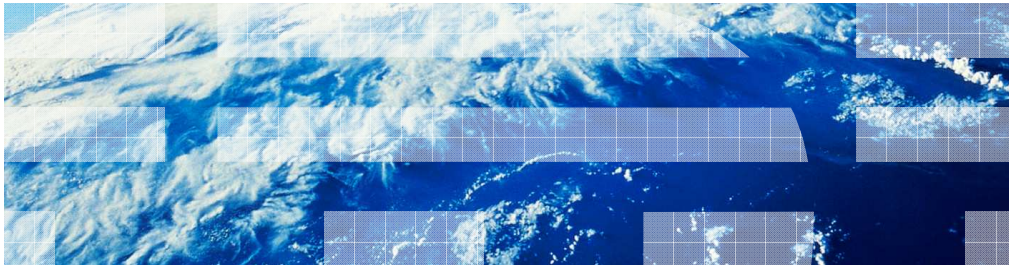


WebSphere Message Broker Version 7

Adapter nodes – Iterative discovery and deployment



This session will describe how to make incremental changes to an adapter definition and message set generated by the adapter connection wizard, without having to completely regenerate the existing message set.

It will also describe the changes that have been made in WebSphere Message Broker version 7 to enable adapter applications to be managed through an application development process.

Iterative development

This section will describe the new techniques for iterative development of message flows using adapter connections.

Background

- Dynamic Methods
 - Message Broker applications needs new EIS service
 - Existing message flows should not be modified and retested when a new service is provided by the EIS
- Dynamic Events
 - A new event type is introduced into the EIS
 - Existing message flows should not be modified and retested when a new event type is provided by the EIS
 - EIS – Enterprise Information System – SAP, Siebel, PeopleSoft

When building message flows which access application systems such as SAP and Siebel, the application system might introduce new methods or events. These new services did not exist when the flow was originally developed. These new services must be incorporated into the existing message flows, which should not need to be retested.

Similarly, a new event type might be introduced into the application system.

When such changes are made, and new Message Broker applications are created to make use of these services, it is important that existing applications are not compromised. It should not be necessary to retest such existing applications.

Dynamic methods

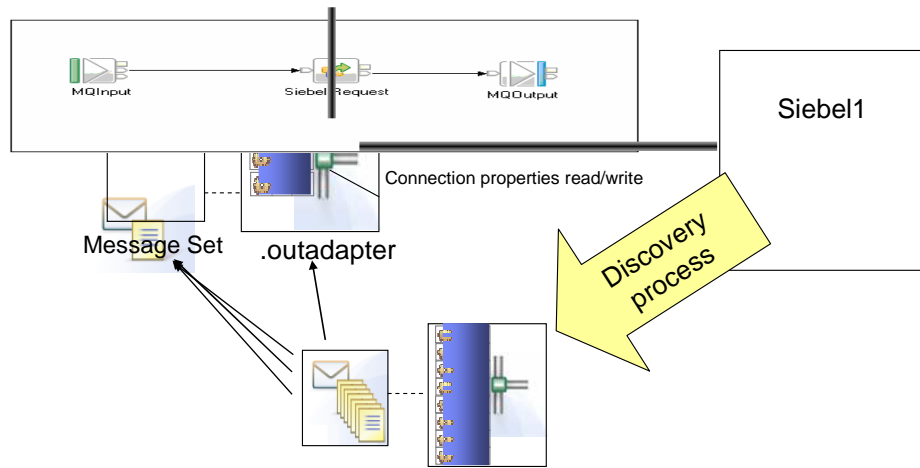
- The main artifacts needed for a Message Broker application are
 - Message flow – including Adapter Request node
 - Message set – including message definitions
 - .outadapter – which contains
 - details of which services are available from that adapter
 - Method names
 - Input Types (corresponds to a message type from the message set)
 - Output Types (corresponds to a message type from the message set)
 - Connection parameters
- In Message Broker version 6.1:
 - Once created, the .outadapter cannot be changed
 - There is no editor for it (other than the connection details)
 - Methods cannot be viewed/changed after running the wizard
 - Method names can be viewed but only using a drop down box on the node
 - Cannot add new methods to an existing .outadapter
 - Need to create a new .outadapter which replaces the existing one

In a Message Broker application using a Request Adapter node, the primary artifacts are the message flow, the message set, and the adapter definition. The adapter definition will contain the method names of the service, and the input and output message types. These message types will correspond to a message type in the message set.

The adapter will also contain the connection parameters for the application system. These allow the adapter to establish the physical connection to the application system.

The “out adapter” therefore contains all the key information about connectivity to the application system. It combines the connection details and the service details into a single definition, which makes changing the service definitions difficult in Message Broker version 6.1.

Incremental development in Message Broker version 6.1



5

Adapter nodes – Iterative discovery and deployment

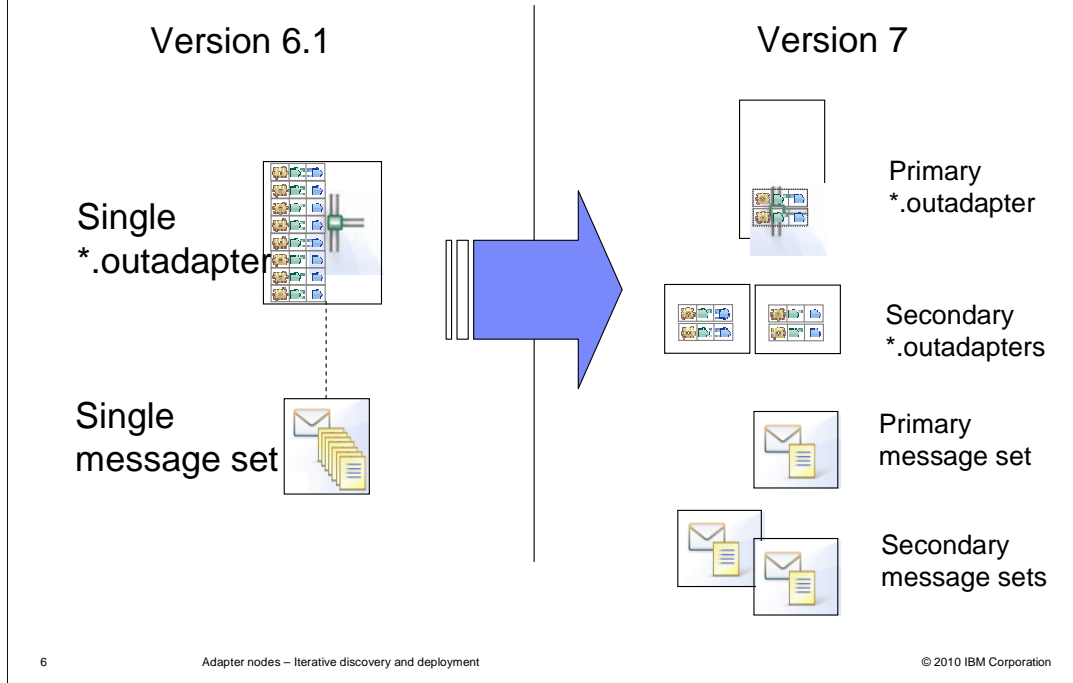
© 2010 IBM Corporation

This example shows a message flow, with a Request node that connects to a Siebel application. Each new Siebel service that needs to be used by a message flow is discovered by the adapter connection wizard. This wizard regenerates the “out adapter”, and adds the required service to the message definitions in the message set.

When you perform this incremental service discovery, you must make sure that you select all the previous services that are used by the existing message set and message flow. If there are a large number of these it can be difficult to replicate the earlier service definitions that have been created already.

Also, since the single message set is being re-created, when the resulting barfile is deployed, it completely overwrites all previous application components. It is therefore necessary to retest all existing applications, to ensure that no regressions have taken place. Finally, if there are a large number of services, the message set can become very large, leading to monolithic applications, which are difficult to maintain.

Incremental development in Message Broker version 7

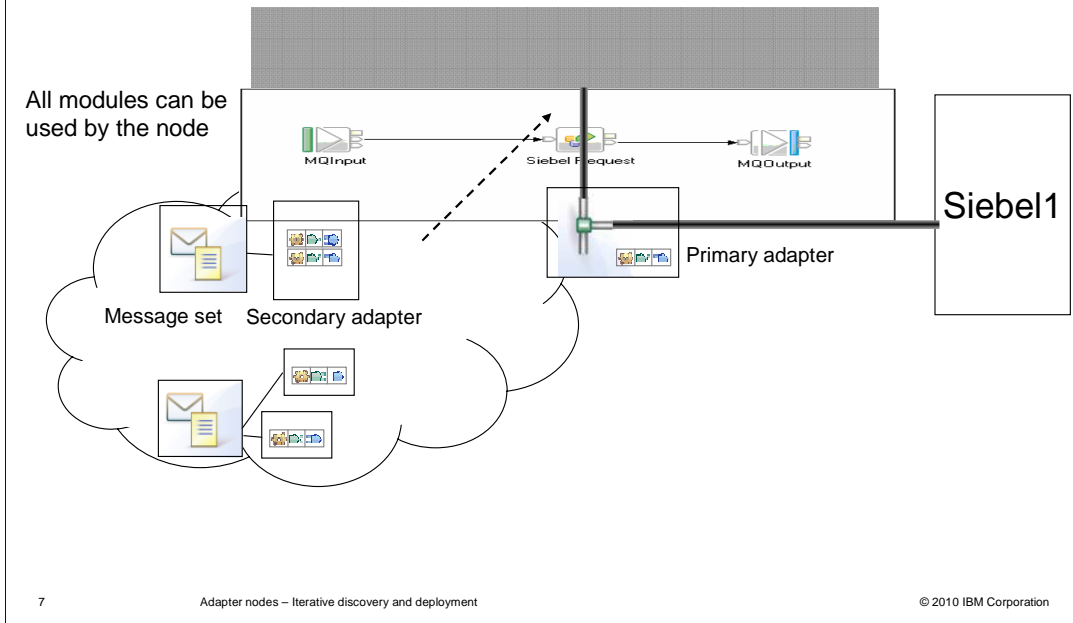


In Message Broker version 7, the adapter artifacts have been redesigned, and split into the two main components.

First, the primary adapter component contains the connection details of the target application system. This will include items such as TCP/IP address, host name, the specific system information, and the security credentials of the system. These will typically be over-ridden in configurable services.

Second, the adapter definitions which define the specific application services that are available in the target application system. Also, the requirement for a single message set is removed, and a separate message set can be defined for each new service. These will correspond to the new generated adapter definitions.

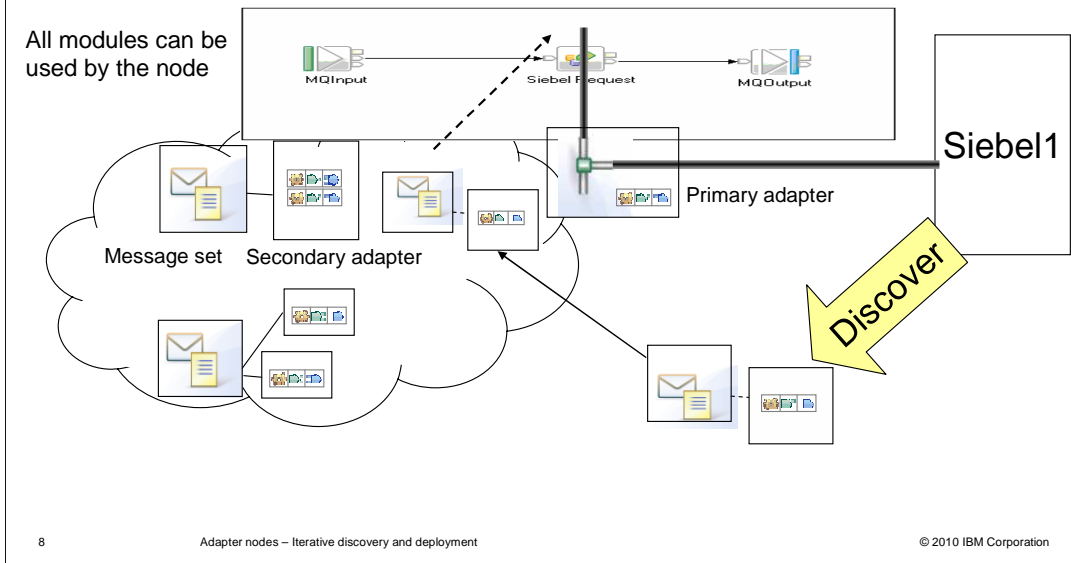
Adapter structure in Message Broker version 7



In Message Broker version 7, each service available in the Siebel system will create its own secondary adapter and message set. The secondary adapter is where the Siebel service definitions are stored, and each will correspond to a separate message definition, optionally located in a separate message set.

The primary adapter is still used to store the connection details to Siebel.

Discover a new Siebel service



8

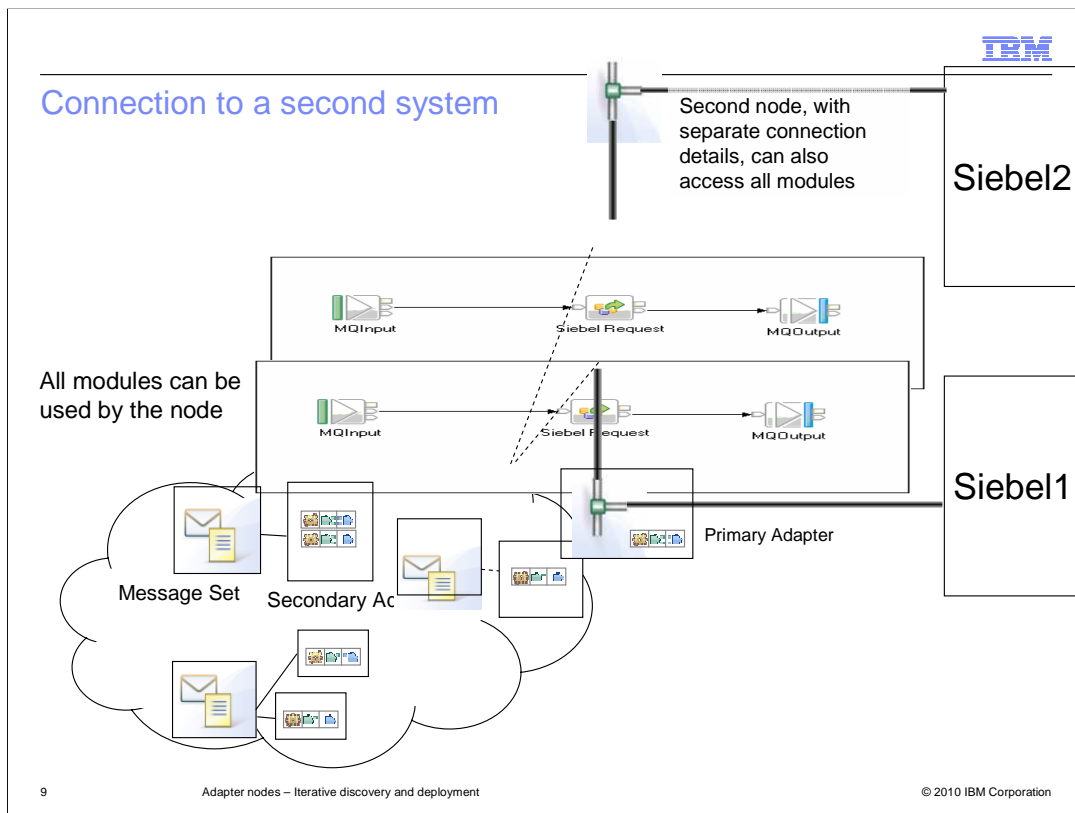
Adapter nodes – Iterative discovery and deployment

© 2010 IBM Corporation

When the adapter connection wizard is invoked to discover new service in Siebel, this will create a new secondary adapter to contain the new service definition, and a new message set to contain the message definition. However, the primary adapter will remain unchanged, since the Siebel connection details have not changed.

When the new service is discovered, it is no longer necessary to remember which services have already been defined.

The new adapters and message definitions that are created do not overwrite the existing definitions. Hence, it is not necessary to retest existing applications. The message flow that performs the integration logic can use any of the adapter definitions and message definitions that are available.



If a second Siebel system is introduced, it will have different connection details to the first, so a new primary adapter connection must be created. However, if the two Siebel systems have the same service definitions, a second message flow which accesses the second Siebel system can use the existing secondary adapter and message sets that have already been discovered. It is not necessary to discover the services again from the second Siebel system.

***Staged deployment -
Deploying applications through development,
test and production environments***

This section describes the new capability that has been provided in Message Broker version 7 to allow adapter-based applications to be promoted through the various stages of development. This promotion can be done without making changes to the message flow, or to the adapter definition contained within the flow.

This function was provided for the SAP adapter in a fixpack in Message Broker version 6.1. This has now been extended to Siebel and People/Soft in version 7.

Deployment process

- Message flow barfile built from message flow/adapter/message set
- Adapter is configured to connect to the Development application system
 - Message Broker administrator deploys that bar file to the broker
 - Message flow is tested against the Development application system
- Now move to production system
 - Adapter must be re-configured to connect to the production system
 - In Toolkit, reconfigure the adapter, then re-build the bar file
- Problems
 - The message flow developer must rebuild barfile
 - (or administrator needs access to and skills required to use the message broker Toolkit)
 - The message flow source must be changed, compromising the testing rigor
 - Security credential changes can mean the adapter must be rebuilt

When you build a message flow with an adapter connection using Message Broker version 6.1, you must make changes to the adapter definition as the application is moved through the testing cycle. This means that the barfile must also be rebuilt for each stage of the development cycle.

To do this, you build the barfile using the message flow, the message set, and the adapter for connection to the application system.

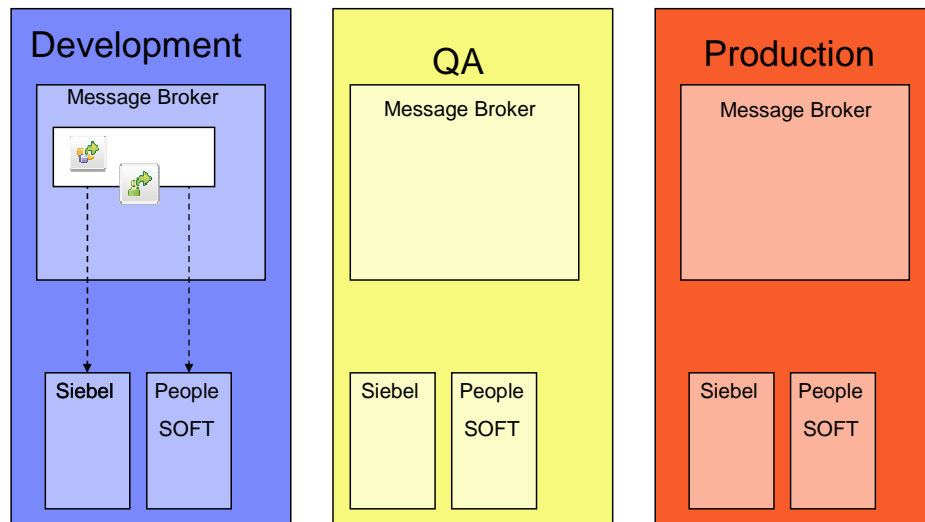
Initially, the adapter is configured to connect to the development application system.

Once all tests are complete, testing will move to the QA system. The adapter must be re-configured to connect to the QA system, and the barfile must be rebuilt. This requires changes to the adapter using the Message Broker Toolkit, followed by a rebuild of the barfile.

The same process applies when moving the application into the production systems.

This means that the message flow developer is required to make changes throughout the testing process. This might compromise testing integrity, since it cannot be guaranteed that the application logic itself has not been changed in the process.

Application promotion – Message Broker version 6.1 (1)



12

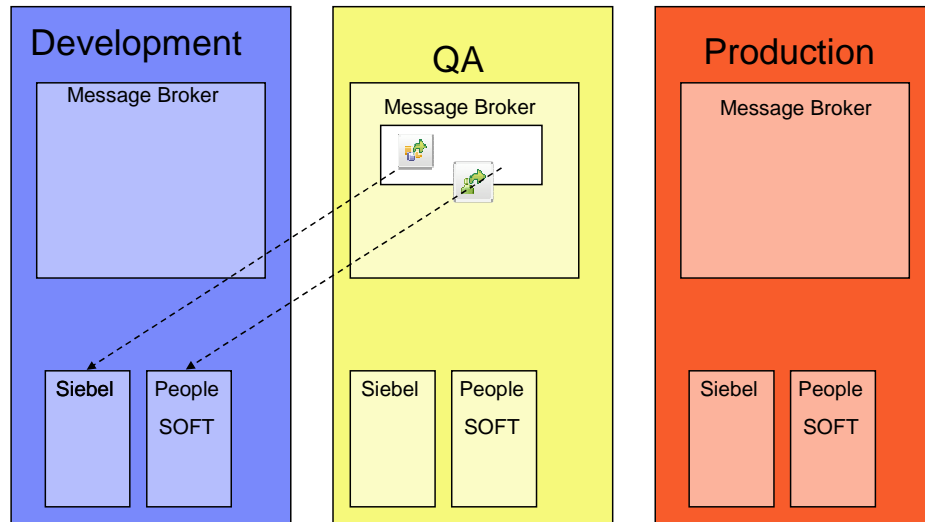
Adapter nodes – Iterative discovery and deployment

© 2010 IBM Corporation

This slide illustrates the deployment method for message flows and adapters, using Message Broker version 6.1.

The message flow and adapter are developed using the connection and security credentials of the development application system. The barfile is built and deployed to the development broker, and testing takes place using these systems.

Application promotion – Message Broker version 6.1 (2)



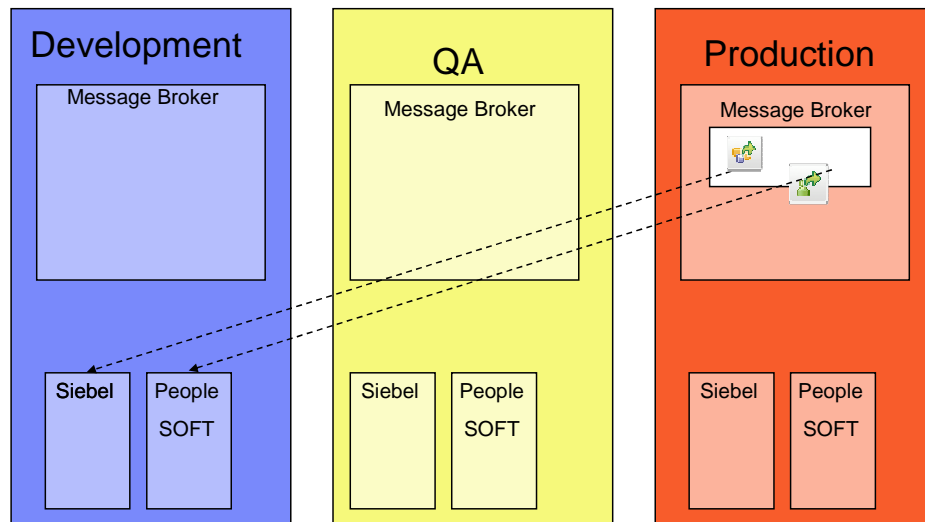
13

Adapter nodes – Iterative discovery and deployment

© 2010 IBM Corporation

The barfile is now deployed to the QA Message Broker instance. The barfile still contains the connection details and security credentials of the development application system.

Application promotion – Message Broker version 6.1 (3)



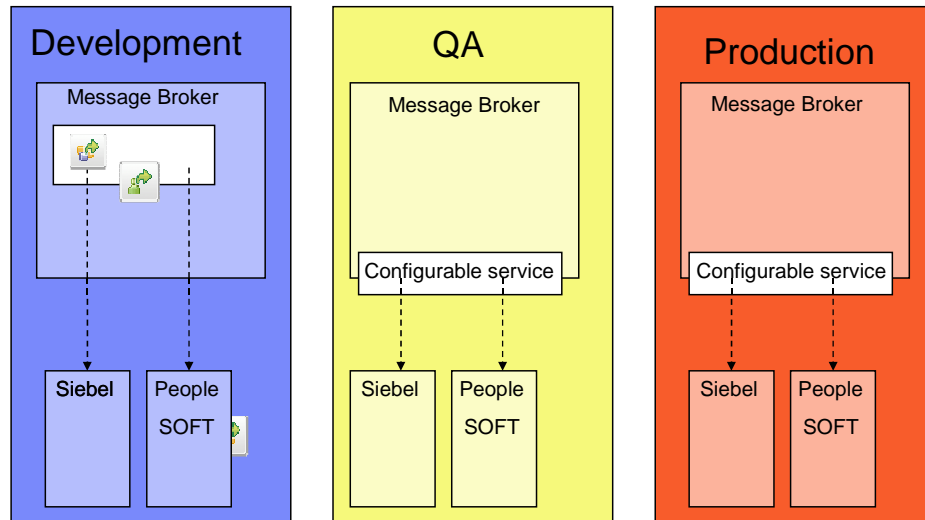
14

Adapter nodes – Iterative discovery and deployment

© 2010 IBM Corporation

The barfile is finally deployed to the production instance of the Message Broker. Again, the barfile contains the details of the development application system. To avoid this when using Message Broker version 6.1, you must manually rebuild the barfile, so that it contains the connection details and security credentials of the appropriate application system.

Application promotion using configurable services in version 7 (1)



15

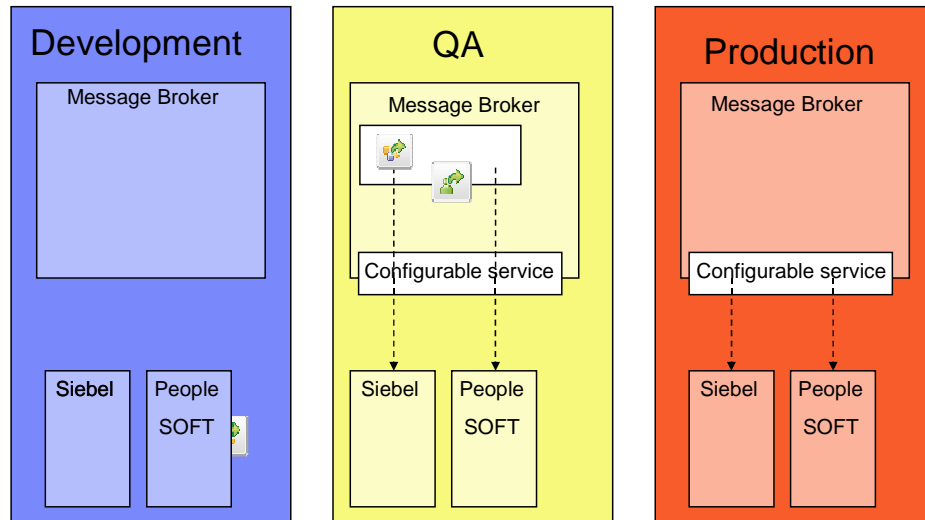
Adapter nodes – Iterative discovery and deployment

© 2010 IBM Corporation

In Message Broker version 7, the connection details and security credentials have been externalized using configurable services.

Initially, the barfile will connect to the development application system, as before.

Application promotion using configurable services in version 7 (2)



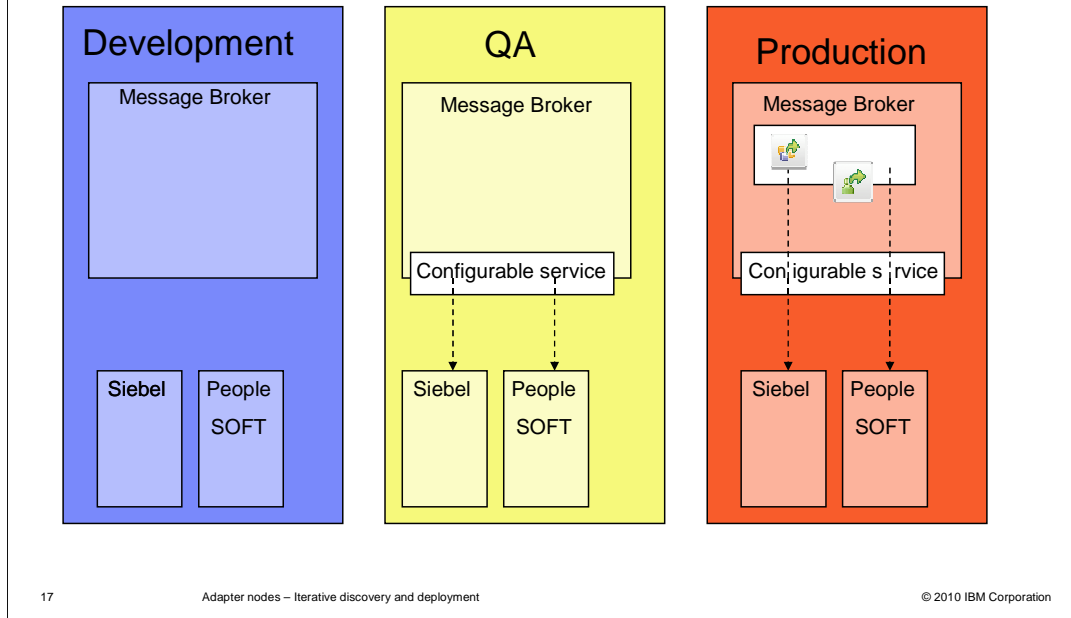
16

Adapter nodes – Iterative discovery and deployment

© 2010 IBM Corporation

When moving to the QA instance of Message Broker, the same barfile is deployed unchanged from the development system. However, the connection details have been provided using new configurable services, referenced in the message flow and adapter. These values are defined and set using Message Broker Explorer, or the appropriate Message Broker command.

Application promotion using configurable services in version 7 (3)



Similarly, when moving to the production system, the barfile is deployed unchanged. Configurable services are again defined, which provide the connection details for the production system. These are used by the message flow and adapter.

This approach also means that the connection details, including the security credentials of the production application instance, do not have to be disclosed to the application development community. They can be restricted to just the administration staff, who is responsible for creating and maintaining secure access to the application system.

It also means that the impact of changing the password for the production system is restricted to just the configurable service, and not the adapter definition or barfile. Configurable services can be easily updated to reflect this type of change.



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_Adapters_Iterative_Development.ppt

This module is also available in PDF format at: ..\\WMB7_Adapters_Iterative_Development.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](http://www.ibm.com/legal/copytrade.shtml)" 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.

© Copyright International Business Machines Corporation 2010. All rights reserved.