Quote {"}** or changed to **<none>**.

Skip rows

Specify the number of rows from the start of the file that you do not want copied. This field works together with the **First row has column names** field. If the **First row has column names** check box is not selected, then the number of rows skipped is equal to the number typed here. If the **First row has column names** check box is checked, then the number of rows skipped starts counting after the first row of column headings has been counted.

First row has column names

Specify that the first row in the text file has column headings rather than data.

Transform

Display the **Column Mappings and Transformations** dialog box, where you can customize the mapping of columns and add transformations to the copied data.

Define Row Width

Use this dialog box to define the row width for data in a text file when the file does not contain delimiters. To display this dialog box, you must clicked **None** in the **Row delimiter** list when configuring the file format.

To specify the number of characters you want in a row (shown in the **Value** box), select the red line and drag it, Alternatively, you can click on the arrows alongside the **Value** box to set the row width.

Specify Table Copy or Query

Use this dialog box to specify whether you want to do a simple copy of data or a more complex copy of data that requires an SQL statement to gather and select the appropriate rows to copy.

Options

Copy table(s) and view(s) from the source database

Display the **Select Source Tables and Views** dialog box, where you can copy fields from the selected tables and views in the source to the specified destination(s). The records will not be filtered or ordered.

Use a query to specify the data to transfer

Display the **Type SQL Statement** dialog box, where you can build SQL statements to retrieve selected rows. Only the rows matching the selection criteria will be available for copying.

Copy objects and data between SQL Server databases

Display the **Select Objects to Transfer** dialog box, where you can specify both objects and data to copy, if both the data source and destination are Microsoft® SQL Server™ databases. The objects you can transfer include tables, views, stored procedures, defaults, rules, constraints, user-defined data types, logins, users, roles, and indexes. You can transfer objects only between multiple instances of SQL Server version 7.0, from an instance of SQL Server 7.0 to an instance of SQL Server 2000, and between multiple instances of SQL Server 2000.

Type SQL Statement

Use this dialog box to type the SQL statement that will generate the data you want from the data source for copying to the destination.

Options

Query Builder

Display the **Select Columns** dialog box, where you can determine the tables, columns, and rows to be selected for copying.

Parse

Check the SQL statement in the Query statement text area for valid syntax. This option does not allow you to verify data fields or the existence of tables and views.

Browse

Display the **Select File** dialog box, where you can select a file. When a file is selected, the text from the file is copied into the Query statement text area.

Select Objects to Copy

Use this dialog box to specify which objects to transfer from one instance of Microsoft® SQL Server™ to another. You can transfer only between multiple instances of SQL Server version 7.0, from an instance of SQL Server 7.0 to an instance of SQL Server 2000, and between multiple instances of SQL Server 2000.

Options

Create destination objects

Create destination objects for all objects (tables, views, stored procedures, defaults, rules, constraints, user-defined data types, logins, users, roles, and indexes) to be transferred. Enable the following options:

Drop destination objects first

Drop all corresponding destination objects before creating new ones.

Include all dependent objects

Include all dependent objects, such as the tables supporting a view, in the transfer of data.

Include all extended properties

Include all extended properties, which are user-supplied definitions on various objects in the database. For more information, see Using Extended Properties on Database Objects.

Copy data

Enable the copying of SQL Server data from source to destination and the following copy options:

Replace existing data

Overwrite existing data in the destination objects with the new data from the specified source.

Append data

Retain existing data in the destination object and append new data from the specified source.

Use Collation

Enable the copying of data between different collations. For more information on using different collations with Data Transformation Services (DTS), see Data Conversion and Transformation Considerations.

Copy all objects

Transfer all objects associated with the specified data source.

Use default options

Set the advanced transfer options to their defaults.

Select objects

Display the **Select Objects** dialog box, where you can select individual objects to be transferred. Clear the **Transfer all objects** check box to make this option available.

Options

Display the **Advanced Transfer Options** dialog box, where you can further specify the transfer of logins and tables, and whether to use quoted identifiers in the generated SQL Script. Clear the **Use default options** check box to make this option available.

Script file directory

Specify the directory to which the script file and log files are written.

The SQL and data files for each selected object have script files written to a directory, which must exist on the same computer on which the task runs. These script files are then run against the destination database.

The script and log files remain accessible in the directory until the next time the package is executed. At that time, the files are overwritten with new information. Because the script files are being executed in this manner, running multiple transformations from the same source database at the same time will entail changing the directory where the script file is being stored for one of these transformations. If the script file directory is not changed, one process will not be able to access the file because it is in use.

See Also

Copy SQL Server Objects Task

Select Objects

Use this dialog box to specify the objects that you want to transfer to the new database.

Note If you select one of the options below, you do not mark automatically the objects to be transferred. Instead, you cause the objects to appear in the **Objects** table. From that table, you then select the objects to transfer.

Options

Show all tables

View all tables in the database in the **Objects** table.

Show all views

Display all views in the database in the **Objects** table.

Show all stored procedures

Display all stored procedures in the database in the **Objects** table.

Show all user-defined functions

Display all user-defined functions in the **Objects** table.

Show all defaults

Display all database defaults in the **Objects** table.

Show all rules

Display all database rules in the **Objects** table.

Show user-defined data types

Display all user-defined data types in the database in the **Objects** table.

Objects table

Select items individually to include them in the transfer or click the following:

Select All

Select all objects in the database as eligible for transfer to the new database.

Check

Select one or more database objects.

Uncheck

Remove one or more database objects from selection.

See Also

Copy SQL Server Objects Task

Advanced Copy Options

Use this dialog box to specify which objects to copy from one instance of Microsoft® SQL Server™ to another.

Options

Copy database users and database roles

Copy all database users and roles.

Copy SQL Server logins (Windows NT and SQL Server logins)

Copy all SQL Server logins.

Copy object-level permissions

Copy all object-level permissions.

Copy indexes

Copy indexes for all tables copied, if applicable.

Transfer triggers

Copy triggers for all tables copied, if applicable.

Copy full text indexes

Copy full-text indexes for all tables copied, if applicable.

Copy PRIMARY and FOREIGN keys

Copy primary and foreign key definitions for all tables copied.

Generate Scripts in Unicode

Copy data as Unicode. Useful if source data contains **nchar** data types or double-byte character set (DBCS) data. For more information about Unicode, see Unicode Data.

Use quoted identifiers when copying objects

Enclose all object names in quotation marks.

See Also

Copy SQL Server Objects Task

Select Columns

Use this dialog box to specify tables, views, and columns you want copied to the destination and to order columns in the destination tables.

Options

Source tables

View the tables to be copied. To move all the columns to the **Selected columns** text area for copying to the destination, select the table and click the **>** button, or double-click the table name. You also can select and move individual columns. To see the individual columns, expand the table that contains the column.

Selected columns

View the columns to be copied. The columns are listed in the order in which they will be created.

Move Up

Move the selected column up in the order.

Move Down

Move the selected column down in the order.

Specify Sort Order

Use this dialog box to order the records to be placed as they are placed in the destination table.

Options

Selected columns

View the columns to be copied into the destination table. To see the column in the **Sorting order** table, highlight a row and double-click, or click the **>** button.

Sorting order

View the columns to be used as the sorting criteria. An ORDER BY clause will be generated from the chosen columns. The ORDER BY clause will be followed by the column names shown in the **Sorting order** table, with the first column as the outer sort field, the next column as the inner sort, and so on. As a result, the records will be placed in the destination table in order of the specified sort. The ordering does not generate a clustered index or affect indexing.

Specify Query Criteria

Use this dialog box to specify whether a WHERE clause is used to filter out rows of data. If you want to use filtering, use a combination of the **Column**, **Oper**, and **Value/Column** fields to write the predicate. If more than three conditions are necessary, type in the additional criteria in the **Type SQL Statement** dialog box that is displayed when you click **Next**, or click **Back** to return to the previous dialog box and type the SQL statement directly.

If you enter multiple conditions that are combined with different logical operators (**AND**, **OR**), the rules of precedence defined for that database management system (DBMS) apply. In most DBMS installations, the **AND** condition is evaluated first.

Options

All rows

Select all rows from the source without constraint. No WHERE clause will be appended to the SQL statement.

Only Rows meeting criteria

Enable the **Column**, **Oper**, and **Value/Column** fields so that the following selection criteria can be defined:

Column
Choose the column to which the constraint will apply.

Oper
Choose the relationship operator to apply in the comparison. Available values are =, <, >, <>, <=, >=.

Value/Column
Select the other value or column to which the field in the **Column** list is compared. Click the browse (**...**) button to list the values stored in the table for the column shown in the **Column** list. The value selected from

the list becomes the comparison value.

Select Source Tables and Views

Use this dialog box to specify the tables and view to be copied to the destination.

Options

Table(s) and View(s)

View the tables and views that are available for copying to the destination. If you select a source and perform no other action, you will copy the schema and data from the source without changes.

...

Display the **Column Mappings and Transformations** dialog box, where you specify which destination columns are to receive source data, edit script code to customize the transfer at the record level, and edit the SQL code required to create the destination table.

Preview

Preview the source data to verify it before running the Data Transformation Services (DTS) package. Click **Back** to return to previous dialog boxes, where you can make corrections that will produce the appropriate data.

Column Mappings and Transformations

Use this dialog box to specify the mappings between the source and destination columns and to specify the transformations the data goes through, if any, as it is moved to the destination. You also can change the data type of the data if a valid data conversion is applicable.

Column Mappings Tab

Use this tab to specify how to treat the destination table and to specify column attributes on any newly created table.

Options

Create destination table

Create the destination table before copying the source data. This is the default option if the destination table you specified does not exist. If this option is selected and the destination table already exists, an error occurs, unless you select the **Drop and re-create destination table** check box.

Edit SQL

Customize the Transact-SQL CREATE TABLE statements used to create the destination table.

Delete rows in destination table

Delete all rows in the destination table before copying the source data. This option is only available if the destination table already exists. Existing indexes and constraints on the destination table are not affected by this option.

Append rows to destination table

Insert source data into the destination table. This option is available only if the table already exists. Existing data, indexes, and constraints on the destination table are not affected. However, rows are not necessarily appended to the end of the destination table. You can determine where rows will be inserted only by having a clustered index on the destination table.

Drop and re-create destination table

Drop the destination table and re-create it before moving data into it. All existing data in the destination table and any indexes are destroyed.

Enable identity insert

Allow explicit values to be inserted into the identity column of a table (Microsoft® SQL Server™ only). This option is only available if an identity column is detected. An identity column is a column that has been defined as having a data type of **bigint**, **decimal**, **integer**, **numeric**, **smallint**, or **tinyint**, where the **Identity** property for the column is set to yes.

Mappings table

Edit the cells in the table. The table does not list the order in which the destination columns will appear, but rather lists each destination column alongside each source column to which it maps.

For each column in the **Mappings** table, you can set the following properties:

Source

Choose the column name in the source table to copy to the destination. Click **<ignore>** on the source column name to set the destination column to NULL for a new table. If the table already exists, the data will be NULL if allowed, or set to its default value if one was defined. If the destination is defined as NOT NULL, clicking **<ignore>** in the source column results in an error when the package is executed if no default value is specified for the destination column.

If you copy a source column defined as an identity column, the data will not be copied unless the **Enable identity insert** check box is selected.

If you copy a source column defined with user-defined data types to a new table, the data type of the destination column is the system-defined data type that corresponds to the user-defined data type.

You may change the following properties only if a new destination table is being created.

Destination

Choose the column name in the destination table to receive the source data. Click <**ignore**> to prevent the source column from being created in the destination table when creating a new table. If you want to remove a column from an existing table, click <**ignore**> in the destination column, but also select the **Drop and re-create destination table** check box in this dialog box. This will re-create the table with the columns specified. Existing data and indexes in the table will be lost.

timestamp columns cannot be copied by the Data Transformation Services (DTS) Import/Export Wizard. If the destination column is a **timestamp** column, the value in that field will be a new timestamp indicating when the row was inserted, not a copy of the timestamp data from the source table.

Type

Select a data type for the destination column. The default setting matches the data type of the destination column to the source.

Note Invalid data conversions can be specified without causing an error, as the default transformation setting allows all possible conversions. For example, converting an **int** data type to a **tinyint** data type will result in data truncation, but the conversion will proceed.

Nullable

Specify if destination can allow null values.

Size

Specify the length of the **Destination** column, in units corresponding to the data type. The value is only applicable for the **char**, **varchar**, **nchar**, **nvarchar**, **binary**, and **varbinary** data types. Specifying a size smaller than the length of the source can result in data truncation.

Precision

Enter the maximum number of decimal digits that can be stored to the left and to the right of the decimal point. This option applies only to **decimal** and **numeric** data types.

Scale

Enter the maximum number of decimal digits that are stored to the right of the decimal point. The number must be less than or equal to the number in the **Precision** column. This option applies only to **decimal** and **numeric** data types.

Transformations tab

Use this tab to specify whether unique transformation code must be written. The column mappings set on the **Column Mappings** tab are reflected in code shown in the text area.

Options

Copy the source column directly to the destination columns

Copy the source column to the destination tables without changing the data or the source and destination mappings.

Transform information as it is copied to the destination

Edit the script in the text area to customize the columns before copying them from the source to the destination. For complex transformations, consider using DTS Designer, which offers support for Microsoft ActiveX® scripting.

Language

Select a scripting language. The default language is Microsoft Visual Basic® Scripting Edition (VBScript). Available languages are **VBScript** or **JScript**.

Browse

Display the **Select File** dialog box, where you can select a file that contains previously written transformation code, if the code already has been written and saved to a .txt, .vbs or .bas file. When a file is selected, the text from the file is copied into the Query statement text area.

See Also

ActiveX Script Transformation

Using ActiveX Scripts in DTS

Save, Schedule, and Replicate Package

Use this dialog box to save the source, destination, and transformations properties as a Data Transformation Services (DTS) package.

Options

Run Immediately

Run the transformation immediately and create the destination data when the wizard completes

Use replication to publish destination data

Use the destination tables for replication. When you click **Use replication to publish destination data**, the Create Publication Wizard starts after the DTS Import/Export Wizard completes.

Schedule DTS package for later execution

Save the package to the Microsoft® SQL Server™ **msdb** database, SQL Server 2000 Meta Data Services, a COM-structured storage file, or a Microsoft Visual Basic® file and schedule it to run at predefined intervals. Click the browse (**...**) button to display the **Edit Recurring Job Schedule** dialog box, where you can schedule the execution of a package. If the schedule is not modified, the default is to run the package daily at 12:00 midnight.

Save DTS Package

Specify that you want to save the DTS package to one of the following formats:

SQL Server
Save the package to SQL Server and store it in the **sysdtspackages** table of the **msdb** database.

SQL Server Meta Data Services
Save the package to Meta Data Services. Use this option if you plan to

track package version, meta data, and data lineage information with Meta Data Services.

Structured Storage File
Save the package as a COM-structured storage file.

Visual Basic File
Save the package into a Visual Basic file.

See Also

Introducing Replication

Saving a DTS Package

Save DTS Package

Use this dialog box to save a Data Transformation Services (DTS) package.

Options

Name

Specify a unique name for the package.

Description

Specify a description for the package.

Owner password

Specify a password for the package to protect any sensitive user name and server password information in the package from unauthorized users. This option is not available if you save to Microsoft® SQL Server™ 2000 Meta Data Services or to a Microsoft Visual Basic® file.

User password

Specify a password for a package user. This password allows a user to execute a package; however, this option does not allow a user to view the package definition. If you set the user password, you also must set the owner password. This option is not available if you save to Meta Data Services or to a Visual Basic file.

Server name

Specify the name of the SQL Server installation storing the package.

Use Windows Authentication

Use Windows Authentication when saving a package to an instance of SQL Server. The Windows Authentication used will be the Microsoft Windows® login of the developer creating the package. If the package is scheduled through SQL Server Agent, the SQL Server Agent service must be run under an account that has permissions to access all the resources.

Use SQL Server Authentication

Use SQL Server Authentication when saving a package to an instance of SQL Server. If the package is scheduled through SQL Server Agent, the SQL Server Agent service account must have access to all the resources required by the package.

User name

Specify the login used to connect to an instance of SQL Server if you clicked **Use SQL Server Authentication**.

Password

Specify the password associated with the user name if you clicked **Use SQL Server Authentication**.

Scanning

Display the **Scanning Options** dialog box, where you can specify how objects referenced by the package should be scanned into Meta Data Services. This capability allows you to relate source and destination objects in a package to database meta data (for example, primary and foreign keys in a table, indexes, and column information such as data type) stored in Meta Data Services.

When you save the package as a COM-structured storage file or a Visual Basic file, the following options are available:

Name
Specify a unique name for the package.

Description
Specify a description for the package.

File name
Specify the package source code file name and path. Click the browse (**...**) button to display the **Save As** dialog box.

See Also

Saving a DTS Package

View Data

Use this dialog box to inspect source data that will be used in the data copying operation. Only the first 100 lines of data are displayed.

Font

Use this dialog box to format the Transact-SQL code in the query box.

Options

Color

Specify a Transact-SQL content element to assign color coding and formatting.

Foreground
Assign a color for the selected content element.

Background
Assign a background color for the selected query element.

Font

Assign a font to the selected Transact-SQL content element.

Size

Assign a font size to the selected Transact-SQL content element.

Sample

View an example of the selected Transact-SQL content element with the assigned font attributes.

Reset All

Reset all options to their original default values.

Scanning Options

Use this dialog box to import database schema information from an OLE DB data source and populate instances of the Database Information Model (DBM) in Microsoft® SQL Server™ 2000 Meta Data Services. The scanner is passed an OLE DB provider, and then it examines the schema and creates a set of corresponding instance objects in Meta Data Services using the DBM and Data Transformation Services (DTS) Meta Data Services information models.

If you save a DTS package to Meta Data Services, you can set how the objects referenced by the package are related to SQL Server catalog meta data scanned into Meta Data Services. Meta data in this context refers to information such as:

- Primary and foreign keys.
- Column type, size, precision, scale, and nullability.
- Indexes.

Options

Resolve package references to scanned catalog meta data

> Enable the scanning options, which link the package to Meta Data Services meta data.

Use scanned catalogs if already present in repository

> Relate the package to Meta Data Services meta data if the meta data has already been saved (for example, by importing the meta data or by a previous scan). This option is useful if the database schema has not changed, Meta Data Services meta data already exists, and saving time is a consideration. If column or meta data information has changed and this option is used, references of the package to Meta Data Services meta data may not be meaningful.

Scan all referenced catalogs into repository

Scan all information about a database (all table, column, and meta data information) into Meta Data Services, even if the information is not used. This is safer than the previous option, but more costly in terms of time and performance.

Scan catalog if not already present in Repository

Add the database if it is not present in Meta Data Services. This is the default selection if scanning options are enabled.

Scan catalog always

Scan a database into Meta Data Services even if it is present already.

See Also

Importing and Saving Meta Data in DTS

Recording Data Lineage in DTS