



IBM Software Group

IBM WebSphere® Data Interchange V3.3

Mapping Hierarchical Loops



@business on demand.

© 2007 IBM Corporation

This presentation will illustrate how to map Hierarchical Loops in Data Transformation maps.

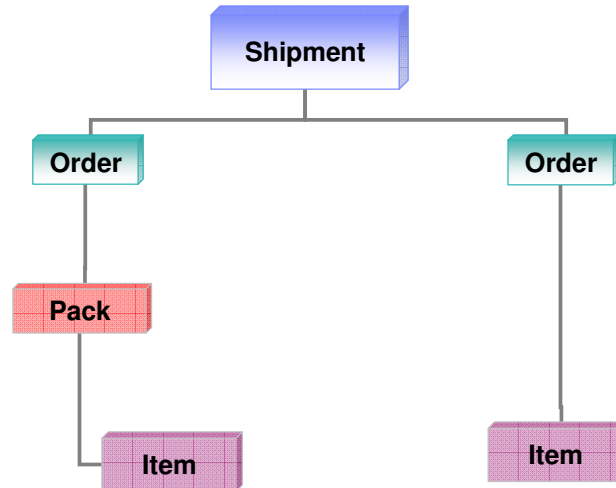
Agenda

- What is special HL Loop mapping?
- Selecting special HL Loop mapping.
- Qualification.
- Add parent child relationships.
- Summary and references



The presentation will describe Hierarchical Level (HL) Loops and describe how to use the WebSphere Data Interchange (WDI) special Hierarchical Loop Mapping commands.

Mapping Hierarchical Loops



This is an example of a hierarchy. The Shipment contains orders, orders contain packs or items, and packs contain items.

Mapping Hierarchical Loops

- What is special HL Loop Mapping?
 - ▶ The EDI HL segment contains values that identify each level of the hierarchy and the parent child relationships.
 - ▶ Specifically developed for Send (EDI target) but can be used for Receive (EDI source).
 - ▶ Should be used when your source data does not contain the values to be mapped to the HL segment.
 - ▶ WDI special HL Loop Mapping defines the hierarchy which allows the translation process to generate the HL segment values.



WDI provides special mapping functions to map the Electronic Data Interchange (EDI) standard Hierarchical Level (HL) segment. This special mapping is specifically for an EDI target document when the source document does not contain the values that define the hierarchy in the HL segment.

Mapping Hierarchical Loops

- Data Transformation mapping:
 - ▶ Creates a visual representation of the hierarchy
- Special HL Loop mapping not wanted and has been selected
 - ▶ Must delete all mapping on the HL Loop to remove special HL Loop mapping.



Data Transformation (DT) mapping allows you to visually create the hierarchical loop structure in the map using special HL Loop Qualification. If special HL Loop mapping has been selected and you do not want to use it you must delete all mapping on the HL Loop to remove special HL Loop mapping.

Mapping Hierarchical Loops

- Selecting Special HL Loop Mapping
 - Source based mapping is required to use special HL mapping support when EDI is the source message.
 - ▶ The EDI parser constructs the HL Hierarchy in the abstract message.
 - ▶ Transformation executes the mapping instructions based on the hierarchy if the special HL mapping was used. Otherwise the abstract message is re-constructed as a flat hierarchy.
 - ▶ If your source message does not contain the parent/child elements in the HL segment, the hierarchy will be flat.



Source based mapping is required to use special HL mapping support when EDI is the source message type. The EDI parser will always construct the HL Hierarchy in the abstract message. HL Loops will be placed as a child of their parent. The Transformation executes the mapping instructions based on the hierarchy in the map and the abstract message if the special HL mapping was used. Otherwise the abstract message is re-constructed as a flat hierarchy. If your source message does not contain the parent/child elements in the HL segment, the hierarchy will be flat. The HL Loops will all be siblings.

Mapping Hierarchical Loops

▪ EDI no hierarchy (no parent id HL02)

- ▶ ST*856*45920001~
- ▶ BSN*30*01*050524*0049~
- ▶ HL*1**S~
- ▶ N1*SF**92*SHIP1~
- ▶ HL*2**O~
- ▶ PRF*ORD1*11113711**050523~
- ▶ HL*3**P~
- ▶ MAN*GM*PAC1111~
- ▶ HL*4**O~
- ▶ PRF*ORD2*11113712**050523~
- ▶ HL*5**P~
- ▶ MAN*GM*PAC2222~
- ▶ HL*6**S~
- ▶ N1*SF**92*SHIP2~
- ▶ HL*7**O~
- ▶ PRF*ORD3*11113711**050523~
- ▶ HL*8**P~
- ▶ MAN*GM*PAC3333~
- ▶ HL*9**O~
- ▶ PRF*ORD4*11113712**050523~
- ▶ HL*10**P~
- ▶ MAN*GM*PAC4444~
- ▶ CTT*10~



This is an example of an EDI source document with HL Loops and no hierarchy defined. The HL segments do not contain the parent identification and is a flat hierarchy.

Mapping Hierarchical Loops

- Selecting Special HL Loop Mapping
 - Target based mapping is required to use special HL mapping support when EDI is the target message.
 - ▶ The Transformation will create the hierarchy with the HL level code value HL03. ONLY if special HL mapping was used.
 - ▶ The EDI serialization generates HL01, HL02, and HL04 values based on the hierarchy produced by the Transformation.



Target based mapping is required to use special HL mapping support when EDI is the target message. The Transformation will create the hierarchy with the HL level code value HL03 in the abstract message. HL Loops will be placed as a child of their parent. ONLY if special HL mapping was used. The EDI serialization generates HL01, HL02, and HL04 values based on the parent/child relationship in the abstract message produced by the Transformation.

Mapping Hierarchical Loops

- Selecting Special HL Loop Mapping
 - ▶ With XML source, the DTD does not always define the the hierarchy. No proper hierarchical nesting for mapping.
 - ▶ With XML source, the input data may look like it contains the hierarchy with spacing and indentation.
 - ▶ XML is free form.
 - ▶ Hierarchy defined by using the DTD or Schema definition.



With XML source, the DTD does not always have the parent/child relationship for the hierarchy. There is no proper hierarchical nesting for mapping. The input data may visually look like it contains the hierarchy with spacing and indentation. XML is free form and the only way to identify the hierarchy is by using the DTD or Schema definition.

Mapping Hierarchical Loops

The screenshot shows the WebSphere Data Interchange Client Overview tab for a DTD named '278A2E'. The DTD definition is displayed in a tree view, showing a hierarchical structure of segments and loops. The root element is 'Transaction-278', which contains several internal properties and segments. The segments include 'Internal-Properties', 'Segment-ST-Transaction-Set-Header', 'Segment-BHT-Beginning-of-Hierarchical-Transaction', and 'Loop-2000A-Utilization-Management-Organization-UMO-Level'. The 'Loop-2000A' segment contains a 'Segment-HL-Utilization-Management-Organization-UMO-Level' which further contains 'Element-628-Hierarchical-ID-Number', 'Element-734-Hierarchical-Parent-ID-Number', and 'Element-735-Hierarchical-Level-Code'. This structure repeats for 'Loop-2000B-Requester-Level' and 'Loop-2000C-Subscriber-Level'. The DTD definition is as follows:

```

Transaction-278 [([Internal-Properties?,Segment-ST-Transaction-Set-Header,Segment-BHT-Beginning-of-Hierarchical-Transaction,Loop-2000A-Utilization-Management-Organization-UMO-Level,Loop-2000B-Requester-Level,Loop-2000C-Subscriber-Level])]
  Internal-Properties [([Data-Structure*])]
  Segment-ST-Transaction-Set-Header [(Element-143-Transaction-Set-Identifier-Code,Element-329-Transaction-Set-Control-Number)]
  Segment-BHT-Beginning-of-Hierarchical-Transaction [(Element-1005-Hierarchical-Structure-Code,Element-353-Transaction-Set-Purpose-Code,Element-127-Reference-Identifier-Code,Element-1006-Transaction-Set-Header-Code)]
  Loop-2000A-Utilization-Management-Organization-UMO-Level [([Segment-HL-Utilization-Management-Organization-UMO-Level,Segment-AAA-Request-Validation*,Loop-2010A-Request-Validation*])]
    Segment-HL-Utilization-Management-Organization-UMO-Level [(Element-628-Hierarchical-ID-Number,Element-734-Hierarchical-Parent-ID-Number?,Element-735-Hierarchical-Level-Code)]
      Element-628-Hierarchical-ID-Number [(#PCDATA)]
      Element-734-Hierarchical-Parent-ID-Number [EMPTY]
      Element-735-Hierarchical-Level-Code [(#PCDATA)]
    Segment-AAA-Request-Validation [(Element-1073-Yes-No-Condition-or-Response-Code,Element-559-Agency-Qualifier-Code?,Element-901-Reject-Reason-Code,Element-1074-Request-Validation-Code)]
    Loop-2010A-Request-Validation [([Segment-NM-1-Requester-Name,Segment-REF-Requester-Supplemental-Identification*,Segment-AAA-2-Requester-Request-Validation*])]
      Segment-NM-1-Requester-Name [(Element-628-Hierarchical-ID-Number,Element-734_1-Hierarchical-Parent-ID-Number,Element-735_1-Hierarchical-Level-Code,Element-736-Hierarchical-Child-Code)]
        Element-628-Hierarchical-ID-Number [(#PCDATA)]
        Element-734_1-Hierarchical-Parent-ID-Number [(#PCDATA)]
        Element-735_1-Hierarchical-Level-Code [(#PCDATA)]
        Element-736-Hierarchical-Child-Code [(#PCDATA)]
      Segment-REF-Requester-Supplemental-Identification [EMPTY]
      Segment-AAA-2-Requester-Request-Validation [([Segment-HL-2-Subscriber-Level,Segment-TRN-Patient-Event-Tracking-Number*,Segment-AAA-3-Subscriber-Request-Validation*,Segment-OT-Request-Validation*])]
        Segment-HL-2-Subscriber-Level [(Element-628-Hierarchical-ID-Number,Element-734_1-Hierarchical-Parent-ID-Number,Element-735_2-Hierarchical-Level-Code,Element-736-Hierarchical-Child-Code)]
          Element-628-Hierarchical-ID-Number [(#PCDATA)]
          Element-734_1-Hierarchical-Parent-ID-Number [(#PCDATA)]
          Element-735_2-Hierarchical-Level-Code [(#PCDATA)]
          Element-736-Hierarchical-Child-Code [(#PCDATA)]
        Segment-TRN-Patient-Event-Tracking-Number [EMPTY]
        Segment-AAA-3-Subscriber-Request-Validation [EMPTY]
        Segment-OT-Request-Validation [EMPTY]
    Loop-2000B-Requester-Level [([Segment-HL-1-Requester-Level,Loop-2010B-Requester-Name])]
      Segment-HL-1-Requester-Level [(Element-628-Hierarchical-ID-Number,Element-734_1-Hierarchical-Parent-ID-Number,Element-735_1-Hierarchical-Level-Code,Element-736-Hierarchical-Child-Code)]
        Element-628-Hierarchical-ID-Number [(#PCDATA)]
        Element-734_1-Hierarchical-Parent-ID-Number [(#PCDATA)]
        Element-735_1-Hierarchical-Level-Code [(#PCDATA)]
        Element-736-Hierarchical-Child-Code [(#PCDATA)]
      Loop-2010B-Requester-Name [([Segment-NM-1-Requester-Name,Segment-REF-Requester-Supplemental-Identification*,Segment-AAA-2-Requester-Request-Validation*])]
        Segment-NM-1-Requester-Name [(Element-628-Hierarchical-ID-Number,Element-734_1-Hierarchical-Parent-ID-Number,Element-735_1-Hierarchical-Level-Code,Element-736-Hierarchical-Child-Code)]
          Element-628-Hierarchical-ID-Number [(#PCDATA)]
          Element-734_1-Hierarchical-Parent-ID-Number [(#PCDATA)]
          Element-735_1-Hierarchical-Level-Code [(#PCDATA)]
          Element-736-Hierarchical-Child-Code [(#PCDATA)]
        Segment-REF-Requester-Supplemental-Identification [EMPTY]
        Segment-AAA-2-Requester-Request-Validation [([Segment-HL-2-Subscriber-Level,Segment-TRN-Patient-Event-Tracking-Number*,Segment-AAA-3-Subscriber-Request-Validation*,Segment-OT-Request-Validation*])]
          Segment-HL-2-Subscriber-Level [(Element-628-Hierarchical-ID-Number,Element-734_1-Hierarchical-Parent-ID-Number,Element-735_2-Hierarchical-Level-Code,Element-736-Hierarchical-Child-Code)]
            Element-628-Hierarchical-ID-Number [(#PCDATA)]
            Element-734_1-Hierarchical-Parent-ID-Number [(#PCDATA)]
            Element-735_2-Hierarchical-Level-Code [(#PCDATA)]
            Element-736-Hierarchical-Child-Code [(#PCDATA)]
          Segment-TRN-Patient-Event-Tracking-Number [EMPTY]
          Segment-AAA-3-Subscriber-Request-Validation [EMPTY]
          Segment-OT-Request-Validation [EMPTY]
    Loop-2000C-Subscriber-Level [([Segment-HL-2-Subscriber-Level,Segment-TRN-Patient-Event-Tracking-Number*,Segment-AAA-3-Subscriber-Request-Validation*,Segment-OT-Request-Validation*])]
      Segment-HL-2-Subscriber-Level [(Element-628-Hierarchical-ID-Number,Element-734_1-Hierarchical-Parent-ID-Number,Element-735_2-Hierarchical-Level-Code,Element-736-Hierarchical-Child-Code)]
        Element-628-Hierarchical-ID-Number [(#PCDATA)]
        Element-734_1-Hierarchical-Parent-ID-Number [(#PCDATA)]
        Element-735_2-Hierarchical-Level-Code [(#PCDATA)]
        Element-736-Hierarchical-Child-Code [(#PCDATA)]
      Segment-TRN-Patient-Event-Tracking-Number [EMPTY]
      Segment-AAA-3-Subscriber-Request-Validation [EMPTY]
      Segment-OT-Request-Validation [EMPTY]
  
```

This is a sample DTD that contains no nesting. The DTD definition has been displayed using the WebSphere Data Interchange Client Overview tab.

Mapping Hierarchical Loops

The screenshot displays the WebSphere Data Interchange Client Overview tab for a DTD named X12_A1_277. The DTD structure is as follows:

- X12_A1_277 [(UserId?,header?,ST?,BHT?,L2000A*,SE?,trailer?)]
 - UserId [(#PCDATA)]
 - header [(ISA?,ISB?,ISE?,TAX?,GS?)]
 - ST [(ST_Transaction_Set_Identifier_Code?,ST_Transaction_Set_Control_Number?)]
 - BHT [(BHT_Hierarchical_Structure_Code?,BHT_Transaction_Set_Purpose_Code?,BHT_Reference_Identifier?,BHT_Date?,BHT_Time?,BHT_Transaction_Type_Code?)]
 - L2000A [(HL?,L2100A*,L2000B*)]
 - L2000B [(HL?,L2100B*,L2000C*)]
 - L2000C [(HL?,L2100C*,L2000D*)]
 - L2000D [(HL?,DMG?,L2100D*,L2200D*,L2000E*)]
 - DMG [(DMG_Date_Time_Period_Format_Qualifier?,DMG_Date_Time_Period?,DMG_Gender_Code?,DMG_Marital_Status_Code?,DMG_Race_Or_Ethnicity_Code?)]
 - L2100D [(NM1?)]
 - L2200D [(TRN?,STC?,REF?,REF_1?,REF_2?,DTP?,L2220D*)]
 - L2000E [(HL?,DMG?,L2100E*,L2200E*)]
 - L2200E [(TRN?,STC?,REF?,REF_1?,REF_2?,DTP?,L2220E*)]
 - TRN [(TRN_Trace_Type_Code?,TRN_Reference_Identifier?,TRN_Originating_Company_Identifier?,TRN_Reference_Identifier?)]
 - STC [(C043?,STC_Date?,STC_Action_Code?,STC_Monetary_Amount?,STC_Monetary_Amount_1?,STC_Date_1?,STC_Payment_Monetary_Amount?,STC_Payment_Monetary_Amount_1?,C043_Industry_Code?,C043_Industry_Code_1?,C043_Entity_Identifier_Code?)]
 - C043_Industry_Code [(#PCDATA)]
 - C043_Industry_Code_1 [(#PCDATA)]

This is a sample DTD that contains nesting. The DTD definition has been displayed using the WebSphere Data Interchange Client Overview tab.

Mapping Hierarchical Loops

- Selecting Special HL Loop Mapping
 - WDI Send maps containing special HL Loop mapping will require Target based Data Transformation map.
 - WDI Receive maps containing special HL Loop mapping will require Source based Data Transformation map.



If you are converting WDI Send or Receive maps that contain the special HL Loop mapping to Data Transformation (DT) maps. Send maps will require the DT map to be target based. Receive maps will require the DT map to be source based.

Mapping Hierarchical Loops

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window displays a Data Transformation Map with a source tree on the left and a target tree on the right. A right-click context menu is open over the '10 M HL Loop [Hierarchical Level]' in the target tree, with a blue callout box pointing to it that says 'Right Click'. The context menu options include 'Insert HL Qualification...', 'ForEach...', 'Insert Before', 'Insert After', 'Insert Within', 'Expand All', 'Collapse All', and 'Find...'. The bottom right of the window shows a table of Global Variable Name, Scope, Data Type, and Local Variab... with columns for Scope and D.

The special HL Loop mapping is accomplished by using the HL Loop Level codes. Drag and Drop operation can NOT be used. To begin, you identify the base level. To create the base HL level:

1. Go to the mapping details tab, and right-click the HL loop in the command window. The Qualification Selection Window displays.
2. Select Insert HL Qualification
3. Select an HL level code from the drop-down list. The list contains valid codes from the code list for HL03 (element 735).

This example is EDI Target and is a Target based map.

Mapping Hierarchical Loops

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window displays a tree view of data sources under 'Sources: Data Format\WDICONFHLAB_DICTIONARY\WDICONFHLAB'. A dialog box titled 'HL Qualification' is open, prompting the user to select an HL level code. The dialog box contains the following text: 'Select the HL level code that identifies this hierarchical level. This is the HL03 element value. Values for the hierarchical level code are obtained from the code list identified by the "735" element.'

The dialog box features a drop-down list with the following options:

- 1* - Service/Billing Provider
- 19* - Provider of Service
- 2* - Billing Arrangement
- 20* - Information Source
- 21* - Information Receiver
- 22* - Subscriber
- 23* - Dependent
- 24* - Supergroup
- 25* - Subgroup
- 90 O PWK [Paperwork]
- 100 O PKG [Marking, Packaging, Loading]
- 110 O TD1 [Carrier Details (Quantity and Weight)]
- 120 O TD5 [Carrier Details (Routing Sequence/Transit Time)]
- 130 O TD3 [Carrier Details (Equipment)]

Below the list is a table with the following columns: Global Variable Name, Scope, Data Type, Local Variab..., Scope, and D. The table contains the following data:

Global Variable Name	Scope	Data Type	Local Variab...	Scope	D.
TotalNumEmployees	Ses...	Integer			
svcerrortcount	Ses...	Real			
EmployeeCnt	Ses...	Integer			
tranerror	Ses...	Real			
SNIPType5	Group	Character			
cmerrortcount	Ses...	Real			

The bottom of the screenshot shows the Windows taskbar with the date and time: Thursday 3/1/2007 4:19 PM. The page number '14' is visible in the bottom right corner of the screenshot area.

Select the Level for this Qualification from the drop down list. The values in the drop down list come from the EDI code list for element 735.

Mapping Hierarchical Loops

Source: Data Format\WDICONFHLLAB_DICTIONARY\WDICONFHLLAB

- H00 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
- H01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
- SHIP_LOOP
 - SH01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 - ORDER_LOOP
 - ORD01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 - PACK_LOOP
 - PA01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 - ITEM_LOOP
 - IT01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]

Target Tree:

- 10 M HL Loop [Hierarchical Level]
 - 10 M HL [Hierarchical Level]
 - 20 O LIN [Item Identification]
 - 30 O SN1 [Item Detail (Shipment)]
 - 40 O SLN [Subline Item Detail]
 - 50 O PRF [Purchase Order Reference]
 - 60 O PO4 [Item Physical Details]
 - 70 O PID [Product/Item Description]
 - 80 O MEA [Measurements]
 - 90 O PWK [Paperwork]
 - 100 O PKG [Marking, Packaging, Loading]
 - 110 O TD1 [Carrier Details (Quantity and Weight)]

Command Window:

Global Variable Name	Scope	Data Type	Local Variab...	Scope	D.
TotalNumEmployees	Ses...	Integer			
sverrorcount	Ses...	Real			
EmployeeCnt	Ses...	Integer			
tranerror	Ses...	Real			
SHIPTypeS	Group	Character			
cmerrorcount	Ses...	Real			

Mapping Configuration:

- 10 M HL Loop [Hierarchical Level]
 - HLLevel (S)
 - 1 M 628 [Hierarchical ID Number]
 - HLAutoMapped
 - 2 O 734 [Hierarchical Parent ID Number]
 - HLAutoMapped
 - 3 M 735 [Hierarchical Level Code]
 - HLAutoMapped
 - 4 O 736 [Hierarchical Child Code]
 - HLAutoMapped
 - 20 O LIN [Item Identification]
 - 30 O SN1 [Item Detail (Shipment)]
 - 40 O SLN [Subline Item Detail]

An HLLevel mapping command is created in the command window as follows:

HLLevel (char "level code")

HLLevel ('S')

The HL segment is automatically selected and all elements mapped with an HLAutoMapped command. This command can not be removed or modified.

Additional mapping for these elements can be accomplished using normal mapping methods.

Mapping Hierarchical Loops

- The HL segment is the first segment of the HL loop
- HL*ID Number*Parent ID Number*Level Code*Child Code!
 - ▶ ID Number - Sequence number for HL loop
 - ▶ Parent ID Number - Sequence number of parent HL loop
 - ▶ Level Code - A code used to identify the instance of the HL loop
 - ▶ Child Code - 0 or 1 indicates HL loop has children



The HL segment is the first segment of the HL loop. The ID Number is a unique sequence number, the parent ID number is the sequence number of the parent HL segment or loop, the level code is a code used to identify this instance of the HL loop, and the child code indicates if this instance has children.

Mapping Hierarchical Loops

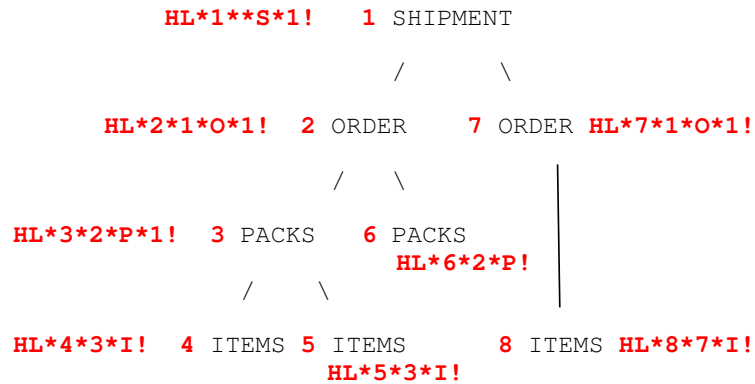
- Hierarchical Rules
 - ▶ Define hierarchical paths.
 - Number the nodes top to bottom and left to right starting with the left most node.
 - ▶ Identify base nodes.
 - Node 1 Shipment
 - ▶ Identify levels.
 - Shipment = S, Order = O, Packs = P, Items = I
 - ▶ Identify relationships
 - Is a Shipment required for an Order?



Here are some rules for defining a hierarchy before you begin mapping. You want to define the hierarchical paths, identify base nodes, and identify levels. Determine the relationships between the levels for example is a shipment level required for the order level or can the order contain an item level without a pack level.

Mapping Hierarchical Loops

- HL segment
- HL*ID Number*Parent ID Number*Level Code*Child Code!



Number the nodes top to bottom and left to right starting with the left most node.

Mapping Hierarchical Loops

- Sample application data

```

H00 DIUTESTPTR      01003020VICS  940830170700000170900062921
H01 00131313131313  9407141416  0001
SH011                S
SH02CTN250000072
OR012                1          O
OR02887378
OR03B                C FOREWAY
OR04BYHLLLOOP, INC.  9 DUNSNUMBR0897
PA013 2              P
PA0200000600000000  000000000054000LB0000000000000000
IT014                3          I
IT02                UP022222640304
IT03                0000000006EA0000000000000000006EA
  
```



This is an example of application data that will be mapped to the HL Loop.

Mapping Hierarchical Loops

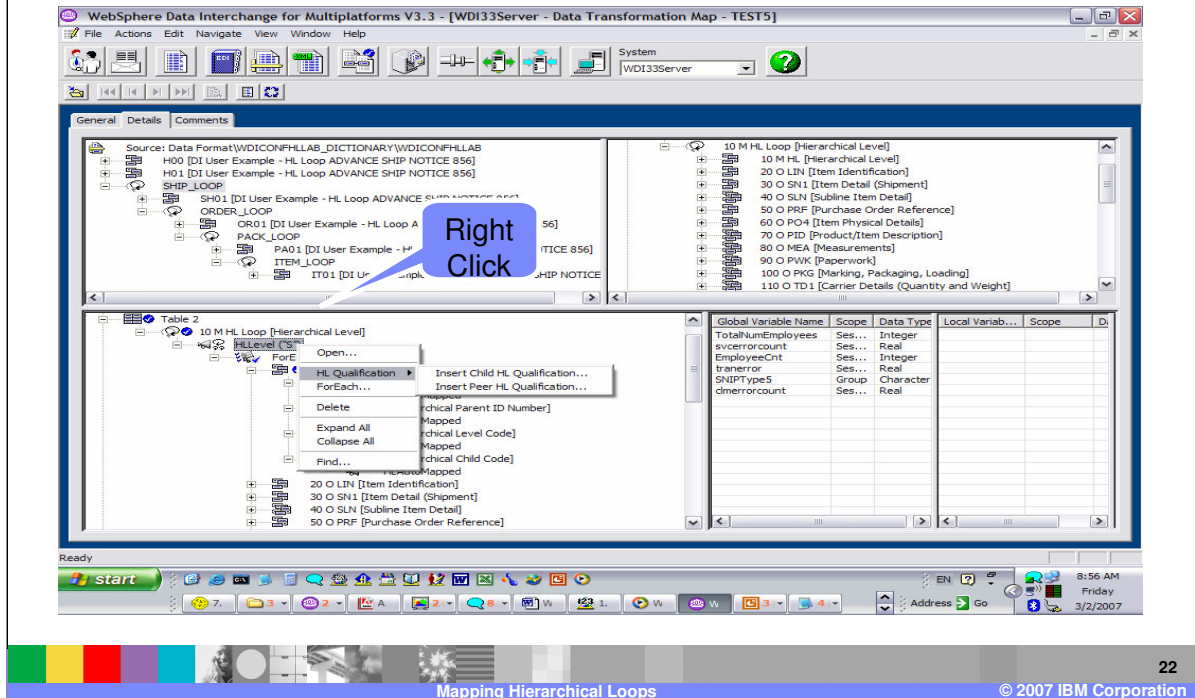


If you use the rules and identify the levels and the relationships, the mapping process will be easier. The hierarchy defined in this example contains 2 possible hierarchies for the Order level. Orders can have Packs and Items or Items. This will require HL Loop mapping for Items when Packs is the parent and Items when Order is the parent. Therefore Items is a child of Packs. Items is a child of Order and also a sibling of Packs.

Section

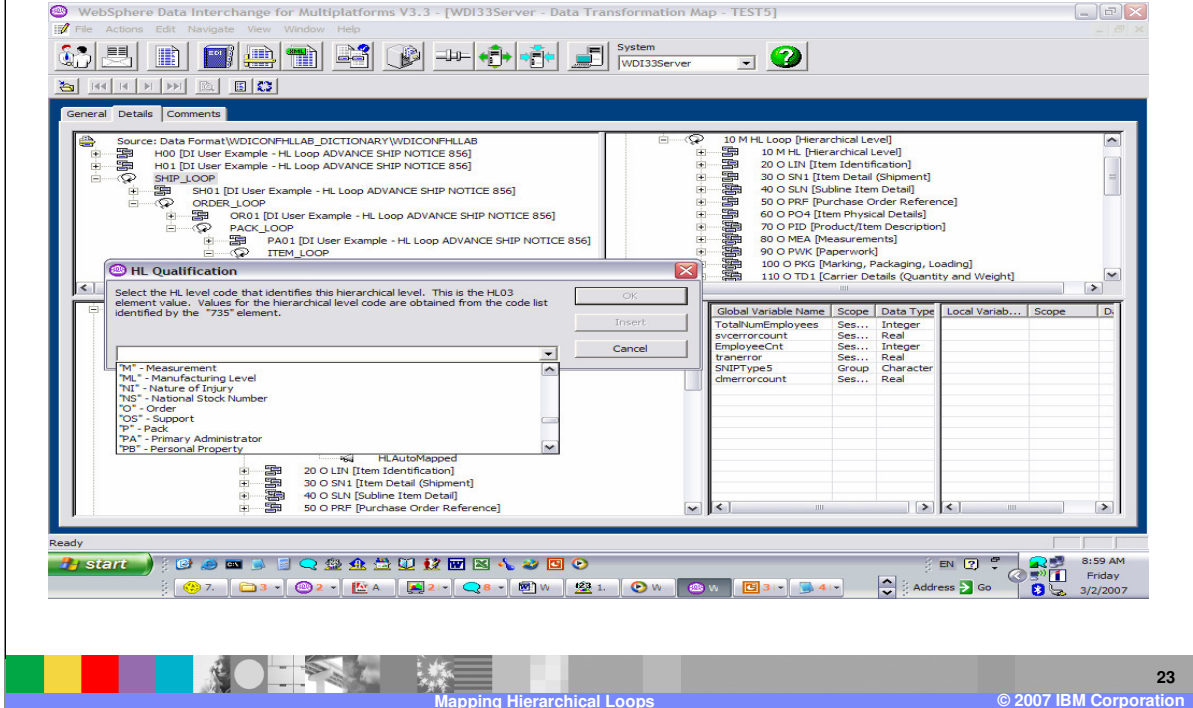
Adding parent/child relationship

Mapping Hierarchical Loops



To create the underlying HL nesting levels or children for this HL level, right-click the HLLevel command and select HL Qualification. The “Insert Child HL Qualification” and “Insert Peer HL Qualification” commands display. To create a child node, select the “Insert Child HL Qualification” command. After selection of Insert Child, select an HL level code from the drop-down list. The list contains valid codes from the code list for HL03 (element 735). The HL loop is inserted as a child of the current HL loop with an HLLevel command.

Mapping Hierarchical Loops



Select the Level for this Qualification from the drop down list. The values in the drop down list come from the EDI code list for element 735.

Mapping Hierarchical Loops

Global Variable Name	Scope	Data Type	Local Variab...	Scope	D.
TotalNumEmployees	Ses...	Integer			
svccerrorcount	Ses...	Real			
EmployeeCnt	Ses...	Integer			
tranerror	Ses...	Real			
SNIPType5	Group	Character			
dmerrorcount	Ses...	Real			

Adding a child or sibling will copy the HL Loop and place it in its hierarchical location. In this example, a child HL Loop was added and the HL Loop was copied after the last segment of the parent HL Loop as a child.

Mapping Hierarchical Loops

WebSphere Data Interchange for Multiplatforms V3.3 - [WDI33Server - Data Transformation Map - TEST5]

Source: Data Format(WDICONFHLLAB_DICTIONARY\WDICONFHLLAB)

H00 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]

H01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]

SHIP_LOOP

SH01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]

ORDER_LOOP

OR01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]

PACK_L

Mapping Command Editor

Enter the parameters of the command:

ForEach (SHIP_LOOP|ORDER_LOOP\)

Right Click

10 M HL Loop [Hierarchical Level]

10 M HL [Hierarchical Level]

20 O LIN [Item Identification]

30 O SN1 [Item Detail (Shipment)]

40 O SLN [Subline Item Detail]

50 O PRF [Purchase Order Reference]

60 O PO4 [Item Physical Details]

330 O GF [Furnished Goods and Services]

335 O YNQ [Yes/No Question]

340 O LM Loop [Code Source Information]

360 O V1 Loop [Vessel Identification]

10 M HL Loop [Hierarchical Level]

HLLevel (O)

10 M HL [Hierarchical Level]

1 M 628 [Hierarchical ID Number]

HLAutoMapped

2 O 734 [Hierarchical Parent ID Number]

Ready

9:04 AM
Friday
3/2/2007

Mapping Hierarchical Loops © 2007 IBM Corporation

You can add qualification at any HL Level by using the right click on the HLLevel command. A ForEach command must be used before additional qualifications may be added.

Mapping Hierarchical Loops

The screenshot shows the 'WebSphere Data Interchange for Multiplatforms V3.3' interface. The main window displays a 'Data Transformation Map' with a source tree on the left and a target tree on the right. The source tree includes nodes like 'SHIP_LOOP', 'ORDER_LOOP', 'PACK_LOOP', 'ITEM_LOOP', and 'IT01'. The target tree includes nodes like '10 M HL Loop [Hierarchical Level]', '10 M HL [Hierarchical Level]', '20 O LIN [Item Identification]', '30 O SN1 [Item Detail (Shipment)]', '40 O SLN [Subline Item Detail]', '50 O PRF [Purchase Order Reference]', '60 O PO4 [Item Physical Details]', '70 O PID [Product/Item Description]', '80 O MEA [Measurements]', '90 O PWK [Paperwork]', and '100 O PKG [Marking, Packaging, Loading]'. A 'Global Variable Name' table is visible on the right side of the interface.

Global Variable Name	Scope	Data Type	Local Variab...	Scope	D
TotalNumEmployees	Ses...	Integer			
sverrorcount	Ses...	Real			
EmployeeCnt	Ses...	Integer			
tranerror	Ses...	Real			
SNIPType5	Group	Character			
dimerorcount	Ses...	Real			

Additional qualification can be used within the ForEach command

Mapping Hierarchical Loops

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window displays a Data Transformation Map for a source file named 'Data Format\WDICONF\HLLAB_DICTIONARY\WDICONF\HLLAB'. The source tree on the left includes nodes like H00, H01, SHIP_LOOP, ORDER_LOOP, PA01, ITEM_LOOP, and ITO1. The target tree on the right includes nodes like 10 M HL Loop, 10 M HL, 20 O LIN, 30 O SN1, 40 O SLN, 50 O PRF, 60 O PO4, 70 O PID, 80 O MEA, 90 O PWK, and 100 O PKG. The central mapping area shows a detailed view of the '10 M HL Loop [Hierarchical Level]' node, with a dropdown menu open showing 'HLLevel (O*)' and 'HLLevel (T*)'. A table of global variables is also visible on the right side of the mapping area.

Global Variable Name	Scope	Data Type	Local Variab...	Scope	D
TotalNumEmployees	Ses...	Integer			
svccerrorcount	Ses...	Real			
EmployeeCnt	Ses...	Integer			
tranerror	Ses...	Real			
SNIPType5	Group	Character			
dmerrorcount	Ses...	Real			

To create a sibling node, select the “Insert Peer” command. After selection of Insert Peer, select an HL level code from the drop-down list. The list contains valid codes from the code list for HL03 (element 735). The HL loop is inserted as a sibling of the current HL loop with an HLLevel command. Continue adding the parent child relationships until the hierarchy is complete.

Section

HL Loop Qualification

Mapping Hierarchical Loops

- Qualification HL Loop Mapping
 - ▶ Multiple Occurrence
 - Occurrence
 - Value
 - Expression
 - ▶ Single Occurrence



Within the HLLLevel command, you can add qualification for this level. For Target based maps you must first do a Multiple Occurrence qualification. Within the Multiple Occurrence qualification you can define additional qualification by occurrence, value, and expression. No qualification would be considered single occurrence for the HLLLevel.

Mapping Hierarchical Loops

The screenshot displays the WebSphere Data Interchange for Multiplatforms V3.3.3 interface. The main window shows a Data Transformation Map (TEST5) with a source document structure on the left and a target document structure on the right. The source document includes elements like SHIP_LOOP, ORDER_LOOP, PACK_LOOP, ITEM_LOOP, and IT01. The target document includes elements like 10 M HL Loop [Hierarchical Level], 10 M HL [Hierarchical Level], 20 O LIN [Item Identification], 30 O SN1 [Item Detail (Shipment)], 40 O SLN [Subline Item Detail], 50 O PRF [Purchase Order Reference], 60 O PO4 [Item Physical Details], 70 O PID [Product/Item Description], 80 O MEA [Measurements], 90 O PWK [Paperwork], 100 O PKG [Marking, Packaging, Loading], and 110 O TD1 [Carrier Details (Quantity and Weight)].

A right-click context menu is open over the HLLevel command in the source document. The menu options include: Open..., HL Qualification [ID Number], ForEach... [Parent ID Number], Insert Within... [Level Code], Delete, Expand All, Collapse All, 20 C Find..., 30 C Find..., and 40 O SLN [Subline Item Detail]. A blue callout bubble with the text "Right Click" points to the HLLevel command.

The bottom right corner of the screenshot shows a table with the following columns: Global Variable Name, Scope, Data Type, and Local Variab... Scope. The table contains the following data:

Global Variable Name	Scope	Data Type	Local Variab... Scope
TotalNumEmployees	Ses...	Integer	
sverrorcount	Ses...	Real	
EmployeeCnt	Ses...	Integer	
trerror	Ses...	Real	
SNIPType5	Group	Character	
clmerrorcount	Ses...	Real	

The bottom of the screenshot shows the Windows taskbar with the Start button, system tray, and the date/time (8:33 AM Friday 3/2/2007). The page number 30 and the text "Mapping Hierarchical Loops © 2007 IBM Corporation" are visible at the bottom.

This is a target based map with application data as the source document and EDI as the target document. The Shipment level has been defined and you need to identify what determines a shipment level should be created. This is called qualification. To add a multiple occurrence qualification, right click on the HLLevel command, and select the ForEach command.

Mapping Hierarchical Loops

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window displays a mapping configuration for a Data Transformation Map. The source window on the left shows a tree structure of loops: SHIP_LOOP, ORDER_LOOP, PACK_LOOP, ITEM_LOOP, and IT01. The target window on the right shows a list of hierarchical levels: 10 M HL [Hierarchical Level], 10 M HL [Hierarchical Level], 20 O LIN [Item Identification], 30 O SN1 [Item Detail (Shipment)], 40 O SLN [Subline Item Detail], 50 O PRF [Purchase Order Reference], 60 O PO4 [Item Physical Details], 70 O PID [Product/Item Description], 80 O MEA [Measurements], 90 O PWK [Paperwork], 100 O PKG [Marking, Packaging, Loading], and 110 O TD1 [Carrier Details (Quantity and Weight)]. A Mapping Command Editor dialog is open, showing the command 'ForEach (SHIP_LOOP\)' with a question mark icon. The dialog has buttons for OK, Insert, and Cancel. The system tray at the bottom shows the date and time as Friday, 3/2/2007, 8:36 AM.

Select the compound element in the source window that should create a Shipment level, drag and drop to the ForEach command argument.

Mapping Hierarchical Loops

WebSphere Data Interchange for Multiplatforms V3.3 - [WDI33Server - Data Transformation Map - TEST5]

Source: Data Format[WDICONFHLLAB_DICTIONARY[WDICONFHLLAB
 H00 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 H01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 SHIP_LOOP [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 ORDER_LOOP [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 OR01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 PACK_LOOP [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 PA01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 ITEM_LOOP [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]
 IT01 [DI User Example - HL Loop ADVANCE SHIP NOTICE 856]

Target: Table 2
 10 M HL Loop [Hierarchical Level]
 HLLevel ("S")
 ForEach (SHIP_LOOP())
 10 M HL [Hierarchical Level]
 1 M 628 [Hierarchical ID Number]
 HLAutoMapped
 2 O 734 [Hierarchical Parent ID Number]
 HLAutoMapped
 3 M 735 [Hierarchical Level Code]
 HLAutoMapped
 4 O 736 [Hierarchical Child Code]
 HLAutoMapped
 20 O LIN [Item Identification]
 30 O SN1 [Item Detail (Shipment)]
 40 O SLN [Subline Item Detail]
 50 O PRF [Purchase Order Reference]

Global Variable Name	Scope	Data Type	Local Variab...	Scope
TotalNumEmployees	Ses...	Integer		
svccerrorcount	Ses...	Real		
EmployeeCnt	Ses...	Integer		
trancerror	Ses...	Real		
SNIPType5	Group	Character		
cmerrorcount	Ses...	Real		

Ready

start

EN 8:39 AM
 Friday
 3/2/2007

Mapping Hierarchical Loops © 2007 IBM Corporation

For each SHIP_LOOP in the source document, an HL Loop will be created in the Target and the HL segment level code value will be 'S'.

Mapping Hierarchical Loops

The screenshot shows the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main workspace displays a Data Transformation Map (DTM) with a source tree on the left and a target tree on the right. The source tree includes nodes like H00, SHIP_LOOP, ORDER_LOOP, PACK_LOOP, ITEM_LOOP, and ITO1. The target tree includes nodes like 10 M HL Loop, 20 O LIN, 30 O SN1, 40 O SLN, 50 O PRF, 60 O PO4, 70 O PID, 80 O MEA, 90 O PWK, 100 O PKG, and 110 O TD1. A context menu is open over the '10 M HL Loop' node in the target tree, showing options like 'Open...', 'ForEach...', 'Qualify', 'Insert Within', 'Delete', 'Expand All', 'Collapse All', and 'Find...'. The 'Qualify' option is highlighted, with a sub-menu showing 'By Occurrence...', 'By Value...', and 'By Expression...'. A table of global variables is visible on the right side of the interface.

Global Variable Name	Scope	Data Type	Local Variab...	Scope	D.
TotalNumEmployees	Ses...	Integer			
svcerrortcount	Ses...	Real			
EmployeeCnt	Ses...	Integer			
tranerror	Ses...	Real			
SNIPType5	Group	Character			
dmerrorcount	Ses...	Real			

Additional qualification can be added under the ForEach command.

Mapping Hierarchical Loops

The screenshot displays the IBM WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window shows a source tree on the left and a target tree on the right. A 'Mapping Command Editor' dialog box is open, showing the 'Qualify' parameter set to 'Occurrence (sourcePath) EQ num'. The dialog box has 'OK', 'Insert', and 'Cancel' buttons. The background shows a table with columns for 'Global Variable Name', 'Scope', 'Data Type', and 'Local Variab...'. The table contains the following data:

Global Variable Name	Scope	Data Type	Local Variab...	Scope	D
TotalNumEmployees	Ses...	Integer			
svccerrorcount	Ses...	Real			
EmployeeCnt	Ses...	Integer			
tranerror	Ses...	Real			
CUSTOMER...	Ses...	Character			

The bottom of the screenshot shows the Windows taskbar with the date and time: 8:43 AM, Friday, 3/2/2007. The page number 34 and the text 'Mapping Hierarchical Loops © 2007 IBM Corporation' are visible at the bottom.

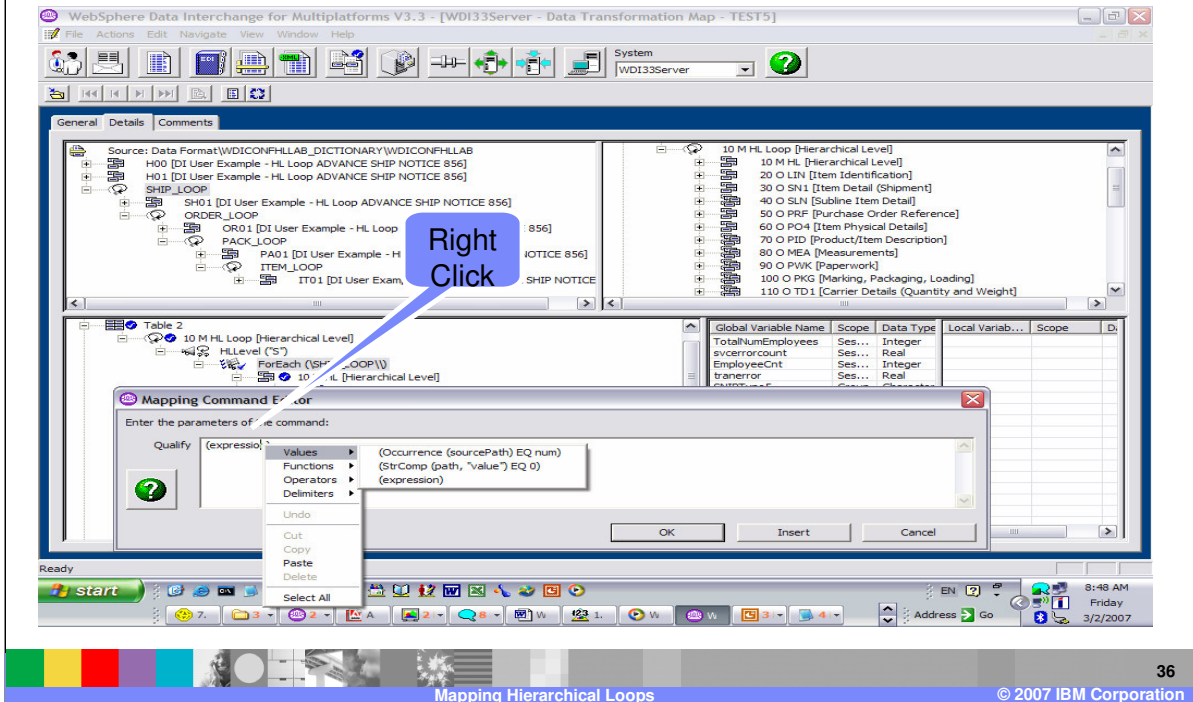
This is the format for the Qualify by Occurrence command. Drag and drop to the sourcePath and enter the occurrence number in the num argument.

Mapping Hierarchical Loops

The screenshot displays the WebSphere Data Interchange for Multiplatforms V3.3 interface. The main window shows a mapping tree with a source tree on the left and a target tree on the right. The source tree includes nodes like H00, H01, SHIP_LOOP, ORDER_LOOP, PACK_LOOP, PA01, ITEM_LOOP, and IT01. The target tree includes nodes like 10 M HL Loop, 10 M HL, 20 O LBN, 30 O SN1, 40 O SLN, 50 O PRF, 60 O PO4, 70 O PID, 80 O MEA, 90 O PWK, 100 O PKG, and 110 O TD1. A 'Mapping Command Editor' dialog box is open, showing the command: `Qualify {trComp (path, "value") EQ 0}`. The dialog has 'OK', 'Insert', and 'Cancel' buttons. The system tray at the bottom shows the date as Friday, 3/2/2007, and the time as 8:45 AM.

This is the format of the Qualify by Value command. Drag and drop to the path argument and enter the value in the "value" argument.

Mapping Hierarchical Loops



With Qualify by Expression you can right click on the expression argument to see available commands and function that can be used for the expression. This is a list of values.

Mapping Hierarchical Loops

The screenshot shows the 'WebSphere Data Interchange for Multiplatforms V3.3 - FWDJ33Server - Data Transformation Map - TEST5' interface. A 'Mapping Command Editor' dialog is open, displaying a list of functions. The functions list includes:

- Char
- Concat
- Created
- Date
- DateCnv
- Exit
- Find
- Found
- GetProperty
- HexDecode
- HexEncode
- IsEmpty
- IsEmpty
- Left
- Length
- Lower
- Number
- NumFormat
- Occurrence
- Overlay
- Right
- Round
- StrComp
- StrCompI
- StrCompN
- StrCompNI
- SubString
- Time
- Translate
- TrimLeft
- TrimRight
- Truncate
- Upper
- Validate

The Mapping Command Editor dialog has a text field for 'Qualify (expression\)' and a 'Functions' dropdown menu. The background shows a data tree with nodes like '10 M HL Loop [Hierarchical Level]' and a table with columns 'Global Variable Name', 'Scope', 'Data Type', and 'Local Variab...'. The table contains entries like 'TotalNumEmployees' (Integer, Real), 'svccorrcount' (Integer, Real), 'EmployeeCnt' (Integer, Real), and 'tranerror' (Integer, Real).

Mapping Hierarchical Loops

This is a list of functions. Functions can be combined to create an expression.

Summary

- Special HL Loop mapping defines the hierarchy to allow WDI translation to generate the HL Segment values.
- Data Transformation Maps provide special Qualification type for HL Loop hierarchy definition which allow you to visually create the hierarchy and provide additional qualifications under the multiple occurrence qualification.



Special HL Loop mapping defines the hierarchy to allow WDI translation to generate the HL Segment values. Data Transformation Maps provide special Qualification type for HL Loop hierarchy definition which allow you to visually create the hierarchy and provide additional qualifications under the multiple occurrence qualification.

Reference

- More information can be found in the WDI V3.3 Mapping Guide, Appendix B. Hierarchical loops.



More information can be found in the WebSphere Data Interchange Version 3.3 Mapping Guide, Appendix B. Hierarchical loops.

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
ef (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.

