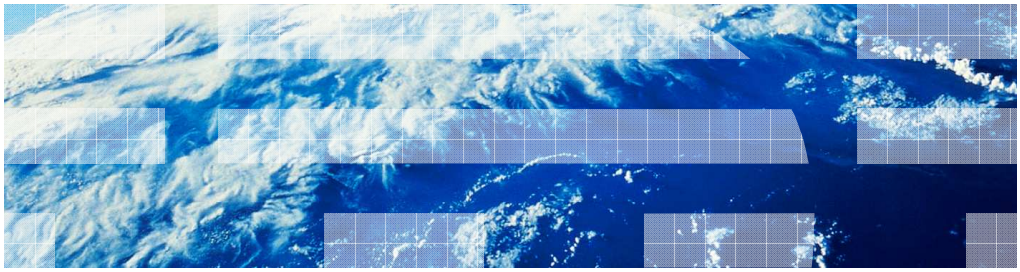
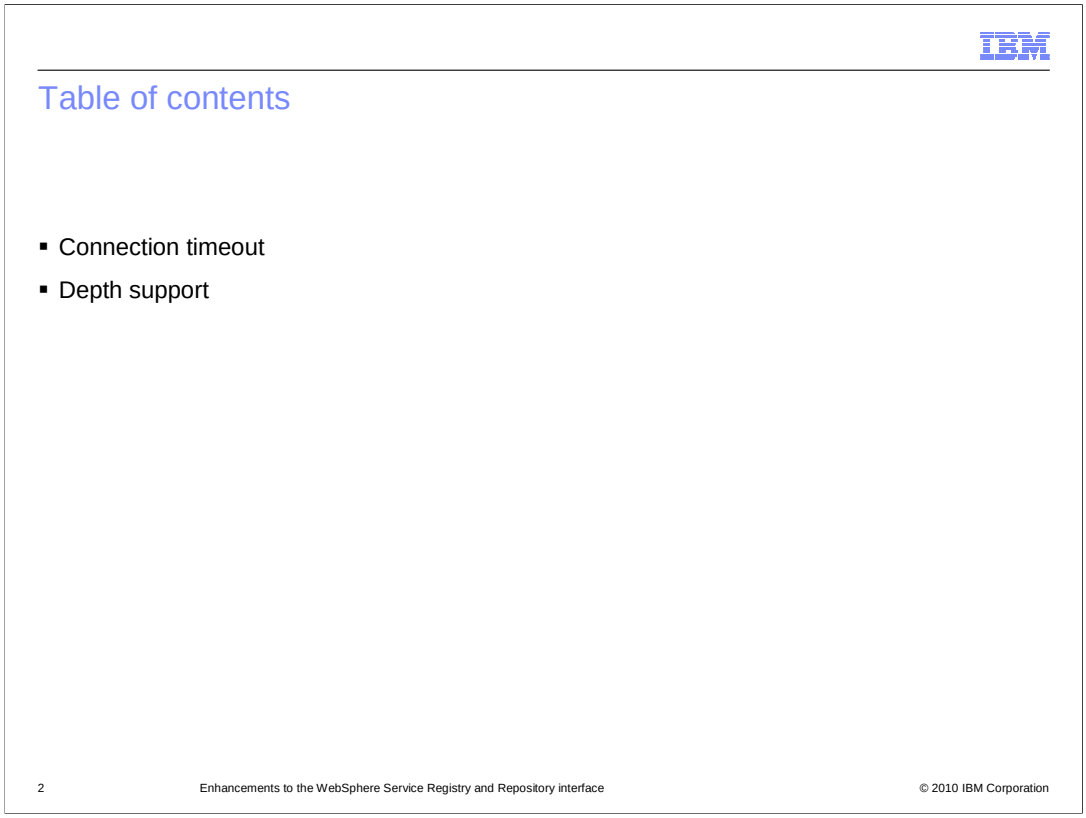
WebSphere Message Broker Version 7

Enhancements to the WebSphere Service Registry and Repository interface

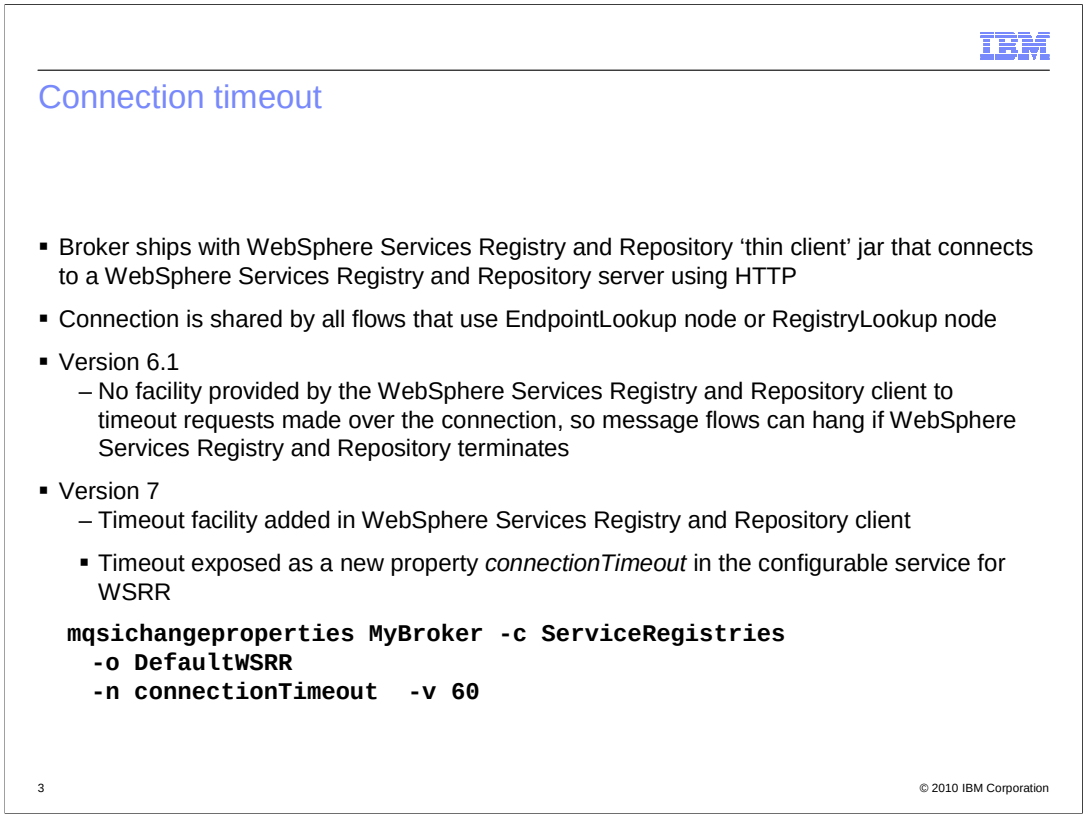


This session discusses the enhancements that have been made to the interface to WebSphere® Service Registry and Repository in Message Broker version 7.



Message Broker version 6.1 introduced two new nodes, the Endpoint Lookup node and the Registry Lookup node. These nodes allow a message flow to connect directly to WebSphere Service Registry and Repository to obtain details about services that the message flow is going to invoke.

These nodes have been enhanced in Message Broker version 7 in two areas. This session will describe the connection timeout property, and the new facility to retrieve additional information from the service repository.



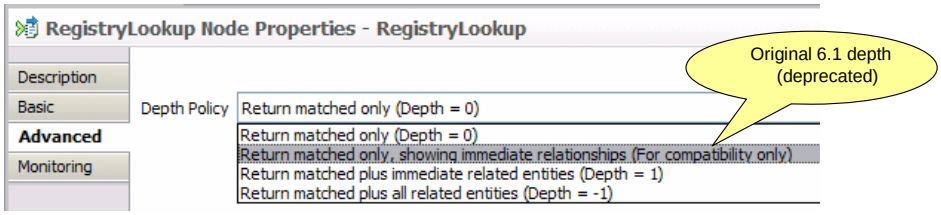
When using WebSphere Service Registry and Repository with Message Broker version 6.1, if the registry becomes unavailable, this is not detected by the broker. Any flows that contain EndpointLookup or RegistryLookup nodes and that cause a request to be made to the registry, the registry can "hang" and never time out.

In Message Broker version 6.1, the registry client API does not expose a timeout value for the broker to set. A new release of the registry client jar, version 6.2.0.2, is now shipped with Message Broker version 7. This allows a timeout to be set. The default is 180 seconds. The value can be changed using a new property connectionTimeout on the broker-wide ServiceRegistries configurable service for the service registry.

The value is specified in seconds. This should not be confused with the property "timeout", which controls cache expiry. A BIP3678 exception is thrown if the connection timeout expires.

Depth support

- WebSphere Services Registry and Repository offers three depth options when queried for objects:
  - **0** returns just the matched object
  - **1** returns the matched object and its immediate related objects
  - **-1** returns the matched object and all related objects
- 6.1 implemented a hybrid of 0 and 1
  - returned matched object, plus information about its relationships, but without the related objects themselves
- Customers have requested that all WebSphere Services Registry and Repository depth options be provided
- RegistryLookup node now offers a new *Depth policy* property



4 Enhancements to the WebSphere Service Registry and Repository interface

The 6.1 implementation of the RegistryLookup node did not allow the 'depth' of the service registry query to be specified. The service registry provides three depth options, although these are not used in Message Broker version 6.1. In version 6.1, the registry lookup node only returns matched objects together with details of their related objects.

In Message Broker version 7, the Registry Lookup node has a new *Depth policy* property. The default value is Match Only (0).

To maintain compatibility with earlier versions, the property retains the 6.1 'hybrid' depth as an option. It is marked as 'compatibility only' to discourage its use, and gives a 'deprecated' warning in the toolkit. All new applications should use one of the standard options.

Depth support – Local environment

- Input to node: The node *Depth policy* property can be overridden at runtime using the local environment (like other lookup properties):

  `LocalEnvironment.ServiceRegistryLookupProperties.DepthPolicy`

- Output from node: When using the WebSphere Services Registry and Repository standard depths, the local environment ServiceRegistry folder of Entity elements has these improvements:
  - The entire tree is owned by XMLNSC domain
  - No unnecessary namespacing
  - No added whitespace elements
  - Binary content handled naturally as BLOBs

In version 7, the structure of the service registry folder created by the registry lookup node in the local environment has been changed.

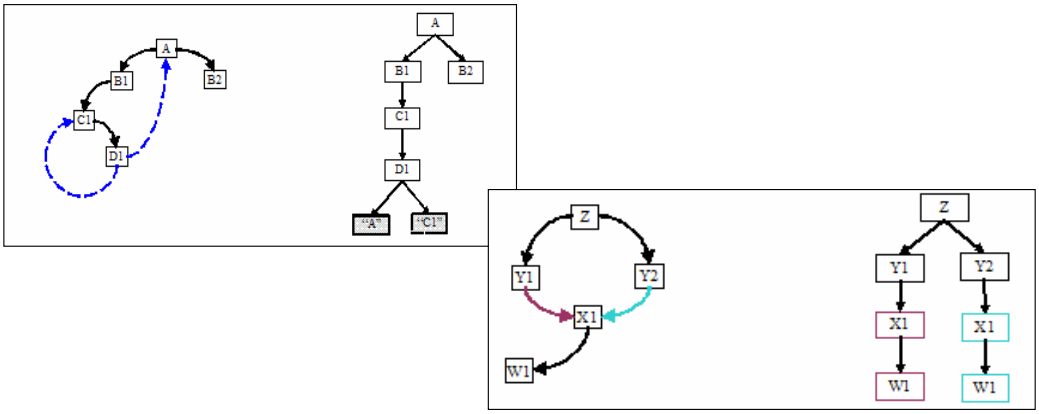
In earlier version, the objects retrieved from the service registry were serialized to XML, then re-parsed using the XMLNS domain. This resulted in namespaces and white space in the tree. Also, binary content was rendered as base64 strings, which required a transformation to get the original data.

In version 7, when the standard depths are used in the registry lookup node, a different service registry folder is created, owned by XMLNSC domain. This is more compact, omits unnecessary information and correctly represents binary content as BLOBs.

The original folder structure is retained for the 6.1 ‘hybrid’ depth, for compatibility.

Depth support – Match All (-1)

- WebSphere Service Registry returns a graph of entities which poses some challenges when creating a local environment tree
  - Circular references are replaced by special ‘reference’ elements
  - Multiply referenced entities are cloned along with all children



When -1 is specified for *Depth policy*, service registry returns the matched entities plus all their related entities, to whatever depth is required. The service registry entities are held as a graph, and therefore it is possible that relationships exist between entities that can not be expressed in a broker tree, which is a strict tree structure.

When a circular reference is encountered, a special EntityRef element is used instead of the usual Entity element, in order to break the circularity.

When a multiply referenced entity is encountered, it and all its children are cloned.

Depth support - predefined queries

- You can specify a list of queries to be performed when the broker starts, to pre-populate the cache
- In version 6.1 these queries always used depth= -1
- Predefined queries can now be suffixed with a depth specifier {depth=0|1|-1} so that a depth-specific result can be cached
- The default is depth -1 for compatibility
- Predefined queries are specified on the configurable service for Service Registry, using command or MBExplorer

```
mqsichangeproperties MyBroker -c ServiceRegistries
–o DefaultWSRR –n predefinedCacheQueries
–v "//*[@name=&quot;ConceptA1&quot;]{depth=1}"
```


As well as the Registry Lookup node, you can also specify service registry queries in the broker-wide “Service Registries” configurable service. Queries are specified using the *predefinedCacheQueries* property, and are issued when the broker starts. The results are used to pre-populate the broker’s cache of service registry entities. It is appropriate therefore to allow the depth query to be configured here as well.

The syntax of the property is a semi-colon separated list of service registry XPath query expressions, which does not cater for depth. Rather than provide a new multi-value property which must be kept in step, the syntax of each XPath in the list has been extended. It now allows an optional suffix, where the values for depth are 0, 1 or -1.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WMB7_GeneralFacilities_WebSphere_Service_Registry.ppt

This module is also available in PDF format at:
../WMB7_GeneralFacilities_WebSphere_Service_Registry.pdf

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

Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.