October 2015
Describes Oracle’s JD Edwards EnterpriseOne Table Conversion tool and how it is used to convert tables and copy data between tables.
# Contents

**Preface** ........................................................................................................................................................................ vii  
Audience ................................................................................................................................................................................ vii  
Documentation Accessibility ........................................................................................................................................ vii  
Related Information ......................................................................................................................................................... vii  
Conventions ................................................................................................................................................................. vii

## 1 Introduction to JD Edwards EnterpriseOne Table Conversion

1.1 JD Edwards EnterpriseOne Table Conversion Overview .................................................................................. 1-1  
1.2 JD Edwards EnterpriseOne Table Conversion Implementation Steps .................................................. 1-1

## 2 Understanding Table Conversion

2.1 Table Conversions .................................................................................................................................................. 2-1  
2.2 Types of Tables to Use ...................................................................................................................................... 2-2  
2.3 Business Views in Table Conversions .......................................................................................................... 2-2  
2.4 Text Files in Table Conversions ..................................................................................................................... 2-3  
2.5 Sort and Selection Criteria in Table Conversions ......................................................................................... 2-3  
2.6 Input and Output Environments ..................................................................................................................... 2-3

## 3 Setting Up Table Conversions

3.1 Understanding Table Conversion Setup ......................................................................................................... 3-1  
3.2 Understanding the Table Conversion Process Flow ....................................................................................... 3-1  
3.3 Adding Table Conversions ................................................................................................................................ 3-4  
3.3.1 Understanding the Navigation Assistant .................................................................................................. 3-4  
3.3.2 Prerequisites ................................................................................................................................................ 3-4  
3.3.3 Forms Used to Add New Table Conversions .......................................................................................... 3-5  
3.3.4 Adding New Table Conversions .............................................................................................................. 3-5  
3.3.5 Starting the Table Conversion Director .................................................................................................. 3-6  
3.4 Converting Data .................................................................................................................................................. 3-6  
3.4.1 Understanding Data Conversion ............................................................................................................... 3-6  
3.4.1.1 Data Conversion Example ................................................................................................................ 3-7  
3.4.2 Forms Used to Convert Data ................................................................................................................... 3-7  
3.4.3 Defining External Data for Table Conversions ......................................................................................... 3-8  
3.4.4 Defining Input and Output Environments for Table Conversions ....................................................... 3-8  
3.4.5 Defining Table Conversion Input ............................................................................................................ 3-8  
3.4.6 Defining Table Conversion Output ........................................................................................................... 3-11
5 Preparing Foreign Tables for Table Conversion

5.1 Understanding Foreign Tables.................................................................................... 5-1
5.2 Adding Data Sources........................................................................................................ 5-2
5.2.1 Forms Used to Add Data Sources ........................................................................... 5-2
5.2.2 Adding Data Sources ................................................................................................. 5-2
5.3 Adding Environments..................................................................................................... 5-5
5.3.1 Forms Used to Add Environments ............................................................................ 5-5
5.3.2 Adding Environments ............................................................................................... 5-5
5.4 Setting Up Default OCM Mappings............................................................................. 5-6
5.4.1 Understanding OCM Mappings.................................................................................. 5-6
5.4.2 Forms Used to Set Up Default OCM Mappings....................................................... 5-7
5.4.3 Setting Up Default OCM Mappings......................................................................... 5-7
Welcome to the JD Edwards EnterpriseOne Tools Table Conversion Guide.

Audience
This guide is intended for system administrators and technical consultants who are responsible for using the JD Edwards EnterpriseOne Table Conversion tool to rapidly manipulate data in tables.

Documentation Accessibility
For information about Oracle’s commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support
Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Information
For additional information about JD Edwards EnterpriseOne applications, features, content, and training, visit the JD Edwards EnterpriseOne pages on the JD Edwards Resource Library located at:
http://learnjde.com

Conventions
The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>boldface</td>
<td>Indicates field values.</td>
</tr>
<tr>
<td>Italics</td>
<td>Indicates emphasis and JD Edwards EnterpriseOne or other book-length publication titles.</td>
</tr>
<tr>
<td>Monospace</td>
<td>Indicates a JD Edwards EnterpriseOne program, other code example, or URL.</td>
</tr>
</tbody>
</table>
This chapter contains the following topics:

- JD Edwards EnterpriseOne Table Conversion Overview
- JD Edwards EnterpriseOne Table Conversion Implementation Steps

1.1 JD Edwards EnterpriseOne Table Conversion Overview

Oracle’s JD Edwards EnterpriseOne Table Conversion is used to convert tables and copy data between tables.

1.2 JD Edwards EnterpriseOne Table Conversion Implementation Steps

In the planning phase of the implementation, take advantage of all JD Edwards EnterpriseOne sources of information, including the installation guides and troubleshooting information.

The following implementation steps need to be performed before working with JD Edwards EnterpriseOne Data Table Conversion:


2. Add yourself to the system in a developer role so that you have permissions to create Oracle’s JD Edwards EnterpriseOne objects.


3. Set up permissions to create OMW projects.


4. Set up the appropriate database permissions so that you can add tables to the database, drop tables, and add and modify data.

   Work with the database administrator to set up these permissions.
Understanding Table Conversion

This chapter contains the following topics:

- Table Conversions
- Types of Tables to Use
- Business Views in Table Conversions
- Text Files in Table Conversions
- Sort and Selection Criteria in Table Conversions
- Input and Output Environments

2.1 Table Conversions

Table conversion is a type of batch process that enables you to rapidly manipulate the data in tables. This table describes the conversion types that the Oracle's JD Edwards EnterpriseOne Table Conversion tool uses to manipulate data:

<table>
<thead>
<tr>
<th>Conversion Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Conversion</td>
<td>Enables you to transfer or copy data from an input table or business view into output tables using the logic that is necessary to perform the transfer. You can also use Data Conversion to update records in a table or business view.</td>
</tr>
<tr>
<td>Data Copy</td>
<td>Enables you to copy tables from one data source or environment to another data source or environment when the tables are identical.</td>
</tr>
<tr>
<td>Data Copy with Table Input</td>
<td>Enables you to copy tables based on information from an input table. For example, the input table might provide information about which tables are copied, where they are copied, and so on.</td>
</tr>
<tr>
<td>Batch Delete</td>
<td>Enables you to delete records from a table or business view.</td>
</tr>
</tbody>
</table>

The JD Edwards EnterpriseOne Table Conversion tool uses Oracle's JD Edwards EnterpriseOne tables, business views, and text files. The tool can also use non-JD Edwards EnterpriseOne tables, also referred to as foreign tables, as long as they reside in a database that is supported by JD Edwards EnterpriseOne software, such as:

- Oracle
- Access
When creating a table conversion, you:

- Set up a table conversion template.
- Review the template.
- Create versions of the template.

Like batch applications, table conversions include a template and versions. You can override some of the properties within a version at runtime.

- Process the table conversion.

Table conversions can be saved and run multiple times. You can test the table conversion by running it in proof mode.

Oracle’s JD Edwards EnterpriseOne Table Conversion Design Aid enables you to access any available environment for both input and output. The environments that you select determine which tables and business views are available for the conversion and where the tables reside. The environments that you select also determine the specifications, or descriptions, of tables and business views.

### 2.2 Types of Tables to Use

You can use these table types in table conversions:

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JD Edwards EnterpriseOne tables</td>
<td>Tables that exist in the JD Edwards EnterpriseOne database and can be accessed from Object Management Workbench. You can design and modify them using Table Design Aid. At design time, only the specifications are needed to reference the table. At runtime, the table must be generated. An instance of the table must occur in a particular database.</td>
</tr>
<tr>
<td>Foreign tables</td>
<td>Tables that do not have a JD Edwards EnterpriseOne definition but reside in a database that is supported by JD Edwards EnterpriseOne. You must set up a data source and environment in JD Edwards EnterpriseOne to point to the location of foreign tables.</td>
</tr>
</tbody>
</table>

**See Also:**

- Preparing Foreign Tables for Table Conversion.

### 2.3 Business Views in Table Conversions

You must create a business view to establish a relationship among the input tables when you need to transfer data from:

- Multiple tables to a single table.
- Multiple tables to multiple tables.
A business view defines the relationship between two or more tables and joins the data into a single view. You can use JD Edwards EnterpriseOne business views for input to the table conversion only, not for output.

The system does not provide direct support for joining foreign tables. To use multiple foreign tables as input to a conversion, you must first define them through JD Edwards EnterpriseOne and then create a business view for them.

2.4 Text Files in Table Conversions

You can use table conversions to import directly from or export directly to a text file. When you convert a text file, it is stored 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 is used.

2.5 Sort and Selection Criteria in Table Conversions

You can specify sort criteria in a table conversion. Sorting is used to process input rows in a sequence that groups related records together. 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 have a single table of customer information, you can sort the data by area code and split the table into individual tables for each area code. Similarly, you can specify selection criteria for the input table to convert only a subset of that table.

JD Edwards EnterpriseOne Table Conversion Design Aid also enables you to add logic to determine when a change should occur to the value in a field.

2.6 Input and Output Environments

An Oracle’s JD Edwards EnterpriseOne environment consists of:

- A path code.
- Oracle’s JD Edwards EnterpriseOne Object Configuration Manager (OCM) mappings.

The path code of the environment is used to locate specification files for the environment. This path code is usually a subdirectory of the JD Edwards EnterpriseOne directory on a workstation. To reference JD Edwards EnterpriseOne tables in an environment, the full path code must exist on the machines where the conversion is created and run. Foreign tables can be referenced even if the path code does not exist.

OCM Mappings indicate where JD Edwards EnterpriseOne objects reside. JD Edwards EnterpriseOne Table Conversion Design Aid enables you to specify an input and output environment, which is used to locate input and output tables. To locate foreign tables, JD Edwards EnterpriseOne Table Conversion Design Aid uses the default OCM mapping for tables.

JD Edwards EnterpriseOne Table Conversion Design Aid uses these three environments when it processes a table conversion:

- The environment that you are signed into. Determines where the table conversion specifications are stored.
- The environment where the input tables reside.
The environment where the output tables reside.

You can select one environment for the input tables (or business views) and one for the converted output tables. The three environments can be the same or they can be different.

When you use JD Edwards EnterpriseOne tables (or business views) in a table conversion, the environment provides the details of each table or business view, such as:

- Column names
- Data types
- Descriptions

Because this information comes from the JD Edwards EnterpriseOne specification tables, the system table does not have to exist in the database before you can design a table conversion; however, it must be created before the conversion is run. If you use a foreign table as input, you need to create it before you design the table conversion. The JD Edwards EnterpriseOne Table Conversion tool obtains its information about the table directly from the database where the foreign table resides. The environment also provides a default path for text files.

See Also:

- JD Edwards EnterpriseOne Tools Table Design Guide.
- Preparing Foreign Tables for Table Conversion.
This chapter contains the following topics:

- Understanding Table Conversion Setup
- Understanding the Table Conversion Process Flow
- Adding Table Conversions
- Converting Data
- Defining File Formats for Table Conversion Input or Output
- Copying Data
- Copying Data with Table Input
- Deleting Groups of Records
- Using Event Rules in Table Conversions

### 3.1 Understanding Table Conversion Setup

Oracle's JD Edwards EnterpriseOne Table Conversion Design Aid uses a director to guide you through the steps of setting up table conversion templates. Using the director, you can design table conversions for:

- Converting data.
- Copying tables between locations.
- Deleting records within tables.

You can also define user-defined formats for use as input or output tables in table conversion templates. User-defined formats are flat-file tables that store data as a continuous string of information, such as bank tapes.

After you set up a table conversion template, you can save it, create a version, and run the version multiple times.

**See Also:**

- Understanding User-Defined Formats.

### 3.2 Understanding the Table Conversion Process Flow

When you process a table conversion, the system triggers events that are similar to the events that are triggered when a report or application is run. These events are specific to the table conversion that you defined. Events provide pauses in the processing of the table conversion where you can attach logic.
In general, the event flow is the same for all table conversion types because these conversion types are all subsets of a data conversion:

- Data Copy.
  This conversion type does not include input and output tables; all actions are accomplished through the Process Begin event.

- Data Copy with Table Input.
  This conversion type does not include output tables; all actions are accomplished through the Process Begin, Process End, and Row Fetched events.

- Batch Delete.
  As with the Data Copy with Table Input type, this conversion type does not include output tables; all actions are accomplished through the Process Begin, Process End, and Row Fetched events.

The flexibility of events enables you to use table conversion types within other conversion types, if necessary.

Depending on the type of conversion that you define, some events might not be triggered. This diagram illustrates all events that can be triggered in a table conversion; events are indicated by yes or no decisions:
Figure 3–1  Table conversion event flow.

This table describes when each event occurs during the table conversion process:

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Begin</td>
<td>Before fetching records from the input table, the system invokes the Process Begin event. At this point, you can attach logic that needs to run only once at the beginning of a table conversion, or attach any value that does not change for each record. This event is useful for mapping output fields that do not change for each record.</td>
</tr>
</tbody>
</table>
3.3 Adding Table Conversions

This section provides an overview of the navigation assistant, lists prerequisites, and discusses how to:

- Add new table conversions.
- Start the Table Conversion director.

### 3.3.1 Understanding the Navigation Assistant

The JD Edwards EnterpriseOne Table Conversion director includes a navigation assistant. The companion Navigation Assistant tracks where you are in the table conversion development process. You can right-click the Navigation Assistant to hide it for the current design process. The navigation assistant provides a visual indicator of the progress in designing the table conversion. Click a step on the navigation assistant to move forward or backward to another step in the process.

### 3.3.2 Prerequisites

Before you use the table conversion process, ensure that you:

- Set up a data source and environment for any foreign tables from which you might be importing data.
  
  See Preparing Foreign Tables for Table Conversion.

- Create a joined table business view if you are converting data from multiple tables.
  
  See "Table Joins" in the JD Edwards EnterpriseOne Tools Business View Design Guide.

- Create a business function to perform validation if you want the data items within a table to validate against the data dictionary as part of the conversion process.
  
  The Table Conversion tool does not provide automatic data dictionary validation for inputs or outputs.

### Event Table Conversions

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Changed</td>
<td>When you use data sequencing, the system invokes a Data Changed event for any sequenced field that changes. Data Changed events are not cascaded or hierarchical. For example, you can attach an event rule to this event to total a field or group of values.</td>
</tr>
<tr>
<td>Format Fetched</td>
<td>When you use user-defined formats (also known as flat files) in the input table, the system invokes a Format Fetched event for each record that is fetched from the input table. If you use multiple user-defined formats in a table conversion, the Format Fetched event is called for the particular format that is found in the record.</td>
</tr>
<tr>
<td>Row Fetched</td>
<td>An input table invokes a Row Fetched event after each row is fetched from the input table.</td>
</tr>
<tr>
<td>Process End</td>
<td>After the processing of all records is complete, the system invokes the Process End event. Attach event rules to the Process End event 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 table conversion.</td>
</tr>
</tbody>
</table>
3.3.3 Forms Used to Add New Table Conversions

<table>
<thead>
<tr>
<th>Form Name</th>
<th>FormID</th>
<th>Navigation</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add EnterpriseOne Object to the Project</td>
<td>W98220C</td>
<td>Click the Objects node of a project and click Add. on the Object Management Workbench form.</td>
<td>Add a new object to a project.</td>
</tr>
<tr>
<td>Add Object</td>
<td>W9861AF</td>
<td>Select Batch Application and click OK on the Add EnterpriseOne Object to the Project form.</td>
<td>Create a batch application object.</td>
</tr>
<tr>
<td>Batch Application Design</td>
<td>W9860AL</td>
<td>Complete the object information and click OK on the Add Object form.</td>
<td>Access the Table Conversion director.</td>
</tr>
</tbody>
</table>

3.3.4 Adding New Table Conversions

Access the Add Object form.

Object Name
Enter a unique name for the table conversion. Use the recommended naming convention for JD Edwards EnterpriseOne objects.

Description
Enter a meaningful description of the table conversion.

Product Code
Select a user-defined code (UDC) (98/SY) from the range of values that is reserved for clients, 55–59.

Product System Code
Select a UDC (98/SY) that represents the Oracle's JD Edwards EnterpriseOne system where the data is used. This value is used for reporting and description override purposes. Example values include:

- 01: Oracle's JD Edwards EnterpriseOne Address Book
- 03B: Oracle's JD Edwards EnterpriseOne Accounts Receivable
- 04: Oracle's JD Edwards EnterpriseOne Accounts Payable
- 09: Oracle's JD Edwards EnterpriseOne General Accounting
- 11: Oracle's JD Edwards EnterpriseOne Multicurrency
Object Use
Select a UDC (98/FU) that indicates the use of the object. For example, the object might be used to create a program, a master file, or a transaction journal. A 132 value represents a data file conversion.

Object Type
Displays an abbreviation that identifies the type of object that is being created. The object type for a table conversion is UBE. This field is populated by the system based on the type of object that you selected on the Add Object form.

Table Conversion
Select to enable JD Edwards EnterpriseOne Table Conversion Design Aid. If you do not select this option, Oracle’s JD Edwards EnterpriseOne Report Design Aid is launched by default.

3.3.5 Starting the Table Conversion Director
Click OK on the Add Object form:
1. Select the Design Tools tab, and click Start Table Conversion Design Aid. The system displays the introduction form for the JD Edwards EnterpriseOne Table Conversion director.
2. Proceed to the task for the type of conversion that you want to perform.

See Also:
- Converting Data.
- Copying Data.
- Copying Data with Table Input.
- Deleting Groups of Records.

3.4 Converting Data
This section provides an overview of data conversion and discusses how to:
- Define external data for table conversions.
- Define input and output environments for table conversions.
- Define table conversion input.
- Define table conversion output.
- Map inputs to outputs.
- Specify table conversion logging options.
- Review table conversion results.

3.4.1 Understanding Data Conversion
Use the Data Conversion feature on the JD Edwards EnterpriseOne Table Conversion director to move data into tables from:
- A single table.
- Multiple tables that are defined in a business view.
- A single text file.
3.4.1.1 Data Conversion Example

This data conversion example illustrates how to copy employee records from the F0101 table in the Login Environment to the F0101 table in the PY812 environment.

1. On the Introduction form of the JD Edwards EnterpriseOne Table Conversion director, select Data Conversion, and click Next.
2. On the External Data form, accept the default selections, and click Next.
3. On the Select Environments form, select <LOGIN ENV> as the input and PY812 as the output environment, and click Next.
4. On the Select Input form, drag the Address Book Master table to the Description column on the Table tab, and click Next.
5. On the Sequencing form, accept the default selection, and click Next.
6. On the Data Selection form, define the data selection as Where AT1 (Search Type) (F0101) is equal to E to select current employees, and click Next.
7. On the Select Outputs form, drag the Address Book Master table to the Description column on the Tables tab, and click Next.
8. On the Table Options form, clear the Run Currency Triggers option because the Address Book Master file does not contain currency information, and click Next.
9. On the Mapping form, select the Row Fetched event because you are converting data from similar files, click Map Same, and click Next.
   The system automatically maps data elements with the same aliases.
10. On the Logging Options form, select Log All Errors and Log Outputs, and click Next.
    These options enable you to review the conversion after it is complete.
11. On the Finish form, select No, I will create a version of this table conversion later, and click Finish.
12. On the Table Conversion form, click Save before quitting.

3.4.2 Forms Used to Convert Data

<table>
<thead>
<tr>
<th>Form Name</th>
<th>FormID</th>
<th>Navigation</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add EnterpriseOne Object to the Project</td>
<td>W98220C</td>
<td>Click the Objects node of a project and click Add. on the Object Management Workbench form.</td>
<td>Add a new object to a project.</td>
</tr>
<tr>
<td>Add Object</td>
<td>W9861AF</td>
<td>Select Batch Application and click OK on the Add EnterpriseOne Object to the Project form.</td>
<td>Create a batch application object.</td>
</tr>
</tbody>
</table>
3.4.3 Defining External Data for Table Conversions

Access the JD Edwards EnterpriseOne Table Conversion Director.

1. Select Data Conversion, and click Next.

2. On the External Data form, click Select to attach a predefined processing option template to the table conversion.

3. On the Select Processing Option Template form, select an appropriate processing option template, and click OK.

4. On the External Data form, click Define to create a data structure.

5. On the Report Data Structure form, define the table conversion data structure, and click OK.

Data structures contain a list of parameters that can be used to pass data into the table conversion when called through a report interconnect.

6. Click Next.

See Also:
- "Understanding Data Structures" in the JD Edwards EnterpriseOne Tools Data Structure Design Guide.

3.4.4 Defining Input and Output Environments for Table Conversions

Click Next on the Introduction form of the JD Edwards EnterpriseOne Table Conversion director to access the Select Environments form.

1. Select the input and output environments to use.

   **Note:** Select <LOGIN ENV> if you are creating table conversions for a user who might not have the same environments that you have. This option ensures that the table conversion uses the environment that the user is signed in to.

2. When creating a table conversion to be run in a different environment from the one where you create it, and the <LOGIN ENV> is not appropriate for the type of conversion that you are creating, select either Force Version to Override Input Environment or Force Version to Override Output Environment, and click Next.

   For example, if you create a conversion to be used by a client who does not have the environments that you have, use these options. When the conversion is invoked at the client site, the table conversion does not run until the user selects the appropriate environments in which to run it.

3.4.5 Defining Table Conversion Input

Click Next on the Select Environments form to access the Select Input form.
1. Select the appropriate tab based on whether the input is a table, a business view, a foreign table, or a text file.

2. Use the query by example (QBE) line to search for the appropriate table, business view, or foreign table.
   You can select only one table or one business view per table conversion. If the input consists of multiple tables, you must create a joined table business view.

3. Enter the name of the text file on the Text File tab, or use the Browse button to locate the file.
   If you want to rename the selected file, enter the new file name, and click Rename.
   For the IBM i, input text files are stored in the Integrated File System (IFS). Enter the path to the IFS before the file name.
   If you are using a text file, or if you need to define a format for a table or business view, select the User Defined Format option.

   **Note:** Text files are the only type of file that you can rename.

4. If the input is a table or a business view, drag it to the Description column.
   If you change a table or business view, the system warns you that deleting tables removes all mappings from the table conversion.

5. To delete an input file, select it and press Delete.

6. Click Next.

7. On the Sequencing form, click Data Sequencing to define data sequencing for a table or business view.
   If you specify a text file for input, you cannot define data sequencing or data selection for that file.

   **Note:** 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 sequence column 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 Data Changed event is invoked for that column. This event is similar to a level break in report templates, except that the Data Changed events are not related to each other. Invoking one does not invoke the others.

8. Click Next.

9. On the Data Selection form, click in the Operator column, and select an appropriate operator.
   On the Data Selection form, **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.

10. Click the Left Operand column, and select from a list of available objects.
**Note:** You can define selection criteria for database table columns only. User-defined format columns are not available because they do not exist in the database.

11. Click the Comparison column, and select one of these comparison operators:
   - is equal to.
   - is equal to or empty.
   - is greater than.
   - is greater than or equal to.
   - is less than.
   - is less than or equal to.
   - is not equal to.

12. Click the Right Operand column, and select from a list of available objects, special values, or variables.
   The options in this column depend on the selection that you made in the Comparison column. Some of these options might be available:
   - Blank
     Enters a blank (space) value.
   - Literal
     Enables you to enter specific values, as described in the next step.
   - Null
     Indicates that no value is associated with the field.
   - Zero
     Enters a value of 0.
   - IC
     Indicates input table columns.
   - RI
     Indicates values passed through a report interconnect to this table conversion.
   - PO
     Indicates processing option values for this table conversion.
   - SL
     Indicates system literals.

13. If you enter a literal in the Right Operand column, you can enter values on these tabs of the Single value form:
   - Single value
     Enables you to enter a single value. An example might be a value that indicates a specific company.
   - Range of values
Enables you to enter a range of values. An example might include companies from **00001** to **00060**. When you are using a range of values, only the *is equal to*, *is equal to or empty*, and *is not equal to* operators are valid.

- **List of values**

  Enables you to enter a list of values. Enter each value individually in the field, and click Add after each entry.

  An example list of values might include several user-defined codes for search types, such as *C* for customers, *E* for employees, and *V* for vendors. When you are using the list of values, only *is equal to*, *is equal to or empty*, and *is not equal to* are valid operators.

  To delete a value, select the value and click Delete.

**14.** To delete a line of criteria on the Data Selection form, select the row header and click Delete.

**15.** To change the order of the criteria, select the row header, and click the Up or Down button.

**16.** Click Next.

### 3.4.6 Defining Table Conversion Output

Click Next on the Select Input form to access the Select Outputs form.

1. Drag the table (or tables) that you want to use as outputs to the Description column, and click Next.

2. For text file conversions, from the Text File tab, select the file to use as the output, and click Use.

   If you want to rename the selected file, enter the new file name, and click Rename.

   For the IBM i, input text files are stored in the IFS. Enter the path to the IFS before the file name.

   If you are using a text file, or if you need to define a format for a table or business view, select the User Defined Format option.

**Note:** Text files are the only type of file that you can rename.

3. Click Next.

4. To delete an output, select the row and press Delete.

   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.

5. Click Next.

6. On the Table Options form, select from these options:

   - **Run Currency Triggers**

     Select this option if a JD Edwards EnterpriseOne table that is included in the table conversion contains currency triggers. If a table contains currency fields and you do not select this option, the system cannot determine where the decimal should be placed within a field. If you do not select the currency trigger option, and the source or destination fields are currency fields that are used in a calculation, you might receive unexpected results.
Do not select the Run Currency Triggers option if the input and output data sources are the same type (for example, Oracle, IBM i, or SQL Server) and no calculations are performed. Selecting this option results in slower performance.

You should not use currency triggers for an environment that has a different path code from the sign-in environment.

- **Clear Output Tables**
  Select this option to clear the output table before the table conversion runs.

- **Force Row by Row Processing**
  Select this option to test the table conversion or to ensure that the conversion always runs in row-by-row mode.

  You can test a conversion to ensure that the mapping logic is correct. In this case, specify the number of rows to process. The number of rows can be defined in the jde.ini under [TCEngine] or when you submit the table conversion.

  Select this option if you know that the values in the input table will produce duplicate keys in the output, and you want only the non-duplicate keys to be inserted.

  Selecting this option results in slower processing.

- **Buffer Inserts To Output Tables**
  Select this option to improve conversion performance if the conversion does not include any event rules to process insertion errors and you are processing row by row.

7. **Click Next.**

### 3.4.7 Mapping Inputs to Outputs

Click Next on the Select Outputs form to access the Mapping form.

1. **Select the event on which mapping will occur.**

   In most cases, you use either the Row Fetched event or Format Fetched event. Use the Format Fetched event if you are working with a user-defined format.

2. **Click Advanced ER to include additional event rule logic in the table conversion.**

   The Event Rules Design form appears for the event that you selected on the Mapping form.

3. **Click Map Same to allow the system to map the inputs directly to the outputs.**

   If the input and outputs share some of the same data, these fields are a direct map. For JD Edwards EnterpriseOne tables, the system maps fields by data dictionary item. For foreign tables, the system maps by column name.

4. **Drag inputs to outputs to manually map the inputs to the outputs.**

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

5. **If multiple output files exist, select each file from the outputs list, and map the appropriate input columns to the appropriate output columns.**
6. To define advanced output, double-click an output column.

The Advanced Outputs form enables you to define literals, calculations, and other mappings without using Advanced ER. You can use an advanced input to add a literal value into a field. You can also define a calculation to populate an output field, such as adding two input fields together.

7. On the Advanced Outputs form, select one of these tabs, and add the appropriate input:
   - Available objects
     Select the appropriate object, and click Apply.
   - Literal
     Enter the appropriate value, and click Apply.
   - Defaults
     Select Use Dictionary Defaults, and click Apply.
     Select this option to use the default value in the data dictionary at runtime. If no default values exist in the data dictionary, the system displays a warning message.
   - Calculation
     Click Define Calculation, create a calculation in Expression Manager, and click Apply.

8. Click Close on the Advanced Outputs form.

9. On the Mapping form, select Issue a Write for this Event? to insert a row to the selected output after performing all column mappings for this event.

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

10. Click Next.

   See Also:
   - Understanding the Table Conversion Process Flow.
   - Using Event Rules in Table Conversions.

### 3.4.8 Specifying Table Conversion Logging Options

Click Next on the Mapping form to access the Logging Options form.

1. Select from these options:
   - Log All Errors.
     Select this option to log every error, regardless of the logging option for type of operation that failed.
   - Log Every Input Record.
     Select this option to log every input record in the table conversion log file.
   - Log Outputs.
     Select this option to log every output record in the table conversion log file.
- Log Deletes.
  Select this option to log every record that is deleted in the table conversion log file.

- Log Updates.
  Select this option to log every record that is updated in the table conversion log file.

- Log Copy Table Actions.
  Select this option to log the tables copied in the table conversion log file.

- Log Details of Copy Table Actions.
  Select this option to log all of the details of each Copy Table action that is performed.

- Run in Proof Mode.
  Select this option to run the conversion in proof mode. In proof mode, the conversion is simulated and all actions are logged, but no data is changed.

---

**Note:** Proof mode is not an absolute proof mode. In some situations, the proof output might differ from the real output. If you insert the same record twice, for example, it may seem as though it worked in proof mode but, in reality, only one of the inserts works when you run the conversion in final mode.

---

2. Click Next.

### 3.4.9 Reviewing Table Conversion Results

Click Next on the Logging Options form to access the Finish form.

1. Select one of these options:
   - Yes, create a version of this table conversion.
     If you select yes, enter the version name.
   - No, I will create a version of this table conversion later.

2. Click Finish to complete the process.

3. On the Table Conversion form, select the Table Conversion Mappings form to review the mapping.

4. Make changes, as necessary, by selecting the appropriate option from the View menu, and then click Save.

5. To review the runtime conversion mode, select Check Conversion Mode from the Conversion menu.
   A dialog box provides you with details of the conversion mode that is defined.

6. From the Conversion menu, select Exit.

### 3.5 Defining File Formats for Table Conversion Input or Output

This section provides an overview of user-defined formats, lists a prerequisite, and discusses how to:
- Define delimited file formats with a single format.
- Define delimited file formats with multiple formats.
- Define fixed-width file formats with a single format.
- Define fixed-width file formats with multiple formats.

### 3.5.1 Understanding User-Defined Formats

JD Edwards EnterpriseOne Table Conversion Design Aid uses user-defined formats to handle fixed-width or comma 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 like relational database tables. Usually, flat files are text files that are stored on a workstation or server. They are used to import or export data from applications that have no other means of interaction. For example, you might need to share information between JD Edwards EnterpriseOne and an external application. If the external application does not support one of the same databases that JD Edwards EnterpriseOne supports, flat files might be the only way to transfer data between the two applications.

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

This table illustrates a single database record with five columns of data: Last, First, Addr, City, and Phone:

<table>
<thead>
<tr>
<th>Last</th>
<th>First</th>
<th>Addr</th>
<th>City</th>
<th>Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Doe</td>
<td>John</td>
<td>123 Main</td>
<td>Anytown</td>
<td>5551234</td>
</tr>
</tbody>
</table>

This record can be formatted as a comma-delimited string:

`Doe, John, 123 Main, Anytown, 5551234`

This record can also be formatted as a fixed-width column format, in which each column is eight characters long.

### 3.5.1.1 Importing and Exporting Text Files

When you select a text file for input or output in a table conversion and do not specify a path, a default path is used. Table conversions that use the default path run on any platform. If an explicit path or IBM i library name is indicated for the file, then the file is created exactly as specified. Table conversions that are stored in this way might not work on other platforms, depending on the nature of the file system on each platform.

The default paths on non-IBM i platforms are:

- `path code\Import\file name`
- `path code\Export\file name`

You cannot specify a default path for the IBM i. Rather, the default is always the Import or the Export directory under the path code of the input or output environment. For example, if you are running a table conversion against the DV812 environment, the path in the file system might be `\E812\DV812\import\myfile.txt`.

If the conversion specifies a file name that includes anything other than the file name and extension, such as `\mytextfiles\myfile.txt`, the conversion attempts to open the file as specified.
3.5.1.2 Using User-Defined Formats as Input

If you use user-defined input formats, add event rules on the Format Fetched event. Without these event rules, the system ignores the format, and the data from the input table is never made available to the table conversion. If the table conversion does not require event rule logic, add comments in the Format Fetched event in Event Rules Design.

User-defined formats work with text files and tables.

Because the procedure for importing and exporting data is database-specific, consult a database administrator for details.

3.5.1.3 Using User-Defined Formats as Output

If you use user-defined output formats, add event rules on the Format Fetched event. Without these event rules, the system ignores the format, and the data from the output table is never made available to the table conversion. If the table conversion does not require event rule logic, add comments in the Format Fetched event in Event Rules Design.

User-defined formats work with text files and tables.

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

3.5.2 Prerequisite

Before you begin defining delimited file formats, ensure that you have selected the input and output files, as appropriate.

3.5.3 Defining Delimited File Formats with a Single Format

From the JD Edwards EnterpriseOne Table Conversion director, access the Select Input form or the Select Outputs form.

1. Select User Defined Format, and then click Define Format(s).
2. On the User Defined Format - Type form, select the Delimited - Characters such as commas or tabs separate each column option.
3. Select the Single Format - All rows have the same format option, and click Next.
4. On the User Defined Format - Column Delimiter form, select the delimiter that separates the columns in the file:
   - Tab
   - Comma
   - Semicolon
   - Space
   - Other
5. Select the textual qualifier that is used to enclose a string of text:
   - None
     Select this option if no character is used to qualify text in this user-defined file.
   - Single Quotations
Defining File Formats for Table Conversion Input or Output

3.5.4 Defining Delimited File Formats with Multiple Formats

From the JD Edwards EnterpriseOne Table Conversion director, access the Select Input form or the Select Outputs form.

1. Select User Defined Format, and then click Define Format(s).

2. On the User Defined Format - Type form, select the Delimited - Characters such as commas or tabs separate each column option.

3. Select the Multiple Formats - Rows are in two (2) or more formats option, and click Next.

4. On the User Defined Format - Column Delimiter form, select the delimiter that separates the columns in the file:
   - Tab
   - Comma
5. Select the textual qualifier that is used to enclose a string of text, and click Next:
   - None
     Select this option if no character is used to qualify text in this user-defined file.
   - Single Quotations
     Select this option if a single quotation mark is used to qualify text in this user-defined file.
   - Double Quotations
     Select this option if a double quotation mark is used to qualify text in this user-defined file.

6. On the User Defined Format - Multiple Format Definition form, enter the number of formats that are included in the user-defined format.

7. To define the character length of the Form Designator Column, enter a value in the Length field, and click Next.

8. On the User Defined Format - Multiple Format Names form, name each format and define the values for each using the Name and Designator columns, and then click Next.

   The Designator name should describe the data in the user-defined formats.

   For example, suppose that a text file contains purchase order information. Lines in the table have information for a whole purchase order with the first field designated as POH; 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 of the third format.

   **Note:** You can rename the columns for each format to help you 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 appear in the Inputs list on the Mapping form of the table conversion.

9. On the User Defined Format - Column Layout form, select 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.

   **Note:** To move rows, including the Format Designator, select the row and drag it to a new location.

10. To add additional columns, click Add to define the column.

11. On the New Column Properties form, modify these fields as required, and click OK:
- Name
- Length
- Type

12. To edit an existing column, select it, click Edit, and change the properties on the Column Properties form.

13. To model the columns after an existing table, business view, or foreign table, click Model, select the appropriate tab, and then select 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.

---

14. Click OK, and then click Next.

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


### 3.5.5 Defining Fixed-Width File Formats with a Single Format

Access the Select Input form or the Select Outputs form.

1. Select User Defined Format, and then click Define Format(s).

2. On the User Defined Format - Type form, select the Fixed Width - Columns are aligned at specific positions with specific widths option.

3. Select the Single Format - All rows have the same format option, and click Next.

4. On the User Defined Format - Column Layout form, click Add to define the column layout.

5. On the New Column Properties form, modify these fields as required, and click OK:
   - Name
   - Start
   - Length
   - Type

6. To edit an existing column, select the column, click Edit, and change the properties on the Column Properties form.

7. To model the columns after an existing table, business view, or foreign table, click Model, select the appropriate tab, and then select 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.

---

8. Click OK, and then click Next.

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

3.5.6 Defining Fixed-Width File Formats with Multiple Formats

Access the Select Input form or the Select Outputs form.

1. Select User Defined Format, and then click Define Format(s).

2. On the User Defined Format - Type form, select the Fixed Width - Columns are aligned at specific positions with specific widths option.

3. Select the Multiple Formats - Rows are in two (2) or more formats option, and click Next.

4. On the User Defined Format - Multiple Format Definition form, enter the number of formats that are included in the user-defined format, and click Next.

   To define the character length of the Designator column, complete the Length field.

5. On the User Defined Format - Multiple Format Names form, enter a name for each format in the Name column, and click Next.

   The Designator name should describe the data in the user-defined formats.

   For example, suppose that a text file contains purchase order information. Lines in the table have information for a whole purchase order with the first field designated as POH; 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 of the third format.

---

**Note:** You can rename the columns for each format to help you 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 appear in the Inputs list on the Mapping form of the table conversion.

---

6. On the User Defined Format - Column Layout form, select 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.

7. To add additional columns, click Add to define the columns.

8. On the New Column Properties form, modify these fields as required, and click OK:

   - Name
   - Start
   - Length
   - Type

9. To edit an existing column, select the column, click Edit, and change the properties on the Column Properties form.
10. To model the columns after an existing table, business view, or foreign table, click Model, select the appropriate tab, and then select 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.

11. Click OK, and then click Next.

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


See Also:
- Reviewing Table Conversion Results.
- Using Event Rules in Table Conversions.
- Understanding Table Conversion Processing.

### 3.6 Copying Data

This section provides an overview of copying data, lists prerequisites, and discusses how to define data copy actions.

#### 3.6.1 Understanding Copying Data

You use the Data Copy feature on the JD Edwards EnterpriseOne Table Conversion director to copy tables from one environment or data source to another. You can also import a copy table script to use in the conversion.

#### 3.6.2 Prerequisites

Before you begin defining the input and output environments for copying data, ensure that you review these tasks:
- Defining External Data for Table Conversions.
- Defining Input and Output Environments for Table Conversions.
- Specifying Table Conversion Logging Options.
- Reviewing Table Conversion Results.

#### 3.6.3 Defining Data Copy Actions

Access the JD Edwards EnterpriseOne Table Conversion Director.

1. Select Data Copy, and click Next.

2. On the External Data form, select a processing option template, define a data structure, as appropriate, and click Next.

3. On the Select Environments form, select an input environment and an output environment, and click Next.

4. On the Select Actions form, enter the name of a table in the Table field of the grid.
If you want to copy a single table, select <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.

When you enter the name of a table, the system automatically populates the remaining fields for you.

5. On the Select Actions form, modify these fields as necessary:

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

   - Source Type
     Select Data Source if the input and output sources are data sources. Select Environment if the input and output sources are environments.

     When you select Data Source, the system retrieves table definitions from the specifications in the sign-in environment.

     When you select Environment, the system uses the input and output environments to locate data and specifications for the tables. This enables the specifications to be different in the input and output environment, but the data is copied.

   - Input Source
     Select the data source or environment from which the input is read.

   - Output Source
     Select the data source or environment to which the output is written.

   - Create
     When you select <If Table Exists>, the system creates the table in the output and runs the conversion only if both the table specifications and the actual table exist in the input.

     When you select <Yes>, the system creates the table in the output. If the table already exists in the output, the system deletes it and creates a new table.

     When you select <No>, the system assumes that the table already exists in the output and does not create it.

   - Clear
     When you select <If Table Exists>, the system clears the table in the output only if it exists in the input.

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

     When you select <No>, the output table is not cleared.

---

**Note:** If you select <No> to not clear the output table, the conversion could result in key conflicts.

---

- Copy
  When you select <Yes>, the system copies the data from the input table to the output table using Map Same.
When you select <No>, no data is copied.

- Owner ID
- Owner Pwd (owner password)

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 sign-in user, or <None> if the data source does not have security.

6. To import an existing copy table script from another location, click Import, locate 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.

7. Click Advanced ER to add event rule logic to the copy table process, and click Next.

You can use event rules to write a custom copy table script.

8. On the Logging Options form, select the appropriate logging options, and click Next.


See Also:
- Using Event Rules in Table Conversions.
- Reviewing Table Conversion Results.
- Running Table Conversions.

3.7 Copying Data with Table Input

This section provides an overview of how to copy data with table input, lists prerequisites, and discusses how to define data copy with table input actions.

3.7.1 Understanding How to Copy Data with Table Input

The Data Copy with Table Input feature is similar to the Data Copy option, except that it allows information for the process to come from an input table. The input table might provide information about which tables are copied, where they are copied, and so on. The Data Copy with Table Input feature enables you to select data.

For example, suppose that you create a table that includes:

- A table name
- The next backup date
- The backup frequency

You might populate this table with a list of tables to be archived and information specifying how often they are archived. You can then use the Data Copy with Table Input feature to select all rows in which the backup date is less than or equal to the current date, and calculate a new backup date.

3.7.2 Prerequisites

Before you begin defining the input and output environments for copying data, ensure that you review these tasks:

- Defining External Data for Table Conversions.
3.7.3 Defining Data Copy with Table Input Actions

Access the JD Edwards EnterpriseOne Table Conversion Director.

1. Select Data Copy with Table Input, and click Next.
2. On the External Data form, select a processing option template, define a data structure, as appropriate, and click Next.
3. On the Select Environments form, select an input environment and an output environment, and click Next.
4. On the Select Input form, select a table, business view, or foreign table, and click Next.
5. On the Sequencing form, define data sequencing as appropriate, and click Next.
6. On the Data Selection form, define data selection as appropriate, and click Next.
7. On the Table Options form, select Run Currency Triggers, if appropriate, and click Next.

Select this option if a JD Edwards EnterpriseOne table included in the table conversion contains currency triggers. If a table contains currency fields and you do not select this option, the system cannot determine where the decimal should be placed within a field. If you do not select the currency trigger option and the source or destination fields are currency fields used in a calculation, you might receive unexpected results.

Do not select Run Currency Triggers if the input and output data sources are the same type (for example, Oracle, IBM i, or SQL Server) and no calculations are being performed. Selecting this option results in slower processing.

You should not use currency triggers for an environment that has a different path code than the login environment.

8. On the Select Actions form, enter the name of a table in the Table field, modify the remaining fields as appropriate, and click Next.
9. To import an existing copy table script from another location, click Import, locate the file that you want to import, and click Open.
10. Click Advanced ER to add event rule logic to the copy table process, and click Next.
11. On the Logging Options form, select logging options, and click Next.
12. On the Finish form, select a version option, and click Finish.

3.8 Deleting Groups of Records

This section provides an overview of batch deletion, lists prerequisites, and discusses how to define batch deletion actions.
3.8.1 Understanding Batch Deletion

The Batch Delete feature enables you to delete a range of records from a JD Edwards EnterpriseOne 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 records in an input table that does not contain valid data. You can also set up table conversions that delete all records from a particular table.

3.8.1.1 Example: Creating Purge Programs as a Batch Delete

This example illustrates how you can create a table conversion to delete records from the input environment. Designing purge programs as batch delete programs enables you to purge records with control and accuracy. You can archive the purged data or remove it from the system permanently. The archiving process is explained in this example.

Before you start this example, create a handle for the table.

See "Understanding Handles" in the JD Edwards EnterpriseOne Tools Table Design Guide.

1. On the introduction form of the JD Edwards EnterpriseOne Table Conversion director, select Batch Delete and click Next.

2. On the External Data form, select the Purge Processing Option (T42000P) processing option template, and click Next.

3. On the Select Environment form, select <LOGIN ENV> as the source environment, select Force Version to Override Input Environment, and click Next.

   The Force Version to Override Input Environment option prompts the person who runs the purge program to provide a valid source environment from which to run the batch-delete.

4. On the Select Input form, drag the User Defined Code Types (F0004) table to the Description column, and click Next.

5. On the Table Options form, select Run Currency Triggers, and click Next.

6. On the Data Selection form, define data selection as Where IC UCD1 (User Defined Code - Class Code 1) is equal to Null, and click Next.

7. On the Logging Options form, do not select any options, and click Next.

8. On the Finish form, select Yes, create a version of this table conversion, and enter VER0001 as the version name.

9. On the warning form that indicates that the table conversion needs to be saved, click OK.

10. On the Selection for Batch Delete form, select the Process Begin event, and click Advanced ER.

11. In Event Rules Design, add these event rule variables with a report scope:

    - FXXXXHandle_HFXXXX
    - szArchiveDataSource_DATS
    - szPurgeDataSource_DATS
    - szErrorCode_DTAI
    - cRenameFlag_EV01
    - mnErrorNumber_MATH01
12. Enter these begin process event rules from R42119P, save the event rules, and quit Event Rules Design:

```
0001 // Check to see if the purged data is being archived
0002 If PO cArchive =
Records is equal to '1'
0003 // If the environment processing option is blank, stop processing.
0004 If PO szArchiveEnvironmentName is equal to <Blank>
   Or PO szArchiveEnvironmentName is equal to <Null>
0005 Stop Conversion Processing("The archive environment is invalid.")
0006 Else
0007 // Check to make sure that the archive environment and data source is not the same as the input environment and data source
0008 If PO szArchiveEnvironmentName is equal to SL SourceEnvironment
0009 Stop Conversion Processing("The source and archive environments are the same")
0010 Else
0011 //Get the data source of the archive environment
0012 Get and validate the data source for an environment/table (B98700)
   PO szArchiveEnvironmentName -> szEnvironment
   "FXXXX" -> szTableName (Replace x's with the name of your table.)
   VA rpt_szArchiveDataSource_DTAI <- szDataSource
   VA rpt_szErrorCode_DTAI <- szErrorDataItem
   VA rpt_mnErrorNumber_MATH01 <- mnErrorNumber
0014//SAR #4337575 - B98700 has been modified to return an error if the table is not found in the data source. Since we do not want the table to exist in the archive environment, we need to bypass that error condition.
0015 If VA rpt_szErrorCode_DTAI is not equal to <Blank>
   And VA rpt_szErrorCode_DTAI is not equal to <Null>
   And VA rpt_szErrorCode_DTAI is not equal to '072W'
0016 Stop Conversion Processing("No data source was found for the archive environment")
0017 Else
0018 //Get the data source of the source environment
0019 Get and validate the data source for an environment/table (B98700)
   SL SourceEnvironment -> szEnvironment
   "FXXXX" -> szTableName (Replace x's with the name of your table.)
   VA rpt_szPurgeDataSource_DTAI <- szDataSource
   VA rpt_szErrorCode_DTAI <- szErrorDataItem
```
VA rpt_mnErrorNumber_MATH01 <- mnErrorNumber
0022If VA rpt_szErrorCode_DTAI is equal to '072W'
0023Stop Conversion Processing ("The table could not be found in the source environment.")
0024Else
0025If VA rpt_szErrorCode_DTAI is not equal to <Blank>
And VA rpt_szErrorCode_DTAI is not equal to <Null>
0026Stop Conversion Processing ("No data source was found for the source environment.")
0027End If
0028End If
0029If VA rpt_szArchiveDataSource_DATS is equal to VA rpt_szPurgeDataSource_DATS
0030Stop Conversion Processing ("The source and archive environments have the same data source")
0031Else
0032// Open a table with the same table name in the output environment.
The table will be renamed later if the table name processing option was populated.
0034//Replace the X's in the following statement to reflect the name of your table.
0035Copy Table Environment('FXXXX', <None>, SL Source Environment, PO szArchiveEnvironmentName, <Yes>, <Yes>, <No>, <None>, <Null>, <Null>)
0036// Open a handle to the archive table
0037//Replace the X's in the following statements to reflect the name of your handle and table.
0038VA rpt_FXXXXHandle_HFXXXX = FXXXX.Open Handle
0039If VA rpt_szFXXXXHandle_HFXXXX is equal to <Null>
0040Stop Conversion Processing ("Failed to open FXXXX in the archive environment")
0041End If
0042End If
0043End If
Note: Ensure that you map all event rule variables, even if you do not use every value.

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

13. On the Selection for Batch Delete form, select the Row Fetched event, and click Advanced ER.

14. On Event Rules Design, enter these row fetched event rules from R42119P, save the event rules, and quit Event Rules Design:

```plaintext
001 // If we are archiving the purged records, write the record to the archive
    table
0002 // Replace the X's in the following statements to reflect the name of your handle and table. Map the fields included in your table.
0003 If PO cArchive
    Records is equal to '1'
0004 FXXXX(VA rpt_FXXXXHandle_HFXXXX).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/Order Number "="
    TK Related PO/Order Number
    IC Related PO/Order Type -> TK Related PO/Order
    /WO Order Type
    IC Related PO/Order Line Number -> TK Related PO/Order Line Number
    IC Agreement Number-Distribution
    Agreement Number-Distribution -> TK Agreement Number-Distribution
    IC Agreement
    Supplement-Distribution -> TK Agreement Supplement-Distribution
    IC Address
    Number -> TK Address Number
```
Deleting Groups of Records

IC Address Number-Ship To -> TK Address Number-Ship
To
IC Address Number-Parent -> TK Address Number-Parent
IC Date-Requested -> TK Date-Requested
TK Date-Requested
IC Date-Order/Transaction -> TK Date-Order/Transaction
IC
Date-Scheduled Pick -> TK Date-Scheduled Pick
IC Date-Actual Ship Date -> TK
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
TK Date-For G/L(and Voucher)
Date-Promised Delivery -> TK Date-Promised Delivery
IC Date-Price Effective
TK Date-Price Effective
Date -> TK Date-Price Effective Date
IC Date-Promised Shipment -> TK Date-Promised Shipment
Promised Shipment
IC Reference -> TK Reference
IC Reference 2 -> TK Reference
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
TK From Potency
IC Days Before Expiration -> TK Days Before Expiration
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
TK Item
Number - Related (Kit)
IC Kit Master Line Number -> TK Kit Master Line Number
TK Kit Master Line Number
IC Component Line Number -> TK Component Line Number
TK Component Line Number
IC Related Kit Component
TK Related Kit Component
Deleting Groups of Records

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 -> 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 Price Override Code -> TK Price Override Code
IC Temporary Price (Y/N) -> TK Temporary Price (Y/N)
IC Unit of Measure -> TK Unit of Measure
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
Deleting Groups of Records

⇒
⇒
⇒
⇒
⇒
⇒
⇒
⇒
⇒
⇒

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 % - 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 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

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 Nature of Transaction
IC Primary/Last Supplier Number -> TK
Deleting Groups of Records

Primary/Last Supplier Number
  IC Carrier Number -> TK Carrier Number
  IC Mode =>
of Transport -> TK Mode 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 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 =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>
  =>

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 G/L Offset -> TK
G/L Offset
  IC Century -> TK Century
  IC Fiscal Year -> TK Fiscal Year
  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
Deleting Groups of Records

Setting Up Table Conversions

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⇒
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 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⇒
Deleting Groups of Records

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
  IC Manufacturing Variance Accounting Flag -> TK Manufacturing
  ⇒
  Variance Accounting Flag
  IC Sales Order Status 17 -> TK Sales Order Status 17
  ⇒
  IC Sales Order Status 18 -> TK Sales Order Status 18
  IC Sales Order Status 19 -> TK Sales Order Status 19
  IC Sales Order Status 20 -> TK Sales Order Status 20
  IC Integration Reference 01 -> TK Integration Reference 01
  IC Integration Reference 02 -> TK Integration Reference 02
  IC Integration Reference 03 -> TK Integration Reference 03
  IC Integration Reference 04 -> TK Integration Reference 04
  IC Integration Reference 05 -> TK Integration Reference 05
  IC Source of Order -> TK Source of Order
  IC Reference -> TK Reference
  IC Demand -> TK Demand
  IC Pull Signal -> TK Pull Signal
  IC Release Number -> TK Release Number
  IC Scheduled Shipment Time -> TK Scheduled Shipment Time
  IC Time-Release -> TK Time-Release
  IC Date-Release -> TK Date-Release
  IC Requested Delivery Time -> TK Requested Delivery Time
  IC Actual -> TK Actual Shipment Time
  IC Time-Original Promised Delivery -> TK Time-Original Promised Delivery
  TK Time-Scheduled Pick -> TK Time-Scheduled Pick
  TK Cross-Docking Flag -> TK Cross-Docking Flag
  TK Cross-Docking Priority for Sales Orders -> TK Cross-Docking Priority for Sales Orders
  TK Dual Unit of Measure Item -> TK Dual Unit of Measure Item
  TK Buying Segment Code -> TK Buying Segment Code
  TK Current Buying Segment Code -> TK Current Buying Segment Code
  TK Change Order Number -> TK Change Order Number
TK Change Order Number
  IC Address Number-Deliver To -> TK Address Number
Deliver To
  IC Pending Approval Flag -> TK Pending Approval Flag
  IC Revision
Reason -> TK Revision Reason
  IC Matrix Control Line Number -> TK Matrix
Control Line Number
  IC Shipment Number -> TK Shipment Number
  IC Promised
Delivery Time -> TK Promised Delivery Time
  IC Project Number -> TK Project
Number
  IC Sequence Number -> TK Sequence Number
  IC Item Revision Level -> TK
Item Revision Level
  IC Hold Orders Code -> TK Hold Orders Code
  IC Business
Unit-Header -> TK Business Unit-Header
  IC Business Unit-Demand -> TK Business
Unit-Demand
  IC Cross Dock Order Company
  (OrderNumber) -> TK Cross Dock Order Company(OrderNumber)
  IC Cross Dock Order No ->
⇒
⇒
⇒
⇒
⇒
⇒
⇒
⇒
⇒
TK Cross Dock Order No
  IC Cross Dock Order Type -> TK Cross Dock Order Type
⇒
  IC Cross Dock Line Number -> TK Cross Dock Line Number
⇒
  IC Cross Dock Order Suffix
  IC Port of Entry or Exit -> TK Port of Entry or Exit
⇒
  IC Payment Terms Override Code -> TK Payment Terms Override Code
⇒
  IC Buyer Number -> TK Buyer Number
  IC Promotion ID -> TK Promotion ID
⇒
  IC Asset Item Number -> TK Asset Item Number
  IC Parent Number -> TK Parent Number
⇒
0005 // Do not delete the record if the insert to the archive table failed.
0006 If⇒
  SV Error_Status is not equal to CO ERROR
0007 Delete Current Input Row
0008 End⇒
  If
0009 Else
0010 Delete Current Input Row
15. On the Selection for Batch Delete form, select the Process End event, and click Advanced ER.

16. On Event Rules Design, enter these process end event rules from R42119P, save the event rules, and quit Event Rules Design:

0001 If PO cArchiveRecords is equal to "1"
0002 // Close the table
0003 //Replace⇒
the X's in the following statement to reflect the name of your handle and table.
⇒
0004 FXXXX(VA rpt_FXXXXHandle_HFXXXX).Close
0005 // If the data was archived and⇒
the table name processing option was populated,
0006 // rename the table.
0007 If PO⇒
szArchiveTableName is not equal to <Blank>
And PO szArchiveTableName is not⇒
equal to <Null>
0008 Rename Table (B0000202)
"FXXXX" -> szOldTableName
PO sz⇒
ArchiveTableName -> szNewTableName
"<Blank>" -> szTableOwnerID
"<Blank>" -> sz⇒
Password
VA rpt_szArchiveDataSource_DATS -> szDataSource
VA rpt_cRenameFlag_⇒
EV01 <- cRenameTableSuccessful
0009 End If
0010 End If

Note: Ensure that you map all parameters to a field, even if you do not use every value.

See Also:
- Deleting Groups of Records.
- "Understanding Hanldes" in the JD Edwards EnterpriseOne Tools Table Design Guide.

3.8.2 Prerequisites

Before you begin defining the input and output environments for copying data, ensure that you review these tasks:
- Defining External Data for Table Conversions.
3.8.3 Defining Batch Deletion Actions

Access the JD Edwards EnterpriseOne Table Conversion Director.

1. Select Batch Delete, and click Next.

2. On the External Data form, select a processing option template, define a data structure, as appropriate, and click Next.

3. On the Select Environments form, select a source environment, and click Next.

4. On the Select Input form, drag the appropriate table to the Description column, and click Next.

To delete an input table, select it and press Delete.

5. On the Table Options form, select Run Currency Triggers, if appropriate, and click Next.

Select this option if a JD Edwards EnterpriseOne table included in the table conversion contains currency triggers. If a table contains currency fields and you do not select this option, the system cannot determine where the decimal should be placed within a field. If you do not select the currency trigger option and the source or destination fields are currency fields used in a calculation, you might receive unexpected results.

Do not select Run Currency Triggers if the input and output data sources are the same type (for example, Oracle, IBM i, or SQL Server) and no calculations are being performed. Selecting this option results in slower processing.

You should not use currency triggers for an environment that has a path code that is different from the sign-in environment.

6. On the Data Selection form, define data selection as appropriate.

7. Select the Row Fetched event when deleting rows; otherwise, no records are deleted.

When you run the table conversion, the system fetches the rows one at a time, runs the conversion for each row, and deletes the record from the input.

8. Click Advanced ER to insert additional event rule logic.

9. Verify that the Delete All Selected Records option is selected, and click Next.

This option inserts the Delete Current Input Row system function into event rules.

10. On the Logging Options form, select logging options, as appropriate, and click Next.

11. On the Finish form, select a version option, and click Finish.

Note: If you change the table, the system warns you that deleting tables removes all mappings from the table conversion.
3.9 Using Event Rules in Table Conversions

This section provides an overview of event rules in table conversions and discusses how to create event rules in table conversions.

3.9.1 Understanding Event Rules in Table Conversions

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

You attach event rules to an event, such as:

- Process Begin
- Row Fetched
- Format Fetched
- Process End

Event rules in table conversions include system functions that are specific to JD Edwards EnterpriseOne Table Conversion Design Aid.

3.9.1.1 Table Conversion System Functions

This table describes each system function that you can use in table conversion event rules:

<table>
<thead>
<tr>
<th>System Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CopyTableDataSource</td>
<td>Enables you to copy a table or range of tables from one data source to another. The system copies tables based on specifications in the sign-in environment.</td>
</tr>
<tr>
<td>CopyTableEnvironment</td>
<td>Enables you to copy a table or range of tables from one environment to another. The system copies tables based on specifications in the input and output environments. If the specifications differ, the system performs a map-and-drop. This means that it creates a mapping between like fields in the source and destination tables and ignores all other fields.</td>
</tr>
<tr>
<td>TCInsertRow</td>
<td>The table conversion system inserts this system function when you select the Issue a write for this event? option and it cannot be moved. This function instructs the system that data should be written to the output table.</td>
</tr>
<tr>
<td>UserInsertRow</td>
<td>Enables you to specify when and where a row is inserted into the specified output table.</td>
</tr>
<tr>
<td>DeleteCurrentInputRow</td>
<td>Enables you to delete the current record from the input table.</td>
</tr>
<tr>
<td>UpdateCurrentInputRow</td>
<td>Enables you to update the current record in the input table after it has been modified.</td>
</tr>
<tr>
<td>SetSelectionAppendFlag</td>
<td>Enables you to determine whether selection criteria that is added by the system function SetUserSelection on the input table should be appended to the existing selection criteria or if it should replace the existing selection criteria.</td>
</tr>
</tbody>
</table>
3.9.2 Creating Event Rules in Table Conversions

Access a table conversion form that includes an Advanced ER button.

1. From the Events drop-down list, select the event to which you want to attach event rules, and click Advanced ER.

2. On the Event Rules Design form, select any of these options to define specific business logic:
   - Assignment
     Select this option to assign a fixed value, a field, or a mathematical expression to a field or variable.
   - If\While
     Select this option to create If and While logic statements, which are conditional instructions for event rule logic.
   - Business Function
     Select this option to attach an existing JD Edwards EnterpriseOne business function to an event. Business functions can retrieve a next number value for a new customer or convert Julian calendar dates to month, day, and year.
   - System Function
     Select this option to attach an existing JD Edwards EnterpriseOne system function, such as Copy Table Environment or User Insert Row.
   - Variables
     Select this option to create event rule variables, using existing data items, to store values for additional processing. Event rule variables can be used to accumulate totals, record the number of records that are read, and so on.
   - Else
     Select this option to create Else logic statements. An Else statement is automatically inserted after an If statement.
   - Table I/O
     Select this option to open tables in the input, output, or sign-in environment. Table I/O enables you to read data from tables other than the input table and use the data to create an output record. For example, suppose that you create a table conversion that loops through records in the F0101 table. The table conversion copies the records to another table, loops back through the records.
Using Event Rules in Table Conversions

to find each customer that has a specific employee as a contact, and copies the
information to the output table.

- Report Interconnect

Select this option to connect a batch process or report to the table conversion.

3. Define the logic, save the event rules, and quit Event Rules Design.

See Also:

- "Working with Table I/O" in the JD Edwards EnterpriseOne Tools Table Design Guide.
- "Using Business Functions" in the JD Edwards EnterpriseOne Tools APIs and Business Functions Guide.
This chapter contains the following topics:

- Understanding Table Conversion Processing
- Submitting Table Conversions
- Testing Table Conversions

### 4.1 Understanding Table Conversion Processing

You run a table conversion by submitting an associated batch version.

To track the table conversion process, you can use the tracing feature. The tracing feature writes the details of the table conversion process to a log. You can define the trace level to control the level of detail that is included in the log. When you test a table conversion, you can select to process the conversion one row at a time, which enables you to isolate problems or unexpected results.

### 4.2 Submitting Table Conversions

This section provides overviews of submitting table conversions and overriding table conversion properties and discusses how to:

- Override table conversion properties at runtime.
- Override table conversion properties of new versions.
- Override table conversion locations.
- Run table conversions on the web client.

### 4.2.1 Understanding Submitting Table Conversions

When you submit versions of table conversions, you can override table conversion properties, such as:

- Input and output environments.
- Trace level.
- Location where the table conversion is processed.

Overriding the location where you process the table conversion enables you to process the table conversion even if the server that you normally use is inoperable.
4.2.2 Understanding Overriding Table Conversion Properties

When you submit versions of table conversions, you can override these properties:

<table>
<thead>
<tr>
<th>Properties</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Environments</td>
<td>Enables you to change the location of the input or output data.</td>
</tr>
<tr>
<td>Data selection</td>
<td>Enables you to limit the number of input records.</td>
</tr>
<tr>
<td>Table options</td>
<td>Enables you to control how the JD Edwards EnterpriseOne Table Conversion system processes records during the conversion.</td>
</tr>
<tr>
<td>Logging options</td>
<td>Enables you to modify logging options.</td>
</tr>
<tr>
<td>Debug logging</td>
<td>Enables you to override the setting for debug logging as defined in the jde.ini.</td>
</tr>
</tbody>
</table>

### 4.2.2.1 Table Conversions on the Web Client

Submitting versions of table conversions from the web client is different from submitting them from the Microsoft Windows client. From the web client, you cannot modify properties. You can select and modify processing options and you can select these options:

- Override Location
- Override Job Queue
- Submit Version Specifications Only
- Data Selection


### 4.2.3 Forms Used to Submit Table Conversions

<table>
<thead>
<tr>
<th>Form Name</th>
<th>FormID</th>
<th>Navigation</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work With Batch Versions - Available Versions</td>
<td>W98305A</td>
<td>EnterpriseOne Life Cycle Tools, Report Management (GH9111), Batch Versions</td>
<td>Enter a table conversion template, and select a version.</td>
</tr>
<tr>
<td>Table Conversion Prompting</td>
<td>W98305D</td>
<td>Highlight a version on the Work With Batch Versions form, and click Select.</td>
<td>Work with versions.</td>
</tr>
</tbody>
</table>
4.2.4 Overriding Table Conversion Properties at Runtime

Access the Work With Batch Versions - Available Versions form.

<table>
<thead>
<tr>
<th>Form Name</th>
<th>FormID</th>
<th>Navigation</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Properties</td>
<td>NA</td>
<td>Select the Properties option on the Table Conversion Prompting form, and click Submit.</td>
<td>Review and override the environments, data selection, table options, and logging options that are specified within the conversion.</td>
</tr>
<tr>
<td>JD Edwards Data Sources</td>
<td>W98305B</td>
<td>Select the Override Location option on the Table Conversion Prompting form, and click Submit.</td>
<td>Select a data source to use as an override location.</td>
</tr>
</tbody>
</table>

1. Enter the name of the table conversion template in the Batch Application field, and click Find.

   The name of the Batch Application field changes to Table Conversion.

   You can also use the visual assist in the Batch Application field to search for the correct table conversion.

2. Select a batch version, and click Select.

3. On the Table Conversion Prompting form, select Properties, and click Submit.

4. On the Properties form, review and override the environments, data selection, table options, and logging options that specified in the table conversion.

   These forms are similar to the forms in Table Conversion Design Aid.

5. Select the Debug Logging tab to perform these functions:

   - Select the Use ini settings for trace level and number of rows to process option to use the jde.ini settings for the trace level and row-by-row conversion process.

     This option ensures that the system uses the settings that are defined in the jde.ini instead of the values that are entered in the Trace Level and Number of Rows fields.

   - Clear the Use ini settings for trace level and number of rows to process option to override the trace level in the jde.ini. Enter a value from 0 to 10 in the Trace Level field.

   - Clear the Use ini settings for trace level and number of rows to process option to convert a specific number of records (for example, if you want to test the table conversion). Enter the number of rows to process in the Number of Rows field.

     When you enter 0 in the Number of Rows field, the system processes all rows.

---

Note: When you override table conversion properties at runtime, the overrides are for that single process only; they are not saved with the version.
This option corresponds to the StopAfterRow setting in the jde.ini file. If you enter a value here, you override any specifications in the jde.ini.

6. Click OK to save the changes to the version.
The system processes the table conversion.

See Also:
- Testing Table Conversions.

4.2.5 Overriding Table Conversion Properties of New Versions

Access the Work With Batch Versions - Available Versions form.

---

**Note:** If you override table conversion properties from the Tools tab, the new properties are saved with the version.

---

1. Enter the name of the table conversion in the Batch Application field, and click Find.
The name of the Batch Application field changes to Table Conversion.
You can also use the visual assist in the Batch Application field to search for the correct table conversion.

2. Add a new version of the table conversion, complete the version information, and click OK.

3. On the Table Conversion Version Design form, select the Tools tab, and click Properties.

4. On the Properties form, review and override the environments, data selection, table options, and logging options that are specified in the table conversion.
These forms are similar to the forms in Table Conversion Design Aid.

5. To use debug logging, select the Debug Logging tab to perform these functions:
   - Select the Use ini settings for trace level and number of rows to process option to use the jde.ini settings for the trace level and row-by-row conversion process.
     This option ensures that the system uses the settings that are defined in the jde.ini instead of the values that are entered in the Trace Level and Number of Rows fields.
   - Clear the Use ini settings for trace level and number of rows to process option to override the trace level in the jde.ini. Enter a value from 0 to 10 in the Trace Level field.
   - Clear the Use ini settings for trace level and number of rows to process option to convert a specific number of records (for example, if you want to test the table conversion). Enter the number of rows to process in the Number of Rows field.
     When you enter 0 in the Number of Rows field, the system processes all rows.
     This option corresponds to the StopAfterRow setting in the jde.ini file. If you enter a value here, you override any specifications in the jde.ini file.
4.2.6 Overriding Table Conversion Locations
Access the Work With Batch Versions - Available Versions form.

1. Enter the name of the table conversion in the Batch Application field, and click Find.
   The name of the Batch Application field changes to Table Conversion.
   You can also use the visual assist in the Batch Application field to search for the correct table conversion.

2. Select a batch version, and click Select.

3. On the Table Conversion Prompting form, select Override Location, and click Submit.

4. On the JD Edwards Data Sources form, select the data source to use, and click Select.

4.2.7 Running Table Conversions from the Web Client
From EnterpriseOne Menu, select Submit Job to access the Submit Job-Work With Batch Versions-Available Versions form.

1. Enter the name of the table conversion in the Batch Application field, and click Find.
   The name of the Batch Application field changes to Table Conversion.

2. Select a batch version, and click Select.

3. On the Table Conversion Prompting form, select from these override options, and click Submit:
   - Override Location
   - Override Job Queue
   - Submit Version Specifications Only
   - Data Selection

4. Modify the appropriate property, and submit the batch version.

See Also:
- "Submitting Table Conversions from the Web Client" in the JD Edwards EnterpriseOne Tools Batch Versions Guide.

4.3 Testing Table Conversions
This section provides an overview of table conversion testing and discusses how to:

■ Set the trace level for debug logging on the workstation.
■ Set the trace level for debug logging on the server.
■ Force row-by-row conversion.

4.3.1 Understanding Table Conversion Testing
You can test table conversions in proof mode to ensure that they are processed as expected without errors. You can log debug information about the conversion while it
is being processed. You can also force the conversion to process one row at a time, which is useful if the conversion normally runs as an insert-from-select.

To log debug information for table conversions, enable tracing and set a trace level in the jde.ini. Use a value in the range of 0 to 10 for the trace level. These values indicate the level of detail that you want to include in the log. If necessary, you can override the jde.ini settings on the batch version of a table conversion.

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
- Number of rows inserted

If you set the trace level at 10, the system logs all information about every column in every format, including:

- User-defined formats.
- Processing options that are associated with the table conversion.
- All other information that is involved in the table conversion process.

The higher that you set the trace level, the more details the system provides regarding the table conversion process.

You can force row-by-row processing on the workstation or on the server when you want to test the table conversion. You can also set a trace level on the workstation or on the server, depending on where you process the table conversion. You can specify the number of rows to process in combination with forcing a row-by-row conversion.

### 4.3.1.1 Difference Between Logging Options and Debug Logging

Logging options, which you define when you set up a table conversion, can log:

- All errors that occur during the table conversion.
- All records that are copied, deleted, or updated.
- Details of copy table actions.

Debug logging logs more detailed information about the table conversion. This information enables you to determine the exact area in the table conversion where errors occurred.

---

**Note:** If you modify the debug logging in the batch version, you override the settings in the jde.ini.

---

### 4.3.1.2 Trace Levels

You can define a trace level for debug logging either on a workstation or on a server, depending on where you run the table conversion.

This table describes the logging information for each trace level:
4.3.2 Setting the Trace Level for Debug Logging on the Workstation

To define the trace level for debug logging on the workstation.

1. Open the jde.ini on the workstation.

2. To enable tracing and to set the trace level on a workstation, add this new information to the [TCEngine] section of the jde.ini:

   ```ini
   [TCEngine]
   _continue=
   TraceLevel=n
   ```

   where n is a number from 0 through 10.

3. To enable tracing and to set the trace level on a workstation, modify this information in the [Debug] and [UBE] sections of the jde.ini:

   ```ini
   [Debug]
   Output=File
   ```

   ```ini
   [UBE]
   UBE_saveLogFile=1
   ```

---

**Important:** Do not set the trace level at 10 when running a table conversion on tables that contain large amounts of data. The system writes the data to the server, and large amounts of data can cause the server to run out of disk space. You can, however, specify the number of rows to process on the batch version properties form.

---

### Trace Level | Logging Information

<table>
<thead>
<tr>
<th>Trace Level</th>
<th>Logging Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1</td>
<td>Logs general information about the conversion, such as name, inputs, outputs, event rule logic, and how many rows were inserted.</td>
</tr>
<tr>
<td>Level 2</td>
<td>Logs function call traces, such as starting conversion, ending conversion, and inserting rows. Also includes all level 1 information.</td>
</tr>
<tr>
<td>Level 3</td>
<td>Logs the points at which event rules are processed. Also includes all level 1 and level 2 information.</td>
</tr>
<tr>
<td>Level 4</td>
<td>Not applicable.</td>
</tr>
<tr>
<td>Level 5</td>
<td>Logs the points at which jdeCallObject is processed, such as calls to business functions from event rules. Also includes all level 1, level 2, and level 3 information.</td>
</tr>
<tr>
<td>Levels 6-8</td>
<td>Not applicable.</td>
</tr>
<tr>
<td>Level 9</td>
<td>Logs the content of columns during input, the processing of event rules, and the content before output. Also includes all level 1, level 2, level 3, and level 5 information.</td>
</tr>
<tr>
<td>Level 10</td>
<td>Logs all information for all applicable levels.</td>
</tr>
</tbody>
</table>
4.3.3 Setting the Trace Level for Debug Logging on the Server

To define the trace level for debug logging on the server:

1. To enable tracing and to set the trace level on a server, set the following information in the [TCEngine] section of the server jde.ini:

   
   [TCEngine]

   TraceLevel=n

   where n is a number from 0 to 10.

2. To enable tracing and to set the trace level on a server, modify this information in the [Debug] section of the server jde.ini:

   
   [Debug]
   Output=File
   KeepLogs=1

4.3.4 Forcing Row-By-Row Conversion

To force row-by-row conversion and specify the number of rows to process:

1. Open the jde.ini file.

2. To force row-by-row conversion, add this new information to the [TCEngine] section of the jde.ini:

   
   [TCEngine]
   ForceRowByRow=1
   _Continue -

3. To specify the number of rows to process, add this new information to the [TCEngine] section of the jde.ini:

   
   StopAfterRow=n

   where n is the number of rows that you want to process.
Preparing Foreign Tables for Table Conversion

This chapter contains the following topics:

- Understanding Foreign Tables
- Adding Data Sources
- Adding Environments
- Setting Up Default OCM Mappings

5.1 Understanding Foreign Tables

Foreign tables are text files, or any other files or tables, that are not recognized by Oracle’s JD Edwards EnterpriseOne software. However, they must reside in a database that is supported by JD Edwards EnterpriseOne, such as:

- Oracle
- Access
- IBM i
- SQL Server

Before you can work with foreign tables in Oracle’s JD Edwards EnterpriseOne Table Conversion Design Aid, you must define the database to JD Edwards EnterpriseOne. This definition is created when you set up an ODBC data source or an Oracle Call Interface (OCI) data source that points to the database where the foreign tables reside.

You must also add a data source in JD Edwards EnterpriseOne that points to the ODBC or OCI data source that you defined. For each ODBC data source, database instance, or library that contains foreign tables, you must set up an environment. The environment points to the JD Edwards EnterpriseOne data source, which in turn points to the database or library. In addition, you must set up an ODBC mapping from the data source to the environment.
Note: When you work with foreign tables, you must discuss database permissions with a database administrator. Your JD Edwards EnterpriseOne user ID (or, if you are using the JD Edwards EnterpriseOne security server feature, the database user to which it maps) might need to be changed to give you permissions to use the tables in the foreign database. Without these permissions, you cannot view the foreign tables in JD Edwards EnterpriseOne Table Conversion Design Aid. Under certain conditions, the table conversion engine must create temporary tables in the output environment and requires create-and-drop permissions for the database.

5.2 Adding Data Sources

This section discusses how to add Oracle's JD Edwards EnterpriseOne data sources.

5.2.1 Forms Used to Add Data Sources

<table>
<thead>
<tr>
<th>Form Name</th>
<th>FormID</th>
<th>Navigation</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine Search &amp; Select</td>
<td>W986115E</td>
<td>EnterpriseOne Life Cycle Tools, System Administration Tools, System Administration Tools (GH9011), Data Source Management, Database Data Sources</td>
<td>Select where the data source resides.</td>
</tr>
<tr>
<td>Work With Data Sources</td>
<td>W986115A</td>
<td>Select the appropriate machine on which the data source resides, and then click Select on the Machine Search &amp; Select form.</td>
<td>Add a data source, or select a data source to modify.</td>
</tr>
<tr>
<td>Data Source Revisions</td>
<td>W986115O</td>
<td>Click Add on the Work With Data Sources form.</td>
<td>Add JD Edwards EnterpriseOne data sources. Enter data source name, use, type, data class, platform, server name, database name, and owner.</td>
</tr>
</tbody>
</table>

5.2.2 Adding Data Sources

Access the Data Source Revisions form.
Data Source Name
Enter a unique name that identifies the data source.

Data Source Use
Select a user-defined code (UDC) (98|SU) that indicates how the data source is configured. Select Servers (SVR) to run UBEs and business functions. Select Database (DB) to access table data.

Only database data sources are used to access data in tables.

Data Source Type
Select a UDC (98|DD) that indicates the type of database.

Data Class
Select a UDC (H96|DU) that describes the type of tables in the database, such as business data, control tables, and data dictionary.

Platform
Select a UDC (H93|PL) that indicates the type of physical hardware that the database resides on, such as IBM Power Systems, Linux, and Sun Microsystems.

Database Server Name
Select the enterprise server where the database or file resides.
Object Owner ID
Enter the database table prefix or owner. This field is available for these data source types:
- SQL Server OLEDB
- SQL Server ODBC
- SSE/ODBC
- MSDE/ODBC
- MSDE/OLEDB
- OEE
- Oracle

Database Name
Enter the name that was assigned to the database during installation, such as HPDEVORAP or HP9000.
How this field is used depends on the data source type. If an Oracle data source is added, then this field contains the Oracle connect string. If a Microsoft Access, SQL*Server, or Client Access data source is added, then this field contains the Windows ODBC data source name. To minimize the number of connections to SQL*Server, the ODBC data source name should be the machine name where the SQL*Server database resides and the catalog name should be defined using the individual database name. For example, if you have two databases, DatabaseA and DatabaseB, on a machine called INTELNT, this field should contain INTELNT and the catalog name should be set to DatabaseA for one data source and DatabaseB for the other data source.
The data source name can be different from the ODBC or Oracle database name, if necessary.

Database Instance
Enter the name of the server instance. This name is required for these data source types:
- SQL Server OLEDB
- SQL Server ODBC
- SSE/ODBC
- MSDE/ODBC
- MSDE/OLEDB

ODBC Data Source Name
Enter the name that is assigned to the database during installation, such as HPDEVORAP or HP9000. This field is available for these data source types:
- Access
- SQL Server ODBC
- DB2 UDB on OS/390
- DB2 for i
- SSE/ODBC
- MSDE/ODBC
5.3 Adding Environments

The easiest way to add an environment to JD Edwards EnterpriseOne is to copy an existing environment.

This section discusses how to add JD Edwards EnterpriseOne environments.

5.3.1 Forms Used to Add Environments

<table>
<thead>
<tr>
<th>Form Name</th>
<th>FormID</th>
<th>Navigation</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy Environment</td>
<td>W0094B</td>
<td>Click an environment to copy, and select Copy Environment from the Row menu on the Work With Environments form.</td>
<td>Enter a name for the new environment.</td>
</tr>
</tbody>
</table>

5.3.2 Adding Environments

Access the Work With Environments form.

1. Click the environment that most closely matches the environment that you want to create.

   For example, you might click the environment that you are signed in to or any other environment that you can access from the workstation.

2. From the Row menu, select Copy Environment.
3. On the Copy Environment form, enter a name for the new environment in the New Environment field.

4. To copy only the *PUBLIC OCM mappings of an environment, select Copy *PUBLIC Records Only and click OK.

The Copy *PUBLIC Records Only option is selected by default. Clear the option to copy all mappings for the environment, including individual users and *PUBLIC.

See Also:

5.4 Setting Up Default OCM Mappings

This section provides an overview of OCM mappings and discusses how to set up default OCM mappings.

5.4.1 Understanding OCM Mappings

You must map all JD Edwards EnterpriseOne objects, including tables, by environment. You can map all objects of a specific object type using a single map, or you can define individual mappings by object name. When you define a default OCM mapping, you can select an existing environment and map the objects for that environment to the data sources where those objects exist.

You must create a default map for the table (TBLE) object type. Create this mapping with:

- Literal value of DEFAULT as the object name.
  
  This option enables you to map all tables in an environment to a specific data source.

- TBLE as the object type.

You must also indicate the data source where this table object resides. When you create a default map for the TBLE object type, all table objects point to the default data source unless the table has its own specific mapping. JD Edwards EnterpriseOne Table Conversion Design Aid uses this mapping for foreign tables.

To map a specific TBLE object type to a data source that is different from the DEFAULT, enter the table name as the object name. For example, F0002 is mapped to Control Tables-Test while the DEFAULT table mapping is mapped to Business Data-Test. The system reads the DEFAULT mapping for all tables except the F0002 table.

Each environment must have a default map for table objects for the *PUBLIC user profile because no inherent default location exists for table objects. If table objects do not have a default map and are not explicitly mapped by name, JD Edwards EnterpriseOne produces a Select/Failed error message when it tries to access the tables. Additionally, the tables do not appear in the input or output forms in JD Edwards EnterpriseOne Table Conversion Design Aid.
### 5.4.2 Forms Used to Set Up Default OCM Mappings

<table>
<thead>
<tr>
<th>Form Name</th>
<th>FormID</th>
<th>Navigation</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine Search and Select</td>
<td>W986110D</td>
<td>EnterpriseOne Life Cycle Tools, System Administration Tools, System Administration Tools (GH9011), Data Source Management, Object Configuration Manager</td>
<td>Select a data source.</td>
</tr>
<tr>
<td>Work With Object Mappings</td>
<td>W986110B</td>
<td>Select the appropriate machine on which the data source resides, and then click Select on the Machine Search &amp; Select form.</td>
<td>Add, modify, and copy mappings.</td>
</tr>
<tr>
<td>Object Mapping Revisions</td>
<td>W986110C</td>
<td>Click Add on the Work With Object Mappings form.</td>
<td>Enter the environment name, object name, primary data source, system role, object type, and data source mode for a new mapping, and indicate whether to allow QBE (query by example) for the mapping.</td>
</tr>
</tbody>
</table>

### 5.4.3 Setting Up Default OCM Mappings

Access the Object Mapping Revisions form.
Figure 5–2  Object Mapping Revisions form

- **Environment Name**: Select the name of the environment where the tables reside.

- **Object Name**: Enter DEFAULT or the name of the specific object that you are mapping. The value DEFAULT creates a default map for all objects of a specific type.

- **Primary Data Source**: Select the name of the primary data source.

- **System Role**: Enter *PUBLIC, or select the JD Edwards EnterpriseOne user or role that had permissions to use these mappings. *PUBLIC gives all users permissions.

- **Object Type**: Select a UDC (98|OM) that indicates the type of object for which you are creating a mapping.

- **Data Source Mode**: Select a UDC (98|DM) that indicates whether the primary or secondary data source should be used.

- **Secondary Data Source**: Select the name of the secondary data source. This field is available only when secondary has been selected for the data source mode. This data source is used if the primary data source or the data item in the primary data source cannot be located.
Allow QBE (query by example)
Select a UDC (98|QB) that indicates whether row-level record locking is selected for the data source. Row-level record locking should be selected to help prevent database integrity issues.

See Also:
- "Understanding Object Configuration Manager" in the *JD Edwards EnterpriseOne Tools Configurable Network Computing Implementation Guide*.
- "Understanding Data Sources" in the *JD Edwards EnterpriseOne Tools Configurable Network Computing Implementation Guide*. 
batch delete
A type of batch process that enables the deletion of records from a table or business view.

business view
A JD Edwards EnterpriseOne object that defines the relationship between two or more tables and joins the data into a single view. You can use JD Edwards EnterpriseOne business views for input to a table conversion, not for output.

control tables
Tables that contain UDCs, menus, and data dictionary items.

data conversion
A type of batch process that enables the transfer or copying of data from an input table or business view into output tables using the logic that is necessary to perform the transfer. You can also use Data Conversion to update records in a table or business view.

data copy
A type of batch process that enables the copying of tables from one data source or environment to another data source or environment when the tables are identical.

data copy with table input
A type of batch process that enables the copying of tables based on information from an input table. For example, the input table might provide information about which tables are copied, where they are copied, and so on.

foreign tables
Tables that do not have a JD Edwards EnterpriseOne definition but reside in a database that is supported by JD Edwards EnterpriseOne. You must set up a data source and environment in JD Edwards EnterpriseOne to point to the location of foreign tables.

JD Edwards EnterpriseOne tables
Tables that exist in the JD Edwards EnterpriseOne database and can be accessed from Object Management Workbench. You can design and modify them using Table Design Aid. At design time, only the specifications are needed to reference the table. At runtime, the table must be generated. An instance of the table must occur in a particular database.