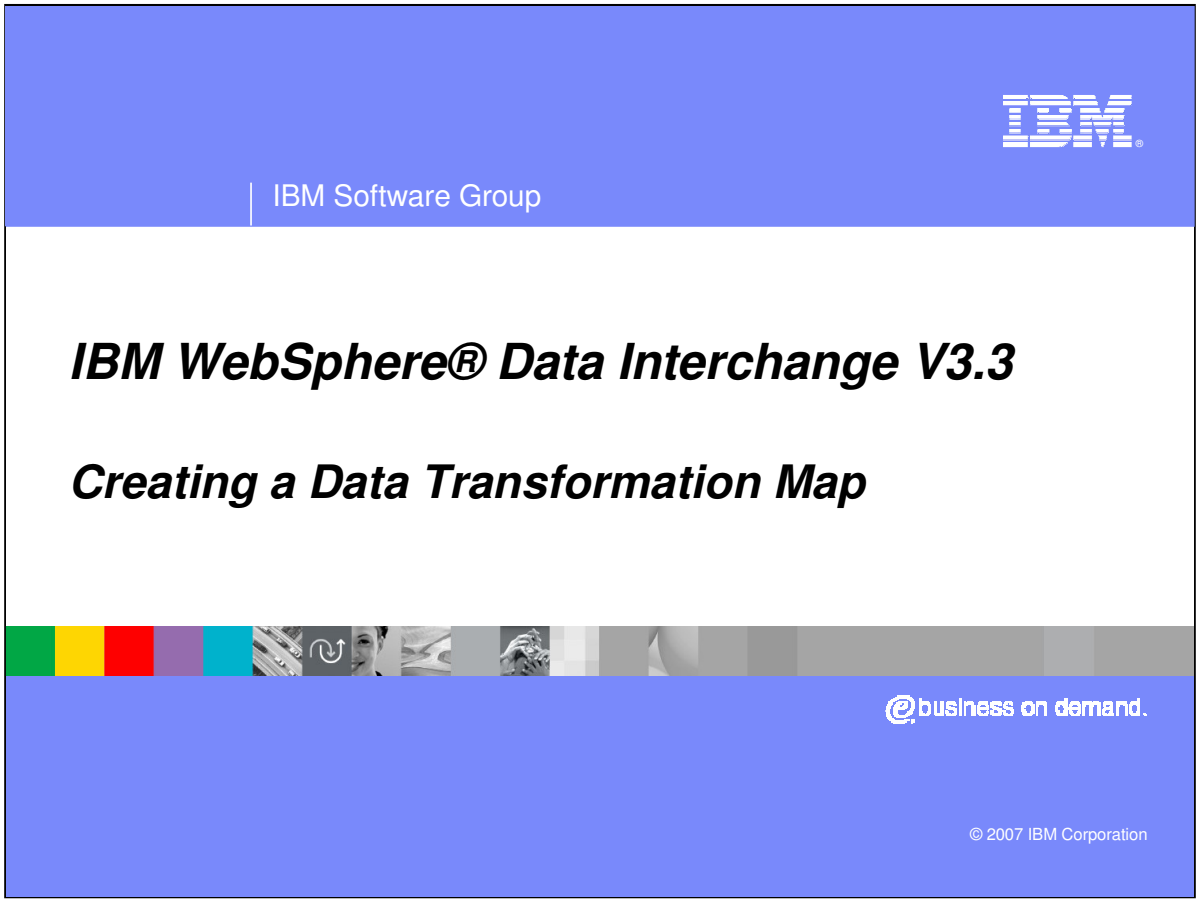


This presentation will demonstrate how to create a Data Transformation Map.

Agenda

- Overview mapping objects
- Data Transformation (DT) Maps
- Common mapping functions
- Summary and references

The presentation will give an overview of the mapping objects, review Data Transformation mapping process, and cover some common mapping functions.

Creating a Data Transformation Map

- Data Transformation mapping process.
  - Load or Define the source document type
    - Electronic Data Interchange (EDI), application data, or XML.
  - Load or Define the target document type
    - EDI, application data, or XML
  - Identify the map base.
    - Mapping based on Source Document type.
    - Mapping based on Target Document type.

Data Transformation maps have no “real” direction association. There is a source document and a target document. The Data Transformation processing flow is deenvelope, translate or transform, envelope. The source message is deenveloped and transformed to the target message. The target message is enveloped.

With WDI Client mapping you have the choice to select which document type you want to use for the mapping commands.

Creating a Data Transformation Map

- Identify the map base
  - Source Based Maps
    - In a source based Data Transformation Map, the map is constructed based on the order elements are defined in the source document definition.
    - The Source document is presented for mapping in the mapping window.
    - The mapping commands appear under the Source document elements.
    - Elements can be simple or compound. Simple elements are any element that contains a single piece of data, such as a Data Format Field, EDI Standard Data Element, or XML simple element. Compound elements are elements that contain other elements. Examples of compound elements are Data Format Records and EDI Standard Segments.

In a source based Data Transformation Map, the map is constructed based on the order elements are defined in the source document definition. The Source document is presented for mapping in the mapping window. The mapping commands appear under the Source document elements. Elements can be simple or compound. Simple elements are any element that contains a single piece of data, such as a Data Format Field, EDI Standard Data Element, or XML simple element. Compound elements are elements that contain other elements. Examples of compound elements are Data Format Records and EDI Standard Segments.

Creating a Data Transformation Map

- Identify the map base
  - Source Based Maps
    - The source document is parsed and an abstract message is created. The parser constructs the input message into the hierarchical representation using the source document definition.
    - The Abstract Message Model (AMM) basically represents messages as a tree of elements. All element have a name. Elements may be compound or simple. If an element has child element it is a compound element. If an element has no child elements, then it is a simple element and will have a type and a value.
    - The model of a message presented to a message processing node is that of a parse tree of syntax elements.
    - Translation occurs with execution of mapping commands based on the hierarchy and order of the elements as defined in the source document definition (which was used during the mapping process).

The source document is parsed and an abstract message is created. The parser constructs the input message into the hierarchical representation using the source document definition. The Abstract Message Model (AMM) basically represents messages as a tree of elements. All element have a name. Elements may be compound or simple. If an element has child element it is a compound element. If an element has no child elements, then it is a simple element and will have a type and a value. The model of a message presented to a message processing node is that of a parse tree of syntax elements. Translation occurs with execution of mapping commands based on the hierarchy and order of the elements as defined in the source document definition (which was used during the mapping process).

Creating a Data Transformation Map

- Identify the map base
  - Target Based Maps
    - In a target based Data Transformation Map, the map is constructed based on the order elements are defined in the target document definition.
    - The Target document is presented for mapping in the mapping window.
    - The mapping commands appear under the Target document elements.
    - Elements can be simple or compound. Simple elements are any element that contains a single piece of data, such as a Data Format Field, EDI Standard Data Element, or XML simple element. Compound elements are elements that contain other elements. Examples of compound elements are Data Format Records and EDI Standard Segments.

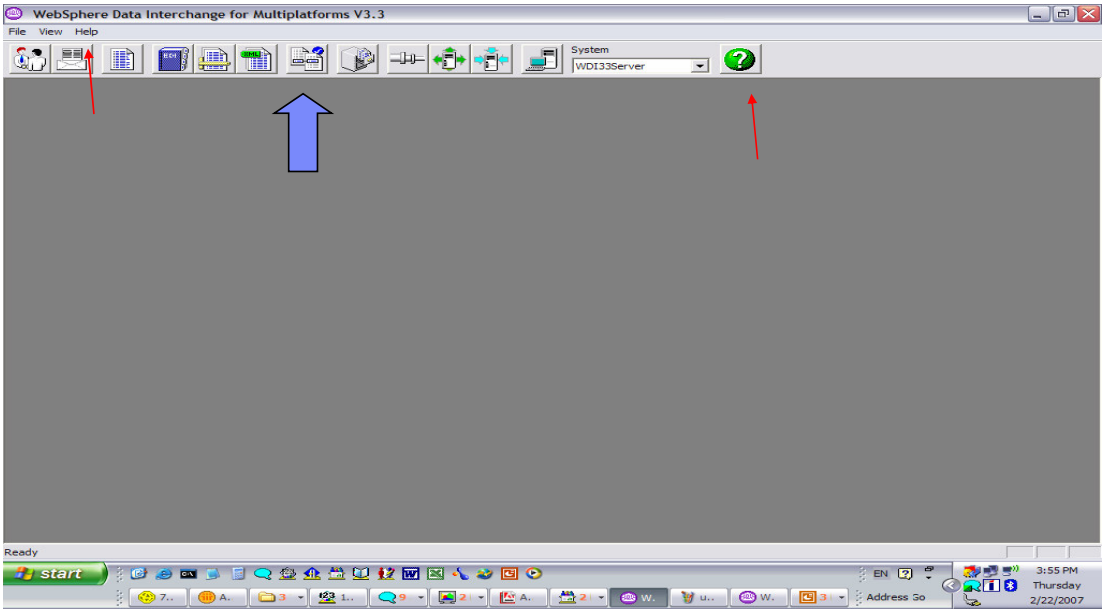
In a target based Data Transformation Map, the map is constructed based on the order elements are defined in the target document definition. The Target document is presented for mapping in the mapping window. The mapping commands appear under the Target document elements. Elements can be simple or compound. Simple elements are any element that contains a single piece of data, such as a Data Format Field, EDI Standard Data Element, or XML simple element. Compound elements are elements that contain other elements. Examples of compound elements are Data Format Records and EDI Standard Segments.

Creating a Data Transformation Map

- Identify the map base
  - Target Based Maps
    - The source document is parsed and an abstract message is created. The parser constructs the input message into the hierarchical representation using the source document definition.
    - The Abstract Message Model (AMM) basically represents messages as a tree of elements. All element have a name. Elements may be compound or simple. If an element has child element it is a compound element. If an element has no child elements, then it is a simple element and will have a type and a value.
    - The model of a message presented to a message processing node is that of a parse tree of syntax elements.
    - Translation occurs with execution of mapping commands based on the hierarchy and order of the elements as defined in the target document definition (which was used during the mapping process).

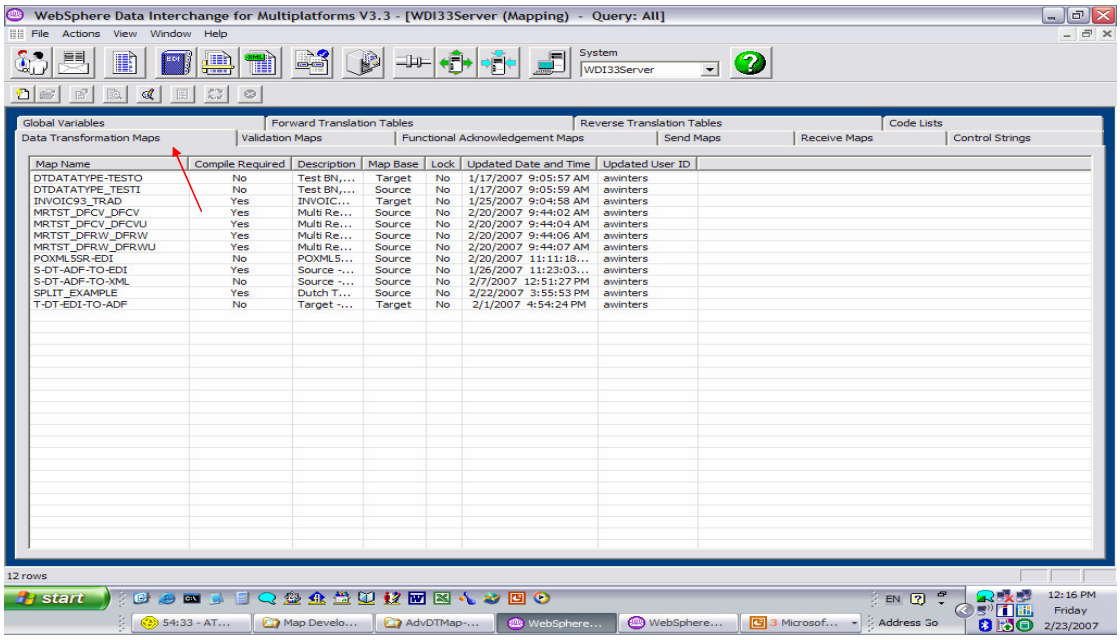
The source document is parsed and an abstract message is created. The parser constructs the input message into the hierarchical representation using the source document definition. The Abstract Message Model (AMM) basically represents messages as a tree of elements. All element have a name. Elements may be compound or simple. If an element has child element it is a compound element. If an element has no child elements, then it is a simple element and will have a type and a value. The model of a message presented to a message processing node is that of a parse tree of syntax elements. Translation occurs with execution of mapping commands based on the hierarchy and order of the elements as defined in the target document definition (which was used during the mapping process).

Creating a Data Transformation Map



The WebSphere Data Interchange (WDI) Client Mapping Functional Area. For general help you can click the question mark at the top, right or select Help->Contents.

Creating a Data Transformation Map

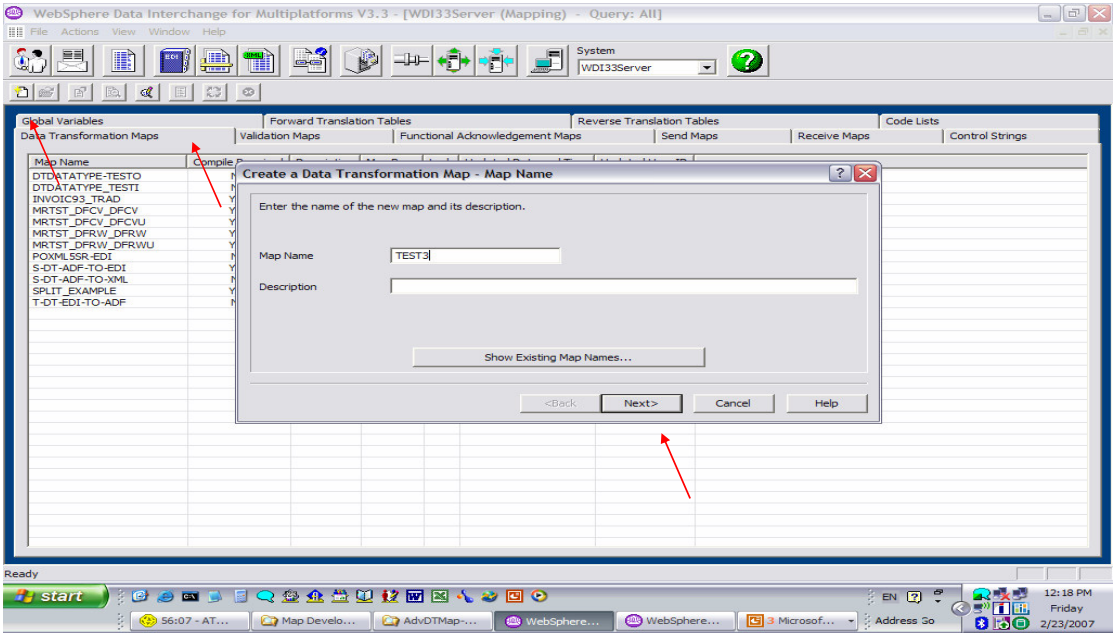


The WDI Client Mapping Functional Area, Data Transformation (DT) Maps tab contains a list of the DT maps.

Section

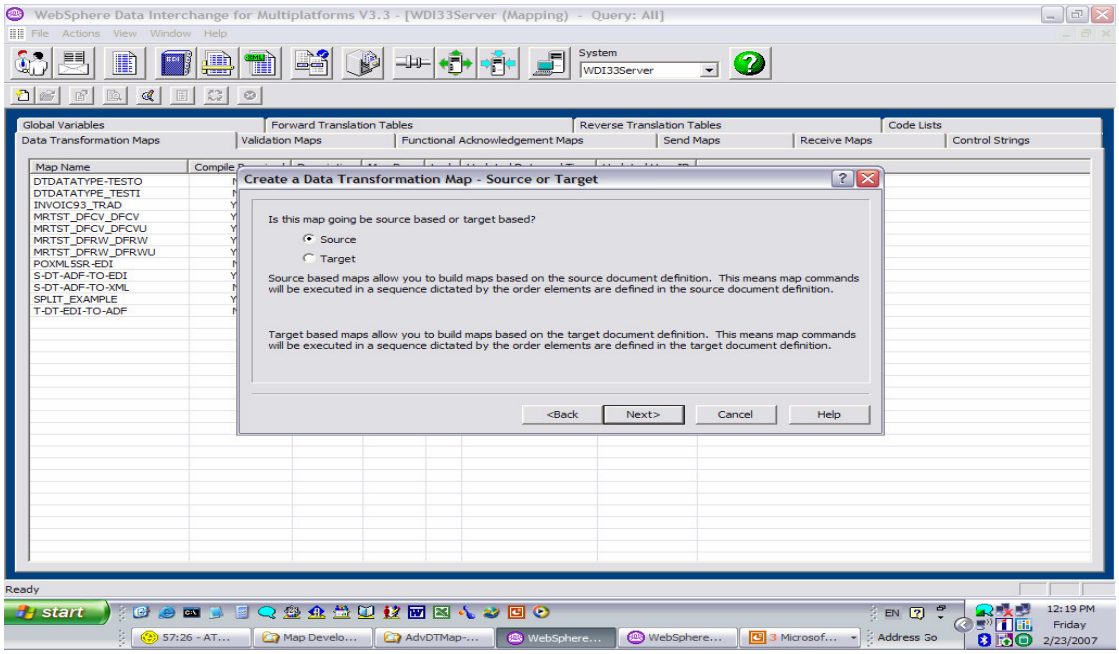
DT Maps

Creating a Data Transformation Map



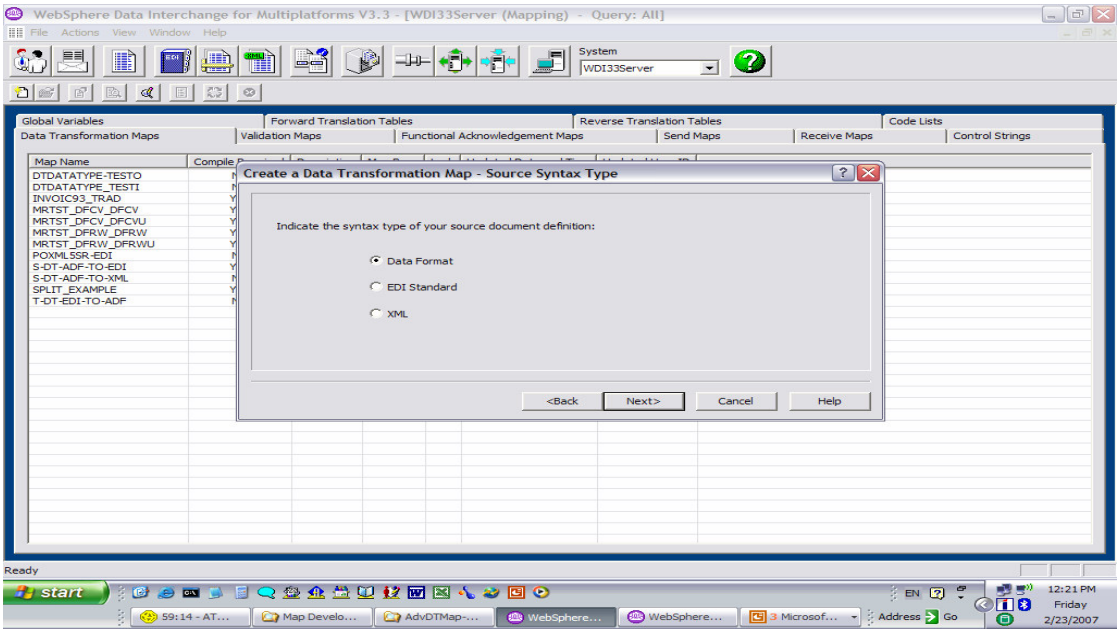
To add a Data Transformation map, navigate to the Data Transformation Maps tab, and use the New button from the list window tool bar. Enter the map name and optional description, and click Next.

Creating a Data Transformation Map



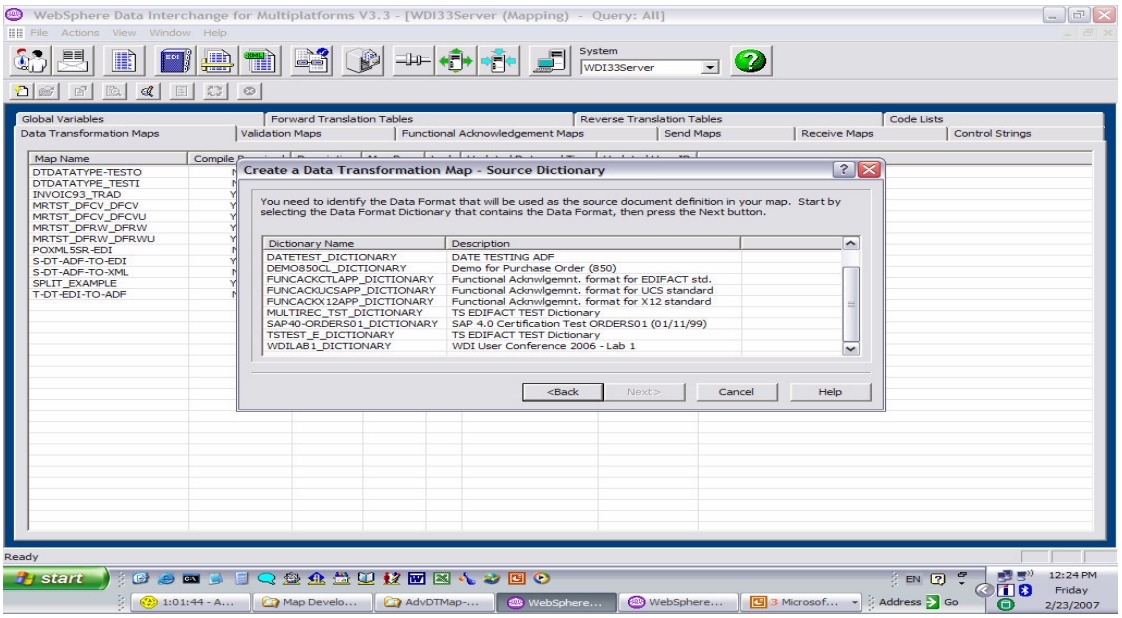
Identify the map base Source based or Target Based and click on Next.

Creating a Data Transformation Map



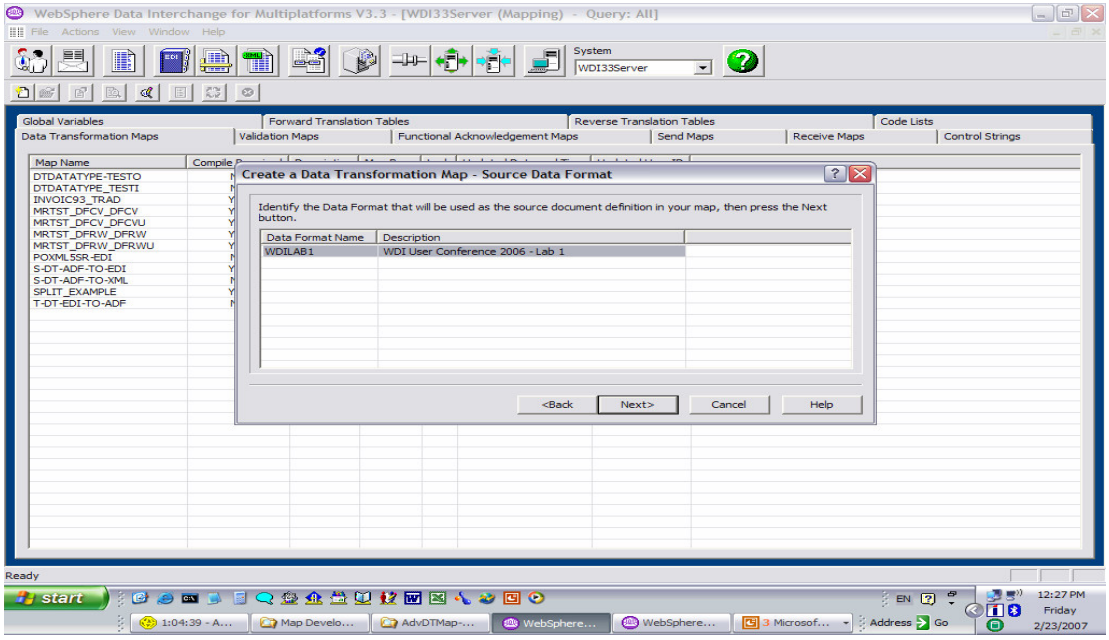
Select the Source document syntax type Data Format or Application Data, EDI, or XML and click on Next. This selection will affect what you will see in the next window.

Creating a Data Transformation Map



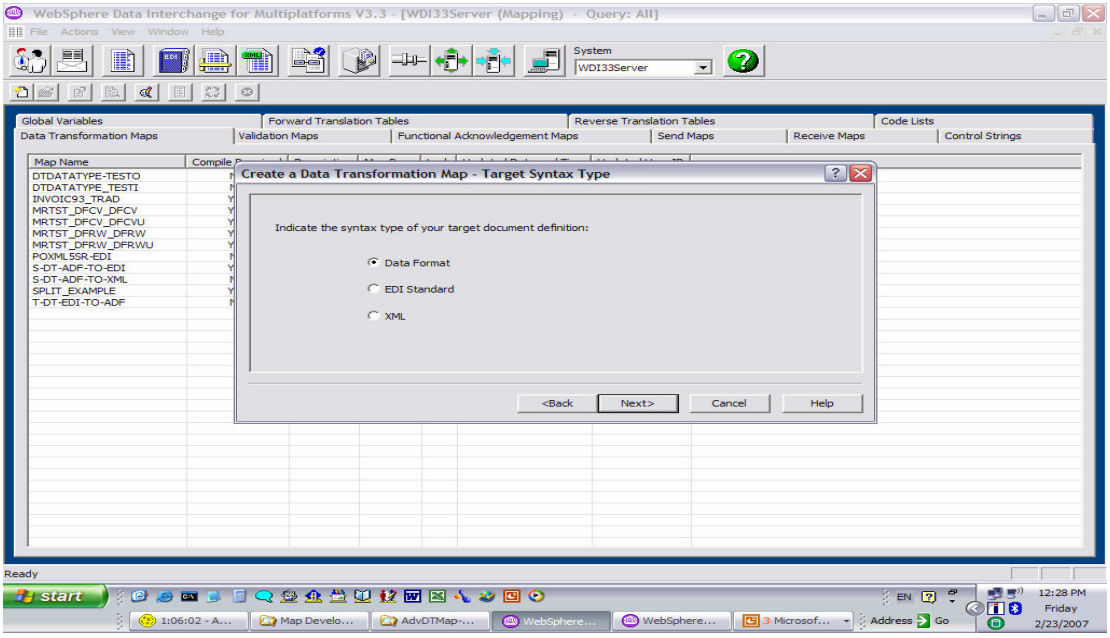
A list of Source Document Dictionaries is displayed. This selection is for the Source dictionary and document type and is based on your selection on the previous window. In this example, data formats was selected which displays a list of application data metadata definitions. This selection is for the Source dictionary and document type. Select the Source dictionary and click on Next.

Creating a Data Transformation Map



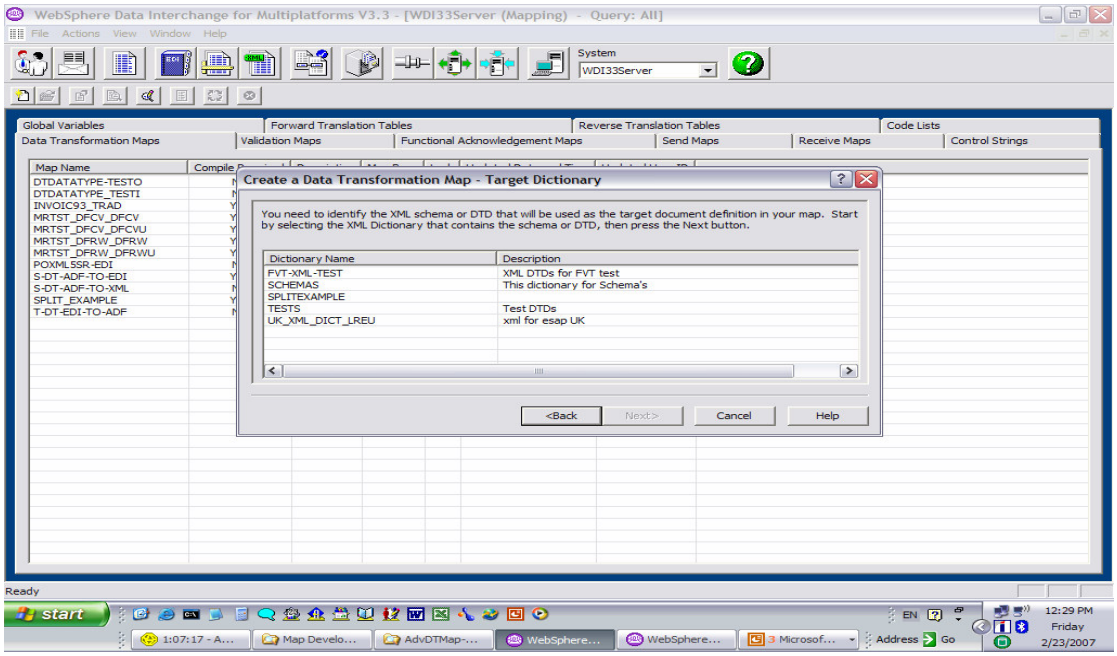
This selection is for the Source dictionary and document type. Select the Source document and click on Next.

Creating a Data Transformation Map



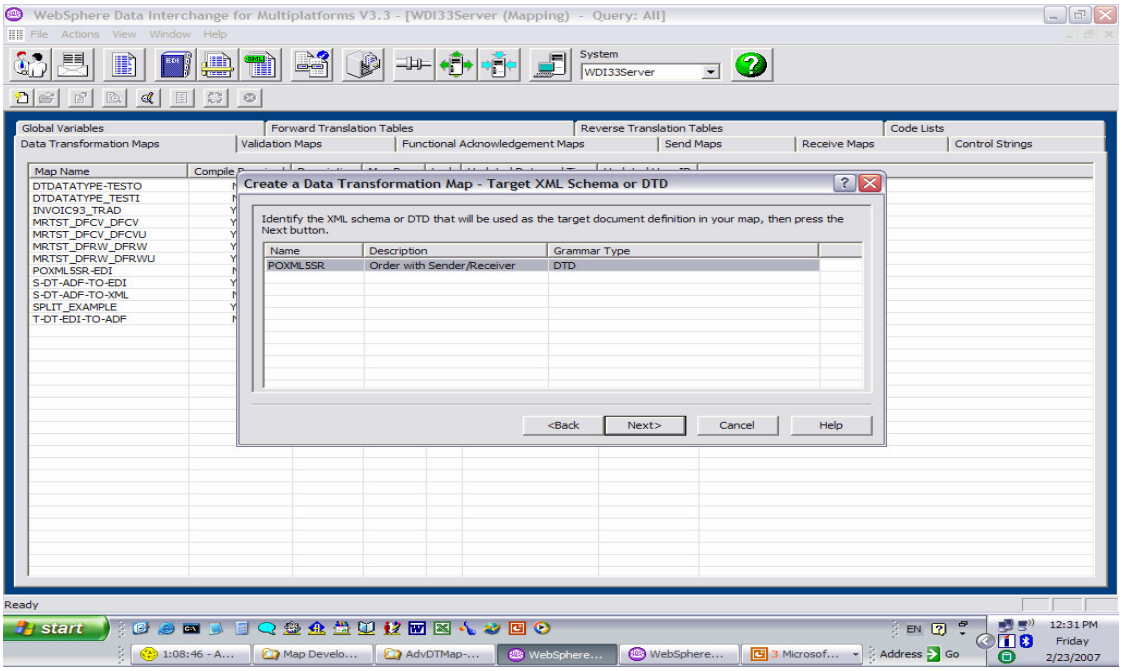
Select the Target document syntax type Data Format or Application Data, EDI, or XML and click on Next. This selection will affect what you will see in the next window.

Creating a Data Transformation Map



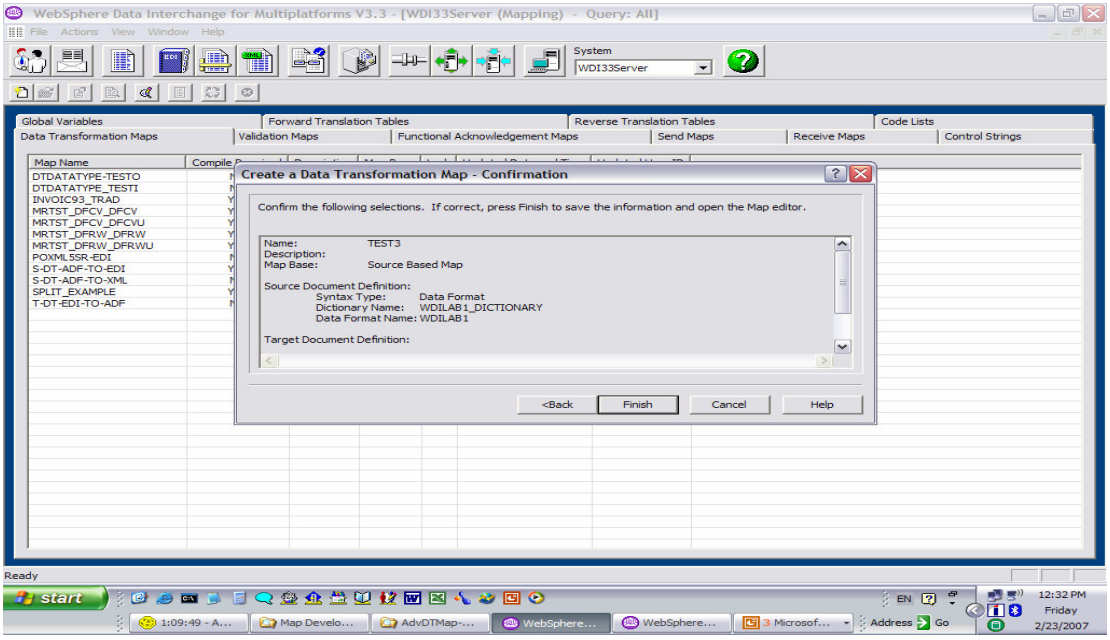
A list of Target Document Dictionaries is displayed. This selection is for the Target dictionary and document type and is based on your selection on the previous window. In this example, XML was selected which displays a list of XML metadata definitions. This selection is for the Target dictionary and document type. Select the Target dictionary and click on Next.

Creating a Data Transformation Map



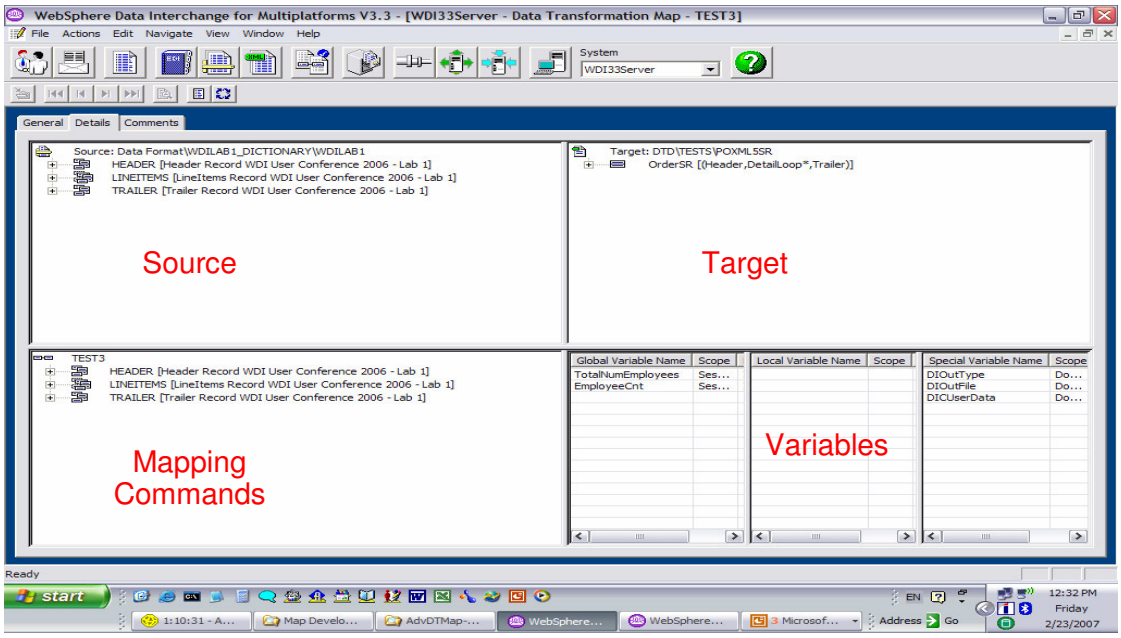
This selection is for the Target dictionary and document type. Select the Target document and click on Next.

Creating a Data Transformation Map



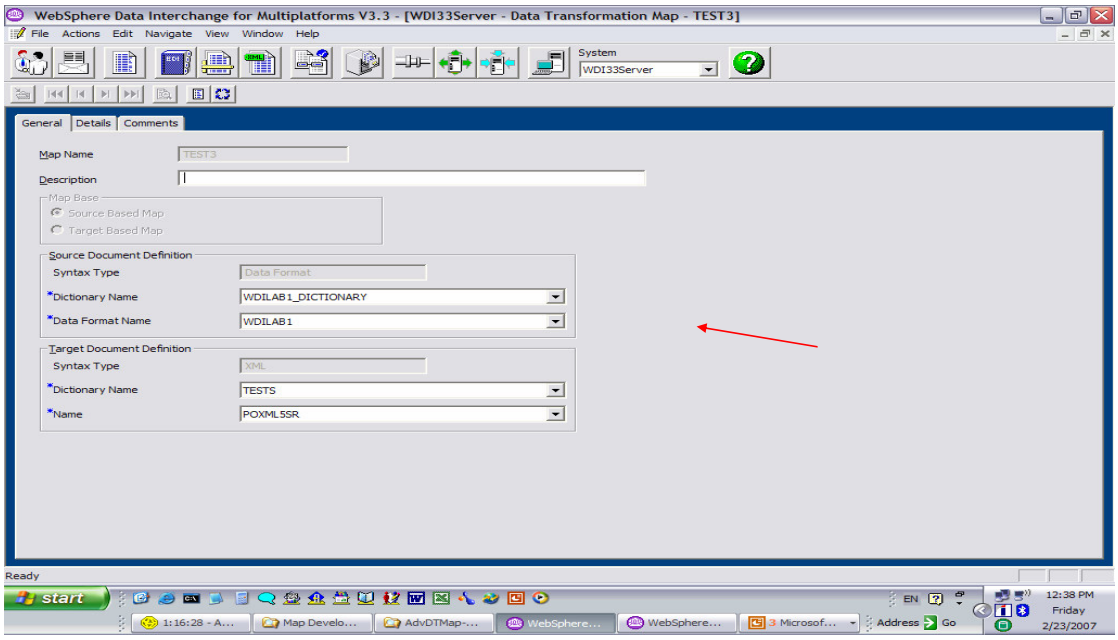
Confirm the Source and Target metadata definitions and click Finish.

Creating a Data Transformation Map



The mapping window displays with the Source document in the top left window and the Target document in the top right window. Mapping commands are located in the lower left window. In this example data format was identified as the Source and Source based mapping was selected as the map base. The Mapping commands window displays the Source document metadata definition because this is a source based map. Variables are located in the lower right window. Records, segments, and loops may be expanded by clicking on the plus sign or right click and select expand all.

Creating a Data Transformation Map



The General Tab shows the source and target metadata definitions. This may be modified which is useful for map creation and migration.

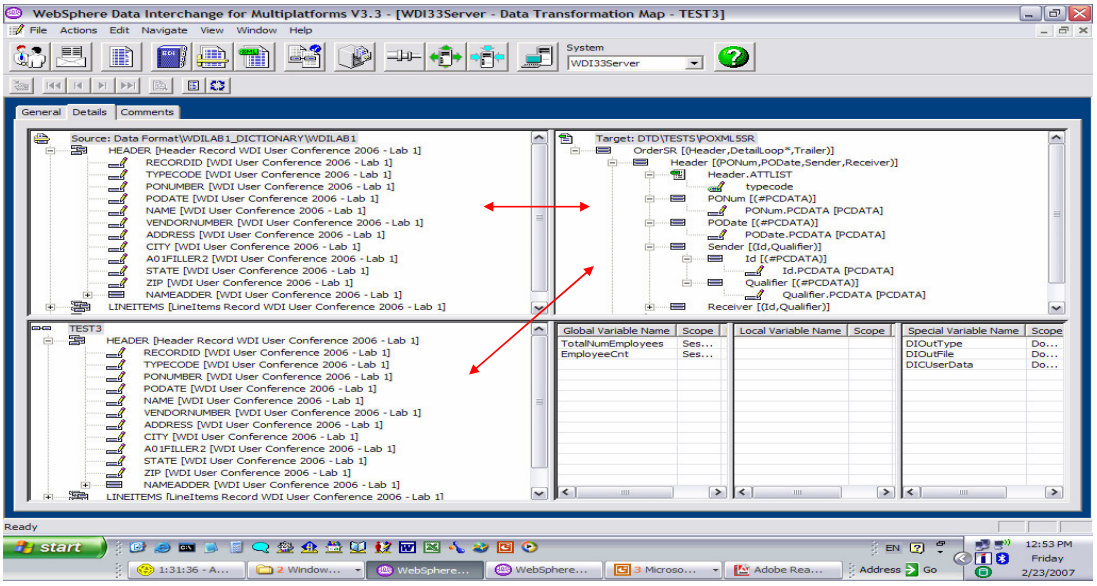
Creating a Data Transformation Map

- 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 / ElseIf / Else / EndIf
  - 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 mapping functions include: drag/drop, supplying literal values not found in the data, conditional mapping, formatting and validating values, and loop qualification.

Creating a Data Transformation Map

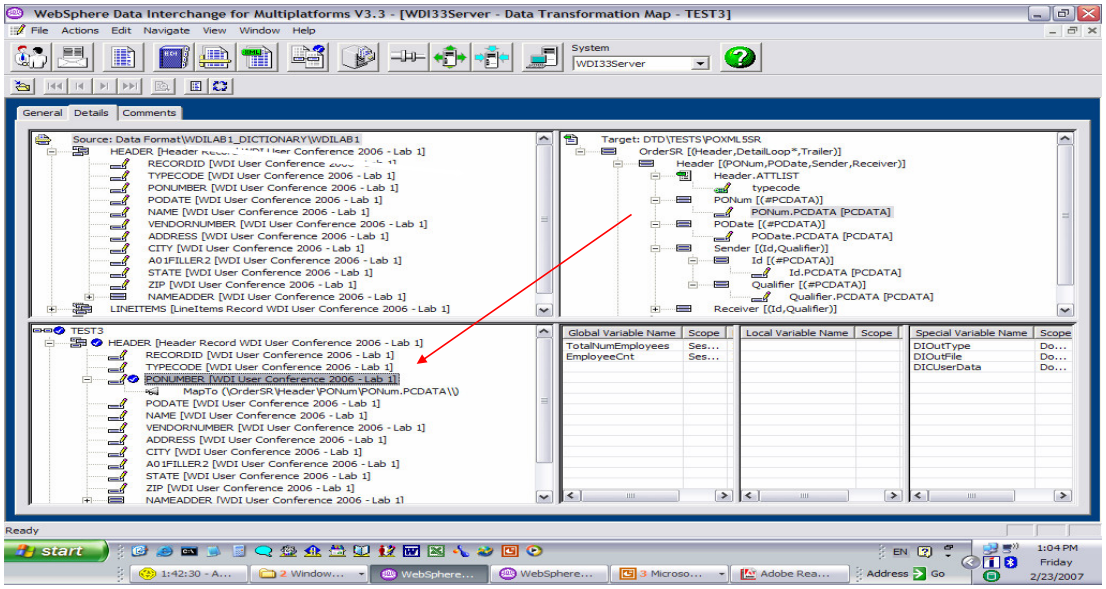
- drag / drop Data Transformation Map



Drag and Drop is the most common mapping function. With Data Transformation mapping, you can drag and drop using any direction that will result in a source element mapping to a target. A target can be a Target document element, variable, mapping command, or expression.

Creating a Data Transformation Map

- drag / drop Data Transformation Map (right to left)

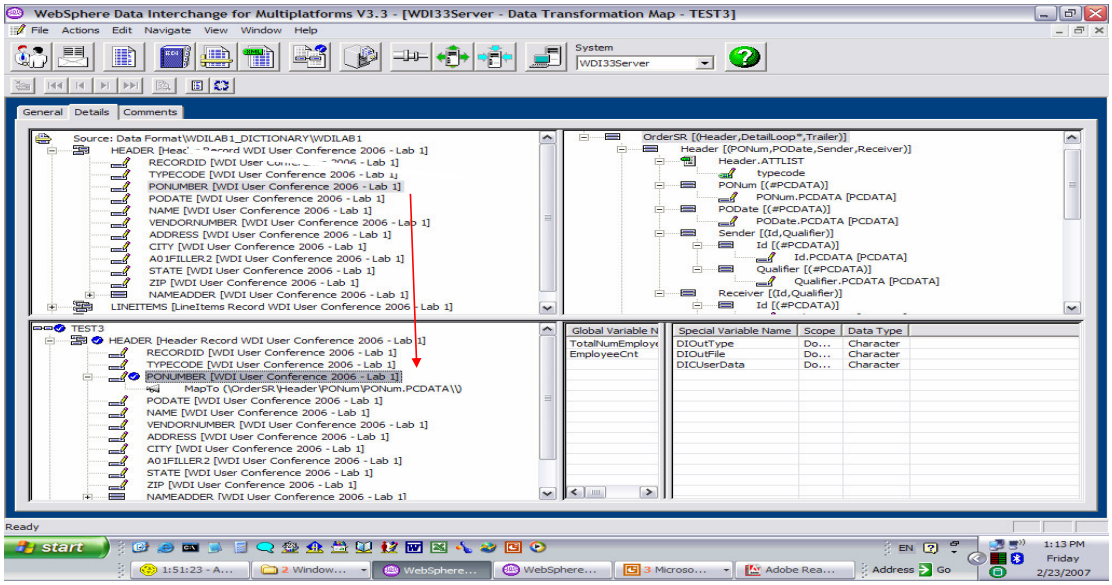


To map a source element to a target element, select the source or target element and drag/drop it on the source or target element. With this example the target element was selected in the Target window (upper right window) and was dropped on the source element in the mapping window (lower left window). The result is a MapTo() command displayed in the lower left window.

.

Creating a Data Transformation Map

- drag / drop Data Transformation Map (top to bottom)

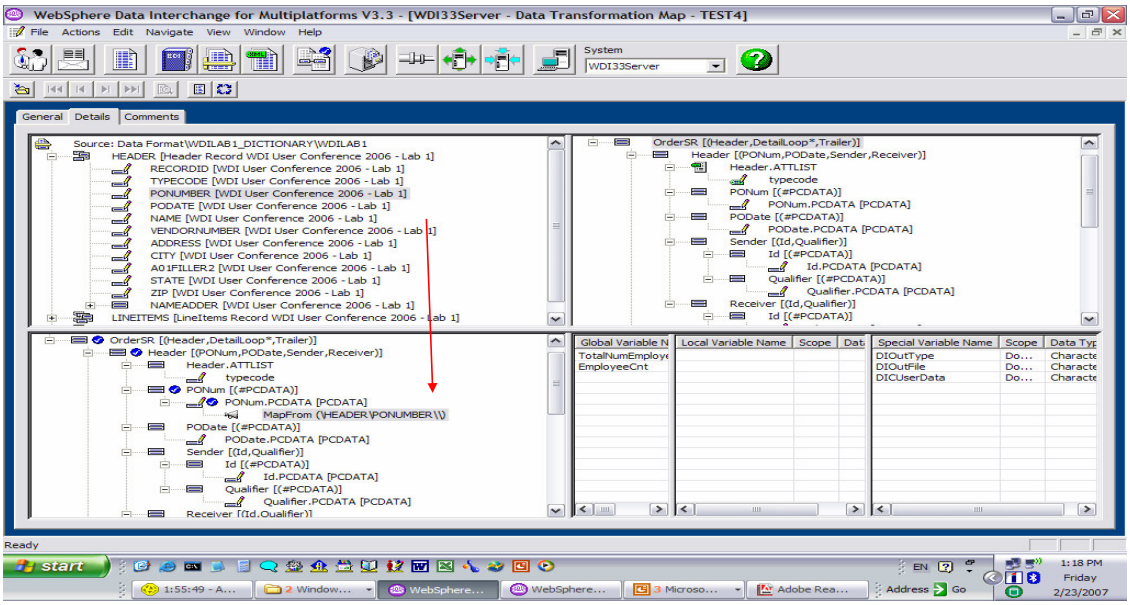


With this example the source element was selected in the Source window (upper left window) and was dropped on the source element in the mapping window (lower left window). This will NOT result in a source element to target mapping and is not allowed because this is a source based map and you cannot map a source element to a source element. You can however drag the source element to the Target window (upper right window).

.

Creating a Data Transformation Map

- drag / drop Data Transformation Map (top to bottom)

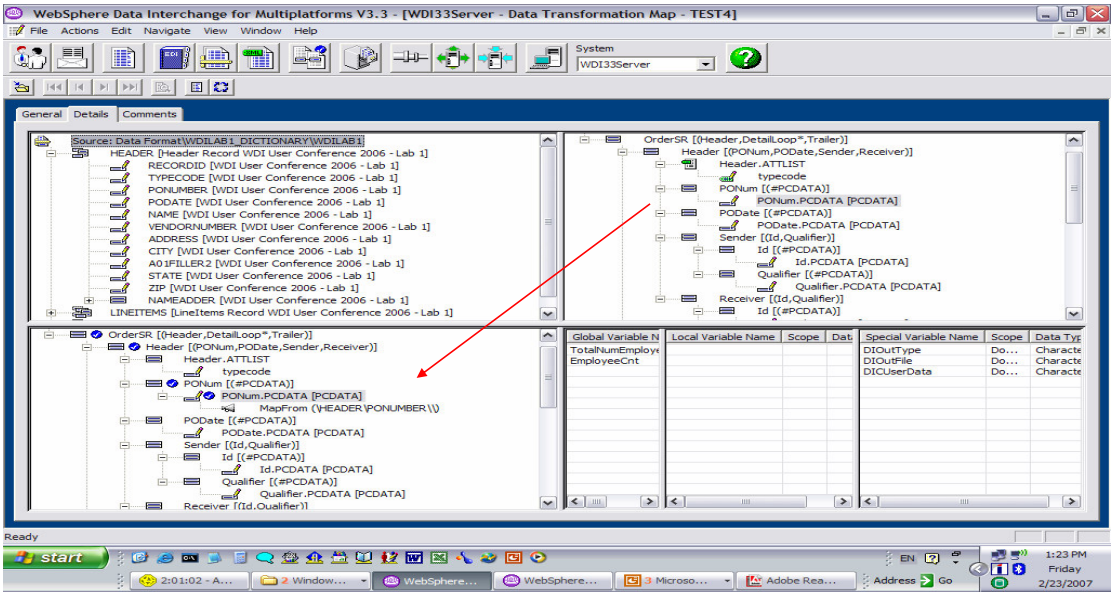


This is an example using the same source and target metadata definitions but the mapping is Target based. The source element was selected in the Source window (upper left window) and was dropped on the target element in the mapping window (lower left window). The result is a MapFrom() command displayed in the lower left window.

.

Creating a Data Transformation Map

- drag / drop Data Transformation Map (right to left)

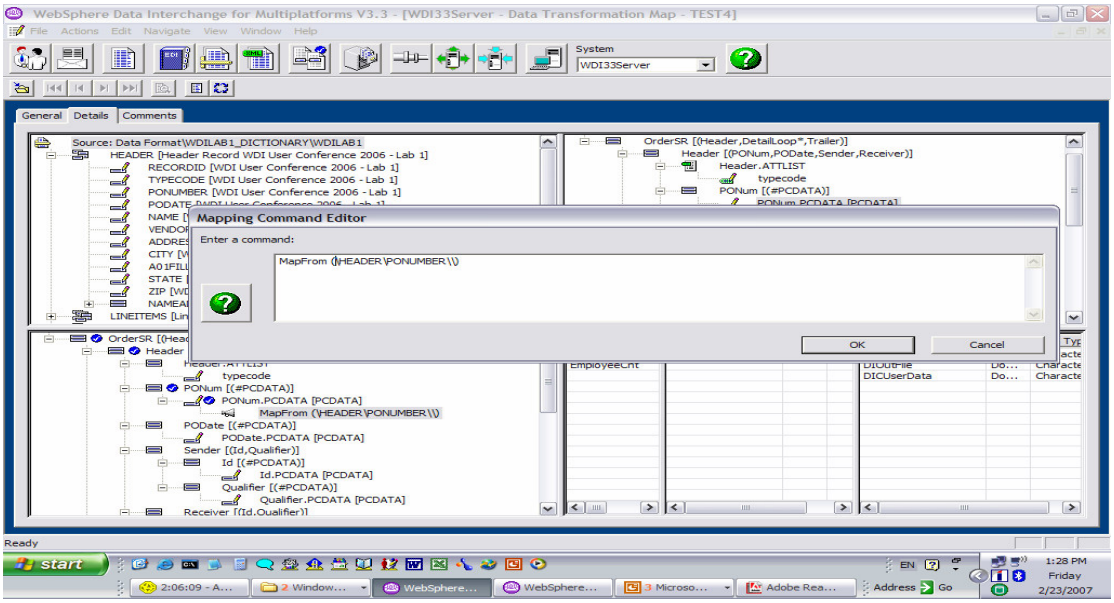


This is an example using the same source and target metadata definitions but the mapping is Target based. The target element was selected in the Target window (upper right window) and was dropped on the target element in the mapping window (lower left window). This will NOT result in a source element to target mapping and is not allowed. You can however drag the target element to the Source window (upper right window).

.

Creating a Data Transformation Map

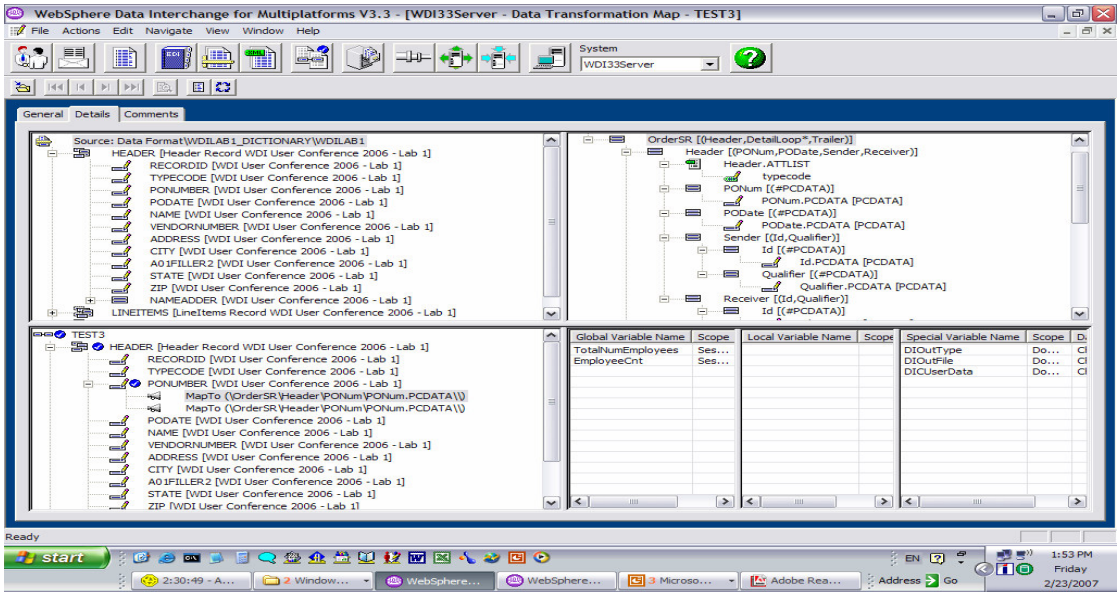
- drag / drop Data Transformation Map



You can modify the mapping commands in the mapping window by selecting the command and using double click. In this example you can assign a different source element by simply selecting the source element in the source window (upper left window) and drag the element to the target with is the MapFrom() command.

Creating a Data Transformation Map

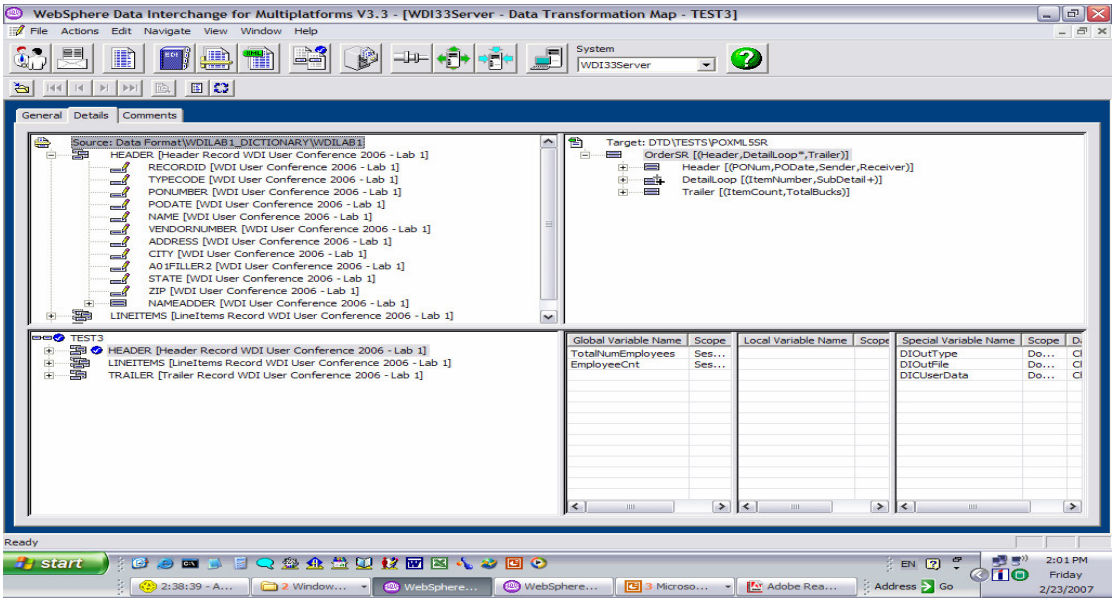
- drag / drop Data Transformation Map



Mapping commands may be moved and deleted. To delete a command select the command and press the Delete button on the keyboard. To move a command select the command and drag/drop the command to the new location. You may move the mapping to any other locating within the mapping window. You can also copy a command by selecting the command, press Shift key on the keyboard, then drag and drop within the mapping window. Automatic scrolling is a feature provided by WDI Client. To scroll in the mapping window during a drag/drop action, simply continue with the drag action and move to the bottom of the mapping window.

Creating a Data Transformation Map

- drag / drop Data Transformation Map



The WDI Client mapping wizard provides some automatic controls for expanding compound elements which can be useful. During a drag and drop action if the target element has not been expanded, you can continue the drag to the compound element containing the target element. This will cause the compound element to expand.

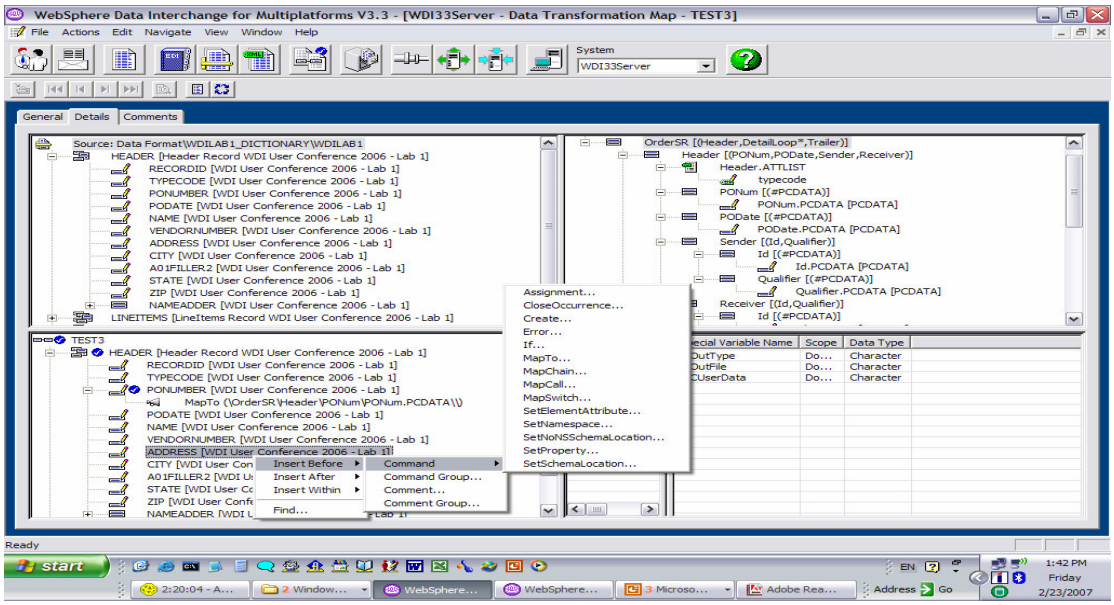
Creating a Data Transformation Map

- 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 / ElseIf / Else / EndIf
  - ➢ 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

The Assignment command can be used to assign a value to a target. A target can be a Target document element, variable, mapping command or expression.

Creating a Data Transformation Map

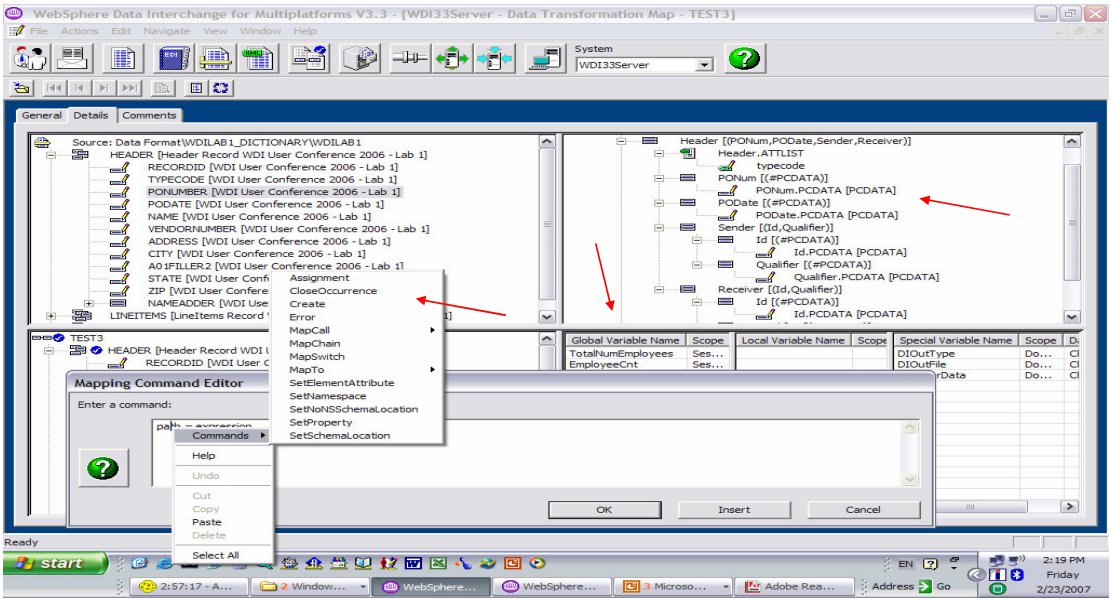
- Assignment Command



To add an Assignment command, right click an element in the mapping window. Select where you want the command to go before the current element, after the current element, or within the current element. Select Command and a list of available commands will be displayed for selection.

Creating a Data Transformation Map

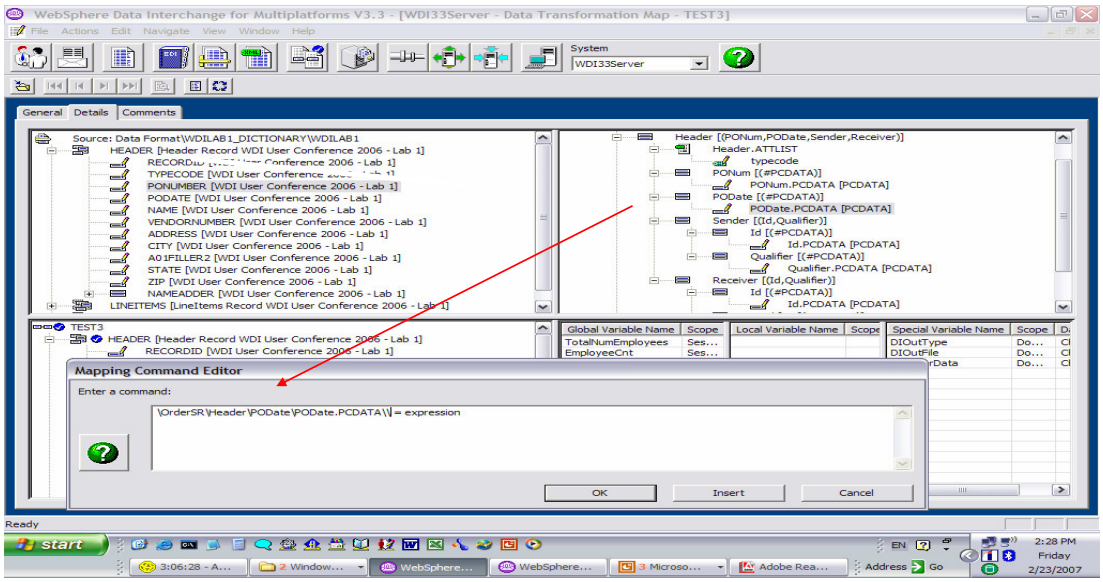
- Assignment Command (path = expression)



The format of the Assignment command looks simple but can contain complex commands and functions to achieve your results. Path is the target and expression is the source. A target can be a Target document element, variable, mapping command, or expression. To view the list of command options, right click the “path” parameter. To assign a value to a target element, select the target element and use drag/drop. To assign a value to a variable, select the variable and use drag/drop.

Creating a Data Transformation Map

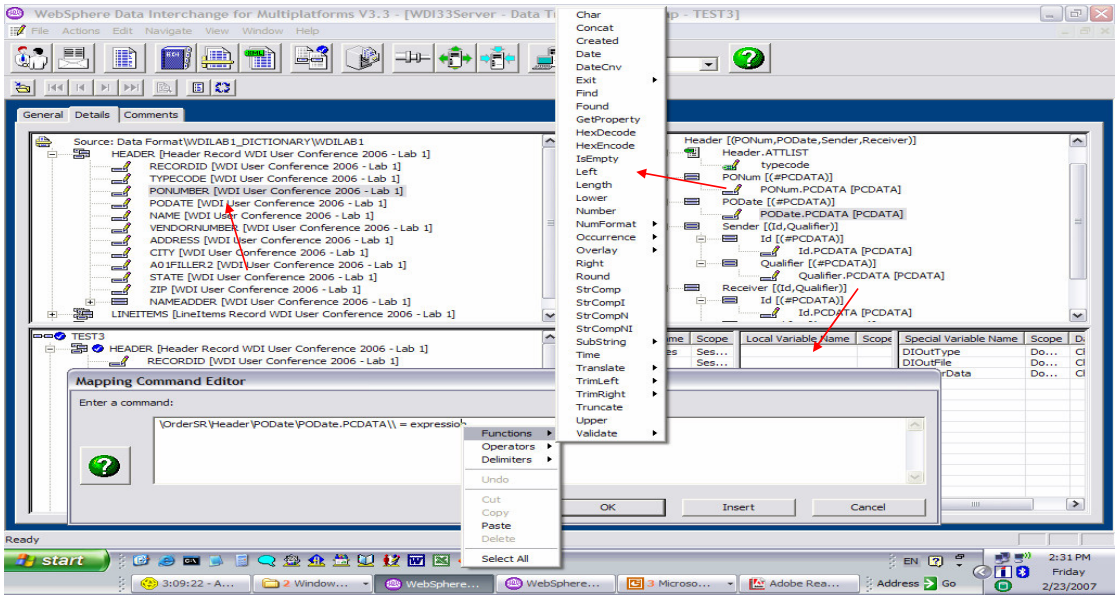
- Assignment Command



This is the results of using drag and drop for the target element.

Creating a Data Transformation Map

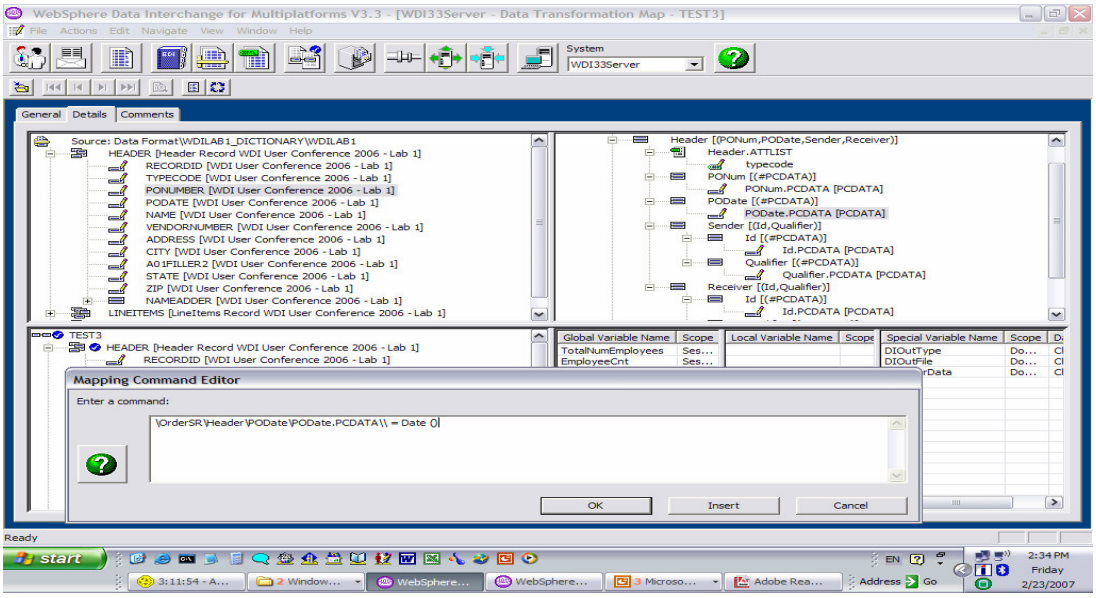
- Assignment Command



The source value is an expression. 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. To select a mapping function, select the function and the expression will be replaced with the function and format.

Creating a Data Transformation Map

- Assignment Command



With this example the Date function was selected which will return the system date and assign the system date to the target element.

Creating a Data Transformation Map

- 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 / ElseIf / Else / EndIf
  - ➢ 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

This presentation is a demonstration on creating Data Transformation Maps. The other common mapping functions are considered Advanced techniques and are reviewed in the Advanced mapping presentation.

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.

Template Revision: 04/25/2006 11:09 AM

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	WMQ	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.

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.