PeopleSoft

EnterpriseOne Xe Table Conversion PeopleBook

September 2000

J.D. Edwards World Source Company
7601 Technology Way
Denver, CO 80237

SKU XeEATC

Table of Contents

Table Conversions Overview

Table conversions are a type of batch process that allows you to do high-speed manipulation of data in tables. The table conversion tool includes four conversion types that allow you to do a variety of data manipulation:

- Data Conversion, which allows you to transfer or copy data from an input table or business view into one or more output tables using any amount of logic necessary to perform the transfer. You can also use Data Conversion to update records in a table or business view.
- Data Copy, which allows you to copy one or more tables from one data source or environment to another data source or environment.
- Data copy with table input, which allows you to copy tables based on information from an input table. For example, the input table might provide information about which tables should be copied, where they should be copied, and so on.
- Batch delete, which allows you to delete records from a table or business view.

The table conversion tool can make use of any OneWorld tables, business views and text files, or any tables that are not OneWorld tables but reside in a database supported by OneWorld, such as Oracle, Access, AS/400, or SQL Server. These non-OneWorld tables are commonly referred to as foreign tables.

When you create a table conversion, you set up the conversion (which can be saved and run multiple times), review it, and then run it. If necessary, you can test the conversion by running it in proof mode.

Like reports, table conversions consist of a template and one or more versions. You can override certain properties within a version at run time.

The table conversion tool allows you to access any available environment, both for input and output. The environments that you choose determines which tables and business views are available for the conversion and where the tables reside. The specification, or description, of tables and business views is also determined by the environments that you choose.

Types of Tables You Can Convert

You can use the following table types in table conversion:

Standard OneWorld tables	These tables exist in Object Librarian, and you design and edit them using Table Design Aid. At design time, only the specification is needed to reference the table. At run time, the table must be generated; an instance of that table must be made in a specific database.
Non-OneWorld tables (foreign tables)	These tables do not have a OneWorld definition, but they reside in a database supported by OneWorld at design time and run time. You must set up a data source and environment in OneWorld to point to the location of a non-OneWorld table. See *Preparing Non-OneWorld Tables for Table Conversion* for more information.

Using Business Views in Table Conversion

If you are transferring data from multiple tables to a single table, or if you are transferring data from multiple tables to multiple tables, you must establish a relationship between the input tables by defining a business view. A business view defines the relationship between two or more tables, and the data is joined together into what looks like a single table. You can use OneWorld business views *only* for input to the table conversion, not for output. The system does not provide direct support for joining foreign tables. If you need to use multiple non-OneWorld tables as input to a conversion, you must first define them through OneWorld and then create a business view over them.

Using Text Files in Table Conversion

You can import directly from, or export directly to, a text file as well. When you convert it, the text file is stored similarly to a foreign table, with a single long text field. User-defined formats are stored the same for a text file as for any table. With a text file conversion, the table name includes the path and the file name. If you do not specify the path with the file name, the default path will be used.

Using Sort and Selection Criteria in Table Conversion

You can specify sort criteria in a table conversion to process input rows in a sequence that groups related records together. The table conversion tool allows you to add logic to the point at which a change occurs in a field's value. The sort and selection features simplify the process of writing records to multiple tables in a typical one-to-many conversion. For example, if you had a table of

customer information, you could sort the table by area code and split the single table into tables for each area code. Similarly, you could specify selection criteria for the input table if you want to convert only a subset of that table.

Understanding Input and Output Environments

An environment consists of a path code and one or more Object Configuration Management (OCM) mapping records that indicate where various OneWorld objects reside. The table conversion tool allows you to specify an input and output environment, and uses the environments you specify to locate input and output tables. To locate non-OneWorld tables, the table conversion tool uses the default OCM mapping records for tables.

The environment's path code is used to locate specification files for the environment. This path code is usually a subdirectory under the OneWorld directory on your workstation. To reference OneWorld tables in an environment, the full path code must exist on the machines where the conversion is designed and run. Non-OneWorld tables can be referenced even if the path code does not exist.

The table conversion tool uses three environments when it processes a conversion:

- The environment that you are signed into
- The environment for the input tables
- The environment for the output tables

The environment that you are signed into determines where the table conversion specifications are stored. You can select an environment for your input table or view, and one for your converted output tables. All three of these environments could be the same, or they could all be different.

When you use OneWorld tables or views in a conversion, the environment provides the details of each table or view, such as the column names, data types, and descriptions. Because this information comes from the OneWorld specification tables, the OneWorld table or view does not have to exist in the database for you to design a conversion. It only needs to be created before the conversion is run. However, if you are using a non-OneWorld table as input, you need to create the foreign table before you design the conversion, because the tool gets its information about the table directly from the database that contains the non-OneWorld table. The environment also provides a default path for text files

Table conversion includes the following tasks:

- ❑ Setting up a table conversion
- ❑ Running a table conversion

❑ Preparing non–OneWorld tables for table conversion

See Also

- *Table Design* in the *OneWorld Development Tools* guide for more information about creating standard OneWorld tables
- *Preparing Non-OneWorld Tables* in this guide for more information about using non-OneWorld tables as input to a conversion
- *CNC Implementation Guide* for more information about data sources, path codes, and environments

Setting Up a Table Conversion

You use a Director to set up a table conversion. The Director steps you through the process and allows you to modify the table conversion at each point. With the help of the Director, you can design conversions for converting data, copying tables between locations, and deleting records within tables.

You can also define user-defined formats (flat files) to use in the conversion. You define user-defined formats when you choose your input or output tables. User-defined formats are tables that store data as one continuous string of information, such as bank tapes. See *Understanding User-Defined Formats* for more information.

After you set up a conversion, you can save it and run it multiple times.

The following topics are described:

- ❑ The flow of events in table conversion
- ❑ Using the Table Conversion Director
- ❑ Converting data
- ❑ Copying data
- ❑ Copying data with table input
- ❑ Deleting records
- ❑ Reviewing your table conversion
- ❑ Using event rules in table conversions

Before You Begin

Before you begin setting up a table conversion, make sure you have completed the following tasks:

- ❑ If you are importing data from non-OneWorld tables, you must set up a data source and environment for those tables. For more information, see *Preparing Non-OneWorld Tables for Table Conversion* in this guide.
- ❑ If you are mapping from multiple tables, you must create a joined business view over the tables. For more information, see *Creating a Table Join* in the *Development Tools Guide.*

- If you want to validate the data items within a table against the data dictionary as part of the conversion process, you must create a business function to perform the validation. The table conversion tool does not provide automatic data dictionary validation for inputs or outputs. See *Business Functions* in the *Tools Guide* for more information about creating business functions.

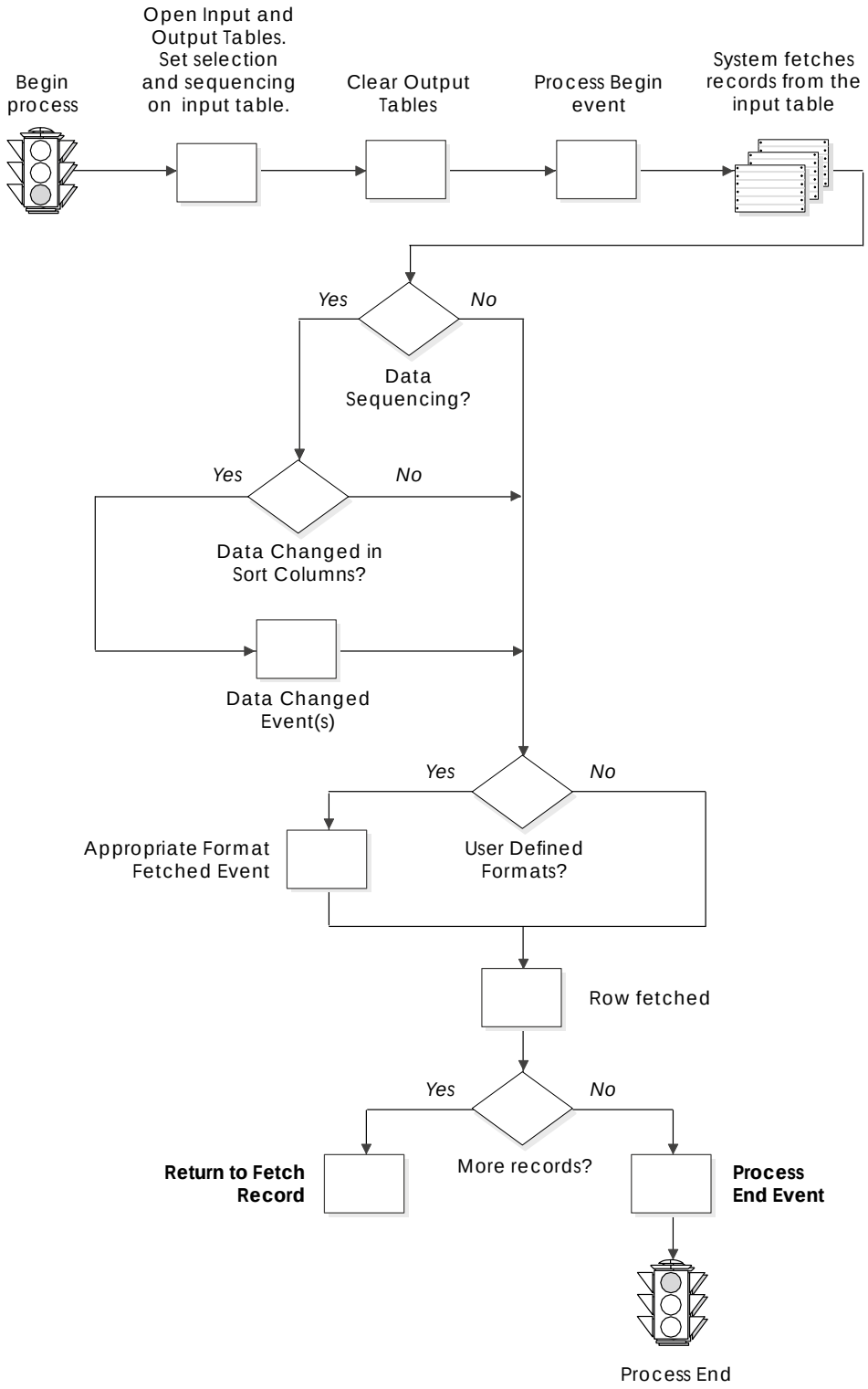
The Flow of Events in Table Conversion

When a table conversion is processed, the system triggers certain events, similar to the events that are triggered when a report or application is run. These events are specific to the conversion that you set up, and they also provide points where you can add logic to the conversion.

The event flow is generally the same for all table conversion types, such as Copy Data, Copy Data with Table Input, and Batch Delete because these conversion types are just subsets of a data conversion. For example, the Data Copy conversion type does not include input and output tables, and all actions are accomplished through the Process Begin event. Moreover, Data Copy with Table Input and Batch Delete conversion types do not include output tables and all actions are accomplished through the Process Begin, Process End, and Row Fetched events. This flexibility allows you to mix and match table conversion types within another conversion type, if necessary.

The graphic on the following page shows all events that can be triggered in a conversion. However, some events may not be triggered, depending on the type of the conversion you set up. In the graphic, events are shown as a yes or no decision.

Table Conversion Event Flow



As shown in the graphic, events in table conversion occur in the following order:

Process Begin

Before fetching any records from the input table, the system invokes the Process Begin event. At this point, you can attach any logic that processes only once at the beginning of a conversion, or any other value that does not change for each individual record. This event is useful for mapping output fields that do not change for each individual record.

Data Changed

If you use data sequencing, the system invokes a Data Changed event for any sequenced field that changed. Data Changed events are not cascaded or hierarchical. For example, you can attach an event rule to this event if you want to total a field or group of values.

Format Fetched

If you use user-defined formats (also known as flat files) on the input table, the system invokes a Format Fetched event for each record fetched from the input table. If you use multiple user-defined formats in a conversion, the Format Fetched event that is called will correspond to the format found in the record.

Row Fetched

An input table invokes a Row Fetched event after each row is fetched from the input table.

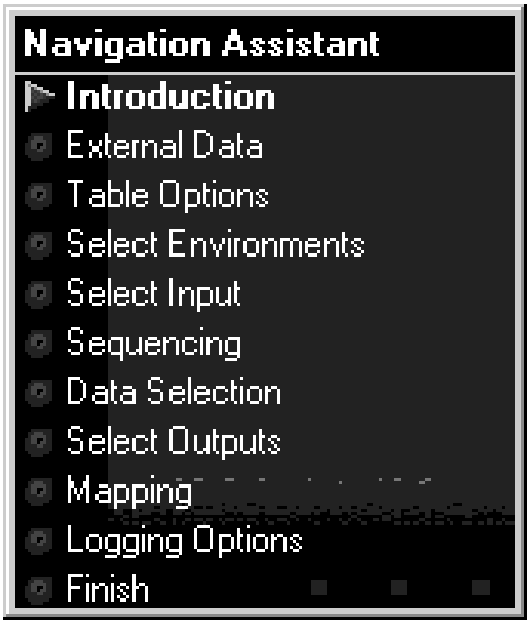
Process End

After all records have been processed, the system invokes the Process End event.

You attach event rules to Process End when you want the system to process logic after all input records have been read – for example, to write a total record to an output table, or to write a record to a log file to record the status of the conversion.

Using the Table Conversion Director

The Table Conversion Director steps you through the process of setting up a table conversion. The Navigation Assistant, which appears within the Director, provides a visual reminder of where you are throughout the the setup process. You can also use the Navigation Assistant to move to a different step in the process by clicking any step listed in the Assistant.



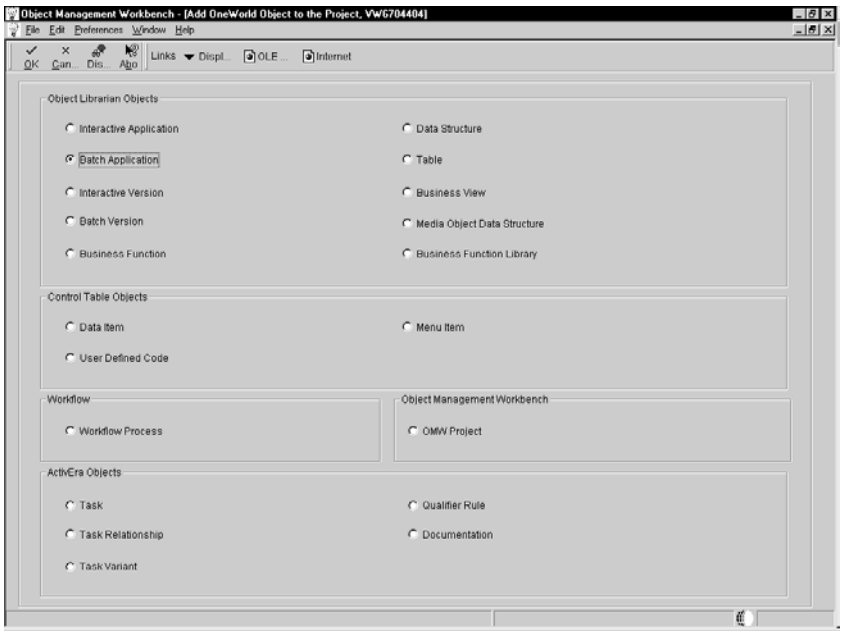
▶ To use the Table Conversion Director

1. From Cross Application Development Tools (GH902), choose Object Management Workbench.

   The Object Management Workbench appears.

2. Click Find.
3. Click the project to which the new batch process will be added, and then click Add.

   The Add OneWorld Object to the Project form appears.



4. Select Batch Application and click OK.

   The Add Object form appears.

5. On Add Object, complete the following:

   - Description
   - Product Code
   - Product Code/Reporting
   - Function Use

6. Click the Table Conversion option, and then click OK.

   The Batch Application Design form appears.

7. Click the Design Tools tab, and then click *Start Report Design Aid.*

   The system displays the Introduction form for the Table Conversion Director. Depending on the conversion you want to perform, follow the steps as described in:

   - *Converting Data*
   - *Copying Data*
   - *Copying Data with Table Input*
   - *Deleting Records*

Converting Data

You use the Data Conversion option on the Table Conversion Director when you want to move data to one or more tables from:

- A single table
- Multiple tables defined through a business view
- A single text file

The Director leads you through a linear process for creating a data conversion batch application by asking questions about its structure and function. When you are finished, you can review and alter the conversion, if necessary.

- Defining external data
- Defining input and output environments
- Defining input
- Defining output
- Mapping input to output
- Choosing logging options
- Reviewing the results of the director
- Understanding user-defined formats
- Defining user-defined formats

Defining External Data

In some cases, you may need to apply a processing option template or a data structure to external data before it is converted.

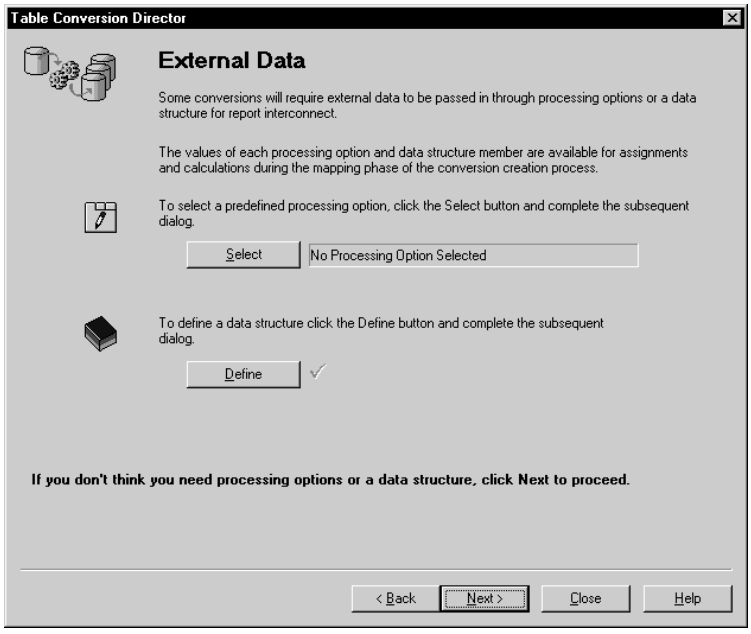
Before You Begin

❑ Create a batch application object. See *Using the Table Conversion Director* for information about starting the data conversion design process and the Director. The last step launches the Director.

To define external data

1. On the Introduction form of the Table Conversion Director, choose the Data Conversion option and click Next.

   The External Data form appears.



2. If you want to attach a predefined processing option template to the table conversion, click Select.

   The Select Processing Option Template form appears.

3. On Select Processing Option Template, find and choose the processing option you want to use, and click OK.
4. If you want to attach data structures, click Define on the External Data form.

   The Report Data Structures form appears.

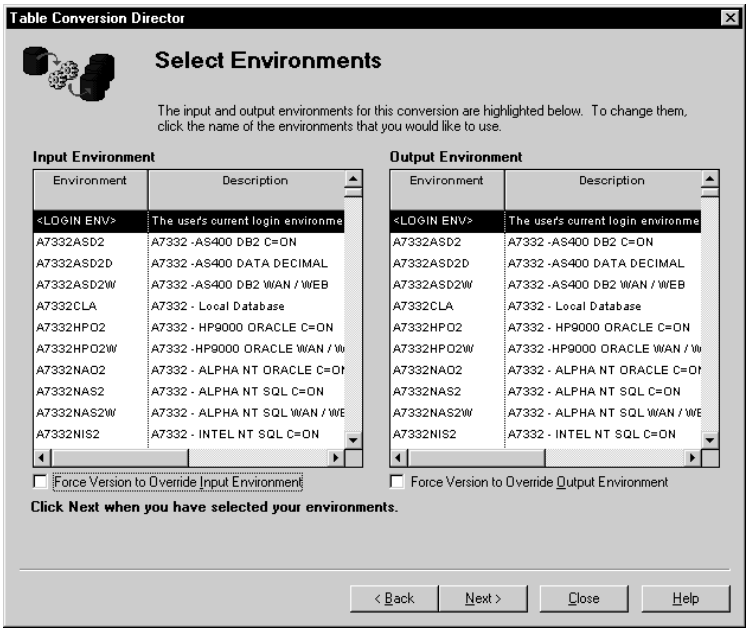
   Data structures contain a list of parameters that can be used to pass data into the conversion when called through Report Interconnect. See *Data Structures* in the *OneWorld Development Tools* guide for more information.

5. Define the data structures that you want to attach to the table conversion, and then click OK.
6. Click Next.

Defining Input and Output Environments

You may define a different environment for the input and the output.

After you click Next from External Data as described in the previous task, the Select Environments form appears.



▶ To define input and output environments

1. On Select Environments, choose the input and output environments that you want to use.

   Note: Choose <LOGIN ENV> if, for example, you are creating a table conversion on your workstation that will be shipped to a client who does not have the environments that you have, and the environment they log into will always be appropriate.

2. Choose the "Force Version to Override Input Environment" or "Force Version to Override Output Environment" if you are creating a table conversion that will run in a different environment than the one in which you are creating it, and the <LOGIN ENV> is not appropriate for the type of conversion that you are creating.

   For example, if you are creating a conversion that will be shipped to a client who does not have the environments that you have, you would choose the "Force Version to Override" option. When the conversion is invoked at the client site, the system will not run the conversion until the user chooses an appropriate environment in which to run it.

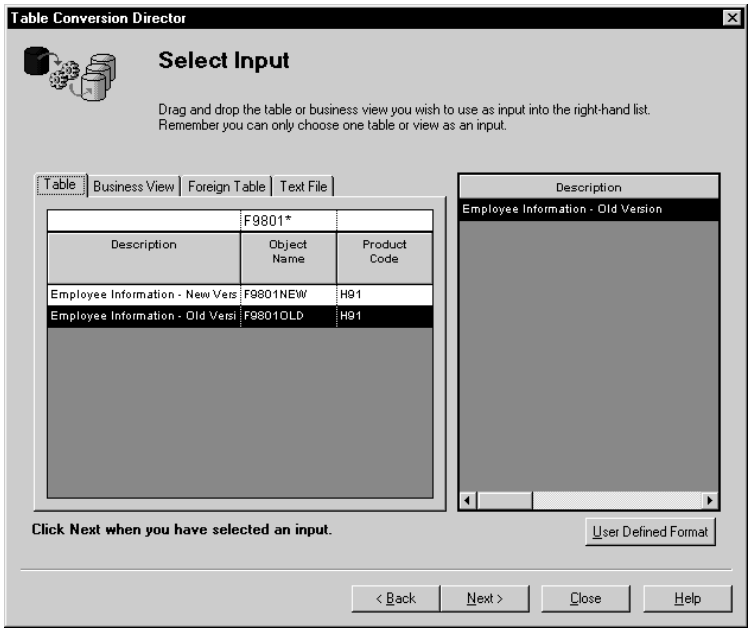
3. Click Next.

Field	Explanation
Force Version to Override Input	If you choose this option, the conversion will not run unless you creates a version that overrides the input environment.
Force Version to Override Output	If this option is chosen, the conversion will not run unless the user creates a version that will override the output environment.

Defining Input

Conversion input may originate from a table, business view, or text file. You can select only one input object.

After you click Next from Select Environments as described in the previous task, the Select Input form appears.



▶ To define input

1. On Select Input, drag the table or business view to the column on the right. You can choose only one table or one business view per conversion. If your input consists of multiple tables, you must create a single, joined business view.

   For a text file conversion, choose the file on the Text File tab.

Tip: If you know the name of the table or business view that you want to use, enter the name in the Name field in the QBE (query by example) row and press Enter. Or for text files, you can select a file from the default directory, enter a new file name, or click the Browse button to locate a file.

Note: If you change the table, business view, or file, the system warns you that deleting tables removes all mappings from the table conversion.

2. If you are using a text file, or if you need to define a user-defined format for a table or business view, click User-Defined Format and follow the steps described in *Defining User-Defined Formats.* When you complete those steps, return here.
3. To delete an input name, choose it and press Delete.
4. Click Next.

   The Sequencing form appears.

5. To define data sequencing for a table or business view, click Data Sequencing. If you specify a text file for input, you cannot define data sequencing or selection for that file.

   When you define data sequencing, you create new events that are available to you in the Mapping section of the Director. One new event is created for each of the sequence columns that you define. The event is called *XXXX* Data Changed, where *XXXX* is the column alias – for example, ALPH Data Changed. Each time the value in one of these columns changes from its previous value, the column's Data Changed event is invoked. This event is similar to a level break in report writing with the exception that the Data Changed events are not related to each other. Invoking one does not invoke the others.

   See *The Flow of Events in Table Conversion* for more information about the Data Changed event.

6. Click Next.

   The Data Selection form appears.

Note: You can only define selection criteria over database table columns. User-defined format columns are not available because they do not exist in the database.

7. On Data Selection, *Where* is the default value in the Operator column for the first set of criteria.

   For subsequent statements, *And* and *Or* become the available values for the Operator column and are selected by double-clicking the appropriate word.

8. Click in the Left Operand column to display the list of available objects, then do one of the following:

   - Scroll through the list until you find the desired object, choose the object, and then double-click the object to populate the Left Operand column.
   - Type in the first letters of the object name in the Left operand field to bring you to the object in the list, and then double-click the highlighted object.

   When you double-click the object for the Left Operand column, the list in the Comparison column automatically appears.

9. Select one of the following comparison operators:

- is equal to
- is greater than
- is greater than or equal to
- is less than
- is less than or equal to
- is not equal to

10. Click in the Right Operand column to display an available list of objects, special values, or variables. Your choices in this column depend on the choice that you made in the Comparison column. Some of the following options could be available:

Blank	Enters a blank (space) value
Literal	Enter specific values (see the following step for information on entering specific values)
Null	Indicates that no value is associated with the field
Zero	Enters a value of zero
IC	Indicates an input table column
RI	Indicates a value passed through report interconnections to this table conversion
PO	Indicates a processing options value for this report
SL	Indicates a system literal

11. If you chose to enter a literal in the Right Operand column, the form that opens automatically enables you to enter the following:

    - Single value

      Enter a single value, and then click OK. For example, a value might be for a particular company.

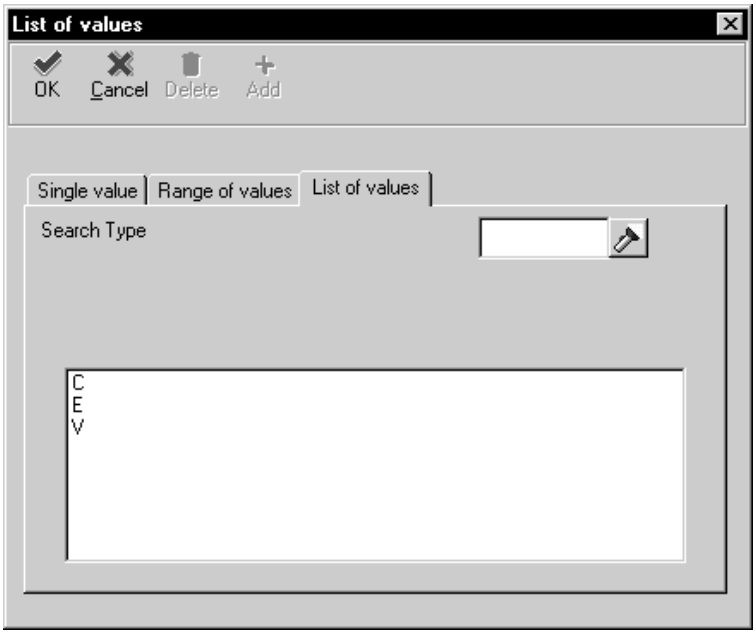
    - Range of values

      Enter a range of values, and then click OK. For example, a range of values might include companies from 00001 to 00060. Only *is equal to* and *is not equal to* are valid logical operators when using a range of values.

- List of values

  To add values to or remove values from the list, do the following:

  - Type each value in the field, and then press Enter or click the Add button at the top of the form.
  - Repeat this process until your list of values is complete. For example, a list of values might include several user defined codes for search types such as C for Customers, E for Employees, and V for Vendors. Only *is equal to* and *is not equal to* are valid logical operators when using list of values.
  - Delete a value by choosing the value, and then click the Delete button at the top of the form.
  - Click OK when you are finished.

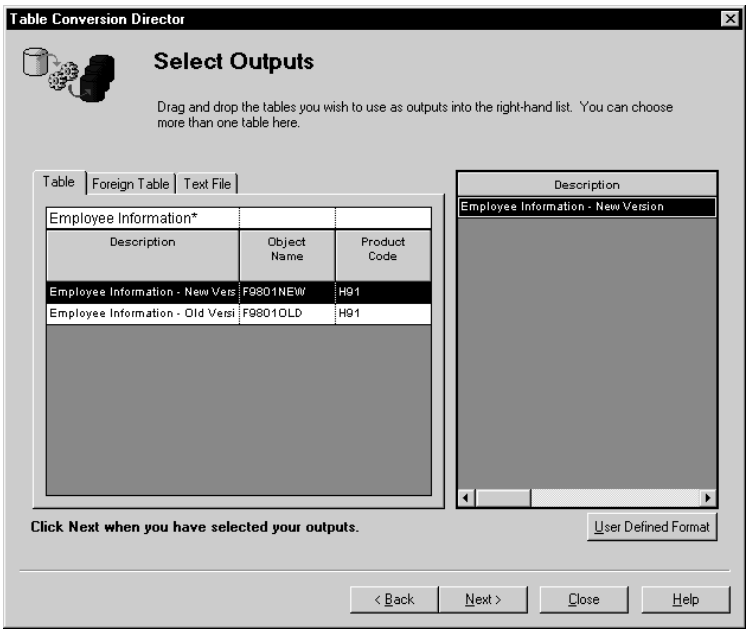


12. To delete a line of criteria on Data Selection, choose the row header to highlight the row, and then click the Delete button at the top of the form.
13. To change the order of the criteria, choose the row header to highlight the row, and then click the up or down button.
14. Click Next.

Defining Output

You can output to a table or text file. You can choose multiple outputs per conversion.

After you click Next from Data Selection as described in the previous task, the Select Outputs form appears.



To define output

1. On Select Outputs, drag the table or tables that you want to use as your outputs to the list on the right and click Next. For text file conversions, choose the file on the Text File tab.
2. To delete an output, choose the row and press Delete.
3. If you are using a text file, or if you need to define a user-defined format for a table or business view, click the User-Defined Format button and follow the steps described in *Defining User-Defined Formats.*
4. Click Next.

   The Table Options form appears.

5. On Table Options, click any of the following options, if applicable:

   - Currency Triggers

     Choose this option if the OneWorld table or tables contain currency triggers. If the tables contain currency fields and you do not choose this option, the system will not know how many decimal places exist in each column. Anytime that the source or destination fields are currency fields and you do not turn on the currency trigger, problems could arise if the value is used in a calculation. The system has no way to determine where the decimal should be within a field.

You might not want to choose the currency trigger option if the input and output data sources are of the same type (for example, Oracle, AS/400, or SQL Server), and no calculations are being performed. Furthermore, choosing the currency trigger option results in slower processing time.

In addition, you should not use currency triggers on an environment that has a different path code than the login environment.

- Clear Output
- Force Row by Row

  Choose this option if, for example, you want to test the table conversion, or if you want to ensure that the conversion always runs in row-by-row mode.

  You might want to test a conversion to ensure that the mapping logic will perform correctly. In this case, you would also want to specify the number of rows to process. You can either specify the number of rows to process in the jde.ini file under [TCEngine] or specify the number of rows when you submit the conversion. See *Running a Table Conversion* for more information.

  Another reason for forcing row-by-row processing is if you know that the values in the input table will produce duplicate keys in the output and you still want the non-duplicate keys to be inserted.

  The disadvantage of forcing row-by-row processing is that it slows down the conversion process.

- Buffer Inserts to Output Tables

  Choose this option to improve conversion performance if you have no event rules in place to process insertion errors and if you are processing row by row.

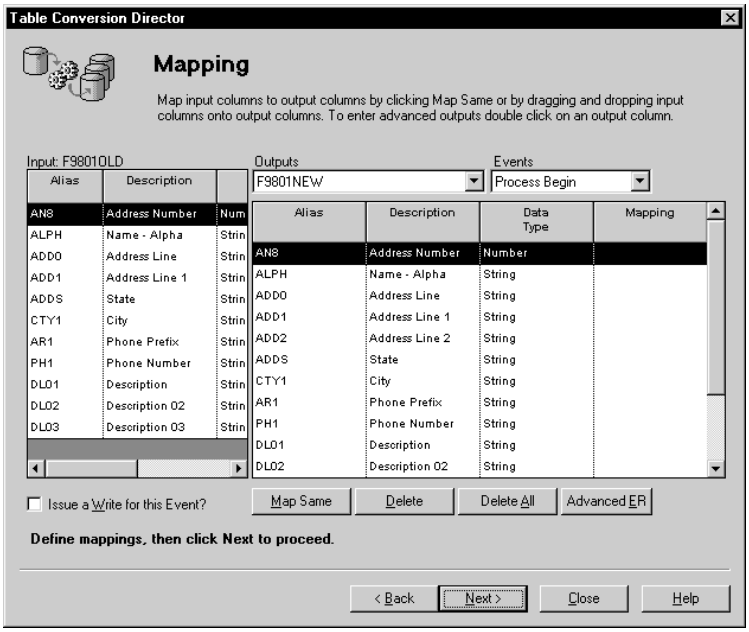
6. Click Next.

Field	Explanation
Currency Triggers	If you choose this option, currency triggers will run for all tables in the conversion. You should only choose this option if your input and output tables use currency fields. Conversions that do not need this option might run faster without it.
Clear Output	Clears the output table before the conversion runs.

Field	Explanation
Force Row by Row	If you choose this option, the conversion will run in a row by row mode, disallowing the conversion to run as an insert from select.

Mapping Input to Output

After you click Next from Table Options as described in the previous task, the Mapping form appears.



▶ To map inputs to outputs

1. Specify the event on which you want mapping to occur by choosing an event from the Events list. In most cases, you use either the Row Fetched Event or Format Fetched event. For example, if you are working with a user-defined format, choose the Format Fetched event.

   See *The Flow of Events in Table Conversion* for more information about Row Fetched and Format Fetched events.

2. Click the Advanced ER button to further modify your mappings, based on a particular event. See *Using Event Rules in Table Conversion* for more information about event rules.
3. Click Map Same to map your inputs directly to outputs. For example, if your input and outputs share some of the same data, these map directly.

For OneWorld tables, the system maps by data dictionary item. For non-OneWorld tables, the system maps by column name.

4. Drag inputs to outputs to define exactly where you want information to map.

   Note: Click the Delete to erase the mapping for a selected output. Click Delete All to erase the mapping for all outputs.

5. If you have multiple output files, choose each file from the outputs list and map the appropriate input columns to output columns.

6. To define advanced output, double-click on an output column.

   The Advanced Outputs form appears.

   The Advanced Outputs form allows you to define literals, calculations, and various other mappings without using Advanced ER. You might want to use an advanced input to add a constant, literal value into a field. Or, you might want to insert a calculation into an output field, such as adding two input fields together.

7. On Advanced Outputs, click one of the following tabs and add the appropriate input:

   - Available objects: Choose the output column, then choose the appropriate value, and then click the Apply button.
   - Literal: Choose the output column, enter the appropriate value, and click the Apply button.
   - Defaults: Choose the Use Dictionary Defaults option, and then click the Apply button.

     Use this option if you want to use the default value in the data dictionary at run time. If there are no default values in the data dictionary, the system displays a warning message.

   - Calculation: Click the Define Calculation button and define a calculation in Expression Manager.

8. When you have finished defining an advanced input, click Apply, then click Close.

9. On Mapping, choose the "Issue a Write for this Event" option to insert a row to the selected output after performing all column mappings for this event.

   When you choose the "Issue a Write for this Event" option, the system attaches the TC Insert Row event. This event is automatically inserted at the end of the event rules. You cannot move it to another area. If you want to specify when a row is inserted and where, attach the User Insert Row system function using Advanced ER and move it wherever you would like.

10. Click Next.

Choosing Logging Options

Use logging options to record specific events that will occur during the conversion.

After you click Next from Mapping as described in the previous task, the Logging Options form appears.

▶ To choose logging options

1. On Logging Options, choose one or more logging options, if applicable.
   - Log All Errors
   - Delete All Selected Records
   - Event List
   - Log Deletes
   - Log Updates
   - Trace Level
   - Log Details of Copy Table Actions
2. If you want to preview the actions of the table conversion before you run the actual conversion, choose the "Run in Proof Mode" option.

   Caution: Proof mode is not an absolute proof mode. In some situations, the proof output might differ from the real output. For example, if you insert the same record twice, it will seem as though it worked in proof mode, but in reality, only one of the inserts will work when you run the conversion in final mode.

3. Click Next.

Field	Explanation
Log All Errors	Indicates that every error should be logged, regardless of the logging option for type of operation that failed.
Delete All Selected Records	Indicates that all selected records should be deleted.
Log Deletes	Indicates that every record deleted should be logged.
Log Updates	Indicates that every record updated should be logged.

Field	Explanation
Trace Level	Indicates that the tables copied by a copy table environment or copy table data source system function call should be logged.
Log Details of Copy Table Actions	Indicates that detailed logging of each Copy Table Environment or Copy Table Data Source system function calls should be performed.
Proof Mode	Indicates that the conversion should be run in proof mode. Proof mode means that the conversion is simulated and all actions are logged, but that no data is changed.

Reviewing the Results of the Director

After you click Next from Logging Options as described in the previous task, the Finish form appears.

▶ To review the results of the director

1. Click Finish on the Finish form to complete the process.

   The system displays the Properties form and Table Conversion Mappings form.

2. Choose the Table Conversions Mappings form and review your choices.
3. Make changes as necessary by choosing the appropriate option from the View menu. If satisfied, click Save.
4. From the Conversion menu, choose Exit.

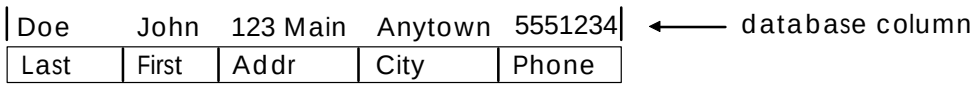
   See *Reviewing Your Table Conversion* for more information.

Understanding User-Defined Formats

User-defined formats are the table conversion tool's way of dealing with fixed-width or character separated value (CSV) files in a table or text file. These files are collectively known as *flat files* because they do not have relationships defined for them like relational database tables do. Flat files are usually text files stored on your workstation or server. They are used to import or export data from applications that have no other means of interaction. For example, you might want to share information between OneWorld and another application, but if the non-OneWorld application does not support one of the same databases that OneWorld supports, then flat files might be the only way to transfer data between the two applications.

In a flat file, records are stored as one continuous string of information. The user-defined format provides instruction on how data is presented.

The following example illustrates a single database character column that has a user-defined format with five columns: Last, First, Addr, City, and Phone.



This is a fixed-width column format, in which all of the data for each column starts in the same relative position in each row of data.

The same data in a character-delimited format would look like this:

"Doe", "John", "123 Main", "Anytown", 5551234

Importing and Exporting Text Files

When you choose a text file as input to or output from a table conversion and do not specify a path, a default path is used. Conversions stored with the default path run on any platform. If an explicit path or AS/400 library name is indicated for the file, then the file is located or created exactly as specified. Conversions stored this way may not work on other platforms, depending on the nature of the file system on each platform.

The default paths on non-AS400 platforms will be:

path code\Import\file name

path code\Export\file name

You cannot specify a default path for the AS/400. Rather, the default will always be the Import or the Export directory under the path code of the input or output environment. For example, if you are running a conversion against an APPL_PGF environment, the path in the file system might be:

\B733\appl_pgf\import\myfile.txt

If the conversion specifies a file name with anything other than the file name and extension, such as library/file(mbr) or \mytextfiles\myfile.txt, the conversion will try to open the file as specified.

Using User-Defined Formats as Input

Note: If you are using user-defined input formats and do not add an event rule at the Format Fetched event, the system ignores the format, and the data from the input table is never made available to the conversion. Therefore, you must add some sort of event rule logic to the Format Fetched event. At the very least, add a comment in Event Rules.

Using User-Defined Formats as Output

The requirements for using user-defined formats as output are basically the same as for using user-defined formats as input. User-defined formats work with text files and tables.

Because the procedure for importing and exporting is database-specific, you should consult your database administrator for details.

Defining User-Defined Formats

To define user-defined formats in table conversion, do one of the following:

- Define delimited, single- or multiple-format files
- Define fixed-width, single- or multiple-format files

A table conversion will not work with data that is formatted in a table.

▶ To define delimited, single- or multiple-format files

Use the Navigation Assistant to move to the Select Input form or the Select Outputs form. You can also use the Back or Next buttons to navigate to these forms in the director. Alternatively, you can click the Select Input tab or Select Outputs tab in Table Conversion Properties.

1. On Select Input or Select Outputs, ensure that you have chosen a table, business view, or file, and then click User Defined Format.

   The User Defined Format – Type form appears.

2. Choose the delimited format type.
3. Choose one of the following Row Formats and click Next:
   - Single Format on Rows On/Off
   - Multiple Formats on Rows On/Off

   The User Defined Format – Column Delimiter form appears.

4. Choose the delimiter that separates the columns in the file:
   - Tab Delimiter
   - Comma Delimiter
   - Semicolon Delimiter
   - Space Delimiter

- Other Delimiter

5. Choose the textual qualifier that is used to enclose a string of text:
   - No Text Qualifier
   - Single Quotation Qualifier
   - Double Quotation Qualifier
6. If you chose Single Format, specify whether the first row contains column headers.
7. Click Next.
8. If you chose Multiple Format, the system displays Multiple Format Definition. If you chose Single Format, skip to Step 13.
9. On User Defined Format - Multiple Format Definition, enter the number of formats that your user-defined format contains.
10. To define the character length of the Designator column, complete the following field:
    - Length
11. Click Next.

    The User Defined Format – Multiple Format Names form appears.

12. In the Designator column, define the values for each format. The Designator name must match what is in your user-defined formats.

    For example, suppose you have a text file that contains purchase order information. Lines in the table with a first field designated as "POH" have information for a whole purchase order. Lines with a first field designated as "POI" contain information about individual items in the purchase order, and lines in the table with a first field designated "POT" contain information about purchase order totals. In this scenario, you would enter "POH" as the designator of the first format, "POI" as the designator of the second format, and "POT" as the designator for the third format.

    Tip: You can also rename the columns for each format to make it easier to remember the formats with which you are working. For example, you can rename the columns according to their function in the file, such as Header, Detail, and Total. These names will appear in the Inputs drop down list in the Mapping section of the conversion. To rename columns, select the column and in the name field, change the name of the column.

13. Click Next.

    The User Defined Format – Column Layout form appears.

14. Choose a format from the list of Available Formats.

You define the columns for the format so that the system can parse the information from the file.

Tip: If you need to move the Format Designator, choose the row and drag it to the new location.

15. For each column, click Add to define the column.

    The New Column Properties form appears.

16. Modify the following fields as needed, and then click OK:
    - Name
    - Length
    - Data Type

17. For each column, repeat steps 13 through 16.
18. To edit an existing column, select it and click the Edit button. Change the properties in the Column Properties form.
19. To model the columns after an existing table, business view, or foreign table, click the Model button, click the appropriate tab, and then choose the table or business view that you want to use as a model for the user-defined format.

    Note: You cannot model the columns after an existing object unless the layout of the two objects match.

20. Click OK.

    The system copies the format from the model that you chose and places it into the column layout grid.

21. Click Next.

    The User Defined Format – Finish form appears.

22. Click Finish when you have completed defining formats.

    The system returns to the Select Input or Select Outputs form.

Field	Explanation
Delimited Format On/Off	A delimited format contains characters such as commas or tabs that separate each column.
Single Format on Rows On/Off	Switches between enable and disable for a single format for rows.

Field	Explanation
Multiple Formats on Rows On/Off	Switches between enable and disable of multiple formats for rows.
No Text Qualifier	Indicates that no character is used to qualify text in this user-defined file.
Single Quotation Qualifier	Indicates that a single quotation mark is used to qualify text in this user-defined file.
Double Quotation Qualifier	Indicates that a double quotation mark is used to quailfy text in this user-defined file.

▶ To define fixed-width, single- or multiple-format files

1. Click User Defined Format.

   The User Defined Format – Type form appears.

2. Choose the fixed width format.
3. Choose one of the following Row Formats and click Next:
   - Single Format on Rows On/Off
   - Multiple Formats on Rows On/Off
4. If you chose Single Format, skip to step 10. If you chose Multiple Formats, the system displays the Multiple Formats Definition form.
5. On User Defined Format - Multiple Formats Definition, enter the number of formats that your user-defined file contains.
6. Complete the following fields:
   - Start Position
   - Length

   The Start Position and Length fields define the position of the data for the rows. For instance, Start Position defines where the column starts. Length defines the character length of the designator.

7. Click Next.

   The User Defined Format – Multiple Format Names form appears.

8. In the Designator column, define the values for each format. The Designator name must match what is in your user-defined formats.

For example, suppose you have a text file that contains purchase order information. Lines in the table with a first field designated as "POH" have information for a whole purchase order. Lines with a first field designated as "POI" contain information about individual items in the purchase order, and lines in the table with a first field designated as "POT" contain information about purchase order totals. In this scenario, you would enter "POH" as the designator of the first format, "POI" as the designator of the second format, and "POT" as the designator for the third format.

Tip: You can also rename the columns for each format to make it easier to remember the formats with which you are working. For example, you can rename the columns according to their function in the file, such as Header, Detail, and Total. These names will appear in the Inputs drop down list in the Mapping section of the conversion. To rename columns, select the column and in the name field, change the name of the column.

9. Click Next.

   The User Defined Format – Column Layout form appears.

10. Choose a format from the list of Available Formats.

    You define the column so that the system can parse the information from the file.

    Tip: If you need to move the Format Locator, choose the row and drag it to the new location.

11. For each column, click Add to define the column layout.

    The New Column Properties form appears.

12. Modify the following fields as needed, and then click OK:

    - Name
    - Start
    - Length
    - Data Type

13. For each column, repeat steps 11 through 12.
14. To edit an existing column, select it and click the Edit button. Change the properties in the Column Properties form.
15. To model the columns after an existing table, business view, or foreign table, click the Model button, click the appropriate tab, and then choose the table or business view that you want to use as a model.
16. Click OK.

The system copies the format from the model that you chose and places it into the column layout grid.

17. Click Next.

    The system displays the summary of user-defined formats that you have defined.

18. Click Finish when you are finished defining formats.

    The system returns to the Select Input or Select Outputs form.

Field	Explanation
Fixed Width Format On/Off	Switches between enable and disable of a fixed width format.
Single Format on Rows On/Off	Switches between enable and disable for a single format for rows.
Multiple Formats on Rows On/Off	Switches between enable and disable of multiple formats for rows.

See Also

- *Defining Data Sequence* in the *Enterprise Report Writing Guide* for information about how to determine a sort sequence for a batch process
- *Reviewing Your Table Conversion*
- *Using Event Rules in a Table Conversion*
- *Running a Table Conversion*

Copying Data

You use the Data Copy option in the Table Conversion Director to copy one or more tables from one environment or data source to another. You can also import a copy table script to use in the conversion.

The Director leads you through a linear process for creating a data conversion batch application by asking you questions about its structure and function. When you are finished, you can review and alter the conversion, if necessary. The steps in performing a Data Copy operation are:

- Defining external data
- Defining input and output environments
- Defining data copy actions
- Choosing logging options
- Reviewing the results of the director

Defining External Data

In some cases, you may need to apply a processing option template or a data structure to external data before it is converted.

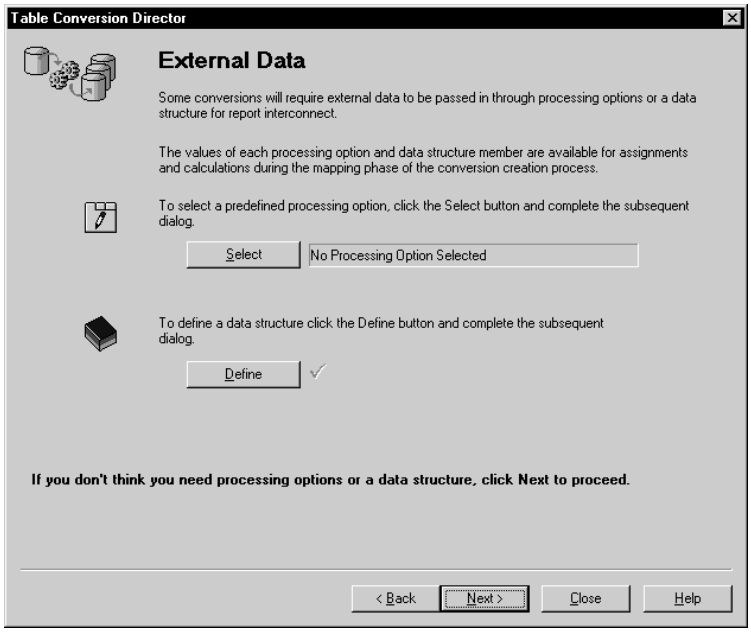
Before You Begin

❑ Create a batch application object. See *Using the Table Conversion Director* for information about starting the data conversion design process and the Director. The last step launches the Director.

▶ To define external data

1. On the Introduction form of the Table Conversion Director, choose the Data Copy option and click Next.

   The External Data form appears.



2. If you want to attach a predefined processing option template to the table conversion, click Select.

   The Select Processing Option Template form appears.

3. On Select Processing Option Template, find and choose the processing option that you want to use and click OK.
4. If you want to attach data structures, click Define on the External Data form.

   The Report Data Structures form appears.

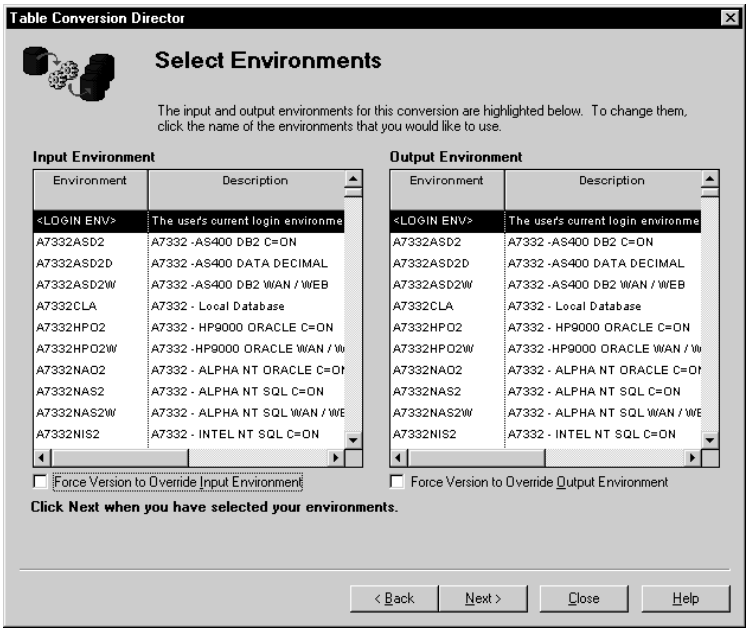
   Data structures contain a list of parameters that can be used to pass data into the conversion when called through Report Interconnect. See *Data Structures* in the *Development Tools Guide* for more information.

5. Define the data structures that you want to attach to the table conversion and click OK.
6. Click Next.

Defining Input and Output Environments

You may define a different environment for the input and the output.

After you click Next from External Data as described in the previous task, the Select Environments form appears.



To define input and output environments

1. On Select Environments, choose the input and output environments you want to use.

   Note: Choose <LOGIN ENV> if, for example, you are creating a table conversion on your workstation that will be shipped to a client who does not have the environments that you have, and the environment they log into will always be appropriate.

2. Choose the "Force Version to Override Input Environment" or "Force Version to Override Output Environment" if you are creating a table conversion that will run in a different environment from the one in which you are creating it, and the <LOGIN ENV> is not appropriate for the type of conversion that you are creating.

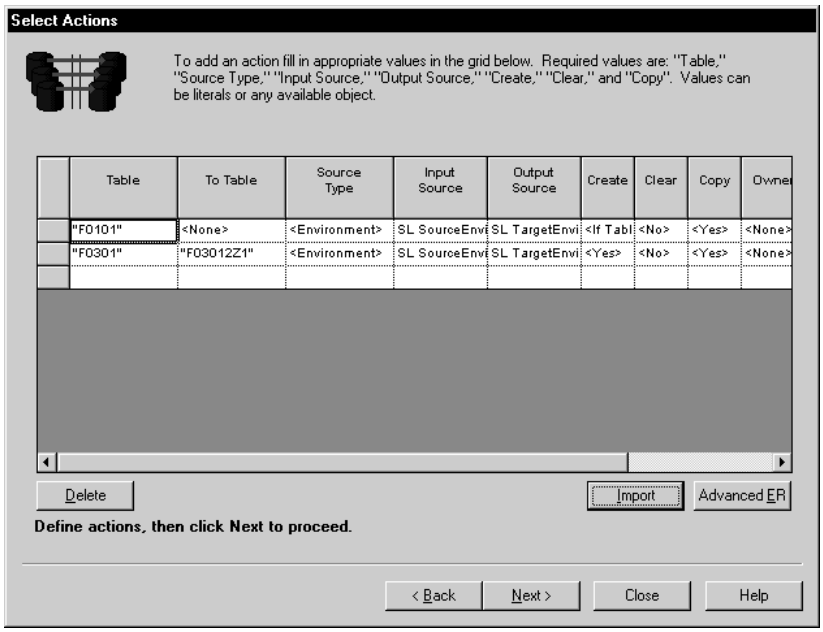
   For example, if you are creating a conversion that will be shipped to a client who does not have the environments that you have, you would choose the "Force Version to Override" option. When the conversion is invoked at the client site, the system will not run the conversion until the user chooses an appropriate environment in which to run it.

3. Click Next.

Field	Explanation
Force Version to Override Input	If you choose this option, the conversion will not run unless you creates a version that overrides the input environment.
Force Version to Override Output	If this option is chosen, the conversion will not run unless the user creates a version that will override the output environment.

Defining Data Copy Actions

After you click Next from Select Environments as described in the previous task, the Select Actions form appears.



▶ To define data copy actions

1. On Select Actions, complete the following fields. Use the drop-down lists in each field to make your choice.

   When you enter the name of a table and tab to the next field, the system automatically populates the remaining fields for you. You can make changes to these fields as necessary.

   - Table

     If you want to copy a single table, choose <Literal> and enter the name of that table on the Single Value Tab.

If you do not know the name of the table that you want to copy, use the <Find a Table> option.

- To Table

  Enter either the last table in a range of tables to be copied, or leave this field blank if you are copying a single table.

- Source Type

  Choose Data Source if your input and output sources are data sources. Choose Environment if your input and output sources are environments. When you choose Data Source or Environment, the appropriate system function (such as CopyTableEnvironment or CopyTableDataSource) is invoked during processing.

  For more information about system functions, see *Using Event Rules in a Table Conversion.*

  The Data Source function works the same way as Copy Table and gets its table descriptions from the specifications in the login environment.

  The Environments function uses the input and output environment to locate data and specifications for the tables. This function allows the specifications to be different in the input and output environment, but the data still gets copied. In this case, the system performs a "copy-map-drop."

- Input Source

  The input source is the data source or environment from which the inputs will be read.

- Output Source

  The output data source is the source or environment where the output is written.

- Create

  If you choose <If Table Exists>, the system creates the table and runs the conversion only if both the table specification and the actual table exist in the input.

  If you choose <Yes>, the system creates the table. If the table already exists in the output, the system will delete and re-create it.

  If you choose <No>, the system assumes the table already exists in the output and will not re-create it.

- Clear

  If you choose <If Table Exists>, the system clears the table only if it exists in the input.

  If you choose <Yes>, the system deletes all rows in the output table before copying the table.

  If you choose <No>, the output table will not be cleared. Note that this can result in key conflicts.

- Copy

  If you choose <Yes>, the system copies the data from the input table to the output table using Map Same.

  If you choose <No>, no data is copied.

- Owner ID
- Owner Pwd

  If the data source requires an owner ID and password, enter them here. If you leave these fields blank, the system enters the ID and password of the login user, or <None> if the data source does not have security.

2. To import an existing copy table script from another location, click the Import button.
3. On Open, find the file that you want to import and click Open.

   The system adds an action for each copy table item in the copy table script.

4. On Select Actions, click Advanced ER to add event rules to the copy table process.

   You can use Event Rules to write a custom copy table script. See *Using Event Rules in a Table Conversion* for more information.

5. Click Next.

Field	Explanation
Source Type	Indicates whether the input and output are specified using a data Source name or an Environment name.
Input Source	The source from which to copy the table(s).
Output Source	The location into which to copy the table(s).

Field	Explanation
Create	Indicates whether the table(s) should be created in the output.
Clear	Indicates whether the table(s) should be cleared in the output.
Copy	Indicates whether records from the input table(s) should be copied to the output.
Owner ID	If creating the table, the owner ID to use when creating it.
Owner Pwd	If creating the table, the password to use when creating it.

Choosing Logging Options

Use logging options to record specific events that will occur during the conversion.

After you click Next from Select Actions as described in the previous task, the Logging Options form appears.

▶ To choose logging options

1. On Logging Options, choose one or more logging options, if applicable.
   - Log All Errors
   - Trace Level
   - Log Details of Copy Table Actions
2. If you want to preview the actions of the table conversion before you run the actual conversion, choose the *Run in Proof Mode* option.
3. Click Next.

Field	Explanation
Log All Errors	Indicates that every error should be logged, regardless of the logging option for type of operation that failed.
Log Details of Copy Table Actions	Indicates that detailed logging of each Copy Table Environment or Copy Table Data Source system function calls should be performed.
Proof Mode	Indicates that the conversion should be run in proof mode. Proof mode means that the conversion is simulated and all actions are logged, but that no data is changed.

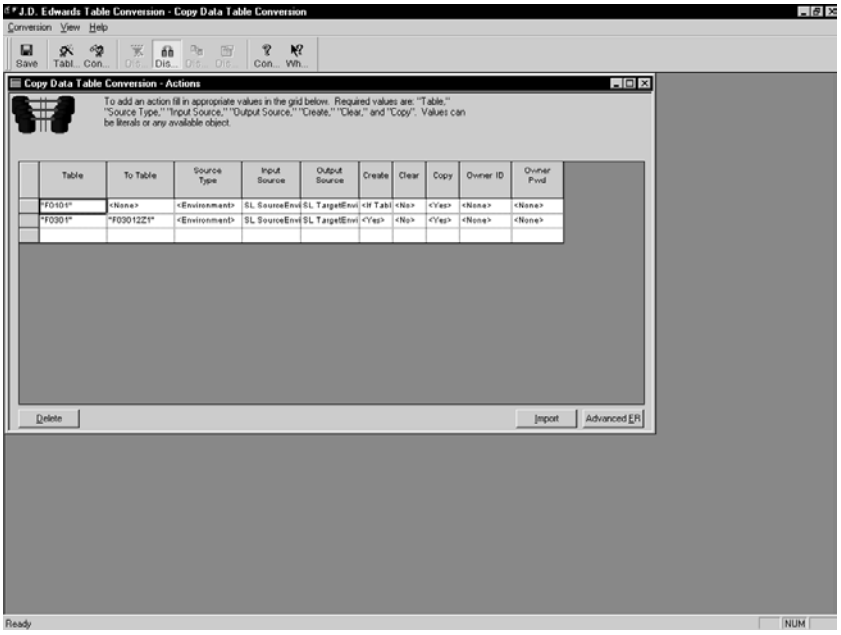
Reviewing the Results of the Director

After you click Next from Logging Options as described in the previous task, the Finish form appears.

▶ To review the results of the director

1. Click Finish on the finish form to complete the process.

   The system displays Table Conversion Actions.



2. Review your choices and, if satisfied, click Save.
3. From the File menu, choose Exit.

   See *Reviewing Your Table Conversion* for more information.

You can now run the table conversion.

See Also

- *Using Event Rules in a Table Conversion*
- *Reviewing Your Table Conversion*
- *Running a Table Conversion*

Copying Data with Table Input

Data Copy with Table Input is similar to Data Copy except that it also allows information for the process to come from an input table. The input table might provide information about which tables should be copied, where they should be copied, and so on. Data Copy with Table Input also allows you to do data selection.

For example, suppose you create a table that has a table name, the next backup date, and the backup frequency. You might populate the table with a list of tables to be archived and how often they should be archived. You can then use Data Copy with Table Input to select all rows in which the backup date is less than or equal to today's date, and then calculate a new backup date.

The Director leads you through a linear process for creating a data conversion batch application by asking you questions about its structure and function. When you are finished, you can review and alter the conversion if necessary. The steps in copying data with table input are:

- Defining external data
- Defining input and output environments
- Defining input
- Defining data copy actions
- Choosing logging options
- Reviewing the results of the director

Defining External Data

In some cases, you may need to apply a processing option template or a data structure to external data before it is converted.

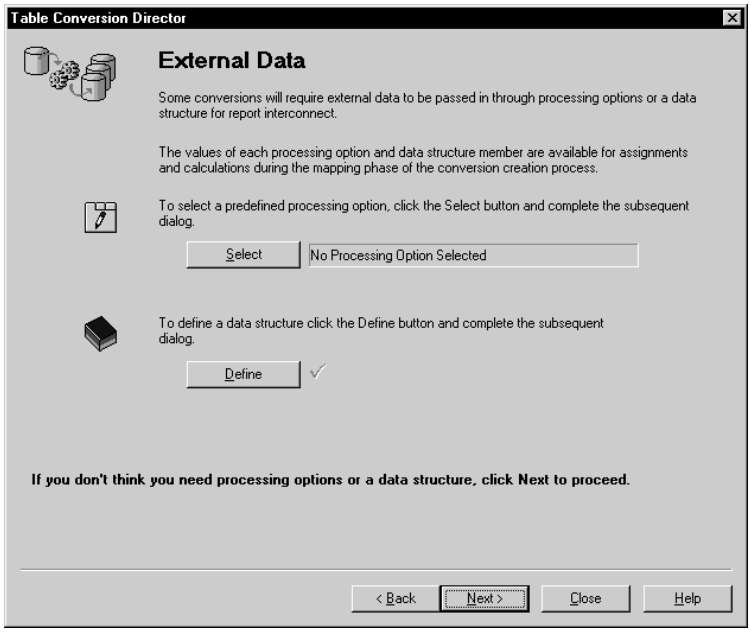
Before You Begin

❑ Create a batch application object. See *Using the Table Conversion Director* for information about starting the data conversion design process and the Director. The last step launches the Director.

To define external data

1. On the Introduction form of the Table Conversion Director, choose the Data Copy with Table Input option and click Next.

   The External Data form appears.



2. If you want to attach a predefined processing option template to the table conversion, click Select.

   The Select Processing Option Template form appears.

3. On Select Processing Option Template, find and choose the processing option you want to use and click OK.
4. If you want to attach data structures, click Define on the External Data form.

   The Report Data Structures form appears.

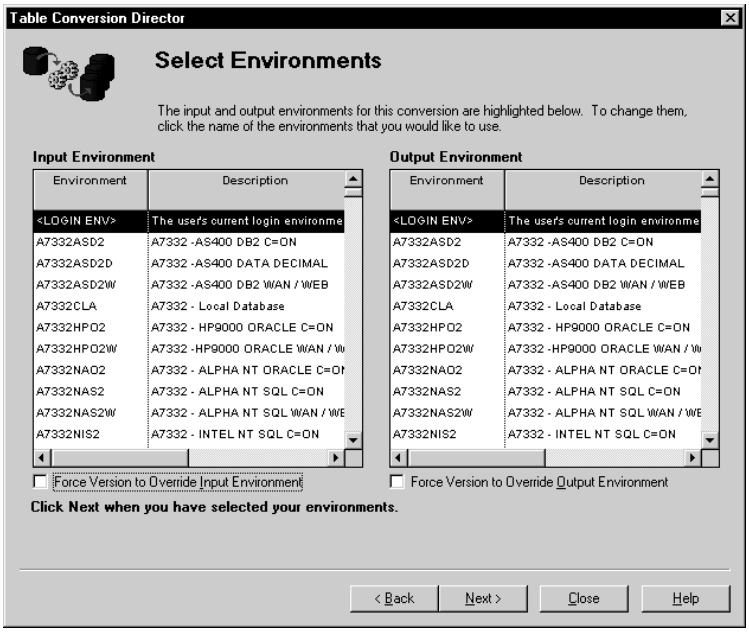
   Data structures contain a list of parameters that can be used to pass data into the conversion when called through Report Interconnect. See *Data Structures* in the *Development Tools Guide* for more information.

5. Define the data structures that you want to attach to the table conversion and click OK.
   - Output Source
6. Click Next.

Defining Input and Output Environments

You may define a different environment for the input and the output.

After you click Next from External Data as described in the previous task, the Select Environments form appears.



▶ To define input and output environments

1. On Select Environments, choose the input and output environments you want to use.

   Note: Choose <LOGIN ENV> if, for example, you are creating a table conversion on your workstation that will be shipped to a client who does not have the environments that you have, and the environment they log into will always be appropriate.

2. Choose the "Force Version to Override Input Environment" or "Force Version to Override Output Environment" if you are creating a table conversion that will run in a different environment than the one in which you are creating it, and the <LOGIN ENV> is not appropriate for the type of conversion that you are creating.

   For example, if you are creating a conversion that will be shipped to a client who does not have the environments that you have, you would choose the "Force Version to Override" option. When the conversion is invoked at the client site, the system will not run the conversion until the user chooses an appropriate environment in which to run it.

3. Click Next.

Field	Explanation
Force Version to Override Input	If you choose this option, the conversion will not run unless you creates a version that overrides the input environment.
Force Version to Override Output	If this option is chosen, the conversion will not run unless the user creates a version that will override the output environment.

Defining Input

Conversion input may originate from a table, a business view, or a flat file. You may select only one input object.

The input table in this conversion scenario acts as a control table. Event rules that you create governing whether data is copied is based on the input table you choose in this part of the process.

After you click Next from Select Environments as described in the previous task, the Select Input form appears.

▶ To define input

1. On Select Input, drag the table or business view to the column on the right. You can choose only one table or one business view per conversion. If your input consists of multiple tables, you must create a single joined business view.

   Tip: If you know the name of the table or business view you want to use, enter the name in the Name field in the QBE (query by example) row and press Enter. Alternately, for text files, you can select a file from the default directory, enter a new file name, or click the Browse button to locate a file.

   Note: If you change the table, business view, or file, the system warns you that deleting tables removes all mappings from the table conversion.

2. If you are working with a user-defined format (flat file), click User-Defined Format and follow the steps described in *Defining User-Defined Formats.* When you complete those steps, return here.
3. To delete an input name, choose it and press Delete.
4. Click Next.

The Sequencing form appears.

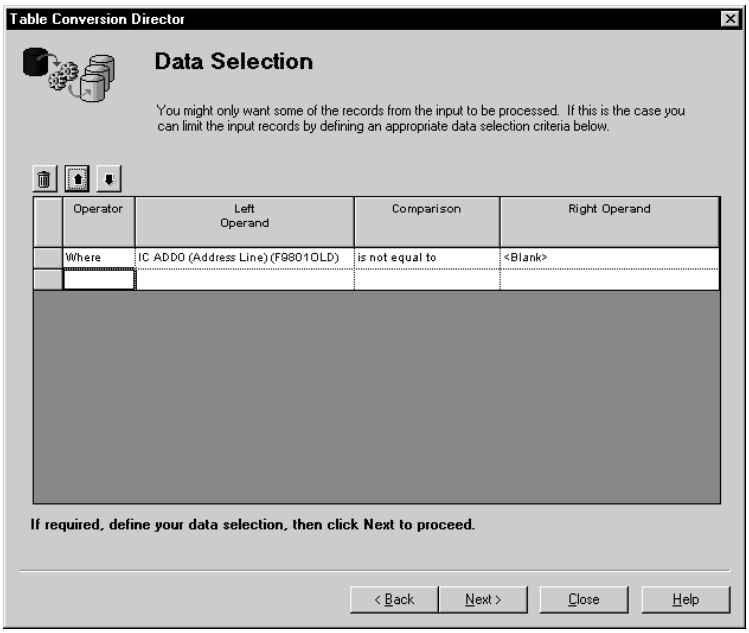
5. To define data sequencing for a table or business view, click Data Sequencing. If you specify a text file for input, you cannot define data sequencing or selection for that file.

   When you define data sequencing, you create new events that are available to you in the Mapping section of the Director. One new event is created for each of the sequence columns that you define. The event is called *XXXX* Data Changed, where *XXXX* is the column alias – for example, ALPH Data Changed. Each time that the value in one of these columns changes from its previous value, the column's Data Changed event is invoked. This event is similar to a level break in report writing with the exception that the Data Changed events are not related to each other. Invoking one does not invoke the others.

   See *The Flow of Events in Table Conversion* for more information about the Data Changed event.

6. Click Next.

   The Data Selection form appears.



7. On Data Selection, you can only define selection criteria over database table columns. User-defined format columns are unavailable because they do not exist in the database.

   Where is the default value in the Operator column for the first set of criteria. For subsequent statements, *And* and *Or* become the available

values for the Operator column and are selected by double-clicking the appropriate one.

8. Click in the Left Operand column to display the list of available objects, and then perform one of the following:
   - Scroll through the list until you find the desired object, choose the object, and then double-click the object to populate the Left Operand column.
   - Type in the first letters of the object name in the Left Operand field to bring you to the object in the list, and then double-click the highlighted object.

   When you double-click the object for the Left Operand column, the list in the Comparison column automatically appears.

9. Select one of the following comparison operators:
   - is equal to
   - is greater than
   - is greater than or equal to
   - is less than
   - is less than or equal to
   - is not equal to

10. Click in the Right Operand column to display an available list of objects, special values, or variables. Your choices in this column depend on the choice that you made in the Comparison column. Some of the following options could be available:

Blank	Enters a blank (space) value
Literal	Enter specific values (see the following step for information on entering specific values)
Null	Indicates that no value is associated with the field
Zero	Enters a value of zero
IC	Indicates an input table column
RI	Indicates a value passed through report interconnections to this table conversion
PO	Indicates a processing options value for this report

SL	Indicates a system literal

11. If you chose to enter a literal in the Right Operand column, the form that opens automatically enables you to enter the following:

    - Single value

      Enter a single value, and then click OK. For example, a value might be for a particular company.

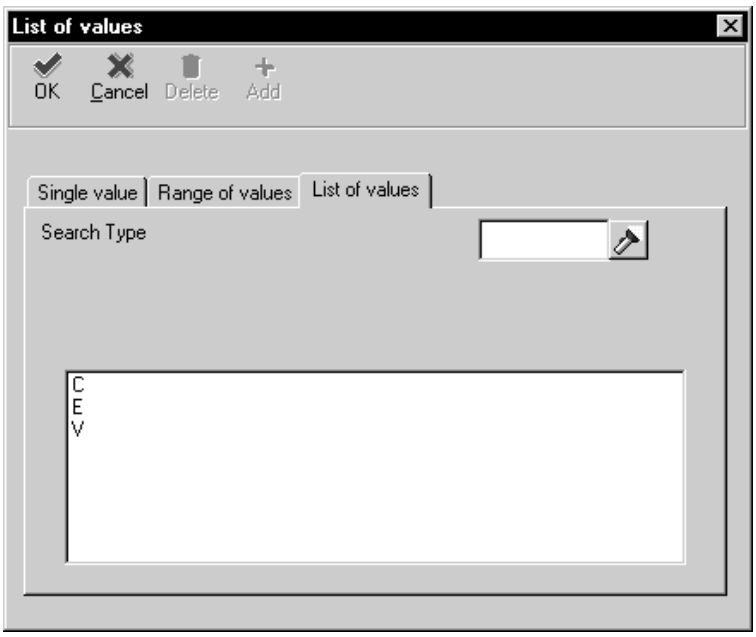
    - Range of values

      Enter a range of values, and then click OK. For example, a range of values might include companies from 00001 to 00060. Only *is equal to* and *is not equal to* are valid logical operators when using range of values.

    - List of values

      To add values to or remove values from the list, do the following:

      - Type each value in the field, and then press Enter or click the Add button at the top of the form.
      - Repeat this process until your list of values is complete. For example, a list of values might include several user defined codes for search types such as C for Customers, E for Employees, and V for Vendors. Only *is equal to* and *is not equal to* are valid logical operators when using list of values.
      - Delete a value by choosing the value, then click the Delete button at the top of the form.
      - Click OK when you are finished.



12. To delete a line of criteria on Data Selection, choose the row header to highlight the row, and then click the Delete button at the top of the form.
13. To change the order of the criteria, choose the row header to highlight the row, then click the up or down button.
14. Click Next.

    The Table Options form appears.

15. Choose the Run Currency Triggers option, if applicable.

    Choose this option if the OneWorld table or tables contain currency triggers. If the tables contain currency fields and you do not choose this option, the system does not know how many decimal places exist in each column. Anytime the source or destination fields are currency fields, and you do not turn on the currency trigger, problems can arise if the value is used in a calculation. The system has no way to determine where the decimal should be within a field.

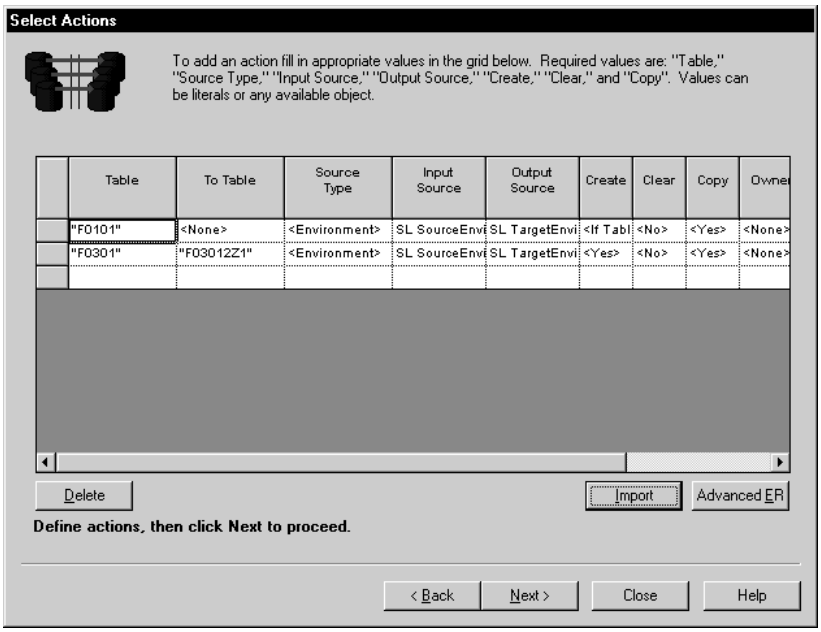
    You might not want to choose the currency trigger option if the input and output data sources are the same type (for example, Oracle, AS/400, or SQL Server), and no calculations are being performed. Choosing this option results in slower processing time.

    In addition, you should not use currency triggers on an environment that has a different path code than the login environment.

16. Click Next.

Defining Data Copy Actions

After you click Next from Select Inputs as described in the previous task, the Select Actions form appears.



▶ To define data copy actions

1. On Select Actions, complete the following fields. Use the drop-down lists in each field to make your choice.

   When you enter the name of a table and tab to the next field, the system automatically populates the remaining fields for you. You can make changes to these fields as necessary.

   - Table

     If you want to copy a single table, choose <Literal> and enter the name of that table on the Single Value Tab.

     If you do not know the name of the table that you want to copy, use the <Find a Table> option.

   - To Table

     Enter either the last table in a range of tables to be copied, or leave this field blank if you are copying a single table.

   - Source Type

Choose Data Source if your input and output sources are data sources. Choose Environment if your input and output sources are environments. When you choose Data Source or Environment, the appropriate system function (such as CopyTableEnvironment or CopyTableDataSource) is invoked during processing.

For more information about system functions, see *Using Event Rules in Table a Conversion.*

The Data Source function works in the same way as Copy Table and gets its table descriptions from the specifications in the login environment.

The Environments function uses the input and output environment to locate data and specifications for the tables. This allows the specifications to be different in the input and output environment, but the data still gets copied. In this case, the system performs a "copy-map-drop."

- Input Source

  The input source is the data source or environment from which the inputs will be read.

- Output Source

  The output data source is the source or environment where the output is written.

- Create

  If you choose <If Table Exists>, the system creates the table and runs the conversion only if both the table specification and the actual table exist in the input.

  If you choose <Yes>, the system creates the table. If the table already exists in the output, the system deletes and re-creates it.

  If you choose <No>, the system assumes the table already exists in the output and does not re-create it.

- Clear

  If you choose <If Table Exists>, the system clears the table only if it exists in the input.

  If you choose <Yes>, the system deletes all rows in the output table before copying the table.

  If you choose <No>, the output table will not be cleared.

Note: Choosing not to clear the output table might result in key conflicts.

- Copy

  If you choose <Yes>, the system copies the data from the input table to the output table using Map Same.

  If you choose <No>, no data is copied.

- Owner ID
- Owner Pwd

  If the data source requires an owner ID and password, enter them here. If you leave these fields blank, the system enters the ID and password of the login user, or <None> if the data source does not have security.

2. To import an existing copy table script from another location, click the Import button.
3. On Open, find the file that you want to import and click Open.

   The system adds an action for each copy table item in the copy table script.

4. On Select Actions, click Advanced ER to add event rules to the copy table process.

   You can use Event Rules to write a custom copy table script. See *Using Event Rules in a Table Conversion* for more information.

5. Click Next.

Field	Explanation
Source Type	Indicates whether the input and output are specified using a data Source name or an Environment name.
Input Source	The source from which to copy the table(s).
Output Source	The location into which to copy the table(s).
Create	Indicates whether the table(s) should be created in the output.
Clear	Indicates whether the table(s) should be cleared in the output.
Copy	Indicates whether records from the input table(s) should be copied to the output.

Field	Explanation
Owner ID	If creating the table, the owner ID to use when creating it.
Owner Pwd	If creating the table, the password to use when creating it.

Choosing Logging Options

Use logging options to record specific events that will occur during the conversion.

After you click Next from Select Actions as described in the previous task, the Logging Options form appears.

▶ To choose logging options

1. On Logging Options, choose one or more logging options, if applicable.
   - Log All Errors
   - Delete All Selected Records
   - Log Deletes
   - Log Updates
   - Trace Level
   - Log Details of Copy Table Actions
2. If you want to preview the actions of the table conversion before you run the actual conversion, choose the "Run in Proof Mode" option.
3. Click Next.

Field	Explanation
Log All Errors	Indicates that every error should be logged, regardless of the logging option for type of operation that failed.
Delete All Selected Records	Indicates that all selected records should be deleted.
Log Deletes	Indicates that every record deleted should be logged.
Log Updates	Indicates that every record updated should be logged.
Trace Level	Indicates that the tables copied by a copy table environment or copy table data source system function call should be logged.
Log Details of Copy Table Actions	Indicates that detailed logging of each Copy Table Environment or Copy Table Data Source system function calls should be performed.

Field	Explanation
Proof Mode	Indicates that the conversion should be run in proof mode. Proof mode means that the conversion is simulated and all actions are logged, but that no data is changed.

Reviewing the Results of the Director

After you click Next from Logging Options as described in the previous task, the Finish form appears.

▶ To review the results of the director

1. Click Finish on the Finish form to complete the process.

   The system displays the Table Conversion Actions form.

2. Review your choices and, if satisfied, click Save.

   See *Reviewing Your Table Conversion* for more information.

3. From the File menu, choose Exit.

You can now run the table conversion.

See Also

- *Using Event Rules in a Table Conversion*
- *Reviewing your Table Conversion*
- *Running a Table Conversion*

Deleting Records

The Batch Delete option allows you to delete a range of records from a OneWorld input table or foreign table, based on selection criteria that you define. For example, you can set up a batch delete table conversion that deletes any records in an input table that do not contain invalid data or records. You might also want to set up a conversion that deletes all records from a particular table.

The Director leads you through a linear process for creating a data conversion batch application by asking questions about its structure and function. When you are finished, you can review and alter the conversion, if necessary. Deleting records involves the following steps:

- Defining external data
- Defining the environment
- Defining input
- Choosing logging options
- Reviewing the results of the director

Defining External Data

In some cases, you may need to apply a processing option template or a data structure to external data before it is converted.

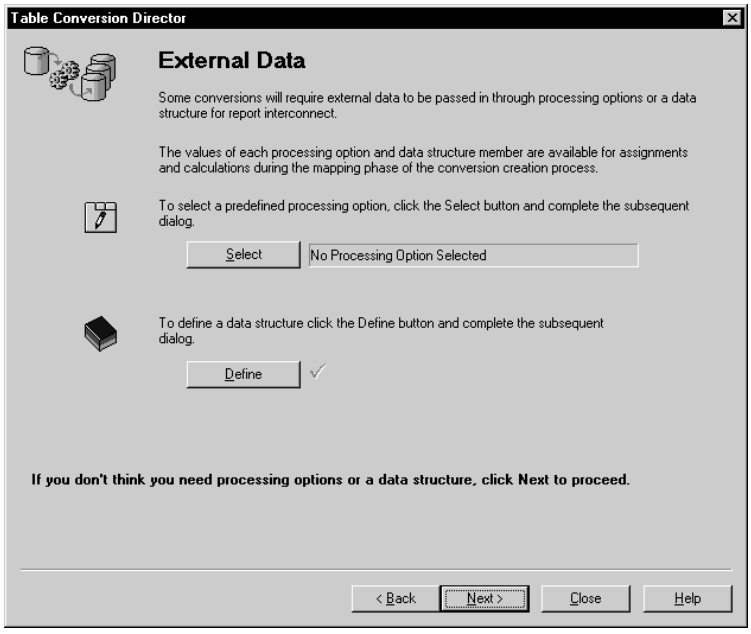
Before You Begin

❑ Create a batch application object. See *Using the Table Conversion Director* for information about starting the data conversion design process and the Director. The last step launches the Director.

▶ To define external data

1. On the Introduction form of the Table Conversion Director, choose the Batch Delete option and click Next.

   The External Data form appears.



2. If you want to attach a predefined processing option template to the table conversion, click Select.

   The Select Processing Option Template form appears.

3. On Select Processing Option Template, find and choose the processing option that you want to use and click OK.
4. If you want to attach data structures, click Define on the External Data form.

   The Report Data Structures form appears.

   Data structures contain a list of parameters that can be used to pass data into the conversion when called through Report Interconnect. See *Data Structures* in the *Development Tools Guide* for more information.

5. Define the data structures that you want to attach to the table conversion and click OK.
   - Output Source
6. Click Next.

Defining the Environment

After you click Next from External Data as described in the previous task, the Select Environment form appears.

▶ To define the environment

1. On Select Environment, choose the environment in which the table resides.

   Note: Choose <LOGIN ENV> if, for example, you are creating a table conversion on your workstation that will be shipped to a client who does not have the environments that you have. The environment they log into will always be appropriate.

2. Choose the "Force Version to Override Input Environment" or "Force Version to Override Output Environment" if you are creating a table conversion that will run in a different environment than the one in which you are creating it, and the <LOGIN ENV> is not appropriate for the type of conversion that you are creating.

   For example, if you are creating a conversion that will be shipped to a client who does not have the environments that you have, you would choose the "Force Version to Override" option. When the conversion is invoked at the client site, the system will not run the conversion until the user chooses an appropriate environment in which to run it.

3. Click Next.

Defining Input

Conversion input may originate from a table. You may select only one input object.

After you click Next from Select Environment as described in the previous task, the Select Input form appears.

▶ To define input

1. On Select Input, drag the table to the column on the right. You can choose only one table per conversion.

   Tip: If you know the name of the table that you want to use, enter the name in the Name field in the QBE (query by example) row and press Enter. Or for text files, you can select a file from the default directory, enter a new file name, or click the Browse button to locate a file.

   Note: If you change the table, the system warns you that deleting tables removes all mappings from the table conversion.

2. To delete an input name, choose it and press Delete.
3. Click Next.

The Table Options form appears.

4. Choose the Run Currency Triggers option, if applicable.

   Choose this option if the OneWorld table or tables contain currency triggers. If the tables contain currency fields and you do not choose this option, the system will not know how many decimal places exist in each column. Anytime that the source or destination fields are currency fields and you do not turn on the currency trigger, problems could arise if the value is used in a calculation. The system has no way to determine where the decimal should be within a field.

   You might not want to choose the currency trigger option if the input and output data sources are the same type (for example, Oracle, AS/400, or SQL Server), and no calculations are being performed. Furthermore, choosing the currency trigger option results in slower processing time.

   In addition, you should not use currency triggers on an environment that has a different path code than the login environment.

   The Data Selection form appears.

5. On Data Selection, you can only define selection criteria over database table columns. User-defined format columns are unavailable because they do not exist in the database.

   Where is the default value in the Operator column for the first set of criteria. For subsequent statements, *And* and *Or* become the available values for the Operator column and are selected by double-clicking the appropriate one.

6. Click in the Left Operand column to display the list of available objects, and then do one of the following:

   - Scroll through the list until you find the desired object, choose the object, and then double-click the object to populate the Left Operand column.
   - Type in the first letters of the object name in the Left Operand field to bring you to the object in the list, and then double-click the highlighted object.

   When you double-click the object for the Left Operand column, the list in the Comparison column automatically appears.

7. Select one of the following comparison operators:

- is equal to
- is greater than
- is greater than or equal to
- is less than
- is less than or equal to
- is not equal to

8. Click in the Right operand column to display an available list of objects, special values, or variables. Your choices in this column depend on the choice you made in the Comparison column. Some of the following options could be available:

Blank	Enters a blank (space) value
Literal	Enter specific values (see the following step for information on entering specific values)
Null	Indicates that no value is associated with the field
Zero	Enters a value of zero
IC	Indicates an input table column
RI	Indicates a value passed through report interconnections to this table conversion
PO	Indicates a processing options value for this report
SL	Indicates a system literal
VA	Indicates an event rule variable

9. If you chose to enter a literal in the Right operand column, the form that opens automatically enables you to enter the following:

   - Single value

     Enter a single value, and then click OK. For example, a value might be for a particular company.

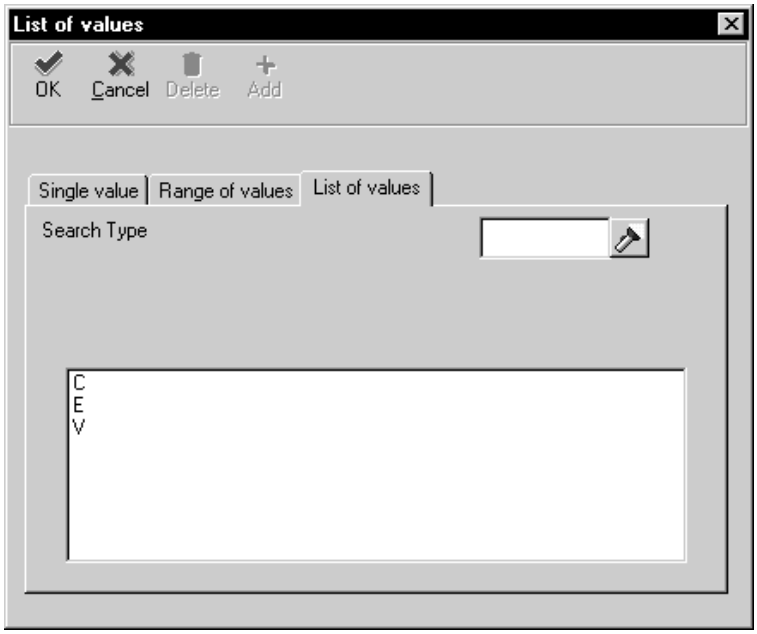
- Range of values

  Enter a range of values, and then click OK. For example, a range of values might include companies from 00001 to 00060. Only *is equal to* and *is not equal to* are valid logical operators when using range of values.

- List of values

  To add values to or remove values from the list, do the following:

  - Type each value in the field, and then press Enter or click the Add button at the top of the form.
  - Repeat this process until your list of values is complete. For example, a list of values might include several user defined codes for search types such as C for Customers, E for Employees, and V for Vendors. Only *is equal to* and *is not equal to* are valid logical operators when using list of values.
  - Delete a value by choosing the value, and then click the Delete button at the top of the form.
  - Click OK when you are finished.



10. To delete a line of criteria on Data Selection, choose the row header to highlight the row, and then click the Delete button at the top of the form.
11. To change the order of the criteria, choose the row header to highlight the row, and then click the up or down button.

12. In the Events field, choose the appropriate event from the drop-down list. You must choose Row Fetched as the event where the delete occurs; otherwise, no records will be deleted.

    When you run the conversion, the system will fetch the rows one at a time, run the conversion for each row, and delete the record from the input.

13. Make sure the "Delete all selected records" option is checked. This option inserts the Delete Current Input Row event into Event Rules.
14. Click the Advanced ER button if you want to add event rules to define more complicated actions than simply deleting all selected records.

    See *Using Event Rules in a Table Conversion* for more information about table conversion event rules.

15. Click Next.

Field	Explanation
Force Version to Override Input	If you choose this option, the conversion will not run unless you creates a version that overrides the input environment.
Force Version to Override Output	If this option is chosen, the conversion will not run unless the user creates a version that will override the output environment.

Choosing Logging Options

Use logging options to record specific events that will occur during the conversion.

After you click Next from Data Selection as described in the previous task, the Logging Options form appears.

▶ To choose logging options

1. On Logging Options, choose one or more logging options, if applicable.
   - Log All Errors
   - Delete All Selected Records
   - Log Deletes
   - Log Updates
   - Trace Level

- Log Details of Copy Table Actions

2. If you want to preview the actions of the table conversion before you run the actual conversion, choose the "Run in Proof Mode" option.
3. Click Next.

Field	Explanation
Log All Errors	Indicates that every error should be logged, regardless of the logging option for type of operation that failed.
Delete All Selected Records	Indicates that all selected records should be deleted.
Log Deletes	Indicates that every record deleted should be logged.
Log Updates	Indicates that every record updated should be logged.
Trace Level	Indicates that the tables copied by a copy table environment or copy table data source system function call should be logged.
Log Details of Copy Table Actions	Indicates that detailed logging of each Copy Table Environment or Copy Table Data Source system function calls should be performed.
Proof Mode	Indicates that the conversion should be run in proof mode. Proof mode means that the conversion is simulated and all actions are logged, but that no data is changed.

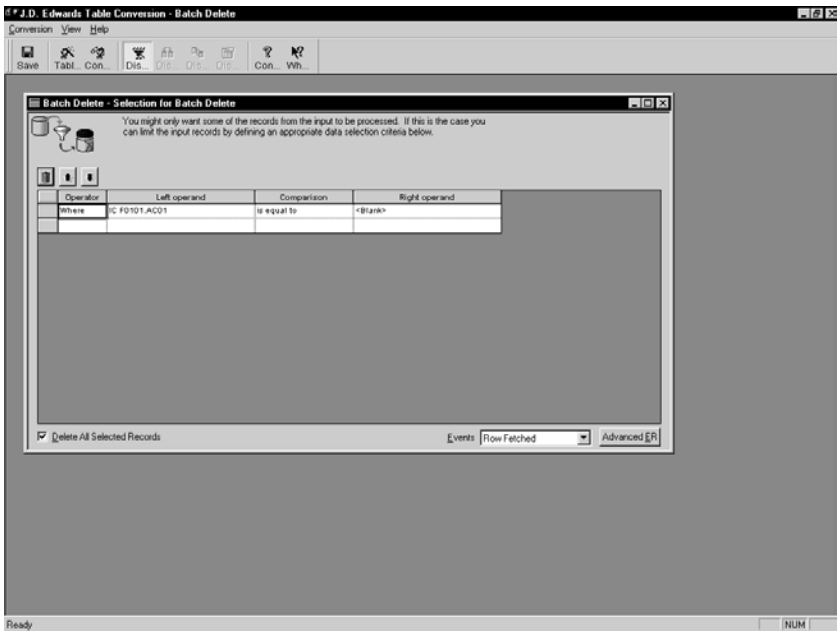
Reviewing the Results of the Director

After you click Next from Logging Options as described in the previous task, the Finish form appears.

▶ To review the results of the director

1. Click Finish on the Finish form to complete the process.

   The system displays the Selection for Batch Delete form.



2. On Table Conversion Actions, review the options that you specified for the batch delete conversion. If you are satisfied with your choices, click Save. Otherwise, make changes as necessary and then click Save.

   See *Reviewing Your Table Conversion* for more information.

3. From the File menu, choose Exit.

You can now run the conversion.

See Also

- *Using Event Rules in a Table Conversion*
- *Reviewing Your Table Conversion*
- *Running a Table Conversion*

Reviewing Your Table Conversion

You can review your table conversion after you create it using the Table Conversion Properties form. This form has multiple tabs and presents the same options as the Director. You can also review processing options, data structures, data sequencing, and data selection by choosing them from the Conversion menu. Each of the tabs on the form works identically to the forms in the Table Conversion Director.

▶ To review your table conversion

1. On Table Conversion, from the View menu, choose Table Conversion Properties.

   The Table Conversion Properties form appears.

2. On Table Conversion Properties, you can review and modify your table conversion.
3. When you are satisfied with your table conversion, click OK.
4. Click the Save button.
5. From the File menu, choose Exit.

Using Event Rules in a Table Conversion

You can use event rules to build complex functional capabilities into table conversions. For example, you can use event rules to insert information into a table or delete one or more rows in a table based on certain conditions.

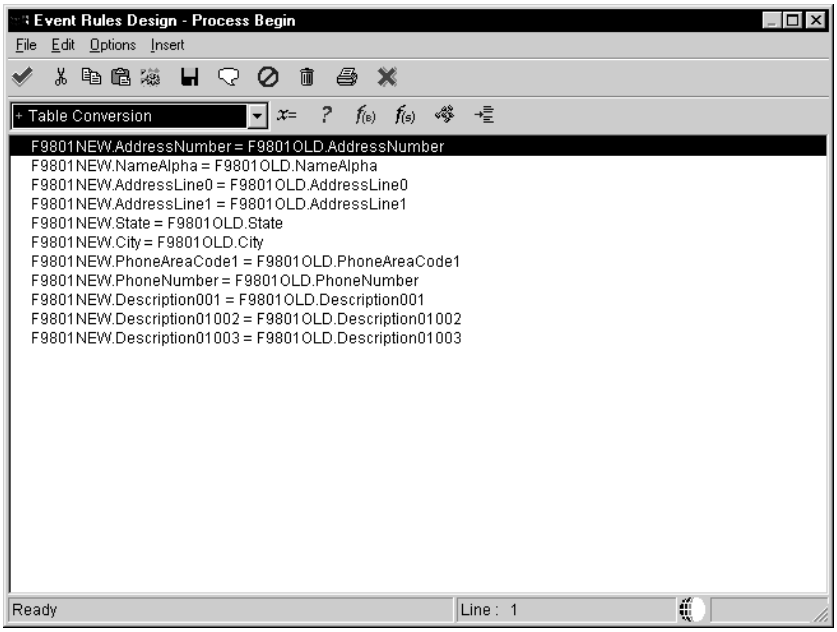
You attach event rules to a particular event, such as Process Begin, Row Fetched, Format Fetched, and Process End.

See *The Flow of Events in Table Conversion* for more information about table conversion events.

Event rules in table conversion include system functions that are specific to the table conversion tool. For a list of these system functions and an explanation of each, see *Table Conversion System Functions* at the end of this chapter.

▶ To use event rules in a table conversion

1. On the appropriate form in a data conversion, data copy, or data copy with table input (or Data Selection for Batch Delete), choose the event to which you want to add event rules from the Events drop-down list.
2. Click the Advanced ER button.



3. On Event Rules Design, choose any of the following buttons to define specific business logic:

   - Assignment: An assignment defines a field as a fixed value or a mathematical expression. For example, you can create an assignment that calculates a value rather than writing a business function to calculate it.
   - If/While: Create If and While logic statements, which are conditional instructions for an event rule.
   - Business Function: You can attach an existing business function, such as a function that retrieves a next number for a new customer, or a function that converts Julian dates to month, day, and year.
   - System Function: You can attach an existing J.D. Edwards system function, such as Copy Table Environment or User Insert Row.

     See *Table Conversion System Functions* for more information. You can also look them up online under *OneWorld Tools APIs*.

   - Variables: You can attach variables to accumulate totals, attach variables that conditionally control what you write to a file, keep a tally of the number of records you read in, and so on.
   - Else: Create Else logic statements. When you create an If statement, an Else statement is automatically inserted after the If statement.
   - Table I/O: Table I/O allows you to open tables in the input, output, or login environment, and also allows you to open the same table twice. It also allows you to pull in data from tables other than the input table and use data from those tables to create an output record. For example, you might want to set up a table conversion that loops through records in F0101 (Address Book Master) and copies them to another table, then loops back through the records, finds each customer that has a certain employee as a contact, and copies that information to the output table as well.

     See *Table I/O* in the *Development Tools Guide* for more information.

   - Report Interconnect: You use report interconnect to connect a batch process or report to the table conversion.

4. After defining your event rules, click OK.
5. Repeat steps as necessary for the different input and output formats.

Table Conversion System Functions

An explanation of each system function within event rules that you can use within a table conversion follows:

CopyTableDataSource	Use this system function to copy a table or range of tables from one data source to another. The system copies tables based on the specifications in the login environment.
CopyTableEnvironment	Use this system function to copy a table or range of tables from one environment to another. The system copies tables based on the specifications in the input and output environment. If the specifications differ, the system performs a "map and drop," meaning that it creates a mapping between like fields in the source and destination tables, and all other fields are ignored.
TCInsertRow	This system function is inserted by the table conversion system when you choose the "Issue a write for this event?" option, and it cannot be moved. It instructs the system that data should be written to the output table.
UserInsertRow	Use this system function to specify when and where a row should be inserted into the specified output table.
DeleteCurrentInputRow	Use this system function to delete the current record from the input table.
UpdateCurrentInputRow	Use this system function to update the current record in the input table after it has been changed.
SetSelectionAppendFlag	Use this system function to determine whether selection criteria added by the system function SetUserSelection will be appended to or replace the existing selection criteria on the input table.
SetUserSelection	Use this system function to conditionally modify data selection on the input table. Call SetSelectionAppendFlag prior to calling SetUserSelection to determine whether to replace or append to the existing data selection information on the input table.

See Also

- *Event Rules Design* in the *OneWorld Development Tools Guide* for more information about event rules.

Running a Table Conversion

When you run a table conversion, you submit it using a batch version. To track what happens during the conversion process, you can turn on tracing, which writes the details of what happened during the conversion to a log. You can set the trace level to control the detail of the log information. When you test a table conversion, you can designate that the conversion proceeds one row at a time, this designation can help you isolate problems or unexpected results.

The following topics are described:

- Submitting a table conversion
- Testing a table conversion

Submitting a Table Conversion

After you create your table conversion, you submit it using a batch version. When you submit the batch version, you can prompt the system to override the properties, such as input and output environment or trace level, and override the location at which your table conversion will process.

Complete the following tasks:

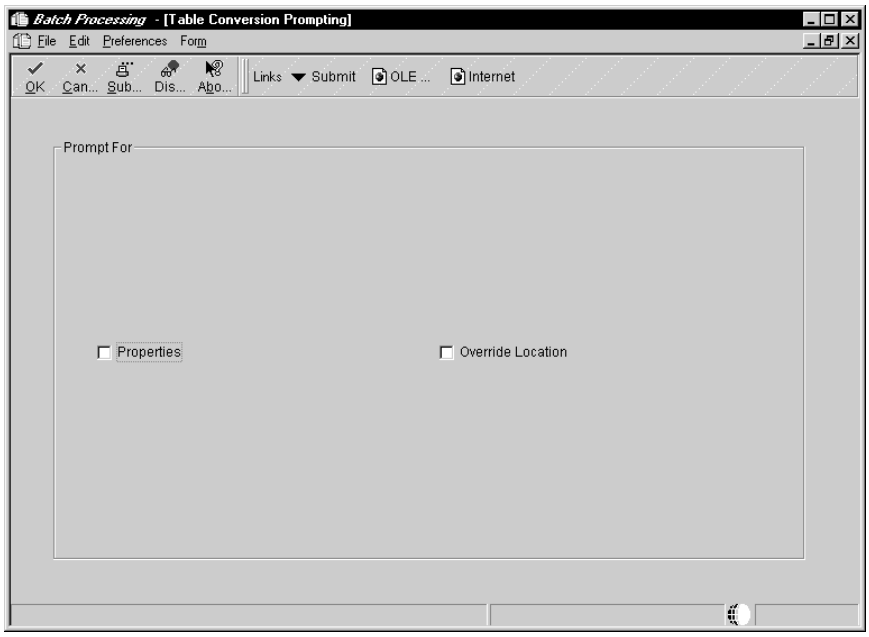
- Override the table conversion properties
- Override the table conversion location

▶ To override the table conversion properties

Note that if you click Cancel while overriding the properties, OneWorld will not save your changes to the version. To change the properties of the version without running it, on Work With Batch Versions, choose Properties from the Row menu.

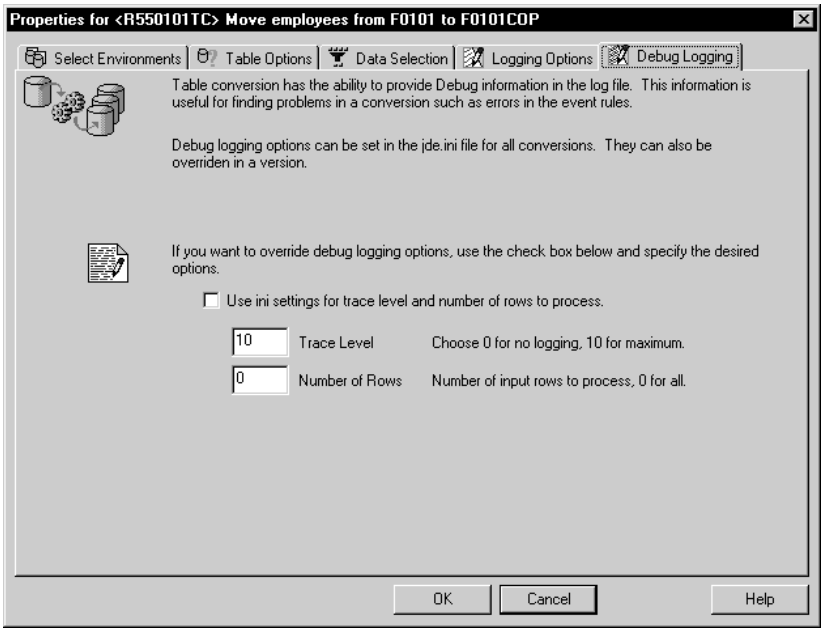
1. On Work With Batch Versions, add a batch version if you did not choose the Auto Create option when you set up the table conversion.
2. On Work With Batch Versions, highlight the version and click Select.

   The system displays Table Conversion Prompting.



3. On Table Conversion Prompting, choose the Properties option and click Submit.

   The system displays the Properties form.



4. On Properties, you can review and override the environments, table options, data selection, and logging options that you specified within the conversion. Note that you can edit these options the same way as when you set up the table conversion. The forms are basically the same.
5. To turn on debug logging, click the Debug Logging tab and do the following:
   - To use the jde.ini settings for the trace level and row-by-row conversion process, make sure that the "Use ini settings for trace level and number of rows to process" option is turned on. The system will use only the settings contained in your jde.ini file and overrides any values that you enter in the Trace Level and Number of Rows fields described below.
   - To override the trace level in the jde.ini file, turn off the "Use ini settings for trace level and number of rows to process" option. Enter a number from 0 to 10 in the Trace Level field.
   - To convert a specific number of records (for example, if you want to test the table conversion), turn off the "Use ini settings for trace level and number of rows to process" option. Enter the number of rows that you want to convert in the Number of Rows field. If you enter 0 for this value, the system processes all rows.

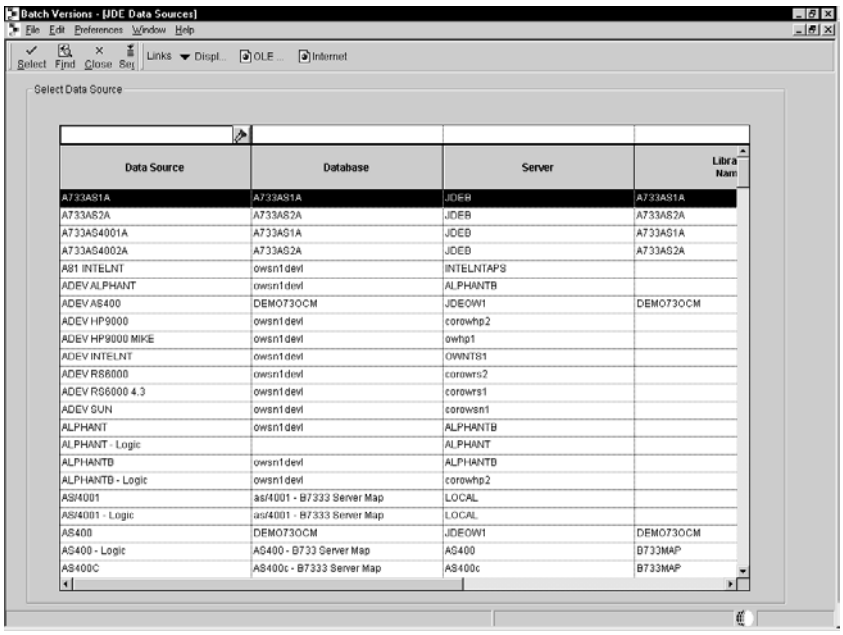
     This option corresponds to the StopAfterRow setting in the jde.ini file. If you enter a value here, you will override any specifications you added to the jde.ini file.

     See *Testing a Table Conversion* for more information about StopAfterRow.
6. Click OK to save your changes to the version. The system submits the table conversion.

▶ To override the table conversion location

You can override the location where you want to process your table conversion if, for example, the server that you normally use is inoperable.

1. On Work With Batch Versions, choose the version and click Select.
2. On Table Conversion Prompting, turn on the Override Location option and click Submit.



3. On JDE Data Sources, choose the data source that you want to use as your override location and click Select.

Testing a Table Conversion

You might want to test a table conversion to ensure that there are no errors within it. To do this, you can log debug information about the conversion while it runs. You can also force the conversion to run one row at a time, which is useful if the conversion is normally running as an insert-from-select.

To log debug information about a table conversion, enable tracing and set a trace level from 0 to 10 in the jde.ini file according to how specific that you want the logged information to be.

You set debug logging in the jde.ini file, but you can override the jde.ini settings through the version of a conversion, if necessary.

If you set the trace level for logging at 1, the system logs basic information about the table conversion, such as name, inputs, outputs, event rule logic, and how many rows were inserted, if applicable. If you set your trace level at 10, the system logs all information about every column in every format, including user-defined formats, whether there is a processing option template associated with the table conversion, and all other information that is involved in the table conversion process. In other words, the higher you set the trace level, the more information the system supplies about the table conversion process.

When you test a table conversion you might want to force a row-by-row conversion, along with setting a trace level on your workstation or on the server, depending on where you want to run the conversion.

The following topics are described:

- Setting a trace level for debug logging
- Forcing row-by-row conversion

The Difference Between Logging Options and Debug Logging

The logging options that you specify when you set up a conversion can log all errors that occur during the conversion, or log all records that are copied, deleted, or updated. They can also log the details of copy table actions.

Debug logging, on the other hand, logs more detailed information about the conversion, which helps you pinpoint the exact area in the conversion where errors have occurred.

Note: Any debug logging changes that you make to the version will override the settings in the jde.ini file.

Trace Levels

The following table gives more specific information about each trace level.

Trace Level	Logging Information
Level 1	Logs general information about the conversion, such as name, inputs, outputs, event rule logic, and how many rows were inserted, if applicable.
Level 2	Logs function call traces, such as starting conversion, ending conversion, inserting rows.
Level 3	Logs the points at which event rules are being executed.
Level 4	Not applicable.
Level 5	Logs the points at which jdeCallObject is executed, such as calls to business functions from event rules.
Levels 6-8	Not applicable.
Level 9	Logs the content of columns during input, event rules, and prior to output.
Level 10	Logs all information contained in the first nine levels.

Caution: J. D. Edwards recommends that you do not set your trace level at 10 when running a table conversion over tables that contain a large amount of data. The system will write a large amount of data onto your server, which could

cause it to run out of disk space. You can, however, specify a certain number of rows to run in the version of the table conversion by choosing the Properties option. See *Submitting a Table Conversion* for more information.

Setting the Trace Level for Debug Logging

You can set a trace level for debug logging on a workstation or on a server, depending on where you are running the conversion.

Complete the following tasks:

- Enable tracing and set the trace level on a workstation
- Enable tracing and set the trace level on the server

▶ To enable tracing and set the trace level on a workstation

To enable tracing and to set the trace level on a workstation, add the following to the jde.ini file on that workstation:

```
[TCEngine]

TraceLevel=n
```

(Where *n* is a number from 0 through 10.)

```
[Debug]

Output=File

[UBE]

UBESaveLogFile=1
```

▶ To enable tracing and set the trace level on a server

To enable tracing and to set the trace level on a server, add the following to the jde.ini file on that server:

```
[TCEngine]

TraceLevel=n
```

(Where *n* is a number from 0 through 10.)

```
[Debug]

Output=File

KeepLogs=1
```

See Also

- *OneWorld System Administration* guide for more information about tracing
- *OneWorld Server and Client Administration* guide for information about reading logs

Forcing Row-By-Row Conversion

You can set the system to force a row-by-row conversion when you want to test your table conversion. You can force row-by-row on your workstation or the server. You can also specify a certain number of rows to process in conjunction with forcing a row-by-row conversion.

Complete the following tasks:

- Force row-by-row conversion
- Specify the number of rows to process

▶ To force row-by-row conversion

To force row-by-row conversion, add the following to the jde.ini file:

```
[TCEngine]

ForceRowByRow=1
```

To specify the number of rows to process

To specify the number of rows to process, add the following to the jde.ini file beneath the `[TCEngine]` header:

`StopAfterRow=`*n*

(Where *n* is the number of rows you want to process.)

Preparing Non-OneWorld Tables for Table Conversion

Non-OneWorld tables are text files, or any other file or database that is not recognized by OneWorld, as long as the database is a type supported by OneWorld, Oracle, Access, AS/400, or SQL Server.

Before you can work with non-OneWorld tables in the table conversion tool, you need to make them known to OneWorld. To do this, you must set up an ODBC data source for the foreign tables, and then set up a data source, environment, and OCM mapping in OneWorld that points to the ODBC data source.

Note: When you work with foreign tables, your database administrator needs to address database authority issues. Your OneWorld user ID (or, if you are using the OneWorld security server feature, the database user to which it maps) must be changed so that you will have authority to use the tables in the foreign database. Without this authority, you will not be able to see the tables in the design tool. Under certain conditions, the table conversion engine will need to create temporary tables in the output environment and will require create-and-drop authority for the database.

The following tasks are described:

- Adding a OneWorld data source
- Adding a OneWorld environment
- Setting up a default OCM mapping

Adding a OneWorld Data Source

Before you can add a OneWorld data source for the foreign table, you must first add a Microsoft ODBC data source or an Oracle OCI data source that points to the foreign table. For complete information about ODBC drivers and data sources, consult the appropriate Microsoft or Oracle documentation.

After you have added an ODBC or Oracle data source, you need to add a data source in OneWorld that points to the data source that you just set up.

▶ To add a OneWorld data source

1. On the System Administration Tools menu (GH9011), choose Database Data Sources.

   The Machine Search & Select form appears.

2. Highlight the machine on which the data source resides and click Select.

   The Work with Data Sources form appears.

3. Click Add.

   The Data Source Revisions form appears.

4. Complete the following fields and click OK:

   - Data Source Use

     Enter DB in this field to identify the data source as a database data source. You use only database data sources when accessing data in tables.

   - Data Source

     The data source name can be different from the ODBC or Oracle database name, if necessary.

   - Data Source Type

     Enter A (Access) in this field if you are using an Access data source or text data source.

- Object Owner ID
- Library Name
- library list name
- DLL Name

  Enter the name of the DLL you want to use to access the foreign table. The correct DLL will depend on the platform and the database of the foreign table.

  To see a list of the DLLs and their use, see the online help.

- Database Name

  This name should be the same as the ODBC or Oracle data source name.

- Server Name
- Platform

5. Click OK.

Field	Explanation
Data Source Use	Indicates how the data source is configured, Servers (SVR) to run UBE's and Business Functions, or a Database (DB) to access table data.
Data Source Type	The type of database.
Object Owner ID	The database table prefix or owner.
library list name	The name associated with a specific list of libraries. The J98INITA initial program uses these library list names to control environments that a user can sign on to. These configurations of library lists are maintained in the Library List Master table (F0094). For OneWorld, this field represents a valid environment that can be used to run OneWorld. The environment encompasses both a path code (objects) and a data source (data). When put together, users have a valid workplace within OneWorld.

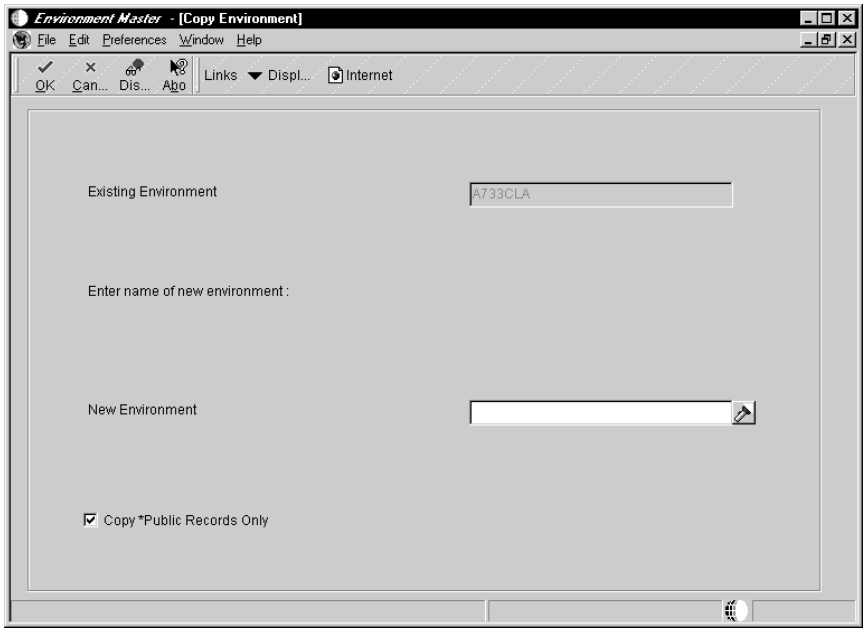
Field	Explanation
DLL Name	Specifies the name of the database driver file. This file is specified in the [DB SYSTEM SETTINGS] section of the enterprise server jde.ini file. The file you specify depends upon the platform and the database. Values for specific machines and databases are: AS/400 to DB2/400 = DBDR AS/400 to any other server DBMS = JDBNET HP9000 to DB2/400 = libjdbnet.sl HP9000 to Microsoft SQL Server = libjdbnet.sl HP9000 to Oracle (Version 8.0) UNIX = libora80.sl RS6000 to DB2/400 = libjdbnet.so RS6000 to Microsoft SQL Server = libjdbnet.so RS6000 to Oracle (Version 7.3) UNIX = libora73.so RS6000 to Oracle (Version 8.0) UNIX = libora80.so Intel to AS/400 = jdbodbc.dll Intel to Oracle (Version 7.2) NT = jdboci32.dll Intel to Oracle (Version 7.3) NT = jdboci73.dll Intel to Oracle (Version 8.0) NT = jdboci80.dll Intel to SQL Server NT = jdbodbc.dll Digital Alpha to AS/400 = jdbnet.dll Digital Alpha to Oracle (Version 7.2) NT = jdboci32.dll Digital Alpha to Oracle (Version 7.3) NT = jdboci73.dll Digital Alpha to Oracle (Version 8.0) NT = jdboci80.dll Digital Alpha to SQL Server NT = jdbodbc.dll
Database Name	The name assigned to the database during installation, such as HPDEVORAP or HP9000.
Server Name	The name of the computer that receives documents from clients.
Platform	The type of physical hardware the database resides on.

Adding a OneWorld Environment

For each ODBC data source, database instance, or library that contains non-OneWorld tables, you must set up an environment. The environment points to the OneWorld data source, which in turn points to the database or library. The easiest way to add an environment is to copy an existing one. If you need more information about environments, see the *CNC Implementation Guide*.

▶ To add a OneWorld environment

1. On Environments (GH9053), choose Environment Master.
2. On Work With Environments, find and highlight the environment that most closely matches the environment that you want to add, (such as the environment you are logged into, or any other environment you can access from your workstation), and then choose Copy Environment from the Row menu.



3. On Copy Environment, type an environment name in the New Environment field.
4. To copy only the *PUBLIC Object Configuration Manager mappings of an environment, choose the *PUBLIC Records Only option.

   Leave this option turned off to copy mappings for the environment, individual users, and *PUBLIC.

5. Click OK.

See Also

- *Working with an Environment* in the *CNC Implementation Guide* for more information about adding environments.

Setting Up a Default OCM Mapping

You map objects, such as tables, by environment. When you set up a default OCM mapping, you select an environment that you have already created and map that environment's objects to the data sources where those objects exist.

You create a default map for a TBLE object type. You create a mapping of an object name with a literal value of DEFAULT, and then enter an object type (TBLE) and a data source. When you create a default map for the object type TBLE, any table objects not mapped individually will point to the default data source. In addition, the table conversion tool will use this mapping for foreign tables.

Each environment must have a default map for table objects for the *PUBLIC user profile because there is no inherent default location for table objects. If table objects do not have a default map and are not explicitly mapped by name, OneWorld produces a Select/Failed error message when it tries to access the tables. Additionally, the tables will not appear in the input or output forms in the Table Conversion Design application.

▶ To set up a default OCM mapping

1. On the System Administration Tools menu (GH9011), choose Object Configuration Manager.

   The Machine Search and Select form appears.

2. Choose the data source that stores the Object Configuration Manager table with which you want to work, and the click Select.

   The Work with Object Mappings form appears.

   See *Understanding Data Sources* and *Working with the Object Configuration Manager* in the *Configurable Network Computing Implementation Guide* for more information about data sources.

3. Click Add.

   The Object Mapping Revisions form appears. On this form, you determine to what data source your table will map.

4. Enter the following information:

   - Environment Name

- Object Name

  Enter DEFAULT in this field.

- Data Source

  The primary data source refers to the location within the environment of the object for which you will create a mapping. Enter the data source name that you set up for your foreign tables.

- User Class/Group

  The user is the individual or group for whom the mapping applies. You would normally enter *PUBLIC in this field.

- Data Source Mode

  Enter P in this field. P stands for primary.

- Allow QBE

  You can use this field to indicate whether applications based on the table include a Query By Example line.

5. Click OK to save your object mapping.

   The system displays the mapping that you created with an inactive status.

6. Change the status to active.

You can now access the tables in this data source as foreign tables by using this environment in your table conversion.

See Also

- *Working with the Object Configuration Manager* in the *Configurable Network Computing Implementation Guide* for more information about OneWorld environments, data sources, and OCM mapping

Field	Explanation
Environment Name	For World, the Environment name is also called the Plan Name and is used to uniquely identify an upgrade environment for Install/Reinstall. For OneWorld (Install Applications), the environment name is also called the Plan Name and is used to uniquely identify an upgrade environment for Install/Reinstall. For OneWorld (Environment or Version Applications), this is the path code that identifies the location of the application or version specification data.
Object Name	The OneWorld architecture is object-based. This means that discrete software objects are the building blocks for all applications, and that developers can reuse the objects in multiple applications. Each object is tracked by the Object Librarian. Examples of OneWorld objects include: • Batch Applications (such as reports) • Interactive Applications • Business Views • Business Functions • Business Functions Data Structures • Event Rules • Media Object Data Structures
User Class/Group	A profile used to classify users into groups for security purposes. Some rules for creating a User Class/Group are as follows: • The 'Class/Group' profile must begin with * so that it does not conflict with any System profiles. • The 'User Class/Group' field must be blank when entering a new group profile.
Data Source Mode	Indicates whether OneWorld uses the primary or secondary data source. Valid values are: P Primary. S Secondary. You should only use this for business function (BSFN) data sources.
Allow QBE	Use this flag to turn ON or OFF row-level record locking for the data source. You should have this flag turned ON to help prevent database integrity issues. JDEBASE middleware uses this flag to determine whether or not to use row-level record locking.

Appendices

Table Conversion Examples

This section contains two examples of table conversions. These examples give you a better understanding of how advanced table conversions can be set up.

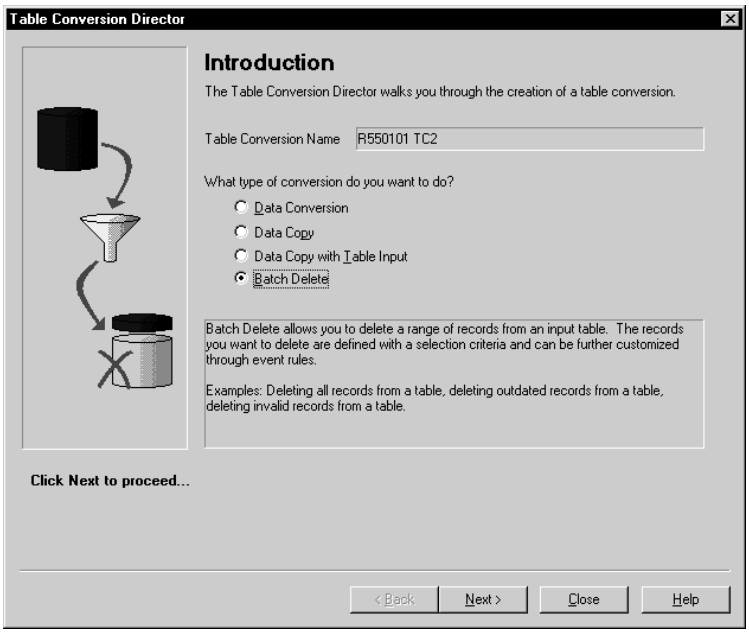
The following examples are described:

- Batch delete/update
- Data conversion/batch delete

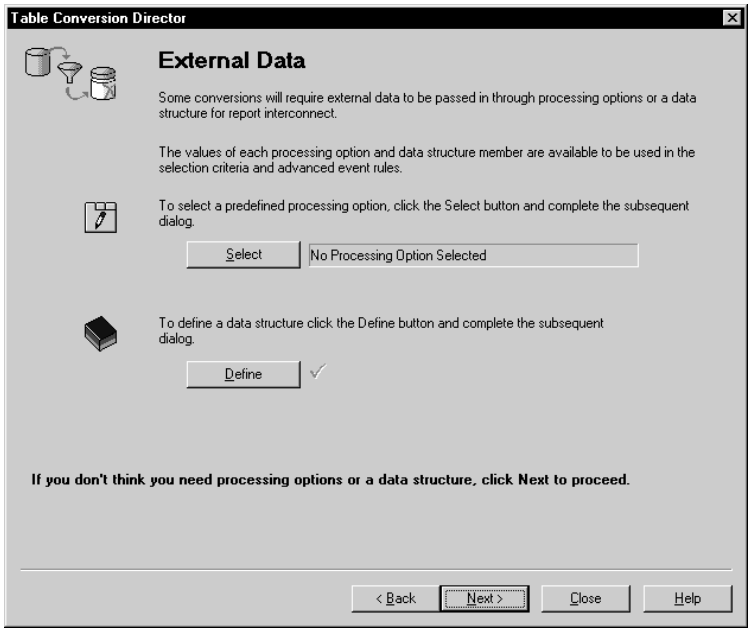
Example 1 - Batch Delete/Update

This table conversion was created using the Batch Delete option, but it does not actually delete records. Instead, it is meant as an example of how you can use a batch delete conversion to do general batch processing over a single table. You can create a table conversion such as this by using the data conversion option, but it takes longer to create and also requires an output table.

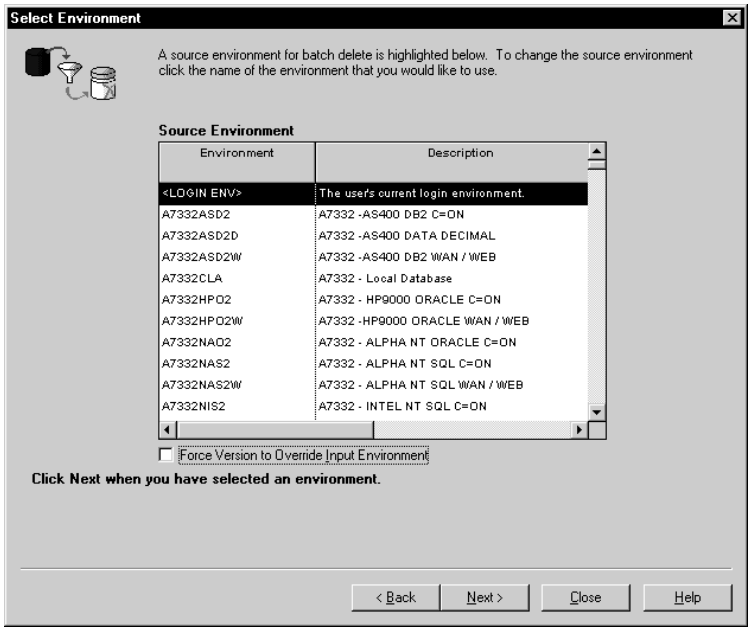
The following example updates all employee records in the F0101 table to make them ex-employees:



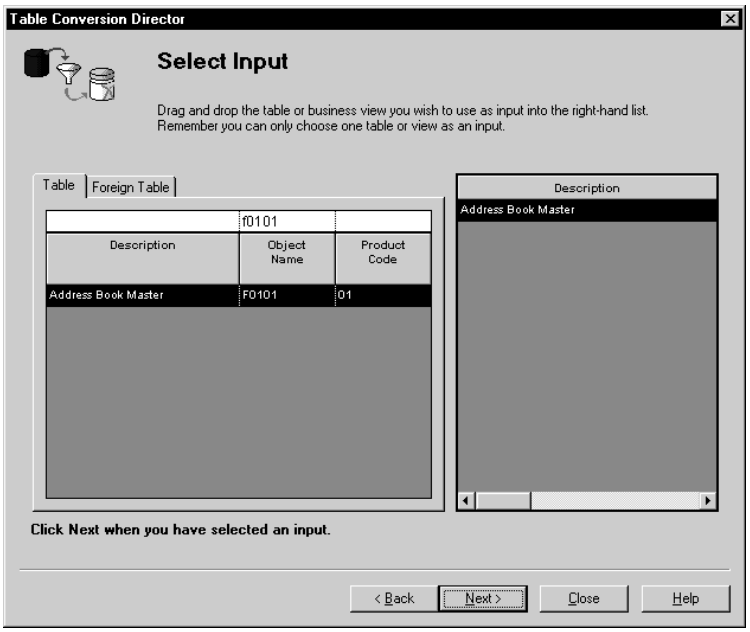
After you choose Batch Delete from the Introduction form of the table conversion director, specify whether you want to include processing options and data structures. In this example, no processing options or data structures are used.



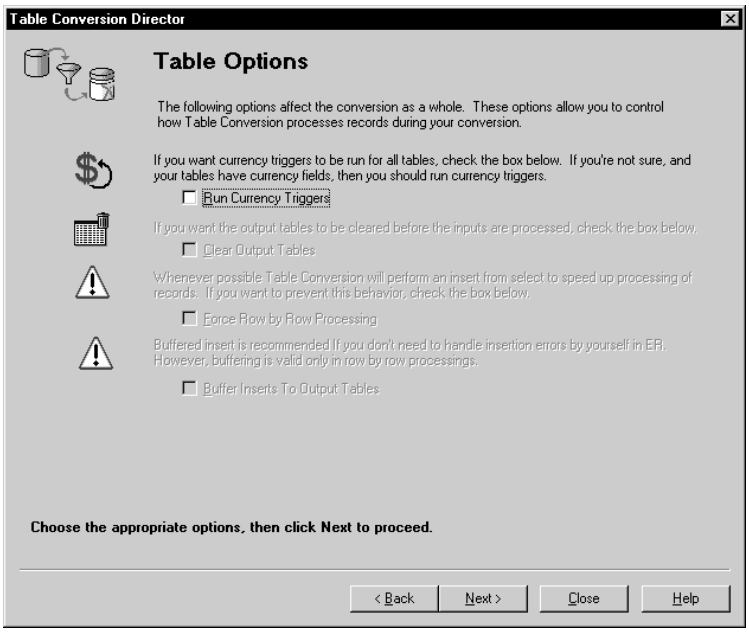
Next, choose the environment in which you want to run the conversion. In this example, create and run the conversion in the login environment. The "Force version to override input environment" option is not chosen because it will not be ported to another system.



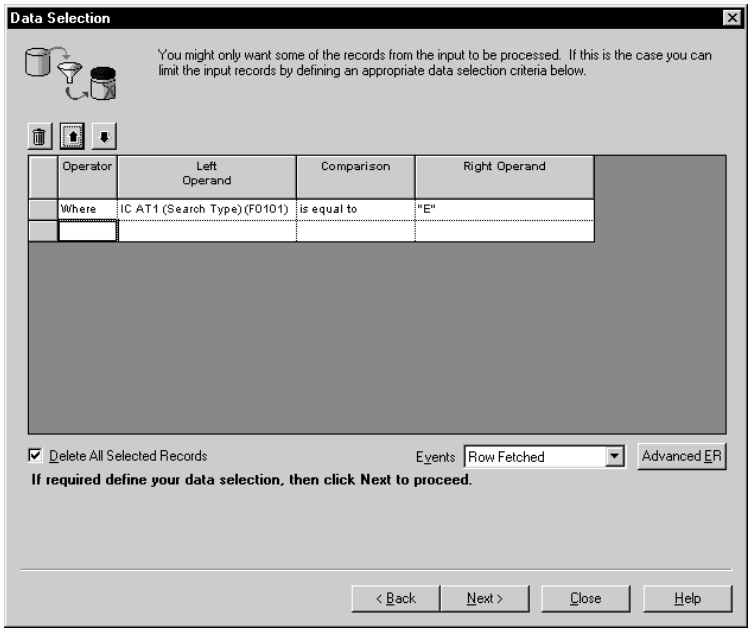
On Select Inputs, the F0101 table is chosen.



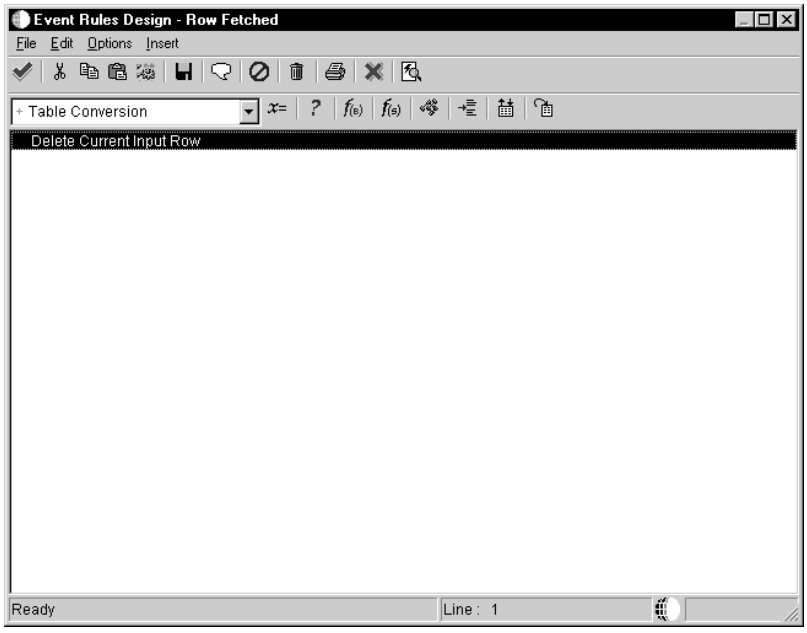
Because the F0101 table does not contain currency fields, the currency trigger option is not chosen for this table conversion.



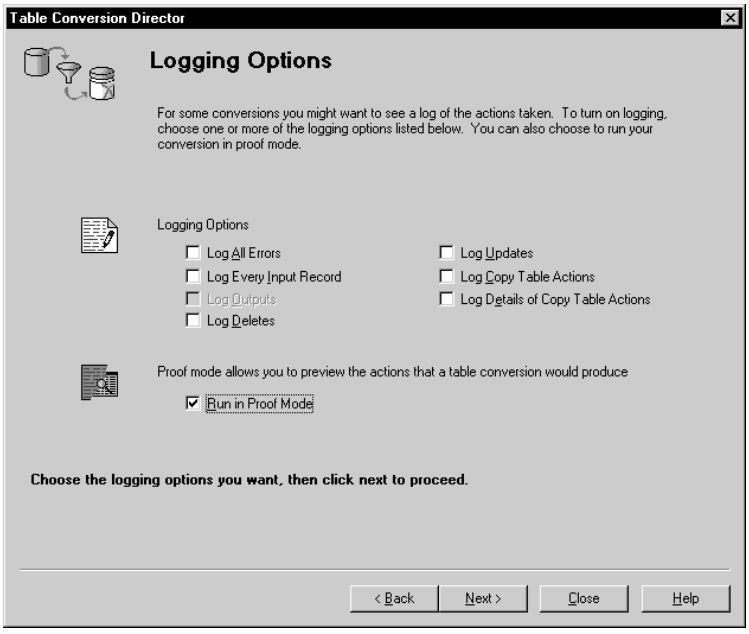
On data selection, specify that all AT1 (Search Type) records that contain E (Employee) should be selected, meaning that all employees will become ex-employees. Note that Delete All Select Records is chosen, as is the Row Fetched event.



When you click the Advanced ER button, you can see the system function that is added when you choose the Delete All Selected Records option on the previous form.



This table conversion will be run in proof mode initially to ensure that all the records that should be changed will, in fact, be changed.



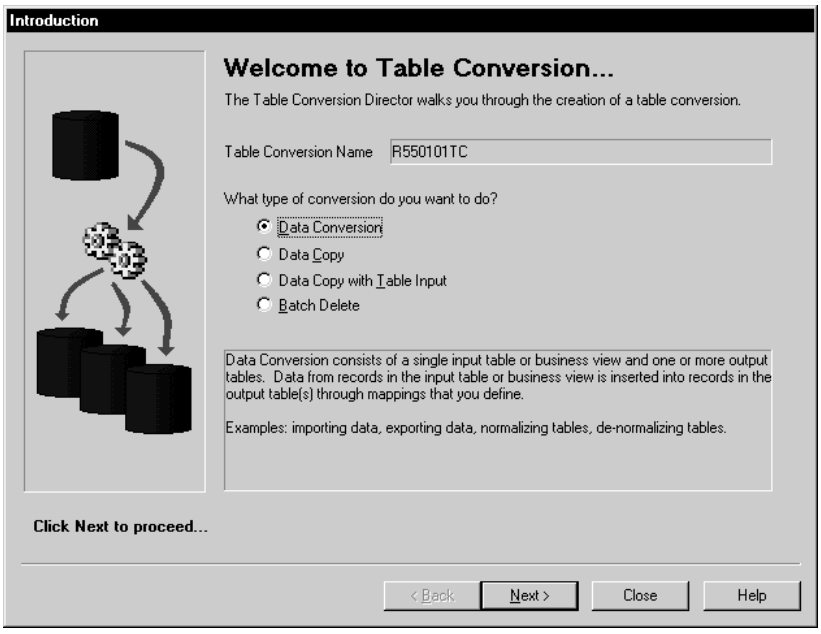
See Also

- *Running a Table Conversion* for more information about turning on logging and setting the trace level

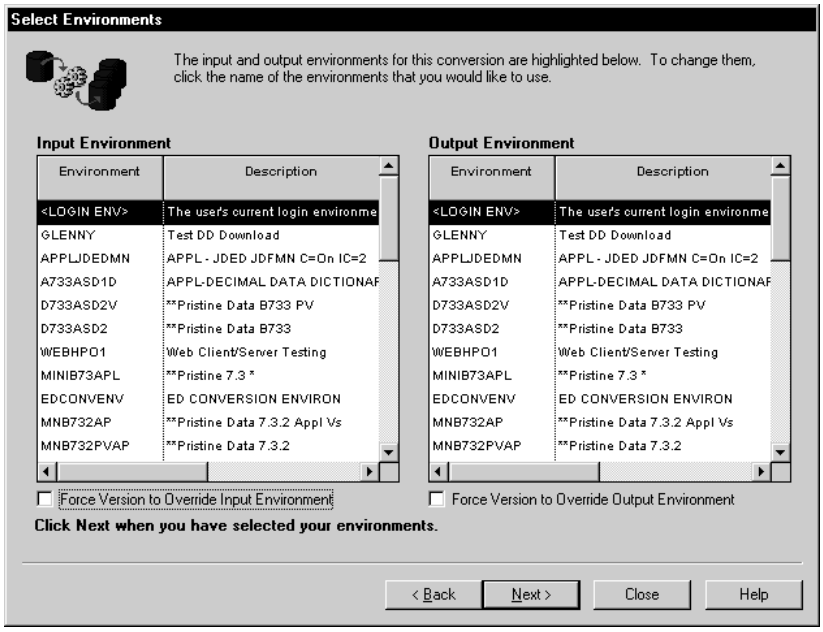
Example 2 - Data Conversion/Batch Delete

The following example is a data conversion that copies employee records from the input environment into a new table and then deletes the records in the input table after the records have been copied.

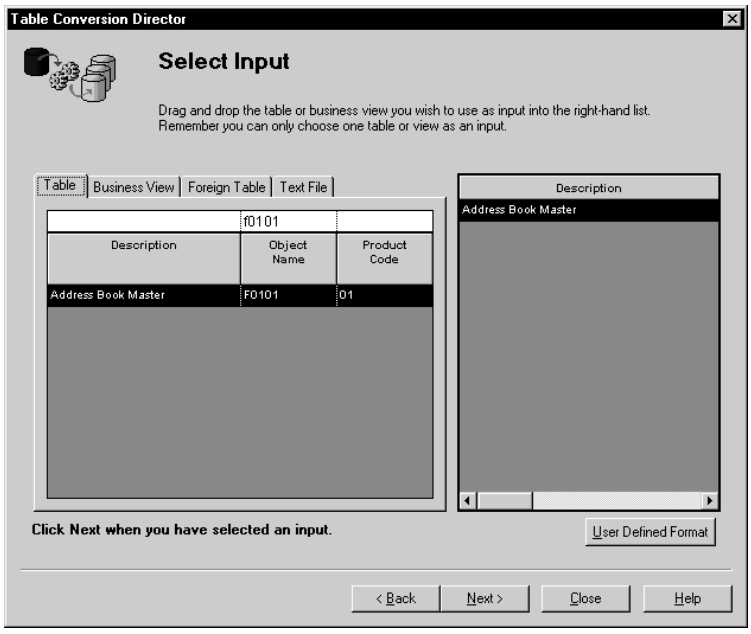
Use the data conversion option to set up this table conversion, and then add event rules in the mapping section of the conversion.



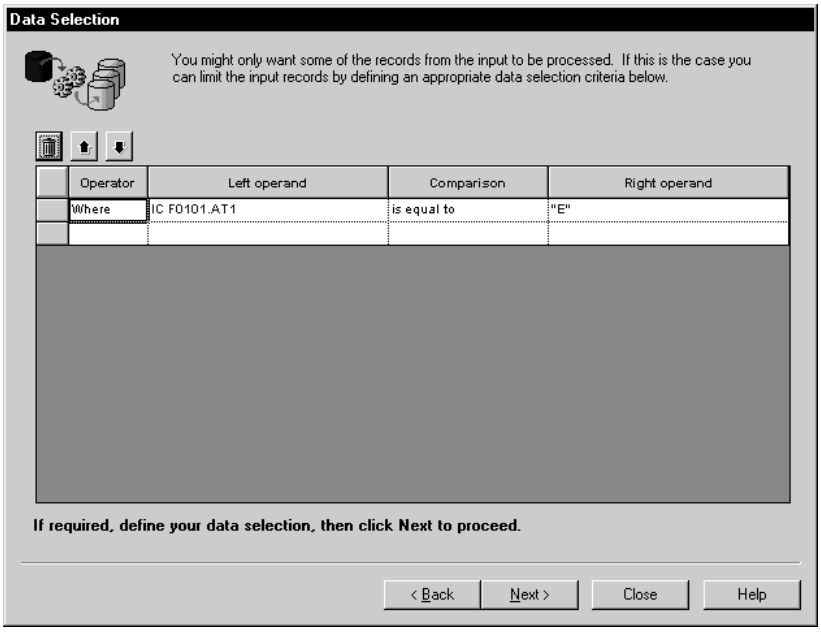
In this example, create and run the conversion in the login environment. The "Force version to override input environment" option is not chosen because it will not be ported to another system.



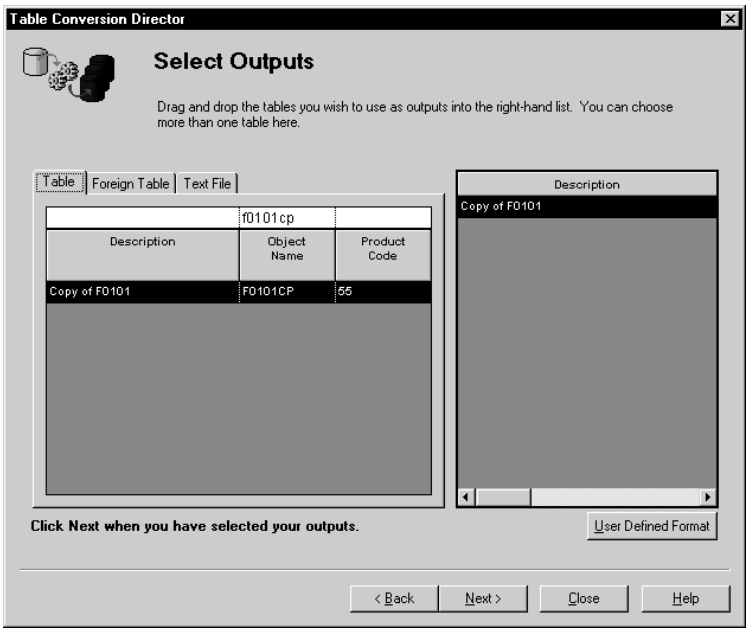
The F0101 table is chosen as the one from which records will be copied to the new table.



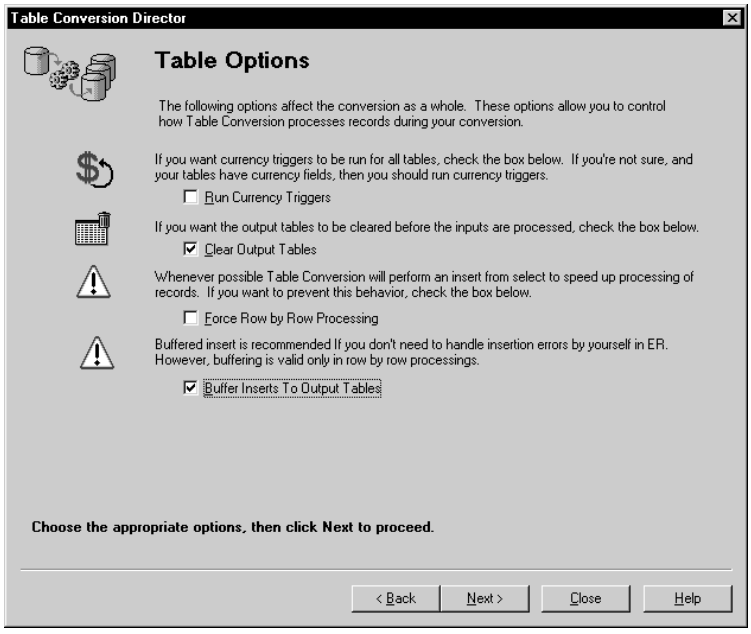
In this conversion, all records with a search type of E (Employee) will be deleted from the input table after they are copied to the output table.



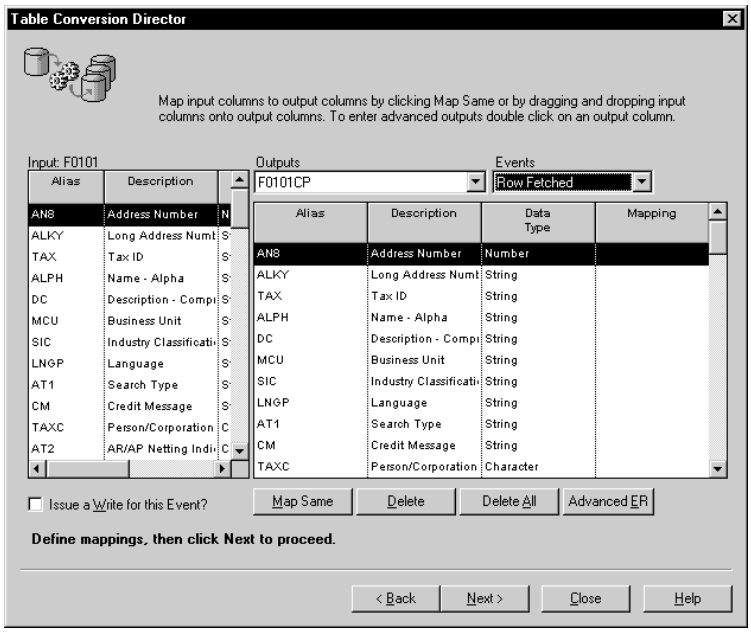
A copy of the Address Book Master (F0101) is made and used as the output table.



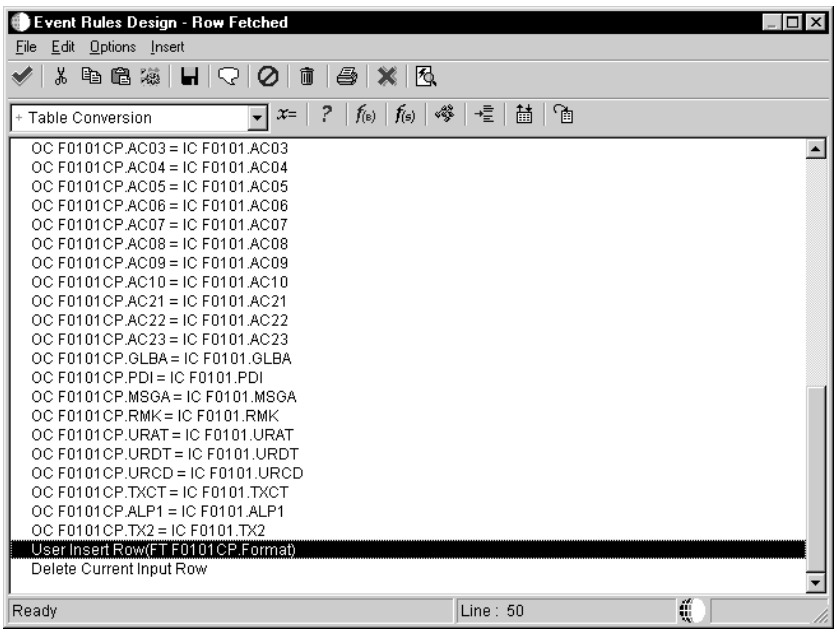
Because the F0101 (Address Book Master) does not contain currency fields, the Run Currency Triggers option is not chosen. However, the Clear Output Tables option is chosen to clear the output table before the conversion is run.



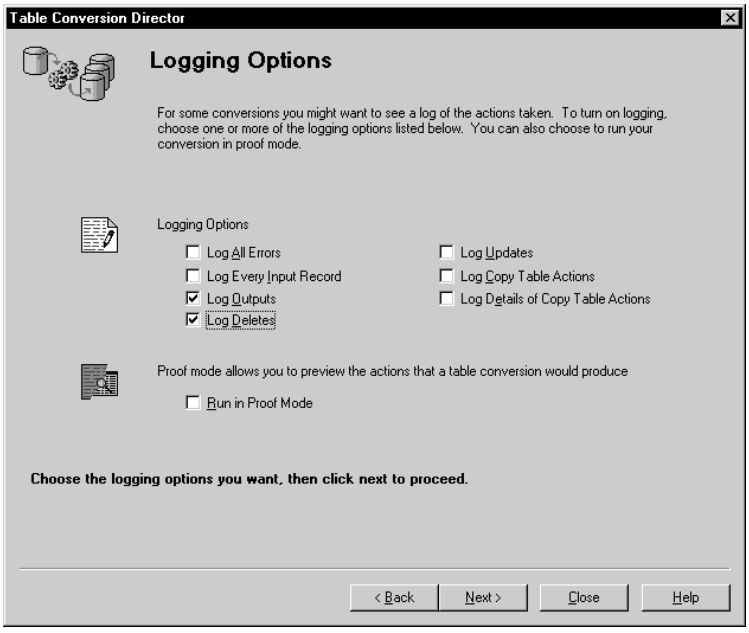
Each time that a row is fetched from the input, the record will be copied to the output and then deleted from the input.



Event rules are added to the Row Fetched Event, as shown in the following example:



In the Mapping section of the conversion, the Delete Current Record Input system function was added immediately after the User Insert Row event. In other words, each time that a record is copied from the input, the system copies the record, and then deletes that record from the input.



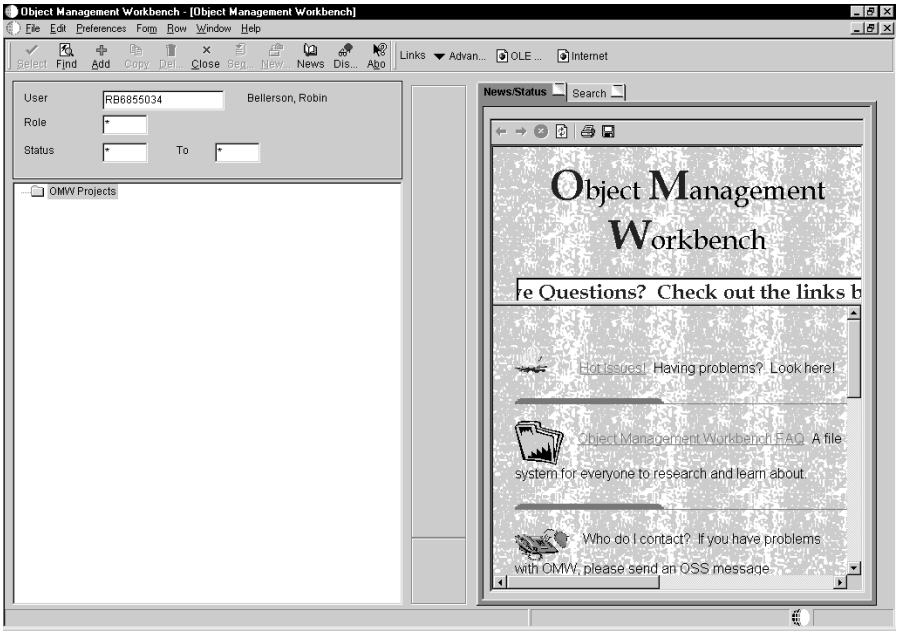
The Log Outputs and Log Deletes options were both turned on for logging options so that all inputs and deletes could be reviewed after the conversion is run.

Example 3 - Creating a Purge Program as a Batch Delete

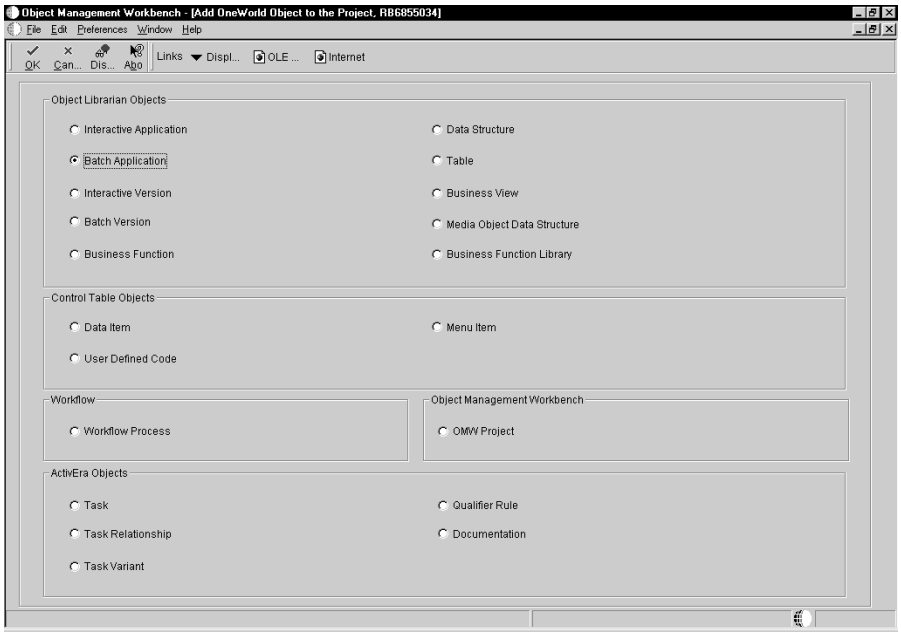
The following example is a table conversion that deletes records from the input environment. Designing purge programs as batch deletes enables you to purge records with better control and greater accuracy. You can archive the data you purge or remove it from the system permanently. This example includes the archiving process.

Before you start this example, create a handle for the table. See *Using a Handle* in the *OneWorld Development Tools* guide for instructions on creating and using handles.

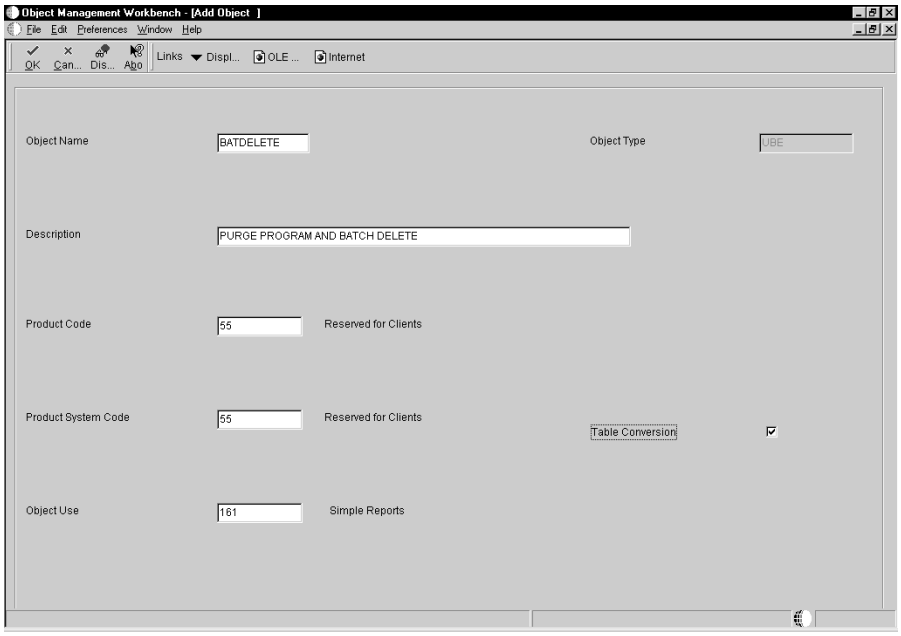
On Object Management Workbench (OMW), click Add.



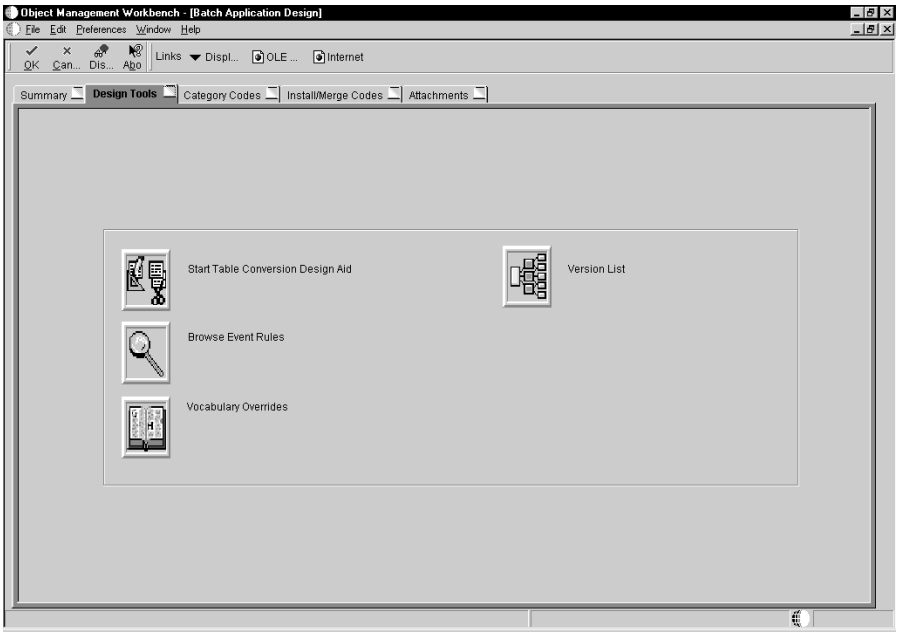
On Add OneWorld Object to the Project, click Batch Application.



On Add Object, enter the appropriate information into the existing fields and choose Table Conversion.

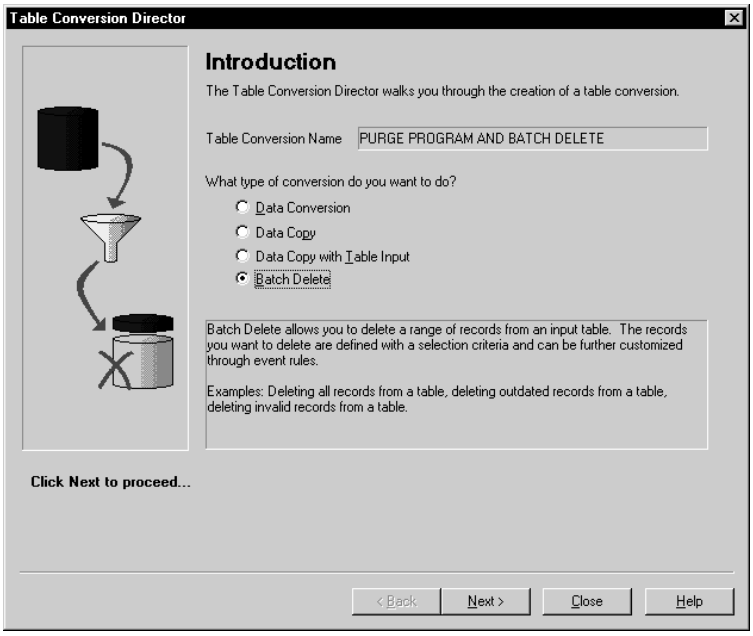


On Batch Application Design, click Design Tools tab.

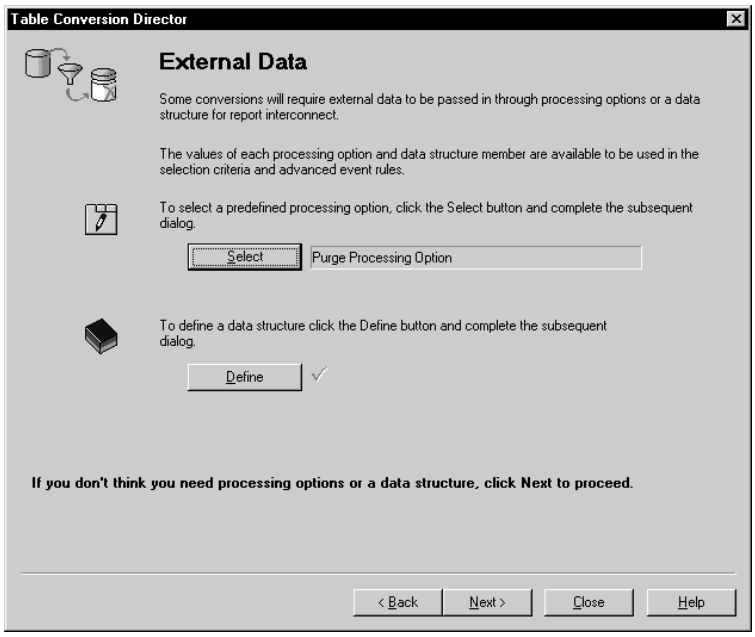


Click *Start Table Conversion Design Aid*.

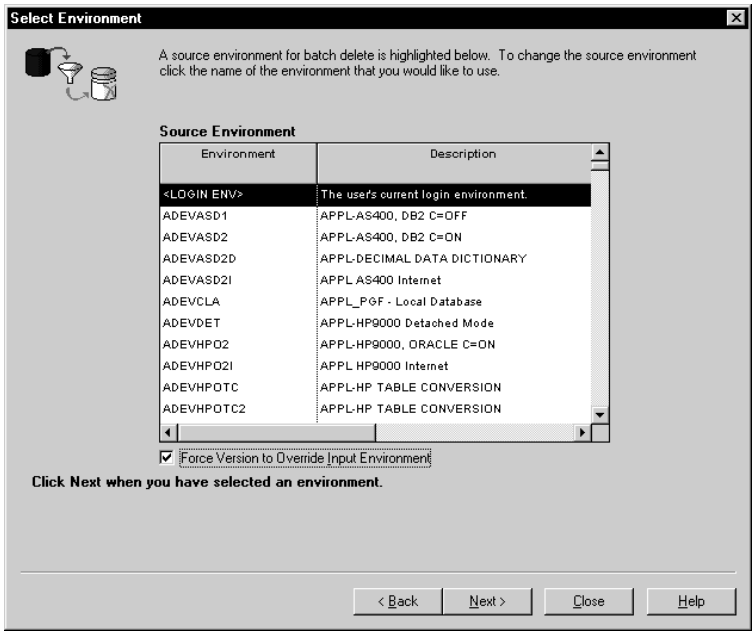
On Introduction, choose Batch Delete.



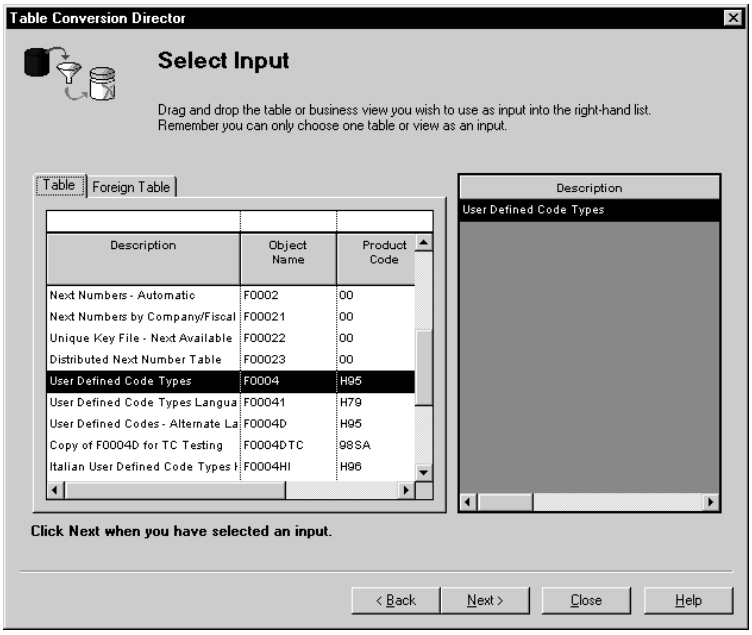
On External Data, choose a Processing Option template. For this example, use Purge Processing Option (T42000P).



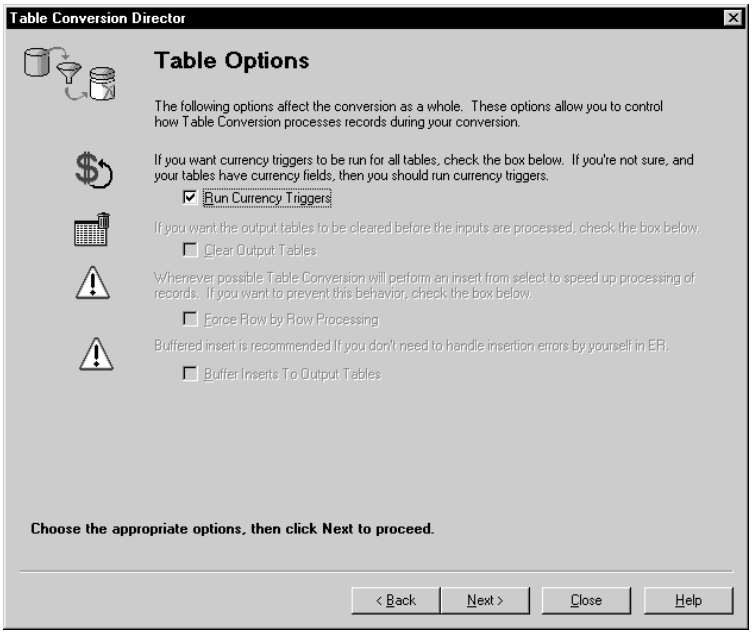
Choose a source environment in which you want to run the batch delete. For this example, choose the login environment. Enable the "Force Version to Override Input Environment" option to make sure that the person who runs the purge program provides a valid source environment from which to run the batch delete.



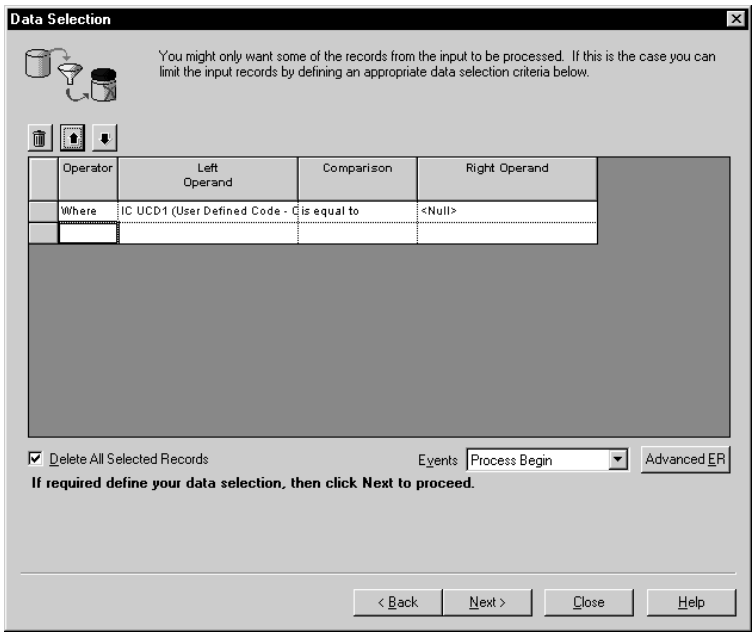
On Select Input, choose the table you want to purge and drag it to the Description area.



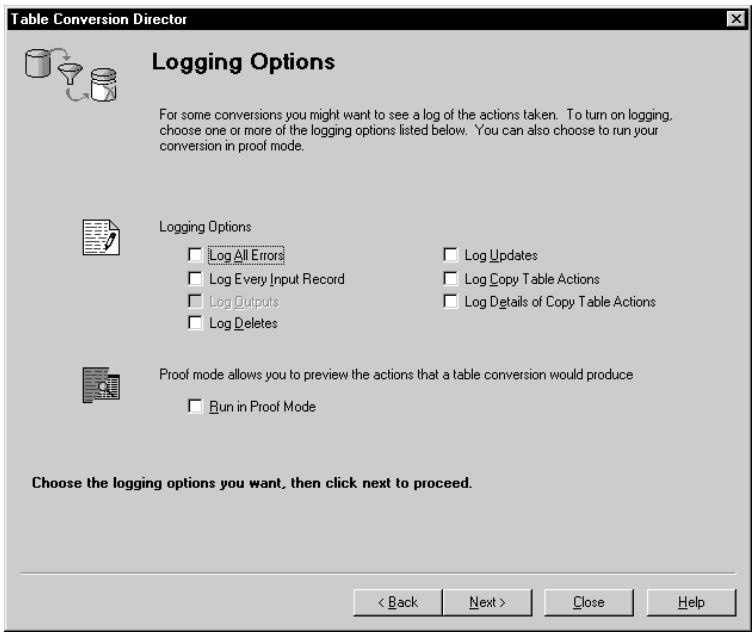
On Table Options, choose Run Currency Triggers. See *Deleting Records* for information about currency triggers.



On Data Selection, choose the data you want purged by clicking in a field and double-clicking an option from the drop-down list.

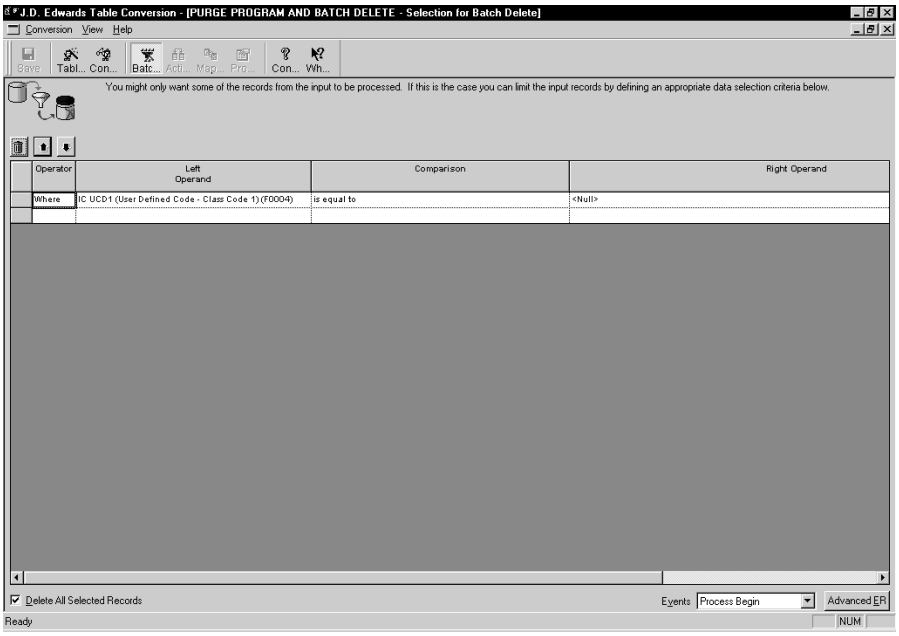


On Logging Options, choose the options you want logged. For this example, do not log any information.



On Finish, choose *Yes, create a version of this table conversion* and enter the version name in the field. For this example, use XJDE001.

On Selection for Batch Delete, choose Process Begin and click Advanced ER.



On Event Rules Design, enter the begin process event rules, along with any special logic. For this example, use event rules for R42119P, as follows:

```
0001 // Check to see if the purged data is being archived
0002 //
0003 If PO cArchiveRecords is equal to "1"
0004    //
0005    // If the environment processing option is blank, stop
processing.
0006    //
0007    If PO szArchiveEnvironmentName is equal to <Blank>
            Or PO szArchiveEnvironmentName is equal to <Null>
0008       Stop Conversion Processing("The archive environment is
invalid.")
0009    Else
0010       //
0011       // Check to make sure that the archive environment and data
source is not the
0012       // same as the input environment and data source
0013       //
0014       If PO szArchiveEnvironmentName is equal to SL
SourceEnvironment
0015          Stop Conversion Processing("The source and archive
environments are the same")
0016       Else
0017          Get and validate the data source for an environment /
table  (B98700)
                     PO szArchiveEnvironmentName -> szEnvironment
                     "F42119" -> szTableName
                     VA rpt_szArchiveDataSource_DATS <- szDataSource
                     VA rpt_szErrorCode_DTAI <- szErrorDataItem
                     VA rpt_mnErrorNumber_MATH01 <- mnErrorNumber
0018          If VA rpt_szErrorCode_DTAI is not equal to <Blank>
                  And VA rpt_szErrorCode_DTAI is not equal to <Null>
0019             Stop Conversion Processing("No data source was found
for the archive environment")
0020          Else
0021             Get and validate the data source for an environment /
table  (B98700)
                        SL SourceEnvironment -> szEnvironment
                        "F42119" -> szTableName
                        VA rpt_szPurgeDataSource_DATS <- szDataSource
                        VA rpt_szErrorCode_DTAI <- szErrorDataItem
                        VA rpt_mnErrorNumber_MATH01 <- mnErrorNumber
0022             If VA rpt_szArchiveDataSource_DATS is equal to VA
rpt_szPurgeDataSource_DATS
0023                Stop Conversion Processing("The source and archive
environments have the same data source")
0024             Else
0025                //
0026                // Open a table with the same table name in the
output environment. The table
0027                // will be renamed later if the table name
processing option was populated.
0028                //
0029                Copy Table Environment("F42119",  <None>, SL
SourceEnvironment              , PO szArchiveEnvironmentName, <Yes>,
<Yes>, <No>, <None>, <None>, <Null>, <Null>)
0030                //
0031                // Open a handle to the archive table
0032                //
0033                VA rpt_F42119Handle_HF42119 = F42119.Open Handle
0034                If VA rpt_F42119Handle_HF42119 is equal to <Null>
0035                   Stop Conversion Processing("Failed to open F42119
in the archive environment")
```

```
0036                End If
0037             End If
0038          End If
0039       End If
0040    End If
0041 End If
```

Using this example, the system writes log messages on Stop Conversion Processing to the JDE.log and JDEDEBUG.log files.

After you have entered the begin process event rules, add the following variables:

- FXXXXHandle_HFXXXX
- szArchiveDataSource_DATS
- szPurgeDataSource_DATS
- szErrorCode_DTAI
- cRenameFlag_EV01
- mnErrorNumber_MATH01

Make sure that you have mapped all parameters to a field, even if you will not use every value.

On Selection for Batch Delete, choose Row Fetched and click Advanced ER. On Event Rules Design, enter the row fetched event rules, along with any special logic. Make sure that you have mapped all parameters to a field, even if you will not use every value. This example includes event rules for R42119P, as follows:

```
0001 //
0002 // If we are archiving the purged records, write the record to the
archive table
0003 //
0004 If PO cArchiveRecords is equal to "1"
0005    F42119(VA rpt_F42119Handle_HF42119).Insert
           IC Order Company (Order Number) -> TK Order Company (Order
Number)
           IC Document (Order No, Invoice, etc.) -> TK Document (Order
No, Invoice, etc.)
           IC Order Type -> TK Order Type
           IC Line Number -> TK Line Number
           IC Order Suffix -> TK Order Suffix
           IC Business Unit -> TK Business Unit
           IC Company -> TK Company
           IC Document Company (Original Order) -> TK Document Company
(Original Order)
           IC Original Order Number -> TK Original Order Number
           IC Original Order Type -> TK Original Order Type
           IC Original Line Number -> TK Original Line Number
           IC Company - Key (Related Order) -> TK Company - Key (Related
Order)
           IC Related PO/SO/WO Number -> TK Related PO/SO/WO Number
           IC Related PO/SO/WO Order Type -> TK Related PO/SO/WO Order
Type
           IC Related PO/SO Line Number -> TK Related PO/SO Line Number
           IC Agreement Number - Distribution -> TK Agreement Number -
Distribution
           IC Agreement Supplement - Distribution -> TK Agreement
Supplement - Distribution
           IC Contract Balances Updated Y/N -> TK Contract Balances
Updated Y/N
           IC Address Number -> TK Address Number
           IC Address Number - Ship To -> TK Address Number - Ship To
           IC Address Number - Parent -> TK Address Number - Parent
           IC Date - Requested -> TK Date - Requested
           IC Date - Order/Transaction -> TK Date - Order/Transaction
           IC Date - Scheduled Pick -> TK Date - Scheduled Pick
           IC Date - Original Promised Delivery -> TK Date - Original
Promised Delivery
           IC Date - Actual Ship Date -> TK Date - Actual Ship Date
           IC Date - Invoice -> TK Date - Invoice
           IC Date - Cancel -> TK Date - Cancel
           IC Date - For G/L (and Voucher) -> TK Date - For G/L (and
Voucher)
           IC Date - Promised Delivery -> TK Date - Promised Delivery
           IC Date - Price Effective Date -> TK Date - Price Effective
Date
           IC Date - Promised Shipment -> TK Date - Promised Shipment
           IC Date - Future Date 2 -> TK Date - Future Date 2
           IC Reference -> TK Reference
           IC Reference 2 -> TK Reference 2
           IC Item Number - Short -> TK Item Number - Short
           IC 2nd Item Number -> TK 2nd Item Number
           IC 3rd Item Number -> TK 3rd Item Number
           IC Location -> TK Location
           IC Lot/Serial Number -> TK Lot/Serial Number
           IC From Grade -> TK From Grade
           IC Thru Grade -> TK Thru Grade
```

```
            IC From Potency -> TK From Potency
            IC Thru Potency -> TK Thru Potency
            IC Days Before Expiration -> TK Days Before Expiration
            IC Description -> TK Description
            IC Description - Line 2 -> TK Description - Line 2
            IC Line Type -> TK Line Type
            IC Status Code - Next -> TK Status Code - Next
            IC Status Code - Last -> TK Status Code - Last
            IC Business Unit - Header -> TK Business Unit - Header
            IC Item Number - Related (Kit) -> TK Item Number - Related
(Kit)
            IC Kit Master Line Number -> TK Kit Master Line Number
            IC Component Line Number -> TK Component Line Number
            IC Related Kit Component -> TK Related Kit Component
            IC Number of Component Per Parent -> TK Number of Component
Per Parent
            IC Sales Catalog Section -> TK Sales Catalog Section
            IC Sub Section -> TK Sub Section
            IC Sales Category Code 3 -> TK Sales Category Code 3
            IC Sales Category Code 4 -> TK Sales Category Code 4
            IC Sales Category Code 5 -> TK Sales Category Code 5
            IC Commodity Class -> TK Commodity Class
            IC Commodity Sub Class -> TK Commodity Sub Class
            IC Supplier Rebate Code -> TK Supplier Rebate Code
            IC Master Planning Family -> TK Master Planning Family
            IC Purchasing Category Code 5 -> TK Purchasing Category Code
5
            IC Unit of Measure as Input -> TK Unit of Measure as Input
            IC Units - Order/Transaction Quantity -> TK Units -
Order/Transaction Quantity
            IC Quantity Shipped -> TK Quantity Shipped
            IC Units - Qty Backordered/Held -> TK Units - Qty
Backordered/Held
            IC Units - Quantity Canceled/Scrapped -> TK Units - Quantity
Canceled/Scrapped
            IC Units - Future Quantity Committed -> TK Units - Future
Quantity Committed
            IC Units - Open -> TK Units - Open
            IC Units - Shipped to Date -> TK Units - Shipped to Date
            IC Units - Relieved -> TK Units - Relieved
            IC Committed (H/S) -> TK Committed (H/S)
            IC Other Quantity (1/2) -> TK Other Quantity (1/2)
            IC Amount - Price per Unit -> TK Amount - Price per Unit
            IC Amount - Extended Price -> TK Amount - Extended Price
            IC Amount - Open -> TK Amount - Open
            IC Price Override Code -> TK Price Override Code
            IC Temporary Price (Y/N) -> TK Temporary Price (Y/N)
            IC Unit of Measure - Entered for Unit Price -> TK Unit of
Measure - Entered for Unit Price
            IC Amount - List Price -> TK Amount - List Price
            IC Amount - Unit Cost -> TK Amount - Unit Cost
            IC Amount - Extended Cost -> TK Amount - Extended Cost
            IC Cost Override Code -> TK Cost Override Code
            IC Extended Cost - Transfer -> TK Extended Cost - Transfer
            IC Print Message -> TK Print Message
            IC Payment Terms Code -> TK Payment Terms Code
            IC Payment Instrument -> TK Payment Instrument
            IC Based on Date -> TK Based on Date
            IC Discount - Trade -> TK Discount - Trade
            IC Trade Discount (Old) -> TK Trade Discount (Old)
            IC Price and Adjustment Schedule -> TK Price and Adjustment
Schedule
```

```
            IC Item Price Group -> TK Item Price Group
            IC Pricing Category Level -> TK Pricing Category Level
            IC Discount Factor -> TK Discount Factor
            IC Discount Factor Type - $ or % (D/P) -> TK Discount Factor
Type - $ or % (D/P)
            IC Discount Application Type -> TK Discount Application Type
            IC Discount % - Cash -> TK Discount % - Cash
            IC Document Company -> TK Document Company
            IC Document (Voucher, Invoice, etc.) -> TK Document (Voucher,
Invoice, etc.)
            IC Document Type -> TK Document Type
            IC Document - Original -> TK Document - Original
            IC Document Type - Original -> TK Document Type - Original
            IC Document Company - Original -> TK Document Company -
Original
            IC Pick Slip Number -> TK Pick Slip Number
            IC Delivery Number -> TK Delivery Number
            IC Number - Promotion Number -> TK Number - Promotion Number
            IC Draft Number -> TK Draft Number
            IC Sales Taxable (Y/N) -> TK Sales Taxable (Y/N)
            IC Tax Rate/Area -> TK Tax Rate/Area
            IC Tax Expl Code 1 -> TK Tax Expl Code 1
            IC Associated Text -> TK Associated Text
            IC Priority - Processing -> TK Priority - Processing
            IC Printed Code -> TK Printed Code
            IC Backorders Allowed (Y/N) -> TK Backorders Allowed (Y/N)
            IC Substitutes Allowed (Y/N) -> TK Substitutes Allowed (Y/N)
            IC Partial Line Shipments Allowed (Y/N) -> TK Partial Line
Shipments Allowed (Y/N)
            IC Line of Business -> TK Line of Business
            IC End Use -> TK End Use
            IC Duty Status -> TK Duty Status
            IC Commodity Code -> TK Commodity Code
            IC Nature of Transaction -> TK Nature of Transaction
            IC Primary / Last Supplier Number -> TK Primary / Last
Supplier Number
            IC Buyer Number -> TK Buyer Number
            IC Carrier Number -> TK Carrier Number
            IC Mode of Transport -> TK Mode of Transport
            IC Conditions of Transport -> TK Conditions of Transport
            IC Route Code -> TK Route Code
            IC Stop Code -> TK Stop Code
            IC Zone Number -> TK Zone Number
            IC Container I.D. -> TK Container I.D.
            IC Freight Handling Code -> TK Freight Handling Code
            IC Apply Freight - Y/N -> TK Apply Freight - Y/N
            IC AIA Document Flag -> TK AIA Document Flag
            IC Freight Calculated (Y/N) -> TK Freight Calculated (Y/N)
            IC Rate Code - Frieght/Misc -> TK Rate Code - Frieght/Misc
            IC Rate Type - Freight/Misc -> TK Rate Type - Freight/Misc
            IC Shipping Commodity Class -> TK Shipping Commodity Class
            IC Shipping Conditions Code -> TK Shipping Conditions Code
            IC Serial Number - Lot -> TK Serial Number - Lot
            IC Unit of Measure - Primary -> TK Unit of Measure - Primary
            IC Units - Primary Quantity Ordered -> TK Units - Primary
Quantity Ordered
```

```
            IC Unit of Measure - Secondary -> TK Unit of Measure -
Secondary
            IC Units - Secondary Quantity Ordered -> TK Units - Secondary
Quantity Ordered
            IC Unit of Measure - Pricing -> TK Unit of Measure - Pricing
            IC Unit Weight -> TK Unit Weight
            IC Weight Unit of Measure -> TK Weight Unit of Measure
            IC Unit Volume -> TK Unit Volume
            IC Volume Unit of Measure -> TK Volume Unit of Measure
            IC Reprice (Basket Price) Category -> TK Reprice (Basket
Price) Category
            IC Order Reprice Category -> TK Order Reprice Category
            IC Order Repriced Indicator -> TK Order Repriced Indicator
            IC Costing Method - Inventory -> TK Costing Method -
Inventory
            IC Commitment Method -> TK Commitment Method
            IC G/L Offset -> TK G/L Offset
            IC Century -> TK Century
            IC Fiscal Year -> TK Fiscal Year
            IC Line Status -> TK Line Status
            IC Inter Branch Sales -> TK Inter Branch Sales
            IC On Hand Updated -> TK On Hand Updated
            IC Configurator Print Flag -> TK Configurator Print Flag
            IC Sales Order Status 04 -> TK Sales Order Status 04
            IC Substitute Item Indicator -> TK Substitute Item Indicator
            IC Preference Commitment Indicator -> TK Preference
Commitment Indicator
            IC Ship date (PDDJ) overridden -> TK Ship date (PDDJ)
overridden
            IC Price Adjustment Line Indicator -> TK Price Adjustment
Line Indicator
            IC Price Adj. History Indicator -> TK Price Adj. History
Indicator
            IC Preference Production Allocation -> TK Preference
Production Allocation
            IC Transfer/Direct Ship/Intercompany Flag -> TK
Transfer/Direct Ship/Intercompany Flag
            IC Deferred entries flag -> TK Deferred entries flag
            IC Euro Conversion Status Flag -> TK Euro Conversion Status
Flag
            IC Sales Order Status 14 -> TK Sales Order Status 14
            IC Sales Order Status 15 -> TK Sales Order Status 15
            IC Salesperson 01 -> TK Salesperson 01
            IC Salesperson Commission 001 -> TK Salesperson Commission
001
            IC Salesperson 02 -> TK Salesperson 02
            IC Salesperson Commission 002 -> TK Salesperson Commission
002
            IC Apply Commission (Y/N) -> TK Apply Commission (Y/N)
            IC Commission Category -> TK Commission Category
            IC Reason Code -> TK Reason Code
            IC Gross Weight -> TK Gross Weight
            IC Gross Weight Unit of Measure -> TK Gross Weight Unit of
Measure
            IC Account Number - Input (Mode Unknown) -> TK Account Number
- Input (Mode Unknown)
            IC Account ID -> TK Account ID
```

```
            IC Project Business Unit -> TK Project Business Unit
            IC Object Account -> TK Object Account
            IC Subsidiary -> TK Subsidiary
            IC Ledger Type -> TK Ledger Type
            IC Subledger - G/L -> TK Subledger - G/L
            IC Subledger Type -> TK Subledger Type
            IC Code - Location Tax Status -> TK Code - Location Tax
Status
            IC Price Code 1 -> TK Price Code 1
            IC Price Code 2 -> TK Price Code 2
            IC Price Code 3 -> TK Price Code 3
            IC Status - In Warehouse -> TK Status - In Warehouse
            IC Work Order Freeze Code -> TK Work Order Freeze Code
            IC Send Method -> TK Send Method
            IC Currency Code - From -> TK Currency Code - From
            IC Currency Conversion Rate - Spot Rate -> TK Currency
Conversion Rate - Spot Rate
            IC Amount - List Price per Unit -> TK Amount - List Price per
Unit
            IC Amount - Foreign Price per Unit -> TK Amount - Foreign
Price per Unit
            IC Amount - Foreign Extended Price -> TK Amount - Foreign
Extended Price
            IC Amount - Foreign Unit Cost -> TK Amount - Foreign Unit
Cost
            IC Amount - Foreign Extended Cost -> TK Amount - Foreign
Extended Cost
            IC User Reserved Code -> TK User Reserved Code
            IC User Reserved Date -> TK User Reserved Date
            IC User Reserved Amount -> TK User Reserved Amount
            IC User Reserved Number -> TK User Reserved Number
            IC User Reserved Reference -> TK User Reserved Reference
            IC Transaction Originator -> TK Transaction Originator
            IC User ID -> TK User ID
            IC Program ID -> TK Program ID
            IC Work Station ID -> TK Work Station ID
            IC Date - Updated -> TK Date - Updated
            IC Time of Day -> TK Time of Day
0006 //
0007 // Do not delete the record if the insert to the archive table
failed.
0008 //
0009 If SV Error_Status is not equal to CO ERROR
0010  Delete Current Input Row
0011 End If
0012 Else
0013  Delete Current Input Row
0014 End If
```

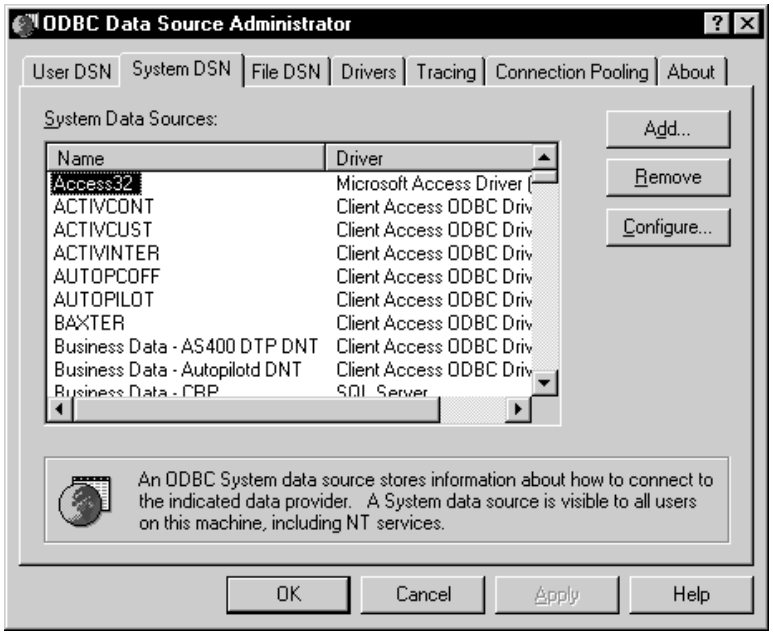
On Selection for Batch Delete, choose Process End and click Advanced ER. On Event Rules Design, enter the process end event rules, along with any special logic. Ensure that you have mapped all parameters to a field, even if you will not use every value. For this example, use event rules for R42119P, as follows:

```
0001 If PO cArchiveRecords is equal to "1"
0002    //
0003    // Close the table
0004    //
0005    F42119(VA rpt_F42119Handle_HF42119).Close
0006    //
0007    // If the data was archived and the table name processing option
was populated,
0008    // rename the table.
0009    //
0010    If PO szArchiveTableName is not equal to <Blank>
        And PO szArchiveTableName is not equal to <Null>
0011       Rename Table   (B0000202)
              "F42119" -> szOldTableName
              PO szArchiveTableName -> szNewTableName
              "<Blank>" -> szTableOwnerID
              "<Blank>" -> szPassword
              VA rpt_szArchiveDataSource_DATS -> szDataSource
              VA rpt_cRenameFlag_EV01 <- cRenameTableSuccessful
0012    End If
0013 End If
```

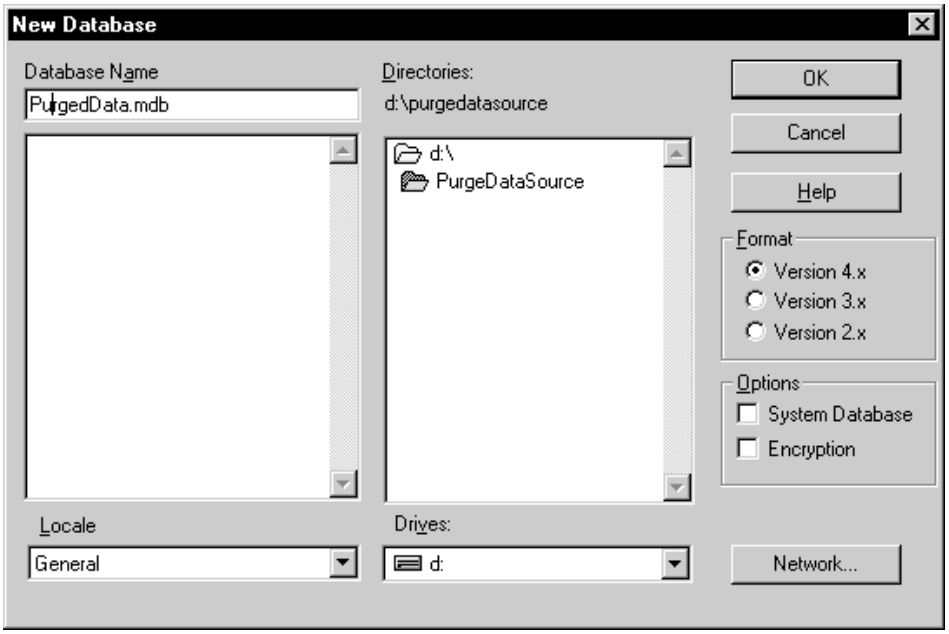
Test the purge program. You must have an archive environment that has a different data source than the source environment. This example explains how to test locally and uses the ODBC Data Source Administrator application to set up the test.

Create a new folder on the D drive.

Using the ODBC Data Source Administrator application, on the System DSN tab, create an ODBC data source. For this example, use Access 32.

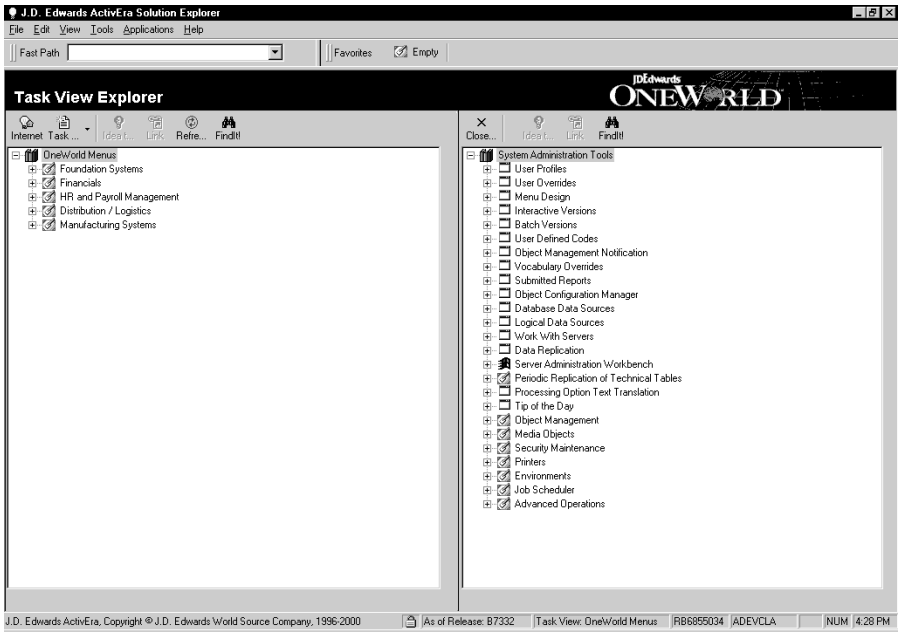


On New Database, create a new database in the new folder you already created. You must associate this database with the ODBC data source.

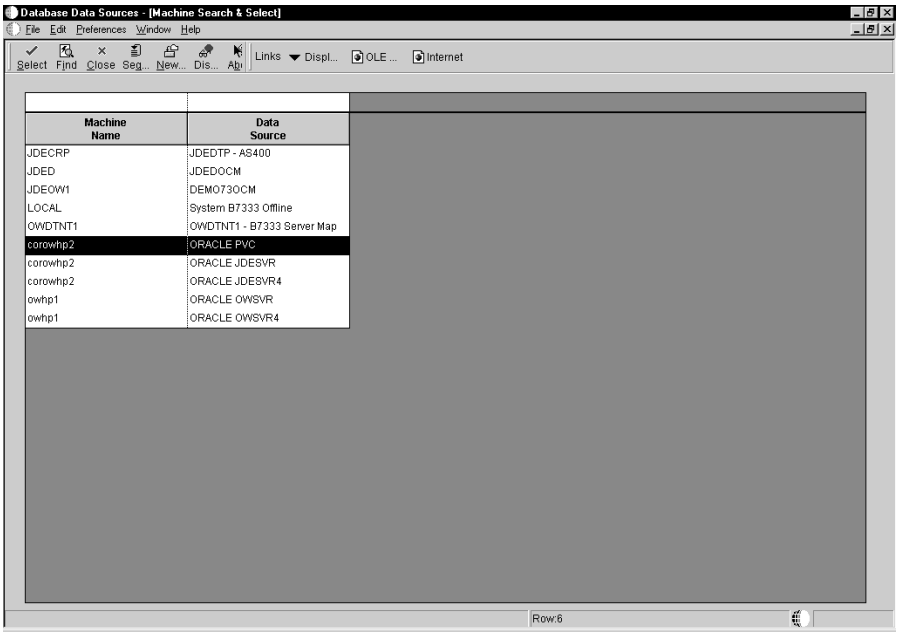


Log into OneWorld. For this example, choose ADEVCLA as the environment.

On GH9011, select Database Data Sources to create a OneWorld data source.

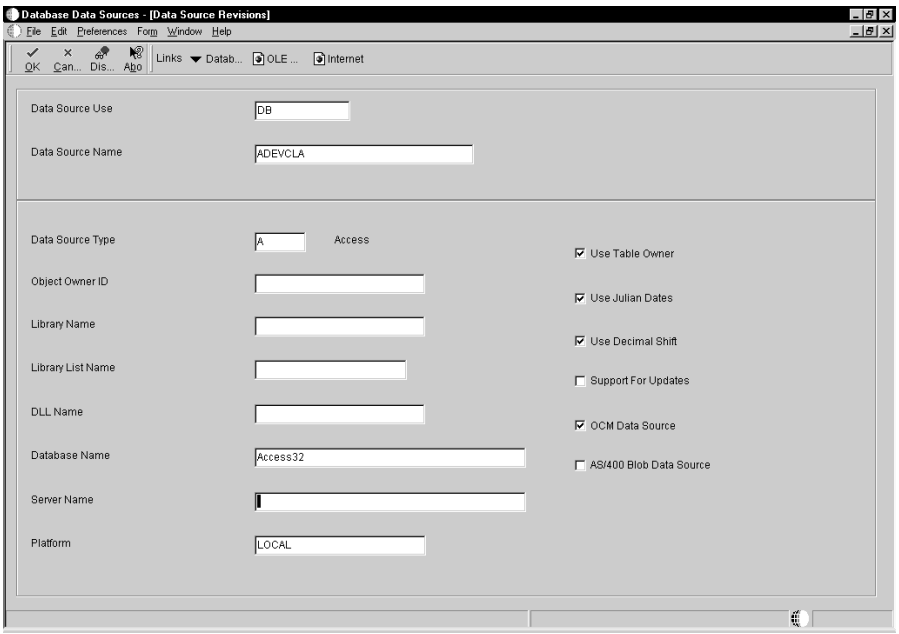


On Machine Search and Select, choose the machine name you are currently using. For this example, use corowhip2 ORACLE PVC.



On Data Source Revisions, add a data source. For this example, use the following information:

- Data Source Name–Same name as ODBC name.
- Database Name–ODBC data source name.
- Library List Name–Select the environment you want to use as your testing environment. Make sure the environment you select has the same path code as your local environment. For this example, use ADEVCLA for the source environment, and ADEVASD2 for the archive environment.



Create a new Object Configuration Manager (OCM) mapping to re–direct the table you want to purge from ADEVASD2 to the new OneWorld data source you created. Make the mapping object status Active (AV).

Glossary

Glossary

AAI. See automatic accounting instruction.

action message. With OneWorld, users can receive messages (system-generated or user-generated) that have shortcuts to OneWorld forms, applications, and appropriate data. For example, if the general ledger post sends an action error message to a user, that user can access the journal entry (or entries) in error directly from the message. This is a central feature of the OneWorld workflow strategy. Action messages can originate either from OneWorld or from a third-party e-mail system.

activator. In the Solution Explorer, a parent task with sequentially-arranged child tasks that are automated with a director.

ActiveX. A computing technology, based on object linking and embedding, that enables Java applet-style functionality for Web browsers as well as other applications. (Java is limited to Web browsers at this time.) The ActiveX equivalent of a Java applet is an ActiveX control. These controls bring computational, communications, and data manipulation power to programs that can "contain" them. For example, certain Web browsers, Microsoft Office programs, and anything developed with Visual Basic or Visual C++.

advance. A change in the status of a project in the Object Management Workbench. When you advance a project, the status change might trigger other actions and conditions such as moving objects from one server to another or preventing check-out of project objects.

alphanumeric character. A combination of letters, numbers, and symbols used to represent data. Contrast with numeric character and special character.

API. See application programming interface.

APPL. See application.

applet. A small application, such as a utility program or a limited-function spreadsheet. It is generally associated with the programming language Java, and in this context refers to Internet-enabled applications that can be passed from a Web browser residing on a workstation.

application. In the computer industry, the same as an executable file. In OneWorld, an interactive or batch application is a DLL that contains programming for a set of related forms that can be run from a menu to perform a business task such as Accounts Payable and Sales Order Processing. Also known as system.

application developer. A programmer who develops OneWorld applications using the OneWorld toolset.

application programming interface (API). A software function call that can be made from a program to access functionality provided by another program.

application workspace. The area on a workstation display in which all related forms within an application appear.

audit trail. The detailed, verifiable history of a processed transaction. The history consists of the original documents, transaction entries, and posting of records, and usually concludes with a report.

automatic accounting instruction (AAI). A code that refers to an account in the chart of accounts. AAIs define rules for programs that automatically generate journal entries, including interfaces between Accounts Payable, Accounts Receivable, Financial Reporting, General Accounting systems. Each system that interfaces with the General Accounting system has AAIs. For example, AAIs can direct the General Ledger Post program to post a debit to a specific expense account and a credit to a specific accounts payable account.

batch header. The information that identifies and controls a batch of transactions or records.

batch job. A task or group of tasks you submit for processing that the system treats as a single unit during processing, for example, printing reports and purging files. The computer system

performs a batch job with little or no user interaction.

batch processing. A method by which the system selects jobs from the job queue, processes them, and sends output to the outqueue. Contrast with interactive processing.

batch server. A server on which OneWorld batch processing requests (also called UBEs) are run instead of on a client, an application server, or an enterprise server. A batch server typically does not contain a database nor does it run interactive applications.

batch type. A code assigned to a batch job that designates to which J.D. Edwards system the associated transactions pertain, thus controlling which records are selected for processing. For example, the Post General Journal program selects for posting only unposted transaction batches with a batch type of O.

batch-of-one immediate. A transaction method that allows a client application to perform work on a client workstation, then submit the work all at once to a server applicationfor further processing. As a batch process is running on the server, the client application can continue performing other tasks. See also direct connect, store and forward.

BDA. See Business View Design Aid.

binary string (BSTR). A length prefixed string used by OLE automation data manipulation functions. Binary Strings are wide, double-byte (Unicode) strings on 32-bit Windows platforms.

Boolean Logic Operand. In J.D. Edwards reporting programs, the parameter of the Relationship field. The Boolean logic operand instructs the system to compare certain records or parameters. Available options are:

EQ	Equal To.
LT	Less Than.
LE	Less Than or Equal To.
GT	Greater Than.
GE	Greater Than or Equal To.
NE	Not Equal To.
NL	Not Less Than.
NG	Not Greater Than.

browser. A client application that translates information sent by the World Wide Web. A client must use a browser to receive, manipulate, and display World Wide Web information on the desktop. Also known as a Web browser.

BSFN. See business function.

BSTR. See binary string.

BSVW. See business view.

business function. An encapsulated set of business rules and logic that can normally be reused by multiple applications. Business functions can execute a transaction or a subset of a transaction (check inventory, issue work orders, and so on). Business functions also contain the APIs that allow them to be called from a form, a database trigger, or a non-OneWorld application. Business functions can be combined with other business functions, forms, event rules, and other components to make up an application. Business functions can be created through event rules or third-generation languages, such as C. Examples of business functions include Credit Check and Item Availability.

business function event rule. See named event rule.

business view. Used by OneWorld applications to access data from database tables. A business view is a means for selecting specific columns from one or more tables whose data will be used in an application or report. It does not select specific rows and does not contain any physical data. It is strictly a view through which data can be handled.

Business View Design Aid (BDA). A OneWorld GUI ool for creating, modifying, copying, and printing business views. The tool uses a graphical user interface.

category code. In user defined codes, a temporary title for an undefined category. For example, if you are adding a code that designates different sales regions, you could change category code 4 to Sales Region, and define E (East), W (West), N (North), and S (South) as the valid codes. Sometimes referred to as reporting codes.

central objects. Objects that reside in a central location and consist of two parts: the central objects data source and central C components. The central objects data source contains OneWorld specifications, which are stored in a relational database. Central C components

contain business function source, header, object, library, and DLL files and are usually stored in directories on the deployment server. Together they make up central objects.

check-in location. The directory structure location for the package and its set of replicated objects. This is usually \\deploymentserver\release\path_code\package\ packagename. The sub-directories under this path are where the central C components (source, include, object, library, and DLL file) for business functions are stored.

child. See parent/child form.

client/server. A relationship between processes running on separate machines. The server process is a provider of software services. The client is a consumer of those services. In essence, client/server provides a clean separation of function based on the idea of service. A server can service many clients at the same time and regulate their access to shared resources. There is a many-to-one relationship between clients and a server, respectively. Clients always initiate the dialog by requesting a service. Servers passively wait for requests from clients.

CNC. See configurable network computing.

component. In the ActivEra Portal, an encapsulated object that appears inside a workspace. Portal components

configurable client engine. Allows user flexibility at the interface level. Users can easily move columns, set tabs for different data views, and size grids according to their needs. The configurable client engine also enables the incorporation of Web browsers in addition to the Windows 95- and Windows NT-based interfaces.

configurable network computing. An application architecture that allows interactive and batch applications, composed of a single code base, to run across a TCP/IP network of multiple server platforms and SQL databases. The applications consist of reusable business functions and associated data that can be configured across the network dynamically. The overall objective for businesses is to provide a future-proof environment that enables them to change organizational structures, business processes, and technologies independently of each other.

constants. Parameters or codes that you set and the system uses to standardize information processing by associated programs. Some examples of constants are: validating bills of material online and including fixed labor overhead in costing.

control. Any data entry point allowing the user to interact with an application. For example, check boxes, pull-down lists, hyper-buttons, entry fields, and similar features are controls.

core. The central and foundation systems of J.D. Edwards software, including General Accounting, Accounts Payable, Accounts Receivable, Address Book, Financial Reporting, Financial Modeling and Allocations, and Back Office.

CRP. Conference Room Pilot.

custom gridlines. A grid row that does not come from the database, for example, totals. To display a total in a grid, sum the values and insert a custom gridline to display the total. Use the system function Insert Grid Row Buffer to accomplish this.

data dictionary. The OneWorld method for storing and managing data item definitions and specifications. J.D. Edwards has an active data dictionary, which means it is accessed at runtime.

data mart. Department-level decision support databases. They usually draw their data from an enterprise data warehouse that serves as a source of consolidated and reconciled data from around the organization. Data marts can be either relational or multidimensional databases.

data replication. In a replicated environment, multiple copies of data are maintained on multiple machines. There must be a single source that "owns" the data. This ensures that the latest copy of data can be applied to a primary place and then replicated as appropriate. This is in contrast to a simple copying of data, where the copy is not maintained from a central location, but exists independently of the source.

data source. A specific instance of a database management system running on a computer.

Data source management is accomplished through Object Configuration Manager (OCM) and Object Map (OM).

data structure. A group of data items that can be used for passing information between objects, for example, between two forms, between forms and business functions, or between reports and business functions.

data warehouse. A database used for reconciling and consolidating data from multiple databases before it is distributed to data marts for department-level decision support queries and reports. The data warehouse is generally a large relational database residing on a dedicated server between operational databases and the data marts.

data warehousing. Essentially, data warehousing involves off-loading operational data sources to target databases that will be used exclusively for decision support (reports and queries). There are a range of decision support environments, including duplicated database, enhanced analysis databases, and enterprise data warehouses.

database. A continuously updated collection of all information a system uses and stores. Databases make it possible to create, store, index, and cross-reference information online.

database driver. Software that connects an application to a specific database management system.

database server. A server that stores data. A database server does not have OneWorld logic.

DCE. See distributed computing environment.

DD. See data dictionary.

default. A code, number, or parameter value that is assumed when none is specified.

detail. The specific pieces of information and data that make up a record or transaction. Contrast with summary.

detail area. A control that is found in OneWorld applications and functions similarly to a spreadsheet grid for viewing, adding, or updating many rows of data at one time.

direct connect. A transaction method in which a client application communicates interactively and directly with a server application. See also batch-of-one immediate, store and forward.

director. An interactive utility that guides a user through the steps of a process to complete a task.

distributed computing environment (DCE). A set of integrated software services that allows software running on multiple computers to perform in a manner that is seamless and transparent to the end-users. DCE provides security, directory, time, remote procedure calls, and files across computers running on a network.

DLL. See dynamic link library.

DS. See data structure.

DSTR. See data structure.

duplicated database. A decision support database that contains a straightforward copy of operational data. The advantages involve improved performance for both operational and reporting environments. See also enhanced analysis database, enterprise data warehouse.

dynamic link library (DLL). A set of program modules that are designed to be invoked from executable files when the executable files are run, without having to be linked to the executable files. They typically contain commonly used functions.

dynamic partitioning. The ability to dynamically distribute logic or data to multiple tiers in a client/server architecture.

embedded event rule. An event rule that is specific to a particular table or application. Examples include form-to-form calls, hiding a field based on a processing option value, and calling a business function. Contrast with business function event rule. See also event rule.

employee work center. This is a central location for sending and receiving all OneWorld messages (system and user generated) regardless of the originating application or user. Each user has a mailbox that contains workflow and other messages, including Active Messages. With respect to workflow, the Message Center is MAPI compliant and supports drag and drop work reassignment, escalation, forward and reply, and workflow monitoring. All messages

from the message center can be viewed through OneWorld messages or Microsoft Exchange.

encapsulation. The ability to confine access to and manipulation of data within an object to the procedures that contribute to the definition of that object.

enhanced analysis database. A database containing a subset of operational data. The data on the enhanced analysis database performs calculations and provides summary data to speed generation of reports and query response times. This solution is appropriate when external data must be added to source data, or when historical data is necessary for trend analysis or regulatory reporting. See also duplicated database, enterprise data warehouse.

enterprise data warehouse. A complex solution that involves data from many areas of the enterprise. This environment requires a large relational database (the data warehouse) that is a central repository of enterprise data, which is clean, reconciled, and consolidated. From this repository, data marts retrieve data to provide department-level decisions. See also duplicated database, enhanced analysis database.

enterprise server. A database server and logic server. See database server. Also referred to as host.

ER. See event rule.

ERP. See enterprise resource planning.

event. An action that occurs when an interactive or batch application is running. Example events are tabbing out of an edit control, clicking a push button, initializing a form, or performing a page break on a report. The GUI operating system uses miniprograms to manage user activities within a form. Additional logic can be attached to these miniprograms and used to give greater functionality to any event within a OneWorld application or report using event rules.

event rule. Used to create complex business logic without the difficult syntax that comes with many programming languages. These logic statements can be attached to applications or database events and are executed when the defined event occurs, such as entering a form, selecting a menu bar option, page breaking on a report, or selecting a record. An event rule can validate data, send a message to a user, call a business function, as well as many other actions. There are two types of event rules:

1 Embedded event rules.
2 Named event rules.

executable file. A computer program that can be run from the computer's operating system. Equivalent terms are "application" and "program.".

exit. 1) To interrupt or leave a computer program by pressing a specific key or a sequence of keys. 2) An option or function key displayed on a form that allows you to access another form.

facility. 1) A separate entity within a business for which you want to track costs. For example, a facility might be a warehouse location, job, project, work center, or branch/plant. Sometimes referred to as a business unit. 2) In Home Builder and ECS, a facility is a collection of computer language statements or programs that provide a specialized function throughout a system or throughout all integrated systems. For example, DREAM Writer and FASTR are facilities.

FDA. See Form Design Aid.

find/browse. A type of form used to:

1 Search, view, and select multiple records in a detail area.
2 Delete records.
3 Exit to another form.
4 Serve as an entry point for most applications.

firewall. A set of technologies that allows an enterprise to test, filter, and route all incoming messages. Firewalls are used to keep an enterprise secure.

fix/inspect. A type of form used to view, add, or modify existing records. A fix/inspect form has no detail area.

form. An element of OneWorld's graphical user interface that contains controls by which a user can interact with an application. Forms allow the user to input, select, and view information. A OneWorld application might contain multiple forms. In Microsoft Windows terminology, a form is known as a dialog box.

Form Design Aid (FDA). The OneWorld GUI development tool for building interactive applications and forms.

form interconnection. Allows one form to access and pass data to another form. Form interconnections can be attached to any event; however, they are normally used when a button is clicked.

form type. The following form types are available in OneWorld:

1 Find/browse.
2 Fix/inspect.
3 Header detail.
4 Headerless detail.
5 Message.
6 Parent/child.
7 Search/select.

fourth generation language (4GL). A programming language that focuses on what you need to do and then determines how to do it. Structured Query Language is an example of a 4GL.

graphical user interface (GUI). A computer interface that is graphically based as opposed to being character-based. An example of a character-based interface is that of the AS/400. An example of a GUI is Microsoft Windows. Graphically based interfaces allow pictures and other graphic images to be used in order to give people clues on how to operate the computer.

grid. See detail area.

GUI. See graphical user interface.

header. Information at the beginning of a table or form. This information is used to identify or provide control information for the group of records that follows.

header/detail. A type of form used to add, modify, or delete records from two different tables. The tables usually have a parent/child relationship.

headerless detail. A type of form used to work with multiple records in a detail area. The detail area is capable of of receiving input.

hidden selections. Menu selections you cannot see until you enter HS in a menu's Selection field. Although you cannot see these selections, they are available from any menu. They include such items as Display Submitted Jobs (33), Display User Job Queue (42), and Display User Print Queue (43). The Hidden Selections window displays three categories of selections: user tools, operator tools, and programmer tools.

host. In the centralized computer model, a large timesharing computer system that terminals communicate with and rely on for processing. In contrasts with client/server in that those users work at computers that perform much of their own processing and access servers that provide services such as file management, security, and printer management.

HTML. See hypertext markup language.

hypertext markup language. A markup language used to specify the logical structure of a document rather than the physical layout. Specifying logical structure makes any HTML document platform independent. You can view an HTML document on any desktop capable of supporting a browser. HTML can include active links to other HTML documents anywhere on the Internet or on intranet sites.

index. Represents both an ordering of values and a uniqueness of values that provide efficient access to data in rows of a table. An index is made up of one or more columns in the table.

inheritance. The ability of a class to recieve all or parts of the data and procedure definitions from a parent class. Inheritance enhances developement through the reuse of classes and their related code.

install system code. See system code.

integrated toolset. Unique to OneWorld is an industrial-strength toolset embedded in the already comprehensive business applications. This toolset is the same toolset used by J.D. Edwards to build OneWorld interactive and batch applications. Much more than a development environment, however, the OneWorld integrated toolset handles reporting and other batch processes, change management, and basic data warehousing facilities.

interactive processing. Processing actions that occur in response to commands you enter directly into the system. During interactive processing, you are in direct communication with the system, and it might prompt you for additional information while processing your

request. See also online. Contrast with batch processing.

interface. A link between two or more computer systems that allows these systems to send information to and receive information from one another.

Internet. The worldwide constellation of servers, applications, and information available to a desktop client through a phone line or other type of remote access.

interoperability. The ability of different computer systems, networks, operating systems, and applications to work together and share information.

intranet. A small version of the Internet usually confined to one company or organization. An intranet uses the functionality of the Internet and places it at the disposal of a single enterprise.

IP. A connection-less communication protocol that by itself provides a datagram service. Datagrams are self-contained packets of information that are forwarded by routers based on their address and the routing table information contained in the routers. Every node on a TCP/IP network requires an address that identifies both a network and a local host or node on the network. In most cases the network administrator sets up these addresses when installing new workstations. In some cases, however, it is possible for a workstation, when booting up, to query a server for a dynamically assigned address.

IServer Service. Developed by J.D. Edwards, this internet server service resides on the web server, and is used to speed up delivery of the Java class files from the database to the client.

ISO 9000. A series of standards established by the International Organization for Standardization, designed as a measure of product and service quality.

J.D. Edwards Database. See JDEBASE Database Middleware.

Java. An Internet executable language that, like C, is designed to be highly portable across platforms. This programming language was developed by Sun Microsystems. Applets, or Java applications, can be accessed from a web browser and executed at the client, provided that the operating system or browser is Java-enabled. (Java is often described as a scaled-down C++). Java applications are platform independent.

Java Database Connectivity (JDBC). The standard way to access Java databases, set by Sun Microsystems. This standard allows you to use any JDBC driver database.

JavaScript. A scripting language related to Java. Unlike Java, however, JavaScript is not an object-oriented language and it is not compiled.

jde.ini. J.D. Edwards file (or member for AS/400) that provides the runtime settings required for OneWorld initialization. Specific versions of the file/member must reside on every machine running OneWorld. This includes workstations and servers.

JDEBASE Database Middleware. J.D. Edwards proprietary database middleware package that provides two primary benefits:

1. Platform-independent APIs for multidatabase access. These APIs are used in two ways:
   a. By the interactive and batch engines to dynamically generate platform-specific SQL, depending on the datasource request.
   b. As open APIs for advanced C business function writing. These APIs are then used by the engines to dynamically generate platform-specific SQL.
2. Client-to-server and server-to-server database access. To accomplish this OneWorld is integrated with a variety of third-party database drivers, such as Client Access 400 and open database connectivity (ODBC).

JDECallObject. An application programming interface used by business functions to invoke other business functions.

JDENET. J.D. Edwards proprietary middleware software. JDENET is a messaging software package.

JDENET communications middleware. J.D. Edwards proprietary communications middleware package for OneWorld. It is a peer-to-peer, message-based, socket based, multiprocess communications middleware solution. It handles client-to-server and

server-to-server communications for all OneWorld supported platforms.

job queue. A group of jobs waiting to be batch processed. See also batch processing.

just in time installation (JITI). OneWorld's method of dynamically replicating objects from the central object location to a workstation.

just in time replication (JITR). OneWorld's method of replicating data to individual workstations. OneWorld replicates new records (inserts) only at the time the user needs the data. Changes, deletes, and updates must be replicated using Pull Replication.

KEY. A column or combination of columns that identify one or more records in a database table.

leading zeros. A series of zeros that certain facilities in J.D. Edwards systems place in front of a value you enter. This normally occurs when you enter a value that is smaller than the specified length of the field. For example, if you enter 4567 in a field that accommodates eight numbers, the facility places four zeros in front of the four numbers you enter. The result appears as: 00004567.

level of detail. 1) The degree of difficulty of a menu in J.D. Edwards software. The levels of detail for menus are as follows:

A	Major Product Directories.
B	Product Groups.
1	Basic Operations.
2	Intermediate Operations.
3	Advanced Operations.
4	Computer Operations.
5	Programmers.
6	Advanced Programmers Also known as menu levels.

2) The degree to which account information in the General Accounting system is summarized. The highest level of detail is 1 (least detailed) and the lowest level of detail is 9 (most detailed).

MAPI. See Messaging Application Programming Interface.

master table. A database table used to store data and information that is permanent and necessary to the system's operation. Master tables might contain data such as paid tax amounts, supplier names, addresses, employee information, and job information.

menu. A menu that displays numbered selections. Each of these selections represents a program or another menu. To access a selection from a menu, type the selection number and then press Enter.

menu levels. See level of detail.

menu masking. A security feature of J.D. Edwards systems that lets you prevent individual users from accessing specified menus or menu selections. The system does not display the menus or menu selections to unauthorized users.

Messaging Application Programming Interface (MAPI). An architecture that defines the components of a messaging system and how they behave. It also defines the interface between the messaging system and the components.

middleware. A general term that covers all the distributed software needed to support interactions between clients and servers. Think of it as the software that's in the middle of the client/server system or the "glue" that lets the client obtain a service from a server.

modal. A restrictive or limiting interaction created by a given condition of operation. Modal often describes a secondary window that restricts a user's interaction with other windows. A secondary window can be modal with respect to it's primary window or to the entire system. A modal dialog box must be closed by the user before the application continues.

mode. In reference to forms in OneWorld, mode has two meanings:

- An operational qualifier that governs how the form interacts with tables and business views. OneWorld form modes are: add, copy, and update.
- An arbitrary setting that aids in organizing form generation for different environments. For example, you might set forms generated for a Windows environment to mode 1 and forms generated for a Web environment to mode 2.

modeless. Not restricting or limiting interaction. Modeless often describes a secondary window that does not restrict a user's interaction with

other windows. A modeless dialog box stays on the screen and is available for use at any time but also permits other user activities.

multitier architecture. A client/server architecture that allows multiple levels of processing. A tier defines the number of computers that can be used to complete some defined task.

named event rule. Encapsulated, reusable business logic created using through event rules rather than C programming. Contrast with embedded event rule. See also event rule.

NER. See named event rule.

network computer. As opposed to the personal computer, the network computer offers (in theory) lower cost of purchase and ownership and less complexity. Basically, it is a scaled-down PC (very little memory or disk space) that can be used to access network-based applications (Java applets, ActiveX controls) via a network browser.

network computing. Often referred to as the next phase of computing after client/server. While its exact definition remains obscure, it generally encompasses issues such as transparent access to computing resources, browser-style front-ends, platform independence, and other similar concepts.

next numbers. A feature you use to control the automatic numbering of such items as new G/L accounts, vouchers, and addresses. It lets you specify a numbering system and provides a method to increment numbers to reduce transposition and typing errors.

non–object librarian object. An object that is not managed by the object librarian.

numeric character. Digits 0 through 9 that are used to represent data. Contrast with alphanumeric characters.

object. A self-sufficient entity that contains data as well as the structures and functions used to manipulate the data. For OneWorld purposes, an object is a reusable entity that is based on software specifications created by the OneWorld toolset. See also object librarian.

object configuration manager (OCM). OneWorld's Object Request Broker and the control center for the runtime environment. It keeps track of the runtime locations for business functions, data, and batch applications. When one of these objects is called, the Object Configuration Manager directs access to it using defaults and overrides for a given environment and user.

object embedding. When an object is embedded in another document, an association is maintained between the object and the application that created it; however, any changes made to the object are also only kept in the compound document. See also object linking.

object librarian. A repository of all versions, applications, and business functions reusable in building applications. You access these objects with the Object Management Workbench.

object librarian object. An object managed by the object librarian.

object linking. When an object is linked to another document, a reference is created with the file the object is stored in, as well as with the application that created it. When the object is modified, either from the compound document or directly through the file it is saved in, the change is reflected in that application as well as anywhere it has been linked. See also object embedding.

object linking and embedding (OLE). A way to integrate objects from diverse applications, such as graphics, charts, spreadsheets, text, or an audio clip from a sound program. See also object embedding, object linking.

object management workbench (OMW). An application that provides check-out and check-in capabilities for developers, and aids in the creation, modification, and use of OneWorld Objects. The OMW supports multiple environments (such as production and development).

object-based technology (OBT). A technology that supports some of the main principles of object-oriented technology: classes, polymorphism, inheritance, or encapsulation.

object-oriented technology (OOT). Brings software development past procedural programming into a world of reusable programming that simplifies development of applications. Object orientation is based on the following principles: classes, polymorphism, inheritance, and encapsulation.

OCM. See object configuration manager.

ODBC. See open database connectivity.

OLE. See object linking and embedding.

OMW. Object Management Workbench.

OneWorld. A combined suite of comprehensive, mission-critical business applications and an embedded toolset for configuring those applications to unique business and technology requirements. OneWorld is built on the Configurable Network Computing technology- J.D. Edwards' own application architecture, which extends client/server functionality to new levels of configurability, adaptability, and stability.

OneWorld application. Interactive or batch processes that execute the business functionality of OneWorld. They consist of reusable business functions and associated data that are platform independent and can be dynamically configured across a TCP/IP network.

OneWorld object. A reusable piece of code that is used to build applications. Object types include tables, forms, business functions, data dictionary items, batch processes, business views, event rules, versions, data structures, and media objects. See also object.

OneWorld process. Allows OneWorld clients and servers to handle processing requests and execute transactions. A client runs one process, and servers can have multiple instances. OneWorld processes can also be dedicated to specific tasks (for example, workflow messages and data replication) to ensure that critical processes don't have to wait if the server is particularly busy.

OneWorld Web development computer. A standard OneWorld Windows developer computer with the additional components installed:

- JFC (0.5.1).
- Generator Package with Generator.Java and JDECOM.dll.
- R2 with interpretive and apllication controls/form.

online. Computer functions over which the system has continuous control. Users are online with the system when working with J.D. Edwards system provided forms.

open database connectivity (ODBC). Defines a standard interface for different technologies to process data between applications and different data sources. The ODBC interface is made up of a set of function calls, methods of connectivity, and representation of data types that define access to data sources.

open systems interconnection (OSI). The OSI model was developed by the International Standards Organization (ISO) in the early 1980s. It defines protocols and standards for the interconnection of computers and network equipment.

operand. See Boolean Logic Operand.

output. Information that the computer transfers from internal storage to an external device, such as a printer or a computer form.

output queue. See print queue.

package. OneWorld objects are installed to workstations in packages from the deployment server. A package can be compared to a bill of material or kit that indicates the necessary objects for that workstation and where on the deployment server the install program can find them. It is a point-in-time "snap shot" of the central objects on the deployment server.

package location. The directory structure location for the package and it's set of replicated objects. This is usually \\deployment server\release\path_code\package\ package name. The sub-directories under this path are where the replicated objects for the package will be placed. This is also referred to as where the package is built or stored.

parameter. A number, code, or character string you specify in association with a command or program. The computer uses parameters as additional input or to control the actions of the command or program.

parent/child form. A type of form that presents parent/child relationships in an application on one form. The left portion of the form presents a tree view that displays a visual representation of a parent/child relationship. The right portion of the form displays a detail area in browse mode. The detail area displays the records for the child item in the tree. The parent/child form supports drag and drop functionality.

partitioning. A technique for distributing data to local and remote sites to place data closer to the users who access. Portions of data can be copied to different database management systems.

path code. A pointer to a specific set of objects. A path code is used to locate:

1. Central Objects.
2. Replicated Objects.

platform independence. A benefit of open systems and Configurable Network Computing. Applications that are composed of a single code base can be run across a TCP/IP network consisting of various server platforms and SQL databases.

polymorphism. A principle of object-oriented technology in which a single mnemonic name can be used to perform similar operations on software objects of different types.

portability. Allows the same application to run on different operating systems and hardware platforms.

portal. A configurable Web object that provides information and links to the Web. Portals can be used as home pages and are typically used in conjunction with a Web browser.

primary key. A column or combination of columns that uniquely identifies each row in a table.

print queue. A list of tables, such as reports, that you have submitted to be written to an output device, such as a printer. The computer spools the tables until it writes them. After the computer writes the table, the system removes the table identifier from the list.

processing option. A feature of the J.D. Edwards reporting system that allows you to supply parameters to direct the functions of a program. For example, processing options allow you to specify defaults for certain form displays, control the format in which information prints on reports, change how a form displays information, and enter beginning dates.

program temporary fix (PTF). A representation of changes to J.D. Edwards software that your organization receives on magnetic tapes or diskettes.

project. An Object Management Workbench object used to organize objects in development.

published table. Also called a "Master" table, this is the central copy to be replicated to other machines. Resides on the "Publisher" machine. the Data Replication Publisher Table (F98DRPUB) identifies all of the Published Tables and their associated Publishers in the enterprise.

publisher. The server that is responsible for the Published Table. The Data Replication Publisher Table (F98DRPUB) identifies all of the Published Tables and their associated Publishers in the enterprise.

pull replication. One of the OneWorld methods for replicating data to individual workstations. Such machines are set up as Pull Subscribers using OneWorld's data replication tools. The only time Pull Subscribers are notified of changes, updates, and deletions is when they request such information. The request is in the form of a message that is sent, usually at startup, from the Pull Subscriber to the server machine that stores the Data Replication Pending Change Notification table (F98DRPCN).

purge. The process of removing records or data from a system table.

QBE. See query by example.

query by example (QBE). Located at the top of a detail area, it is used to search for data to be displayed in the detail area.

redundancy. Storing exact copies of data in multiple databases.

regenerable. Source code for OneWorld business functions can be regenerated from specifications (business function names). Regeneration occurs whenever an application is recompiled, eitherfor a new platform or when new functionality is added.

relationship. Links tables together and facilitates joining business views for use in an application or report. Relationships are created based on indexes.

release/release update. A "release" contains major new functionality, and a "release update" contains an accumulation of fixes and performance enhancements, but no new functionality.

replicated object. A copy or replicated set of the central objects must reside on each client

and server that run OneWorld. The path code indicates the directory the directory where these objects are located.

run. To cause the computer system to perform a routine, process a batch of transactions, or carry out computer program instructions.

SAR. See software action request.

scalability. Allows software, architecture, network, or hardware growth that will support software as it grows in size or resource requirements. The ability to reach higher levels of performance by adding microprocessors.

search/select. A type of form used to search for a value and return it to the calling field.

selection. Found on J.D. Edwards menus, selections represent functions that you can access from a menu. To make a selection, type the associated number in the Selection field and press Enter.

server. Provides the essential functions for furnishings services to network users (or clients) and provides management functions for network administrators. Some of these functions are storage of user programs and data and management functions for the file systems. It may not be possible for one server to support all users with the required services. Some examples of dedicated servers that handle specific tasks are backup and archive servers, application and database servers.

servlet. Servlets provide a Java-based solution used to address the problems currently associated with doing server-side programming, including inextensible scripting solutions. Servlets are objects that conform to a specific interface that can be plugged into a Java-based server. Servlets are to the server-side what applets are to the client-side.

software. The operating system and application programs that tell the computer how and what tasks to perform.

software action request (SAR). An entry in the AS/400 database used for requesting modifications to J.D. Edwards software.

special character. A symbol used to represent data. Some examples are *, &, #, and /. Contrast with alphanumeric character and numeric character.

specifications. A complete description of a OneWorld object. Each object has its own specification, or name, which is used to build applications.

Specs. See specifications.

spool. The function by which the system stores generated output to await printing and processing.

spooled table. A holding file for output data waiting to be printed or input data waiting to be processed.

SQL. See structured query language.

static text. Short, descriptive text that appears next to a control variable or field. When the variable or field is enabled, the static text is black; when the variable or field is disabled, the static text is gray.

store and forward. A transaction method that allows a client application to perform work and, at a later time, complete that work by connecting to a server application. This often involves uploading data residing on a client to a server.

structured query language (SQL). A fourth generation language used as an industry standard for relational database access. It can be used to create databases and to retrieve, add, modify, or deleta data from databases. SQL is not a complete programming language because it does not contain control flow logic.

subfile. See detail.

submit. See run.

subscriber. The server that is responsible for the replicated copy of a Published Table. Such servers are identified in the Subscriber Table.

subscriber table. The Subscriber Table (F98DRSUB), which is stored on the Publisher Server with the Data Replication Publisher Table (F98DRPUB) identifies all of the Subscriber machines for each Published Table.

subsystem job. Within OneWorld, subsystem jobs are batch processes that continually run independently of, but asynchronously with, OneWorld applications.

summary. The presentation of data or information in a cumulative or totaled manner in which most of the details have been

removed. Many of the J.D. Edwards systems offer forms and reports that are summaries of the information stored in certain tables. Contrast with detail.

system. See application.

System Code. System codes are a numerical representation of J.D. Edwards and customer systems. For example, 01 is the system code for Address Book. System codes 55 through 59 are reserved for customer development by customers. Use system codes to categorize within OneWorld. For example, when establishing user defined codes (UDCs), you must include the system code the best categorizes it. When naming objects such as applications, tables, and menus, the second and third characters in the object's name is the system code for that object. For example, G04 is the main menu for Acounts Payable, and 04 is its system code.

system function. A program module, provided by OneWorld, available to applications and reports for further processing.

table. A two-dimensional entity made up of rows and columns. All physical data in a database are stored in tables. A row in a table contains a record of related information. An example would be a record in an Employee table containing the Name, Address, Phone Number, Age, and Salary of an employee. Name is an example of a column in the employee table.

table design aid (TDA). A OneWorld GUI tool for creating, modifying, copying, and printing database tables.

table event rules. Use table event rules to attach database triggers (or programs) that automatically run whenever an action occurs against the table. An action against a table is referred to as an event. When you create a OneWorld database trigger, you must first determine which event will activate the trigger. Then, use Event Rules Design to create the trigger. Although OneWorld allows event rules to be attached to application events, this functionality is application specific. Table event rules provide embedded logic at the table level.

TAM. Table Access Management.

TBLE. See table.

TC. Table conversion.

TCP/IP. Transmission Control Protocol/Internet Protocol. The original TCP protocol was developed as a way to interconnect networks using many different types of transmission methods. TCP provides a way to establish a connection between end systems for the reliable delivery of messages and data.

TCP/IP services port. Used by a particular server application to provide whatever service the server is designed to provide. The port number must be readily known so that an application programmer can request it by name.

TDA. See table design aid.

TER. See table event rules.

Terminal Identification. The workstation ID number.Terminal number of a specific terminal or IBM user ID of a particular person for whom this is a valid profile.Header Field: Use the Skip to Terminal/User ID field in the upper portion of the form as an inquiry field in which you can enter the number of a terminal or the IBM user ID of a specific person whose profile you want the system to display at the top of the list. When you first access this form, the system automatically enters the user ID of the person signed on to the system.Detail Field: The Terminal/User ID field in the lower portion of the form contains the user ID of the person whose profile appears on the same line.A code identifying the user or terminal for which you accessed this window.

third generation language (3GL). A programming language that requires detailed information about how to complete a task. Examples of 3GLs are COBOL, C, Pascal and FORTRAN.

token. A referent to an object used to determine ownership of that object and to prevent non-owners from checking the object out in Object Management Workbench. An object holds its own token until the object is checked out, at which time the object passes its token to the project in which the object is placed.

trigger. Allow you to attach default processing to a data item in the data dictionary. When that data item is used on an application or report, the trigger is invoked by an event associated with the data item. OneWorld also has three

visual assist triggers: calculator, calendar and search form.

UBE. Universal batch engine.

UDC Edit Control. Use a User-Defined Code (UDC) Edit Control for a field that accepts only specific values defined in a UDC table. Associate a UDC edit control with a database item or dictionary item. The visual assist Flashlight automatically appears adjacent to the UDC edit control field. When you click on the visual assist Flashlight, the attached search and select form displays valid values for the field. To create a UDC Edit Control, you must:

- Associate the data item with a specific UDC table in the Data Dictionary.
- Create a search and select form for displaying valid values from the UDC table.

uniform resource identifier (URI). A character string that references an internet object by name or location. A URL is a type of URI.

uniform resource locator (URL). Names the address (location) of a document on the Internet or an intranet. A URL includes the document's protocol and server name. The path to the document might be included as well. The following is an example of a URL: http://www.jdedwards.com. This is J.D. Edwards Internet address.

URI. See uniform resource identifier.

URL. See uniform resource locator.

user defined code (type). The identifier for a table of codes with a meaning you define for the system, such as ST for the Search Type codes table in Address Book. J.D. Edwards systems provide a number of these tables and allow you to create and define tables of your own. User defined codes were formerly known as descriptive titles.

user defined codes (UDC). Codes within software that users can define, relate to code descriptions, and assign valid values. Sometimes user defined codes are referred to as a generic code table. Examples of such codes are unit-of-measure codes, state names, and employee type codes.

UTB. Universal Table Browser.

valid codes. The allowed codes, amounts, or types of data that you can enter in a field. The system verifies the information you enter against the list of valid codes.

visual assist. Forms that can be invoked from a control to assist the user in determining what data belongs in the control.

vocabulary overrides. A feature you can use to override field, row, or column title text on forms and reports.

wchar_t. Internal type of a wide character. Used for writing portable programs for international markets.

web client. Any workstation that contains an internet browser. The web client communicates with the web server for OneWorld data.

web server. Any workstation that contains the IServer service, SQL server, Java menus and applications, and Internet middleware. The web server receives data from the web client, and passes the request to the enterprise server. When the enterprise server processes the information, it sends it back to the web server, and the web server sends it back to the web client.

WF. See workflow.

window. See form.

workflow. According to the Workflow Management Coalition, worlflow means "the automation of a business process, in whole or part, during which documents, information, or tasks are passed from one participant to another for action, according to a set of procedural rules.".

workgroup server. A remote database server usually containing subsets of data replicated from a master database server. This server does not performance an application or batch processing. It may or may not have OneWorld running (in order to replicate data).

workspace. In the ActivEra Portal, the main section of the Portal. A user might have access to several workspaces, each one configured differently and containing its own components.

worldwide web. A part of the Internet that can transmit text, graphics, audio, and video. The

World Wide Web allows clients to launch local or remote applications.

z file. For store and forward (network disconnected) user, OneWorld store and forward applications perform edits on static data and other critical information that must be valid to process an order. After the initial edits are complete, OneWorld stores the transactions in work tables on the workstation. These work table are called Z files. When a network connection is established, Z files are uploaded to the enterprise server and the transactions are edited again by a master business function. The master business function will then update the records in your transaction files.

Index

Index

A

B

C

D

E

F

I

L

M

N

O

P

R

S

T

U

W