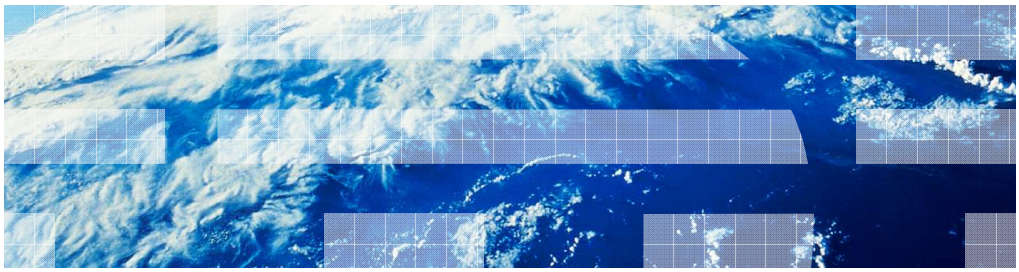
WebSphere Business Process Management Suite V7.0
WebSphere Integration Developer V7.0

Data mapping support



This presentation provides mapping enhancements in WebSphere Integration Developer version 7.

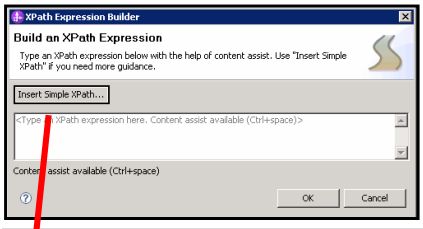
Table of contents

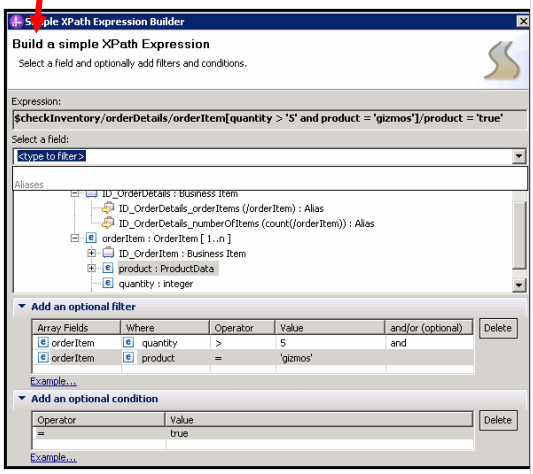
This presentation covers details of many mapping enhancements that were introduced in version 7. It starts with enhancements in XPath editor and with XPath aliases. Then you get to look at additional conditional support with new expanded built in and lookup functions. You also learn how to use data map catalog to work with mapping. You will finish with a summary of all the enhancements to help review.

This section provides details on XPath editor and aliases in WebSphere Integration Developer version 7.

XPath editor

- XPath 1.0 Expression builder improved
- Simple Xpath Expression Builder
  - Easier access from BPEL, MFC, and XML Map
  - Type filter to find elements or business vocabulary aliases
  - Add optional filters or conditions
  - Immediate validation
  - Content assist available in text area

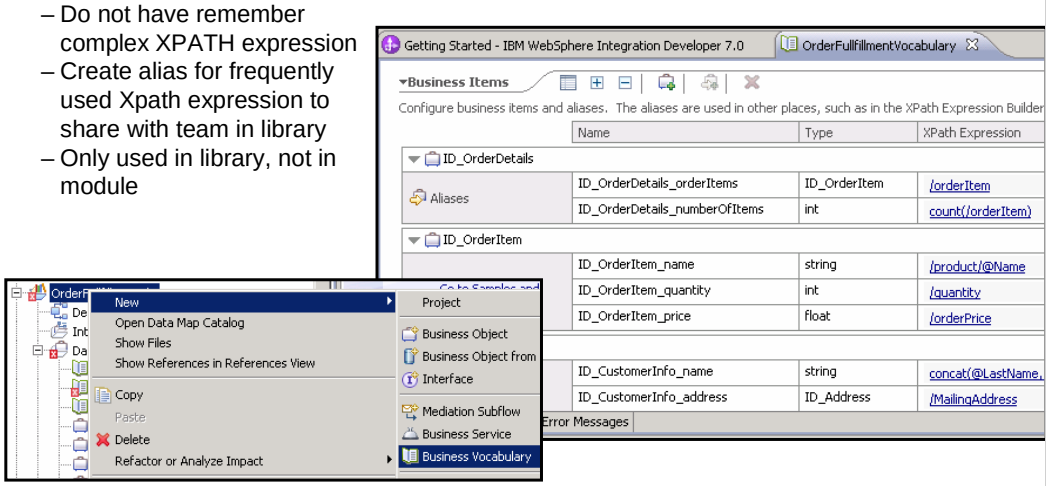




The XPath 1.0 expression builder is improved in version 7. Content assist is available in text area and can call the simple Xpath expression builder. This means more complex Xpath expressions can be created easier and you have easier access from BPEL Editor, Mediation Flow Editor, and XML map editor. In the simple Xpath expression builder, you first select a field with the ability to drill down or filter to find elements or business vocabulary aliases. Business vocabulary aliases are new for version 7 and give you the ability to assign aliases to XPATH expressions so that you or your team does not have to remember complex XPATH expressions. Aliases are the only things you or your team needs to remember. Then in the expression builder, you add optional filters or conditions on those elements, and you will receive immediate validation on the expression.

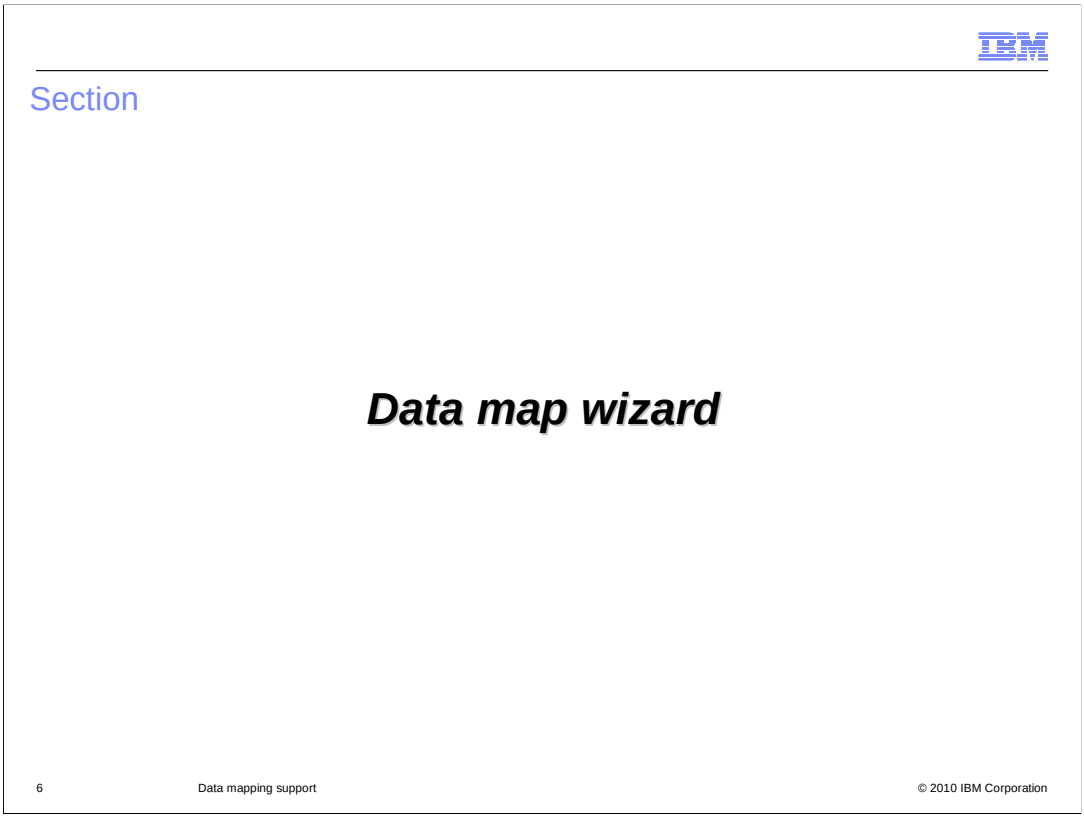
XPath aliases

- Assign aliases to XPath expressions
  - Do not have remember complex XPATH expression
  - Create alias for frequently used Xpath expression to share with team in library
  - Only used in library, not in module



5 Data mapping support © 2010 IBM Corporation

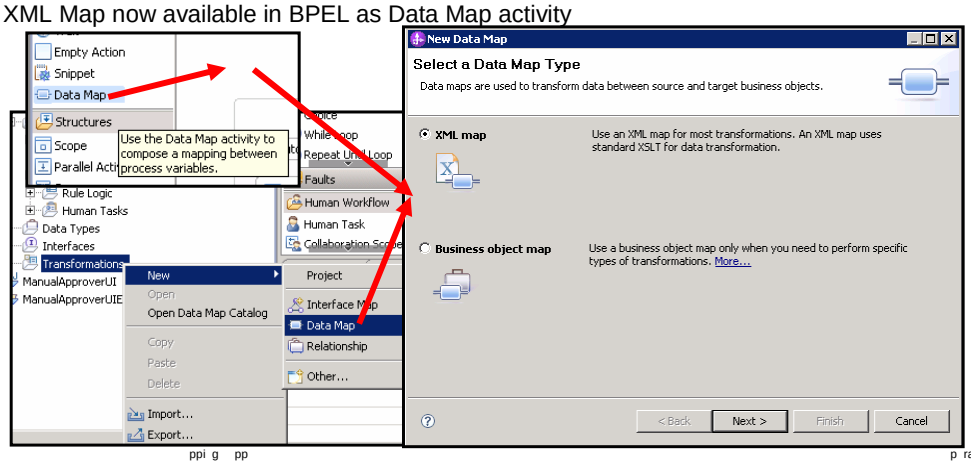
Xpath aliases are for frequently used Xpath expressions to share with team in library. Xpath aliases are not available in modules, but in libraries only. Those aliases are provided by business vocabulary. Business vocabulary is a BPMN document that contains metadata that defines your processes. For your purposes, business vocabulary is the collection of business items where aliases are elements of the business item. Create a new business vocabulary by selecting File, then New then Business Vocabulary. Then create a new business item with an alias inside the business item. Once the alias is created, you can now use it in the Xpath expression editor.



This section provides details about data map wizard in version 7.

Data map wizard

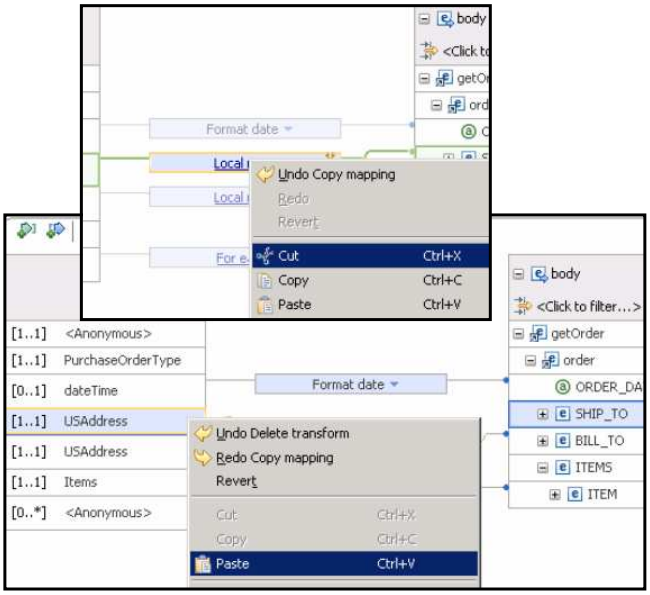
- New Data Map Wizard converges the business object map and XML map wizards
  - Single wizard used to create either business object and XML Maps
  - Guides you to making an informed choice
- XML Map now available in BPEL as Data Map activity



The new data map wizard converges the business object map and XML map wizards into one wizard to help you decide which type of map you want to use. Text and additional help guide you to making an informed choice. In addition, the XML Map is now available in BPEL as Data Map activity. Previously, the business object map was the only map available to be used in BPEL.

Cut, copy, and paste transforms

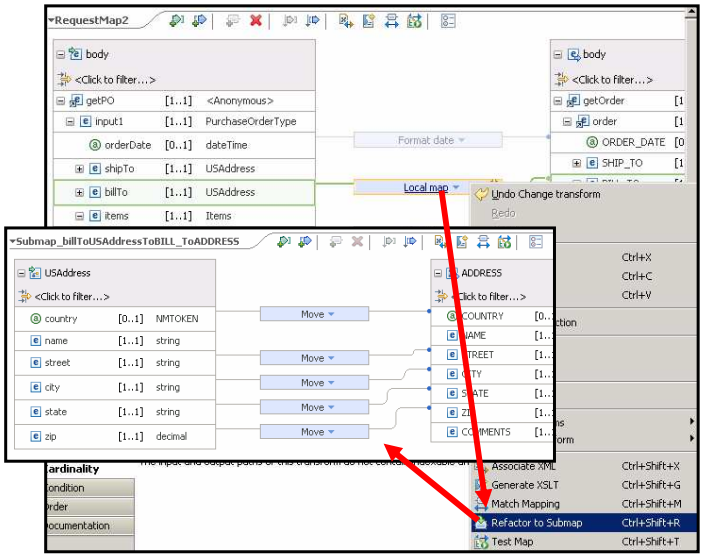
- Ability to cut, copy, and paste transforms in XML Mapping Editor
- Speed development when reusing maps
  - Scenario – copy and paste previously built schema into new map or submap
- Validation
  - Paste action not available if target schema does not match source that was copied



In version 7, XML Mapping Editor has the ability to cut, copy, and paste transforms. This option speeds development when reusing maps. For example, when building new maps from old maps, you can copy schema from the old map to the new map or submap. There is validation on the use of the paste function in that if the target schema does not match the source schema you copied, you are not allowed to paste.

Refactor to submap

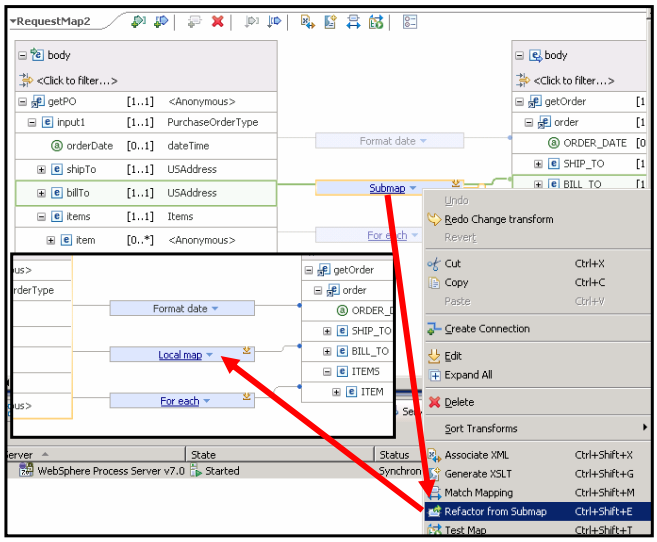
- Ability to quickly change a map into a submap
- Large maps can be hard to manage and not reusable
- Submaps splits up maps into reusable pieces and easier to manage



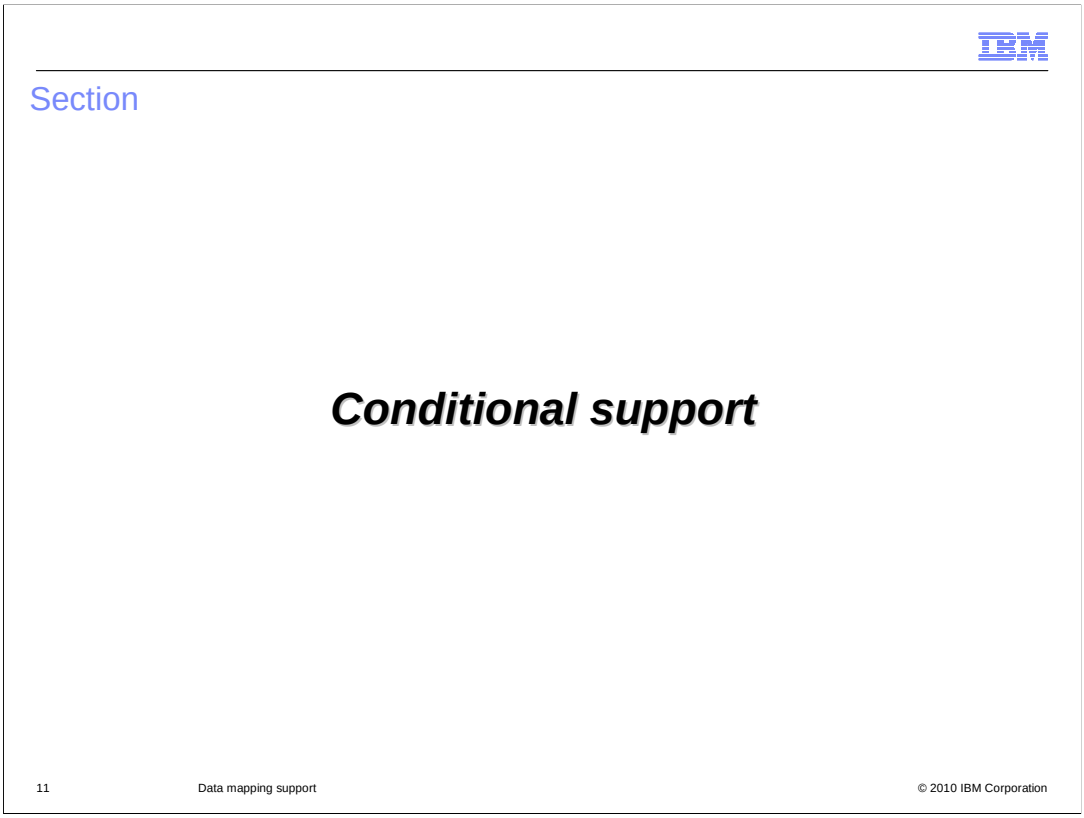
Another enhancement in data mapping is the ability to quickly change a map transform into a submap. The reason for refactoring into a submap is to split up maps into reusable pieces. Large maps can be hard to manage and not reusable. Using submaps makes map content reusable and are easier to manage. The refactor from submap action allows you to create a local map from submap along with its containing mappings.

Refactor from submap

- Ability to quickly change a submap into a local map
- Able to include submaps or refactor out of a submap
- Local map created with containing mappings



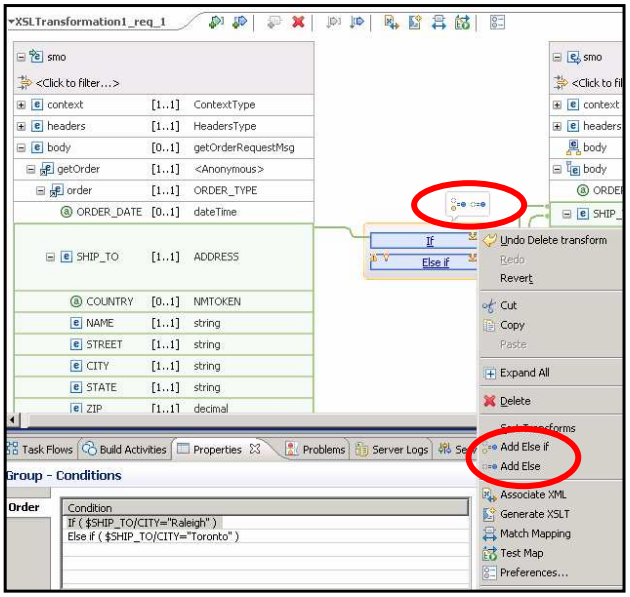
Continued from the previous slide, this new enhancement also provides ability to change from a submap into a local map. This diagram shows how you can quickly change a submap into a local map by selecting the refactor from submap feature.



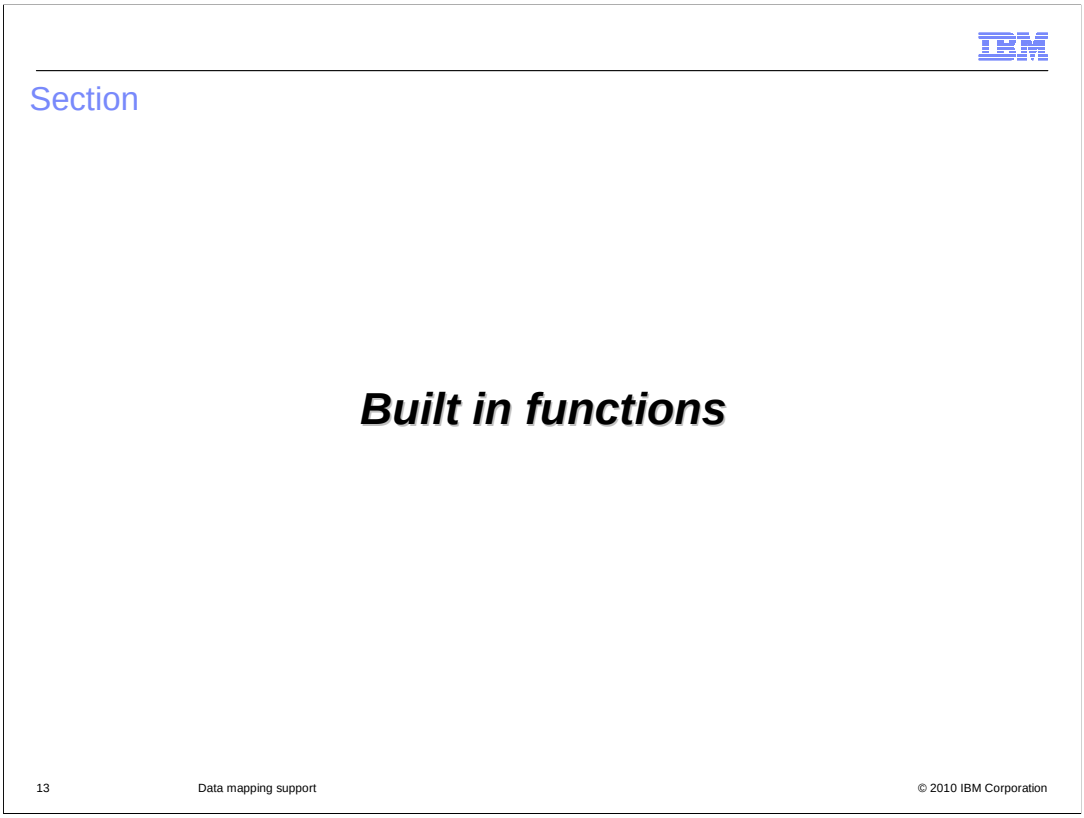
This section provides some new conditional support for data mapping in version 7.

Conditional support

- Provide ability to make logic decisions in map
- WebSphere Integration Developer 6.2.0.1 only supported “if” condition
  - Now support if, else if, and else conditions
- XPath 1.0 support for expressions
- Logic evaluated in order
- Fires transform for true logic



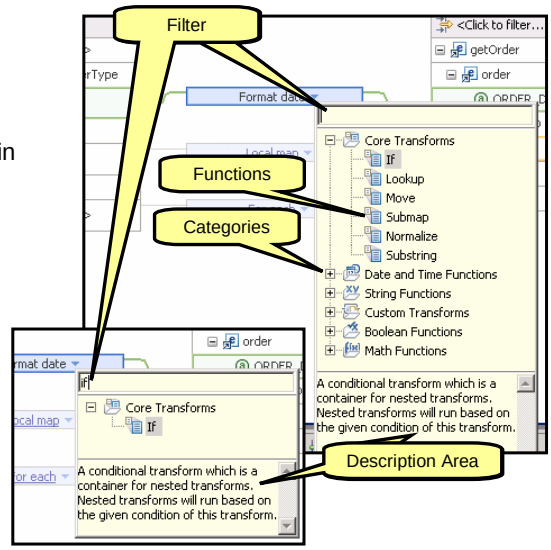
The conditional support in maps have also been expanded. Starting with WebSphere Integration Developer version 6.2.0.1, it supported the if condition. And with version 7, the if, else if, and else conditional logic are supported. They provide ability to make logic decisions in maps. The logic is evaluated in order and fires the transform for the true case.



This section provides details about expanded built in functions for data mapping in version 7.

Built in functions (1 of 4)

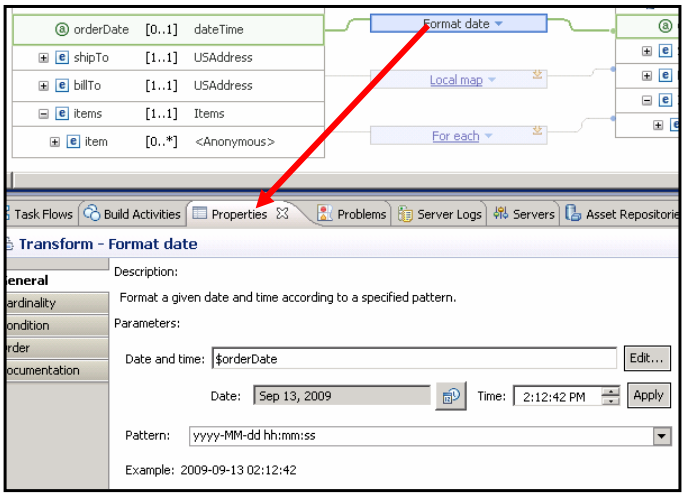
- Expanded built-in function support
  - Allows user to transform data from source to target using function
- Can filter for certain function or drill down in categories
- Text description of function provided right up front
- Easier to call Custom Java Extension Functions and Custom XSL functions



In addition to the conditional support, the built-in function support has been greatly expanded in version 7. This allows you to transform data from source to target using many different functions. You can filter for a certain function or drill down in categories to find the certain function. Also, a text description of the function details is provided in the dialog to help understand how to use the function. Custom transforms enables easier calls to custom Java extension functions and custom XSL functions.

Built in functions (2 of 4)

- Provide first class property pages for date formatting, concat, substring
- Other functions
  - generic properties page for passing in the function parameters
- Resources
  - F1 to xpath 1.0 specs and eXLST functions in Apache



The properties view also helps provide editing details and is first class for date formatting, concat, and substring. For references, you can hit F1 on your keyboard to view XPath 1.0 specifications and eXLST functions in Apache.

Built in functions (3 of 4)

String functions

concat	contains	string	starts-with	substring-before
string-length	substring	translate	tokenize	normalize-space
align	padding	split	substring-after	

Math functions

ceiling	floor	round	number	sum
abs	acos	asin	atan	atan2
log	lowest	max	min	power
tan	random	sin	sqrt	highest
constant	cos	exp		

Here are lists of string and math built in functions that have been expanded in version 7.

Built in functions (4 of 4)

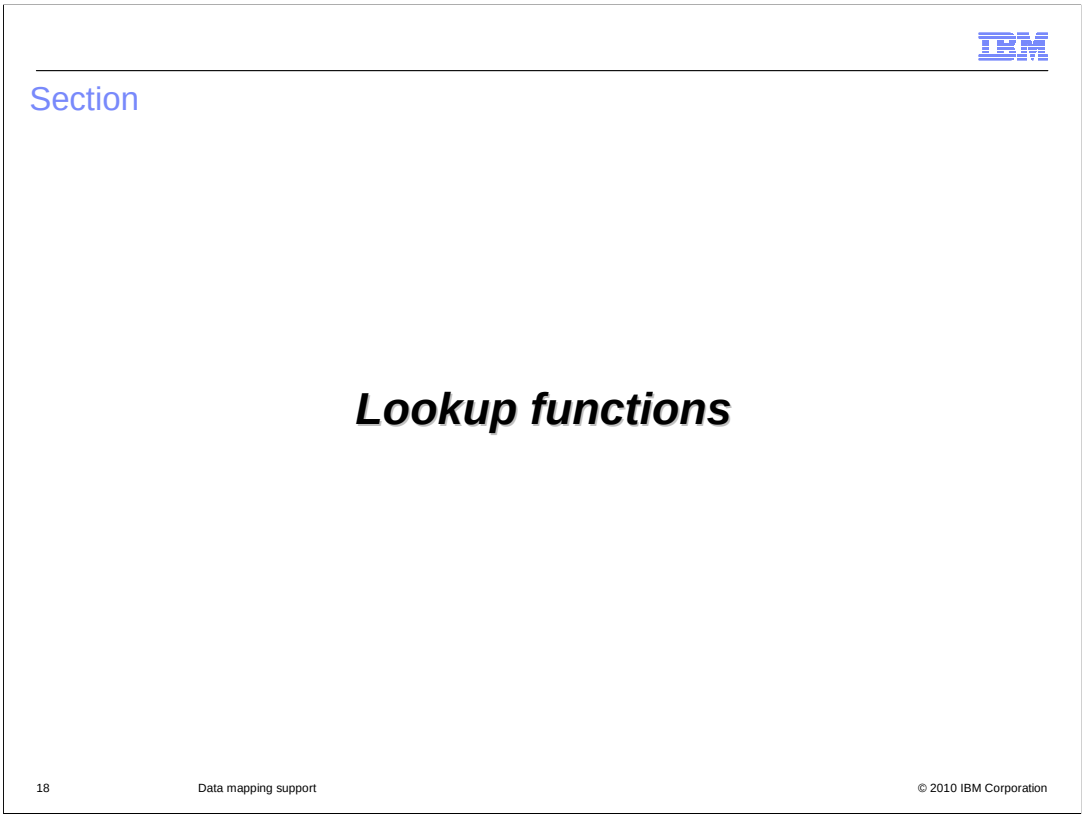
Boolean functions

boolean	false	true	lang	not

Date and time functions

date	dateTime	dayInWeek	dayInMonth	
dayInYear	dayName	hourInDay	formatDate	dayAbbreviation
leapYear	minuteInHour	monthName	monthInYear	dayOfWeekInMonth
time	secondInMinute	weekInYear	year	monthAbbreviation

Here are some additional lists of Boolean and date / time built in functions in version 7



This section provides details about lookup functions in data mapping in version 7.

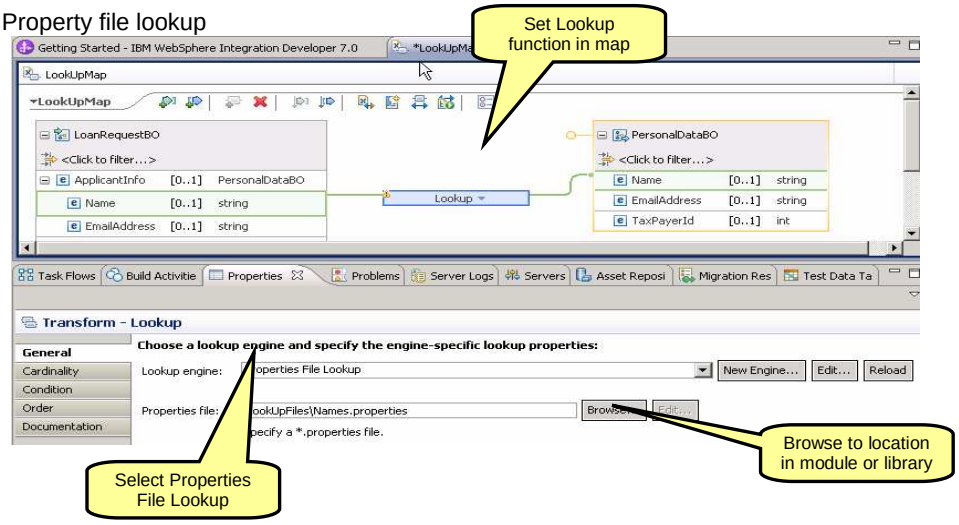
Lookup functions (1 of 4)

- Facility to lookup a key from a file and return the specific value
- Three new lookup functions
  - Comma-separated values (CSV)
  - Properties file
  - Custom function engine
    - Easy to create your own Java file
- Previously, only relationship lookup
- Provides access to key value lookup
  - For example: hash table look up key retrieval

Three new built-in mapping facilities are provided in version 7 to lookup a key and return the specific value. Before version 7, there was only the relationship lookup function. The purpose of a lookup function is to lookup a key and return the specific value. This is done through a Comma-Separated Values (CSV) file, a properties file, or a custom function engine. For the custom function engine it is easy to create your own Java file. The most used example for the look up functions is the use of a hash table to look up a key and retrieve another value.

Lookup functions (2 of 4)

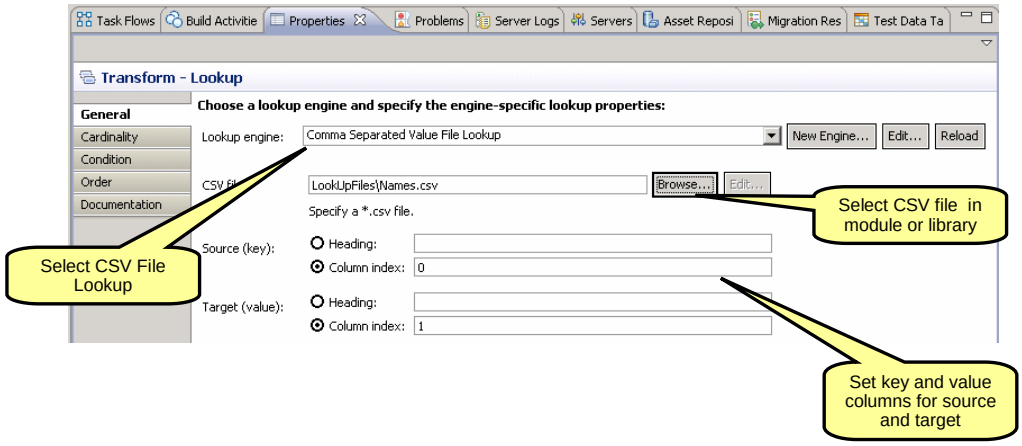
- Property file lookup



This diagram shows how you can look up a key value using built in property file lookup function. First, you set the lookup function in your lookup map. And then in properties view, choose the lookup engine and specify the engine-specific lookup properties. In this scenario, select Properties File lookup and then browse to the location of the properties file in your module or library.

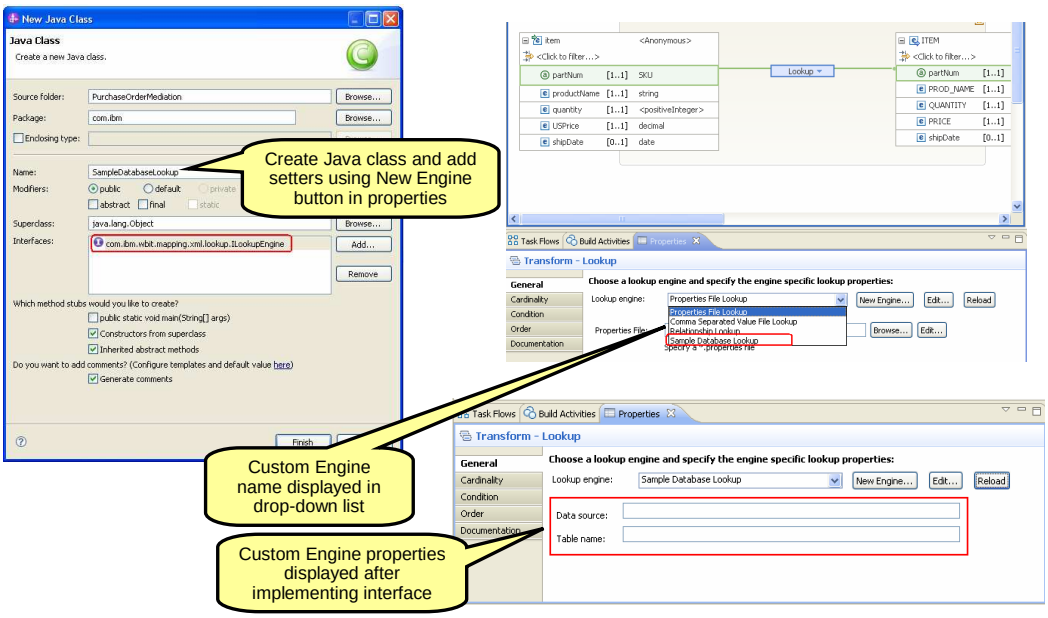
Lookup functions (3 of 4)

- CSV file lookup



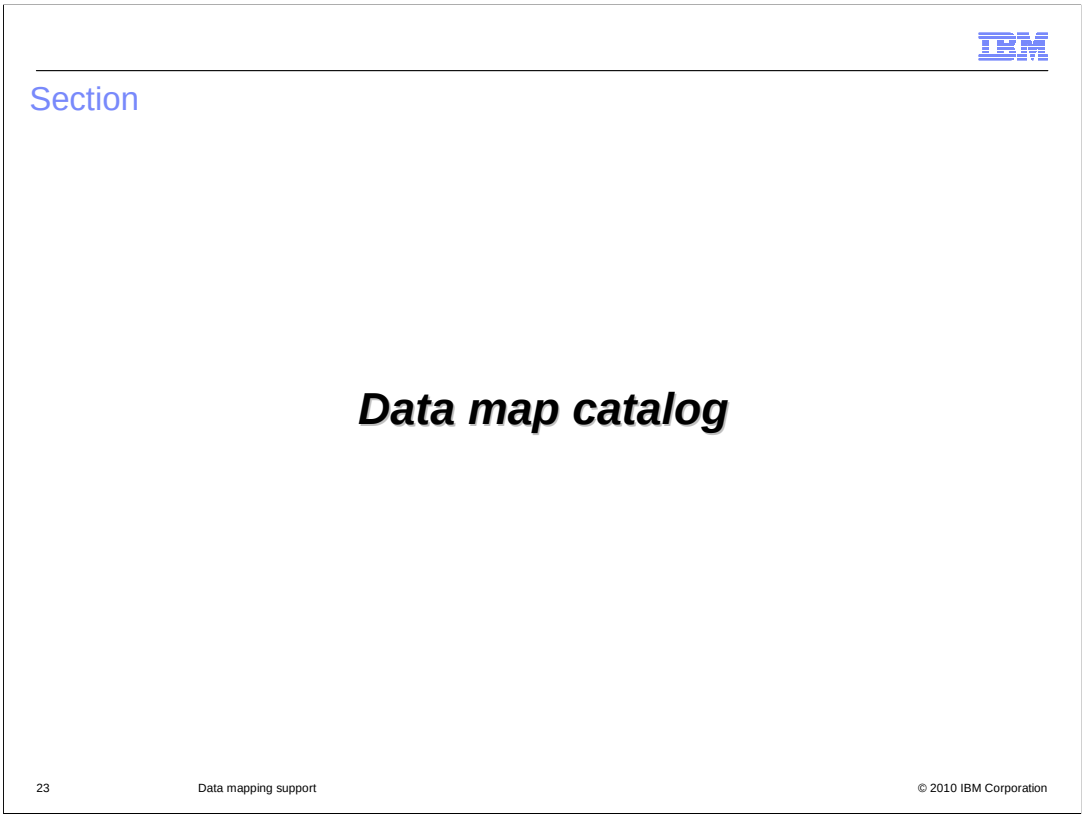
The second built in lookup function is CSV file lookup. From properties view, select CSV file lookup engine and its location in your module or library. Then set key and value columns for source and target.

Lookup functions (4 of 4)



22 Data mapping support

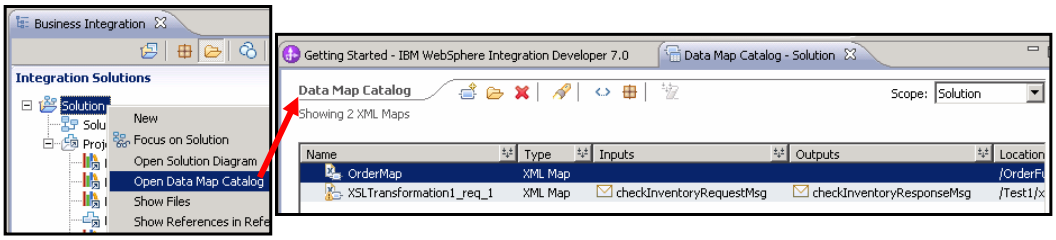
Another new built in look up function is custom function engine, with which you can easily create your own Java file. In properties view, you can create a new Java class and setters using the New Engine feature. Once you have completed creating the new Java class, you can see that the custom java name is displayed in the drop down list of the lookup engine**.** The custom engine properties are displayed after implementing the interface.



This section provides an overview of new data map in WebSphere Integration Developer version 7.

Data map catalog

- Easier to view and work with all maps in solution, module, or library
  - Sort by column headers
    - Can use wildcards
  - Can text search entire page
    - Can use wildcards
  - Filter by content tags and namespace
  - Show in references view to see integration points
  - Can create a new data map from catalog view



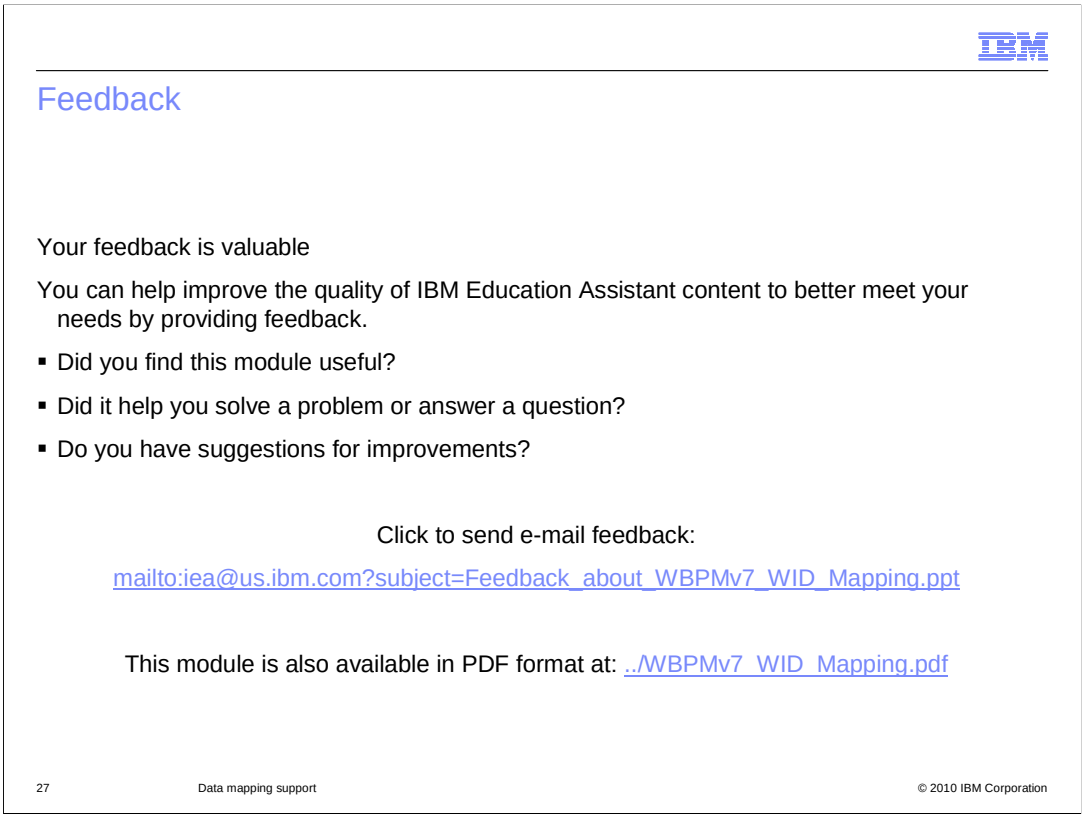
Previously, there was not an efficient way to see or work with all maps in solution, module, or library. WebSphere Integration Developer introduces a data map catalog in version 7 to make this possible. To help view, you can filter rows and sort by column headers like name, type of maps, inputs, and outputs. There is also a text search for all the results. Wildcards are accepted when filtering to find a map or certain maps. Toggle buttons in the data map catalog allow you to show and hide content tags and namespaces. In addition, you can see other objects that use the map in the references view. Last, you can create a new data map from the data map catalog.

This section provides summary of all enhancements for data map support in WebSphere Integration Developer version 7.

Summary

- **Data map support**
  - Combined data map wizard provides one entry point to create business objects and XML Maps and text to helps users choose which one
  - Refactoring of maps with Cut, Copy, Paste, and refactor into and from submaps
    - Help split up maps into reusable pieces, easier to manage
  - Conditional support – if, else-if, else support provides ordered condition capability in maps
  - Built-In Functions are provided to help speed mapping development
  - **Three** new lookup functions provide access to lookup a key from a file and return the specific value
  - Mapping catalog added to work easier with all maps in solution, module, or library
  - Content tagging added to signify special meaning for schemas
  - XPath 1.0 Expression builder improved providing access to Xpath aliases
  - Casting derived types

In summary, several mapping enhancements are introduced in version 7. The combined data map wizard provides one entry point to create business object and XML Maps along with text to help users choose which map to use. The Xpath 1.0 expression builder is enhanced for easier creation of Xpath expressions. You can now create aliases in business vocabulary for frequently used Xpath expression to share with team in library. Refactoring of maps is enhanced with cut, copy, and paste of transforms**,** and then refactoring into and from submaps to help split up maps into reusable pieces and make them easier to manage. Conditional support is improved with if, else-if, else, providing ordered condition capability in maps. Built-In Functions are provided to help speed mapping development. Three lookup functions are added to retrieve a key from a properties, CSV, or custom Java file. Data map catalog has been added for an easier view of all maps in solution, module, or library. You can now cast a selected type to a derived type in an XML map. This means you can map a source type to a different target type if the type is derived (abstract).



You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, disclaimer, and copyright information