



IBM Software Group

IBM WebSphere® Data Interchange V3.3

Using Qualification in Source Based Maps



@business on demand.

© 2007 IBM Corporation

This presentation will demonstrate how to use Element, Segment, and Loop Qualifiers in a Data Transformation Source Based Map.

Agenda

- Review common mapping commands
- Define Qualification
- Demonstrate Qualification types
- Summary and references



The presentation will give a review of common mapping commands, define qualification, and demonstrate the qualification types for Data Transformation Source Based maps.

Using Element, Segment and Loop Qualifiers

- Common Data Transformation mapping functions.
 - ▶ Drag/Drop - Map the association from the source compound or simple element to a target.
 - ▶ Assignment – Assign a value to a target
 - ▶ Conditional mapping – If / Elself / Else / Endlf
 - ▶ SetElementAttribute() - &ZEROSIG and left/right pad and adjust.
 - ▶ **Loop Qualification - Multiple Occurrence, Occurrence, Value**
 - ▶ Translation Table - apply value conversions
 - ▶ Validation Table (Code List) - apply validation to values
 - ▶ Use of Variables – Global and Local



Some common Data Transformation (DT) mapping functions include: drag/drop, supplying literal values not found in the data, conditional mapping, formatting and validating values, and loop qualification. This presentation will review how to use qualification.

Using Element, Segment and Loop Qualifiers

- What is qualification
 - ▶ Within your document, certain elements and groups of elements can repeat. When you map an element that repeats within the source document (a compound or simple element), you must tell WebSphere Data Interchange which occurrence of the segment or loop you are using as well as the occurrence of the repeating simple element you are using. This is called *qualifying* the segment, element, or loop.



Within your document, certain elements and groups of elements can repeat. When you map an element that repeats within the source document (a compound or simple element), you must tell WebSphere Data Interchange (WDI) which occurrence of the segment or loop you are using as well as the occurrence of the repeating simple element you are using. This is called *qualifying* the segment, element, or loop.

Using Element, Segment and Loop Qualifiers

- Examples of Looping Structures

- ▶ EDI

- NAD Loop, PO1 Loop, HL Loop
 - REF repeating segment
 - repeating elements and composite elements

- ▶ Application data - Data Format

- Records
 - Structures within records
 - Loops

- ▶ XML:

- `<!ELEMENT XMLORDERS (RoutingInfo, BGM, DTM+, PAI?,
NAD009*,`

Examples of looping structures for Electronic Data Interchange (EDI) standard NAD or PO1 loop and the REF repeating segment. For application data, data format records, structures within records, and loops. And for XML the DTD element definition for compound elements.

Using Element, Segment and Loop Qualifiers

- WebSphere Data Interchange supports four types of qualification for send and receive maps.
 - ▶ Occurrence
 - ▶ Multiple occurrence
 - ▶ Value
 - ▶ Expression



WebSphere Data Interchange supports four types of qualification for maps. You can qualify by: Occurrence, multiple occurrence, Value, and Expressions.

Using Element, Segment and Loop Qualifiers

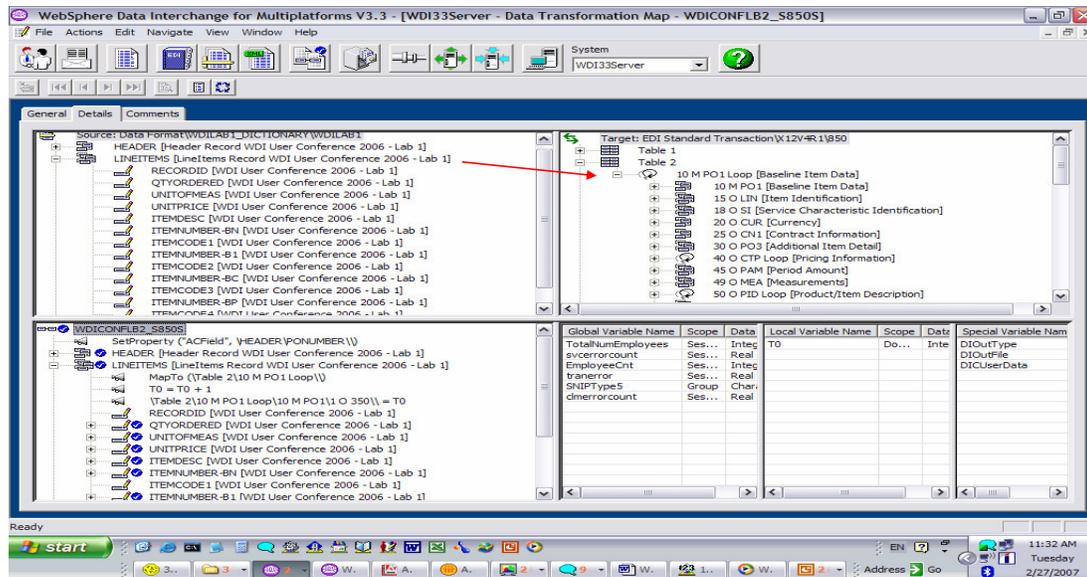
- Expressions are literals, elements and variables that are combined using operators
- Expression yield a single value
- Expressions can use
 - ▶ Arithmetic operators: +, -, *, /
 - ▶ Logical operators: AND OR NOT
 - ▶ Relational operators: EQ, NE, LT, GT, LE, GE
- Expressions can be arbitrarily complex
- Example: NumFormat(Quantity * Price, 2)



An expression can be as simple as a source document element, literal value, or a variable, but can contain any number of functions, operators, and delimiters. Expression yield a single value. Expressions can use Arithmetic operators, Logical operators, and Relational operators. Expressions can be arbitrarily complex.

Using Element, Segment and Loop Qualifiers

- Qualification (Multiple Occurrence) use drag /drop



Using Qualification Source Based Maps

Multiple Occurrence Qualification is the most common qualification used. Qualify by multiply occurrence when you want to create multiple elements in the target to correspond to repeating elements in the source using the same mapping instructions. All iterations of the repeating element that are not otherwise qualified are handled under the multiple occurrence qualification. To create a multiple occurrence qualification, select the repeating element from the source window (upper left window) or the mapping window (lower left window) and drag and drop to the repeating element in the target window (upper right window).

For example, say you are working on a data format source to EDI standard transaction target map and you find that the LIN loop in an EDI order repeats to handle multiple purchase-order line items. You need WebSphere Data Interchange to create a separate instance of the PO1 loop for each POLINEITEM record and when each occurrence of the PO1 loop is created the same mapping instructions should be executed. Consequently, you would qualify the PO1 loop by multiple occurrence. That way, WebSphere Data Interchange creates as many PO1 loops in your trading partner's transaction as there are occurrences of the POLINEITEM record in your application data.. The mapping commands under the repeating element are executed for each occurrence of the path used for the qualification.

To create a multiple occurrence qualification, select the repeating element from the source window (upper left window) or the mapping window (lower left window) and drag and drop to the repeating element in the target window (upper right window). You can also drag and drop from the target window to the source or mapping window.

Using Element, Segment and Loop Qualifiers

- Multi-Occurrence Qualification
 - ▶ Source-based maps – Mapto() command
 - The qualifications appear directly under the repeating source element.
 - ▶ Target-based maps - ForEach() command
 - Additional qualifications related to that source element appear under the For Each command.
 - ▶ The main difference is that because you can map more than one repeating source element to a repeating target element, the tree display has an extra level of hierarchy.
 - ▶ Implied “Close Occurrence” with each instance of the source or target path.



Source-based maps use the Mapto() qualification command, but target-based maps use the ForEach() qualification command. The main difference between multi-occurrence qualification in source-based and target-based maps is that because you can map more than one repeating source element to a repeating target element, the tree display has an extra level of hierarchy. In a source based map the qualifications appear directly under the repeating source element. In a target based map, each repeating source element associated with the repeating target element results in the creation of a ForEach command under the repeating target element. Additional qualifications related to that source element appear under the ForEach command. For Multi-Occurrence Qualification there is an implied “Close Occurrence” with each instance of the source or target path.

Section

Source Based Multiple Occurrence

Using Element, Segment and Loop Qualifiers

- Qualification (Multiple Occurrence) use drag /drop

Global Variable Name	Scope	Data	Local Variable Name	Scope	Data	Special Variable Name
TotalNumEmployees	Ses...	Integ	TO	Do...	Inte	DICOutType
svccerrorcount	Ses...	Real				DICOutFile
EmployeeCnt	Ses...	Integ				DICUserData
tranerror	Ses...	Real				
SNIPType5	Group	Char				
cmerrorcount	Ses...	Real				

Using Qualification Source Based Maps

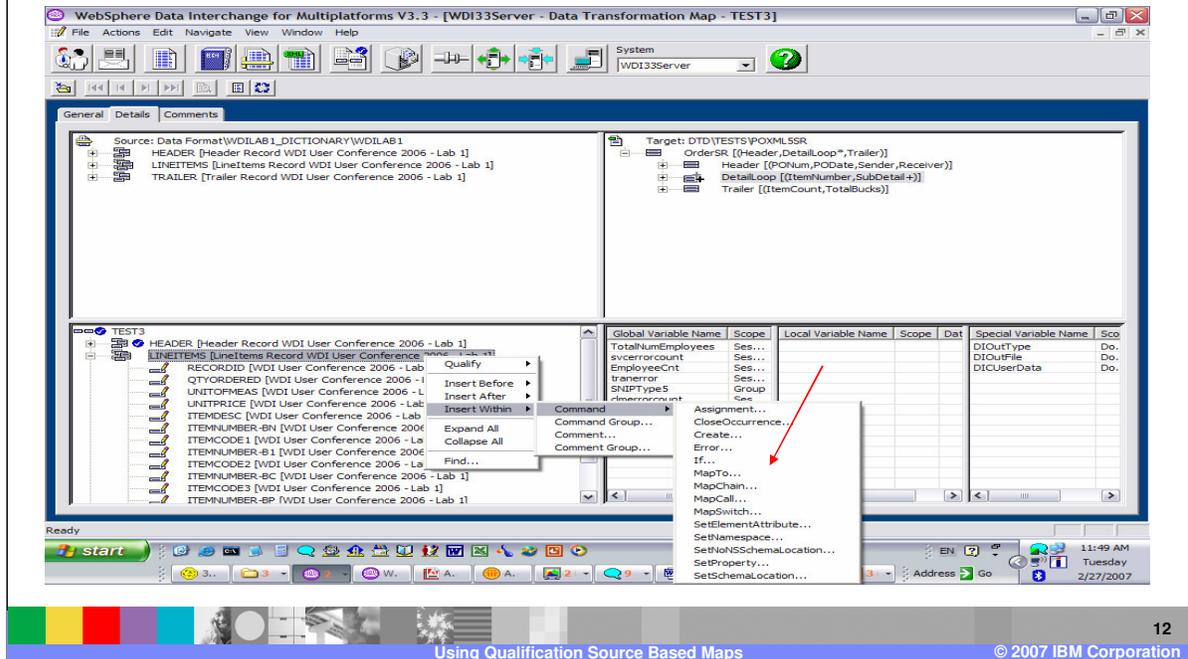
© 2007 IBM Corporation

11

Using the drag and drop operation, the result is a MapTo command in a source based map. Which tells the transformation execution to Map the data (where this command appears in the source document) To the “targetPath”. Since the MapTo command appears under the repeating source record LINEITEMS and the target path is PO1 Loop, each occurrence of the LINEITEMS record in the source will create a PO1 loop in the target and all mapping commands under this mapping will be executed for each occurrence of LINEITEMS.

Using Element, Segment and Loop Qualifiers

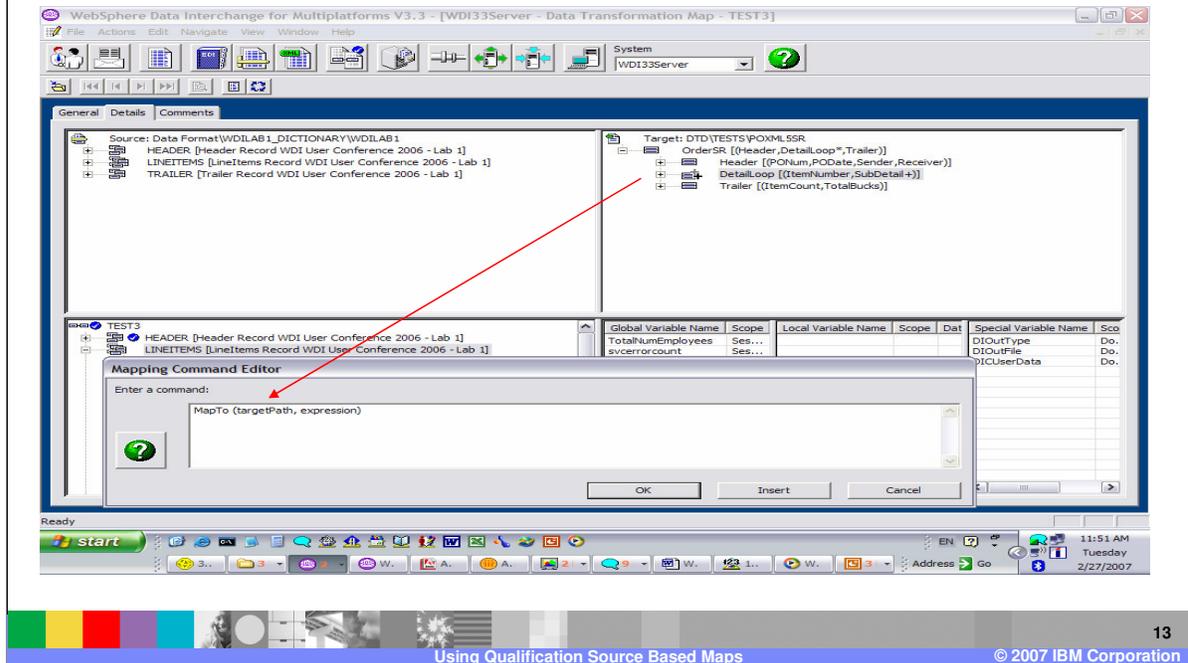
- Qualification (Multiple Occurrence) Command



You can also select the MapTo command in the mapping window. Right click on the source, select Insert within, command, and the MapTo command.

Using Element, Segment and Loop Qualifiers

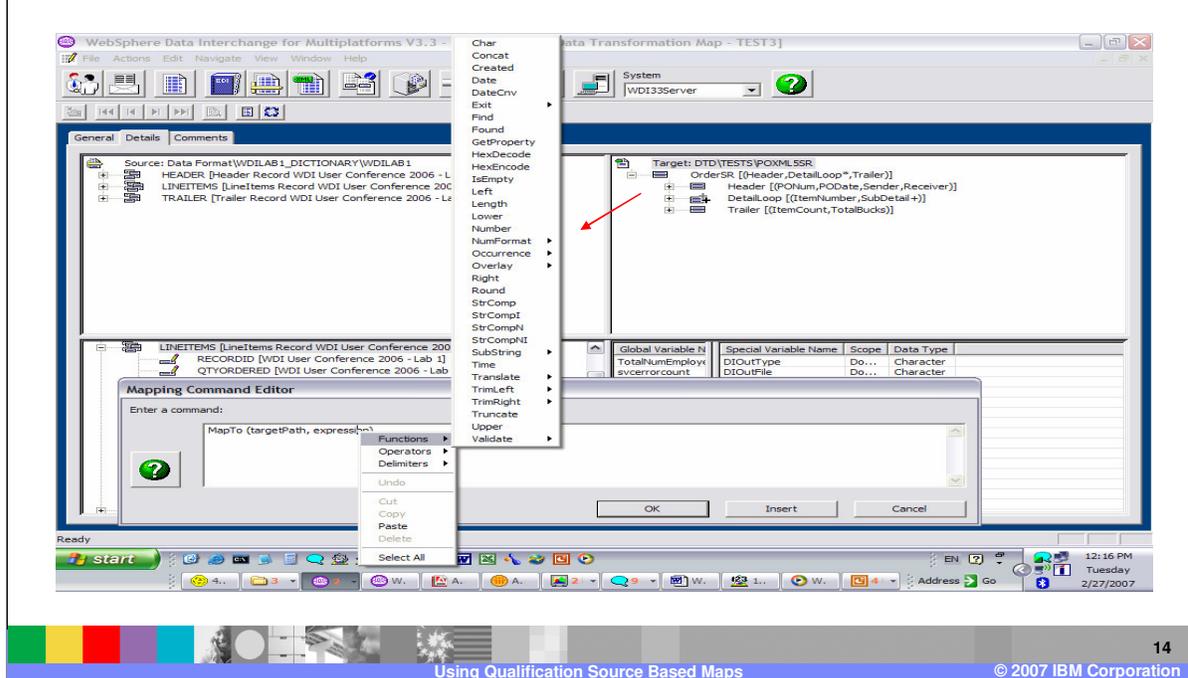
- Qualification (Multiple Occurrence) Command



The MapTo command has two parameters targetPath and expression. Expression is optional. The command is used to qualify loops and to move data values (Simple elements) to the target document. To add the target path, select the target and drag and drop onto the targetPath parameter. Then remove the expression parameter and click on the OK button. Both the target path and expression parameters can be a function or combination of functions.

Using Element, Segment and Loop Qualifiers

- Qualification (Multiple Occurrence) Command



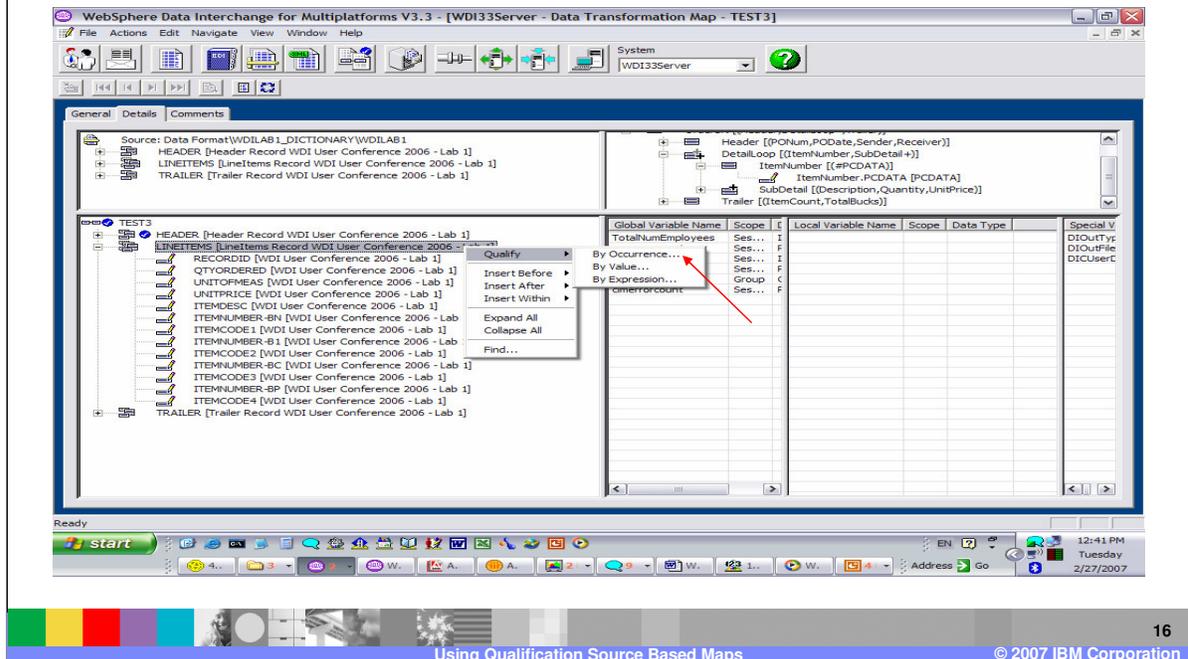
To select a function, right click on the targetPath or expression and select from the list of functions.

Section

Source Based Occurrence

Using Element, Segment and Loop Qualifiers

- Qualification (Occurrence)



Qualify an element by occurrence when a specific instance of a repeating element requires mapping instructions specific to that occurrence of the repeating element.

For example, say that you are working with a source document that has a buyers part number in the field ITEMNUMBER-BP in the first Occurrence of the LINEITEM record. The first occurrence of the DetailLoop compound element should contain the buyer's part number.

To add an Occurrence qualification, right click on the source element, choose Qualify, and select By Occurrence.

Using Element, Segment and Loop Qualifiers

- Qualification (Occurrence)

The screenshot displays the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window shows a Data Transformation Map configuration for a source named 'Data Format\WDLAB1_DICTIONARY\WDLAB1'. The source structure includes elements like HEADER, LINEITEMS, and TRAILER. The target structure includes elements like Header, DetailLoop, ItemNumber, SubDetail, and Trailer. A dialog box titled 'Qualify by Occurrence Number - LINEITEMS' is open, prompting the user to 'Enter the Occurrence Number'. A red arrow points to the input field. The background shows the map structure with source elements like HEADER, LINEITEMS, and TRAILER, and target elements like DetailLoop and SubDetail. A table of Global Variable Name, Scope, and Local Variable Name is also visible.

Global Variable Name	Scope	Local Variable Name	Scope	Data Type	Special V
TotalNumEmployees	Ses...				
svcerrorcunt	Ses...				
EmployeeCnt	Ses...				
trancorr	Ses...				
SNIPType5	Group				
elmerrorcunt	Ses...				

Map occurrence 1 to the appropriate target element using occurrence qualification.

Using Element, Segment and Loop Qualifiers

- Qualification (Occurrence)

WebSphere Data Interchange for Multiplatforms V3.3 - [WDI33Server - Data Transformation Map - TEST3]

Source: Data Format\WDILAB1_DICTIONARY\WDILAB1

- HEADER [Header Record WDI User Conference 2006 - Lab 1]
- LINEITEMS [LineItems Record WDI User Conference 2006 - Lab 1]
 - Qualify (Occurrence) EQ 1
 - RECORDID [WDI User Conference 2006 - Lab 1]
 - QTYORDERED [WDI User Conference 2006 - Lab 1]
 - UNITFORMEAS [WDI User Conference 2006 - Lab 1]
 - UNITPRICE [WDI User Conference 2006 - Lab 1]
 - ITEMDESC [WDI User Conference 2006 - Lab 1]
 - ITEMNUMBER-BN [WDI User Conference 2006 - Lab 1]
 - ITEMCODE1 [WDI User Conference 2006 - Lab 1]
 - ITEMNUMBER-B1 [WDI User Conference 2006 - Lab 1]
 - ITEMCODE2 [WDI User Conference 2006 - Lab 1]
 - ITEMNUMBER-BC [WDI User Conference 2006 - Lab 1]
 - ITEMCODE3 [WDI User Conference 2006 - Lab 1]
 - ITEMNUMBER-BP [WDI User Conference 2006 - Lab 1]
 - ITEMCODE4 [WDI User Conference 2006 - Lab 1]
- TRAILER [Trailer Record WDI User Conference 2006 - Lab 1]

Global Variable Name	Scope	Local Variable Name	Scope	Data Type	Special V
TotalNumEmployees	Ses...				DIOUtyr
svccerrorcount	Ses...				DIOUFile
EmployeeCnt	Ses...				DIOUserC
tranerror	Ses...				
SNIPTYPE5	Group				
dmererrorcount	Ses...				

Ready

Using Qualification Source Based Maps © 2007 IBM Corporation

This results in a Qualify command with the Occurrence function equal 1.

Using Element, Segment and Loop Qualifiers

■ Qualification (Occurrence)

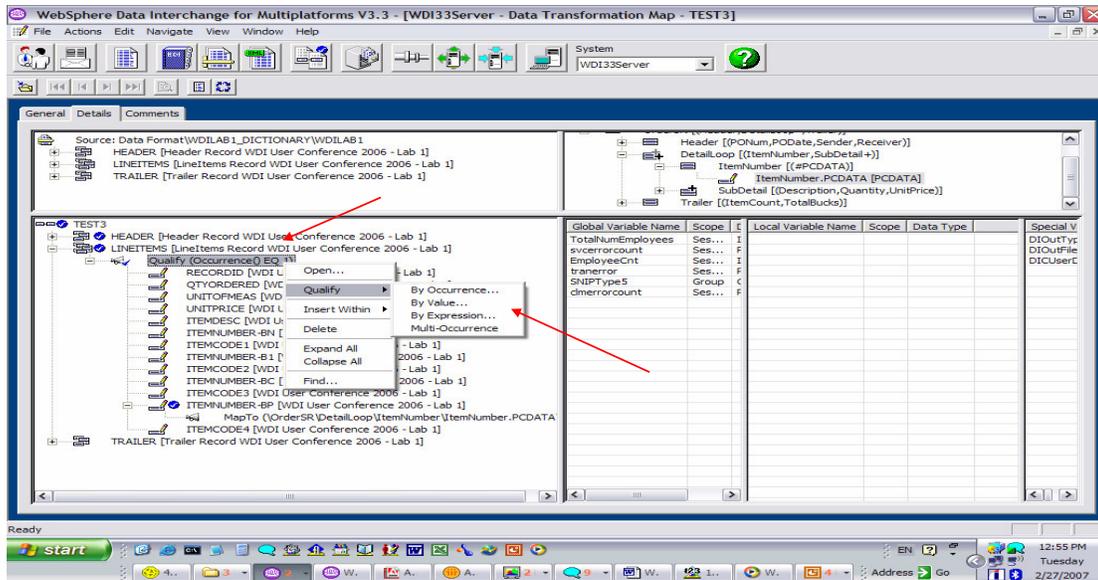
The screenshot displays the WebSphere Data Interchange for Multiplatforms V3.3.3 interface. The main window shows a Data Transformation Map configuration for 'TEST3'. The 'LINEITEMS' record is selected, and a context menu is open with 'Qualify' selected. A table on the right shows global and local variable definitions.

Global Variable Name	Scope	Local Variable Name	Scope	Data Type	Special V
TotalNumEmployees	Ses...		I		DICOutTyp
	Ses...		F		DICOutFile
	Ses...		F		DICUser
	Ses...		F		
	Group		C		
	SEG...		F		

Additional qualifications can be added for the LINEITEMS record using right click and selecting Qualify.

Using Element, Segment and Loop Qualifiers

■ Qualification (Occurrence)



20

Using Qualification Source Based Maps

© 2007 IBM Corporation

Additional qualifications can also be added under the current Qualification by using right click on the Qualification and selecting Qualify.

Using Element, Segment and Loop Qualifiers

- Qualification (Occurrence)

Global Variable Name	Scope	Data	Local Variable Name	Scope	Data Type	Spec
TotalNumEmployees	Ses...	Integ	TO	Do...	Integer	
sverrorcount	Ses...	Real				DICOut
EmployeeCnt	Ses...	Integ				DICOut
tranerror	Ses...	Real				DICOut
SNPType5	Group	Char				
clmerrorcount	Ses...	Real				

If a multiple occurrence qualification exists and you choose to add additional qualification for the record, this results in the multiple occurrence qualification becoming a “default” qualification.

Using Element, Segment and Loop Qualifiers

■ Qualification (Occurrence)

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3.3 interface. The main window displays a Data Transformation Map for 'WDI33Server - Data Transformation Map - WDICONFLB2_S850S'. The source is 'Data Format\WDILAB1_DICTIONARY\WDILAB1' and the target is '10 M PO 1 Loop [Baseline Item Data]'. The mapping window shows a 'Qualify (Occurrence) EQ 1' command being applied to the 'LINEITEMS' source element. A context menu is open over the 'ITEMCODE4' target element, with 'CloseOccurrence...' selected. The menu options include: Insert Before, Insert After, Insert Within, Find..., Command, Command Group..., Comment..., Comment Group..., Assignment..., CloseOccurrence..., Create..., Error..., If..., MapTo..., MapChain..., MapCall..., MapSwitch..., SetElementAttribute..., and SetProperty... The bottom of the window shows the Windows taskbar with the date 2/27/2007 and time 1:05 PM.

When doing Occurrence qualification with additional qualification, there should also be a CloseOccurrence command to close the target Occurrence after all the mapping has been executed. You can place a CloseOccurrence command anywhere in the mapping window. To place the command at the end of the Occurrence Qualification, right click the last source element, choose insert after, select Command, then CloseOccurrence. The close occurrence can also be placed as the first command before an qualification.

Using Element, Segment and Loop Qualifiers

- Qualification (Occurrence)

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3.3 interface. The main window displays a Data Transformation Map configuration for 'WDICONFLB2_S850S'. The map includes a source 'Data Format WDLAB1_DICTIONARY(WDLAB1)' and several target elements. A 'Qualify (Occurrence EQ 1)' element is highlighted, and a red arrow points from it to the 'targetPath' parameter in the 'CloseOccurrence (targetPath)' command entered in the Mapping Command Editor dialog. The dialog also shows 'OK', 'Insert', and 'Cancel' buttons. The system tray at the bottom indicates the date and time as Tuesday, 2/27/2007, 1:06 PM.

The targetPath parameter should be the target that was created by the Occurrence qualification.

Using Element, Segment and Loop Qualifiers

- Qualification (Occurrence)

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window displays a Data Transformation Map configuration for 'WDICONFLB2_S850S'. The 'Qualify (Occurrence) EQ 1' node is selected, and a red arrow points to the 'CloseOccurrence ((Table 2)\10 M PO 1 Loop\)' property. The 'Global Variable Name' table is visible on the right side of the interface.

Global Variable Name	Scope	DI	Local Variable Name	Scope	Data Type	Special Variab
TotalNumEmployees	Ses...	In	T0	Do...	Integer	
svccerrorcount	Ses...	Rx				DIOutType
EmployeeCnt	Ses...	In				DIOutFile
tranerror	Ses...	Rx				DIUserData
SNIPType5	Group	C				
dmerrorcount	Ses...	Rx				

This will close the occurrence for PO1 loop before executing the default qualification which is a multiple occurrence qualification. Remember that multiple occurrence qualifications have an implied close occurrence and do not need this command.

Section

Source Based Value

Using Element, Segment and Loop Qualifiers

- Qualification (Value)

Global Variable Name	Scope	Local Variable Name	Scope	Data	Spe
TotalNumEmployees	Ses...	Prod_ID_VAL	Do...	Char	DIO...
sverrorcount	Ses...	Prod_ID_Qual	Do...	Char	DIO...
EmployeeCnt	Ses...	TO	Do...	Intex	DICL
tranerror	Ses...				
SNIPTypeS	Group				
clerrorcount	Ses...				

Using Qualification Source Based Maps

© 2007 IBM Corporation

Qualify an element by value when you want the value of data in a simple element or variable to drive WebSphere Data Interchange's translation of a repeating element.

For example, say you want to qualify the N1 loop with the value of ST in Element 98, which is the "Entity Identifier Code," received in a purchase order to create a buyer record or to populate the buyer fields in a record. Further, say that you want the buyer's name to be mapped into the header record depending on the value in Element 98 of the N1 loop.

Using Element, Segment and Loop Qualifiers

- Qualification (Value)

Global Variable Name	Scope	D	Local Variable Name	Scope	Data	Spe
TotalNumEmployees	Ses...	I	Prod_ID_VAL	Do...	Char	DIO
sverrorcount	Ses...	R	Prod_ID_Qual	Do...	Char	DIO
EmployeeCnt	Ses...	I	TO	Do...	Integ	DICL
tranerror	Ses...	R				
SNIPType5	Group	C				
sverrorcount	Ses...	R				

To add a Value qualification, right click on the source element, choose Qualify, and select By Value.

Using Element, Segment and Loop Qualifiers

- Qualification (Value)

The screenshot shows the IBM WebSphere Data Interchange for Multiplatforms V3.3.3 interface. The main window displays a Data Transformation Map editor with a tree view of a map. The map structure includes loops and qualifiers. A 'Mapping Command Editor' dialog box is open, showing the command 'Qualify (StrComp (path, 'value') EQ 0)'. A red arrow points to the 'value' parameter in the command. The background shows a table with columns for Global Variable Name, Scope, Local Variable Name, Scope, Data, and Specialization.

Global Variable Name	Scope	Local Variable Name	Scope	Data	Spe
TotalNumEmployees	Ses...	Prod_ID_VAL	Do...	Char	DIOA
sverrorcount	Ses...	Prod_ID_Qual	Do...	Char	DIOA
EmployeeCnt	Ses...	TO	Do...	Inte	DIOA
transferor	Ses...				
SNIPType5	Group				

This results in a Qualify command with the StrComp function equal 0. The StrComp function contains two parameters path and a literal value. The path should be a source path but can be a variable or function. The literal value can also be a variable or function. Use right click on the parameter to select from a list of functions.

Using Element, Segment and Loop Qualifiers

- Qualification (Value)

The screenshot displays the WebSphere Data Interchange for Multiplatforms V3.3.3 interface. The main window shows a Data Transformation Map configuration. The left pane displays a tree view of the map structure, including a loop '310 O N1 Loop [Name]' with a qualification 'Qualify (StrComp ((Table 1{310 O N1 Loop{310 O N1{1 M 98\\, \"ST\"}) EQ 0))'. The right pane shows a list of global variables and local variables. The bottom status bar indicates 'Using Qualification Source Based Maps' and '© 2007 IBM Corporation'.

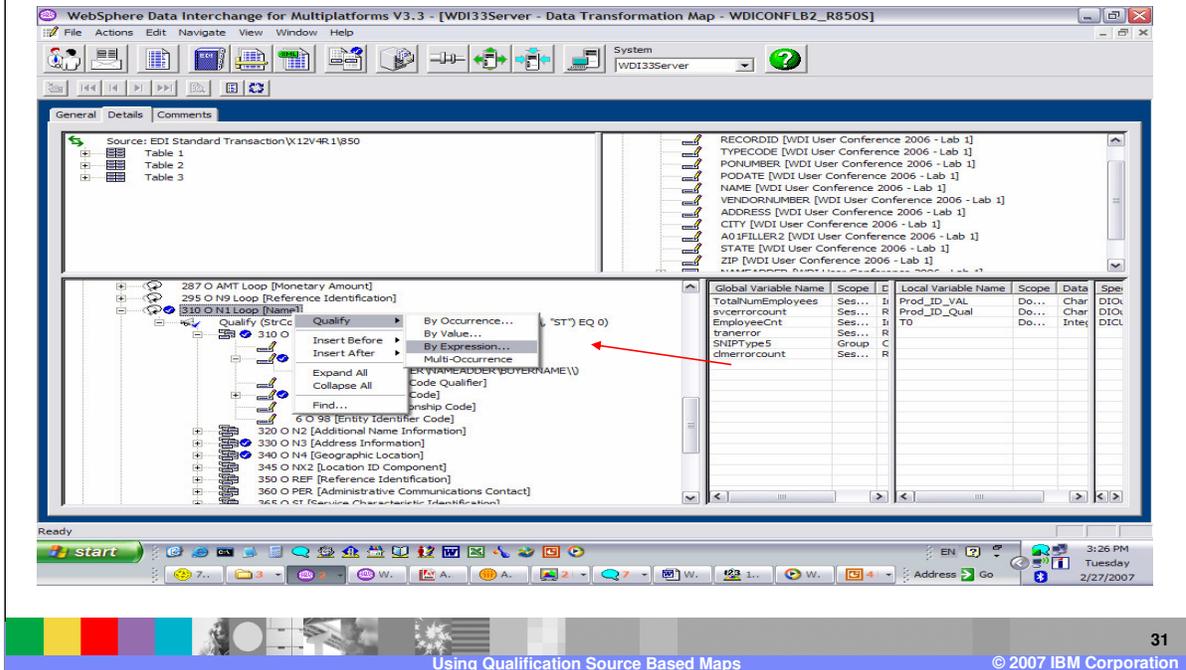
The mapping commands under this qualification will only be executed with the value “ST” is found in element 98 in the N1 segment.

Section

Source Based Expression

Using Element, Segment and Loop Qualifiers

- Qualification (Expression)



To create a qualification using an Expression, right click the source element, choose Qualify, and select By Expression.

Using Element, Segment and Loop Qualifiers

■ Qualification (Expression)

The screenshot displays the IBM WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window shows a data transformation map with a tree view of tables and loops. A 'Mapping Command Editor' dialog box is open, showing the 'Qualify (expression)' tab. The dialog box contains a list of values and functions, with a red arrow pointing to the 'Values' section. The list includes:

- Values
- Functions
- Operators
- Delimiters
- Undo
- Cut
- Copy
- Paste
- Delete

The 'Values' section is expanded, showing the following items:

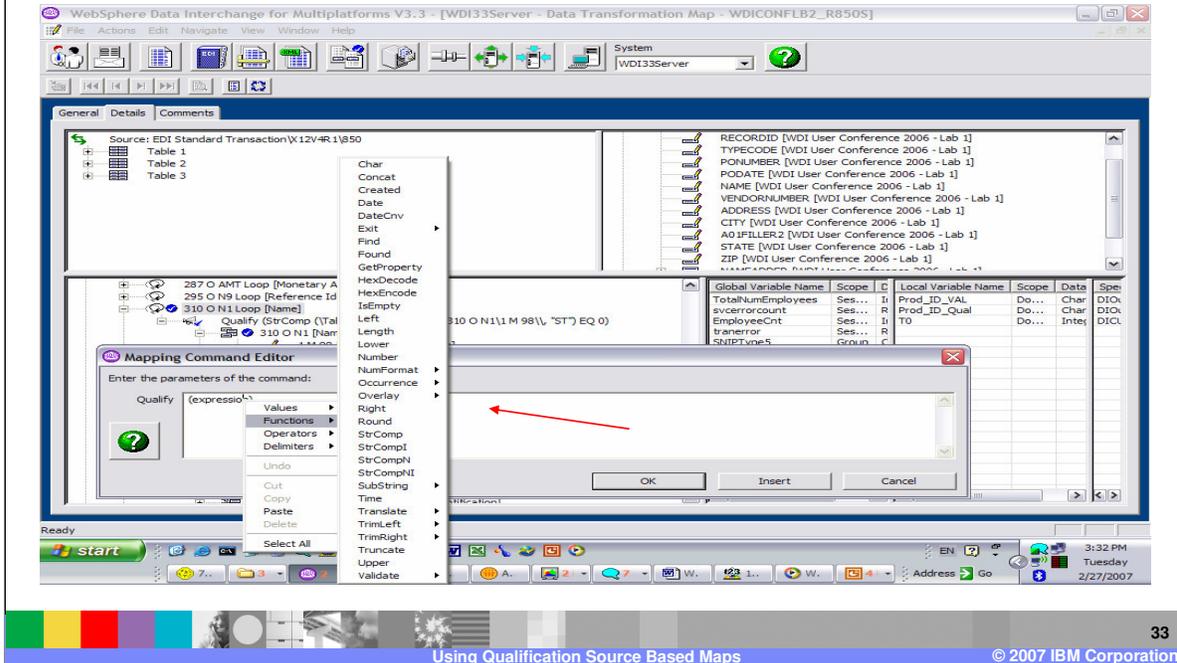
- (Occurrence 0 EQ num)
- (Occurrence (sourcepath) EQ num)
- (StrComp (path, "value") EQ 0)
- (expression)

The dialog box also has 'OK', 'Insert', and 'Cancel' buttons. The background window shows a data transformation map with various tables and loops, including 'Table 1' and 'NAMEADDR'. The system tray at the bottom shows the date and time as Wednesday, 2/28/2007, 9:05 AM.

Any number of Values and Functions can be combined to use as an expression. This is a list of Values.

Using Element, Segment and Loop Qualifiers

- Qualification (Expression)



This is a list of functions.

Using Element, Segment and Loop Qualifiers

- Qualification (Expression)

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3.3 interface. The main window displays a Data Transformation Map configuration. The left pane shows a tree view of data elements, including a loop named '310 O N1 Loop [Name]'. The right pane shows a table of global variables. Red arrows indicate the mapping of the 'MapTo' field in the tree view to the 'Global Variable Name' column in the table.

Global Variable Name	Scope	Local Variable Name	Scope	Data	Spe
TotalEmployees	Ses...	Prod_ID_VAL	Do...	Char	DIO...
EmployeeCnt	Ses...	TO	Do...	Char	DIO...
tranerror	Ses...		Do...	Intex	
SNPTypes	Group				
emerrorcount	Ses...				

Continuing with this example, say you want to qualify the N1 loop with the values of VN or ZZ in Element 98, which is the “Entity Identifier Code,” received in a purchase order to populate the vendor number field in a record. Further, say that you want the vendor number to be mapped into the header record depending on the value in Element 98 of the N1 loop.

Using Element, Segment and Loop Qualifiers

- Qualification (Expression)

The screenshot displays the IBM WebSphere Data Interchange for Multiplatforms V3.3 interface. A 'Mapping Command Editor' dialog box is open, showing a 'Qualify' field with the following expression: `(Str-Comp (Table 1\310 O N1 Loop\310 O N1\1 M 98\ \\, \"VN\") EQ 0 OR Str-Comp (charValue1, charValue2))`. Two red arrows point to the `EQ 0` and `OR` operators within the expression. The background shows a tree view of data elements and a list of variables.

You can construct this qualification using an Expression containing functions and operators. In this example the operator OR was used.

Using Element, Segment and Loop Qualifiers

- Qualification (Expression)

WebSphere Data Interchange for Multiplatforms V3.3 - [WDI33Server - Data Transformation Map - WDICONFLB2_R850S]

Mapping Command Editor

Enter the parameters of the command:

Qualify (StrComp ((Table 1\310 O N1 Loop\310 O N1\1 M 98\, \"VN\")EQ 0 OR StrComp ((Table 1\310 O N1 Loop\310 O N1\1 M 98\, \"ZZ\") = 0)

OK Cancel

310 O N1 Loop [Name]

- 1 M 98 [Entity Identifier Code]
- 2 C 93 [Name]
- 3 C 66 [Identification Code Qualifier]
- 4 C 67 [Identification Code]
- 5 O 706 [Entity Relationship Code]
- 6 O 98 [Entity Identifier Code]
- 320 O N2 [Additional Name Information]
- 330 O N3 [Address Information]
- 340 O N4 [Geographic Location]

Global Variable Name	Scope	Local Variable Name	Scope	Date	Spe
RECORDID [WDI User Conference 2006 - Lab 1]					
TYPECODE [WDI User Conference 2006 - Lab 1]					
POINUMBER [WDI User Conference 2006 - Lab 1]					
PODATE [WDI User Conference 2006 - Lab 1]					
NAME [WDI User Conference 2006 - Lab 1]					
VENDORNUMBER [WDI User Conference 2006 - Lab 1]					
ADDRESS [WDI User Conference 2006 - Lab 1]					
CITY [WDI User Conference 2006 - Lab 1]					
AD1FILLER2 [WDI User Conference 2006 - Lab 1]					
STATE [WDI User Conference 2006 - Lab 1]					
ZIP [WDI User Conference 2006 - Lab 1]					

345 O N42 [Location ID Components]

350 O REF [Reference Identification]

360 O PER [Administrative Communications Contact]

36 O ST [Source Characteristic Identification]

Ready

Using Qualification Source Based Maps © 2007 IBM Corporation

This is the completed Expression for our example.

Using Element, Segment and Loop Qualifiers

- WebSphere Data Interchange supports four types of qualification for send and receive maps.
 - ▶ Occurrence
 - ▶ Multiple occurrence
 - ▶ Value
 - ▶ Expression



WebSphere Data Interchange supports four types of qualification for maps. You can qualify by: Occurrence, multiple occurrence, Value, and Expressions. With the exception of Multiple occurrence, you can also use logic in the mapping to execute mapping based on a condition.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot displays the IBM WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window shows a Data Transformation Map (DTM) titled "WDICONFLB2_R8505". The map structure is visible in the left pane, showing a hierarchy of elements and segments. The right pane displays the logic for the selected element, "6 C 235 [Product/Service ID Qualifier]". The logic includes a loop and conditional statements that filter elements based on their "Prod_ID_Qual" value. A red arrow points to the logic for the "6 C 235" element. The bottom status bar indicates "Using Qualification Source Based Maps" and "© 2007 IBM Corporation".

In this example logic is used to accomplish the mapping for qualified element pairs. With this type of qualification the segment contains elements that are repeated and paired with one element containing a code which identifies the type of value in the other element. These elements can occur in any order which requires the element containing the code to be scanned. This type of qualification is an element qualification.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3.3 interface. The main window displays a Data Transformation Map for 'WDI33Server - Data Transformation Map - WDICONFLB2_R850S'. The left pane shows a tree view of map elements, including '10 M PO1 [Baseline Item Data]' and various qualifiers like '2 C 330 [Quantity Ordered]'. The central workspace shows a selected element with a context menu open, highlighting the 'If' command. The right pane shows a table of variables:

Global Variable Name	Scope	Data	Spec	Local Variable Name	Scope	Date	Spec
TotalNumEmployees	Ses...	I	DIO	Prod_ID_VAL	Do...	Char	DIO
gvcerrorcount	Ses...	R	DIO	Prod_ID_Qual	Do...	Char	DIO
EmployeeCnt	Ses...	I	DIO	TO	Do...	Integ	DIO
Trerror	Ses...	R	DIO				
SNIPType5	Group	C					
cmerrorcount	Ses...	R	DIO				

The bottom of the screenshot shows the Windows taskbar with the system clock indicating 9:18 AM on Wednesday, 2/28/2007. The footer of the slide contains the text 'Using Qualification Source Based Maps' and '© 2007 IBM Corporation'.

To add logic to a map, right click the element in the mapping window where you want the logic, choose Insert within, select Command, and the If command.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot shows the IBM WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window displays a tree view of a Data Transformation Map. A loop qualifier '10 M PO1 Loop' is selected, and its parameters are being edited in the 'Mapping Command Editor' dialog box. The dialog box contains the text 'If (expression)' and a red arrow pointing to the text field. The background shows various data elements and a table of global and local variable names.

Global Variable Name	Scope	Local Variable Name	Scope	Data	Spe
TotalNumEmployees	Ses...	Prod_ID_VAL	Do...	Char	DIO
svccerrorcount	Ses...	Prod_ID_Qual	Do...	Char	DIO
EmployeeCnt	Ses...	TO		Inte	
tranerror	Ses...			Inte	
SNIPType5	Group			Char	
clmerrorcount	Ses...			Inte	

Enter the condition or expression.

Using Element, Segment and Loop Qualifiers

Logic

WebSphere Data Interchange for Multiplatforms V3.3 - [WDI33Server - Data Transformation Map - WDI33CONFLB2_R850S]

General Details Comments

10 M PO1 [Baseline Item Data]

- 1 O 350 [Assigned Identification]
- 2 C 330 [Quantity Ordered]
- 3 O 355 [Unit or Basis for Measurement Code]
- 4 C 212 [Unit Price]
- 5 O 639 [Basis of Unit Price Code]
- 6 C 235 [Product/Service ID Qualifier]
- 7 C 234 [Product/Service ID]
- 8 C 234 [Product/Service ID]
- 9 C 234 [Product/Service ID]

RECORDID [WDI User Conference 2006 - Lab 1]

QTYORDERED [WDI User Conference 2006 - Lab 1]

UNITOFMEAS [WDI User Conference 2006 - Lab 1]

UNITPRICE [WDI User Conference 2006 - Lab 1]

ITEMDESC [WDI User Conference 2006 - Lab 1]

ITEMNUMBER-BN [WDI User Conference 2006 - Lab 1]

ITEMCODE1 [WDI User Conference 2006 - Lab 1]

ITEMNUMBER-B1 [WDI User Conference 2006 - Lab 1]

ITEMCODE2 [WDI User Conference 2006 - Lab 1]

ITEMNUMBER-BC [WDI User Conference 2006 - Lab 1]

ITEMCODE3 [WDI User Conference 2006 - Lab 1]

Global Variable Name Scope C Local Variable Name Scope Data Spe

Global Variable Name	Scope	C	Local Variable Name	Scope	Data	Spe
TotalNumEmployees	Ses...	I	Prod_ID_VAL	Do...	Char	DIC\
svccerrorcount	Ses...	R	Prod_ID_Qual	Do...	Char	DIC\
EmployeeCnt	Ses...	I	T0	Do...	Inter	DIC\
trnerror	Ses...	R				
SNIPTYPE5	Group	C				
cmerrorcount	Ses...	R				

Ready

Using Qualification Source Based Maps © 2007 IBM Corporation

This results in an If and EndIf command.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot shows the WebSphere Data Interchange (WDI) interface for Multiplatforms V3.3. The main window displays a Data Transformation Map (DTM) for 'WDICONFLB2_R8505'. The left pane shows a tree view of the map structure, including elements like '10 M PO1 [Baseline Item Data]' and '2 C 330 [Quantity Ordered]'. A context menu is open over an 'If' statement, with 'Insert Within' selected, and a sub-menu showing 'Command' as the chosen option. A red arrow points to the 'Command' option. The right pane shows a list of variables and their scopes. The bottom status bar indicates 'Using Qualification Source Based Maps' and '© 2007 IBM Corporation'.

To add mapping commands under the If, right click the If, choose Insert Within, select Command and the command to be executed. Continue adding commands within the If command using these steps.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window displays a Data Transformation Map (DTM) for a transformation named 'WDICONFLB2_R8505'. The map is structured as follows:

- 10 M PO1 [Baseline Item Data]
 - 1 O 350 [Assigned Identification]
 - 2 C 330 [Quantity Ordered]
 - 3 O 355 [Unit or Basis for Measurement Code]
 - 4 C 212 [Unit Price]
 - 5 O 639 [Basis of Unit Price Code]
 - 6 C 235 [Product/Service ID Qualifier]
 - 7 C 234 [Product/Service ID]
 - 8 C 235 [Product/Service ID Qualifier]
 - 9 C 234 [Product/Service ID]

The 'If' command is expanded to show its internal logic:

```

If (StrCompN (Table 2\10 M PO1 Loop\10 M PO1\6 C 235\
  Prod_ID_Qual = Table 2\10 M PO1 Loop\10 M PO1\
  Prod_ID_VAL = Table 2\10 M PO1 Loop\10 M PO1\
  EndIf
  If (StrCompN (Prod_ID_Qual, "B", 2) = 0)
  EndIf
  If (StrCompN (Prod_ID_Qual, "B1", 2) = 0)
  EndIf
  If (StrCompN (Prod_ID_Qual, "BC", 2) = 0)
  EndIf
  If (StrCompN (Prod_ID_Qual, "BP", 2) = 0)
  EndIf

```

The context menu is open over the 'If' command, and the 'Else' option is selected. A red arrow points to the 'Else' option in the menu.

The bottom of the screenshot shows the Windows taskbar with the system tray displaying the time as 9:27 AM on Wednesday, 2/28/2007. The page number 43 is visible in the bottom right corner.

To add Else and Elself, right click the If command, choose Insert After, select Else or Elself.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot displays the WebSphere Data Interchange for Multiplatforms V3.3.3 interface. The main window shows a Data Transformation Map configuration with a tree view on the left and a list of mappings on the right. A 'Mapping Command Editor' dialog is open in the foreground, with the text 'Enter the parameters of the command:' and 'ElseIf (expression)'. A red arrow points to the input field for the expression. The dialog has 'OK', 'Insert', and 'Cancel' buttons. The background window shows a table of mappings with columns for Global Variable Name, Scope, Local Variable Name, Scope, Data, and Specialization.

Global Variable Name	Scope	Local Variable Name	Scope	Data	Spec
TotalNumEmployees	Ses...	Prod_ID_VAL	Do...	Char	DIO...
svccorrcount	Ses...	Prod_ID_Qual	Do...	Char	DIO...
EmployeeCnt	Ses...	TO	Do...	Integ	DIO...
transerror	Ses...		R		
SNPTYPE5	Group		C		

Enter the condition or expression.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot shows the WebSphere Data Interchange for Multipatforms V3.3 interface. The main window displays a Data Transformation Map (DTM) for a source system 'WDI33Server'. The left pane shows a tree view of mapping commands, including a loop '10 M PO1 [Baseline Item Data]' and a conditional '6 C 235 [Product/Service ID Qualifier]'. The right pane lists data elements such as 'RECORDID', 'QTYORDERED', and 'UNITPRICE'. The bottom right pane shows a table of variables:

Global Variable Name	Scope	Local Variable Name	Scope	Data	Spec
TotalNumEmployees	Ses...	Prod_ID_VAL	Do...	Char	DIO...
svccerrorcount	Ses...	Prod_ID_Qual	Do...	Char	DIO...
EmploymentCnt	Ses...		Do...	Integ	DICL
tranerror	Ses...				
SNIPType5	Group				
clmerrcount	Ses...				

The status bar at the bottom indicates 'Using Qualification Source Based Maps' and '© 2007 IBM Corporation'.

Similar to the If command, to add mapping commands under the ElseIf, right click the ElseIf, choose Insert Within, select Command and the command to be executed.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot shows the WebSphere Data Interchange for Multipatforms V3.3 interface. The main window displays a Data Transformation Map (DTM) for a mapping project. The tree view on the left shows a hierarchy of data elements, including '10 M PO1 [Baseline Item Data]' and its children. A context menu is open over the 'Product/Service ID Qualifier' element, with the 'Command Group...' option highlighted by a red arrow. The right pane shows a list of global variables with columns for Name, Scope, and Data Type. The bottom status bar indicates the page number 46 and the copyright notice © 2007 IBM Corporation.

Global Variable Name	Scope	Local Variable Name	Scope	Data	Spe
TotalNumEmployees	Ses...	Prod_ID_VAL	Do...	Char	DIO
svccerrorcount	Ses...	Prod_ID_Qual	Do...	Char	DIO
EmployeeCnt	Ses...	T0	Do...	Inte	DICL
tranerror	Ses...				
SNIPTYPE5	Group				
cmerrorcount	Ses...				

When using logic that will be used on several elements for example element pairs, consider using Command Groups. With Command Groups you can group the mapping commands and copy the commands to different elements.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot displays the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window shows a Data Transformation Map for 'WDI33Server'. A 'Command Group Editor' dialog box is open, with the text 'Enter a brief Description:' and a text area for input. The dialog has 'OK', 'Insert', and 'Cancel' buttons. In the background, a tree view shows a hierarchy of data elements, including '10 M PO1 [Baseline Item Data]' and '6 C 235 [Product/Service ID Qualifier]'. A table of global variables is also visible, with columns for 'Global Variable Name', 'Scope', 'Local Variable Name', 'Scope', 'Data', and 'Spec'. The table contains entries like 'TotalNumEmployees' and 'SuperAccount'.

Global Variable Name	Scope	Local Variable Name	Scope	Data	Spec
TotalNumEmployees	Ses...	Prod_ID_VAL	Do...	Char	DIO...
SuperAccount	Ses...	Prod_ID_Val	Do...	Char	DIO...
			Do...	Inte...	DICL

When defining a command group, enter a description for the Command Group.

Using Element, Segment and Loop Qualifiers

Logic

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window displays a Data Transformation Map with the following logic:

```

10 M PO1 [Baseline Item Data]
  1 O 350 [Assigned Identification]
  2 C 330 [Quantity Ordered]
  3 O 355 [Unit or Basis for Measurement Code]
  4 C 212 [Unit Price]
  5 O 639 [Basis of Unit Price Code]
  6 C 235 [Product/Service ID Qualifier]
    Product ID Qual/ID Pairs
    Prod_ID_Qual = (Table 2\10 M PO1 Loop\10 M PO1\6 C 235\
    Prod_ID_VAL = (Table 2\10 M PO1 Loop\10 M PO1\7 C 234\
    If (StrCompN (Prod_ID_Qual, "B", 2) = 0)
    EndIf
    If (StrCompN (Prod_ID_Qual, "B1", 2) = 0)
    EndIf
    If (StrCompN (Prod_ID_Qual, "BC", 2) = 0)
    EndIf
    If (StrCompN (Prod_ID_Qual, "BP", 2) = 0)
    EndIf
  7 C 234 [Product/Service ID]
  8 C 235 [Product/Service ID Qualifier]
    Product ID Qual/ID Pairs
    Prod_ID_Qual = (Table 2\10 M PO1 Loop\10 M PO1\8 C 235\
    Prod_ID_VAL = (Table 2\10 M PO1 Loop\10 M PO1\9 C 234\
    If (StrCompN (Prod_ID_Qual, "B", 2) = 0)
    EndIf
    If (StrCompN (Prod_ID_Qual, "B1", 2) = 0)
    EndIf
    If (StrCompN (Prod_ID_Qual, "BC", 2) = 0)
    EndIf
  
```

A red arrow points to the 'Product ID Qual/ID Pairs' command group within the loop for '6 C 235 [Product/Service ID Qualifier]'. The right-hand pane shows a table of global and local variables:

Global Variable Name	Scope	Local Variable Name	Scope	Data	Spe
TotalNumEmployees	Ses...	Prod_ID_VAL	Do...	Char	DIO
sverrorcount	Ses...	Prod_ID_Qual	Do...	Char	DIO
EmployeeCnt	Ses...	TO	Do...	Integ	DICL
transerror	Ses...				
SNIPTypeS	Group				
clmerrorcount	Ses...				

The bottom of the screenshot shows the Windows taskbar with the system clock at 9:34 AM on Wednesday, 2/28/2007. The footer text reads: 'Using Qualification Source Based Maps © 2007 IBM Corporation'.

To add commands within a command group use right click on the command group and select insert within. Or you can use drag and drop to move commands within the command group. This is an example of command groups. To Copy command groups or any other mapping command, press the Shift key on the command or command group, then drag and drop. Don't forget to check and change the source and target paths with the copied mapping. With command groups this will copy or move all commands within the command group.

Using Element, Segment and Loop Qualifiers

- Mapping Tip:
 - ▶ With complex qualifications, try doing the qualification and only map a few elements.
 - ▶ Test and check the source and target looping to ensure it is correct before completing the mapping.



With complex qualifications, try doing the qualification and only map a few elements. Test and check the source and target looping to ensure it is correct before completing the mapping.

Using Element, Segment and Loop Qualifiers

- Summary

- ▶ WebSphere Data Interchange supports four types of qualification for send and receive maps.
 - Occurrence
 - Multiple occurrence
 - Value
 - Expression
 - Logic may be used as an alternative to Occurrence, Value and Expression qualification.



WebSphere Data Interchange supports four types of qualification for maps. You can qualify by: Occurrence, multiple occurrence, Value, and Expressions. With the exception of Multiple occurrence, you can also use logic in the mapping to execute mapping based on a condition.

Reference

- More information can be found in the WDI V3.3 Mapping Guide.



More information can be found in the WebSphere Data Interchange Version 3.3 Mapping Guide.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	WMO	Tivoli
IBM (logo)	Cloudscape	Informix	OS/390	WebSphere
e! (logo)/business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2006. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.