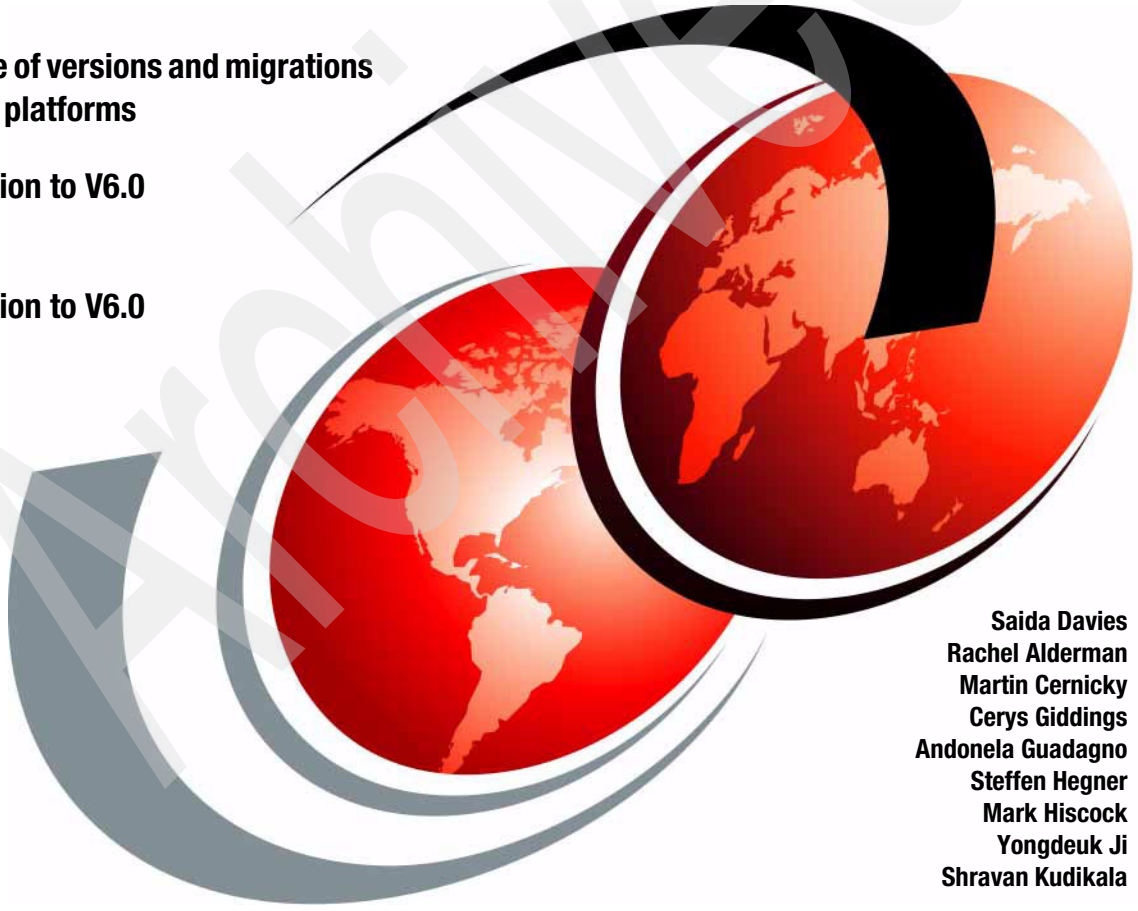
WebSphere® software

IBM

Migrating to WebSphere Message Broker Version 6.0

- Coexistence of versions and migrations on multiple platforms
- V2.1 migration to V6.0
- V5.0 migration to V6.0



Saida Davies
Rachel Alderman
Martin Cernicky
Cerys Giddings
Andonela Guadagno
Steffen Hegner
Mark Hiscock
Yongdeuk Ji
Shravan Kudikala

Redbooks

ibm.com/redbooks

International Technical Support Organization

Migrating to WebSphere Message Broker Version 6.0

March 2006

SG24-7198-00

Note: Before using this information and the product it supports, read the information in “Notices” on page xv.

First Edition (March 2006)
This edition applies to:

Version	Release	Modification	Product name	Product Number	Platform
6	**0**	**0**	WebSphere Message Broker	5724J0500	Windows
6	**0**	**0**	WebSphere Message Broker	5724J0501	AIX
6	**0**	**0**	WebSphere Message Broker	5655M7400	z/OS
6	**0**	**0**	WebSphere Message Broker	5724J0505	Linux
6	**0**	**0**	WebSphere Message Broker Rules and Formatter Extension	5724J0600	Windows
5	**0**	**1**	WebSphere Business Integration Message Broker	5724E2601	Windows
5	**0**	**1**	WebSphere Business Integration Message Broker	5724E2600	AIX
5	**0**	**1**	WebSphere Business Integration Message Broker	5655K6000	z/OS
5	**0**	**1**	WebSphere Business Integration Message Broker	5724E2604	Linux
5	**0**	**1**	WebSphere Business Integration Message Broker with Rules and Formatter Extension	5724E7301	Windows
2	**1**	**0**	WebSphere MQ Integrator	5724A8201	Windows
2	**1**	**0**	WebSphere MQ Integrator	5724A8200	AIX
2	**1**	**0**	WebSphere MQ Integrator	5655G9700	z/OS
5	**3**	**0**	WebSphere MQ	5724B4100	Windows
5	**3**	**0**	WebSphere MQ	5724B4101	AIX
5	**3**	**1**	WebSphere MQ	5655F1000	z/OS
5	**3**	**0**	WebSphere MQ	5724B4104	Linux

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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 and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

Redbooks (logo)  ™
pSeries®
z/OS®
zSeries®
AIX®
ClearCase®
Cloudscape™
DB2 Universal Database™
DB2®
Everyplace®
IBM®
NetView®
Parallel Sysplex®
Rational®
Redbooks™
SupportPac™
WebSphere®

The following terms are trademarks of other companies:

Java, Javadoc, Solaris, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries 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 trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

This IBM® Redbook will help you to migrate WebSphere® MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0. It begins with an overview of the WebSphere Message Broker V6.0 solution, which plays a key role in the integration of disparate applications and platforms by providing functional and transport capabilities to support and facilitate enterprise-level business integration. Then it explains the new features and enhanced capabilities in WebSphere Message Broker V6.0, and describes the migration path from the previous supported versions.

The book provides extensive guidance and instruction about the various activities involved in migration including planning, backing up resources, migration steps, verification, coexistence, and restoration. It covers the migration of development resources, Configuration Manager, User Name Server, and brokers on multiple operating systems. It also shows you how to move the Configuration Manager component and development resources to other platforms.

The book covers both generic and platform-specific considerations. It also demonstrates the migration process, with the help of coexistence, on several major platforms using two scenarios, one for WebSphere MQ Integrator V2.1 and the other for WebSphere Business Integration Message Broker V5.0. The information included in this redbook complements but does not replace product documentation.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Hursley Center.

Figure 1 Steffen, Yongdeuk, Mark (back row); Martin, Andonela, Rachel (middle row); Shravan, Saida, Cerys (front row)

Saida Davies is a Project Leader for the International Technical Support Organization (ITSO) and has 17 years of experience in IT. She has published several Redbooks™ about various business integration scenarios. Saida has experience in the architecture and design of WebSphere MQ solutions, extensive knowledge of the IBM z/OS® operating system, and a detailed working knowledge of both IBM and Independent Software Vendor (ISV) operating system software. In a customer-facing role as a Senior IT Specialist with IBM Global Services, her role included the development of services for z/OS and WebSphere MQ within the z/OS and Windows® platform. This covered the architecture, scope, design, project management and implementation of the software on standalone systems and on systems in a Parallel Sysplex® environment. Saida has received Bravo awards for her project contributions. She has a degree in Computer Studies and her background includes z/OS systems programming. Saida supports Women in Technology activities, contributing and participating in their meetings.

Rachel Alderman is a Software Engineer and has worked in the WebSphere Brokers Level 3 Service team since joining IBM Hursley in 2001. In addition to her technical job, she has been a games designer for the annual IBM ThinkPad Challenge since 2002. Rachel graduated from the University of Durham with a first class honours degree in Computer Science, and is a Member of the British Computer Society (MBCS). She participated in the review and completion of this redbook through her additional, post-residency contribution.

Martin Cernicky is an IT Specialist with IBM Software Services in the Czech Republic, and has 16 years of experience in Information Technology. He joined IBM in 1995 and is currently a member of the pSeries® support team. Martin has extensive knowledge of AIX/pSeries systems and solutions. For the last five years, he has supported WebSphere MQ middleware messaging and

WebSphere Message Broker. He has designed and implemented solutions using the WebSphere MQ family products, and has experience in the implementation of messaging and broker solutions on distributed platforms, especially on AIX® and Windows. He holds a degree in Automated Technology Systems from the Czech Institute of Technology.

Cerys Giddings is a usability practitioner for WebSphere Message Broker, and has worked on many of the usability enhancements for the product over the last two versions. She has worked on the WebSphere MQ family of products since joining IBM Hursley in 2000. A former team leader of the WebSphere MQ Test organization, Cerys has participated in producing the IBM Certified System Administrator WebSphere Business Integration Message Broker V5 certification tests. She is co-author of the IBM Redbooks *WebSphere Business Integration Message Broker Basics*, *WebSphere InterChange Server Migration Scenarios* and *WebSphere Message Broker Basics*. She has more than 10 years of experience in providing IT education and support. Cerys holds both an undergraduate degree and a Masters from the University of Wales, and the BCS Professional Examinations at Certificate and Diploma levels. She has made an additional, post-residency writing contribution to this redbook.

Andonela Guadagno is a Senior IT Specialist for WebSphere Business Integration in Brazil. She has 17 years of experience in Information Technology, which includes seven years working on the design and implementation of WebSphere MQ-based application integration solutions, and assisting IBM customers with business integration projects on a variety of platforms. She has also taught classes on business integration and WebSphere MQ Workflow. Andonela holds a degree in Mathematics from the University of Santo Andre, São Paulo, Brazil.

Steffen Hegner is an IT Specialist with IBM Germany. After earning a degree in microprocessor technology, he worked for two years in Information Technology before joining IBM in 1990. Steffen worked in IBM Global Services until 2002; his responsibilities included working on customer projects involving designing and implementing broker and messaging solutions. He then joined IBM Software Group. Steffen's areas of expertise include IBM WebSphere Business Integration products, Java™, and C and C++ software development.

Mark Hiscock joined IBM in 1999, while also earning a Computer Science degree at Portsmouth University. He has worked in the Hursley Park Laboratory in the United Kingdom testing the IBM middleware suite of applications, from WebSphere MQ Everyplace® to WebSphere Message Brokers. He now works as a customer scenario tester for WebSphere MQ and WebSphere Message Brokers, basing his testing on real world customer scenarios. Mark has made an additional, post-residency contribution by reviewing this redbook.

Yongdeuk Ji is an Advisory IT Specialist who supports WebSphere MQ and WebSphere Message Broker in IBM Technical Sales Support, IBM Korea. Since joining IBM in 1999, he has engaged in solution design, implementation and technical support on distributed platforms for major enterprise application integration customers such as POSCO. His areas of expertise includes the implementation of the WebSphere Business Integration product family on the Linux® platform.

Shravan Kudikala holds a Bachelor of Technology degree in Computer Science and Engineering from Jawaharlal Nehru Technological University. He has 6 years of experience in Information Technology. Before joining IBM in 2003, he worked as a systems and application developer. He currently works as a software engineer for IBM India Software Labs in the change team, providing support for WebSphere Business Integration Message Broker. His areas of expertise include resolving customer problems, defect fixing, programming, and skills transfer. Shravan developed the SupportPac™ "IA97- Any Queue Manager Input Node" as an extension to the product.

The team thanks the following people for their contributions and support to this project:

Miles Banister, WebSphere Message Broker Level 3 Service
IBM Hursley, UK

Ralph Bateman, WebSphere Message Broker Level 3 Service
IBM Hursley, UK

Jason Edmeades, WebSphere MQ Service Architect
IBM Hursley, UK

Pete Edwards, WebSphere Message Broker System Test
IBM Hursley, UK

Dave Gorman, WebSphere Message Broker Development
IBM Hursley, UK

Robert Haimowitz, z/OS systems and WMB setup
ITSO Poughkeepsie, USA

James P Hodgson, WebSphere MQ & ESB Delivery - Test
IBM Hursley, UK

Colin Holyoake, Networking and Windows servers
IBM Hursley, UK

Martyn Honeyford, WebSphere Message Broker Development
IBM Hursley, UK

Matt Lucas, WebSphere Message Broker Development
IBM Hursley, UK

Colin Paice, WebSphere MQ Scenarios Test
IBM Hursley, UK

Alek Seeleman, WebSphere Message Broker Toolkit Development
IBM Toronto, Canada

Simon Stone, WebSphere Message Broker Build Team
IBM Hursley, UK

The team thanks the following people for reviewing the book:

Miles Banister, WebSphere Message Broker Level 3 Service
IBM Hursley, UK

Ralph Bateman, WebSphere Message Broker Level 3 Service
IBM Hursley, UK

Kirstine Clapperton, WebSphere Message Broker Level 3 Service
IBM Hursley, UK

Rob Convery, WebSphere MQ Brokers System Test
IBM Hursley, UK

Trevor Dolby, WebSphere MQ Brokers Development
IBM Hursley, UK

Dave Gorman, WebSphere Message Broker Development
IBM Hursley, UK

Guy Hochstetler, WebSphere MQ Software Services
IBM Piscataway, USA

Martyn Honeyford, WebSphere Message Broker Development
IBM Hursley, UK

Laurence Hook, WebSphere Message Broker Level 3 Service
IBM Hursley, UK

Callum Jackson, WebSphere IT Specialist
IBM Hursley, UK

Matt Lucas, WebSphere Message Broker Development
IBM Hursley, UK

Alasdair Paton, WebSphere MQ Brokers System Test
IBM Hursley, UK

Alek Seeleman, WebSphere Message Broker Toolkit Development
IBM Toronto, Canada

Crystal Su, WebSphere Message Broker Toolkit Development
IBM Toronto, Canada

Xiaoming Zhang, WebSphere MQ Brokers Development
IBM Hursley, UK

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our Redbooks to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. 8IB IBM United Kingdom Ltd
  ITSO, Building 662
  Raleigh, NC 27709

Part 1

WebSphere Message Broker Version 6.0

This redbook is divided into three parts. In the first part we discuss the scope and aims of the book, including a brief introduction to the scenarios that are used. The second and third parts cover the detailed migration from WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0, respectively.

In this part, an overview of the capabilities and positioning of WebSphere Message Broker V6.0 is given, together with details of new features, enhancements, and changes for this version of the product. Various migration considerations and planning for WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 are also covered.

This part covers the following:

- Introduction of WebSphere Message Broker V6.0
- Functional changes and enhancements for V2.1 and V5.0 users
- Migration considerations and planning

1

Introduction to the redbook

Here we provide an overview of the scope of this redbook, information about its intended audience, and the assumptions made by the redbook team. We also discuss aspects of migration to WebSphere Message Broker V6.0 and provide a brief overview of each of the chapters.

1.1 The scope of this redbook

The aim of this book is to provide a step-by-step guide to migrating from WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0.

Most of the instructions and information provided here also apply to users migrating from WebSphere MQ Integrator Broker V2.1 and WebSphere MQ Event Broker V 2.1. However, we recommend that for specific instructions on V2.1 products you should consult the migration topics in the WebSphere Message Broker V6.0 documentation. For WebSphere Business Integration Event Broker V5.0 , WebSphere Business Integration Message Broker with Rules and Formatter Extension V 5.0, and WebSphere Business Integration Message Broker V5.1 , the migration instructions are identical except for the following:

- WebSphere Business Integration Event Broker V5.0

  Some of the instructions relating to the hmigration of application artifacts do not apply to this product (for example, the migration of mapping nodes).

- WebSphere Business Integration Message Broker with Rules and Formatter Extension V 5.0

  The migration of the Rules and Formatter Extension is briefly discussed in the broker migration chapters. Detailed instructions are provided separately in the New Era Of Networks documentation.

The migrations are demonstrated by migrating the existing WebSphere MQ Integrator V2.1 and the WebSphere Business Integration Message Broker V5.0 configurations end to end. Besides demonstrating the standard migration tasks, these migrations are expanded to utilize new features such as coexistence and the ability to migrate a Configuration Manager to a new platform.

After performing all the migration steps for the previous versions, we verified the steps to ensure that the migration was successful. We documented all the known problems and the problems encountered during the migrations, and created tips and workarounds.

Details about the platforms covered and the scenarios used are in 1.1.2, “Scenarios demonstrated” on page 5.

1.1.1 Intended audience

This book is divided into three parts, with each part aimed at a specific audience.

Part 1, “WebSphere Message Broker Version 6.0” on page 1 is aimed at management and architect roles. It is also useful for users interested in the positioning and new features of WebSphere Message Broker V6.0. or planning a migration from a previous version.

Part 2, “Migrating from WebSphere MQ Integrator V2.1” on page 105 is targeted at users working with an existing WebSphere MQ Integrator V2.1 environment who require detailed instructions on how to migrate to WebSphere Message Broker V6.0.

Part 3, “Migrating from WebSphere BI Message Broker V5.0” on page 311 addresses users working with an existing WebSphere Business Integration Message Broker V5.0 environment who want to migrate to WebSphere Message Broker V6.0. This part provides detailed technical instructions and guidance for the required steps for the migration.

This book is not targeted at new users and as such does not provide information about how to use the product beyond those tasks required for or associated with migration. Information about how to set up an environment on the previous versions and how to use all of the new functions of WebSphere Message Broker V6.0 is beyond the scope of this book. For detailed information about how to use WebSphere Message Broker V6.0, refer to *WebSphere Message Broker Basics*, SG24-7137, or to the product documentation.

Many new samples are available in WebSphere Message Broker that demonstrate how to use the new functions of the product.

1.1.2 Scenarios demonstrated

The migration scenarios described and demonstrated in this book cover a range of platforms. The instructions and information supplied for each of these platforms are also applicable to the platforms not covered in this book. Where known differences exist between platforms, these are detailed in the individual sections.

The migration scenarios demonstrated here were performed on the following products:

- WebSphere MQ Integrator V2.1
- WebSphere Business Integration Message Broker V5.0

Following are the platforms for which the migrations are demonstrated in this book:

- WebSphere MQ Integrator V2.1:
  - Windows

  - AIX
  - z/OS
- WebSphere Business Integration Message Broker V5.0:
  - Windows
  - AIX
  - z/OS
  - Linux (x86 platform)

The scenarios include:

- WebSphere MQ Integrator V2.1:
  - Migrating the Control Center
  - Migrating broker resources
  - Migrating message flows and message sets
  - Migrating the Configuration Manager
  - Moving the Configuration Manager to z/OS
  - Migrating the User Name Server
  - Migrating a Windows broker
  - Migrating New Era Of Networks Rules and Formatter resources
  - Migrating an AIX broker
  - Migrating a z/OS broker
- WebSphere Business Integration Message Broker V5.0:
  - Migrating the Message Brokers Toolkit
  - Migrating broker resources
  - Migrating message flows and message sets
  - Introducing a Message Brokers Toolkit on Linux
  - Migrating the Configuration Manager
  - Moving the Configuration Manager to AIX
  - Migrating the User Name Server
  - Migrating a Windows broker
  - Migrating New Era Of Networks Rules and Formatter resources
  - Migrating an AIX broker
  - Migrating a z/OS broker
  - Migrating a Linux broker

Further details about the specific scenarios and environment setups used in the migration can be found in Chapter 7, “WebSphere MQ Integrator Version 2.1 environment setup” on page 107 and in Chapter 17, “WebSphere Business Integration Message Broker Version 5.0 environment setup” on page 313.

1.1.3 What is not covered in the book

This section provides information about the topics not covered in this book.

Installation and use of WebSphere Message Broker

This book focuses on tasks relating to the migration of WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 and does not cover any other versions. Although the book describes the new and improved functionality of WebSphere Message Broker V6.0, only those features that are required to get an existing configuration up and running after the migration are included in the demonstration. Instructions on how to use features other than those required for a migration or setup relating to an existing environment are beyond the scope of this book.

This book does *not* cover:

- Details on the installation of software products, including WebSphere Message Broker V6.0; therefore, no step-by-step instructions are included.
- Installation of prerequisite and corequisite software, such as WebSphere MQ and operating system updates.
- Instructions about the installation, configuration, or use of any previous version of this product, unless required to migrate to WebSphere Message Broker.

Upgrades and migration of prerequisites

Upgrades and migration to existing software other than WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 are not included in this book. For example, instructions for the migration of WebSphere MQ V5.3 to WebSphere MQ V6.0 are not included. An exception to this is the upgrade of IBM Agent Controller Version 5.0.2 to Rational Agent Controller Version 6.0.0.1.

If an existing WebSphere MQ Integrator V2.1 or WebSphere Business Integration Message Broker V5.0 configuration is on an unsupported platform, then an upgrade to the operating system or prerequisite software may be required. This book does not discuss these migration routes nor the implications or actions required to upgrade or migrate to these new platforms, for example, migrating from Microsoft® Windows 2000 to Microsoft Windows 2003.

For assistance with applying updates or service levels to prerequisite software, refer to the documentation or service organization for the appropriate products.

Migrating components to new platforms

This book discusses and demonstrates how to move a migrated Configuration Manager and the Message Brokers Toolkit to a new platform. However, no instructions are provided for moving the entire set of installables or components of WebSphere Message Broker V6.0 or any previous version to a new machine (for example, how to move to a new operating system).

User-defined nodes

We do not describe the complete migration of user-defined nodes, but some guidance is given for this process. Migration of SupportPacs is not covered. For information about which SupportPacs are supported on WebSphere Message Broker V6.0, refer to:

http://www.ibm.com/software/integration/support/supportpacs/category.html

High-availability environments

We provide guidance and hints for migrating in a high-availability environment, but do not provide instructions on setting up such an environment within WebSphere Message Broker V6.0 or previous versions. Further considerations for migrating in a high-availability environment can be found in the WebSphere Message Broker V6.0 documentation. For instructions on setting up a high-availability environment, refer to:

http://www.ibm.com/developerworks/websphere/library/techarticles/0403_humphreys/0403_humphreys.html

Databases

We provide information and instructions for database tasks where required for migration tasks. These are only provided for DB2 Universal Database and not any of the other supported databases, for example, Oracle and Sybase. For the minimum version and service requirements for migration on databases other than DB2 Universal Database, refer to the system requirements documentation for WebSphere Message Broker at:

http://www.ibm.com/software/integration/wbimessagebroker/requirements/

For instructions on how to back up and restore tables on databases other than DB2 Universal Database, refer to the database documentation or the database administrator.

1.1.4 Assumptions

We make a number of assumptions in order to simplify the information and to make it useful and relevant to as many customers as possible, as follows:

- You are familiar with the use of either WebSphere MQ Integrator V2.1 or WebSphere Business Integration Message Broker V5.0, or other related products such as WebSphere MQ Integrator Broker V2.1 or WebSphere Business Integration Event Broker.
- You are familiar with the use of WebSphere MQ.

- You have a working configuration on the product versions mentioned in the previous section.
- Prerequisite and corequisite software for WebSphere Message Broker has been upgraded to supported versions, and includes any required fix packs.
- The latest fix pack for WebSphere MQ Integrator V2.1 or WebSphere Business Integration Message Broker V5.0 is installed, working, and tested on the current configuration before the migration.
- You have a set of tests available for testing their broker domain configuration and deployed resources.
- You have the appropriate security privileges on all the components that are involved in the migration, including operating system privileges where appropriate.

1.1.5 Overview of topics covered

This section gives a brief overview of the chapters in this book.

Chapter 1, "Introduction to the redbook"

This chapter describes the scope of this book, its intended audience, the assumptions made by the redbook team, and aspects of migration to WebSphere Message Broker V6.0.

Chapter 2, "WebSphere Message Broker overview"

This chapter gives a high-level introduction to WebSphere Message Broker, its components, and capabilities. The new functionality added in Version 6.0 is described briefly here. It also discusses how WebSphere Message Broker is utilized in the service-orientated architecture (SOA) and can be deployed as an advanced enterprise service bus (ESB).

Chapter 3, "What is new for WebSphere MQ Integrator V2.1 users"

This chapter describes the functional changes and enhancements since Version 2.1 and what WebSphere Message Broker V6.0 has to offer you. It provides a table of the enhancements and the version in which they first became available.

Chapter 4, "New features in WebSphere Business Integration Message Broker V5.0"

This chapter describes the functional changes and enhancements in WebSphere Message Broker V6.0 and what this version has to offer Version 5.0 and Version 5.1 users. It also includes a table of all of the enhancements since WebSphere Message Broker V5.0 and the version in which they first became available.

Chapter 5, “Migration considerations”

This chapter provides a non-technical overview of the considerations for the migration to WebSphere Message Broker. This includes the migration methods, supported software levels, preparing the environment for the migration along with recommendations for verifying the installation of WebSphere Message Broker V6.0.

Chapter 6, “Planning for migration”

This chapter gives a general overview of the common concepts and tasks involved in migrating to WebSphere Message Broker from the previous versions such as the concepts and importance of coexistence, mixed level components in domains, and the considerations for moving components between platforms.

Part 2, “Migrating from WebSphere MQ Integrator V2.1”

Part 2 discusses the detailed technical considerations and steps for performing a migration from a WebSphere MQ Integrator V2.1 environment to WebSphere Message Broker V6.0.

Chapter 7, “WebSphere MQ Integrator Version 2.1 environment setup”

This chapter describes the environment used by the redbook team to demonstrate a migration from WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0. This includes the machines, platforms, and software levels used in the migration scenario. The application resources used in the scenarios are documented in “Message flows for WebSphere MQ Integrator V2.1 scenario” on page 526. The chapter also provides a background on the differences between the WebSphere MQ Integrator V2.1 and the WebSphere Message Broker V6.0 environments that affect the migration procedures in the following chapters.

Chapter 8, “WebSphere MQ Integrator V2.1 migration steps”

This chapter provides a high-level overview of the sequence and steps required for the migration between WebSphere MQ Integrator V2.1 and WebSphere Message Broker V6.0 (including considerations for Rules and Formatter components, user-defined nodes and parsers), as well as best practices for ensuring a successful migration.

Chapter 9, “Backing up the WebSphere MQ Integrator V2.1 domain”

This chapter provides instructions on how to back up a WebSphere MQ Integrator V2.1 domain including Control Center artifacts, components, and databases.

Chapter 10, "Migrating the WebSphere MQ Integrator V2.1 Control Center resources"

This chapter discusses the coexistence of the Control Center with the WebSphere Message Broker Toolkit, and provides instructions on exporting resources from the Control Center and the steps required to migrate to the workspace for the new WebSphere Message Broker Toolkit. It also introduces the concept of using the tooling on Linux (x86 platform).

Chapter 11, "Migrating the Configuration Manager"

This chapter provides technical details on the migration of the Configuration Manager, and confirmation that the migration is successful. It includes instructions for moving the Configuration Manager to a new platform, and a demonstration of a move to z/OS. It also includes a discussion of some of the enhancements to the Configuration Manager and the benefits they may bring in migrating from an earlier version.

Chapter 12, "Migrating the User Name Server"

This chapter demonstrates the steps required for the migration of the User Name Server, including verification of a successful migration.

Chapter 13, "Migrating the Windows Broker"

This chapter provides technical instructions about premigration tasks and migrating a broker on Windows. It also contains steps for the migration of deployed applications and New Era Of Networks components. The chapter also provides information about how to verify whether the migration steps for the broker are successful.

Chapter 14, "Migrating the AIX Broker"

This chapter provides technical instructions about premigration tasks and migrating a broker on AIX. It also contains steps for the migration of deployed applications and New Era Of Networks components. It also provides information about how to verify whether the migration steps for the broker are successful.

Chapter 15, "Migrating the z/OS Broker"

This chapter provides technical instructions about premigration tasks and migrating a broker on z/OS. It also contains steps for the migration of deployed applications and New Era Of Networks components. It also provides information about how to verify whether the migration steps for the broker are successful.

Chapter 16, "Migration verification"

This chapter details the steps that are required after a migration to WebSphere Message Broker V6.0 including tips for verifying whether the migration process is successful. It also provides instructions for restoring a WebSphere MQ Integrator V2.1 environment.

Part 3, "Migrating from WebSphere BI Message Broker V5.0"

This section of the book demonstrates the migration of a WebSphere Business Integration Message Broker V5.0 environment to WebSphere Message Broker V6.0.

Chapter 17, "WebSphere Business Integration Message Broker Version 5.0 environment setup"

This chapter describes the environment used by the redbook team to demonstrate a migration from WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0. This includes the machines, platforms, and software levels used in the migration scenario. The application resources used in the scenarios are documented in "Message flows for WebSphere Business Integration Message Broker V5.0 scenario" on page 533. The chapter also provides a background on the differences between the WebSphere Business Integration Message Broker V5.0 and WebSphere Message Broker V6.0 environments that affect the migration procedures in the following chapters.

Chapter 18, "WebSphere Business Integration Message Broker V5.0 migration steps"

This chapter provides a high-level overview of the sequence and steps required for the migration between WebSphere Business Integration Message Broker V5.0 and WebSphere Message Broker V6.0, including considerations for New Era Of Networks components, user-defined nodes and parsers, as well as best practices for ensuring a successful migration.

Chapter 19, "Backing up the WebSphere Business Integration Message Broker V5.0 domain"

This chapter provides step-by-step instructions on how to back up a WebSphere Business Integration Message Broker V5.0 domain. This includes backing up application artifacts from the Message Brokers Toolkit, domain configuration, and databases.

Chapter 20, "Migrating WebSphere Message Broker V5.0 and V5.1 tools and resources"

This chapter discusses the coexistence of the Version 5.0 and Version 5.1 Message Brokers Toolkit with the Version 6.0 Message Brokers Toolkit. It

also covers the migration of the appropriate nodes along with guidance for the migration of user-defined nodes. It also provides a demonstration on using the Message Brokers Toolkit on Linux (x86 platform).

Chapter 21, "Migrating the WebSphere Business Integration Message Broker V5.0 Configuration Manager"

This chapter provides technical details on the migration of the Configuration Manager and confirmation that the migration is successful. It includes instructions for moving the Configuration Manager to a new platform, and a demonstration of a move to AIX. It also includes a discussion of some of the enhancements to the Configuration Manager and the benefits they may bring in migrating from an earlier version.

Chapter 22, "Migrating the User Name Server"

This chapter demonstrates the steps required for the migration of the User Name Server including verification of a successful migration.

Chapter 23, "Migrating a Windows broker"

This chapter provides technical instructions about premigration tasks and migrating a broker on Windows. It also contains steps for the migration of deployed applications and New Era Of Networks components. It also provides information about how to verify whether the migration steps for the broker are successful.

Chapter 24, "Migrating an AIX broker"

This chapter provides technical instructions about premigration tasks and migrating a broker on AIX. It also contains steps for the migration of deployed applications and New Era Of Networks components. It also provides information about how to verify whether the migration steps for the broker are successful.

Chapter 25, "Migrating the z/OS broker"

This chapter provides technical instructions about premigration tasks and migrating a broker on z/OS. It also contains steps for the migration of deployed applications and New Era Of Networks components. It also provides information about how to verify whether the migration steps for the broker are successful.

Chapter 26, "Migrating the Linux broker"

This chapter provides technical instructions about premigration tasks and migrating a broker on Linux. It also contains steps for the migration of deployed applications and New Era Of Networks components. It also

provides information about how to verify whether the migration steps for the broker are successful.

Chapter 27, “Verifying migration to WebSphere Message Broker V6.0”

This chapter details the steps that are required after a migration to WebSphere Message Broker V6.0, including tips for verifying whether the migration process was successful. It also provides instructions for restoring a WebSphere Business Integration Message Broker V5.0 environment.

2

WebSphere Message Broker overview

This chapter gives an introduction to WebSphere Message Broker and how the product provides a solution to the problems of application integration that many organizations face today. It provides a brief overview of application integration, service-oriented architecture (SOA), and the concept of an enterprise service bus (ESB). It then discusses the specific capabilities and components of WebSphere Message Broker.

This chapter is intended as background information to the positioning of WebSphere Message Broker as an advanced ESB and to provide a high-level overview of the main functions of WebSphere Message Broker.

2.1 Application integration

Application integration at a high level refers to solutions implemented to integrate software applications within and between organizations. Historically, application integration has been concerned with the integration of software applications between different departments and divisions within companies, or new acquisitions. Often these applications vary considerably across departments in an organization, existing on different platforms, written in different programming languages, and using different data formats. Integrating the applications is a more practical and cost-effective solution than the alternative of rewriting the existing applications.

Application integration is now considered to be important in enabling flexibility within organizations to be able to respond to shifting marketplace demands to create an *on demand* business. Organizations increasingly should be able to connect to suppliers, business partners, and customers, as well as within their own organizations. The advantages of an application integration solution include the notion of *future proofing*, enabling new applications and standards to be easily plugged into the existing architecture.

Application integration is a big challenge for enterprises. IBM provides a number of software solutions and offerings to assist companies with integrating their applications. WebSphere Message Broker is an important part of these solutions; how it fits together with these offerings is described in the following sections.

In application integration there are a number of terms that are used to describe the solutions and architectures used to implement those solutions. The following sections briefly discuss the principles of an SOA, Message-Oriented Middleware (MOM), and an ESB, and the role of WebSphere Message Broker in these solutions.

2.1.1 Service-oriented architecture

Service-oriented architecture (SOA) is promoted as a framework for IT organizations to build their IT applications upon, which will facilitate growth and development into the future. The definition of an SOA is that it is an application architecture within which all functions are defined as independent services with well-defined, invokable interfaces that can be called in defined sequences to form business processes.

Within an SOA, applications, information, and other IT assets are viewed as services or “building blocks.” Each of these services can be mixed and matched to create new, flexible business processes. Using an SOA provides the flexibility to take business processes and the underlying IT infrastructure and treat them as

services that can be reused and combined to address changing business priorities.

The interfaces are defined to be standardized and independent of hardware platform, operating system, and the programming language in which the service is implemented. This enables them to quickly and flexibly exchange structured information about business events, including messages, documents, and “business objects. This flexibility enables new and existing applications to be easily and quickly combined to address changing business needs, and the ability to easily combine and choreograph applications allows IT services to more readily reflect business processes.

WebSphere Message Broker can be used in the implementation of an SOA because it provides a mechanism for connecting, routing, and transforming business data from a variety of transports without the need to change the underlying applications generating the data. The method in which WebSphere Message Broker is used in application integration and SOA is described in detail in the next few sections.

2.1.2 Message-Oriented Middleware

Message-Oriented Middleware (MOM) enables applications to package their business information as messages, which are handled by the MOM, for example WebSphere MQ. Using such middleware enables the distribution of messages to be taken care of without the need to write complex networking code. The addition of WebSphere Message Broker further enhances the flow and distribution of information by enabling the transformation and intelligent routing of messages without the need to change either the applications that are generating the messages or the applications that are consuming them. This style of connectivity is called a *message-oriented architecture* and is based on applications communicating through messages.

2.1.3 Enterprise service bus (ESB)

An ESB is an implementation of an SOA that provides the following functionalities:

- Distributes information to all parts of your business easily and rapidly.
- Hides differences in platform, software architecture, and network protocols.
- Assures delivery of information, even when systems and networks go offline.
- Reroutes, logs, and enriches information, under central control, without rewriting applications.

- Deploys solutions incrementally, project by project, to better manage expense.
- Minimizes risk by using industry standard interfaces and protocols.

An ESB can be constructed from many of the business integration middleware products available from IBM, including WebSphere MQ and WebSphere Message Broker. An ESB provides an open, standards-based connectivity infrastructure for an SOA with the following characteristics:

- Is standards-based.
- Enables all parts of a business to react instantly to new information.
- Minimizes risk by using industry standard interfaces and protocols.
- Overcomes differences in platform, software architecture, and network protocols.
- Assures delivery of transactions, even when systems and networks go offline.
- Reroutes, logs, and enriches information without rewriting applications.
- Provides an infrastructure that is highly distributed and yet can be managed centrally.
- Distributes data throughout your business and beyond to your customers and business partners.
- Spans different operating systems, programming models, application types, and locations.
- Is deployed incrementally, project by project, to better manage expense.
- Combines new and existing technologies and standards.
- Supports message-oriented, service-oriented, and event-oriented architecture.

WebSphere Message Broker is promoted as an *advanced* ESB due to its extensive capabilities beyond the routing and handling of different messages and sources across multiple protocols and standards. The following capabilities enable WebSphere Message Broker to extend an ESB:

- Distributes any type of information across and between multiple diverse systems and applications, providing delivery of the right information in the right format at the right time.
- Reduces the number of point-to-point interconnections and simplifies application programming by removing integration logic from the applications themselves.
- Using a powerful publish/subscribe matching engine, routes information in real time based on topic and content to any endpoint.

- Validates and transforms messages in-flight between any combination of different message formats, including Web Services, other XML and non-XML formats.
- Routes messages based on (evaluated) business rules to match information content and business processes.
- Improves business agility by dynamically reconfiguring information distribution patterns without reprogramming end-point applications.
- Accesses control to securely deliver personalized information to the right place at the right time.

As discussed in the previous sections, WebSphere Message Broker is an important part of the IBM portfolio for SOA and ESB solutions. It is also a mature product with extensive capabilities, supporting multiple platforms and communication protocols. It can therefore be used in a broad range of scenarios. The specific functions and aspects of WebSphere Message Broker are covered in detail in the following section.

2.2 WebSphere Message Broker

WebSphere Message Broker is a powerful information broker that allows business data and information in the form of messages to flow between disparate applications across multiple hardware and software platforms. Business rules can be applied to the data flowing through the message broker to route, store, retrieve, and transform the information.

2.2.1 Versions of WebSphere Message Broker

WebSphere Message Broker V6.0 has three versions of the message broker. A brief overview of the functionalities and differences of the different versions is given in the following sections.

WebSphere Event Broker

WebSphere Event Broker is a powerful engine for the distribution and routing of messages from disparate applications. It can distribute information and data generated by business events in real time to people, applications, and devices throughout an enterprise. WebSphere Event Broker supports multiple transport protocols and extends the flow of information in an organization beyond point to point, utilizing flexible distribution mechanisms such as publish/subscribe and multicast.

WebSphere Message Broker

WebSphere Message Broker contains all the functionalities of WebSphere Event Broker and includes additional capabilities to enable storage, transformation, and enrichment of data flowing through the broker. The detailed capabilities of the product described in 2.2.2, “Capabilities of WebSphere Message Broker” on page 20 are specifically based upon the functional capabilities of the WebSphere Message Broker.

Rules and Formatter Extension

This includes the Rules and Formatter Extension from New Era Of Networks that provides Rules and Formatter nodes and associated runtime elements. These maintain the functionality supplied with earlier releases of WebSphere MQ Integrator. The functionality provided by the Rules and Formatter Extension is not discussed any further in this chapter.

2.2.2 Capabilities of WebSphere Message Broker

The primary capabilities of WebSphere Message Broker are message routing, message transformation, message enrichment, and publish/subscribe. These capabilities make WebSphere Message Broker a powerful tool for business integration.

Message routing

WebSphere Message Broker provides connectivity for both standard-based and nonstandard-based applications and services. The routing can be simple point-to-point routing or it can be based on matching the content of the message to business rules defined to the broker.

WebSphere Message Broker contains a choice of transports that enable secure business to be conducted at virtually any time and any place, providing powerful integration using mobile, telemetry, and Internet technologies. WebSphere Message Broker is built upon WebSphere MQ and supports the same transports, and it also extends the capabilities of WebSphere MQ by adding support for other protocols including real-time Internet, intranet, and multicast endpoints.

WebSphere Message Broker supports the following transports:

- WebSphere MQ Enterprise Transport
- WebSphere MQ Web Services Transport
- WebSphere MQ Real-time Transport
- WebSphere MQ Multicast Transport
- WebSphere MQ Mobile Transport
- WebSphere MQ Telemetry Transport
- JMS Transport

Message transformation and enrichment

Transformation and enrichment of in-flight messages is an important capability of WebSphere Message Broker. This enables business integration without the need for additional logic in the applications.

Messages can be transformed between applications to use different formats, for example, from a custom format in a legacy system to XML for use with a Web service. This provides a powerful mechanism to unify organizations, because business information can now be distributed to applications that handle completely different message formats without a need to reprogram or add to the applications.

Messages can also be transformed and enriched by integration with multiple sources of data such as databases, applications, and files to perform any type of data manipulation including logging, updating, and merging. Business information from messages flowing through the broker can be stored in databases, or extracted from databases and files and added to the message for processing in the target applications.

Complex manipulation of message data can be performed using the facilities provided in the Message Brokers Toolkit such as Extended Structured Query Language (ESQL) and Java.

Message transformation and enrichment depend on the message broker's understanding of the structure and content of the incoming message. Self-defining messages such as XML contain information about their own structure and format. Other messages, such as those with a custom format, should have a message definition of their structure created to enable the message broker to transform or enhance them. The Message Brokers Toolkit contains facilities for defining messages to the message broker, which are discussed in more detail in the following section.

Publish/subscribe

The simplest method of routing messages is to use point-to-point messaging, by sending messages directly from one application to another. Publish/subscribe provides an alternative method of messaging where messages are sent to all applications that have subscribed to a particular topic.

The broker handles the distribution of messages between publishing applications and subscribing applications. Applications can publish on or subscribe to many topics as well as apply more sophisticated filtering mechanisms.

The use of publish/subscribe and the related technology of multicast improve the information flow around the business by moving away from hard-coded point-to-point links to these more flexible distribution mechanisms.

2.2.3 Components of WebSphere Message Broker

WebSphere Message Broker is comprised of two principal parts:

- A *Development environment* for the creation of message flows, message sets, and other broker application resources
- A *Run-time environment* that contains the components for running the broker applications created in the development environment

Development environment

The development environment develops the logic that the broker uses to process messages from business applications. The Message Brokers Toolkit is the tool that is used to develop the applications that are run in the message broker run-time environment. These are:

- Message flows
- Message sets

Message flows

Message flows are applications that provide the logic that the broker uses to process messages from business applications. Message flows are created in the Message Brokers Toolkit using the graphical Message Flow editor to click and place nodes and connect them. Each node performs some basic logic, and a selection of nodes are provided to perform particular tasks that can be combined to perform complex manipulations and transformations of messages.

A choice of methods is available for defining transformations of data. These methods can be used to match different types of data or the skills of the broker application developer:

- ESQL
- Java
- Extensible Stylesheet Language for Transformations
- Drag-and-drop mappings

The nodes in the message flows define the source and the target transports of the message, any transformations and manipulations based on the business data, and any interactions with other systems such as databases and files.

Message sets

A message set is a definition of the structure of the messages that are processed by the message flows in the broker. As mentioned in the section, “Message transformation and enrichment” on page 21, the message broker must know the structure of a message in order to enable a message flow to manipulate or transform the message. This definition can then be used for verification of the

message structure in the message broker, and within the Message Brokers Toolkit to assist with the construction of message flows and mappings.

Message sets are compiled from deployment to a message broker as a message dictionary, which provides a reference to check the structure of the messages as they flow through the broker.

Broker Application Development perspective

The Broker Application Development perspective is a part of the Message Brokers Toolkit that is used to design and develop message flows and message sets. This contains editors to create message flows, transformation code such as ESQL, and message definitions.

Run-time environment

The run-time environment is a set of components that are required to deploy and run the developed broker applications, such as message flows and message sets, and their configuration.

Broker

The broker is a set of execution processes that hosts and runs message flows. When a message arrives at the broker from a business application, the broker processes the message before passing it on to one or more other business applications. The broker routes, transforms, and manipulates messages according to the logic that is defined in its message flows.

A broker uses WebSphere MQ as the transport mechanism to communicate with the Configuration Manager from which it receives configuration information, and any other brokers to which it is associated.

Each broker has a database in which it stores the information in order to process messages at run-time.

Execution groups

Execution groups enable message flows within the broker to be grouped together. Each broker contains a default execution group, and more execution groups can be created as long as unique names are given to them within the broker.

Each execution group is a separate operating system process so that the contents of an execution group remain separate from the contents of another execution group within the same broker. This can be useful for isolating pieces of information for security. These message flows then execute in separate address spaces or as unique processes.

Broker applications such as message flows and message sets are deployed to a specific execution group, but the same message flows and message sets can be run in different execution groups in order to enhance performance.

Configuration Manager

The Configuration Manager is the interface between the Message Brokers Toolkit and the brokers in the broker domain. It stores configuration details for the broker domain in an internal repository, providing a central store for resources in the broker domain.

The Configuration Manager is responsible for deploying message flow applications to the brokers. It also reports back on the progress of the deployment and the status of the broker. When the Message Brokers Toolkit connects to the Configuration Manager, the status of the brokers in the domain is derived from the configuration information stored in the Configuration Manager's internal repository.

Broker domain

Brokers are grouped together in broker domains. The brokers in a single broker domain share a common configuration that is defined in the Configuration Manager. A broker domain contains one or more brokers and a single Configuration Manager. It may also contain a User Name Server. The components in a broker domain may exist on multiple machines and platforms, and are connected using WebSphere MQ channels.

A broker can only belong to one broker domain.

User Name Server

A User Name Server is an optional component that is only required where publish/subscribe broker applications are run, and where extra security is required for applications to publish or subscribe to topics. The User Name Server provides authentication for topic-level security for users and groups performing publish/subscribe operations.

Broker Administration perspective

The Broker Administration perspective is a part of the Message Brokers Toolkit that is used for the administration of the broker domains defined to the Message Brokers Toolkit. This perspective is also used for the deployment of message flows and message sets to brokers in the defined broker domains.

The Broker Administration perspective also contains tools for creating broker archive files that are used to deploy broker application resources such as message flows and message sets. Other tools available include enqueue and dequeue for putting messages into, and getting them from, the WebSphere MQ queues to help test message flows.

3

What is new for WebSphere MQ Integrator V2.1 users

This chapter describes the new features of WebSphere Message Broker V6.0 on a function-by-function basis. An overview is given for each of the major changes and enhancements made in WebSphere MQ Integrator V2.1.

This chapter contains a list of key changes in each release since WebSphere MQ Integrator V2.1.

We also assess differences between WebSphere MQ Integrator V2.1 and WebSphere Message Broker V6.0 to highlight those aspects that are of most interest to a WebSphere MQ Integrator user.

3.1 Installation and initial user experience

A number of enhancements have been made to the installation and initial user experience for WebSphere Message Broker. Many of these changes are specific to Windows, but there are enhancements that improve the experience for other platforms as well.

3.1.1 Installation

The prerequisite software must be installed to run WebSphere Message Broker. The installation wizards no longer enforce installation of this software.

Installation of WebSphere Message Broker is now accomplished using InstallShield for Multiplatforms, which provides a consistent interface for installation on different platforms (excluding z/OS.)

The WebSphere Message Broker runtime and the Message Brokers Toolkit are now separated and need to be installed individually.

3.1.2 Preinstalled LaunchPad

On Windows, a preinstalled LaunchPad is provided to assist with the installation of prerequisite software. It also assists in the installation of WebSphere Message Broker runtime and Message Brokers Toolkit. This LaunchPad provides two methods for performing a WebSphere Message Broker install:

- Express installation
- Advanced installation

The Express installation installs the WebSphere Message Broker, the Message Brokers Toolkit, and the prerequisite products. This installation is suitable for test and development purposes and enables you to start working with the product quickly. It chooses default settings and requires minimum intervention.

The Advanced installation options are designed for more experienced users who can use it to manually choose and install settings according to their individual preferences for prerequisites and also for WebSphere Message Broker and the Message Brokers Toolkit.

3.1.3 WebSphere Message Broker on DVD

WebSphere Message Broker is supplied on DVD for Windows and on Intel® for Linux. This is to reduce the number of CD changes required, particularly for the Message Brokers Toolkit that is supplied on four compact disks for each platform. WebSphere Message Broker for the other distributed platforms is

supplied on CD, as are Windows and Linux (x86 platform), for users without a DVD player.

WebSphere Message Broker can also be obtained in electronic form for existing users.

3.1.4 Reduced prerequisites for test and development

The prerequisites for WebSphere Message Broker have been reduced to simplify the install experience and also to reduce the time required to start working with the product.

The Configuration Manager now no longer requires DB2 Universal Database, whereas at WebSphere MQ Integrator V2.1 the Configuration Manager required DB2 Universal Database as a repository for the message repository manager and the configuration data. Message definition data is now stored in the Message Brokers Toolkit workspace and the broker domain configuration information is now stored in an internal repository in the Configuration Manager.

On Windows, the broker is able to utilize the embedded Derby database for its repository database, requiring only the Open Database Connectivity (ODBC) drivers for Cloudscape™ that is supplied with the product. Use of the embedded Derby database is supported only for test and development, but if used, it means that, for these environments, DB2 Universal Database does not need to be installed at all on Windows.

3.1.5 WebSphere Message Broker Welcome Page

The Welcome Page is new for WebSphere Message Broker V6.0. It provides a useful starting point for new and experienced users to find out more about WebSphere Message Broker. It also provides links to the Quick Tour, the Default Configuration wizard, and the Samples Gallery. These are designed to help you get started with the product by giving a product overview, creating a simple broker configuration, and trying out new product functions. Links in the documentation and useful external Web sites are provided to help you track down useful information quickly.

The Welcome Page is displayed in the Message Brokers Toolkit the first time it is opened. When it is closed, the Broker Application Development perspective in the Message Brokers Toolkit is displayed. The Welcome page can be reopened at any time from the Help menu.

3.1.6 Default Configuration wizard

The Default Configuration wizard is a wizard in the Message Brokers Toolkit that creates a simple configuration that is suitable to verify the installation of WebSphere Message Broker and to run the WebSphere Message Broker samples. The wizard accepts a username and a password and then creates a simple broker domain consisting of a Configuration Manager called WBRK6_DEFAULT_CONFIGURATION_MANAGER and a broker called WBRK6_DEFAULT_BROKER. It also creates the database, queue manager, listener, and domain connection required for these components to view the broker domain in the Message Brokers Toolkit.

3.1.7 Samples Gallery

The Samples Gallery contains a selection of sample resources and instructions that demonstrate how to use WebSphere Message Broker, including new functionality since V2.1. The resources, in the form of message flows and message sets, are imported into the Message Brokers Toolkit for display and manipulation.

The Sample Preparation wizard, which sets up any resources required for the sample, such as database tables and WebSphere MQ queues, deploys samples as well. The samples can also be automatically removed from the environment once you have finished with them.

3.2 Migration and coexistence

A number of features have been introduced to enable a smooth migration from WebSphere MQ Integrator V2.1 and other previous versions of WebSphere Message Broker to the present one.

The following products can migrate to WebSphere Message Broker V6.0:

- WebSphere MQ Publish\Subscribe Broker
- WebSphere MQ Integrator V2.1
- WebSphere MQ Integrator Broker V2.1
- WebSphere MQ Event Broker V2.1

The features that allow migration from these versions to WebSphere Message Broker V6.0 (and WebSphere Event Broker V6.0) are discussed next.

3.2.1 Coexistence

In order to make migration simpler, you can install WebSphere Message Broker V6.0 alongside WebSphere MQ Integrator V2.1. Coexistence also extends to the broker domain components. It is possible to view and administer the WebSphere MQ Integrator brokers from the Message Brokers Toolkit, which replaces the Control Center.

User profiles and the Command Console

Because multiple versions of the product can now exist on a single machine, a profile needs to be created to select the instance that you want to work with. This initializes the environment so that the correct commands can be invoked from the correct installation of the code.

On Windows, typing a command such as **`mqsilist`** on the command line invokes the WebSphere MQ Integrator V2.1 command. In order to run the command against the WebSphere Message Broker V6.0 of the product and any V6.0 components, you have to do it using the Command Console. This is available via **Start → Programs → IBM WebSphere Message Brokers 6.0 → Command Console**. It is also possible to create a customized command profile for Windows using the sample profile, which is found in the WebSphere Message Broker Information Center.

On Linux and Unix platforms, an environment profile is provided in the bin directory of the install location that can be run to initialize the environment for the selected installed instance.

Domain coexistence

Another feature intended to make migration easier by doing it in stages is the coexistence of components. For WebSphere MQ Integrator only the broker can coexist with WebSphere Message Broker V6.0 components. This is because the Control Center and the WebSphere MQ Integrator Configuration Manager cannot administer WebSphere Message Broker V6.0 components. The Control Center, Configuration Manager, and User Name Server need to migrate to Version 6.0, but the broker can remain at Version 2.1 and still be part of a Version 6.0 domain.

It is possible to maintain WebSphere MQ Integrator V2.1 brokers and to administer them and deploy to them from the Message Brokers Toolkit and the WebSphere Message Broker V6.0 Configuration Manager. The brokers must be from the following products and at least at the stated service levels:

- WebSphere MQ Integrator Broker V2.1 at fix pack six or later
- WebSphere MQ Integrator V2.1 at fix pack six or later

3.2.2 Commands for migration

A single command is provided to migrate individual components to Version 6.0. The command migrates queues, databases, registry keys, and files where appropriate, allowing for a step-by-step approach to migration. For example, for WebSphere MQ Integrator V2.1 the Configuration Manager and User Name Server can migrate before the brokers.

More about the features associated with migration and the individual steps are covered in Part 2, “Migrating from WebSphere MQ Integrator V2.1” on page 105.

3.3 Message Brokers Toolkit

The most obvious difference between V2.1 and WebSphere Message Broker V6.0 is the replacement of the Control Center with the Message Brokers Toolkit. The Message Brokers Toolkit is built on the Rational® Application Development Platform. This is an Eclipse-based Unified Modeling Language (UML) visualization and Java Development tool that has been extended with tools specifically for WebSphere Message Broker, including:

- Message flow development
- Message set development
- User-defined node development
- Broker domain administration
- Flow debugger

3.3.1 Key differences with the Message Brokers Toolkit

There are a number of key differences between the Control Center and the Message Brokers Toolkit that are important for Version 2.1 users to understand. The Message Brokers Toolkit is available on Linux as well as Windows. This, together with the Configuration Manager being available on all platforms, means that Windows is not a required platform.

Checking in and checking out

Resources in the Message Brokers Toolkit are stored in the local file system and are no longer stored within a repository in the Configuration Manager. This means that there is no need to check files in or out of a repository. The Configuration Manager no longer has a repository for the storage of broker application resources. Also, there is no need to connect to a Configuration Manager in order to use the Message Brokers Toolkit to develop message flows and message sets, because connection is required only for deployment or debugging.

The default location on Windows is C:\Documents and Settings*user*\IBM\wmbt6.0\workspace, and on Linux the default location for the workspace is /home/*user*/IBM/wmqi6.0/workspace. Broker application resources and other files are stored in the local file system in a directory called the workspace, chosen when the Message Brokers Toolkit is started up, as shown in Figure 3-1.

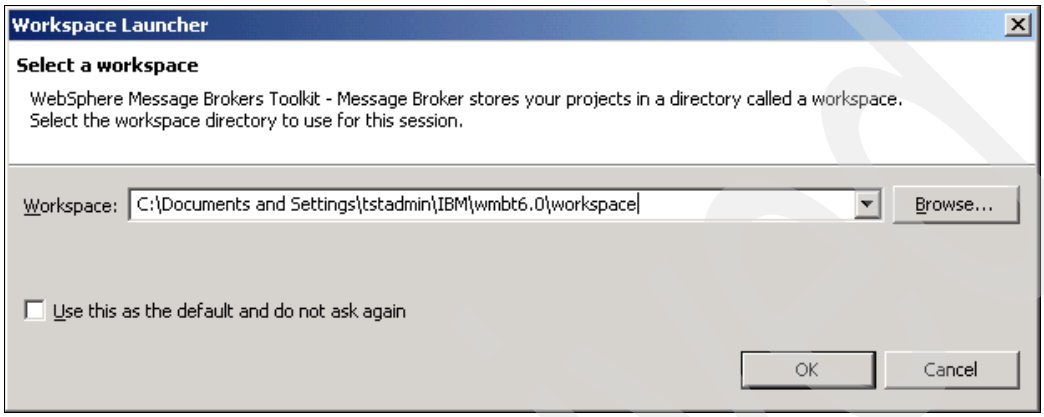


Figure 3-1 Selecting the location to store resources

Resources created with the Message Brokers Toolkit are saved to the workspace directory and are displayed in the Resource Navigator. It is also possible to Import resources from other locations into the workspace using the Message Brokers Toolkit, which creates a reference to the resources within the tooling, making them visible through the Resource Navigator.

Broker application resources are stored in projects in the Message Brokers Toolkit workspace. Message flows are stored in message flow projects and message sets are stored in message set projects. Projects have associations with the Message Brokers Toolkit that are involved with verification, building, version management, sharing, and resource organization.

Resources can be shared between users and systems either by moving resources between machines or by setting up supported development repositories and version control systems such as CVS, IBM Rational ClearCase®, or any other repository with an Eclipse 3.0 client. Further, information about repositories on Eclipse can be found at the following Web address:

`http://www.eclipse.org/community/team.html`

Broker Application Development

The Broker Application Development perspective contains editors for the creation of message flows and message sets and can be performed in the Message Brokers Toolkit without a connection to a Configuration Manager. In order to test the developed message sets and message flows, a Configuration Manager with at least one defined broker is required as before.

Message flow editor

Message flow development in WebSphere Message Broker V6.0 is very similar to how it was in V2.1, with a selection of built-in nodes (IBM Primitives) in a palette and a canvas for constructing the flow and connecting nodes. Properties for the nodes can be edited by right-clicking and selecting **Properties**.

Extended Structured Query Language (ESQL) code and mappings no longer reside within the node properties for editing but in a separate .esql file that opens in its own specific editor. This makes the process of coding ESQL or defining mappings much easier. It also enables reuse of code between nodes. Because ESQL is not the only way to code transformation of messages, some of these types of nodes are associated with either mapping, or Java, and open the appropriate editor for this function.

ESQL editor

The ESQL editor is specifically designed for the editing of ESQL code, thereby simplifying coding in ESQL. Each message flow uses a single ESQL file containing one module per node in the flow. The code in these modules can then be reused in other nodes or message flows.

Content assistance is provided to help with coding ESQL files. Pressing the Control and Space key at the same time in the ESQL editor or selecting **Content Assist** from the Edit menu helps access this assistance. The results of the content assist depend upon the context when it was invoked.

Content assistance is provided for ESQL statements and functions, but even more for helping in determining the structure of input and output messages where the message flow is associated with a message set.

When an ESQL file is saved, it is validated with any errors found displayed in the Problems view. A message flow with errors in the ESQL cannot be deployed. This helps to minimize failed deploys, because many problems are resolved before a deploy to the broker can be attempted.

The following nodes produce ESQL:

- Compute node
- Filter node
- Database node

Mapping editor

In WebSphere Message Broker, mappings are created in a separate editor and are generated for different nodes to ESQL. A map contains a source and a target that can either be a message as defined in a message set or a database table that can be defined to the Message Brokers Toolkit. This is an advantage over V2.1, because the definitions in the map are generated from existing objects.

To create a mapping, fields from the source are dragged to the target. For message flows, using message definitions in particular, this enables the creation of transformations very rapidly.

It is also possible to add logic to the mappings, for example to perform conditional statements, handle repeating elements, and make choices based on the content of a message or database. A sample mapping with `for` statements between two messages is shown in Figure 3-2.

Submaps can be created to provide reusable mapping function and to allow multiple input elements. Maps can also call existing ESQL functions, as well as be invoked from ESQL themselves.

The types of sources and targets that a mapping can contain depend upon the type of node that the map is created from. The following is a list of nodes that use maps:

- Mapping node
- DataInsert node
- DataUpdate node
- DataDelete node
- Warehouse node

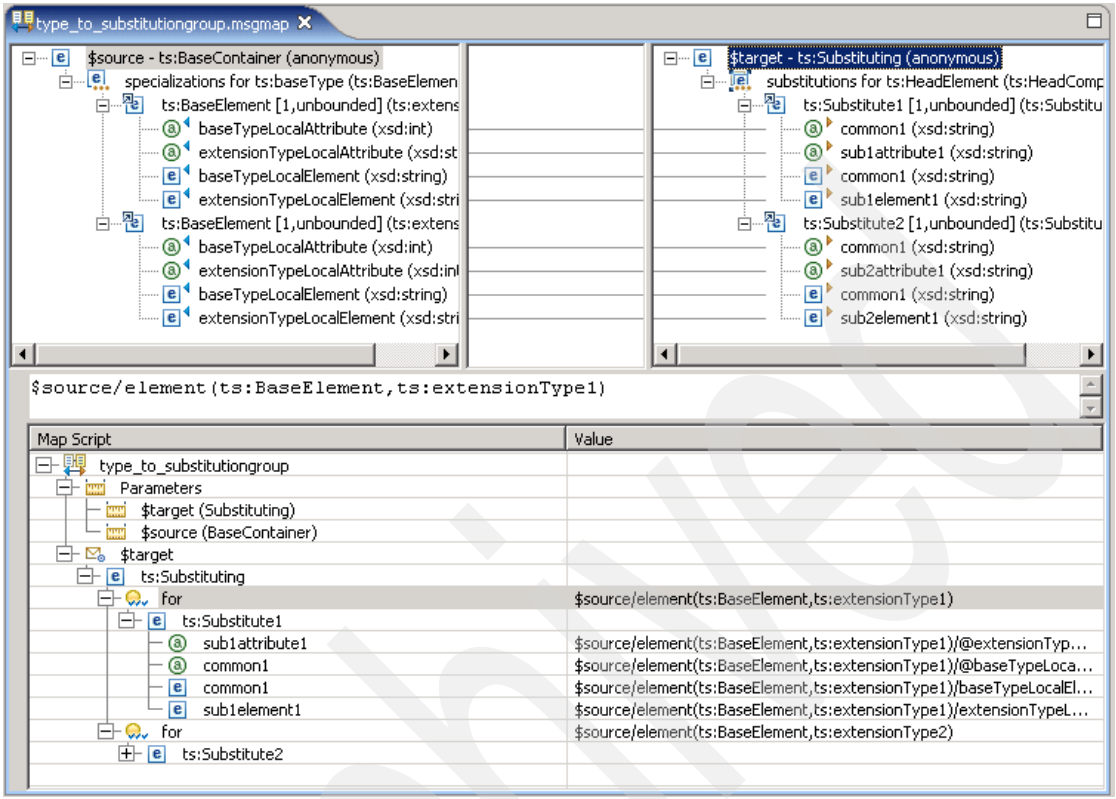


Figure 3-2 A sample message map

Message set and message definition editors

Message sets are contained in message set projects in the Message Brokers Toolkit with a separate file for a message set and individual message definitions. The usability of message modeling created with the Message Brokers Toolkit is much improved from the Control Center.

In the message set editor, properties are set for the physical and logical formats of the message set, such as namespace settings and datetime configuration.

The message definition editor is where messages are constructed from elements, attributes, groups, and complex types.

A wizard is provided to create message definitions from existing C, COBOL, Web Services Description Language (WSDL), Extensible Markup Language (XML) schema, and Extensible Markup Language Document Type Definition (XML DTD) files. Some message definitions are included with WebSphere Message

Broker V6.0, including Simple Object Access Protocol (SOAP), multipart Multipurpose Internet Mail Extensions (MIME), and SAP IDoc messages. In addition, XML schema and WSDL definitions can be created from message definitions created in the Message Brokers Toolkit.

The message set project can be referenced by message flow projects. The referenced message definitions can then be used to resolve a message structure in ESQL or mappings.

Java perspective

The Java perspective is used when coding Java for its use with the JavaCompute node. Any valid Java code can be added to a JavaCompute node, to make full use of the existing Java user-defined node in the application programing interface (API) to process an incoming message.

The JavaCompute node uses the same API as the user defined nodes (Java plug-in nodes and parsers) that includes extra methods to simplify code, involving message routing and transformation, such as accessing named elements in a message tree, setting their values, and creating elements without the need to navigate the tree explicitly.

A query language called XPath can also be used in the Java code for navigating the structure of XML documents. Using XPath makes coding for XML messages simpler, because referring to the message structure is more precise, takes away the need for manual navigation, and reduces the number of statements in the code. XPath can query any tree structure, so it is not restricted to XML messages.

The Java files are verified on saving, and any errors are displayed in the file and also in the Problems view in the Broker Application Development perspective. Interactive assistance with errors is provided along with the other Java editing facilities including code completion, integrated Javadoc™ documentation, and automatic compilation.

Data tools

A Data perspective is provided in the Message Brokers Toolkit as part of the Rational Application Developer Platform. The tools that are provided are useful in message flow development where the message flows interact with databases. The tool to create a Relational Data Base (RDB) Definition File enables a database to be accessed and definitions of its tables to be added to a message flow project in the workspace. After a definition for a database is added to a message flow project, it is included in mappings to assist with validation for ESQL.

Broker Administration

The Broker Administration perspective in the Message Brokers Toolkit replaces the Assignments and Operations view in the Control Center. Deployment of resources such as message flows and message sets is performed using this perspective. The resources are compiled in a broker archive file and then deployed directly to the required execution group on a broker. The operational status of the broker is displayed in the Broker Administration perspectives Domains view, including what is deployed to each execution group and the components (brokers, execution groups, message flows) that are started and stopped.

In addition to deployment and component status, the following tools are also available in the Broker Administration perspective:

- Broker Topology configuration - including collectives
- Topics - topic definition and security settings
- Subscriptions - filter subscriptions by topic
- Event Log - deployment results from the Configuration Manager
- Start, Stop, Cancel Deployment, Redeploy operations

Multiple broker domains can be administered in the Broker Administration perspective at any time. Also, multiple Message Brokers Toolkit can administer the same Broker domain.

Plug-in nodes and parsers

In WebSphere MQ Integrator V2.1 support was introduced to enable users to develop plug-in nodes and parsers in Java as well as C. In WebSphere Message Broker V6.0 these Java plug-in nodes and parsers are created using the Java perspective, using the same API as used for the JavaCompute node.

In WebSphere Message Broker these nodes are no longer called plug-in nodes and parsers. They are referred to as *user-defined nodes*.

User-defined nodes cannot be developed in WebSphere Event Broker, but WebSphere Event Broker users can deploy nodes the third parties provide.

Flow debugging

Flow debugging was enhanced after WebSphere MQ Integrator V2.1 to not only allow the setting of breakpoints between nodes in a message flow, but also to step through the code in the message flow. This means that when a message enters a node with ESQL, Java, or mapping code, the debugger provides the option to step through individual lines or actions in the code.

The message is visible as it travels through the flow and changes are made to it. Headers and other properties of the message are also visible. The Environment tree and the Exceptionlist tree are also visible. In order to check certain conditions in the message flow, values in the message can be manipulated in-flight through the debugger.

The debugger can attach to a running execution group without a need to redeploy a debug version of the code.

3.4 New nodes

A variety of new nodes have been provided to enhance the functionality of WebSphere Message Broker. Table 3-1 contains a list of new nodes after WebSphere MQ Integrator V2.1. A brief description of the functionality of each node is shown.

Table 3-1 New nodes since WebSphere MQ Integrator V2.1

Node name	Description
MQGet	Can be used anywhere in a flow to retrieve a message from a WebSphere MQ queue.
MQOptimized node	Is used to publish a persistent JMS message to a single subscriber to improve performance over ordinary publish/subscribe message flows.
HTTPInput	Used to receive a Web Services request. Must be used in conjunction with a HTTPReply node.
HTTPReply	Returns a response from a message flow to a Web Services client.
HTTPRequest	This node constructs a Web Service request using the contents of an input message. The response is parsed for inclusion in the output tree.
JMSInput	Receives messages from JMS destinations accessed through a connection to a JMS provider.
JMSOutput	Sends messages to JMS destinations using the Java Message Service Specification.
JMS MQTransform	Transforms a message with a JMS message tree into a message compatible with a WebSphere MQ JMS provider.

Node name	Description
MQ JMSTranform	Receives messages that have a WebSphere MQ JMS provider message tree format, and transforms them into messages for JMS destinations.
Real-time Input	Receives messages from clients using WebSphere MQ Real-time Transport or WebSphere MQ Multicast Transport and JMS applications.
Real-time Optimized Flow	For high performance publish/subscribe using WebSphere MQ Real-time Transport or WebSphere MQ Multicast Transport and JMS.
JavaCompute	Used for transforming messages using a Java API.
Mapping	Used for message transformation and database interactions using mappings.
TimeoutControl	Used in conjunction with a TimeoutNotification node. This node receives an input message that contains a timeout request.
TimeoutNotification	Processes timeout request messages that are set by their associated TimeoutControl node and propagates copies or parts of the message to the next node in the message flow.
Validate	This node checks whether a message matches a given message structure definition. If it does, it is output to the match terminal, else it is propagated to the failure terminal.
Passthrough	Used to provide a label to indicate the subflow it is contained in. Messages are not processed, only passed through this node.
XMLTransformation	Transforms an XML message to another form of XML message, according to the rules provided by an Extensible Stylesheet Language (XSL) style sheet.

Changes have also been made to some of the existing nodes, for example, to separate ESQL and mappings. In addition, the aggregation nodes now use WebSphere MQ queues, instead of a database, to store state information. This improves the throughput of all requests, in particular the performance of non-persistent requests.

3.5 Enhanced message modeling

This section discusses Web Services support, XML Schema, Namespaces, Parser Domains, and runtime validation.

3.5.1 Web Services support

Support is provided for modelling and working with SOAP messages for Web Services. Full support for the construction and parsing of SOAP messages, and the addition of XML namespace support through the extension of the MRM XML Wire Format and the addition of the XML Namespace (NS) domain is provided. Sample message definitions are provided for SOAP 1.1 and 1.2. Existing Web Services Description Language (WSDL) definition of a variety of styles can be imported into the Message Brokers Toolkit.

WSDL definitions generated from a message set can be used by external applications such as . NET. The WSDL generator also leaves a message set SOAP-ready for use by the MRM XML parser.

A new MIME parser domain provides support for SOAP with attachments.

3.5.2 XML Schema

The WebSphere Message Broker uses the logical message model that is based on the industry standard XML Schema 1.0. Message definition files use XML Schema to represent the structure of the messages. Each message definition file describes the logical structure of the messages, and the physical format of the message bit stream during transmission.

3.5.3 Namespaces

XML Schemas can enable the definition of a target namespace that allows elements and attributes to share the same name within the same XML instance. Namespaces assist in the development of a library of XML Schemas, which can be developed independently. Using a unique namespace name means that developers do not need to be concerned about name clashes with objects defined within other XML Schemas.

The target namespace, if defined, qualifies the global elements, attributes, groups, and types within an XML Schema. Optionally, the target namespace can also qualify local elements and attributes.

The scope of a namespace extends beyond that of its containing document and is identified by a Uniform Resource Identifier (URI), which should be unique.

Elements from multiple namespaces can be used within the same message definition.

3.5.4 Parser domains

The following new domains have been introduced since WebSphere MQ Integrator V2.1:

- XMLNS
- XMLNSC
- JMSMap
- JMSStream
- MIME

More information about the new domains can be found in the WebSphere Message Broker Information Center at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp

3.5.5 Runtime validation

A number of enhancements have been made to improve the validation of messages flowing through, as follows:

- Validation options available on more nodes.
- A new Validate node to validate a message in the middle of a message flow.
- The ability to detect all validation failures in a message before throwing an exception.
- The ability to force a complete parse of a message independently of runtime validation.

3.6 Configuration and administration

There have been a number of changes and enhancements to configuration and administration after WebSphere MQ Integrator V2.1. There are a number of changes to the Configuration Manager itself and there is also the Message Brokers Toolkit that replaces the Control Center, as mentioned previously. There are also a variety of new and improved commands and utilities that are briefly described here.

3.6.1 Configuration Manager changes

As mentioned previously, the Configuration Manager has undergone a number of changes to separate it more from the tooling, and to reduce its prerequisites. These changes are briefly discussed here, and Chapter 11, “Migrating the Configuration Manager” on page 177 also provides some information about changes that are important after migration.

Configuration Manager databases

The Configuration Manager no longer requires an external database to remove the requirement for DB2 Universal Database for use with the Configuration Manager, because it uses an internal repository. The MRM database is no longer required because the message set data is now stored in the local workspace or in the version control repository.

Multiple Configuration Managers on a single machine

Specifying a name on the command line can now create multiple Configuration Managers on a single machine, for example:

```
mqsicreateconfigmgr CONFIG_MGR1 -i userid -a -q CONFIG_QM
```

There is still only one Configuration Manager per domain, but multiple domains can be administered from the Message Brokers Toolkit.

Configuration Manager available on all platforms

The Configuration Manager can now be created on all of the supported broker platforms including AIX, HP-UX, Linux (x86 platform), Linux (zSeries platform), Solaris™, Windows, and z/OS.

3.6.2 Configuration Manager Proxy API

The Configuration Manager Proxy (CMP) is a Java API to the Configuration Manager. It enables programs to be written that automatically create and administer broker domains. Java objects map to domain objects to make the Configuration Manager Proxy API simpler to code. A sample API Exerciser is supplied to demonstrate the capabilities and how to program an application to utilize them.

The Configuration Manager Proxy API can be used for the following tasks:

- Viewing domain objects
- Adding and removing brokers
- Modifying broker properties
- Setting up Access Control Lists

- Backing up and restoring domains
- Editing and deploying topology
- Editing and deploying topics
- Querying and deleting active subscriptions
- Creating and deleting execution groups
- Deploying bar files
- Viewing deployed resources
- Querying logs
- Starting and stopping message flows
- Controlling user traces

The available administration functions are comparable to those found within the Message Brokers Toolkit and the command line, and can be seen to be a powerful tool for managing the administration of the domain if desired.

3.6.3 Versioning of deployed resources

In WebSphere Message Broker, resources such as message flows and message sets can have version information included in the files, such as when the resources were compiled, when they were deployed, and the name of the broker archive file that they were deployed with. This can be used to assist with version control and aids in returning to a working version of a message set or message flow if the latest version is discovered to have a problem or defect.

In addition to built-in version information, you can specify your own unique version numbers and also your own keyword, such as author, for enhanced identification of deployed objects. This information can be added to message flows, message sets, ESQL files, and Java files.

This information can then be easily viewed through the Broker Administration perspective in the Message Brokers Toolkit.

3.6.4 Statistics and accounting

Function has been added to WebSphere Message Broker after V2.1 to enable the collection of statistics on the behavior of message flows in order to monitor the performance of these flows. Message flow accounting and statistics data records dynamic information about the runtime behavior of a message flow, such as the number of messages processed, the size of the messages, CPU usage, and elapsed message processing times.

Options are provided for the timing of collection of statistics:

- Snapshot data is collected for an interval of approximately 20 seconds.
- Archive data is collected for a user-defined length of time between 10 and 14400 minutes.

The commands used for statistics collection are:

- `mqsichangeflowstats`
- `mqsireportflowstats`

3.6.5 New command line utilities

This section discusses the new command line utilities.

Creating and deleting execution groups

Commands now exist to create and delete execution groups on the command line, in addition to the Message Brokers Toolkit. The new commands to enable this are:

- `mqsicreateexecutiongroup`
- `mqsideleteexecutiongroup`

Stopping and starting message flows

Commands now exist to start and stop message flows on the command line, in addition to the Message Brokers Toolkit. The new commands to enable this are:

- `mqsistartmsgflow`
- `mqsistopmsgflow`

Command line deploy

This utility is available on all broker platforms. It is an alternative method to deploying broker archive files in the Message Brokers Toolkit. It can assist in the automation of deploy operations. Other functions include the ability to cancel deployment on a broker and remove deployed message flows and message sets from execution groups. The new command to enable this is:

`mqsideploy`

Backing up and restoring domain data

A command now exists to save a copy of the Configuration Manager's repository, with a name provided as a parameter, to a specific directory. A command is also provided that replaces the entire Configuration Manager's repository with a previously backed up version. The new commands to enable this are:

- `mqsibackupconfigmgr`

- mqsirestoreconfigmgr

Creating and deleting broker databases on Windows

Commands are provided on Windows to create a broker database on the command line and also to delete a database created this way. The command creates a named database and sets up an ODBC data source name. The command only supports Derby and DB2 Universal Database. If Derby is used as the database then a Windows service is created and started for that database. The new commands to enable this are:

- mqsicreatedb
- mqsideletedb

Other new commands

Other new commands added after WebSphere MQ Integrator V2.1 are not discussed in any further detail here, but information about them can be found in the Reference section of the WebSphere Message Broker Information Center. These new commands are:

- mqsicbrreport
- mqsichangedbimgr
- mqsi_setupdatabase
- mqsichangeproperties
- mqsicreatebar
- mqsicreatemsgdefs
- mqsicreatemsgdefsfromwsdl
- mqsireadbar
- mqsimigratemsgflows
- mqsimigratemsgmets

3.6.6 Other administration updates

This section discusses other administration updates.

Password prompt

To improve the security of using command line administration utilities there is now a method of entering commands without the password being displayed on the screen. Enter a command such as **mqsicreatebroker** without including a password, such as the following:

```
mqsicreatebroker BRK2 -i userid -a -q BRK2_QM -n BRK2DB
```

Note: The -a flag has no password. A prompt appears asking to enter a password. After entering the password, nothing is displayed while you type. On clicking Enter, you are prompted to retype the password for verification. If the two passwords match, the command continues to be processed.

Commands can be typed with a password after the -a flag if desired.

3.7 Security

There have been a number of changes to security after WebSphere MQ Integrator V2.1 that are important for using and migrating to WebSphere Message Broker V6.0.

3.7.1 Access control lists

Access control lists replace the Group authority used at WebSphere MQ Integrator V2.1. The Configuration Manager maintains the access control list entries for the various runtime objects. These entries contain details of which users and groups have permission to view and modify the various domain objects. The following are runtime objects that have access control list entries:

- Configuration Manager
- Topology
- Subscriptions
- Root topic
- Brokers
- Execution Groups

Access control lists are granular, such that if an access control list is set up that way, it is possible for a user to deploy to one execution group and not another even though they share a broker. Previously, using the group security model, a user could either deploy or not deploy, but there was no control of specific objects. The permission levels that can be set are:

- Full
- View
- Deploy (broker, execution group, root topic)
- Edit (root topic)

Access control lists can be viewed, created, and deleted with these command line tools:

- `mqsilistaclentry`
- `mqsicreateaclentry`

- `mqsideleteaclentry`

3.7.2 Configuration Manager and Message Brokers Toolkit security

Unlike the Control Center, no specific authority is required to launch the Message Brokers Toolkit. However, when creating and connecting a domain to connect the Message Brokers Toolkit to the Configuration Manager, the user running it must have WebSphere MQ authorities. You can also create Security Exits and a Secure Sockets Layer (SSL) for the connection between the Message Brokers Toolkit and the Configuration Manager to develop a secure environment for deploys.

No security is provided in the WebSphere Message Broker, because resources are stored on the local file system in the Message Brokers Toolkit workspace. The operating system security settings or the use of a version control repository for the storage of file resources can be used to protect the files.

3.7.3 Other security enhancements

Some of the other security enhancements since WebSphere MQ Integrator V2.1 are discussed briefly here.

Tunneling

You can set up WebSphere Message Broker to use Hypertext Transfer Protocol (HTTP) Tunneling and Connection via Proxy to enable connection between clients and brokers when they exist on different networks with a firewall between them.

Quality of Protection

To protect messages from tampering and eavesdropping, use cryptographically based protection of messages. Message protection is configurable on a per topic basis to allow performance to stay high for topics that are less sensitive. Quality of Protection (QoP) is set on the properties for a broker and the default value is `none`. The following values can be selected for Quality of Protection:

- Channel Integrity - Messages cannot be added or deleted without detection.
- Message Integrity - The content of the message cannot be changed without detection.
- Encrypted for Privacy - The content of the message cannot be viewed.

Domain awareness

Domain awareness means that the Configuration Manager can retrieve information about domains other than the local domain. This allows users from

trusted domains, if they have the appropriate permissions on objects, to access the Configuration Manager using the Message Brokers Toolkit to perform administrative and deployment actions.

3.8 Other enhancements

WebSphere Message Broker has seen numerous other enhancements and new features after WebSphere MQ Integrator V2.1, and it is beyond the scope of this redbook to mention all of them. This section groups together a few of the remaining enhancements, but information about the other enhancements can be found in the WebSphere Message Broker Information Center in the What is New topics in the Introduction at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp

3.8.1 Rational Agent Controller

The Rational Agent Controller (RAC) is used with the flow debugger for WebSphere Message Broker. Installation of this application is a must on systems where brokers are to be debugged. If you are debugging a broker on the Message Brokers Toolkit machine then RAC must be installed on that system, or on the remote system if the broker is remote.

RAC also requires TCP/IP connectivity to the system that the Message Brokers Toolkit is running on, to allow it to communicate with the RAC client that is embedded in the Message Brokers Toolkit.

This is an optional piece of software that is only required for debugging using the flow debugger in the Message Brokers Toolkit. Instructions for installing this product can be found in the WebSphere Message Broker Information Center.

If other products that use tooling built upon the Rational Application Development platform exist on a machine, these may also use RAC to perform tasks such as debugging. Only one installation of RAC can be on a machine at any one time.

3.8.2 Information center replacing hardcopy documentation

WebSphere Message Broker V6.0 does not come with any hardcopy manuals except for the Installation Guide. The product documentation is now available in the product Information Center and Eclipse-based help system, in common with the majority of recently released IBM products. Access this from the Message Brokers Toolkit Help menu, or for Windows directly from the Start menu.

The Information Center is also available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?

Information is provided on the first page of the Information Center to give guidance on how to use the information, what information is available, and how to navigate in the Information Center. It is useful to spend some time using this page, as it can help give hints to the best ways to use the Information Center for retrieving information most useful in a given situation.

The advantage of having information in this form is that it is easily searchable, multiple users can access it, and it reduces the cost to the environment. Individual topics can be printed as required.

The Information Center can include links to external Web sites that are updated to contain the latest information.

Information is no longer structured as books, but collections of topics downloadable in PDF format from the Information Center:

ftp://ftp.software.ibm.com/software/integration/wbibrokers/docs/V6.0/

3.8.3 Enhancements to ESQL

There are many updates and enhancements to ESQL in WebSphere Message Broker that are not listed here. The Reference section of the WebSphere Message Broker iInformation center contains Information about ESQL.

Some examples of the enhancements for transforming and routing messages using ESQL are as follows:

- In-memory cache to reduce access to databases for read-only routing or validation data
- Improved support for creating DATETIME variables
- Access to multiple databases from the same Compute, JavaCompute, Database, or Filter node
- Dynamic database schemas
- New user-defined message flow properties that can be passed as parameters to ESQL to modify standard behavior
- Access to environment information, for example message flow name and broker name
- Improved support for result sets returned by database stored procedures
- Improved error recovery using SQL handlers
- Multiple out terminals to combine the function of Compute and Filter nodes

- Improved CAST error handling

3.9 List of enhancements and changes

Table 3-2 notes the majority of the enhancements in WebSphere Message Broker since WebSphere MQ Integrator V2.1.

Table 3-2 Major enhancements in the product since WebSphere MQ Integrator V2.1

Feature	Release introduced
Message Brokers Toolkit	WebSphere Business Integration Message Broker V5.0
Install LaunchPad	WebSphere Business Integration Message Broker V5.0(updated for WebSphere Message Broker V6.0)
WebSphere Message Broker Quick Tour	WebSphere Business Integration Message Broker V5.0(updated for WebSphere Message Broker V6.0)
WebSphere Message Broker samples	WebSphere Business Integration Message Broker V5.0 (extended and updated for WebSphere Message Broker V6.0)
Web Services support	WebSphere Business Integration Message Broker V5.0
Logical message model based on XML Schema 1.0	WebSphere Business Integration Message Broker V5.0
Support for XML namespaces	WebSphere Business Integration Message Broker V5.0
User defined simple types	WebSphere Business Integration Message Broker V5.0
Extended database user ID and password support	WebSphere Business Integration Message Broker V5.0
Message flow statistics and accounting	WebSphere Business Integration Message Broker V5.0
Multicast publish/subscribe	WebSphere Business Integration Message Broker V5.0
SSL authentication	WebSphere Business Integration Message Broker V5.0

Feature	Release introduced
WebSphere MQ Telemetry Transport	WebSphere Business Integration Message Broker V5.0
HTTP tunneling	WebSphere Business Integration Message Broker V5.0
Object level security	WebSphere Business Integration Message Broker V5.0
Linux on Intel	WebSphere Business Integration Message Broker V5.0
Linux on zSeries®	WebSphere Business Integration Message Broker V5.0
Graphical mapping	WebSphere Business Integration Message Broker V5.0(enhanced for WebSphere Message Broker V6.0)
Command line administration	WebSphere Business Integration Message Broker V5.0(extended for WebSphere Message Broker V6.0)
TLOG messaging standard	WebSphere Business Integration Message Broker V5.0 fix pack 4
Invoke Java from ESQL	WebSphere Business Integration Message Broker V5.0 fix pack 4
HL7 messaging standard	WebSphere Business Integration Message Broker V5.0 fix pack 5
Configuration Manager no longer needs a database	WebSphere Message Broker V6.0
Hard copy installation guide	Removed at WebSphere Business Integration Message Broker V5.0, reinstated for WebSphere Message Broker V6.0
Default Configuration wizard	WebSphere Message Broker V6.0
Coexistence with previous versions	WebSphere Message Broker V6.0
Migration of individual components	WebSphere Message Broker V6.0
Simplification of z/OS install	WebSphere Message Broker V6.0
Message Brokers Toolkit for Linux (x86 platform)	WebSphere Message Broker V6.0

Feature	Release introduced
Routing and transformation using Java	WebSphere Message Broker V6.0
Access to multiple databases from Compute, JavaCompute, Database, and Filter node	WebSphere Message Broker V6.0
Dynamic database schemas	WebSphere Message Broker V6.0
Access to environment information	WebSphere Message Broker V6.0
Multiple out terminals for Compute	WebSphere Message Broker V6.0
XSLT performance	WebSphere Message Broker V6.0
Broker performance	WebSphere Message Broker V6.0
Aggregation performance	WebSphere Message Broker V6.0
SOAP 1.2 supported	WebSphere Message Broker V6.0
HTTP 1.1 supported	WebSphere Message Broker V6.0
SOAP schemas provided	WebSphere Message Broker V6.0
WSDL importer	WebSphere Message Broker V6.0
HTTPS support	WebSphere Message Broker V6.0
JMS Transport support	WebSphere Message Broker V6.0
TimeoutControl & TimeoutNotification nodes	WebSphere Message Broker V6.0
MQ Get node	WebSphere Message Broker V6.0
Message parsing performance	WebSphere Message Broker V6.0
MIME parser	WebSphere Message Broker V6.0
Validation improvements	WebSphere Message Broker V6.0
Validate node	WebSphere Message Broker V6.0
Runtime Versioning	WebSphere Message Broker V6.0
Configuration Manager Proxy	WebSphere Message Broker V6.0
Configuration Manager available on all broker platforms	WebSphere Message Broker V6.0
Multiple Configuration Managers on a single machine	WebSphere Message Broker V6.0

Feature	Release introduced
Code page conversion updated	WebSphere Message Broker V6.0

4

New features in WebSphere Business Integration Message Broker V5.0

This chapter describes the new features of WebSphere Message Broker since the release of WebSphere Business Integration Message Broker V5.0 on a function-by-function basis.

It provides a listing of key changes, including some additions from the WebSphere Business Integration Message Broker V5.0 fix packs.

4.1 Installation and initial user experience

A number of enhancements have been made to the installation and initial user experience for WebSphere Message Broker. Many of these are specific to Windows, but there are enhancements that improve the experience for other platforms as well.

4.1.1 Installation

In order to run WebSphere Message Broker, you must install the prerequisite software. The installation wizards no longer enforce that the prerequisite software be installed before WebSphere Message Broker, but you should install it before using the product.

The installation of WebSphere Message Broker is implemented using InstallShield for Multi-platforms (excluding z/OS) as with WebSphere Business Integration Message Broker V5.0, but now the installation of the WebSphere Message Broker runtime and the Message Brokers Toolkit is separated, and they need to be installed individually.

4.1.2 Preinstalled LaunchPad

Windows provides a preinstalled LaunchPad to assist with the installation of prerequisite software, as well as the WebSphere Message Broker runtime and Message Brokers Toolkit. This LaunchPad provides two methods for performing a WebSphere Message Broker installation:

- Express installation
- Advanced installation

The Express installation performs a simple installation of WebSphere Message Broker, the Message Brokers Toolkit, and the prerequisite products. This installation is suitable for test and development purposes, and enables quick installation of the product. The Express installation chooses default settings and requires minimum intervention by the user.

The Advanced installation option is designed for more experienced users who want to manually install and choose settings for individual prerequisites as well as for WebSphere Message Broker and Message Brokers Toolkit.

4.1.3 Supplied on DVD

WebSphere Message Broker is supplied on DVD for Windows and Linux on Intel. It reduces the number of CD changes required, particularly for the Message Brokers Toolkit, which is supplied on four CDs for each platform. For users

without a DVD player, WebSphere Message Broker for the other distributed platforms is supplied on CD, as are Windows and Linux (x86 platform).

Existing users can also obtain WebSphere Message Broker in electronic form.

4.1.4 Reduced prerequisites for test and development

The prerequisites for WebSphere Message Broker have been reduced to simplify the installation experience and to reduce the time to get the product running.

The Configuration Manager now no longer requires DB2 Universal Database, whereas at WebSphere Business Integration Message Broker V5.0, the Configuration Manager required DB2 Universal Database as a repository for the configuration data. An internal repository in the Configuration Manager now stores broker domain configuration information.

On Windows, the broker utilizes this embedded Derby database for its configuration database, requiring only the Open Database Connectivity (ODBC) drivers for Cloudscape that is supplied with the product. Use of the embedded Derby database is supported only for test and development. This means that on Windows, DB2 Universal Database does not need to be installed for test and development environments.

4.1.5 WebSphere Message Broker Welcome Page

The Eclipse 3.0 style of the Welcome Page is new for WebSphere Message Broker V6.0. It provides a useful starting point for finding out about WebSphere Message Broker, for both new and experienced users. It also provides links to the Quick Tour, Default Configuration wizard, and the Samples Gallery. These are designed to help you get started with the product by providing a product overview, creating a simple broker configuration, and trying out new product functions. You can track down required information quickly, with the links to the documentation and helpful external Web sites provided.

Message Brokers Toolkit displays the Welcome Page the first time it is opened. When it is closed, the Broker Application Development perspective in the Message Brokers Toolkit is displayed. You can reopen the Welcome Page at any time from the Help menu.

4.1.6 Default Configuration wizard

The Default Configuration wizard in the Message Brokers Toolkit creates a simple configuration to verify the installation of WebSphere Message Broker and for running the WebSphere Message Broker samples.

The wizard accepts a user name and password, and creates a simple broker domain consisting of a Configuration Manager called WBRK6_DEFAULT_CONFIGURATION_MANAGER and a broker called WBRK6_DEFAULT_BROKER. It also creates the database, queue manager and listener and domain connection required for these components and for viewing the broker domain in the Message Brokers Toolkit.

This replaces the Getting Started Wizard from WebSphere Business Integration Message Broker V5.0. The Default Configuration wizard is more robust than the Getting Started wizard, and also reduces the available options to make a clearly defined configuration for verification and running the samples.

4.1.7 Samples Gallery

The Samples Gallery contains a selection of sample resources and instructions that demonstrate how to use WebSphere Message Broker, including new functionality for V6.0. You can import the resources in the form of message flows and message sets into the Message Brokers Toolkit for display and manipulation.

There are 13 new samples for WebSphere Message Broker V6.0, many of which demonstrate new functions such as JavaCompute, Java Message Service (JMS) nodes, and the updates to message aggregation. Others show examples of previous functions such as large messaging and Web services.

The samples are deployed using the Sample Preparation wizard, which sets up any resources required for the sample such as database tables and WebSphere MQ queues. The samples can be automatically removed from the environment once they are finished with. The Sample Preparation wizard replaces the Samples Cheat Sheet from V5.0, performing all the tasks to set up and deploy a sample in one click, and provides function to clean up the default configuration and workspace after they have been used, if required.

4.2 Migration and coexistence

A number of features have been introduced to enable a smooth migration from WebSphere Business Integration Message Broker V5.0 and other previous versions of WebSphere Message Broker to WebSphere Message Broker V6.0.

4.2.1 Coexistence

In order to make migration simpler, it is possible to install WebSphere Message Broker V6.0 alongside WebSphere Business Integration Message Broker V5.0.

Coexistence also extends to the broker domain components. It is possible to view and administer the components from earlier versions in the WebSphere Message Broker V6.0 Message Brokers Toolkit.

User profiles and the Command Console

Because multiple versions of the product can now exist on a single machine, you need to use a profile to select the instance that you want to work with. This initializes the environment so that commands can be invoked from the correct installation of the code.

On Windows, typing a command such as **`mqsilist`** on a command line invokes the WebSphere Business Integration Message Broker command. In order to run the command against the WebSphere Message Broker V6.0 version of the product and any V6.0 components, run the command using the Command Console. This is available from **Start** → **Programs** → **IBM WebSphere Message Brokers 6.**0 → **Command Console**. You can also create a customized command profile for Windows using the sample profile. You can find this in the WebSphere Message Broker Information Center.

On Linux and UNIX® platforms, the bin directory of the installation location provides an environment profile. You can run it to initialize the environment for the selected instance that has been installed.

Domain coexistence

Another feature designed to make migration easier and possible to perform in stages is the coexistence of components. For WebSphere Business Integration Message Broker V5.0, all of the components, including the Message Brokers Toolkit, Configuration Manager, and brokers can coexist with WebSphere Message Broker V6.0 components.

This means that you can administer a WebSphere Business Integration Message Broker V5.0 domain and its components from a WebSphere Message Broker V6.0 Message Brokers Toolkit. Brokers can remain at V5.0 and take part in a V6.0 domain. The brokers must be at one of the following service levels:

- WebSphere Business Integration Event Broker V5.0 (Fix pack 4 or later)
- WebSphere Business Integration Message Broker V5.0 (Fix pack 4 or later)
- WebSphere Business Integration Message Broker with Rules and Formatter Extension V5.0 (Fix pack 4 or later)

4.2.2 Commands for migration

A single command is provided to migrate individual components to WebSphere Message Broker V6.0. The command migrates queues, databases, registry

keys, and files where appropriate, allowing for a stepped approach to migration. For example, you can migrate the Configuration Manager and User Name Server before the brokers.

Part 3, “Migrating from WebSphere BI Message Broker V5.0” on page 311 offers more details about the features associated with migration and the individual steps.

4.3 Message Brokers Toolkit

WebSphere Business Integration Message Broker V5.0 and WebSphere Business Integration Message Broker V5.1 were based on Eclipse version 2.0 and 2.1, respectively. The Message Brokers Toolkit is now built on the Rational Application Development Platform. This is based on Eclipse 3.01.

This section covers the major changes in this version of the Message Brokers Toolkit.

4.3.1 Workspace

The default location for the workspace has now changed from the installation directory of WebSphere Business Integration Message Broker V5.0 to a location specific to each user.

For example: C:\Documents and Settings*user*\IBM\wmbt6.0\workspace on Windows and /home/*user*/IBM/wmqi6.0/workspace on Linux. Figure 4-1 shows the broker application resources and other files stored in the local file system in a directory called the *workspace*, chosen when the Message Brokers Toolkit is started up.

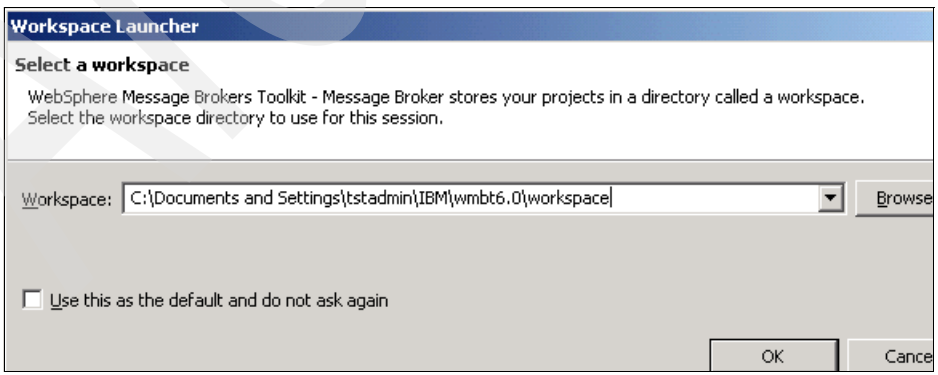


Figure 4-1 Selecting the location to store resources

Mapping editor

The mapping editor has been updated for WebSphere Message Broker V6.0 to provide new functions and to simplify how mappings are created. A new wizard is provided to select the source and target messages and databases. You can map properties, headers, as well as the message body using the mapping editor. Logic is added to the mappings using a spreadsheet-style interface, for example to perform conditional statements, handle repeating elements, and make choices based on the content of a message or database.

You can create submaps to provide reusable mapping function and to allow multiple input elements. Maps can also call existing extended Structured Query Language (ESQL) functions, as well as be invoked from ESQL themselves.

You can use an expression editor section to enter XPath expressions. XPath provides functions for extracting and manipulating data contained in the message or database fields. You can add references to the messages and databases to the expression editor by dragging and dropping a field from a selected source. Figure 4-2 shows an example of mapping with *for* statements between two messages.

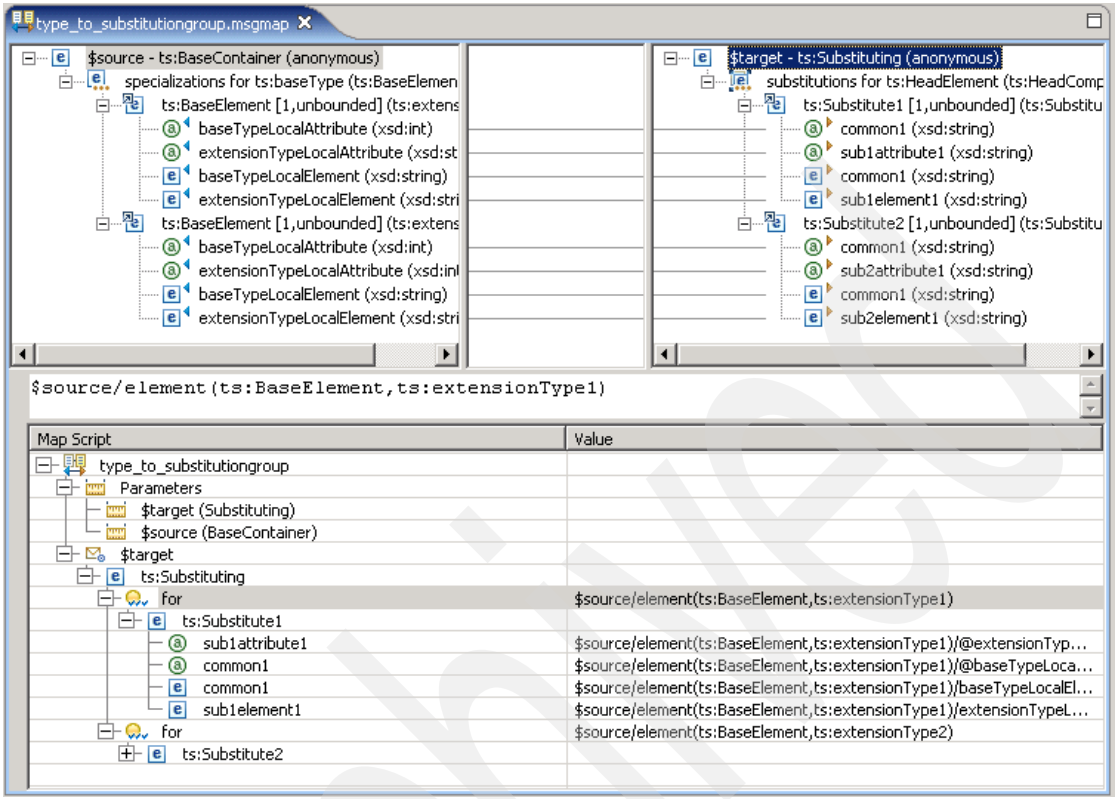


Figure 4-2 An example message map

Message definitions

The Create New Message Definition wizard has been enhanced for WebSphere Message Broker V6.0 for the import and setup of properties for existing definitions of C, COBOL, Web Services Description Language (WSDL), Extensible Markup Language (XML) schema, and Extensible Markup Language Document Type Definition (XML DTD) files. Additionally, it provides some ready-created message definitions, available through the Create Message Definition wizard, and includes Simple Object Access Protocol (SOAP), multipart Multipurpose Internet Mail Extensions (MIME) and SAP IDoc messages.

Java perspective

Use the Java perspective when coding Java for use with the JavaCompute node. You can add any valid Java code to a JavaCompute node to make full use of the existing Java user-defined node application programming interface (API) to process an incoming message.

The JavaCompute node uses the same API as the user-defined nodes (Java plug-in nodes and parsers) which includes extra methods to simplify code involving message routing and transformation such as accessing named elements in a message tree, setting their values, and creating elements without the need to navigate the tree explicitly.

You can use aquery language called XPath in the Java code for navigating the structure of XML documents. XPath makes coding for XML messages simpler because it makes referring to the message structure more precise, takes away the need for manual navigation, and reduces the number of statements in the code. You can use XPath to query any tree structure so it is not restricted for use with just XML messages.

The Java files are verified on saving. Any errors are displayed in the file and in the Problems view in the Broker Application Development perspective. Interactive assistance with errors is provided as well as the other Java editing facilities, including code completion, integrated Javadoc documentation, and automatic compilation.

Data tools

The Message Brokers Toolkit provides a Data perspective as part of the Rational Application Developer Platform. The tools provided are useful in message flow development where the message flows interact with databases. The tool to create a relational database (RDB) definition file makes it possible to access a database and add the definitions of its tables to a message flow project in the workspace. Once you add definitions for a database to a message flow project, they can be included in mappings and assist with validation for ESQL.

Plug-in nodes and parsers

User-defined nodes cannot be developed in WebSphere Event Broker, but WebSphere Event Broker users can now deploy nodes provided by third parties.

Flow debugging

The flow debugger has been enhanced for WebSphere Message Broker V6.0, with the major difference being the integration of the flow, ESQL, and Java perspectives. This makes the debugger easier to use because there is no need now to swap between different perspectives.

Support has also been added to enable mappings and Java code from the JavaCompute node to be debugged. You can use a new configuration wizard to create a configuration for specific execution groups, which can be reused.

You can debug brokers from WebSphere Business Integration Message Broker V5.0 using the Message Brokers Toolkit, but the version of Rational Agent Controller you use must match the broker version.

Removal of Rapid Application Deploy

Deploy using the Message Brokers Toolkit is now only possible using broker archive deploy, because the function for rapid application deploy has been removed.

4.4 New nodes

A number of new nodes have been provided to enhance the functionality of WebSphere Message Broker, shown in Table 4-1.

Table 4-1 New nodes for WebSphere Message Broker V6.0

Node name	Description
MQGet	Can be used anywhere in a flow to retrieve a message from a WebSphere MQ queue.
MQOptimized node	Is used to publish a persistent JMS message to a single subscriber to improve performance over ordinary publish/subscribe message flows.
JMSInput	Receives messages from JMS destinations accessed through a connection to a JMS provider.
JMSOutput	Sends messages to JMS destinations using the Java Message Service specification.
JMSMQTransform	Transforms a message with a JMS message tree into a message compatible with a WebSphere MQ JMS provider.
MQJMSTranform	Receives messages that have a WebSphere MQ JMS provider message tree format, and transforms them into messages for JMS destinations.
JavaCompute	Used for transforming messages using a Java API.
TimeoutControl	Used in conjunction with a TimeoutNotification node. This node receives an input message that contains a timeout request.
TimeoutNotification	Processes timeout request messages that are set by its associated TimeoutControl node and propagates copies or parts of the message to the next node in the message flow.

Node name	Description
Validate	Checks whether a message matches a given message structure definition. If it does, it is output to the match terminal; else it is propagated to the failure terminal.
Passthrough	Used to provide a label to indicate the subflow it is contained in. Messages are not processed; they just pass through this node.

Changes have also been made to some of the existing nodes, for example the compute and mapping nodes. In addition, the aggregation nodes now use WebSphere MQ queues to store state information instead of a database. This improves the throughput of all requests, especially the performance of nonpersistent requests.

4.5 Enhanced message modeling

This section discusses enhanced Web Services support, Parser Domains, and runtime validation.

4.5.1 Enhanced Web services support

WebSphere Message Broker V6.0 provides support for modeling and working with SOAP messages for Web services. It provides full support for the construction and parsing of SOAP messages, the addition of XML namespace support through the extension of the Message Repository Manager (MRM) XML Wire Format, and the addition of the XML Namespace domain. It also provides sample message definitions for SOAP 1.1 and 1.2. Existing Web Services Description Language (WSDL) definitions of a variety of styles can be imported into the Message Brokers Toolkit.

WSDL definitions generated from a message set can be used by external applications such as . NET. The WSDL generator also leaves a message set SOAP-ready for use by the MRM XML parser.

It also provides support for SOAP with attachments using a new MIME parser domain.

4.5.2 Parser domains

The following new domains have been introduced in WebSphere Message Broker V6.0:

- XMLNSC

- JMSMap
- JMSStream
- MIME

You can find more information about the new domains in the WebSphere Message Broker Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp

4.5.3 Runtime validation

Improvements have been made to validation of messages at runtime including:

- Validation options available on more nodes
- A new Validate node to validate a message in the middle of a message flow
- The ability to detect all validation failures in a message before throwing an exception
- The ability to force a complete parse of a message independently of runtime validation

4.6 Configuration and administration

There have been a number of changes and enhancements to configuration and administration including the Configuration Manager. There are also a variety of new and improved commands and utilities that are briefly described in this section.

4.6.1 Configuration Manager changes

The changes to the Configuration Manager are briefly discussed here. Chapter 11, “Migrating the Configuration Manager” on page 177 also provides some information on changes that are important after migration.

Configuration Manager databases

The Configuration Manager no longer requires any external databases because it uses an internal repository, removing the requirement for DB2 Universal Database for use with the Configuration Manager.

Multiple Configuration Managers on a single machine

You can now create multiple Configuration Managers on a single machine by specifying a name on the command line, for example:

```
mqsicreateconfigmgr CONFIG_MGR1 -i userid -a -q CONFIG_QM
```

There is still only one Configuration Manager per domain, but you can administer multiple domains from the Message Brokers Toolkit.

Configuration Manager available on all platforms

You can now create the Configuration Manager on all of the supported broker platforms including AIX, HP-UX, Linux (x86 platform), Linux (zSeries platform), Solaris, Windows, and z/OS.

4.6.2 Configuration Manager Proxy API

The Configuration Manager Proxy (CMP) is a Java application programming interface to the Configuration Manager. It enables programs to be written to automatically create and administer broker domains. Java objects map to domain objects to make the Configuration Manager Proxy API simpler to code. A sample API Exerciser is supplied to demonstrate the capabilities and how to program an application to utilize them.

You can use the Configuration Manager Proxy API for the following tasks:

- Viewing domain objects
- Adding and removing brokers
- Modifying broker properties
- Setting up Access Control Lists
- Backup and restore of domain
- Editing and deploying topology
- Editing and deploying topics
- Querying and deleting active subscriptions
- Creating and deleting execution groups
- Deploying bar files
- Viewing deployed resources
- Querying logs
- Starting and stopping message flows
- Controlling user trace

The available administration functions are comparable to those found in the Message Brokers Toolkit and the command line, and can be a powerful tool for managing administration of the domain if desired.

4.6.3 Versioning of deployed resources

In WebSphere Message Broker, resources such as message flows and message sets can have version information included in the files, such as when the resources were compiled, when they were deployed, and the name of the broker archive file that they were deployed with. This assists with version control, and helps to return to a working version of a message set or message flow if the latest version is discovered to have a problem or defect.

In addition to built-in version information, you can specify your own unique version numbers and your own keywords, such as author, for enhanced identification of deployed objects. You can add this information to message flows, message sets, ESQL files and Java files. You can then easily view this information through the Broker Administration perspective in the Message Brokers Toolkit.

4.6.4 New command line utilities

This section presents the new command line utilities.

Creating and deleting execution groups

Commands now exist to create and delete execution groups on the command line, in addition to the Message Brokers Toolkit. The new commands to enable this are:

- `mqsicreateexecutiongroup`
- `mqsideleteexecutiongroup`

Stopping and starting message flows

Commands now exist to start and stop message flows on the command line, in addition to the Message Brokers Toolkit. The new commands to enable this are:

- `mqsistartmsgflow`
- `mqsistopmsgflow`

Backing up and restoring domain data

A command now exists to save a copy of the Configuration Manager's repository, with a name provided as a parameter, to a specific directory. A command also exists that replaces the entire Configuration Manager's repository with a previously backed-up version. The new commands to enable this are:

- `mqsibackupconfigmgr`
- `mqsirestoreconfigmgr`

Creating and deleting broker databases on Windows

Commands are provided on Windows to create a broker database on the command line, and also to delete a database created this way. The command creates a named database and sets up an ODBC data source name.

The command only supports Derby and DB2 Universal Database. If you use Derby as the database, then a Windows service is created and started for that database.

The new commands to enable this are:

- `mqsicreatedb`
- `mqsideletedb`

Other new commands

The other new commands added since WebSphere Business Integration Message Broker V5.0 are:

- `mqsichangedbimgr`
- `mqsicreatemsgdefsfromwsdl`
- `mqsireadbar`

This redbook does not discuss these in any further detail, but you can find information about them in the Reference section of the WebSphere Message Broker Information Center.

4.6.5 Other administration updates

This section presents other administration updates.

Password prompt

To improve the security of using command line administration utilities, there is now a method of entering commands without displaying the password on the screen. Enter a command such as **`mqsicreatebroker`** without including a password, for example:

```
mqsicreatebroker BRK2 -i userid -a -q BRK2_QM -n BRK2DB
```

Note: The `-a` flag has no password. A prompt appears asking for a password to be entered. When you enter the password, nothing is displayed as you type. When you press Enter, you are prompted to retype the password for verification. If the passwords match, the command continues to be processed. You can type commands with a password after the `-a` flag as before if you want.

4.7 Security

There are a number of enhancements for security for WebSphere Message Broker V6.0, including the following:

- The Hypertext Transfer Protocol (HTTP) transport has been extended to provide Hypertext Transfer Protocol Secure (HTTPS) support, which provides added privacy and security.
- Communication between the Message Brokers Toolkit and the Configuration Manager has been extended to allow the use of Secure Sockets Layer (SSL) for added security.

4.8 Other enhancements

There are numerous other enhancements and new features that have been introduced in WebSphere Message Broker V6.0, and it is beyond the scope of this redbook to discuss all of them. This section groups together a few of the remaining enhancements, but you can find information about the other enhancements in the WebSphere Message Broker Information Center, in the What is New topics under the Introduction on the Web at:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp`

4.8.1 Rational Agent Controller

You must use the Rational Agent Controller with the flow debugger for WebSphere Message Broker. Install this application on systems where brokers are to be debugged. If you are debugging a broker on the Message Brokers Toolkit machine, then you must install Rational Agent Controller on that system, or on the remote system if the broker is remote.

The Rational Agent Controller also requires TCP/IP connectivity to the system that the Message Brokers Toolkit is running on to allow it to communicate with the Rational Agent Controller client that is embedded in the Message Brokers Toolkit.

This is an optional piece of software which is only required for debugging using the flow debugger in the Message Brokers Toolkit. You can find instructions for installing this product in the WebSphere Message Broker Information Center.

If other products that use tooling built upon the Rational Application Development platform exist on a machine, these may also use Rational Agent Controller to perform tasks such as debugging. You can have only one installation of Rational Agent Controller on a machine at any given time.

To debug a domain with WebSphere Business Integration Message Broker V5.0 brokers in addition to V6.0 brokers, you must use Rational Agent Controller V5.0.2.1, because Rational Agent Controller V6.0.0.1 cannot be used with WebSphere Business Integration Message Broker V5.0 brokers.

4.8.2 Enhancements to ESQL

There are many updates and enhancements to ESQL in WebSphere Message Broker V6.0 that are not listed here. You can find iInformation about ESQLin the Reference section of the WebSphere Message Broker Information Center.

Some examples of the enhancements for transforming and routing messages using ESQL are:

- In-memory cache to reduce access to databases for read-only routing or validation data
- Improved support for creating DATETIME variables
- Access to multiple databases from the same Compute, JavaCompute, Database, or Filter node
- Dynamic database schemas
- New user-defined message flow properties that can be passed as parameters to ESQL to modify standard behavior
- Access to environment information, for example, message flow name and broker name
- Improved support for result sets returned by database-stored procedures
- Improved error recovery using Structured Query Language (SQL) handlers
- Multiple out terminals to combine the function of Compute and Filter nodes
- Improved CAST error handling**To author: please give full form of CAST. Unable to find expansion on IBM Terminology etc.Should it be “Cast”?**

4.9 List of enhancements and changes

Table 4-2 shows most of the major enhancements in WebSphere Message Broker since V5.0.

Table 4-2 Major enhancements in the product since Message Broker V5.0

Feature	Release in which introduced
WebSphere Message Broker samples	WebSphere Business Integration Message Broker V5.0 (extended and updated for WebSphere Message Broker V6.0)
Graphical mapping	WebSphere Business Integration Message Broker V5.0 (enhanced for WebSphere Message Broker V6.0)
Command line administration - starting and stopping message flows and execution groups	WebSphere Message Broker V6.0
TLog messaging standard	WebSphere Business Integration Message Broker V5.0 Fix pack 4
Invoke Java from ESQL	WebSphere Business Integration Message Broker V5.0 Fix pack 4
HL7 messaging standard	WebSphere Business Integration Message Broker V5.0 Fix pack 5
Configuration Manager no longer needs a database	WebSphere Message Broker V6.0
Hard copy installation guide	Removed at WebSphere Business Integration Message Broker V5.0, reinstated for WebSphere Message Broker V6.0
Default Configuration wizard	WebSphere Message Broker V6.0
Coexistence with previous versions	WebSphere Message Broker V6.0
Migration of individual components	WebSphere Message Broker V6.0
Simplification of z/OS install	WebSphere Message Broker V6.0
Message Brokers Toolkit for Linux on Intel	WebSphere Message Broker V6.0
Routing and transformation using Java	WebSphere Message Broker V6.0
Access to multiple databases from Compute, JavaCompute, Database and Filter node	WebSphere Message Broker V6.0
Dynamic database schemas	WebSphere Message Broker V6.0
Access to environment information	WebSphere Message Broker V6.0

Feature	Release in which introduced
Multiple out terminals for Compute	WebSphere Message Broker V6.0
Extensible Stylesheet Language Transformation (XSLT) performance	WebSphere Message Broker V6.0
Broker performance	WebSphere Message Broker V6.0
Aggregation performance	WebSphere Message Broker V6.0
SOAP 1.2 supported	WebSphere Message Broker V6.0
HTTP 1.1 supported	WebSphere Message Broker V6.0
SOAP schemas provided	WebSphere Message Broker V6.0
WSDL importer	WebSphere Message Broker V6.0
HTTPS support	WebSphere Message Broker V6.0
JMS transport support	WebSphere Message Broker V6.0
TimeoutControl & TimeoutNotification nodes	WebSphere Message Broker V6.0
MQGet node	WebSphere Message Broker V6.0
Message parsing performance	WebSphere Message Broker V6.0
MIME parser	WebSphere Message Broker V6.0
Validation improvements	WebSphere Message Broker V6.0
Validate node	WebSphere Message Broker V6.0
Run-time Versioning	WebSphere Message Broker V6.0
Configuration Manager Proxy	WebSphere Message Broker V6.0
Configuration Manager available on all broker platforms	WebSphere Message Broker V6.0
Multiple Configuration Managers on a single machine	WebSphere Message Broker V6.0
Code page conversion updated	WebSphere Message Broker V6.0

5

Migration considerations

This chapter discusses the high-level considerations for migration to, including the following:

- Selecting a migration method
- Checking supported software levels
- Preparing the environment for migration
- Installation and verification of WebSphere Message Broker V6.0

5.1 Initial considerations

A number of decisions, and their correct order, are required to successfully migrate to WebSphere Message Broker V6.0. This section discusses these decisions and the options that need to be considered when making them.

5.1.1 Selecting a migration method

There are three primary ways of migrating to WebSphere Message Broker V6.0:

- Migrating from a previous version and completely replacing with WebSphere Message Broker V6.0.

  This is a potential method for migration when all the components reside on a single machine. This machine may be a test machine where high availability is not an issue, for example, for development or test environments. This method involves preparing the previous version for migration, migrating all of the components at the same time, and then removing the previous version after testing.

- Migrating from the previous version to WebSphere Message Broker V6.0 and coexisting temporarily with the previous version.

  This is a potential method of migration for more complex environments. These environments may be where the components of the domain are located across multiple machines, or where multiple domains exist.

  This method involves preparing the previous version for migration, installing and configuring WebSphere Message Broker V6.0, and performing staged migration of individual components. This is facilitated by the ability of WebSphere Message Broker V6.0 to coexist with previous versions of the product, which is discussed further in 6.2, “Coexistence with WebSphere Message Broker V6.0” on page 93. This method also allows you to create new resources and components in WebSphere Message Broker V6.0, while allowing the administration of components created in the previous version.

- Migrating from the previous version to WebSphere Message Broker V6.0 and migrating some of the components to new machines.

  You can use this method for two scenarios:

  – Setting up a mirror of the previous version in a different location for maintaining high availability during migration.

  – Taking the opportunity to change the platform of the Configuration Manager to a new platform or changing the tooling platform to Linux (x86 platform).

You can choose a combination of these methods depending upon the machine, configuration, and environment. The migration order of individual components is flexible and depends upon the method of migration you choose. The components that you can migrate are as follows:

- Message flow application resources (such as message flows, message sets, and user-defined nodes)
- Configuration Manager
- User Name Server
- Control Center or Message Brokers Toolkit
- Brokers

Information about how these components can coexist from previous versions with WebSphere Message Broker V6.0 is presented in 6.2, “Coexistence with WebSphere Message Broker V6.0” on page 93.

Different strategies for migration are appropriate for different environments. For example, you can use a straightforward migrate and replace method for development environments, while a staged migration of components using coexistence is more suitable for a test or production environment. Decide when and where to migrate the components to match your requirements.

5.1.2 Maintaining active brokers during migration

As discussed in the previous section, if existing brokers need to be kept active during migration—for example, in a production environment—you need to adopt a high-availability strategy.

A possible configuration to maintain active brokers has at least three brokers supporting the executing applications. While Broker 1 is being migrated, Broker 2 and Broker 3 can provide backup support for each other. It is also important that the test domain is identical to the production domain, so that you can identify any problems during migration of the test domain before migration of the production domain.

Alternatively, you can set up new WebSphere Message Broker V6.0 components and migrate the message flow application resources to the new components. You can separately test and verify these components for consistency and performance. Subsequent to this testing, you can switch the environment over to the WebSphere Message Broker V6.0 components.

An important consideration when using this method is that it does not maintain the broker state. This impacts users of publish/subscribe functionality.

5.1.3 Finding out what is new

It is useful to consider the impact of new and changed function in WebSphere Message Broker V6.0 on your configuration. Planning these changes before migration may influence the order of migration. Find out what is new in WebSphere Message Broker V6.0 using the following chapters:

- Integrator V2.1 users, refer to Chapter 3, “What is new for WebSphere MQ Integrator V2.1 users” on page 27.
- Message Broker V5.0 users, refer to Chapter 4, “New features in WebSphere Business Integration Message Broker V5.0” on page 55.

These chapters provide assistance on the differences between the previous versions of the product and WebSphere Message Broker V6.0. Further guidance on the differences is also provided where appropriate in the migration chapters in:

- Part 2, “Migrating from WebSphere MQ Integrator V2.1” on page 105
- Part 3, “Migrating from WebSphere BI Message Broker V5.0” on page 311

An additional consideration common to both previous versions is to include a development repository for the storage of message flow application resources for WebSphere Message Broker V6.0. There are a number of development repositories available that are compatible with Eclipse. You can find information about the development repository providers on the Eclipse Web site at:

`http://www.eclipse.org/community/index.html`

Refer to the WebSphere Message Broker V6.0 documentation for specific details on how to use changed and new function in WebSphere Message Broker. The following Tip box provides a link to the online version of the WebSphere Message Broker Information Center, and instructions on how to locate documents referenced in this redbook.

Tip: The subsequent chapters of this book refer to the online help documentation using a topic reference, followed by a navigation path to where the topic can be found in the Information Center's table of contents.

The topic reference is indicated at the bottom right of each topic in the WebSphere Message Broker Information Center. You can use this topic reference to identify the specific topic in the Information Center provided with the product or the online version available at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp

While searching for a particular topic in the Information Center, use double quotation marks. For example, if you are searching for topic ab00015_, go to the Information Center's search box and type "ab00015_" including the double quotation marks.

In this book, reference to all such topics is done at V6.0.0.0, but we recommend that you refer to the latest available documentation with the same topic reference.

5.1.4 Deciding on a testing strategy

When performing a migration to WebSphere Message Broker V6.0 it is important to ensure at each stage that the migration or configuration of the WebSphere Message Broker components is successful. The following steps might be required to ensure that each migration step is successful:

- Verify successful backup of the previous version components and resources.
- Verify successful installation of WebSphere Message Broker V6.0.
- Verify successful migration of components.
- Verify successful migration of resources.
- Verify successful setup of the development environment.
- Verify successful deployment of migrated resources.

A chapter is included for each of the versions to provide details of suggested checks for verifying that the migration steps were successful:

- Integrator V2.1 users, refer to Chapter 16, "Migration verification" on page 287.
- Message Broker V5.0 users, refer to Chapter 27, "Verifying migration to WebSphere Message Broker V6.0" on page 495.

If the following environments exist in your organization, you must migrate them in this order:

1. Development environment
2. Test environment
3. Production environment

Prepare a test plan as part of the migration planning and consider using a sample set of test flows and other resources for performing initial verification during migration. Select resources that cover a broad range of functionality. Examples are provided in the chapters mentioned above. Using these test resources may help in locating any problem areas at an early stage in the migration. You can also use these tests as a baseline to measure the performance of your configuration at the previous level. You can measure the performance again after migration. Any major discrepancies you find potentially indicate an unseen problem.

If your organization does not have a test environment, it is advisable to test as much of the migration as possible on a different machine to prevent downtime on the production system. You can set up a temporary test environment for the purpose of testing the migration of resources. Another approach is to create a mirror of the domain that you can migrate to check and resolve any unexpected issues, while the original domain remains available.

It is important to ensure that WebSphere Message Broker V6.0 has been installed and configured successfully before beginning migration tasks. Some guidance on installing and verifying the configuration of WebSphere Message Broker V6.0 is provided in 5.4, “Installation and verification of Message Broker V6.0” on page 85.

5.2 Checking the supported environments for migration

Migration to WebSphere Message Broker V6.0 from versions of the product prior to WebSphere MQ Integrator V2.1 are not supported and are not discussed in this redbook. In order to migrate from previous versions, you must first upgrade to one of the supported versions.

We recommend that before migration is attempted, the current version environment should be up-to-date with the latest supported service levels, both for the operating systems and software. Details of the systems requirements for all versions of the product can be found at:

http://www-1.ibm.com/support/docview.wss?rs=849&uid=swg27006551

Details of the latest supported environments for WebSphere Message Broker V6.0 are available on the product support page at:

http://www.ibm.com/software/integration/wbimessagebroker/requirements/

Important: WebSphere MQ Integrator users must either install fix pack 6 or apply APAR IY45459 before attempting to migrate if they have installed fix pack 2, 3, 4, or 5.

The latest version of the product readme may also contain useful information about supported software levels.This is found at:

http://www.ibm.com/software/integration/mqfamily/support/readme/all60_read.html

Some guidance is provided on the minimum software levels for migration for the platforms covered in this book in the following sections:

- 7.2.2, “The physical topology” on page 110
- 17.2.2, “The physical topology” on page 316

5.2.1 Upgrading other software

The minimum supported levels of software for migration to WebSphere Message Broker V6.0 may mean that you must upgrade other software before the migration to WebSphere Message Broker V6.0 can begin.

If you need to upgrade other software in order to perform the migration to WebSphere Message Broker V6.0, ensure that the upgraded configuration works as expected before attempting to migrate to WebSphere Message Broker V6.0.

Migration of other software such as upgrading the operating system, migrating to WebSphere MQ V6.0, or upgrading of databases is beyond the scope of this book.

Although WebSphere MQ 5.3 is supported for use with Message Broker V6.0, you may decide to upgrade to WebSphere MQ V6.0 as part of the migration because it is supplied in the WebSphere Message Broker V6.0 package. You can integrate the WebSphere MQ V6.0 Explorer into the WebSphere Message Broker V6.0 Toolkit. This is explained in Appendix B, “Integrating WebSphere MQ Version 6.0 Explorer into Message Brokers Toolkit Version 6.0” on page 549.

WebSphere MQ V6.0 Fundamentals, SG24-7128 may be a useful guide to the new version.

5.2.2 Hardware considerations

You must consider the following hardware issues before migration:

- Hard disk storage is required for backing up resources, including databases before migration.
- Extra hard disk space is required if the migration is carried out by installing WebSphere Message Broker V6.0 on the same machine as a previous version.
- If any components are to be moved to new machines or platforms during the migration, these machines must have sufficient hard disk and memory resources.
- There are minimum requirements for Random Access Memory (RAM) for distributed platforms. Check the latest System Requirements information from the WebSphere Message Broker Support Web site at:

  http://www.ibm.com/software/integration/wbimessagebroker/requirements/
- There are minimum requirements for hardware on all platforms. Check the latest System Requirements information from the WebSphere Message Broker Support Web site at:

  http://www.ibm.com/software/integration/wbimessagebroker/requirements/
- Additional hard disk space and memory are required for prerequisite and corequisite software. More information is provided in the Installation Guide.
- Additional hard disk space is required in a configured environment, and when service is applied to the system in the future.

Note: You can find the hardware requirements for WebSphere Message Broker V6.0 in Chapter 1 of the hardcopy Installation Guide. You can also find this in the WebSphere Message Broker Information Center, under the topics **Installing** → **Installation Guide** on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah24100_.htm

5.3 Preparing the environment for migration

This section provides some tips on preparing your existing environment for migration.

In addition to upgrading the environment to the latest supported levels, you must perform the following tasks:

1. Remove any components or execution groups that are no longer required.
2. Remove any message flow application resources that are no longer required.
3. Document any system-specific information such as any environment variables or tuning parameter settings.
4. Document user IDs and passwords where appropriate.
5. Document details of components and deployed resources in case any problems occur. These need to be recreated or redeployed.

Important: Back up the existing environment prior to migration. Instructions on doing this are provided in the following chapters:

- If you are a WebSphere MQ Integrator V2.1 user, refer to Chapter 9, "Backing up the WebSphere MQ Integrator V2.1 domain" on page 133.
- If you are a WebSphere Business Integration Message Broker V5.0 user, refer to Chapter 19, "Backing up the WebSphere Business Integration Message Broker V5.0 domain" on page 339.

5.3.1 Preparing message flow application resources for migration

There have been a number of changes to message flow and message set behavior in WebSphere Message Broker V6.0 that might impact resources created in previous versions when they are migrated. Some of the major changes that might cause problems during migration are covered here, but refer to the production documentation for more detailed information.

There are more impacts for Integrator V2.1 users migrating to WebSphere Message Broker V6.0 than for WebSphere Business Integration Message Broker V5.0 users. This is because the Message Brokers Toolkit includes a number of major changes to the Control Center — for example, the use of an Eclipse graphical user interface (GUI), the addition of mapping nodes, and separate ESQL files.

Message flow migration

Check for the following major changes before migration:

- The use of MQeInput and MQeOutput nodes is deprecated in WebSphere Message Broker V6.0.

  You can still deploy flows that contain these nodes to previous version brokers, but these nodes are treated as MQInput and MQOutput nodes to a WebSphere Message Broker V6.0 message flow. You can find further

information on this in the WebSphere Message Broker Information Center under the topics **Reference → Migration and Upgrade → Message flow migration notes → Migrating a message flow that contains WebSphere MQ Everyplace nodes** on the Web at*:*

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac34700_.htm

- WebSphere Message Broker V6.0 has been modified to allow style sheets and XML files used by XMLTransformation nodes to be deployed.

  You should decide whether to deploy the files later on, or whether to move them to the correct directory structure. For more information about this, select the path **Reference → Migration and Upgrade → Message flow migration notes → Migration of style sheets and XML files** in the WebSphere Message Broker V6.0 Information Center on the Web at:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah35007_.htm

Specific details for other changes to message flows from previous versions to WebSphere Message Broker V6.0 are covered in the relevant sections.

For further information, select the path **Reference → Migration and Upgrade → Message flow migration notes** in the WebSphere Message Broker Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah20290_.htm

Plug-in extensions and parsers (user-defined nodes)

Plug-ins and parsers are expected to work “as is” without problems in WebSphere Message Broker V6.0. User-defined nodes (plug-in nodes) and parsers do not require recompilation, but there are steps that are required, post-migration, to make them available for viewing and editing in the Message Brokers Toolkit. For further information on these steps:

- Select the path **Reference → Migration and Upgrade → Message flow migration notes → Migrating a user-defined node** in the WebSphere Message Broker V6.0 Information Center on the Web at:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/as10009_.htm

- Select the path **Developing applications → Developing user-defined extensions → Creating the user interface representation of a user-defined node in the workbench** in the WebSphere Message Broker V6.0 Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/as04480_.htm

SupportPacs and third-party resources

Functionality from some SupportPacs from previous versions of the product has now been incorporated in whole or in part in WebSphere Message Broker V6.0. Refer to the following chapters for information about the new function in WebSphere Message Broker V6.0:

- Integrator V2.1 users, refer to Chapter 3, “What is new for WebSphere MQ Integrator V2.1 users” on page 27.
- Message Broker V5.0 users, refer to Chapter 4, “New features in WebSphere Business Integration Message Broker V5.0” on page 55.

For more information, select the path **Product overview** → **Introduction** → **What’s new in Version 6.0?** in the WebSphere Message Broker Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ab00025_.htm

Many of the SupportPacs that were available for previous versions will be migrated or updated for WebSphere Message Broker V6.0, but not all of them are available at release.

Refer to the SupportPacs Web site for information about which SupportPacs are supported for WebSphere Message Broker V6.0 at:

http://www.ibm.com/software/integration/support/supportpacs/category.html

For third-party extensions and resources, contact the third-party supplier for updated versions and further information.

5.4 Installation and verification of Message Broker V6.0

This section briefly covers the considerations for installing and verifying WebSphere Message Broker V6.0. Step-by-step instructions for installing WebSphere Message Broker V6.0 are not covered here. For detailed instructions for installing the product on any supported platform, refer to the hard copy installation manual with the product package, or select the path **Installing** → **Installation Guide** in the WebSphere Message Broker Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah24100_.htm

When WebSphere Message Broker V6.0 is installed, you can verify the installation by creating a simple configuration and deploying one of the WebSphere Message Broker V6.0 samples, or by creating a simple message flow. On Windows or Linux (x86 platform), you can use the Default Configuration wizard to quickly set up a simple domain for test and verification purposes. You can then use the Getting Started samples to verify the configuration. Follow the instructions for running these from the installation guide and in the Samples Gallery after selecting these from the Welcome Page.

On other platforms, you have to create the components manually and create a broker domain in the Message Brokers Toolkit. You can import WebSphere Message Broker V6.0 samples into the Message Brokers Toolkit and deploy to the broker on a remote machine. A suggested sample to use for simple verification purposes is the Soccer Sample. This sample was also present in WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0, and works on WebSphere Event Broker as well as WebSphere Message Broker.

For further information on using WebSphere Message Broker V6.0, including installation and basic configuration, refer to *WebSphere Message Broker Basics*, SG24-7137.

6

Planning for migration

This chapter discusses the planning considerations for migration to WebSphere Message Broker V6.0.

The following topics are discussed:

- Coexistence of previous version products with the latest WebSphere Message Broker V6.0 family of products
- Planning migration from WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0
- Planning migration from WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0
- Changing the component platform during migration

6.1 Creating a plan for migration

This section offers a high-level guide to producing a WebSphere Message Broker V6.0 migration plan. Some premigration considerations for this section are available in Chapter 5, “Migration considerations” on page 75.

A number of decisions need to be made about how to carry out the migration, the steps to take, and in which order. Decisions also need to be made about which roles need to take part in the migration.

6.1.1 Defining the migration tasks

A section is provided for each of the previous versions for version-specific high-level premigration, migration and post-migration tasks. These sections include:

- 6.3, “Planning migration from WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0” on page 96.
- 6.4, “Planning migration from WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0” on page 100.

To determine the migration order for components in your environment, it is important to understand the coexistence of WebSphere Message Broker V6.0 with previous versions. Coexistence of components differs between Integrator V2.1 and Message Broker V5.0. It is therefore important that you understand the implications when migrating your environment. Refer to the following sections for a description of coexistence and how it works with different versions:

- 6.2, “Coexistence with WebSphere Message Broker V6.0” on page 93
- 6.2.1, “Coexistence of WebSphere MQ Integrator V2.1 with WebSphere Message Broker V6.0” on page 94
- 6.2.2, “Coexistence of WebSphere Business Integration Message Broker V5.0 with WebSphere Message Broker V6.0” on page 95

For technical information and step-by-step instructions for the individual premigration and post-migration tasks, refer to the following parts of the book:

- Part 2, “Migrating from WebSphere MQ Integrator V2.1” on page 105.
- Part 3, “Migrating from WebSphere BI Message Broker V5.0” on page 311.

6.1.2 Defining roles

It is likely that personnel with different roles are required at different stages of the migration. Planning helps in assigning roles to tasks. It also helps to know where the tasks are in the migration process.

Table 6-1 shows an example of how you can organize human resources in a migration plan.

Table 6-1 Human resource needs per stage

Stage post-migration	Activity	Role
Premigration	Develop the migration plan	Project lead
Premigration	Determine hardware and software requirements	Project lead
Premigration	Application software upgrades and/or operating system upgrades	Database administrator, WebSphere MQ administrator, system administrator
Premigration	Message flow application updates	Developer, QA analyst
Premigration	Documenting the current environment	Project lead, system administrator
Premigration	Baseline testing of the current environment	QA analyst, performance analyst
Premigration	Backups of the existing environment	Project lead, developer, systems administrator, database administrator, WebSphere MQ administrator
Premigration	Premigration check of components	Systems administrator
Install and migration	Developer workstations	Developer, Project lead
Install and migration	Testing the environment	Project lead, developer systems administrator
Install and migration	Migrating tooling and resources	Project lead, developer, systems administrator, database administrator, WebSphere MQ administrator
Install and migration	Migrating the Configuration Manager	Project lead, developer, systems administrator, database administrator, WebSphere MQ administrator
Install and migration	Migrating the User Name Server	Project lead, developer, systems administrator
Install and migration	Migrating brokers	Project lead, developer, systems administrator, database administrator, WebSphere MQ administrator

Stage post-migration	Activity	Role
Install and migration	Quality Assurance	Project lead, developer, systems administrator, QA analyst, performance analyst
Post-migration	Verifying the migration	Project lead, developer, systems administrator, QA analyst, performance analyst
Post-migration	Removing no longer required software/database tables	Project lead, developer, systems administrator, database administrator, WebSphere MQ administrator

The steps required and their order of occurence may differ from those shown, depending on the chosen migration method, and whether a single domain or multiple domains are migrated. More information about choosing the order of migrating components is available in 6.2, “Coexistence with WebSphere Message Broker V6.0” on page 93.

6.1.3 Recording the current environment

Before migration, we recommend that you document the current environment configuration. For example, record the following items:

- A complete list of domains
- A complete list of components
- Any environment variables
- Tuning parameter settings
- User IDs and passwords
- User privileges
- Deployed resources and execution groups

6.1.4 Planning the schedule

After determining the individual steps of the migration, you can use these steps to estimate a duration and produce a schedule. For each migration task, consider the following questions:

- What is the task?
- When is it expected to start?
- When is it expected to complete?
- What is the absolute deadline?

Certain tasks have to be performed in a particular order. Information about prerequisite tasks may be included in the schedule to highlight which activities rely on others.

6.1.5 Fault planning and testing

Adequate preplanning makes resources available. A sequential migration procedure minimizes faults and problems when you migrate to the WebSphere Message Broker V6.0. With a complex environment, unforeseen problems may occur. Problems may arise if minor configuration steps are missed, or unique situations are encountered. We do not cover these unique situations in this book. It is therefore important that you plan to recover the configuration if there is a failure. Also, consider migrating with redundant brokers in place.

Ensure that the components and resources are backed up sucessfully before performing the migration.

It is also important to verify that the migration has been successful and that there are no hidden problems. To do this, test the environment before the migration and once again after each of the major migration tasks. This enables checking of functional consistency and that performance has not been adversely affected.

Allocate some time for testing and resolving issues within the migration plan schedule. Include contingency plans for restoring the existing environment should major problems occur during the migration. Alternatively, migrate with a backup environment in place.

6.1.6 Stage checklists

For a successful migration, it is important to understand whether each high-level migration task, and then each migration stage, were sucessfully completed. To do this, consider how each task can be assessed for completion. Table 6-2, Table 6-3, and Table 6-4 provide examples of the questions that can be considered for assessing the completeness of typical migration tasks in each stage.

Table 6-2 shows a sample premigration planning checklist.

Table 6-2 Sample of the premigration planning checklist

Premigration tasks	Completion or assessment
Hardware and software planning	Are the prerequisites met for migration? Has the appropriate service been applied? Is the hardware available for installing the new version? Is the hardware available for backing up components and resources?

Premigration tasks	Completion or assessment
Human resources planning	Are the right personnel available and in place to perform the assigned tasks?
Are message set application resources ready for migration?	Are there any premigration steps that need to be taken to make existing message flows, , or custom code ready for migration? How are updates to be verified?
Assess impact on production systems	What strategies are to be used to minimize downtime, or retain availability during migration? What are the plans for recovery if problems occur? If downtime is required, when is the best time to schedule it?
Does a full system backup exist?	Migrating without a backup is inadvisable. Can it be verified that the backup was successful?

Table 6-3 shows a sample migration planning checklist.

Table 6-3 Example of the migration planning checklist

Migration tasks	Completion or assessment
Evaluate and choose a migration path	What is the requirement for availability of brokers during the migration? Can the migration be carried out in one step, or should the migration be staged? Which brokers are to be migrated first? Will components be moved to new machines/platforms as part of the migration?
Create a test plan for migration tasks	How will components be tested? What functionality will be covered by test plans? How will performance of the environment be assessed before and after migration? How will success be measured for the tests?
Resolving issues	Is time included for solving problems? What strategy will be adopted in the event of problems? How will problems be detected?

Table 6-4 shows a sample post-migration planning checklist.

Table 6-4 Example of the post-migration checklist

Post-migration tasks	Completion or assessment
Training and education	Are personnel trained on WebSphere Message Broker V6.0? Is any of the new or changed function going to have an immediate impact on the environment or business?
Has testing been completed to verify no loss or unexpected change of function after migration?	Which tests are used to verify this? Is more thorough testing required? What steps are needed if tests failed? Are only critical functions tested initially? Will new function be included in the testing? Will the testing be performed on each broker/each platform?

Post-migration tasks	Completion or assessment
Keep or remove the previous system?	After successful migration, will the previous system be removed? Are hard disk space requirements an issue? Can backups be retained for the future? Will some components be coexisted for a certain time or indefinitely after migration?

6.2 Coexistence with WebSphere Message Broker V6.0

WebSphere Message Broker V6.0 can coexist with either WebSphere MQ Integrator V2.1 or WebSphere Business Integration Message Broker V5.0. You do not have to uninstall the previous version for the purpose of migration. However, you must install WebSphere Message Broker V6.0 into a different location on the system to prevent overwriting of the previous installation.

Coexistence also makes it possible to install multiple versions of WebSphere Message Broker V6.0 runtime on the same machine. These could, for example, have different levels of service applied to them. This does not apply to Message Brokers Toolkit for WebSphere Message Broker V6.0, which can only be installed once on a machine.

Note: It is not possible to have both Version 2.1 and Version 5.0 on the same machine. Also, it is not possible to install several instances of either Version 2.1 or Version 5.0 on a single machine.

It is important to understand the advantages of coexistence in the planning of migration. Coexistence makes possible a controlled migration by allowing it to proceed in an incremental manner. One component is migrated, then tested before continuing with another component. This strategy allows comprehensive functional and performance testing to be conducted at each stage of the migration, therefore reducing the risk associated with a “big bang” approach (that is, migrating all of the components on a system or in an environment in one go). Decisions about which components to coexist determine the migration order of individual components.

The coexistence of WebSphere Message Broker V6.0 with previous versions provides flexibility in the order of migration of individual components. The setup of your environment determines the appropriate order for migrating components. In the scenarios described in this redbook, the components are migrated in the following order:

1. Message flow application resources and tooling
2. Configuration Manager
3. User Name Server

4. Brokers

All WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 brokers can participate in a WebSphere Message Broker V6.0 domain. Therefore, a domain can have its application resources and Configuration Manager migrated, while still retaining its brokers at their previous version. These brokers can then be administered by the Message Brokers Toolkit.

Once the whole domain has been migrated to WebSphere Message Broker V6.0 and verified, the previous version can be uninstalled. Brokers and other components may be retained at the previous version level for some time after the start of migration.

Certain restrictions determine which level of components can participate in a WebSphere Message Broker V6.0 domain.

6.2.1 Coexistence of WebSphere MQ Integrator V2.1 with WebSphere Message Broker V6.0

Table 6-5 shows that the broker is the only WebSphere MQ Integrator V2.1 component that can coexist or operate within a Message Broker V6.0 broker domain. You can administer WebSphere MQ Integrator V2.1 brokers from the Message Broker V6.0 toolkit.

Table 6-5 Coexistence of Integrator V2.1 and Message Broker V6.0 components

Coexistence in the same domain as Version 6.0?	Version 6.0 Toolkit	Version 6.0 Configuration Manager	Version 6.0 Broker	Version 6.0 User Name Server
Version 2.1 Control Center	No	No	No	No
Version 2.1 Configuration Manager	No	Not applicable	No	No
Version 2.1 Broker	Yes	Yes	Yes	Yes
Version 2.1 User Name Server	Not applicable	No	No	Not applicable

The Control Center and Configuration Manager must be migrated to WebSphere Message Broker V6.0 before the brokers can be administered by the Message Brokers Toolkit. If, for any reason, new Integrator V2.1 brokers are created, they

can be added to a WebSphere Message Broker V6.0 domain. Because multiple Configuration Managers can be created at WebSphere Message Broker V6.0, a new Configuration Manager can be created while Integrator V2.1 Configuration Manager exists.

Note: For the Message Broker V6.0 Configuration Manager to manage an Integrator V2.1 broker:

- WebSphere MQ Integrator V2.1 must be at least at fix pack 6.
- WebSphere MQ Integrator Broker V2.1 must be at least at fix pack 6.
- Integrator V2.1 brokers must use a different database schema if they share a database with Message Broker V6.0 broker; that is, they cannot use the same sets of database tables.
- Only function available at Integrator V2.1 can be deployed to an Integrator V2.1 broker; that is, new functionality introduced since Version 2.1 cannot be deployed.

6.2.2 Coexistence of WebSphere Business Integration Message Broker V5.0 with WebSphere Message Broker V6.0

Table 6-6 shows the ways that WebSphere Business Integration Message Broker V5.0 components can coexist or operate with WebSphere Message Broker V6.0 components.

Table 6-6 Coexistence of Message Broker V5.0 and Message Broker V6.0 components

Coexistence in the same domain as Version 6.0?	Version 6.0 Toolkit	Version 6.0 Configuration Manager	Version 6.0 Broker	Version 6.0 User Name Server
Version 5.0 or 5.1 Toolkit	Yes	Yes	Yes	Yes
Version 5.0 Configuration Manager	Yes	Not applicable	Yes	Yes
Version 5.0 Broker	Yes	Yes	Yes	Yes
Version 5.0 User Name Server	Not applicable	Yes	Yes	Not applicable

As Table 6-6 indicates, all WebSphere Business Integration Message Broker V5.0 components can coexist or operate within a Message Broker V6.0 broker domain. This means that the Message Broker V5.0 components can be migrated in any order, and can be administered in the version 6.0 Message Brokers Toolkit.

Note: The prerequisites for the Message Broker V6.0 Configuration Manager to manage a Message Broker V5.0 broker include:

- Message Broker V5.0 must be at least at fix pack 4.
- Message Broker V5.0 brokers must use a different database schema if they share a database with Message Broker V6.0 broker; that is they cannot use the same sets of database tables.
- Only function available at Message Broker V5.0 can be deployed to an Message Broker V5.0 broker; that is, new functionality introduced since Version 5.0 cannot be deployed.
- All the Message Broker V5.0 or 5.1 broker development resources can be used “as is” in the Message Broker V6.0 Toolkit with the exception of user-defined extensions and mapping files, which require further migration.
- Once resources are saved in the Message Broker V6.0 Toolkit, they can no longer be used in a Message Broker V5.0 or 5.1 Toolkit.

6.3 Planning migration from WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0

This section provides a high-level overview of the tasks required for the migration of Integrator V2.1 to WebSphere Message Broker V6.0. More detailed technical instructions can be found in Chapter 8, “WebSphere MQ Integrator V2.1 migration steps” on page 121. This includes information on the differences between migrating single and multiple domains, as well as changing the platform of components during the migration.

6.3.1 Premigration planning

This section provides a brief overview of the tasks that need to be completed before migration begins:

- Ensure that the environment is at the latest supported service level.
- Check the names of the message flows to be migrated.

Note: The migration process allows certain special characters when creating names for migrated message flows. You may experience problems with message flow names that contain characters other than alphabetic characters, numerics, and the underscore character.

This is because resources are being migrated to a file system and must adhere to the file system naming conventions. It is, therefore, necessary to rename these flows to adhere to these naming standards before attempting a migration; see 10.2.1, “Preparing message flows for migration” on page 146 for further details.

- Investigate any migration steps required for SupportPacs or third party software as appropriate.
- Confirm that the environment is working as expected using appropriate tests as defined in the test plan.
- Back up all product and user resources such as databases, file systems, and message flow application resources. For more information about backup procedures, refer to Chapter 9, “Backing up the WebSphere MQ Integrator V2.1 domain” on page 133.
- Export all Version 2.1 message flows, message sets, and user-defined nodes.
- Export these files into a directory other than the Integrator V2.1 install directory.

6.3.2 Migration planning

This section briefly describes the high-level steps during migration. Not all brokers need to be migrated at the same time. Some can be preserved at Integrator V2.1 level.

If a group of brokers share the same database schema, then all the brokers must be at the same broker version. That is, brokers sharing the same database schema cannot coexist at different version levels, and must all be migrated at the same time. To perform such a migration, after stopping all the brokers, one broker can be migrated using the **mqsimigratecomponents** command. The first broker’s migration helps in the migration of the shared database tables. Thus, the remaining brokers must then be migrated using the **mqsimigratecomponents-1** and **mqsimigratecomponents -2** flags to migrate their WebSphere MQ resources and file and registry components only.

Following are the high-level steps required for migrating to WebSphere Message Broker V6.0:

1. Stop and debug sessions in progress.
2. Ensure that no aggregations are in progress.
3. Install WebSphere Message Broker V6.0 in a different location on the systems where Integrator V2.1 components are to be migrated.
4. Migrate the tooling and import the message flow application resources into the Message Brokers Toolkit using the **mqsimigratemsgflows** and **mqsimigratemsgsets** commands.
5. Stop any components that are to be migrated.
6. Use the **mqsimigratecomponents** command with the `-c` option to perform a premigration check to determine whether a component can be safely migrated.
7. Use the **mqsimigratecomponents** command to migrate each individual component.
8. Start the migrated components.

Also, no redeploys should be necessary after a successful migration, but under certain circumstances they may be required. Part 2, “Migrating from WebSphere MQ Integrator V2.1” on page 105 addresses these issues.

6.3.3 Post-migration considerations

Here, we discuss the post-migration activities. After migration of Integrator V2.1, components and message flow application resources must be tested to ensure that there is no loss of functionality. They may also be checked for consistency and performance. The scenarios in this book use the following order for verifying the migration:

1. Test the migrated components.
2. Test the development environment.
3. Test the deployed resources.

Manual steps are required to fully migrate plug-in nodes and parsers

Refer to the following topics in the WebSphere Message Broker Information Center:

- Migrating a user-defined node under the topic **Reference** → **Migration and Upgrade** → **Message flow migration notes** → **Migrating a user-defined node,** on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/as10009_.htm

- Creating the user interface representation of a user-defined node in the workbench under **Developing applications → Developing user-defined extensions → Creating the user interface representation of a user-defined node in the workbench** on the Web at:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/as04480_.htm

Access Control List (ACL) entries must be created for Message Brokers Toolkit users on the new Configuration Manager. These allow the toolkit to connect to the domain and administer the brokers. For more information on ACLs, refer to 11.3.4, “Access Control Lists” on page 224.

The migration of the broker component from WebSphere MQ Event Broker does not retain the subscriptions and retained publications. Hence, after the migration, subscribers must renew their subscriptions and publishers must republish.

6.3.4 Planning for migrating a production environment

Here, we briefly discuss planning issues for migration in a production environment.

Before migrating a production environment, we recommend that you migrate the development and test domains first, to help identify any migration issues.

We recommend the following tasks before attempting a migration of a production environment:

1. Install Message Broker V6.0 toolkit, Configuration Manager, and broker, alongside the previous version in a development or test environment.
2. Migrate production broker application resources and test them on the WebSphere Message Broker V6.0 broker.
3. Simulate a production environment on the test environment and verify that the functionality is consistent, by comparing it to the premigration test results.
4. Migrate the simulated production broker from WebSphere MQ Integrator V2.1 and verify that the deployed resources continue to run as expected after migration.
5. Once confident that the test environment is working as expected, then migrate the production environment.

Refer to Chapter 8, “WebSphere MQ Integrator V2.1 migration steps” on page 121 for more details.

6.4 Planning migration from WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0

This section provides a high-level overview of the tasks required for the migration of a WebSphere Business Integration Message Broker V5.0 environment to WebSphere Message Broker V6.0. Detailed instructions are available in Chapter 18, “WebSphere Business Integration Message Broker V5.0 migration steps” on page 327. This includes information on the differences between migrating single and multiple domains.

A WebSphere Message Broker V6.0 broker can be migrated back to a WebSphere Business Integration Message Broker V5.0 broker using the **`mqsimigratecomponents`** command, if required, to restore an environment. However, a backup should still be taken of the components and data.

6.4.1 Premigration planning

This section provides a brief overview of the tasks that need to be completed before migration begins:

- Ensure that the environment is at the latest supported service level.

  Note: WebSphere Business Integration Message Broker V5.0 fix pack 4 is the minimum supported environment for migration to WebSphere Message Broker V6.0.

- Investigate any migration steps required for SupportPacs or third-party software, as appropriate.
- Confirm that the environment is working as expected using appropriate tests as defined in the test plan.
- Back up all product and user resources such as databases, file systems, and message flow application resources.

  For more information about backup procedures, refer to Chapter 19, “Backing up the WebSphere Business Integration Message Broker V5.0 domain” on page 339.

6.4.2 Migration planning

This section briefly describes the high-level steps during migration. Not all brokers need to be migrated at the same time. Some can be preserved at Message Broker V5.0 level.

If a group of brokers share the same database schema, then all the brokers must be at the same broker version. That is, brokers sharing the same database schema cannot coexist at different version levels, and must all be migrated at the same time. To perform such a migration, having stopped all the brokers, one broker can be migrated using the **mqsimigratecomponents** command. The first broker's migration will have already migrated the shared database tables, therefore the remaining brokers must then be migrated using the **mqsimigratecomponents -1** and **-2** flags to migrate their WebSphere MQ resources and file and registry components only.

These are the high-level steps required for migrating to WebSphere Message Broker V6.0:

1. Stop any debug sessions in progress.
2. Ensure that no aggregations are in progress.
3. Install WebSphere Message Broker V6.0 in a different location on the systems where Message Broker V5.0 components are to be migrated.
4. Migrate the Message Brokers Toolkit and message flow application resources.

   Except for message maps and user-defined nodes, there are no tasks to migrate development and deployment resources, such as message flow files, message set definition files, ESQL files, XML Schema files, and broker archive files. Use the **mqsimigratemfmaps** command to migrate any message maps. Information about migrating user-defined nodes can be found in 6.4.3, "Post-migration considerations" on page 102.
5. Import the Message Broker V5.0 message flow application resources into the Version 6.0 Message Brokers Toolkit.

   When an application resource is saved in the WebSphere Message Broker V6.0 Toolkit, it can no longer be used in either the WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 Toolkit. Therefore, ensure that all resources are backed up before migration begins.
6. Stop any components that are to be migrated.
7. Use the **mqsimigratecomponents** command with the `-c` option to perform a premigration check to determine whether a component can be safely migrated.

8. Use the **mqsimigratecomponents** command to migrate each individual component.
9. Start the migrated components.

No redeploys should be necessary after a successful migration, but under certain circumstances they may be required. These are highlighted in Part 3, “Migrating from WebSphere BI Message Broker V5.0” on page 311.

6.4.3 Post-migration considerations

This section offers high-level steps for planning for post-migration.

After migration of Message Broker V5.0, components and message flow application resources must be tested to ensure no loss of functionality. They may also be checked for consistency and performance. The scenarios in this book use the following order for verifying the migration:

1. Test the migrated components.
2. Test the development environment.
3. Test the deployed resources.

Manual steps are required to fully migrate the user-defined nodes, after migration. Information on how to perform these tasks can be found in the following topics:

- Migrating a user-defined node under **Reference** → **Migration and Upgrade** → **Message flow migration notes** → **Migrating a user-defined node** on the Web at:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/as10009_.htm

- Creating the user interface representation of a user-defined node in the workbench under **Developing applications** → **Developing user-defined extensions** → **Creating the user interface representation of a user-defined node in the workbench** on the Web at:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/as04480_.htm

Access control list entries must be created for Message Brokers Toolkit users on the new Configuration Manager. These allow the toolkit to connect to the domain and administer the brokers. For more information on ACLs, refer to 21.3.4, “Access Control Lists” on page 423 .

6.4.4 Planning for migrating a production environment

In this section we briefly discuss planning issues for migration of a production environment. It is possible to administer a Message Broker V5.0 broker domain from a Message Broker V6.0 toolkit and vice versa. This is useful for maintaining a mixed version environment during staged migration of a production environment.

Before migrating a production environment, we recommend that you migrate any development and test domains first to help identify any migration issues. The following tasks are recommended before attempting a migration of a production environment:

1. Install Message Broker V6.0 toolkit, Configuration Manager, and broker, alongside the previous version in a development or test environment.
2. Migrate production broker application resources and test them on the WebSphere Message Broker V6.0 broker.
3. Simulate a production environment on the test environment and verify that the functionality is consistent, by comparing it to the premigration test results.
4. Migrate the simulated production broker from WebSphere Business Integration Message Broker V5.0 and verify that the deployed resources continue to run as expected after migration.
5. Migrate the production environment when you are confident that the test environment is working as expected.

For more details, refer to Chapter 18, “WebSphere Business Integration Message Broker V5.0 migration steps” on page 327.

6.5 Changing platforms during migration

This section considers the planning required when changing the platform of components during migration.

There may be a number of reasons for changing the platform of individual components of your environment during migration such as, for example, to enable introduction of higher specification hardware, an upgrade to the operating system, or to take advantage of new platform support. For example, the Configuration Manager is now available on all supported platforms, and the Message Brokers Toolkit can be installed on Linux (x86 platform).

The migration scenarios in this book include examples of changing the Configuration Manager platform to z/OS and AIX, and of using the Message Brokers Toolkit platform on Linux (x86 platform).

6.5.1 Moving the Configuration Manager

In Message Broker V6.0, the Configuration Manager can exist on all supported runtime platforms. Also, there can be multiple Configuration Managers on the same system. Users may plan to migrate the existing Configuration Manager and move it to a different platform.

You can refer to the following sections for more information and an example:

- If you are an Integrator V2.1 user, refer to 11.2, “Changing the Configuration Manager platform to z/OS” on page 197.
- If you are a Message Broker V5.0 user, refer to 21.2, “Changing the Configuration Manager platform to Advanced Interactive Executive (AIX)” on page 403.

6.5.2 Using the Message Brokers Toolkit on Linux

In Message Broker V6.0, the toolkit can exist on Windows and Linux platforms. For more information on how to set up the toolkit on Linux, refer to 20.4.3, “Using the Linux (x86 platform) Message Brokers Toolkit” on page 380.

Part 2

Migrating from WebSphere MQ Integrator V2.1

The second part of the book covers the migration of all components of WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0. It illustrates the coexistence of Integrator V2.1 with Message Broker V6.0 and provides details about how to migrate the components to WebSphere Message Broker V6.0. It covers the migration steps specific to Integrator V2.1 using the business scenario established for the purposes of the redbook.

This part covers the following:

- Setting up the domain environment
- Backing up and restoring
- Migrating all components on various platforms
- Verifying migration

7

WebSphere MQ Integrator Version 2.1 environment setup

This chapter describes the environment used for the migration of WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0.

The following topics are discussed:

- The environment prior to the migration
- The logical and physical topology of the environment to be migrated
- The logical and physical topology of the environment and the changes to the environment following the migration
- The main differences between Integrator V2.1 and Message Broker V6.0 from the migration perspective

7.1 Assumptions during migration

- The installation and configuration of the WebSphere MQ Integrator V2.1 environment was completed according to the installation guide.
- The installation of WebSphere Message Broker V6.0 components has been completed according to the installation guide that comes with the product. You can find this guide on the Web at:

  http://www.elink.ibmlink.ibm.com/public/applications/publications/cgibin/pbi.cgi?CTY=GB&%2O%2OCTY=US&FNC=SRX&PBL=GC34-6621-OO&
- WebSphere MQ V5.3 was used on all platforms. The migration to WebSphere MQ V6.0 is not covered in this book. It can be completed after WebSphere Message Broker migration, as a separate step.
- DB2 Universal Database V8.1.7 was used both before and after the migration.
- The latest available fix pack versions were used for WebSphere MQ Integrator V2.1, WebSphere Message Broker V6.0, and WebSphere MQ V5.3.
- It was decided not to migrate one Windows broker. This demonstrates the coexistence with a previous version of the broker, and that a Message Broker V6.0 Configuration Manager can control a heterogeneous domain of brokers.

7.2 The environment before the migration

In this section we cover the Integrator V2.1 environment:

- The logical topology before the migration.
- The components that are planned to be migrated.
- The physical topology before the migration.

7.2.1 The logical topology

Figure 7-1 shows the logical topology of the Integrator V2.1 environment before the migration:

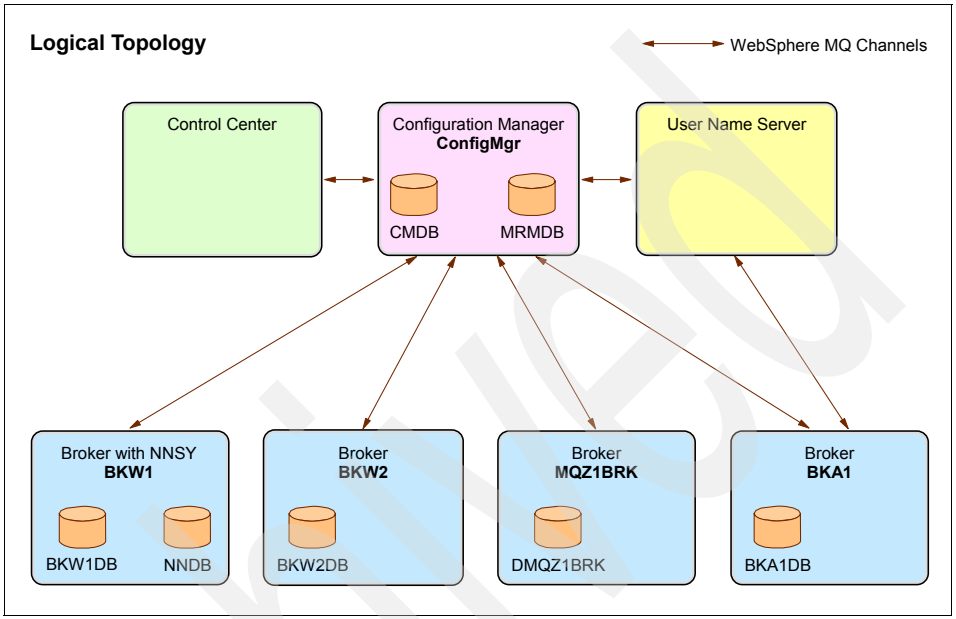


Figure 7-1 The logical topology of Integrator V2.1 scenario before the migration

The environment consists of one broker domain, with a Configuration Manager, a Control Center, a User Name Server and four brokers in the domain. All components and their required resources are listed in Figure 7-1.

Table 7-1 The domain components before the migration, V2.1 scenario

Component type	Component name	Queue manager	Database
Control Center	N/A	N/A	N/A
Configuration Manager	ConfigMgr	CMQM	CMDB, MRMDB
User Name Server	UserNameServer	UNSQM	N/A
Broker with NNSY	BKW1	BKW1QM	BKW1DB, NNDB
Broker	BKW2	BKW2QM	BKW2DB
Broker	BKA1	BKA1QM	BKA1DB[a]
Broker	MQZ1BRK	MQZ1	DMQZ1BRK

a. The TCP/IP loopback communication was used instead of shared memory to enable more than 10 sessions from the broker into the database at the same time.

Note: Each component uses its own queue manager in this scenario even if more than one component is on a single machine.

There are user databases for message flow interactions on each broker as well. However, this is not documented for reasons of simplicity in the table and figures. For the message flows used in this scenario, refer to “Message flows for WebSphere MQ Integrator V2.1 scenario” on page 526.

Components to be migrated

All components are going to be migrated in this scenario, except the broker BKW2, which remains at Integrator V2.1 to demonstrate the coexistence of the two versions of the products in an environment.

7.2.2 The physical topology

Figure 7-2 shows the physical topology of the Integrator V2.1 environment before the migration.

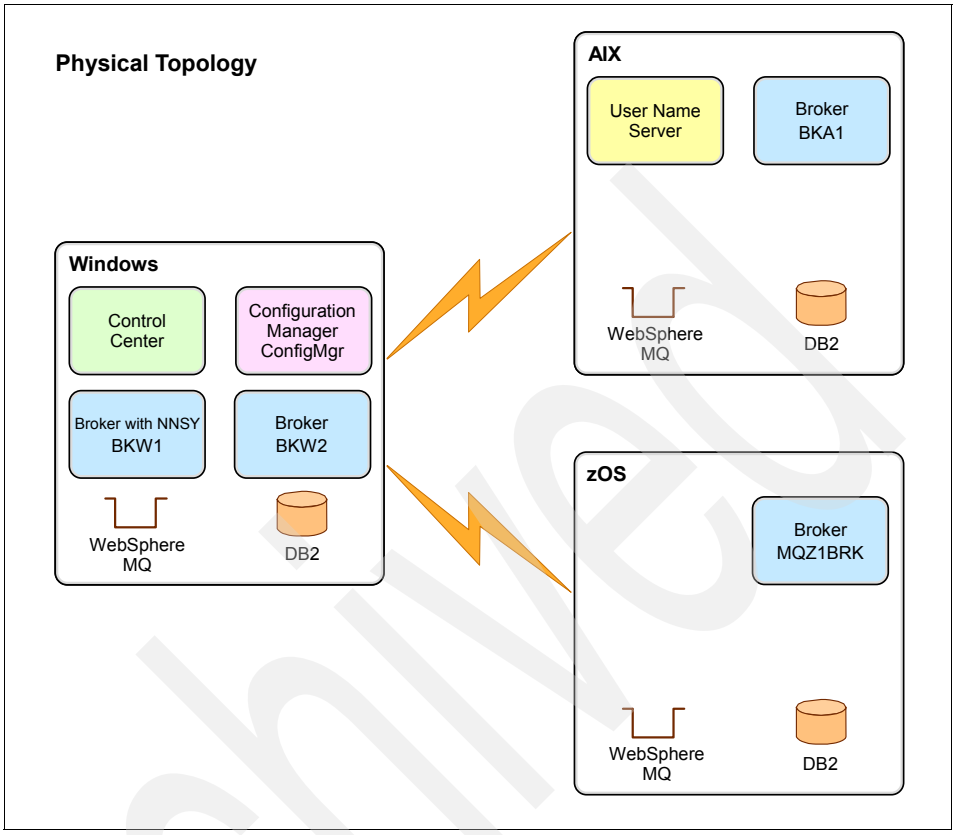


Figure 7-2 The physical topology of the Integrator V2.1 scenario before the migration

The physical environment consists of three machines that run Windows, AIX, and z/OS operating systems. Their details are as follows:

- A Control Center, a Configuration Manager, and two brokers on the Windows machine.
- A User Name Server and one broker on the AIX machine.
- One broker on the z/OS machine.

Note: Network communication is not needed between the AIX machine and the z/OS machine, because there is no interaction between the components on these machines.

All software products used, and their versions, are listed in Table 7.2, Table 7.3, and Table 7.4.

Table 7.2 shows the Windows machine before the migration in the V2.1 scenario.

Table 7-2 The Windows machine before the migration, V2.1 scenario

Software	Installed level	Minimum level for migration
Operating system	Windows 2000 Service Pack 4[4]	Windows 2000 Service Pack 2[5]
WebSphere MQ Integrator	V2.1 fix pack 8	V2.1 fix pack 6
WebSphere MQ	V5.3 fix pack 11	V5.3 fix pack 1[1]
DB2®	V8.1 fix pack 7	V8.2[2,3]
RAC	N/A	N/A
JRE	V1.4.2 (only for RAC)	V1.4.1 (only for RAC)

Table 7-3 shows the AIX machine before the migration in the V2.1 scenario.

Table 7-3 The AIX machine before the migration, V2.1 scenario

Software	Installed level	Minimum level for migration
Operating system	V5.2 Maintenance Level 07	V5.2 Maintenance Level 03
WebSphere MQ Integrator	V2.1 fix pack 8	V2.1 fix pack 6
WebSphere MQ	V5.3 fix pack 11	V5.3 fix pack 1[1]
DB2	V8.1 fix pack 7	V8.2[2]
RAC	N/A	N/A
JRE	V1.4.2 (1.4.2.10, only for RAC)	V1.4.1 (only for RAC)

Table 7-4 shows the z/OS machine before the migration in the V2.1 scenario.

Table 7-4 The z/OS machine before the migration, V2.1 scenario

Software	Installed level	Minimum level for migration
Operating system	V1R6 RSU0508	V1R5 RSU0507 plus PTF for OA11699
WebSphere MQ Integrator	V2.1 RSU0508	V2.1 fix pack 6

Software	Installed level	Minimum level for migration
WebSphere MQ	V5.3.1 RSU0508	V5.3.1 plus PTF for PQ80677
DB2	V8.1 RSU0508	V8.1 plus PTF for PQ84976
RAC	N/A	N/A
JRE	V1.4.2 SR3	V1.4.2 SR2

Table notes:

1. WebSphere MQ V5.3 fix pack 10 is required for real-time support and JMS MQ Optimized node.

   If you develop message flow that uses WebSphere MQ Real-time Transport with Multicase PGM support, you must install WebSphere MQ V6.0 or later on each affected broker system.

   For more information, refer to the WebSphere Message Broker Information Center, under the topics **Reference** → **Installation** → **Software requirements** → **Additional required products** on the Web at:

   http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah09088_.htm

2. DB2 Universal Database V8.2 is equivalent to DB2 Universal Database V8.1 with fix pack 7.
3. XA coordination for messages in the MRM domain on Windows requires DB2 V8.2 fix pack 10 or later.
4. Windows 2000 is no longer a supported environment with WebSphere Message Broker V6.0.
5. This is the minimum supported level for WebSphere MQ Integrator V2.1; see Note 4.

Products to be migrated

The only component selected for migration in this scenario is WebSphere MQ Integrator V2.1. All other components, including the operating system, remain at the same level.

The Configuration Manager is going to be moved from the Windows machine to the z/OS machine.

7.3 The environment after the migration

This section describes the WebSphere Message Broker environment after migration.

- The logical topology after migration
- The physical topology after migration
- The changes after migration

7.3.1 The logical topology

Figure 7-3 shows the logical topology of the Message Broker V6.0 environment after the migration process demonstrated in the rest of this part of the book.

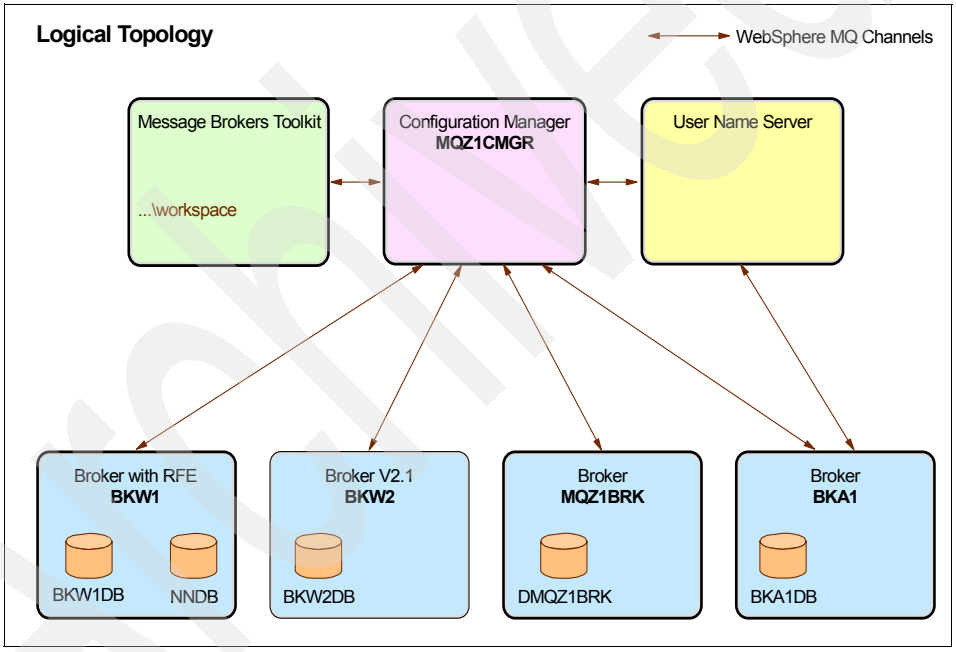


Figure 7-3 The logical topology of Integrator V2.1 scenario after the migration

The environment after the migration still contains a single broker domain. All components and their resources after the migration are listed in Table 7-5.

Table 7-5 The domain components after the migration, V2.1 scenario

Component type	Component name	Queue manager	Database
Message Brokers Toolkit	N/A	N/A	N/A
Configuration Manager	MQZ1CMGR	N/A	N/A
User Name Server	UserNameServer	UNSQM	N/A
Broker with RFE	BKW1	BKW1QM	BKW1DB, NNDB
Broker at V2.1 level	BKW2	BKW2QM	BKW2DB
Broker	BKA1	BKA1QM	BKA1DB
Broker	MQZ1BRK	MQZ1	DMQZ1BRK

Changes after the migration

The changes can be seen in Figure 7-3 on page 114 and Table 7-5:

- The graphical user interface was changed from the Control Center to the Message Brokers Toolkit.
- The DB2 Universal Database for the Configuration Manager was removed because it is no longer needed for Message Broker V6.0, which uses an internal repository.
- Broker BKW2 remains at Integrator V2.1 to demonstrate the coexistence of the two versions of the products in a heterogeneous domain.

7.3.2 The physical topology

Figure 7-4 shows the physical topology of the Message Broker V6.0 environment after the migration process demonstrated in the rest of this part of the book.

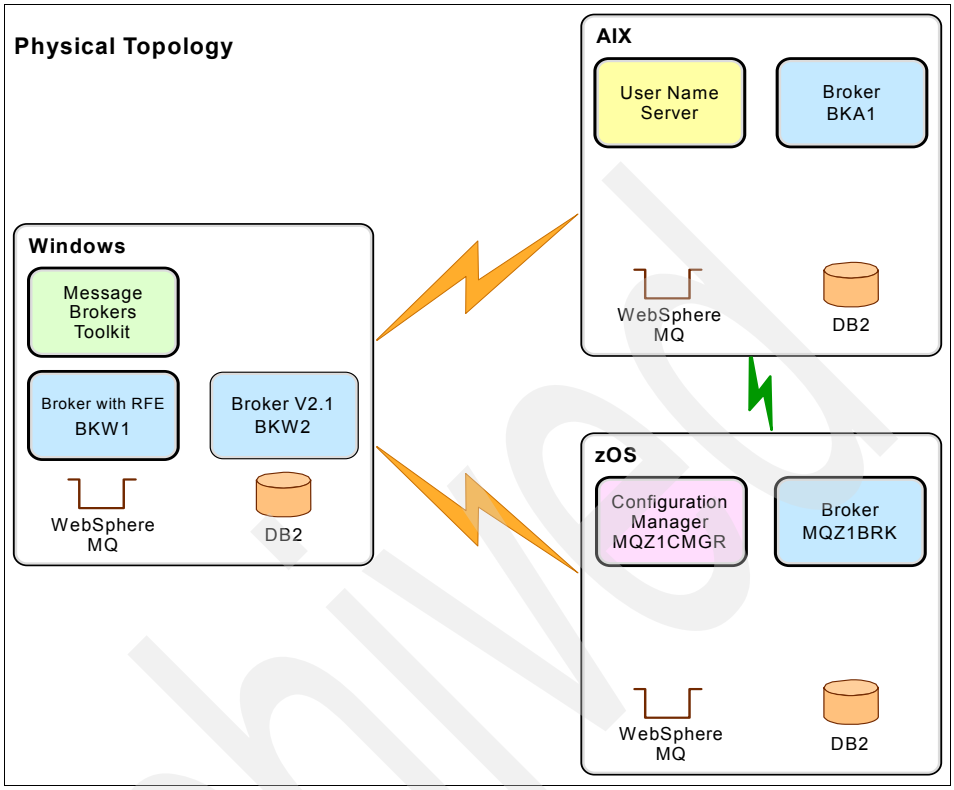


Figure 7-4 The physical topology of the Integrator V2.1 scenario after the migration

The physical environment after the migration still contains three machines that run the Windows, AIX, and z/OS operating systems. All software products and their versions after the migration are listed in Table 7-6 through Table 7-8. Table 7-6 shows the Windows machine after the migration in the V2.1 scenario.

Table 7-6 The Windows machine after the migration, V2.1 scenario

Software	Installed level
Operating system	Windows 2000 Service Pack 4[1]
WebSphere MQ Integrator	V2.1 fix pack 8
WebSphere Message Broker with Rules and Formatter Extension	V6.0.0.0
WebSphere MQ	V5.3 fix pack 11
DB2	V8.1 fix pack 7 (which is equivalent to V8.2)

Software	Installed level
RAC	V6.0.0.1
JRE	V1.4.2 (only for RAC)

Table note:

1. Windows 2000 is no longer a supported environment with WebSphere Message Broker V6.0. Following a migration to WebSphere Message Broker V6.0 the operating system should be upgraded to a supported version. See the information about supported environments for WebSphere Message Broker V6.0 at:

   http://www.ibm.com/software/integration/wbimessagebroker/requirements/index.html

   This upgrade is beyond the scope of this book.

Table 7-7 shows the AIX machine after the migration in the V2.1 scenario.

Table 7-7 The AIX machine after the migration, V2.1 scenario

Software	Installed level
Operating system	V5.2 Maintenance Level 07
WebSphere Message Broker	V6.0.0.0
WebSphere MQ	V5.3 fix pack 11
DB2	V8.1 fix pack 7 (which is equivalent to V8.2)
RAC	V6.0.0.1
JRE	V1.4.2 (1.4.2.10, only for RAC)

Table 7-8 shows the z/OS machine after the migration in the V2.1 scenario.

Table 7-8 The z/OS machine after the migration, V2.1 scenario

Software	Installed level
Operating system	V1R6 RSU0508
WebSphere Message Broker	V6.0 RSU0508
WebSphere MQ	V5.3.1 RSU0508
DB2	V8.1 RSU0508
RAC	V6.0.0.1

Software	Installed level
JRE	V1.4.2 SR3
XML Toolkit	V1.8

7.4 Changes after the migration

The changes can be seen in the previous figures and tables:

- WebSphere Message Broker V6.0 and Rational Agent Controller V6.0.0.1 were installed on all three machines.
- The XML Toolkit V1.8 was installed on the z/OS machine.
- The Configuration Manager was moved from the Windows machine to the z/OS system and the WebSphere MQ environment was configured with new channels and transmission queues, as appropriate.

7.4.1 The differences between WebSphere MQ Integrator V2.1 and WebSphere Message Broker V6.0

This section discusses the differences between Integrator V2.1 and Message Broker V6.0 from the migration perspective.

Note: Refer to Chapter 3., "What is new for WebSphere MQ Integrator V2.1 users" on page 27 and/or the WebSphere Message Broker 6.0 Information Center for further details about the differences.

One of the differences is the replacement of the Control Center with the Message Brokers Toolkit:

- The Control Center is a standalone Java application. Message Brokers Toolkit is built on the Rational Application Developer Platform.
- Resources in the Message Brokers Toolkit are stored in the local file system and are no longer stored in a repository in the Configuration Manager. Broker application resources and other files are stored in the local file system in a directory called the *workspace*.
- Broker application development can be done in the Message Brokers Toolkit without a connection to a Configuration Manager.
- There is no "check in/out" mechanism for Message Brokers Toolkit.
- The broker archive file (BAR) is used for deploying resources to execution groups.

Tip: A single broker archive file can be transferred from development environment to test and production environments for deployment.

- The Message Brokers Toolkit can now be installed on Linux (x86 platform).

Other differences include:

- Access Control List (ACL) entries are used for domain security and roles instead of user groups (*mqbrasgn, mqbrdevt, mqbrops, mqbrtpic,* and *mqbrkrs*).
- There are changes in console commands: new options for existing commands, and new commands. For more details refer to the WebSphere Message Broker 6.0 Information Center under **Reference** → **Operations** → **Commands** on the Web at:

  `http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an07060_.htm`

Tip: The commands **`mqsistartmsgflow`** and **`mqsistopmsgflow`** can be used for starting and stopping message flows as an alternative to using the Message Brokers Toolkit.

- The MRM repository database is not used in Message Broker V6.0. The Message Brokers Toolkit workspace is used instead for this purpose.
- The Configuration Manager now uses its own internal repository, so the DB2 Universal Database is no longer needed.

Tip: The command **`mqsibackupconfigmgr`** can be used to back up a Configuration Manager.

- The Configuration Manager can, now, be created on all the supported platforms: AIX, HP-UX, Linux (x86 platform), Linux (zSeries platform), Solaris, Windows, and z/OS.
- The Configuration Manager can be administered by the Configuration Manager Proxy (CMP) application programming interface (API). This is a Java API for the Configuration Manager. It enables programs to be written to administer broker domains as an alternative to the Message Brokers Toolkit.

Tip: A Configuration Manager Proxy API Exerciser sample application is supplied to demonstrate the capabilities of the Configuration Manager Proxy (CMP) API in a graphical use interface.

- Multiple named Configuration Managers can be created on the same system in Message Broker V6.0.

  Restriction: A broker can still only be managed by a single Configuration Manager, but multiple domains can be managed from a single system.

- The Rational Agent Controller (RAC) is used with the Message Brokers Toolkit flow debugger. This application must be installed on systems containing brokers in order to be able to use the debugger with these brokers.

  Note: This is an optional piece of software that is only required for debugging when using the flow debugger in the Message Brokers Toolkit.

8

WebSphere MQ Integrator V2.1 migration steps

This chapter discusses high-level steps for migrating a WebSphere MQ Integrator V2.1 domain to WebSphere Message Broker V6.0.

The order in which WebSphere MQ Integrator V2.1 components are migrated to WebSphere Message Broker V6.0 is very important. This chapter covers the following scenarios:

- The migration of a single domain
- The migration of multiple domains (development, test, and production)

8.1 Premigration planning

Before beginning the migration, the following should be read:

- Chapter 5, “Migration considerations” on page 75
- Chapter 6, “Planning for migration” on page 87

These chapters cover the necessary premigration steps, which ensures the following:

- Minimum software levels for the current WebSphere MQ Integrator are correctly configured.
- WebSphere Message Broker V6.0 prerequisite software and hardware levels are correct.
- New migration features such as coexistence are understood.

Understanding these chapters helps ensure that your WebSphere MQ Integrator V2.1 domain is correctly configured for migration.

8.2 Migrating a single domain

When migrating a broker domain, the order of migration is important. Only the WebSphere MQ Integrator V2.1 broker component can participate in a WebSphere Message Broker V6.0 domain. Therefore, the migration of the Control Center, Configuration Manager, and User Name Server must occur before the migration of the broker.

The following section explains the migration of a single WebSphere MQ Integrator domain to WebSphere Message Broker V6.0. For information about migrating multiple domains, for example a development, test, and production environment, see 8.3, “Migrating multiple domains” on page 124.

8.2.1 Back up the domain

Before any components are migrated, the first task is to back up the current domain. Chapter 9., “Backing up the WebSphere MQ Integrator V2.1 domain” on page 133 explains how the backup of the current domain should be performed.

Important: Performing a backup of the domain allows you to restore your current WebSphere MQ Integrator V2.1 environment if any problems are encountered during migration.

8.2.2 Migrate the Control Center and Configuration Manager

After a backup of the WebSphere MQ Integrator domain has been successfully performed, begin the migration process.

The Control Center and the Configuration Manager are migrated first. The WebSphere MQ Integrator V2.1 Control Center and Configuration Manager are not interoperable with WebSphere Message Broker V6.0. Therefore, the WebSphere MQ Integrator Control Center and Configuration Manager must be migrated at the same time.

1. Migrate the Control Center and resources

   The first stage in the migration process is to migrate the Control Center and any resources such as message flows, message sets, and user-defined nodes. Chapter 10, “Migrating the WebSphere MQ Integrator V2.1 Control Center resources” on page 145 discusses the steps required to perform this migration, which is done before the migration of the Configuration Manager to ensure all resources migrate successfully.

2. Migrate the Configuration Manager

   Once the Control Center, message flows, and message sets have been successfully migrated, the Configuration Manager can be migrated. Chapter 11, “Migrating the Configuration Manager” on page 177 discusses the details of how the Configuration Manager can be migrated.

8.2.3 Migrate the User Name Server

Before any brokers are migrated, the User Name Server must be migrated. User Name Server migration is discussed in Chapter 12, “Migrating the User Name Server” on page 229.

The WebSphere Message Broker V6.0 User Name Server can support both WebSphere MQ Integrator V2.1 and WebSphere Message Broker V6.0 brokers. However, the WebSphere MQ Integrator V2.1 User Name Server cannot support the same. Therefore, the User Name Server must be migrated first to ensure it can communicate with all brokers in the domain.

8.2.4 Migrate the brokers

Once all other WebSphere MQ Integrator V2.1 components have been migrated, the migration of the brokers can begin. This task is discussed in the following chapters:

- Chapter 13, “Migrating the Windows Broker” on page 239
- Chapter 14, “Migrating the AIX Broker” on page 253

- Chapter 15, “Migrating the z/OS Broker” on page 267

These chapters cover the steps required to migrate a WebSphere MQ Integrator V2.1 broker to WebSphere Message Broker V6.0 on the stated platform.

The broker migration chapters also cover the migration of New Era Of Networks components.

8.2.5 Ensure a successful migration

Once a broker has been migrated to WebSphere Message Broker V6.0, the steps discussed in Chapter 16, “Migration verification” on page 287 must be followed to ensure that the WebSphere Message Broker V6.0 migration has been successful.

8.2.6 Single domain migration checklist

Table 8-1 is a checklist summarizing the migration steps for a single domain.

Table 8-1 Checklist for migrating a single broker domain

Step	Completed
1. Read the considerations and planning chapters.	
2. Prepare the domain for migration.	
3. Backup the current domain.	
4. Migrate the Control Center and Resources.	
5. Migrate the Configuration Manager.	
6. Migrate the User Name Server (if present).	
7. Take a current backup of the brokers, so that any changes are backed up.	
8. Migrate the brokers.	
9. Check for successful migration.	

8.3 Migrating multiple domains

A high-level overview for migrating a single broker domain is given in 8.2, “Migrating a single domain” on page 122. However, in an organization there may be multiple domains that require migration. In this section, three domains are migrated: development, test, and production.

Before the migration of multiple domains begins, make sure that 8.1, “Premigration planning” on page 122 has been read and understood.

During the migration to WebSphere Message Broker V6.0, it may be essential to keep the production brokers operational. Therefore, the development and test domains can run WebSphere MQ Integrator V2.1 and WebSphere Message Broker V6.0 in parallel. This allows any critical WebSphere MQ Integrator V2.1 changes to be made, tested, and deployed to production while the migration is in progress.

We recommend that application code be frozen during the migration. This reduces the number of steps required to migrate to WebSphere Message Broker V6.0. However, it is not always possible to freeze code changes for this length of time. If changes must be made to WebSphere MQ Integrator V2.1 code during the migration, they must also be migrated to WebSphere Message Broker V6.0. This can be done on a daily or weekly basis as the migration is in progress.

8.3.1 Coexistence in the development domain

In the development domain, developers create message flows and unit test them in a sandbox environment. The brokers in the domain are not responsible for handling business-critical data.

Here, WebSphere Message Broker V6.0 should be installed alongside WebSphere MQ Integrator V2.1. Both should be run in parallel until the production domain has been successfully migrated. This allows developers to run both versions of the tooling on their machines and to migrate their message flows and message sets to WebSphere Message Broker V6.0.

New WebSphere Message Broker V6.0 brokers and a new Configuration Manager should be created alongside their WebSphere MQ Integrator V2.1 counterparts. These new components should form a new broker domain and should not interoperate with the existing WebSphere MQ Integrator V2.1 domain.

Developers should use this new WebSphere Message Broker V6.0 environment to unit test their migrated message flows and message sets.

Note: Use care when brokers coexist on a single machine. If applications running in the broker are not designed to share resources, such as database tables, then they may encounter problems when running in parallel.

Once the resources have been successfully migrated and unit tested, they should be deployed to the test domain.

Important: No new WebSphere Message Broker V6.0 functionality, such as new nodes in message flows, should be used until all of the production domain brokers have been fully migrated.

8.3.2 Coexistence in the test domain

In the test domain, message flows from the development domain are tested against test data in a realistic broker configuration.

In a similar manner to the development domain, the test domain should exploit the coexistence feature of WebSphere Message Broker V6.0 to allow multiple brokers and Configuration Managers on a single computer.

The WebSphere MQ Integrator brokers and the WebSphere Message Broker brokers should be kept in separate domains and administered from different Configuration Managers. This parallel approach allows the WebSphere MQ Integrator components to be easily uninstalled once WebSphere Message Broker V6.0 has been successfully configured in the production environment.

Once the migrated application has been successfully tested, the migration of the production environment can begin.

8.3.3 Migrate the production domain

When message flows in the test domain are deemed to be robust enough, they are promoted to the production domain. This is the domain responsible for performing actual business transactions, and message flows in the domain work with live data.

The development and test environments have verified that the migrated applications work correctly on the new WebSphere Message Broker V6.0 brokers. Therefore, a staged migration of the production domain can begin.

Control Center and Configuration Manager

The first task in migrating the production domain is to migrate both the Configuration Manager and Control Center to WebSphere Message Broker V6.0. Before this migration begins, back up your Configuration Manager and Control Center resources. This process is described in Chapter 9, “Backing up the WebSphere MQ Integrator V2.1 domain” on page 133.

The Control Center and Configuration Manager components must be migrated to WebSphere Message Broker V6.0 at the same time.

These components are migrated first because WebSphere Message Broker V6.0 Configuration Manager is able to administer WebSphere MQ Integrator V2.1 brokers that run at fix pack 6 or later. Therefore, the brokers can remain at the WebSphere MQ Integrator V2.1 code level while the Control Center and Configuration Manager are migrated. Further information can be found in the online documentation:

You can find further information in the WebSphere Message Broker Information Center, under the topics **Migrating** → **Coexistence** → **Conditions** for a Version 2.1 broker participating in a Version 6 broker domain, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13700_.htm

User Name Server

Once the Control Center and Configuration Manager have been successfully migrated, the User Name Server must be migrated. The WebSphere MQ Integrator V2.1 User Name Server cannot communicate with WebSphere Message Broker V6.0 brokers. Therefore, it is migrated before the brokers' turn.

If the User Name Server is not used in the broker domain, ignore this step.

Brokers

Now that the production domain brokers are being controlled by the WebSphere Message Broker V6.0 and Configuration Manager, it is possible to migrate the brokers. These should be migrated one at a time to ensure that existing WebSphere MQ Integrator V2.1 brokers are still running to process business transactions. However, brokers which share a database schema must be migrated simultaneously. Further information about this can be found in the online documentation at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23926_.htm

under the topics **Migrating** → **Migrating from Version 2.1 products** → **Migrating from WebSphere MQ Integrator Broker Version 2.1** → **Migrating components on different computers** → **Migrating the broker**.

Once the first broker has been successfully migrated and has run for a period of time without error, the next broker can be migrated. Migrate the brokers one at a time until all brokers in the production domain are running at the WebSphere Message Broker V6.0 level.

The message flows and message sets that are running in the WebSphere MQ Integrator V2.1 broker continue to run after the broker has been migrated. Currently at WebSphere Message Broker V6.0.0.0, some message sets may

require redeployment after migration. The output of the **mqsimigratecomponents** command should be checked thoroughly to see if any message flows or message sets require redeployment.

Deploy migrated resources

When all brokers have been successfully migrated and have been running for a period of time, it is possible to deploy the migrated message flows and message sets to them. Note that this is not required if no functional changes have occurred in the applications.

Use the Message Brokers Toolkit to deploy the migrated resources to a single broker. Once the migrated applications have been running successfully for a period of time, deploy the migrated applications to the remaining brokers one at a time.

At this point it is possible for the developers to begin using new WebSphere Message Broker V6.0 functionality in their message flows because all production brokers are running at WebSphere Message Broker V6.0 and are able to support it.

Important: Any change to applications, including the use of new WebSphere Message Broker V6.0 nodes, should be tested in the development and test domains before deployment to production.

Uninstall WebSphere MQ Integrator V2.1

Once the production domain has been running at WebSphere Message Broker V6.0 for a period of time, the WebSphere MQ Integrator components in the development and test environments can be uninstalled.

8.3.4 Multiple domain migration checklists

The following steps are required to migrate a development, test, and production environments. This is described in detail in section 8.3, “Migrating multiple domains” on page 124.

Table 8-2 illustrates the migration of a development domain.

Table 8-2 Checklist for migrating a development broker domain

Development Steps	Domain	Completed
1. Read the considerations and planning chapters	Test/Dev/Prod	

Development Steps	Domain	Completed
2. Install V6 Message Brokers Toolkit along side V2.1 Control Center, exploiting coexistence.	Dev	
3. Migrate the V2.1 resources to V6.	Dev	
4. Install V6 Configuration Manager and brokers alongside V2.1 counterparts.	Dev	
5. Create V6 brokers and Configuration Managers.	Dev	
6. Connect the Message Brokers Toolkit to the V6 Configuration Manager.	Dev	
7. Add the V6 brokers to a new domain and deploy the migrated resources to test them.	Dev	

Table 8-3 illustrates the migration of a test domain.

Table 8-3 Checklist for migrating a test broker domain

Test Steps	Domain	Completed
1. Install V6 Message Brokers Toolkit alongside V2.1 Control Center, exploiting coexistence.	Test	
2. Import the migrated V6 resources into the Message Brokers Toolkit.	Test	
3. Install V6 Configuration Manager and brokers alongside V2.1 counterparts.	Test	
4. Create V6 brokers and Configuration Managers.	Test	
5. Connect the Message Brokers Toolkit to the V6 Configuration Manager.	Test	
6. Add the V6 brokers to a new domain and deploy the migrated resources to test them.	Test	

Likewise, Table 8-4 illustrates the migration of a production domain.

Table 8-4 Checklist for migrating a production broker domain

Production Steps	Domain	Completed
1. Install V6 on the Configuration Manager and Control Center machines.	Prod	
2. Backup the Configuration Manager	Prod	

Production Steps	Domain	Completed
3. Migrate the Configuration Manager and Control Center to V6.	Prod	
4. Import the migrated resources into the Message Brokers Toolkit.	Prod	
5. Migrate the User Name Server to WebSphere Message Broker V6.0	Prod	
6. Backup your brokers	Prod	
7. Migrate a single broker to WebSphere Message Broker V6.0, leaving the others running.	Prod	
8. Leave the broker running for a period of time to ensure the migration is successful.	Prod	
9. When successful, backup and migrate the remaining brokers, one at a time	Prod	
10. When all brokers in the domain are stable, Connect to the Configuration Manager and deploy the migrated resources to a single broker.	Prod	
11. Run the migrated resources in the broker for a period of time to ensure they run successfully.	Prod	
12. Deploy the migrated resources to the remaining brokers, one at a time.	Prod	

If any problems are encountered during this migration it is possible to use the backups for restoring components back to their WebSphere MQ Integrator V2.1 level. See 16.2, “Restoring the WebSphere MQ Integrator Version 2.1 environment” on page 301 for more information.

8.4 Best practices for a successful migration

The following best practices provide some guidance to reduce the possibility of encountering problems or suffering an outage while migrating.

Redundant brokers for high availability

If the domain to be migrated contains a single broker, then the migration process impacts the availability of the broker because it must be stopped. Therefore, an ideal broker domain has a minimum of three brokers processing workload. This

allows one broker to be migrated while the remaining two provide failover for each other.

Test migration

By exploiting new features of WebSphere Message Broker V6.0 it is possible to test the migration of a component using the command:

```
mqsimigratecomponents -c
```

This command ensures that the component migrates successfully when the actual command is run. The use of this command is discussed in the broker migration chapters.

Once the migration of a component has been completed, the component should be left to run at the new WebSphere Message Broker V6.0 level for a period of time. By running the component for a short while, any problems can be resolved before migrating the next component.

Stepwise approach

Performing migration steps in parallel is highly discouraged. If a problem occurs, it is extremely difficult to know which new component caused the problem.

Therefore, the process described in this chapter emphasizes a step-wise approach. That is, migrating one component after another. Also, old WebSphere MQ Integrator V2.1 components should be removed only when the new WebSphere Message Broker V6.0 components are functioning correctly.

Migrating message sets

When migrating a WebSphere MQ Integrator V2.1 broker to WebSphere Message Broker V6.0, you may need to redeploy your message sets.

The **`mqsimigratecomponents`** command notifies you if any resources require redeployment. Check the output of this command.

9

Backing up the WebSphere MQ Integrator V2.1 domain

This chapter describes basic backup strategies for the WebSphere MQ Integrator Broker V2.1 domain including Control Center resources, components, and databases before starting the migration to WebSphere Message Broker V6.0.

The purpose of backup is to ensure business continuity, high availability, and planning disaster recovery. A simple example is illustrated in this chapter. Many elements must be considered because these are environment-specific and depend on your individual installation. It is beyond the scope of this redbook to cover every backup requirement.

9.1 Backup overview

This section explains the backup tasks that are recommended before migrating to WebSphere Message Broker V6.0. These are:

- Backing up databases
  - Broker databases
  - Configuration Manager database
  - Message Repository Manager database
  - New Era Of Networks database
- Backing up Universally Unique ID (UUID)
- Backing up Control Center resources
  - Workspace
  - Topics
  - User-defined nodes/parsers

9.2 Database backup

Before performing backups of Configuration Manager, Message Repository Manager, Broker, and New Era Of Networks repositories, ensure that all topology components, for example message flows and message sets, are *checked in* by browsing through the Control Center. Select the **File** → **Check in all** option.

This section only covers the steps for backing up WebSphere MQ Integrator components that use DB2 Universal Database. Other databases are outside of the scope this book.

Perform DB2 backup of Configuration Manager, Message Repository Manager, New Era Of Networks, and the Broker databases:

1. Choose a safe storage location, like an external disk drive or DVD, to store the backup information.

   For z/OS, create a sequential non-VSAM data set.

   For Windows or UNIX, create a directory or file system.

2. As illustrated in Figure 9-1, check for applications with active database connections by issuing the following command in the DB2 command window:

   ```
   DB2 list applications global
   ```

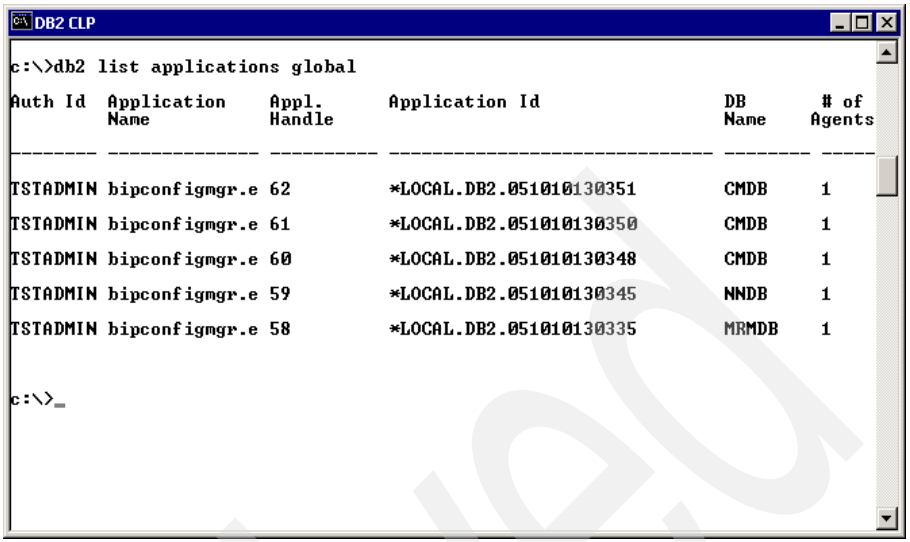


Figure 9-1 Check active DB2 applications

3. Stop all WebSphere MQ Integrator components on the machine where you are performing the backup by entering the following command on Windows or UNIX:

   `mqsistop <broker_name>`

   Follow it by entering the command from SDSF on z/OS:

   `/P <broker_name>`

   For example:

   `mqsistop BKW1`

   For stopping the User Name Server, enter the command on Windows or UNIX:

   `mqsistop UserNameServer`

   and follow it by entering the comand from SDSF on z/OS:

   `/P UserNameServer`

   For stopping the Configuration Manager, enter the command on Windows:

   `mqsistop ConfigMgr`

   Ensure that the components are stopped by checking the operating system application user/error log.

Important: After stopping all domain components, back up every component in stages. Do not restart any component until after they have all been backed up. This avoids inconsistency between broker and configuration manager backups.

4. As illustrated in Figure 9-2, check whether all applications are stopped by issuing the following command in the DB2 command window:

   ```
   DB2 list applications global
   ```

   If there are still applications running, issue the following command in the DB2 command window:

   ```
   DB2 force application all
   ```

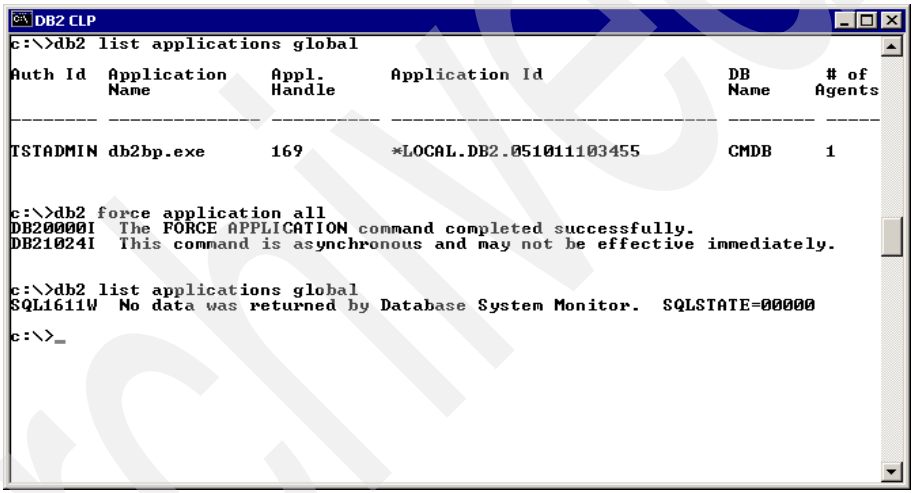


Figure 9-2 Force DB2 applications

5. Back up the Configuration Manager and Message Repository Manager databases on a Windows machine by using the DB2 command window:

   ```
   DB2 connect to <database_name>
   DB2 quiesce database immediate force connections
   DB2 connect reset
   DB2 backup database <database_name> to <directory> with 2 buffers
   buffer 1024 parallelism 1 without prompting
   DB2 connect to <database_name>
   DB2 unquiesce database
   DB2 connect reset
   ```

Attention: Make a note of the time stamp displayed after successful completion of a backup, as illustrated in Figure 9-3.

The reason for this is as follows: The time stamp is unique for each backup, and is used as a parameter for the command to restore the database. If a value for the time stamp is not known when a restore is performed, then only one backup image must be available on the source media. So noting the time stamp ensures that the correct backup is used in a restore operation, if required at a later stage.

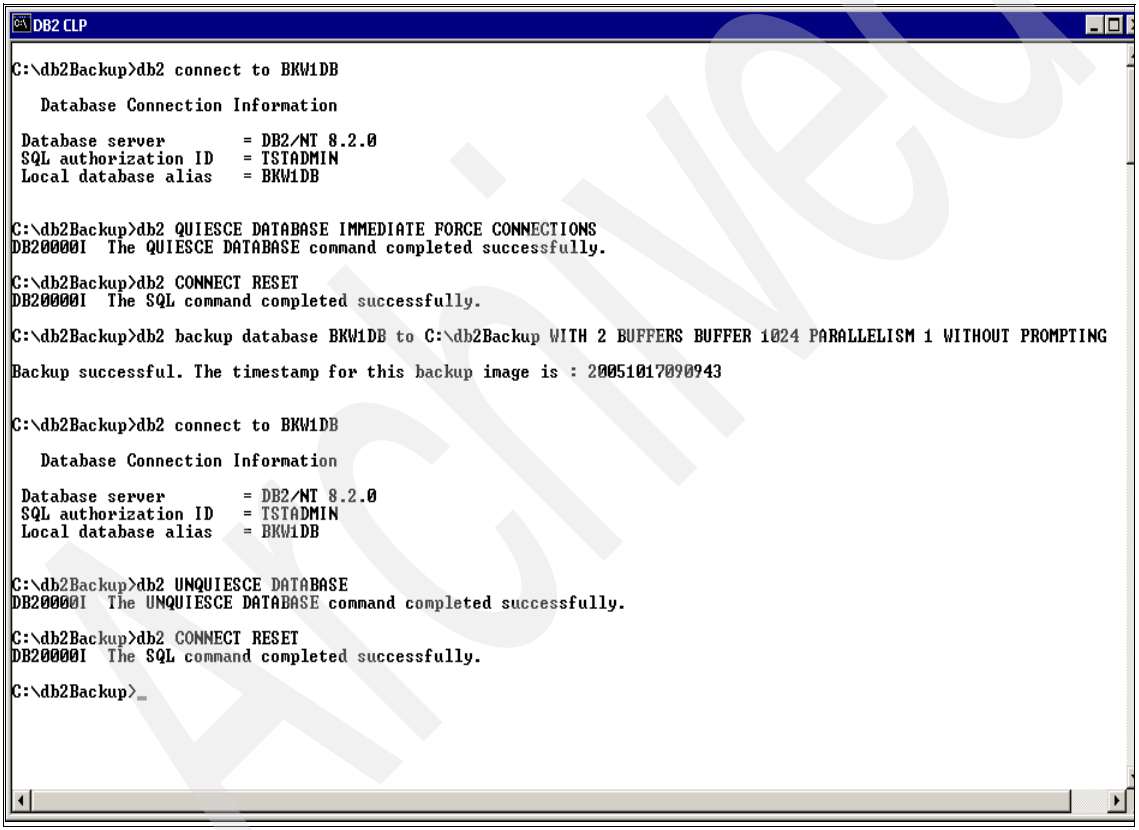


Figure 9-3 DB2 backup

6. Back up the Broker and New Era Of Networks databases on Windows or Unix machines using the commands documented in Step 5.

On z/OS systems use an Image Copy Utility (DBBACKUP) job and get the LRSN value for this backup using a DBREPORT job, as illustrated in the following examples.

The JCL samples can be downloaded from the Web. See Appendix C, “Additional material” on page 555 for instructions.

Example 9-1 DBBACKUP JCL sample

```
/DBBACKUP JOB MSGLEVEL(1,1),MSGCLASS=T,NOTIFY=&SYSUID
//PROCS JCLLIB ORDER=(XXXXXX.PROCLIB)
/*JOBPARM SYSAFF=XXXX
//*
//UTIL EXEC DSNUPROC,SYSTEM=DBXX,UID='XXXXXXX',UTPROC=''
//*
//*********************************************
//*
//* JCL FOR THE IMAGE COPY UTILITY
//* DATE: 13/10/05
//* CREATED BY: DEANEJ
//* UPDATED BY: Mark Hiscock
//* COPY THIS MEMBER INTO YOUR BROKER PDSE AND
//* EDIT THIS JOB AND UPDATE THE FOLLOWING:
//*
//* 1. The proclib for your DB2 (line 1)
//* 2. The system name for the job to run on (line 2)
//* 3. The DB2 sub system and user ID (line 4)
//* 4. The broker database name for the LISTDEF command
//* 5. The broker name for the DSN template (not required but consistent)
//*
//* THE JOBS SHOULD BE RUN IN THE FOLLOWING ORDER
//*
//* 1. BACKUP DB WITH DBBACKUP
//* 2. GET LRSN VALUE FOR FIC WITH DBREPORT
//* 3. RECOVER DB WITH DBRESTOR
//*
//*********************************************
//*
//* INCLUDE INDEXSPACES DATABASE DMQXXBRK ALL
//* DOES NOT WORK AS OUR INDEX SPACES DO NOT
//* HAVE THE COPY YES ATTRIBUTE. SPECIFYING THIS
//* RESULTS IN MESSAGE DSNU425I.
//*
//DSNUPROC.SYSIN DD *
LISTDEF COMPONENT INCLUDE TABLESPACES DATABASE DMQXXBRK ALL
TEMPLATE MQXXCOPY
  DSN('WMQI.&DB..&TS..D&DATE..T&TIME..IC')
  UNIT(SYSALLDA) VOLUMES(TOTDCN) SPACE(300,300) TRK
  DISP(NEW,CATLG,DELETE)
```

```
COPY LIST COMPONENT COPYDDN(MQXXCOPY)
  FULL YES
/*
```

Example 9-2 DBREPORT JCL sample

```
//DBREPORT JOB MSGLEVEL(1,1),MSGCLASS=T,NOTIFY=&SYSUID
//PROCS JCLLIB ORDER=(XXXXX.PROCLIB)
/*JOBPARM SYSAFF=XXXX
//*
//UTIL EXEC DSNUPROC,SYSTEM=XXXXX,UID='XXXXX',UTPROC=''
//*
//*********************************************
//*
//* JCL FOR THE REPORT UTILITY
//* DATE: 13/10/05
//* CREATED BY: DEANEJ
//* UPDATED BY: Mark Hiscock
//* COPY THIS MEMBER INTO YOUR BROKER PDSE AND
//* EDIT THIS JOB AND UPDATE THE FOLLOWING:
//*
//* 1. The proclib for your DB2 (line 1)
//* 2. The system name for the job to run on (line 2)
//* 3. The DB2 sub system and user ID (line 4)
//* 4. The broker database name for the LISTDEF command
//*
//* THE JOBS SHOULD BE RUN IN THE FOLLOWING ORDER
//*
//* 1. BACKUP DB WITH DBBACKUP
//* 2. GET LRSN VALUE FOR FIC WITH DBREPORT
//* 3. RECOVER DB WITH DBRESTOR
//*
//*********************************************
//*
//DSNUPROC.SYSIN DD *
LISTDEF COMPONENT INCLUDE TABLESPACES DATABASE DMQXXBRK ALL
QUIESCE LIST COMPONENT
//
/*
```

Example 9-3 Sample output from DBREPORT JCL

```
-DB8W DSNUQUIA - QUIESCE AT RBA 000004290EC2 AND AT LRSN 000004290EC2
```

Tip: For New Era Of Networks components it is also recommended to export the formats and rules using NNRie and NNFie utilities, though it is not mandatory.

9.3 Universally Unique ID (UUID) backup

Each WebSphere MQ Integrator component has its own unique ID. During the first deployment to a WebSphere MQ Integrator, the Configuration Manager exchanges this ID with the broker to guarantee consistency.

We recommend that you back up the UUID (registry information) located on the following:

- For Unix machines

  In the /var/mqsi directory, issue the command **`tar cvf mqsiconfig.tar /var/mqsi`**. This creates a tar file called mqsiconfig.tar that contains the contents of the /var/mqsi directory structure.

- For z/OS machines

  Look at the value of the variable MQSI_REGISTRY in environment file ENVFILE of the broker service userID and issue the **`tar`** command. For example, if MQSI_REGISTRY is /var/wmqi, issue the command **`tar cvf mqsiconfig.tar /var/wmqi.`** This creates a tar file called mqsiconfig.tar that contains the contents of the /var/wmqi directory structure.

- For Windows machines

  Run the command **`regedit`** on a command line. Navigate to the BrokerUUID key by expanding **HKEY_LOCAL_MACHINE** → **SOFTWARE** → **IBM** → **WebSphereMQIntegrator** → **2** → ***<BrokerName>***

  <BrokerName> is the name of the broker.

  Select the BrokerUUID key as shown in Figure 9-4 on page 141. Right-click **BrokerUUID** and select **Modify**.

  Select the contents of “Value data:” from the Edit String dialog box and copy these contents into a text file, or make a note of them.

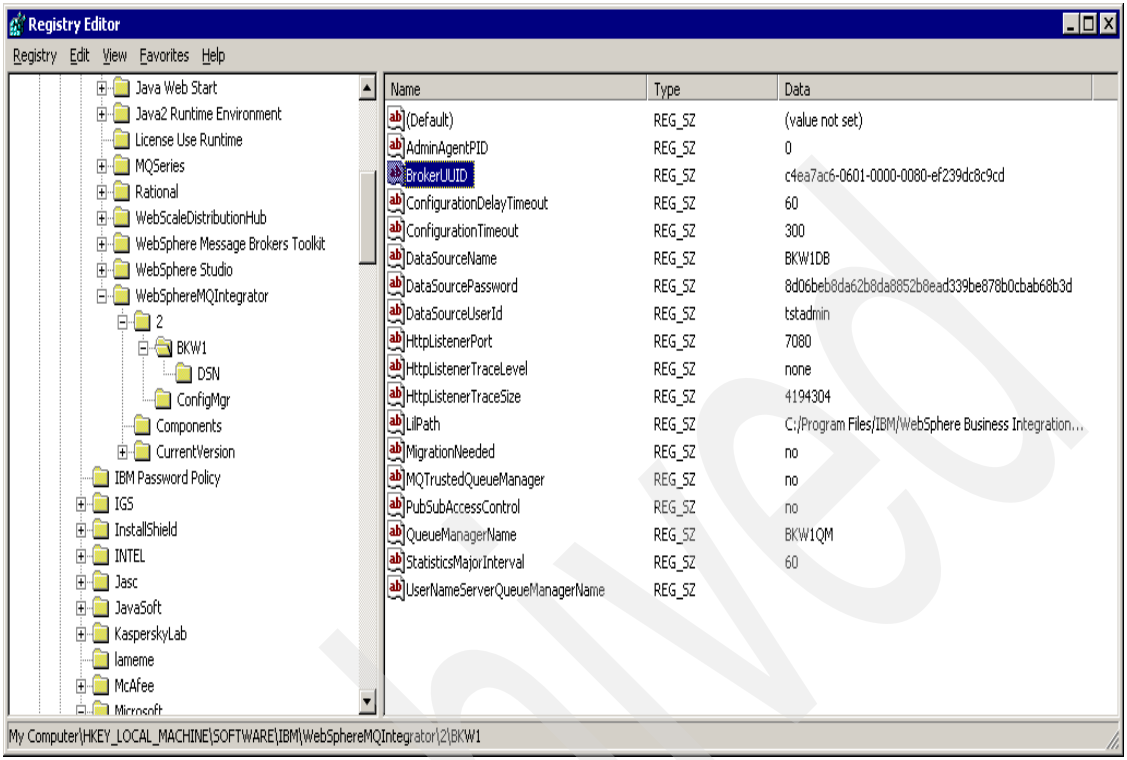


Figure 9-4 UUID key in the Registry Editor

9.4 Control Center

Before backing up the Control Center resources, for example message flows and message sets, stop any debug sessions in the Control Center.

9.4.1 Workspace backup

On the Configuration Manager machine, save the workspace from WebSphere MQ Integrator Broker V2.1:

- Go to the tool\repository\private subdirectory, for example, the C:\Program Files\IBM\WebSphere MQ Integrator 2.1\Tool\repository\private directory:
  - Copy the complete hostname directory into the tool\repository\private subdirectory, for example 127.0.0.1.

- Copy the file <localhost>.wdp, for example 127.0.0.1.wdp, so that the plug-in representation files are also preserved.

- In C:\Program Files\IBM\WebSphere MQ Integrator 2.1\Tool, copy also the Panes.ini file (not mandatory).

Important: Saving your workspace in this way does not save the resources associated with the workspace; only references are saved. Go to “Exporting message flows from the WebSphere MQ Integrator Broker V2.1 Control Center” on page 152 for an explanation on how to save the resources.

9.4.2 Topics, topology, and assignments

Using Configuration Manager on the same machine, with the same queue manager and configuration repository from Version 2.1 preserves the assignments, topology, and topics data during migration to WebSphere Message Broker V6.0.

For backup purposes, manually record the following information from the Control Center, which can also be used to verify the environment after migration:

- For each broker that needs to be migrated, and for the associated assignments configuration data that needs to be preserved, record:
  - The name of the broker
  - The name of each message set that is assigned to the broker
  - The name of each execution group within the broker
  - For each execution group within the broker, the name of each message flow that is assigned to the execution group
  - The status of all message flows (that are running or stopped)
  - For each message flow assigned to an execution group, the properties:
    - Additional instances
    - Commit count
    - Commit interval
    - Coordinated transaction
- For the topology configuration data that needs to be preserved, record:
  - The name of each collective and the names of the brokers that are in the collective
  - All direct connections between brokers
- For the topics configuration data that needs to be preserved, record:

- The hierarchy of topics, including the name of each topic
- The Access Control List for each topic

Export the contents of the Control Center workspace by clicking **File** → **Export All in Workspace**. Assignments configuration data can be extracted from the export file that is produced.

9.4.3 User-defined nodes or parsers

1. Check whether the broker is using the user-defined nodes or parsers.
2. Find out the install directory for defined nodes or parsers by reviewing the registry entry LilPath.
   - On Unix machines, check the directories specified in the registry entry /var/mqsi/registry/<broker_name>/LilPath.
   - On z/OS machines, look at the value of the variable MQSI_REGISTRY in the environment file ENVFILE of the broker service userID and check the directories at subdirectory registry/<broker_name>/LilPath inside this directory.

     For example, if MQSI_REGISTRY is /var/wmqi, and the broker name is MQZIBRK, check the directories specified in the /var/wmqi/MQZIBRK/LilPath entry.
   - On Windows machines, check the directories specified in the registry entry HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphereMQIntegrator\<broker_name>\LilPath.
3. Save all *.jar, *.zip and *.lil files from these directories to a user directory.

10

Migrating the WebSphere MQ Integrator V2.1 Control Center resources

This chapter describes migrating the WebSphere MQ Integrator V2.1 Control Center resources to the WebSphere Message Broker V6.0 Toolkit. We also discuss using the WebSphere Message Broker V6.0 Toolkit on Windows, as well as sharing V6.0 Message Brokers Toolkit projects.

10.1 The WebSphere Message Broker V6.0 Toolkit

This section describes the differences between the WebSphere Message Broker V6.0 Toolkit and the WebSphere MQ Integrator V2.1 Control Center.

Important: We assume that you have installed the WebSphere Message Broker V6.0 Toolkit, as described in the topic Installing in the WebSphere Message Broker V6.0 Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ax01445_.htm

into a different location on the same machine that has either the WebSphere MQ Integrator Broker V2.1 or WebSphere MQ Integrator V2.1 Control Center installed.

The WebSphere Message Broker V6.0 Toolkit is based on the latest release of Rational Application Developer V6.0.0.1 and Eclipse V3.01.

Just as WebSphere MQ Integrator V2.1 Control Center is the graphical user interface (GUI) for the WebSphere MQ Integrator V2.1 product, the WebSphere Message Broker V6.0 Toolkit is the GUI for WebSphere Message Broker V6.0.

Tasks such as message flow, message set, and Embedded Structured Query Language (ESQL) development are performed in a task-specific editor for each file type (.msgflow, .mxsd, and .esql). Each development file type is supported by a validation builder. When you save a file, the validation builder automatically checks the files for errors or warnings. The ESQL required for a message flow appears in one view. The WebSphere Message Broker V6.0 Toolkit prompts you to save each view as a file.

The WebSphere Message Broker V6.0 Toolkit resources are stored in the local file system and are no longer in a repository in the Configuration Manager. Broker application resources and other files are stored in the local file system in a directory called *workspace*.

Developers can perform message flow and message set development functions without connecting to the Configuration Manager. However, testing requires a connection with the Configuration Manager, with at least one active broker.

The WebSphere Message Broker V6.0 Toolkit allows users to concurrently connect to multiple broker domains via different Configuration Managers. This helps developers connect to the development and test domains from the same Message Broker Toolkit for testing the WebSphere Message Broker resources.

The WebSphere Message Broker V6.0 Toolkit is available on Windows and Linux (x86 platform). You can install a WebSphere Message Broker V6.0 Toolkit on Windows to coexist with either a WebSphere MQ Integrator V2.1 Control Center or a WebSphere Message Broker Toolkit V5.0.

A WebSphere Message Broker V6.0 Toolkit manages WebSphere MQ Integrator V2.1, WebSphere Business Integration Message Broker V5.0, and WebSphere Message Broker V6.0 brokers when it is connected to the Configuration Manager of either a WebSphere Business Integration Message Broker V5.0 or a WebSphere Message Broker V6.0.

10.2 Migrating the tooling resources to WebSphere Message Broker V6.0

This section discusses the planning considerations that should go into the migration of tooling resources to WebSphere Message Broker V6.0. This includes the tasks that have to be performed to migrate the resources. Figure 10-1 illustrates the before-migration and after-migration scenario in the WebSphere MQ Integrator V2.1 Control Center.

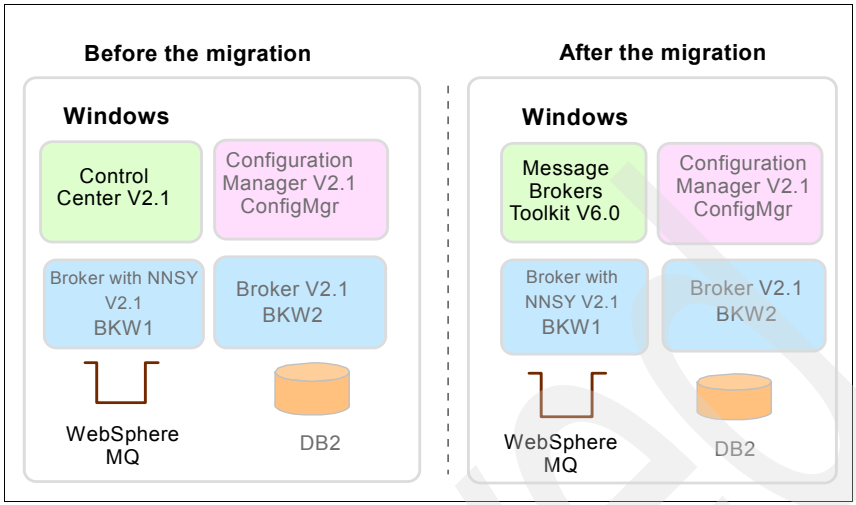


Figure 10-1 WebSphere MQ Integrator V2.1 Control Center scenario

Note: We recommend that you refer to the latest version of the WebSphere Message Broker V6.0 readme file before starting the migration, at:

`http://www.ibm.com/software/integration/mqfamily/support/readme/all60_read.html`

While migrating the Control Center resources, the development resources move from the databases to the file system. In the WebSphere Message Broker V6.0 Toolkit, resource migration is a one-way process. The migrated resources cannot be restored to the databases.

Important: We recommend that you save the resources as a back-up document as described in Chapter 9, “Backing up the WebSphere MQ Integrator V2.1 domain” on page 133.

After the migration is complete, the Configuration Manager database and the Message Repository database can be dropped at your discretion.

10.2.1 Preparing the message flows for migration

Before starting the migration of message flow resources, refer to the topic Migrating a message flow, by selecting the path **Migration → Migration from Version 2.1 products → Migration from WebSphere MQ Integrator → Migrating a message flow** in the WebSphere Message Broker V6.0 Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac02355_.htm

Naming conventions

During migration, the ESQL code is removed from the ESQL-based nodes and stored in a separate .esql file. In WebSphere Message Broker V6.0, the ESQL code is converted into ESQL modules.

The ESQL module name is created by concatenating the message flow name and the compute node name within quotes, as follows:

`ESQL module name = "message_flow_name_compute_node_name"`

The ESQL file name is the message flow name from the WebSphere MQ Integrator V2.1 Control Center, as follows:

`ESQL file name = message_flow_name.esql`

If there are special characters in the ESQL module names and file names, naming conversion problems could occur after migration. Replace the special characters with a series of characters representing the special characters' unicode hexadecimal values. For example, a full stop (.) should be converted to "X2e".

We recommend that you remove certain special characters from the message flow names before migrating. For further details on the same, refer to the topic Migration message flows from Version 2.1, by selecting the path **Reference** → **Migration and upgrade** → **Message flow migration notes** → **Migration message flows from Version 2.1** in the WebSphere Message Broker V6.0 Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an18530_.htm

Promoted property name

In the WebSphere MQ Integrator V2.1 Control Center, promoted property names are created through a drag-and-drop process and can have certain special characters. We recommend that you remove the special characters from the property name. For more information about promoted property names, refer to the topic Promoted property name, by selecting the path **Reference** → **Migration and upgrade** → **Message flow migration notes** → **Migration message flows from Version 2.1** → **Promoted property name** in the WebSphere Message Broker V6.0 Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an18530_.htm

Grouping message flows

In the WebSphere Message Broker V6.0 Toolkit, resources such as message flows and message sets are stored in projects. One project can reference another project.

Groups of message flows that form an application implementation should be grouped together for export, so that they are migrated to a single message flow project.

Namespace in message flows

Use namespaces for message flows that have to be migrated. For more details, refer to Making a message flow namespace aware, by selecting the path **Migration → Migration from Version 2.1 products → Migration from WebSphere MQ Integrator Broker V2.1 → Migrating a message flow → Making a message flow namespace aware** in the WebSphere Message Broker V6.0 Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah20280_.htm

Mapping of node type

In the WebSphere Message Broker V6.0 Toolkit, some of the nodes from the WebSphere MQ Integrator V2.1 Control Center have been replaced. Table 10-1 shows the node type mapping.

Table 10-1 V2.1 nodes converted to V6.0 nodes

Version 2.1 node	Version 6.0 node
Compute	Compute
Database	Database
DataDelete	Database
DataInsert	Database
DataUpdate	Database
Extract	Compute
Filter	Filter
Warehouse	Database

User-defined nodes, SupportPacs, and third-party resources

This section provides information about migrating resources that are not a built-in part of WebSphere MQ Integrator V2.1 Control Center.

User-defined nodes

Before starting the migration of user-defined nodes, refer to the topic Migrating a message flow, by selecting the path **Migration → Migration from Version 2.1 products → Migration from WebSphere MQ Integrator → Migrating a message flow** in the WebSphere Message Broker V6.0 Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac02355_.htm

The steps for migrating user-defined nodes are as follows:

1. Write down the plug-in node properties from the WebSphere MQ Integrator V2.1 Control Center.
2. Remove the plug-in node from the message flows in the WebSphere MQ Integrator V2.1 Control Center.
3. Export the message flows for migration.
4. Build the plug-in node in the WebSphere Message Broker V6.0 Toolkit. For more information, refer to the topic Creating the user interface representation of a user-defined node in the workbench, by selecting the path **Developing application → Developing user-defined extension → Creating the user interface representation of a user-defined node in the workbench** in the WebSphere Message Broker V6.0 Information Center on the Web at:

   http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/as04480_.htm
5. Add the plug-in node to the message flows in the WebSphere Message Broker V6.0 Toolkit.

For details about the WebSphere Message Broker V6.0 built-in nodes that can be substituted for user-defined nodes, refer to the topic Built-in nodes, by selecting the path **Reference → Message flow → Built-in nodes** in the WebSphere Message Broker V6.0 Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.eb.doc/ac04550_.htm

Some SupportPac nodes have the equivalent function built in. For example, you can replace the WebSphere MQ Integrator Broker V2.1 category 3 SupportPac IA0G with the built-in eXtensible Markup Language (XML) Transformation node.

Refer to the following Web site to obtain information about which SupportPacs are supported in WebSphere Message Broker V6.0:

http://www.ibm.com/software/integration/support/supportpacs/category.html

Attention: If a SupportPac or third-party node cannot be migrated or replaced, message flows utilizing this function cannot be migrated to WebSphere Message Broker V6.0.

Exporting message flows from the WebSphere MQ Integrator Broker V2.1 Control Center

Export each message flow group to a separate directory. This makes it much easier to migrate the groups to their respective project.

To export the message flows from the WebSphere MQ Integrator V2.1 Control Center, perform the following tasks:

1. Click the **Message Flows** tab in the Control Center.
2. Right-click the **Message Flows** groups in the left panel.
3. Select **Export** in the context menu, as shown in Figure 10-2.

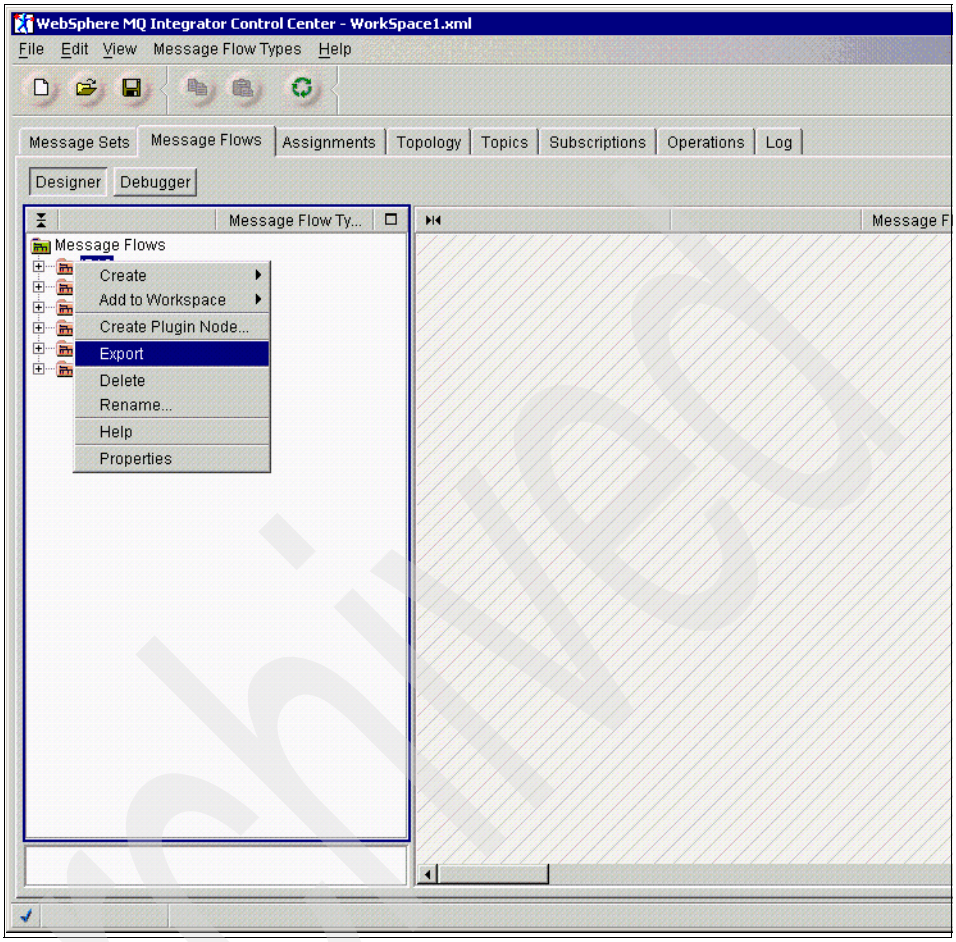


Figure 10-2 Export message flow groups

4. Enter a file name for the message flows to be exported, as shown in Figure 10-3

.

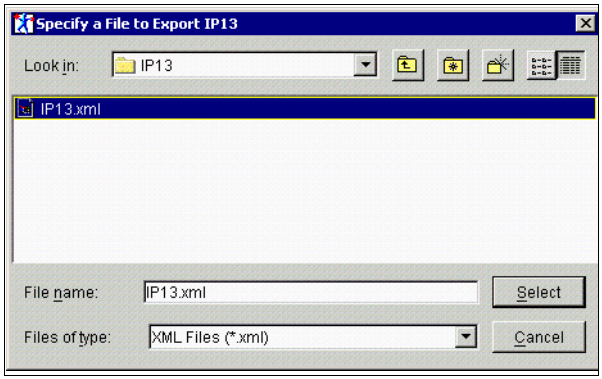


Figure 10-3 Save export to directory

5. Repeat step 4, as shown in Figure 10-3, for all the message flow groups or all the single message flows, as the case maybe.

Tip: Use SupportPac IC01 utilities to create files according to the instructions given on the Web at:

`http://www.ibm.com/software/integration/support/supportpacs/product.html#wmq`

Use these exported files as input for the migration process.

10.2.2 Preparing message sets for migration

Before migrating the message set resources, refer to the topic Migrating Message Sets from Version 2.1, by selecting the path **Migration → Migration from Version 2.1 products → Migration from WebSphere MQ Integrator → Migrating Message Sets from Version 2.1** in the WebSphere Message Broker V6.0 Information Center on the Web at:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ad15750_.htm`

Once the message set resources have been migrated to the WebSphere Message Broker V6.0 Toolkit, the message set datatypes are mapped, as shown in Table 10-2.

Table 10-2 Message set datatypes

MRM type	Schema type
BINARY	xsd:hexBinary

MRM type	Schema type
BOOLEAN	xsd:boolean
DECIMAL	xsd:decimal
DATETIME	xsd:dateTime (also see Table 10-3)
FLOAT	xsd:float
INTEGER	xsd:int
STRING	xsd:string

For the DATETIME type, the simple type mapping is changed by the presence of a Date template value constraint, as shown in Table 10-3.

Table 10-3 DATETIME type

MRM DATETIME Date Template	Schema type
CCYY-MM-DDThh:mm:ss.s	xsd:dateTime
CCYY-MM-DD	xsd:date
CCYY-MM	xsd:gYearMonth
CCYY	xsd:gYear
--MM-DD	xsd:gMonthDay
--MM	xsd:gMonth
---DD	xsd:gDay
Thh:mm:ss.s	xsd:time

If the Date template is not in the preceding list, DATETIME is mapped to an xsd:time. However, if the Date template has only a time component, that is, an xsd:dateTime, or if the Date template has a date and time component, these changes are reported with a BIP0175 warning message in the migration log.

However, this mapping can cause errors to appear in the Problems view after import into the WebSphere Message Broker V6.0 Toolkit.

If the element in question also has the V2.1 Default Value, Min Inclusive, Max Inclusive, or Enumeration value constraints, the values for these do not match the lexical space for an xsd:time or xsd:dateTime, and so fails the validation. These must be corrected manually with the help of the WebSphere Message Broker V6.0 Toolkit editor.

The same task list error also appears for any V2.1 DATETIME type that supplies a Default Value, Min Inclusive, Max Inclusive, or Enumeration value constraint where the value is not fully specified. For example, in the Date template CCYY-MM, the Enumeration 2003 was allowed in WebSphere MQ Integrator V2.1 because it was interpreted as 2003-01at runtime. However, in the new model, the value must match the lexical space of the simple type, and so must include -01.

Exporting message sets from the WebSphere MQ Integrator Broker V2.1 Control Center

Before migrating the message sets, export them from the message repository database in the system that is running the Configuration Manager. All exported message sets can be exported to one directory. Each message set export file must have the extension .mrp, as follows:

message_set_name.mrp

To export message sets from the WebSphere MQ Integrator V2.1 Control Center, enter the following command in the Windows command prompt:

```
mqsiimpexpmsgset -e -n data_source_name -u user_ID -p password -s
messageSetName -l 1 -f target_file_name
```

Example 10-1 illustrates the output of this command.

Example 10-1 Output of the command

```
C:\toMigrate>mqsiimpexpmsgset -e -n MRMDB -u <userid> -p <password> -s
ResultLoan -l 1 -f C:\toMigrate\mrm\ResultLoan.mrp
BIP8071I: Successful command completion.
```

10.2.3 Preparing subscriptions and topics for migration

Applications that have to participate in a publish/subscribe network should support the publish/subscribe interface. Since the rules for this have not changed in the WebSphere Message Broker V6.0 Toolkit, the applications can publish/subscribe the same way it is done in WebSphere MQ Integrator V2.1.

Note: The migration does not preserve the subscriptions and the retained publications in the brokers in WebSphere MQ Event Broker. To recreate this information after migration, subscribers must renew their subscriptions and publishers must republish. For further details, refer to the topic Migration from WebSphere MQ Event Broker Version 2.1 by selecting the path **Migration** → **Migration from Version 2.1 products** → **Migration from WebSphere MQ Event Broker Version** 2.1 in the WebSphere MQ Event Broker Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13740_.htm

10.3 Migrating the tooling resources to WebSphere Message Broker V6.0

This section provides a step-by-step guide to migrating the exported WebSphere MQ Integrator V2.1 Control Center resources to the WebSphere Message Broker V 6.0 Toolkit.

Important: Make sure that the migration of the broker resources is carried out as described in the previous sections, and all the resources are exported successfully from the WebSphere MQ Integrator V2.1 Control Center to the file system.

10.3.1 Migrating message flows

To migrate the message flows, perform the following steps:

Tip: Before issuing the commands for migrating, you can open a Windows command prompt and set the PATH environment variable as follows:

```
set PATH=%PATH%;<toolkit_install_directory>\eclipse
```

Issue all subsequent migration commands in this command prompt.

1. :Migrate the message flows by entering the following command in the command prompt:

   ```
   mqsimigratemsgflows -data workspace location -p project_name -d
   flow_location -log log_file_location
   ```

The output of the command will appear as shown in Example 10-2.

Example 10-2 Migrating a message flow

```
C:\toMigrate>mqsimigratemsgflows -data C:\toMigrate\workspace -p Loan
-d C:\toMigrate\flows\Loan -log c:\toMigrate\Loan.log
Migrating export file C:\toMigrate\flows\Loan\Loan.xml
Migrating message flow Loan Request
Migrating message flow LoanErrorHandlingSub
Migration completed with no errors
Refer to report file c:\toMigrate\Loan.log for details of what was
imported.
When opening the workbench, you need to refresh and rebuild the
project.
```

2. Check the appropriate log file for each message flow. If errors occur, resolve them, clean the workspace location, and repeat the steps. Example 10-3 illustrates the sample log file of a successful migration.

Example 10-3 Sample log file

```
Migrating export file C:\toMigrate\flows\IP13\IP13.xml.
------------------------------------------------
Migrating message flow REQUEST_to_REPLY.
------------------------------------------------
Migrating message flow DB2U.
------------------------------------------------
Migrating message flow FANOUT_plus_original_msg.
------------------------------------------------
Migrating message flow FANIN.
Migration completed with no errors.
```

User-defined nodes, SupportPacs, and third-party resources

To migrate the user-defined nodes, use the same command you used for migrating the message flows. The sequence of actions should be as follows:

1. Migrate a user-defined node by entering the following command in the Windows command prompt:

   `mqsimigratemsgflows -data` *`workspace_location`* `-p` *`project_name`* `-d` *`flow_location`* `-log` *`log_file_location`*

   The output of the command is displayed in Example 10-4.

Example 10-4 Migrating a user-defined node

```
C:\toMigrate>mqsimigratemsgflows -data C:\toMigrate\workspace -p
myNode-d C:\toMigrate\flows\myNode -log c:\toMigrate\myNode.log
Migrating export file C:\toMigrate\flows\myNode\myNode.xml
Migrating message flow myNode Request
Migration completed with no errors
Refer to report file c:\toMigrate\myNode.log for details of what was
imported.
When opening the workbench, you need to refresh and rebuild the
project.
```

2. Check the appropriate log file for each user-defined node. If errors occur, resolve them, clean the workspace location, and repeat the steps.

When a user-defined node is migrated, only the XML interface definition file is migrated into a node.msgnode file. This defines only the terminals and properties of the node. In this version of the product, complete the migration and definition of a user-defined node manually.

For more details on this, refer to the topic Creating the user interface representation of a user-defined node in the workbench, by selecting the path **Developing application → Developing user-defined extension → Creating the user interface representation of a user-defined node in the workbench** in the WebSphere Message Broker Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/as04480_.htm

For details about migrating SupportPacs and third-party resources, refer to the documentation about those resources, since this is outside the scope of this redbook.

10.3.2 Migrating message sets

To migrate the message sets, carry out the following steps in the sequence in which they have been listed:

1. Migrate the message sets by entering the following command in the Windows command prompt:

   ```
   mqsimigratemsgsets  -d message_sets_location -data
   workspace_location -log log_file_location
   ```

   Example 10-5 illustrates the output of the command.

Example 10-5 Migrating a message flow

```
C:\>mqsimigratemsgsets  -d C:\toMigrate\mrm -data
C:\toMigrate\workspace -log c:\toMigrate\MsgSets.log
Migrating export file "C:\toMigrate\mrm\RequestLoan.mrp"
Migrating export file "C:\toMigrate\mrm\ResultLoan.mrp"
BIP0130I Command completed.
Refer to report file for details of what was imported.
```

Tip: Before issuing the commands for migration, open a Windows command prompt and set the PATH environment variable as follows:

```
set PATH=%PATH%;toolkit_install_directory\eclipse
```

Enter all the subsequent migration commands in the same command prompt.

2. If errors occur during migration, review the log file and correct the message set. If the message set is modified in the Control Center, export the message set again. Repeat this until there are no errors in the log file. Ignore warning messages that are logged with `ignoring`, as illustrated in Example 10-6.

Example 10-6 Sample log file

```
Parameter -d (directory) is "C:\toMigrate\mrm"

Migrating export file "C:\toMigrate\mrm\RequestLoan.mrp"

BIP0169W Message Set state of frozen not supported, ignoring
Writing workspace file "/RequestLoan/RequestLoan/messageSet.mset"
Writing workspace file "/RequestLoan/RequestLoan/RequestLoan.mxsd"

Elapsed time processing this message set: 9.391 seconds
Number of warnings for this message set: 1
Number of objects created for this message set: 26

Migrating export file "C:\toMigrate\mrm\ResultLoan.mrp"

BIP0169W Message Set state of frozen not supported, ignoring
Writing workspace file "/ResultLoan/ResultLoan/messageSet.mset"
Writing workspace file "/ResultLoan/ResultLoan/ResultLoan.mxsd"

Elapsed time processing this message set: 0.563 seconds
Number of warnings for this message set: 1
Number of objects created for this message set: 20
Number of export files processed: 2
```

10.4 Using the WebSphere Message Broker V6.0 Toolkit

This section provides a few details about using the Message Broker V6.0 Toolkit. For detailed information, refer to the topic “Administering the broker domain and developing applications” in the WebSphere Message Broker V6.0 Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an03990_.htm

and

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac18710_.htm

The Toolkit provides an integrated development environment to perform application development and broker administration tasks for WebSphere Message Broker V6.0.

These tasks are mainly performed from two perspectives, the Broker Application Development perspective and the Broker Administration perspective.

Open these perspectives from the Message Broker V6.0 Toolkit, as illustrated in Figure 10-4.

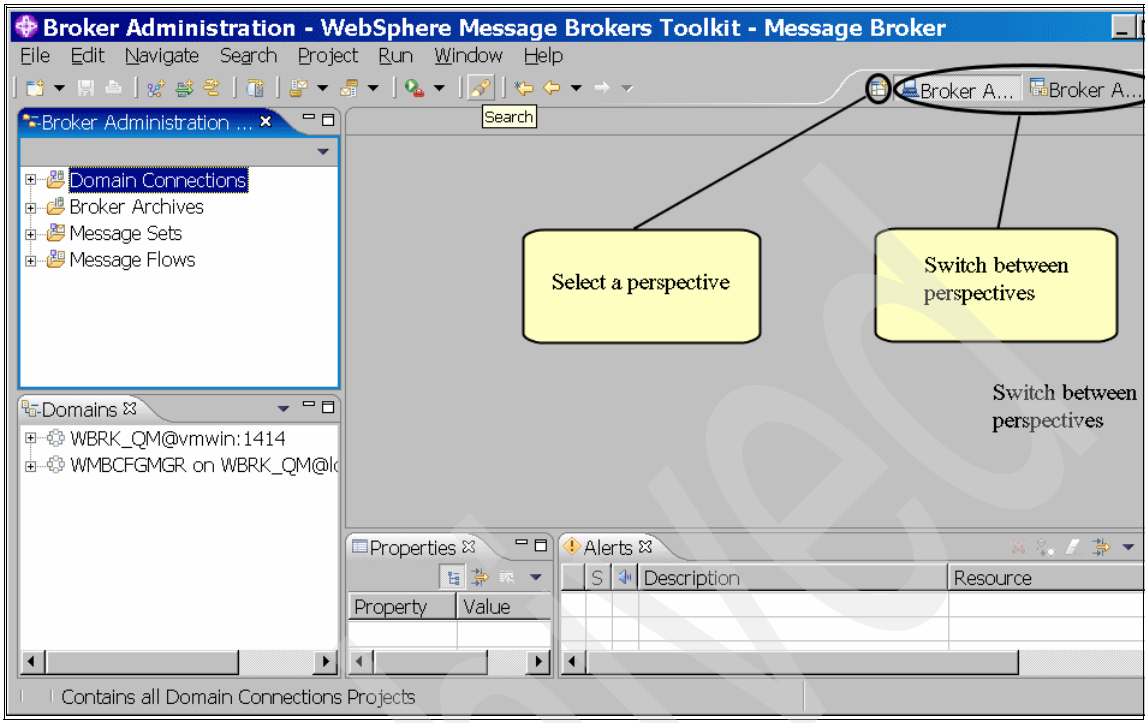


Figure 10-4 Toolkit perspective

To choose a perspective, click the icon indicated in the top right corner of the screen, as shown in Figure 10-4. To switch between perspectives, click the corresponding button.

Broker Application Development perspective

The Broker Application Development perspective is used for application development tasks. It is the default perspective displayed when Message Broker V6.0 Toolkit is started for the first time and is used to:

- Develop message flows and message sets.
- Place and get messages from queues for debugging purposes, with the help of enqueue and dequeue files.

To open an object inside the Broker Application Development perspective, double-click the object in the Resource Navigator view, as shown in Figure 10-5.

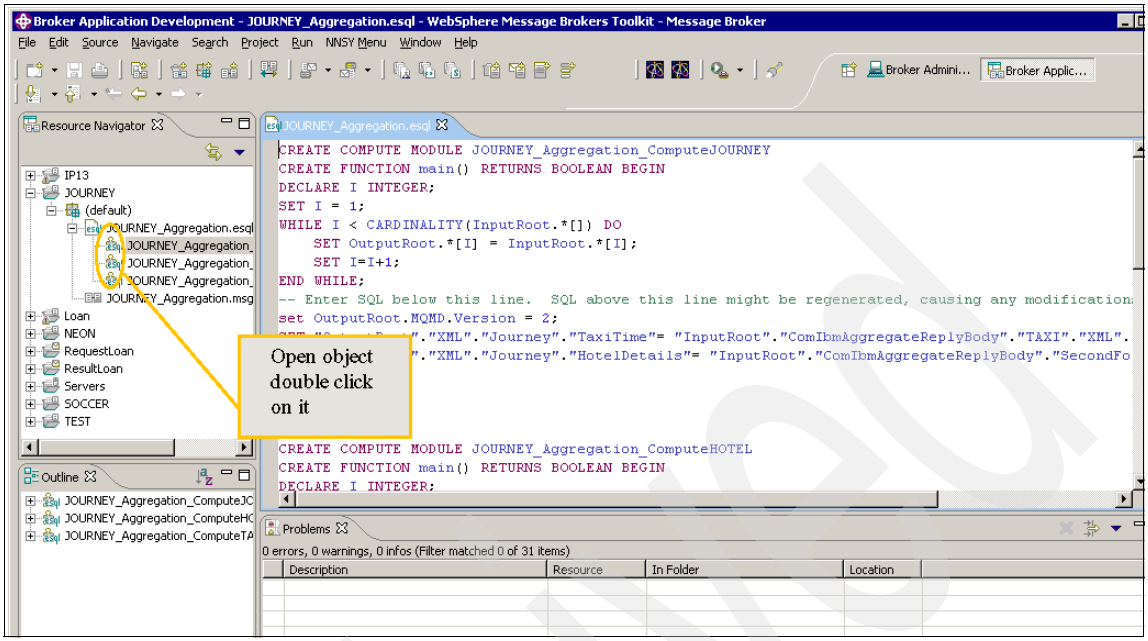


Figure 10-5 Opening an object in the Broker Application Development perspective

One difference in the Control Center in the WebSphere MQ Integrator V2.1 is that the ESQL code is stored in a separate .esql file and the code is structured as schema-scope objects with module names to allow the function to be called from a node. These module names can be assigned to Compute, Database, or Filter nodes.

Figure 10-6 illustrates the process of assigning an ESQL module to a Compute node. Right-click the node, select **Properties**, and assign the ESQL module by clicking **Browse**.

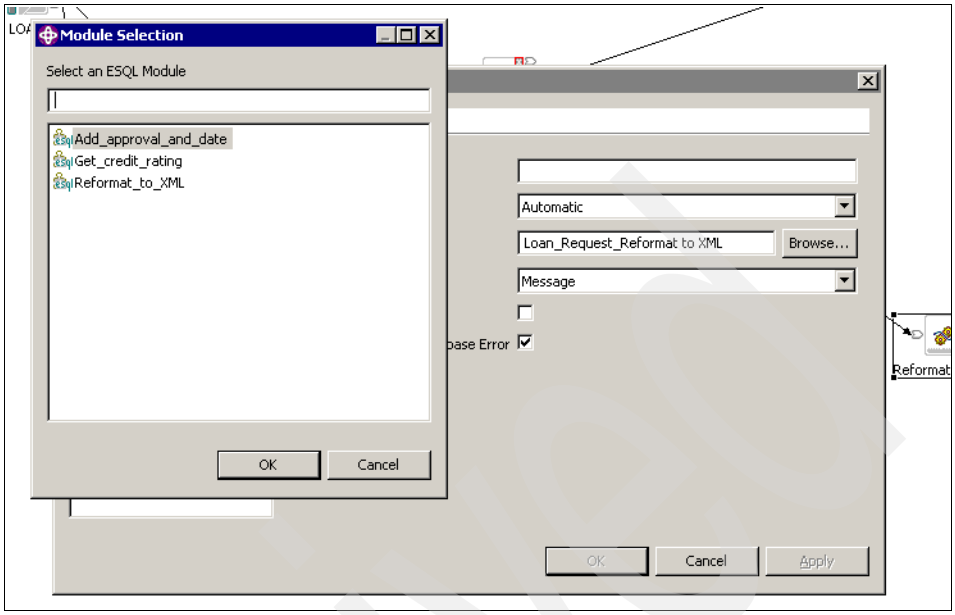


Figure 10-6 Assigning an ESQL module

Broker Administration perspective

The Broker Administration perspective performs tasks that help manage one or more brokers. It is used to:

- Create and manage broker domains and topology
- Create and manage execution groups
- Create and deploy broker archive (bar) files to execution groups
- Manage, publish, and subscribe topics and subscriptions
- Manage event logs and alerts

A single WebSphere Message Broker V6.0 Toolkit can concurrently connect to multiple Configuration Managers, allowing it to manage multiple broker domains.

Many of the broker administration functions are also available as command line utilities. This allows for scripting of commands to enable automated operation of tasks such as deployment and configuration. For more information about this, refer to the topic Commands by selecting the path **Reference** → **Operations** → **Commands** in the WebSphere Message Broker V6.0 Information Center on the Web at:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an07060_.htm`

10.4.1 Loading the migrated resources into the WebSphere Message Broker Toolkit

This section discusses using the migrated tooling resources in the V6.0 Message Brokers Toolkit.

1. Open the migrated workspace with the new WebSphere Message Broker V6.0 Toolkit. All the resources are available in the file system. When opening the Message Broker V6.0 Toolkit for the first time, a dialog box requesting a workspace location appears. Specify the location where the migrated broker resources are located, as shown in Figure 10-7, and click **OK**.

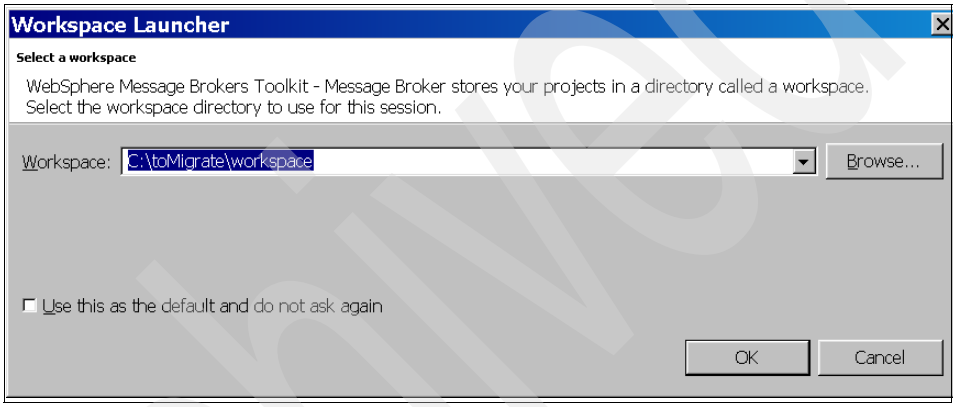


Figure 10-7 Selecting a workspace

To switch to a different workspace location, select **File** → **Switch Workspace** from the menu.

Important: Do not change anything before cleaning the projects as per the directions given in Step 2.

2. Clean the workspace to resolve the errors shown in the Problems view. Clean and rebuild the workspace by selecting **Project** → **Clean** from the menu.
3. A dialog box, as illustrated in Figure 10-8, appears. Select the **Clean all projects** option and click **OK**.

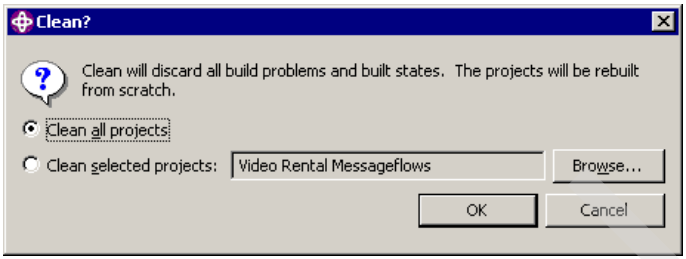


Figure 10-8 Clean all projects

The resources in each project are now rebuilt. The time taken to complete the cleaning process depends on the number and complexity of the projects. View the progress of the process on the progress bar located at the bottom right corner of the WebSphere Message Broker V6.0 Toolkit.

4. On completion of the cleaning process, all errors should be gone. If not, investigate the reason for the errors by clicking **Error** in the Problems view. This displays the reason for the errors in detail.

   A yellow warning sign in the Problems view of the WebSphere Message Broker V6.0 Toolkit indicates a warning; for instance, when the reference to a resource cannot be found. This can be solved by expanding the project references to other dependent projects. Right-click the project, select **Properties** → **Project References** to check the dependent projects, and click **OK**.

5. To confirm that persistent subscriptions are being retained after migration, double-click **Subscriptions** in the Domains view. In the Subscriptions editor that opens (Figure 10-9), click **Query** (To Author: I cant seem to be able to find “Query” in the screen) and review the subscription entries.

   Note: While dealing with a domain with a large number of active subscriptions, we recommend that you filter the subscriptions by topic, user, or broker categories, before clicking **Query**.

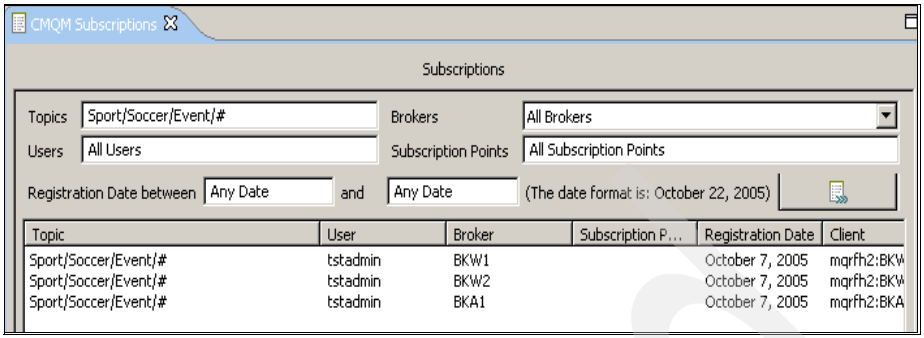


Figure 10-9 Review subscription entries

10.4.2 Deploying the migrated resources

In the Broker Administration perspective, the message flows and message sets can now be deployed to the appropriate broker and execution group. This is different from WebSphere MQ Integrator V2.1, in that, here, the message sets should be deployed to an execution group and not to a broker, and the scope of a message set is the execution group to which it is deployed.

To deploy the message flows and the message sets to the broker, add them to a broker archive file, as follows:

1. To create a broker archive file, switch to or open the Broker Administration perspective, as shown in Figure 10-10. In the Broker Administration Navigator view, select **Broker Archives** and right-click the relevant server project. In the context menu that appears, select **New** → **Message Broker Archive**.

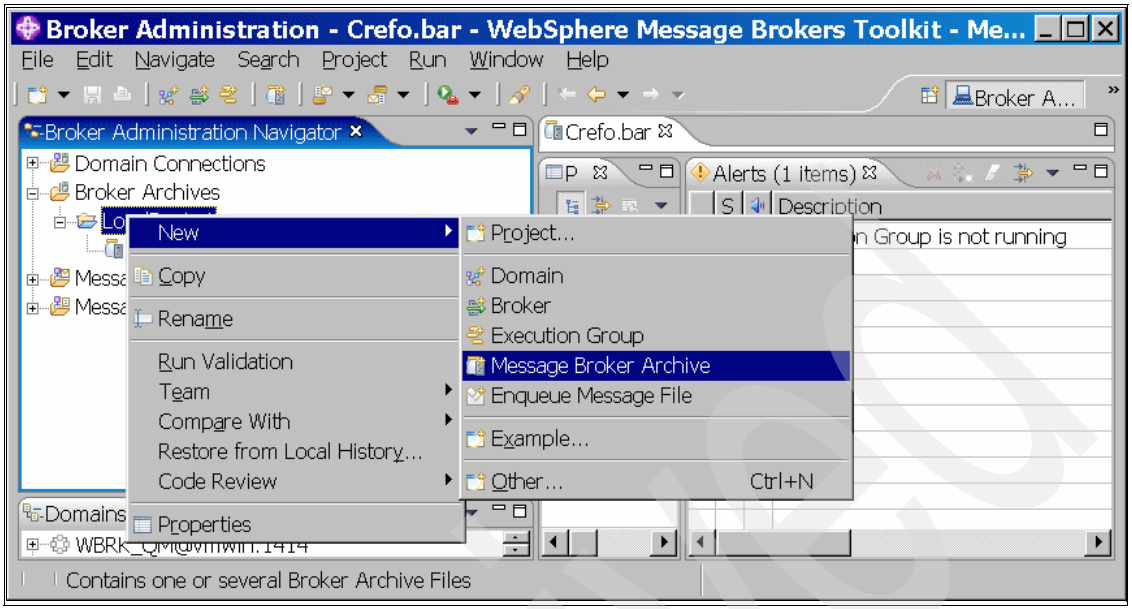


Figure 10-10 Create a broker archive file

2. Add to this file the resources that have to be deployed to an execution group on the broker. Open the broker archive file, click the green plus (**+**) sign, and add the resources as illustrated in Figure 10-11. To deploy the broker archive file to the execution group, right-click the broker archive file in the Broker Administration Navigator, select **Deploy File**, and click the appropriate execution group.

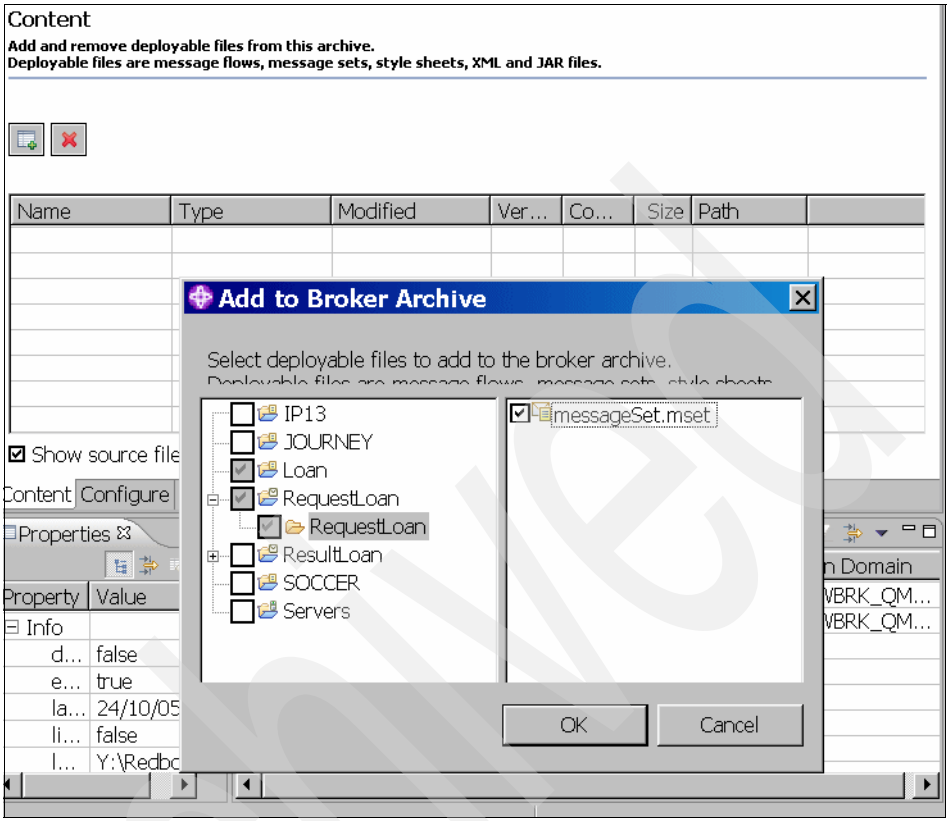


Figure 10-11 Build a broker archive file

Important: When adding message flows to a broker archive file that is going to be deployed to a WebSphere MQ Integrator V2.1 broker, select the **Compile ESQL for broker Version 2.1** check box in the Broker Archive Editor. This is essential because the WebSphere MQ Integrator V2.1 product does not understand the ESQL generated in later versions that implement modules and schemas. However, a broker archive file created with the help of this option can be deployed to the WebSphere Business Integration Message Broker V5.0 or WebSphere Message Broker V6.0.

After adding the resources to a broker archive file, select **Details>>** in the Adding to Broker Archive File dialog box that appears. This reports messages on the process of adding resources to the broker archive. For example, adding a migrated WebSphere Business Integration Message Broker V5.0 message set will output the following warning:

```
BIP0177W This dictionary requires a Version 5 (or later) broker.
This message set has namespace support enabled. Namespaces are not
supported by brokers earlier than Version 5.
```

3. In the Domains view of the Broker Administration perspective, double-click **Event Log**. This opens the Configuration Manager's Event Log window. Review the events to check whether deployment was successful, as shown in Figure 10-12.

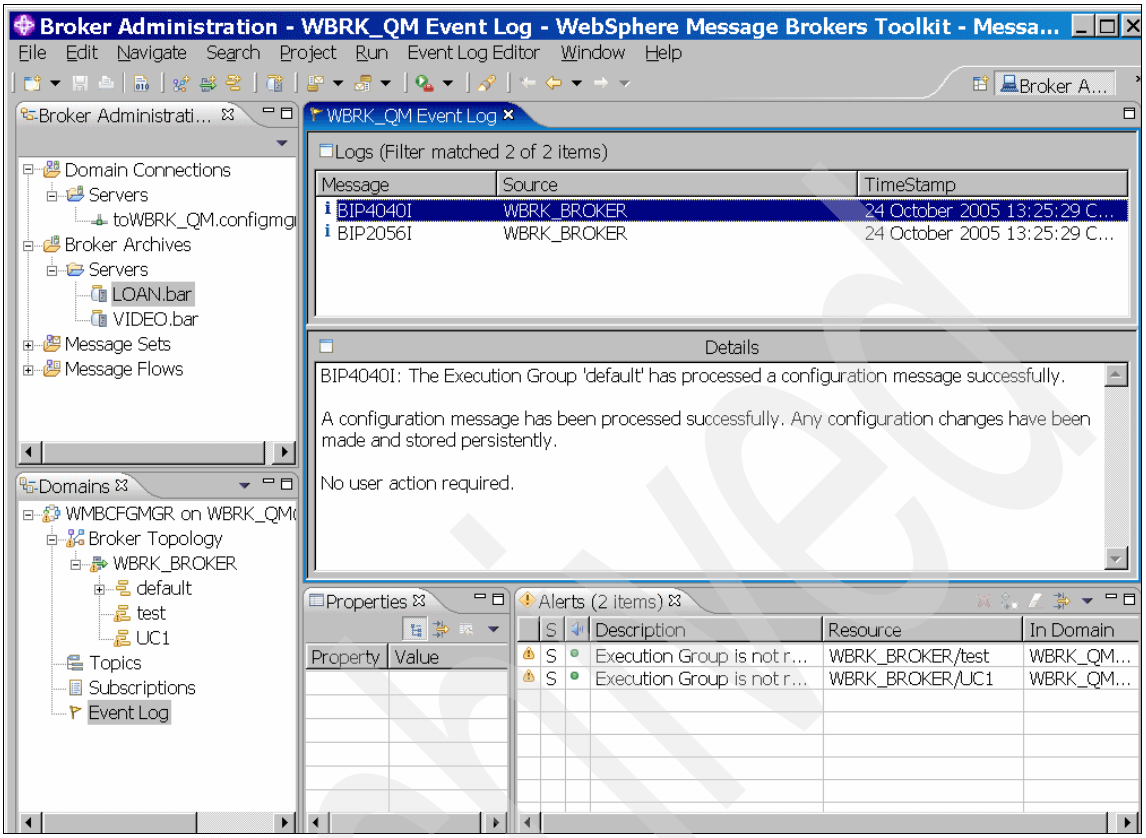


Figure 10-12 Event log

If the IP4040I and BIP2056I information messages appear, it means that the deployment operation to the broker is successful.

10.4.3 Sharing broker resources

The workspace of the WebSphere Message Broker V6.0 Toolkit is file-based. Therefore, we recommend that you use a code repository to share the resources with multiple concurrent users. For more information about the repositories that are supported, refer to the topic “Development repository” by selecting the path **Product overview** → **Technical overview** → **Client environment** → **Development repository** in the WebSphere Message Broker Information Center on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ab00240_.htm

Using a version control code repository has the following benefits:

- Concurrent users of the Message Broker V6.0 Toolkit can share the resources without the risk of overwriting each other's work.
- Each Message Broker V6.0 Toolkit resource can be versioned, that is, produce an archive of previous versions that can be accessed when needed. For example, it is possible to revert to a known working version of a message flow if the latest version has problems.
- Version control repositories can maintain a history of the resource. This can be used to track the changes to a resource, for example, obtain information about who made the change, the date and time of the change, and a description of the change.

Any repository with an Eclipse 3.0 client is supported.

WebSphere Message Broker V6.0 Toolkit provides two perspectives to interact with code repositories. These are:

- The Team Synchronizing perspective provides a view of the project-sharing status of a resource and tasks to synchronize project resources with supported repositories. Figure 10-13 illustrates the Team Synchronizing perspective from the WebSphere Message Broker V6.0 Toolkit.

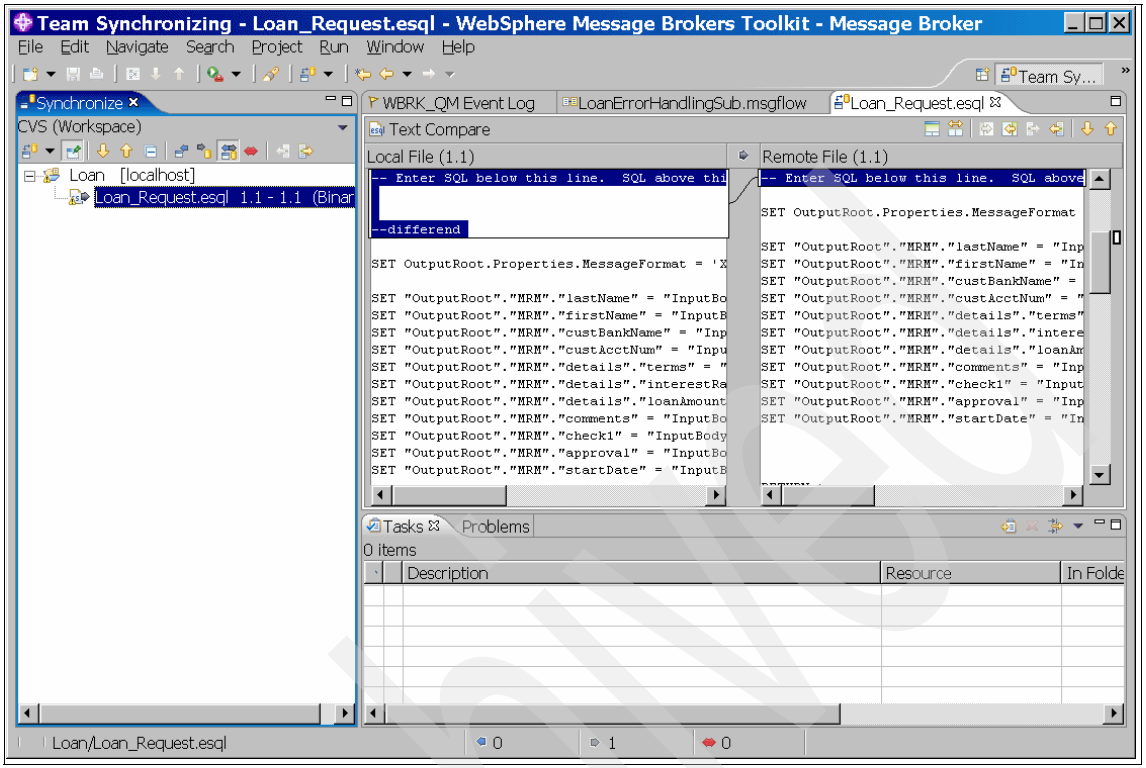


Figure 10-13 The Team Synchronizing perspective

- The CVS Repository perspective shows the content of a CVS-based repository. Figure 10-14 illustrates the CVS Repository perspective from the WebSphere Message Broker V6.0 Toolkit.

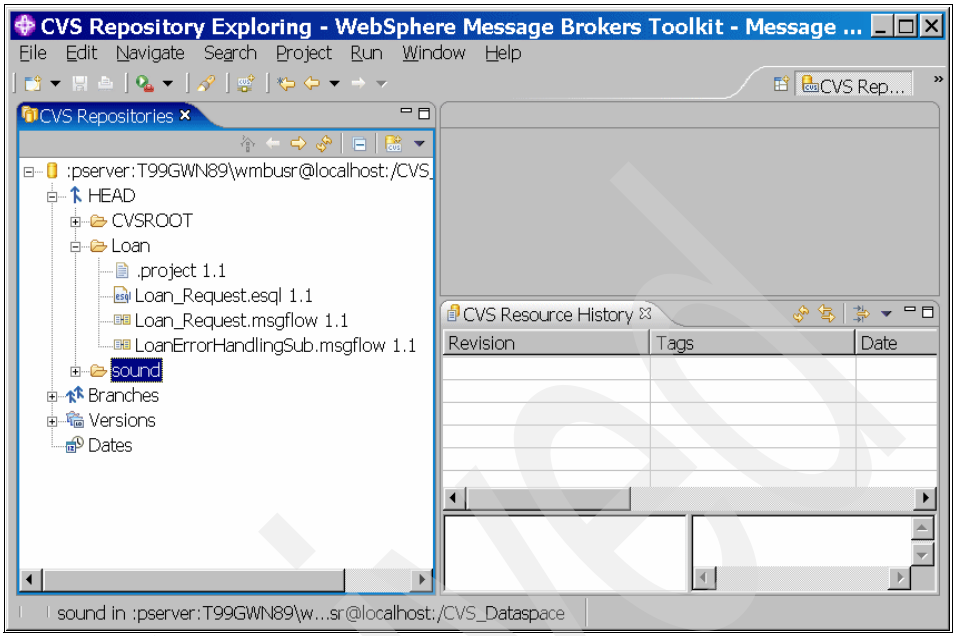


Figure 10-14 The CVS Repository perspective

For more information about configuring CVS to run with the WebSphere Message Broker V6.0 Toolkit, refer to the topic “Configuring CVS” by selecting the path **Configuring the broker domain** → **Configuring the workbench** → **Configuring CVS** in the WebSphere Message Broker V6.0 Information Center on the Web at:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac26530_.htm`

To share a project in the WebSphere Message Broker V6.0 Toolkit with a code repository, right-click the project and select **Team** → **Share Project** from the context menu that appears. Use the Wizard to add the project to a repository.

Having added the project to a repository, the Broker Application Development perspective’s Resource Navigator view indicates the projects are managed by a code repository by changing the icons.

Figure 10-15 illustrates the RequestLoan message set project and the JOURNEY message flow project sharing a code repository. The **>** symbol in the JOURNEY message flow project and its ESQL file indicate that the project’s ESQL file, which is at version 1.1 in the repository, has been changed in the workbench, and should be synchronized with the repository.

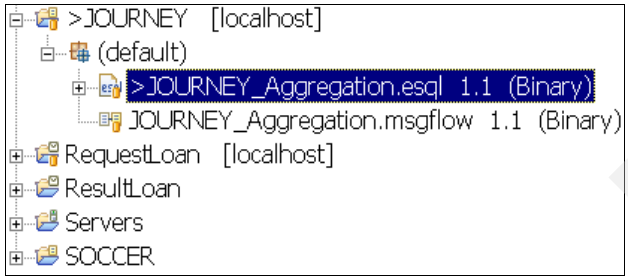


Figure 10-15 Projects in the Broker Application Development perspective shared with a code repository

11

Migrating the Configuration Manager

This chapter describes the process of migrating the WebSphere MQ Integrator V2.1 Configuration Manager to WebSphere Message Broker V6.0. We also discuss the ability to change the platform of the Configuration Manager to z/OS, and some of the enhancements of the WebSphere Message Broker V6.0 Configuration Manager.

Attention: This chapter assumes that the migration of the WebSphere MQ Integrator V2.1 Control Center to the WebSphere Message Broker V6.0 Toolkit has been completed, as documented in Chapter 10, “Migrating the WebSphere MQ Integrator V2.1 Control Center resources” on page 145.

11.1 Migrating Configuration Manager to WebSphere Message Broker V6.0

This section assumes that:

- The WebSphere MQ Integrator V2.1 domain has been backed up, as described in Chapter 9, “Backing up the WebSphere MQ Integrator V2.1 domain” on page 133, should you need to reinstate the environment.
- The WebSphere MQ Integrator V2.1 Control Center resources has been migrated to the WebSphere Message Broker V6.0 Toolkit, as described in Chapter 10, “Migrating the WebSphere MQ Integrator V2.1 Control Center resources” on page 145.

The Configuration Manager maintains the broker domain configuration, interfacing between the Message Brokers Toolkit and a set of executing brokers. e It is a central repository for the broker domain components, communicating with other components in the domain using WebSphere MQ.. The Configuration Manager also deploys to the broker domain as a result of the actions initiated through the Message Brokers Toolkit and reports the resultant status messages back to the Event Log.

There are several options for migrating a Configuration Manager: it can be migrated in situ on the same computer, migrated to another computer without DB2 Universal Database installed, or migrated to another computer with DB2 Universal Database installed. It is important to select the scenario most appropriate to your environment. To learn more about the various options, refer to the WebSphere Message Broker Information Center, under the topic **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 → Migrating components on different computers → Migrating the Configuration Manager**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23925_.htm

Whatever option you choose, it is essential that you stop the Configuration Manager prior to the migration.

Note: The reason all references to WebSphere Message Broker Information Center's topic regarding the Configuration Manager migration refer to WebSphere MQ Integrator Broker rather than WebSphere MQ Integrator is indicated in the WebSphere Message Broker Information Center, under the topic **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Version 2.1**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13720_.htm

To migrate components from WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0, follow the instructions under the topic **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23929_.htm

Replace all references to "WebSphere MQ Integrator Broker" with "WebSphere MQ Integrator."

11.1.1 Migrating the Configuration Manager in situ

For information about the process of migrating the Configuration Manager on the same system, refer to the WebSphere Message Broker Information Center, under the topic **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 → Migrating components on different computers → Migrating the Configuration Manager → To the same computer**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23938_.htm

This section demonstrates the process of replacing the WebSphere MQ Integrator V2.1 Configuration Manager with a WebSphere Message Broker V6.0 Configuration Manager in situ. Figure 11-1 illustrates the environment before and after the migration of the Configuration Manager.

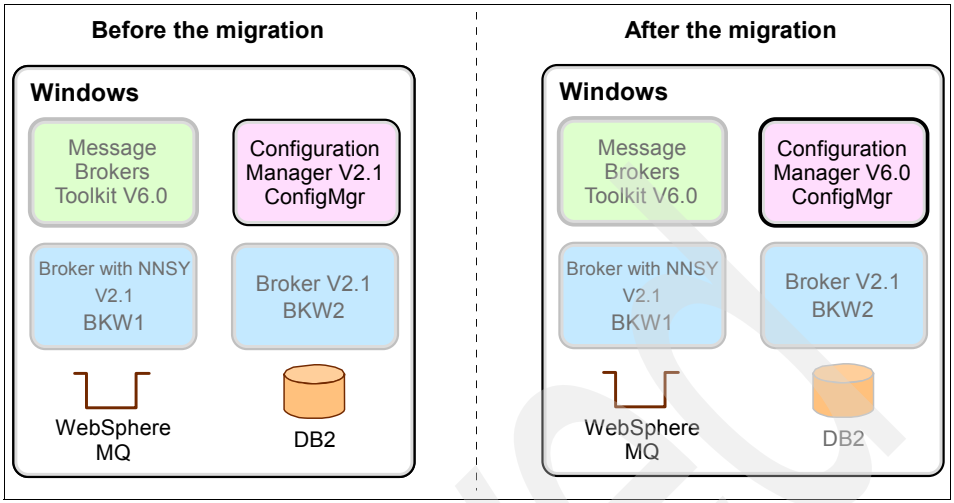


Figure 11-1 Environment before and after the migration of the Configuration Manager V2.1 in situ

The environment shown here illustrates the coexistence of WebSphere Message Broker V6.0 components with brokers at WebSphere MQ Integrator V2.1. A migrated Configuration Manager can manage a heterogeneous set of brokers. For information about which WebSphere Message Broker V6.0 components can coexist with components from the previous versions, refer to the WebSphere Message Broker Information Center, under the topic **Migrating** → **Coexistence**, on the Web at:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23930_.htm`

This Web site also provides links to subsquent topics detailing the conditions under which components from the previous versions can participate in the WebSphere Message Broker V6.0 domain.

Important: The instructions in this chapter assume that the WebSphere Message Broker V6.0 Configuration Manager component has been installed (as described in the installation guide) into a different location on the same machine that has the WebSphere MQ Integrator V2.1 Configuration Manager installed. You can find this guide on the Web at:

`http://www.elink.ibmlink.ibm.com/public/applications/publications/cgibin/pbi.cgi?¶¶CTY=US&FNC=SRX&PBL=GC34-6621-00`

1. If a Control Center is open, close it. If the Message Brokers Toolkit is connected to the Configuration Manager, disconnect from it in the Domains view.
2. In the Windows command prompt, stop the Configuration Manager by entering the following command:

   ```
   mqsistop ConfigMgr
   ```

   Start the Event Viewer by selecting **Start** → **Programs** → **Administrative Tools** → **Event Viewer** and select **Application Log**. When the Configuration Manager has successfully stopped, a warning event message with a source identifier of WMQIv210 and event identifier 2002 is displayed.

   Important: Make sure that you have stopped the Configuration Manager before attempting the migration.

3. Open a WebSphere Message Broker V6.0 Command Console by selecting **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **Command Console**, and enter the following command:

   ```
   mqsilist -a
   ```

   This lists all the components that have been created on the system. Prior to the migration, the Configuration Manager entry indicates that it is a Version 2 component. In the environment demonstrating the migration, the Configuration Manager and two Windows brokers are currently configured on the system, as illustrated in Figure 11-1 on page 180 and in Example 11-1.

Example 11-1 Output from mqsilist -a prior to the Configuration Manager migration

```
C:\Program Files\IBM\MQSI\6.0>mqsilist -a
BIP8221I: Broker: BKW1 (Version 2)  -  BKW1QM
BIP8221I: Broker: BKW2 (Version 2)  -  BKW2QM
BIP8221I: ConfigMgr: ConfigMgr (Version 2)  -  CMQM
BIP8071I: Successful command completion.
```

4. Before migrating the Configuration Manager, you must perform a premigration check, which determines whether the component can be safely migrated. To do a premigration check of the Configuration Manager, enter the following command in the Command Console:

   ```
   mqsimigratecomponents ConfigMgr -c
   ```

   Example 11-2 shows the output from a successful premigration check.

Example 11-2 Output from a premigration check of the Configuration Manager

```
C:\Program Files\IBM\MQSI\6.0>mqsimigratecomponents ConfigMgr -c
```

```
BIP8674I: Configuration Manager 'ConfigMgr' (Version 2.1) specified for
migration.
BIP8680I: Pre-migration check succeeded.
BIP8071I: Successful command completion.
```

Note: The premigration check does not check whether it is possible to access the Configuration Manager's database. Therefore, even if a premigration check is successful, the migration may fail if the datasource user ID and password combination is incorrect.

5. To migrate the Configuration Manager to WebSphere Message Broker V6.0, enter the following command in the Command Console:

   ```
   mqsimigratecomponents ConfigMgr
   ```

   This causes the Configuration Manager's queues, registry, and file system information to migrate from WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0. Example 11-3 shows the output from a successful in situ Configuration Manager migration.

Example 11-3 Output from the Configuration Manager migration command

```
C:\Program Files\IBM\MQSI\6.0>mqsimigratecomponents ConfigMgr
BIP8674I: Configuration Manager 'ConfigMgr' (Version 2.1) specified for
migration.
BIP8755I: Copied value 'QueueManagerName' into the new location
BIP8755I: Copied value 'DataBaseName' into the new location
BIP8755I: Copied value 'NTDomainAware' into the new location
BIP8755I: Copied value 'DataBaseUserId' into the new location
BIP8755I: Copied value 'DataBasePassword' into the new location
BIP8755I: Copied value 'MRMDataSourceName' into the new location
BIP8755I: Copied value 'MRMDataSourceUserId' into the new location
BIP8755I: Copied value 'MRMDataSourcePassword' into the new location
BIP8755I: Copied value 'JDBCDriverName' into the new location
BIP8755I: Copied value 'JDBCConnectionName' into the new location
BIP8755I: Copied value 'NTSecurityDomainName' into the new location
BIP8755I: Copied value 'UserNameServerQueueManagerName' into the new
location
BIP8755I: Copied value 'AdminAgentPID' into the new location
BIP8763I: Deleted value 'QueueManagerName' from the old location
BIP8763I: Deleted value 'DataBaseName' from the old location
BIP8763I: Deleted value 'NTDomainAware' from the old location
BIP8763I: Deleted value 'DataBaseUserId' from the old location
BIP8763I: Deleted value 'DataBasePassword' from the old location
BIP8763I: Deleted value 'MRMDataSourceName' from the old location
```

```
BIP8763I: Deleted value 'MRMDataSourceUserId' from the old location
BIP8763I: Deleted value 'MRMDataSourcePassword' from the old location
BIP8763I: Deleted value 'JDBCDriverName' from the old location
BIP8763I: Deleted value 'JDBCConnectionName' from the old location
BIP8763I: Deleted value 'NTSecurityDomainName' from the old location
BIP8763I: Deleted value 'UserNameServerQueueManagerName' from the old
location
BIP8763I: Deleted value 'AdminAgentPID' from the old location
BIP8768I: Finished registry migration for component 'ConfigMgr'.
BIP8654I: Moving filesystem artefacts from '' to 'C:\Documents and
Settings\All Users\Application Data\IBM\MQSI'
BIP8071I: Successful command completion.
```

6. After migrating the Configuration Manager, you can run a postmigration check. This ensures that the migration has been completed successfully, confirming that the correct queues exist and the registry is in the correct format. To run a postmigration check of the Configuration Manager, enter the following command in the Command Console:

   ```
   mqsimigratecomponents ConfigMgr -v
   ```

   Example 11-4 shows the output from a successful postmigration check.

Example 11-4 Output from a postmigration check of the Configuration Manager

```
C:\Program Files\IBM\MQSI\6.0>mqsimigratecomponents ConfigMgr -v
BIP8674I: Configuration Manager 'ConfigMgr' (Version 6.0) specified for
migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

7. To complete the migration of the Configuration Manager, start by entering the following command in the Command Console:

   ```
   mqsistart ConfigMgr
   ```

   Starting the Configuration Manager for the first time after migration causes it to detect the domain information held in the DB2 Universal Database of the WebSphere MQ Integrator V2.1 Configuration Manager and migrate it to the internal repository of the WebSphere Message Broker V6.0 Configuration Manager. In this way, the WebSphere Message Broker V6.0 Configuration Manager controls the same broker domain configuration as the WebSphere MQ Integrator V2.1 Configuration Manager used to.

Note: Be aware that the time for the Configuration Manager startup is longer than normal. This is because starting the Configuration Manager for the first time after migration causes the migration process to complete copying of the configuration data held in the database into the internal repository.

11.1.2 Validating the Configuration Manager migration

This section describes several ways to validate the Configuration Manager migration.

Application Log messages

Start the Event Viewer, if not already started by selecting **Start** → **Programs** → **Administrative Tools** → **Event Viewer** and then select **Application Log**. When the Configuration Manager has successfully started for the first time after migration, seven information event messages are logged, as illustrated in Figure 11-2.

Type	Date	Time	Source	Category	Event
Information	14/10/2005	14:23:52	WebSphere Broker v6000	None	1003
Information	14/10/2005	14:23:50	WebSphere Broker v6000	None	8280
Information	14/10/2005	14:23:50	WebSphere Broker v6000	None	8255
Information	14/10/2005	14:23:26	WebSphere Broker v6000	None	1228
Information	14/10/2005	14:23:17	WebSphere Broker v6000	None	1224
Information	14/10/2005	14:23:17	WebSphere Broker v6000	None	1223
Information	14/10/2005	14:23:04	WebSphere Broker v6000	None	2001

Figure 11-2 Expected events when Configuration Manager is first started after migration

Note: The source identifier of a WebSphere Message Broker V6.0 component event message is WebSphere Broker v6000 rather than WMQIv210, which identified a WebSphere MQ Integrator V2.1 component event message.

Subsequent successful Configuration Manager starts only have four information event messages logged: event identifiers 2001, 8255, 8280, and 1003.

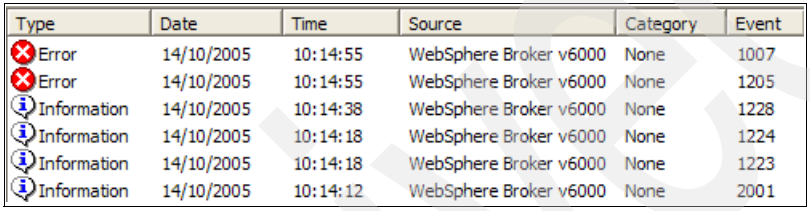
Troubleshooting

Error event messages are logged in the system's local error log, if any problems are encountered when starting the Configuration Manager for the first time after migration. For details about some of the common issues that may be encountered while migrating resources, refer to the WebSphere Message Broker Information Center, under the topic **Troubleshooting and support** → **Dealing**

with problems → Problems when importing or migrating resources, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/au16660_.htm

The issues encountered may depend on the size and complexity of the domain. For example, while performing the migration that is demonstrated here, we encountered the following errors because the scenario's result had thousands of persistent subscriptions in each of the brokers. The Windows Application Log, illustrated in Figure 11-3, appears when a Configuration Manager DB2 Universal Database parameter was tuned to deal with the size of the domain to be migrated.



Type	Date	Time	Source	Category	Event
Error	14/10/2005	10:14:55	WebSphere Broker v6000	None	1007
Error	14/10/2005	10:14:55	WebSphere Broker v6000	None	1205
Information	14/10/2005	10:14:38	WebSphere Broker v6000	None	1228
Information	14/10/2005	10:14:18	WebSphere Broker v6000	None	1224
Information	14/10/2005	10:14:18	WebSphere Broker v6000	None	1223
Information	14/10/2005	10:14:12	WebSphere Broker v6000	None	2001

Figure 11-3 Configuration Manager failed to migrate successfully due to a DB2 tuning issue

Opening the error event message 1205 indicates the need to tune the DB2 Universal Database APP_CTL_HEAP_SZ parameter, as illustrated in Figure 11-4.

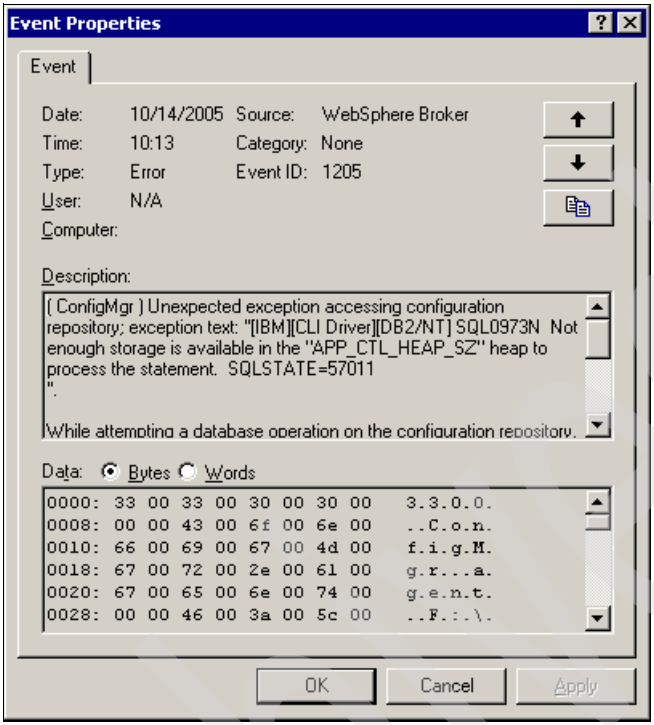


Figure 11-4 Error event 1205: need to tune the DB2 APP_CTL_HEAP_SZ parameter

This set of event messages is repeated until the Configuration Manager is stopped. This is because the error encountered is an unexpected exception and the Configuration Manager is designed to retry the operation after a short delay, as explained in error event message 1007.

This issue can be encountered because of the size of the broker domain that the Configuration Manager manages. To address this problem, perform the following steps:

1. Stop the Configuration Manager by entering the following command in the Command Console:

   ```
   mqsistop ConfigMgr
   ```

   When the Configuration Manager has successfully stopped, a warning event message with a source identifier of WebSphere Broker v6000 and event identifier 2002 is displayed in the system's local error log.

2. At DB2 Universal Database command prompt, perform the following steps:

   a. Connect to the Configuration Manager database by entering the following command:

      ```
      db2 connect to cm_db user db_userid using db_password
      ```

   b. Ascertain the current value of the APP_CTL_HEAP_SZ parameter for the Configuration Manager database:

      ```
      db2 get db cfg for cm_db | find "APP_CTL_HEAP_SZ"
      ```

      The output produced by this command is as follows:

      ```
      Max appl. control heap size (4KB)     (APP_CTL_HEAP_SZ) = 128
      ```

   c. Increase the value of the APP_CTL_HEAP_SZ parameter for the Configuration Manager database:

      ```
      db2 update db cfg for cm_db using APP_CTL_HEAP_SZ higher_value
      ```

   d. Force all applications to disconnect from the database to allow the changes to become effective:

      ```
      db2 force application all
      ```

   e. To force the change to become effective, immediately stop and restart the DB2 Universal Database by entering the following commands:

      ```
      db2stop
      db2start
      ```

3. Start the Configuration Manager by entering the following command in the Command Console:

   ```
   mqsistart ConfigMgr
   ```

   When the Configuration Manager has successfully started, a series of seven information event messages with a source identifier of WebSphere Broker v6000 and event identifiers 2001, 1223, 1224, 1228, 8255, 8280, and 1003 are displayed in the system's local error log (see Figure 11-2 on page 184).

   If the system's error log still displays the set of six event identifiers 2001, 1223, 1224, 1228, 1205, and 1007 (see Figure 11-3 on page 185) repeatedly, then repeat steps 1 to 3 on page 187, increasing the value assigned to APP_CTL_HEAP_SZ.

Listing the components

List all the components that are in the system by entering the following command in the Command Console:

```
mqsilist -a
```

After the migration, the Configuration Manager entry does not display a version indicating that it is at the latest level, namely WebSphere Message Broker V6.0. In the environment that is migrated, the Configuration Manager and the two Windows brokers reside in the same system. The output shown in Example 11-5 illustrates that although the Configuration Manager has been migrated, the brokers have been unaffected and remain at WebSphere MQ Integrator V2.1 as they did prior to the Configuration Manager migration (see Example 11-1 on page 181).

Example 11-5 Output from mqsilist -a following the Configuration Manager migration

```
C:\Program Files\IBM\MQSI\6.0>mqsilist -a
BIP8221I: Broker: BKW1 (Version 2)  -  BKW1QM
BIP8221I: Broker: BKW2 (Version 2)  -  BKW2QM
BIP8099I: ConfigMgr: ConfigMgr  -  CMQM
BIP8071I: Successful command completion.
```

Connecting to the domain in the Message Brokers Toolkit

Start the Message Brokers Toolkit, if it is not already open, by slecting **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **WebSphere Message Brokers Toolkit** and open the Broker Administration perspective. Then perform the following steps:

1. In the Domains view, right-click and select **New** → **Domain** as illustrated in Figure 11-5.

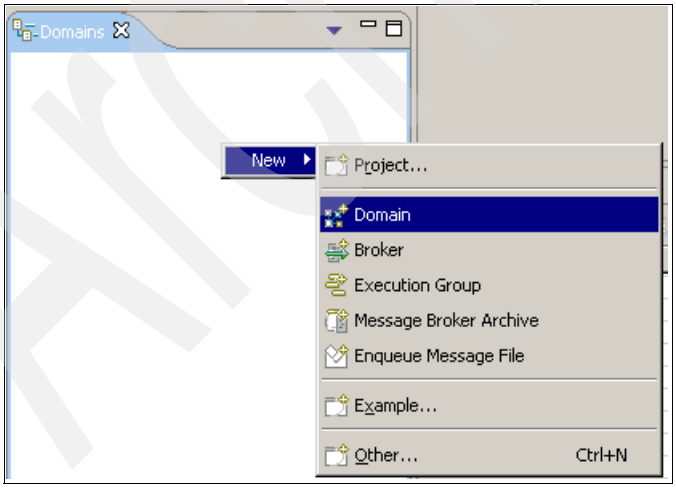


Figure 11-5 Creating a new domain connection in the Domains view

2. In the dialog that appears, complete the Configuration Manager's queue manager Name, Host, and Port, then select **Next,** as illustrated in Figure 11-6.

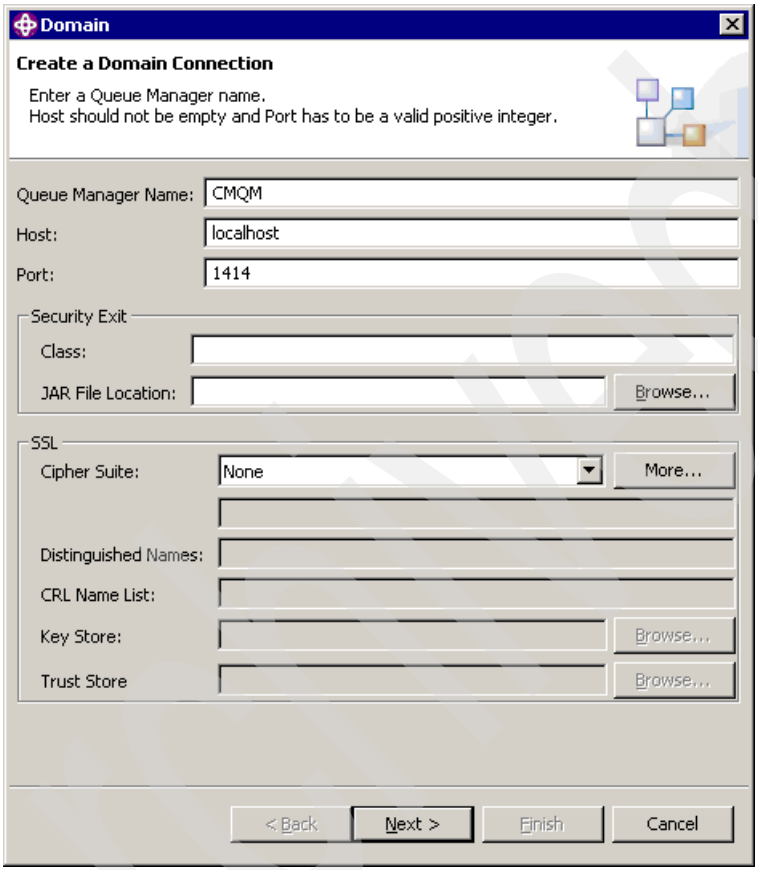


Figure 11-6 Creating a new domain connection to the migrated Configuration Manager

3. When the Configuration Manager connection has been established, another dialog appears. Complete the Connection name and select **Finish**, as illustrated in Figure 11-7.

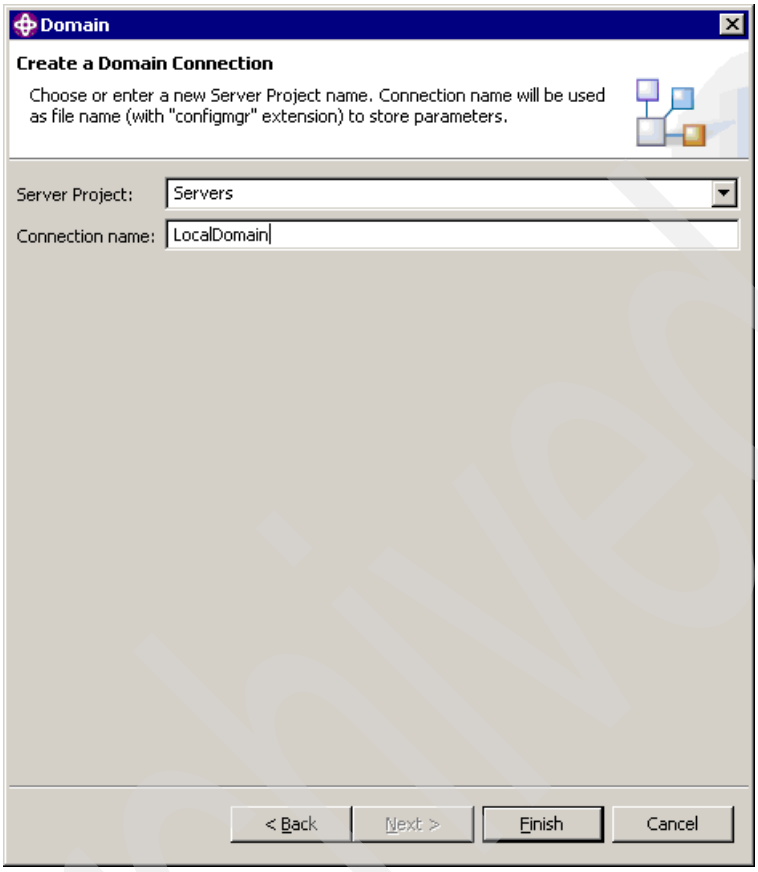


Figure 11-7 Specify a domain connection name

If you have not previously created a Server project in the Message Brokers Toolkit, a dialog appears, as seen in Figure 11-8. Select **Yes**.

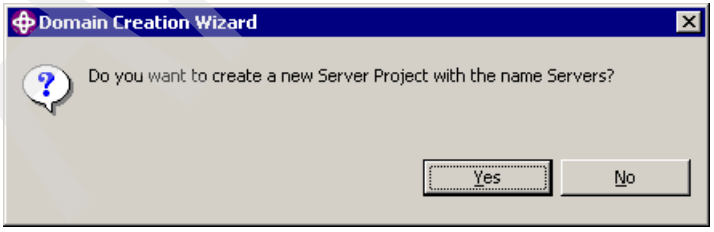


Figure 11-8 Creating a Server project for the new domain connection to be stored

The Domains view shows the new connection to the migrated Configuration Manager. When the broker topology is expanded (as illustrated in Figure 11-9), it

shows the domain of brokers previously controlled by WebSphere MQ Integrator V2.1 Configuration Manager in the state they were when the Configuration Manager was stopped prior to the migration.

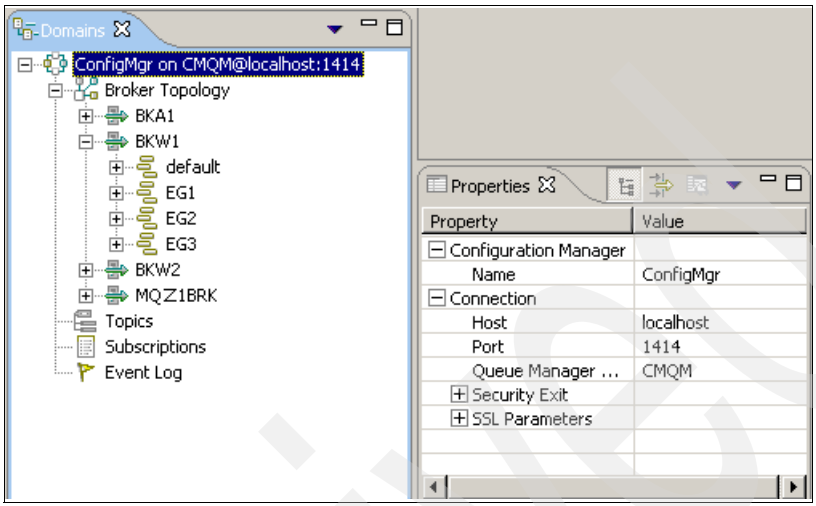


Figure 11-9 Migrated Configuration Manager's broker domain in the Message Brokers Toolkit

When the execution groups of the brokers in the domain are expanded, each one has a copy of any message sets that were previously deployed to the broker at WebSphere MQ Integrator V2.1, as illustrated in Figure 11-10. This is expected behavior because the method by which message sets are deployed has changed since WebSphere MQ Integrator V2.1. At WebSphere MQ Integrator V2.1, message sets were deployed to the broker. However, since WebSphere Business Integration Message Broker V5.0, message sets are deployed to the execution group.

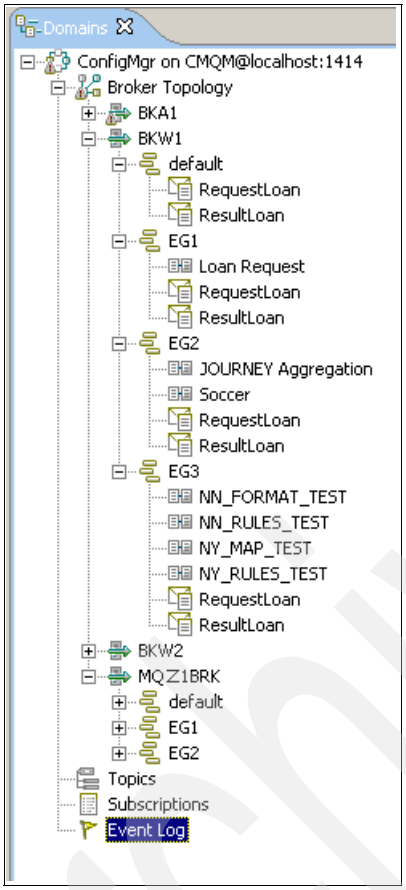


Figure 11-10 A migrated WebSphere MQ Integrator V2.1 broker domain

Therefore, in order to retain the same runtime environment at WebSphere Message Broker V6.0, we have placed a copy of all the message sets deployed to the broker at WebSphere MQ Integrator V2.1 in each execution group belonging to that broker. It is the responsibility of the user to delete additional copies of the message sets from the appropriate execution groups, based on application knowledge. To delete unrequired copies of the message sets, perform the following steps:

1. In the Domains view, select a message set that is not required in the execution group and right-click it.
2. In the context menu that appears, select **Remove**.
3. When the dialog appears (as illustrated in Figure 11-11), select **OK**.

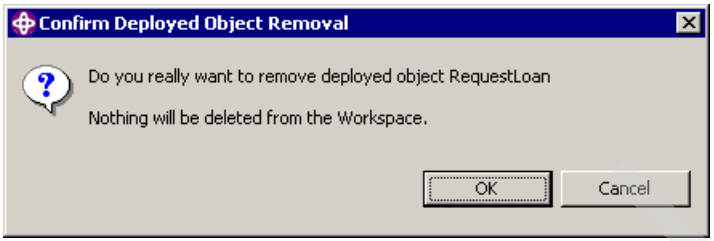


Figure 11-11 Confirm the removal of the deployed message set selected

This removes the deployed message set from the execution group, but does not affect the message set stored in the workspace. The result is reflected in the Message Brokers Toolkit Domains view and the Event Log, with BIP2056I and BIP4040I successful response messages.

Alternatively, you can remove several message sets from an execution group by entering the following command in the Command Console, using a colon-separated list for the deployed objects to be removed:

```
mqsideploy -i cm_host -p cm_port -q cm_qm -b broker -e exgrp -d
deployed_object1:deployed_object2
```

Example 11-6 demonstrates the removal of the RequestLoan and ResultLoan message sets from the default execution group of the Windows Broker BKW2, and the response messages received.

Example 11-6 Removing message sets from an execution group using the mqsideploy command

```
C:\Program Files\IBM\MQSI\6.0>mqsideploy -i localhost -p 1414 -q CMQM
-b BKW2 -e default -d RequestLoan.dictionary:ResultLoan.dictionary
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1062I: Removing the following objects from execution group
'default':
          RequestLoan.dictionary ResultLoan.dictionary
BIP1092I: Broker BKW2 successfully processed the deployment request.
```

The result from the command is also reflected in the Message Brokers Toolkit Domains view and the Event Log, with BIP2056I and BIP4040I successful response messages.

Restriction: Currently, performing either of these methods to remove the extra message sets resulting from the migration, does not remove them from the execution group. To overcome this issue, deploy the message flows and message sets needed for each execution group by entering the following command:

```
mqsideploy -i cm_host -p cm_port -q cm_qm -b broker -e exgrp -a bar
-m
```

This deploys the specified broker archive file to the broker's specified execution group. The `-m` parameter causes a complete rather than an incremental or delta deployment, and removes all currently deployed message flows and message sets from the execution group as part of the deployment.

To completely clear an execution group of the message sets and leave it empty, first create an empty broker archive file and then perform the command as detailed in the previous section.

To confirm that persistent subscriptions are retained after the migration, double-click **Subscriptions** in the Domains view. In the Subscriptions editor that is opened, click **Query**, as shown in Figure 11-12.

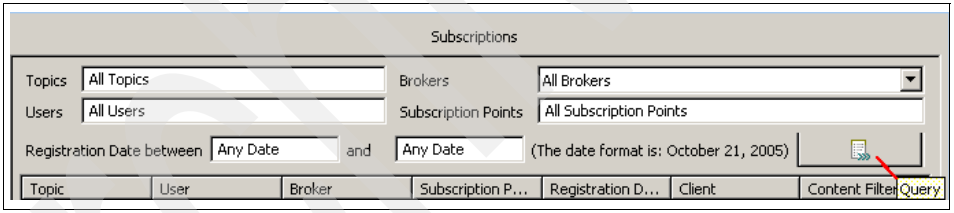


Figure 11-12 Select Query in the Subscriptions editor to see subscriptions

Note: In a domain with a large number of active subscriptions, it is advisable to filter the subscriptions by topic, user, or broker before clicking **Query**.

Manipulating the domain

After connecting to the Configuration Manager in the Message Brokers Toolkit, you can administer the domain from the Broker Administration perspective, for example, deploying broker archive files and using the Domains view to start and stop message flows. There are also a series of new and enhanced command line commands (discussed in the following section) that offer an alternative method to administer a domain. The result of these commands is also reflected in the Message Brokers Toolkit Domains view, Alerts view, and the Event Log, with the BIP2056I and BIP4040I successful response messages.

Note1: Both WebSphere Business Integration Message Broker V5.0 and WebSphere Message Broker V6.0 broker archive files can be deployed to WebSphere MQ Integrator V2.1, WebSphere Business Integration Message Broker V5.0, and WebSphere Message Broker V6.0 brokers as long as the functionality deployed is applicable to the runtime. For example, if you are developing new resources in the WebSphere Message Broker V6.0 Toolkit, ensure that the new functionality available only at WebSphere Message Broker V6.0 is not deployed to Message Broker V5.0 or WebSphere MQ Integrator V2.1 brokers.

Note2: When new broker archive files are created in WebSphere Message Broker V6.0 from the existing broker resources, validate the runtime results to ensure that the migrated resources are working as expected.

The following inserts are used in the examples of the commands:

- *cm_host* is the IP address or host name of the Configuration Manager.
- *cm_port* is the port number of the Configuration Manager.
- *cm_qm* is the queue manager of the Configuration Manager.
- *broker* is the name of the broker to deploy to.
- *exgrp* is the name of the execution group to deploy to.
- *bar* is the location (relative or absolute reference) and name of the broker archive file to be deployed.
- *deployed_objects* is a colon-separated list of deployed objects to be removed from the execution group.
- *msgflow* is the name of the message flow to stop.

Some examples of the commands for administering the domain from the command line are:

- Performing an incremental deployment of a broker archive file to an execution group by entering the following command in the Command Console:

  ```
  mqsideploy -i cm_host -p cm_port -q cm_qm -b broker -e exgrp -a bar
  ```

  Example 11-7 demonstrates deploying the TEST.bar broker archive file from the workspace directory to the default execution group of the Windows Broker BKW1 and the response messages received.

Example 11-7 Deploying a broker archive file using the mqsideploy command

```
mqsideploy -i localhost -p 1414 -q CMQM -b BKW1 -e default -a
C:\toMigrate\workspace\Servers\TEST.bar
```

```
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1039I: Deploying BAR file 'C:\toMigrate\workspace\Servers\TEST.bar'
to broker 'BKW1' (execution group 'default') ...
BIP1092I: Broker BKW1 successfully processed the deployment request.
```

- Stopping a message flow in an execution group by entering the following command in the Command Console:

  `mqsistopmsgflow -i` *`cm_host`* `-p` *`cm_port`* `-q` *`cm_qm`* `-b` *`broker`* `-e` *`exgrp`* `-m` *`msgflow`*

  Example 11-8 demonstrates stopping the TEST message flow in the default execution group of the Windows Broker BKW1 and the response messages received.

Example 11-8 Stopping a message flow in an execution group using the mqsistopmsgflow command

```
mqsistopmsgflow -i localhost -p 1414 -q CMQM -b BKW1 -e default -m TEST
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1028I: Submitting request to the Configuration Manager to stop
'BKW1/default/TEST'...
BIP1029I: Waiting up to 60 second(s) for broker BKW1 to update its
configuration...
BIP1031I: Message flow 'BKW1/default/TEST' is reported as stopped.
BIP8071I: Successful command completion.
```

- Removing a message flow or message set from an execution group by entering the following command in the Command Console:

  `mqsideploy -i` *`cm_host`* `-p` *`cm_port`* `-q` *`cm_qm`* `-b` *`broker`* `-e` *`exgrp`* `-d` *`deployed_objects`*

  Example 11-9 demonstrates removing the TEST message flow and RequestLoan message set from the default execution group of the Windows Broker BKW1 and the response messages received.

Example 11-9 Removing a message flow and message set from an execution group using the mqsideploy command

```
mqsideploy -i localhost -p 1414 -q CMQM -b BKW1 -e default -d
TEST.cmf:RequestLoan.dictionary
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1062I: Removing the following objects from execution group
'default':
          TEST.cmf RequestLoan.dictionary
```

```
BIP1092I: Broker BKW1 successfully processed the deployment request.
```

Restriction: If a message flow is running in two brokers in the same named execution group and you right-click the message flow in the Message Brokers Toolkit Domains view and select **Stop** from the context menu, the message flow in the other broker may stop. This is the same for the **Start** command. To resolve this issue in a similar scenario, use the following commands**:**

```
mqsistopmsgflow
mqsistartmsgflow
```

This is a known issue and should be resolved in a future fix pack.

Postmigration

After the migration, the WebSphere MQ Integrator V2.1 Configuration Manager's DB2 Universal Database still exists unmodified. The WebSphere Message Broker V6.0 Configuration Manager uses an internal repository instead of a database. Therefore, after a successful validation of the migrated Configuration Manager, the database can be removed at the user's discretion.

After following the backup procedures in Chapter 9, "Backing up the WebSphere MQ Integrator V2.1 domain" on page 133, if you need to restore the WebSphere MQ Integrator V2.1 environment, you can restore this database. However, it is not possible to roll back a changed domain to DB2 Universal Database. Therefore, any changes made to a migrated domain will not be reflected in the restored database.

Important: Ensure that the Configuration Manager has started successfully after migration and prior to removing the Configuration Manager database, because this is when the configuration data held in the database is copied into the internal repository.

11.2 Changing the Configuration Manager platform to z/OS

In WebSphere Message Broker V6.0, the Configuration Manager runtime component is supported on the same platforms as the broker runtime. Therefore, you may want to move the Configuration Manager's platform as well as migrating it to WebSphere Message Broker V6.0. For details of the environment needed, refer to the topic **WebSphere Message Broker system requirements**, on the Web at:

http://www-306.ibm.com/software/integration/wbimessagebroker/requirements/index.html

11.2.1 Moving the Configuration Manager's platform after migrating to WebSphere Message Broker V6.0

For information about the two-step process of migrating the Configuration Manager on the same system and then moving the migrated Configuration Manager to another system, refer to the WebSphere Message Broker Information Center, under the topic **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 → Migrating components on different computers → Migrating the Configuration Manager → To a different computer that does not have DB2 installed,** on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23941_.htm

For information about the steps to follow when moving a Configuration Manager to a new queue manager, refer to the topic **Configuring the broker domain → Configuring broker domain components → Modifying a Configuration Manager → Moving the Configuration Manager to a new queue manager**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae20240_.htm

We have already demonstrated the process of migrating the Configuration Manager in situ in 11.1.1, "Migrating the Configuration Manager in situ" on page 179. We now demonstrate the process of moving the Configuration Manager from a Windows system to a queue manager on a z/OS system. Figure 11-13 illustrates the environment after completing the Configuration Manager move to the z/OS system.

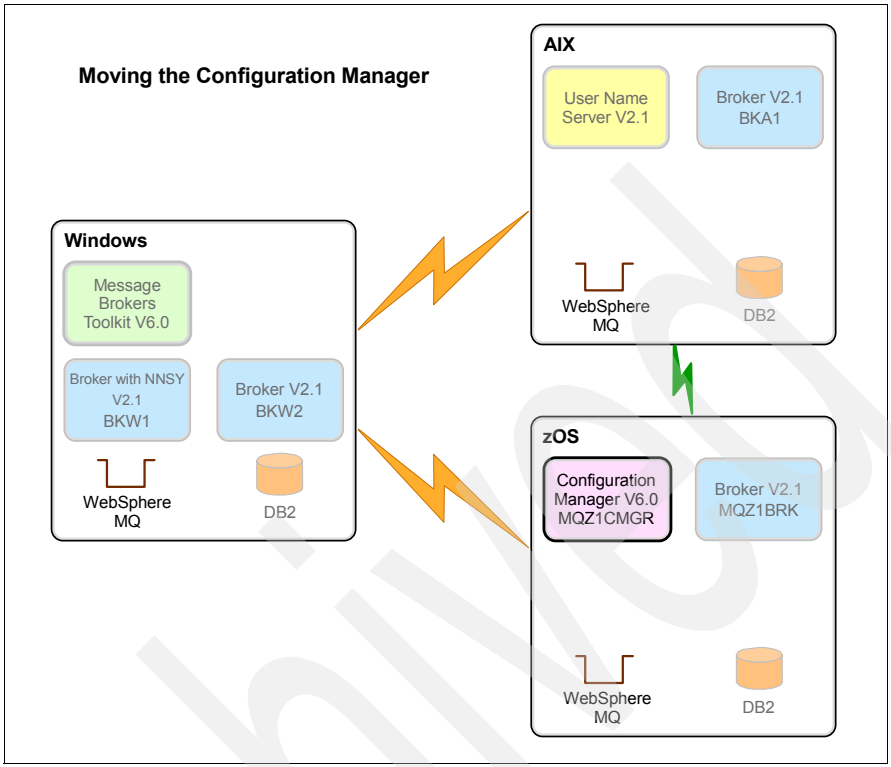


Figure 11-13 Environment after moving the Configuration Manager from Windows to z/OS

To move the Configuration Manager from a Windows system to a z/OS system, perform the following steps:

1. If any Message Brokers Toolkits are open, disconnect from the domain. In the Broker Administration perspective's Domains view, right-click the Configuration Manager connection and select **Disconnect**.
2. If possible, stop all the brokers in the domain by entering the following command:

   ```
   mqsistop
   ```

   Attention: If it is not possible to stop the brokers in the domain because it is a high-availability system, then you need to perform an additional step documented in step 6 on page 213.

3. On the Windows system, open a WebSphere Message Broker V6.0 Command Console by slecting **Start** → **Programs** → **IBM WebSphere**

Message Brokers 6.0 → **Command Console**, and stop the Configuration Manager by entering the following command:

```
mqsistop cm_name
```

4. Start the Event Viewer be selecting **Start** → **Programs** → **Administrative Tools** → **Event Viewer** and then select **Application Log**. When the Configuration Manager has successfully stopped, a warning event message with a source identifier of WebSphere Broker v6000 and event identifier 2002 is displayed.

> **Important:** Ensure that you have stopped the Configuration Manager before attempting to back up the Configuration Manager's internal repository.

5. To create a backup archive of the Configuration Manager's internal repository, enter the following command in the Command Console on the Windows system:

```
mqsibackupconfigmgr cm_name -d directory -a backup_filename
```

> **Note:** The directory specified for the -d command option must exist before the command is entered to write the backup archive to it.

Example 11-10 shows the output from a successful Configuration Manager repository backup.

Example 11-10 Output from the Configuration Manager backup command

```
C:\Program Files\IBM\MQSI\6.0>mqsibackupconfigmgr ConfigMgr -d
C:\backup -a ConfigMgr_backup
BIP1075I: Creating backup archive for Configuration Manager 'ConfigMgr'
in directory 'C:\backup' ...
BIP1017I: A backup archive called 'ConfigMgr_backup.zip' was created
successfully.
BIP8071I: Successful command completion.
```

6. Create a new Configuration Manager component on the z/OS system.

Creating a Configuration Manager on z/OS

For information about the steps required to create a Configuration Manager on z/OS, refer to the WebSphere Message Broker Information Center, under the topic **Configuring the broker domain** → **Configuring broker domain components** → **Creating a Configuration Manager** → **z/OS**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae24280_.htm

Note: This section assumes that the WebSphere Message Broker V6.0 Configuration Manager component has been installed on the z/OS system, as described in the installation guide. You can find this guide on the Web at:

http://www.elink.ibmlink.ibm.com/public/applications/publications/cgibin/pbi.cgi?¶¶CTY=US&FNC=SRX&PBL=GC34-6621-00

Ensure that the mandatory prerequisite APAR OA11699, referenced in Preventive Service Planning (PSP), has been applied to the WebSphere Message Broker installation. This allows the output from the broker jobs to be viewed in the job logs.

First, gather information about the current system. This is essential for configuring the Configuration Manager's job control language (JCL) in later stages. Then allocate a new partitioned data set extended (PDSE) for the Configuration Manager's JCL and configuration files. Finally, copy the template JCL provided with WebSphere Message Broker V6.0 to the new data set and customize it for the local system.

Collect required system information

Table 11-1and Table 11-2 can be used to note the local system's information for use in the subsequent instructions, which are detailed in the first two steps of creating the Configuration Manager on z/OS.

Table 11-1 summarizes the installation information required.

Table 11-1 Installation information

Description	JCL variable	Redbook variables	Your installation variable
Fully qualified name of the product's SBIPPROC data set	N/A	<hlq>.SBIPPROC	
Fully qualified name of the product's SBIPSAMP data set	N/A	<hlq>.SBIPSAMP	

Table 11-2 summarizes the component information required.

Table 11-2 Configuration Manager runtime information for JCL customization

Description	JCL variable	Redbook variables	Your installation variable
File system directory where the product has been installed	++INSTALL++	/usr/lpp/mqsi/V6M0R0	
HFS directory where the Configuration Manager is to exist	++COMPONENTDIRECTORY++	/var/wmqi/MQZ1CMGR	
Configuration Manager name	++COMPONENTNAME++	MQZ1CMGR	
The Configuration Manager's user ID HFS home directory	++HOME++	/u/MQZ1CMGR	
mqsicreateconfigmgr options	++OPTIONS++	‘’ Note: No options were specified, so two single quotation marks are specified.	
Locale of environment where commands are run by submitting JCL	++LOCALE++	C	
Time zone of environment where commands are run by submitting JCL	++TIMEZONE++	GMT0BST	
Location of Java installation	++JAVA++	/usr/lpp/java/J1.4	
LE high-level qualifier	++LEHLQ++	PP.ADLE370.ZOS150	
WebSphere MQ high-level qualifier	++WMQHLQ++	MQM.V531	
queue manager associated with the broker	++QUEUEMANAGER++	MQZ1	
The data set where all JCL relevant to the Configuration Manager is saved	++COMPONENTDATASET++	MQSI6.MQZ1CMGR.CNTL	

Description	JCL variable	Redbook variables	Your installation variable
Name of the Started Task JCL (can be a maximum of 8 characters)	++STARTEDTASKNAME++	MQZ1CMGR	
Profile name	++COMPONENTPROFILE++	BIPCPROF	
Location of the XML toolkit	++XMLTOOLKIT++	/usr/lpp/ixm/IBM/xml4c-5_5	

Important: WebSphere Message Broker V6.0 requires Version 1.8 of XML Parser for z/OS. See the XML Toolkit Web site for more information about how to obtain the latest version:

http://www-03.ibm.com/servers/eserver/zseries/software/xml/cparser/install.html

Create the Configuration Manager PDSE

This new data set stores the JCL and configuration members for the Configuration Manager. The data set must exist so that the template JCL and template configuration members can be copied into it. Each Configuration Manager requires either a PDSE or a partitioned data set (PDS). However, a PDSE is preferable to a PDS because free space is available without the need to compress the data set.

For up-to-date information about how to create the Configuration Manager's component data set, refer to the WebSphere Message Broker Information Center, under the topic **Configuring the broker domain → Configuring broker domain components → Creating a Configuration Manager → z/OS → Creating the Configuration Manager PDSE**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae24180_.htm

We have taken the following instructions from the topic and replicated them for convenience.

Create the Configuration Manager's component data set, for example, using option 3.2 on Interactive System Productivity Facility (ISPF). The name of the PDSE must be the same as the JCL variable ++COMPONENTDATASET++ collected in "Collect required system information" on page 201.

Allocate a data set with:

- Eight directory blocks
- 15 tracks (or 1 cylinder) of 3390 DASD with a record format of fixed blocked 80
- A suitable block size (for example, 27920)
- Data set type of LIBRARY

Create the Configuration Manager home directory

Create the home and component directories in the Hierarchical File System (HFS), if they do not already exist. These are defined as ++HOME++ and ++COMPONENTDIRECTORY++ in “Collect required system information” on page 201.

Ensure that the Configuration Manager’s started task user ID and the Configuration Manager system administrator’s user ID have read, write, and execute permissions to both the ++COMPONENTDIRECTORY++ and the ++HOME++ directory.

Customize the Configuration Manager component data set

For up-to-date information about how to customize the Configuration Manager’s component data set, refer to the WebSphere Message Broker Information Center, under the topic **Configuring the broker domain → Configuring broker domain components → Creating a Configuration Manager → z/OS → Customizing the Configuration Manager component data set**, on the Web at:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae24230_.htm`

We have taken the following instructions from the topic and replicated them for convenience.

Copy the members in the PDSEs SBIPPROC and SBIPSAMP to the new Configuration Manager data set. Use the Data Set Utility panel, usually option 3.2 in ISPF, to do the following:

- Copy the BIPCPROF (Configuration Manager profile) JCL from <hlq>.SBIPSAMP to ++COMPONENTDATASET++.
- Copy the following JCL from <hlq>.SBIPPROC to ++COMPONENTDATASET++:
  - BIPALMQ: Job to alter WebSphere MQ table spaces and tables
  - BIPBUCM: Job to enter the **`mqsibackupconfigmgr`** command
  - BIPCHCM: Job to enter the **`mqsichangeconfigmgr`** command

- BIPCRACL: Job to create an access control list (ACL) entry by entering the **mqsicreateaclentry** command
- BIPCRCM: Job to create a Configuration Manager
- BIPCREG: Job to enter the **mqsicreateexecutiongroup** command
- BIPDLACL: Job to delete an ACL entry by entering the **mqsideleteaclentry** command
- BIPDLCM: Job to enter the **mqsideleteconfigmgr** command
- BIPDLEG: Job to enter the **mqsideleteexecutiongroup** command
- BIPDPLY: Job to enter the **mqsideploy** command
- BIPEDIT: Job to edit macro for customization
- BIPFMLG: Job to format the log
- BIPGEN: Job to generate the ENVFILE
- BIPLIACL: Job to list the ACL entries by entering the **mqsilistaclentry** command
- BIPLIST: Job to enter the **mqsilist** command
- BIPRELG: Job to read the logs
- BIPRSCM: Job to enter the **mqsirestoreconfigmgr** command
- BIPSPMF: Job to enter the **mqsistopmsgflow** command
- BIPSTMF: Job to enter the **mqsistartmsgflow** command
- BIPCMGRP: Job for started task

Important: Customize the following files for your Configuration Manager:

- Rename BIPEDIT to a unique name that identifies it to the current component, for example, MQZ1EDCM.
- Rename BIPCMGRP to the same as ++STARTEDTASKNAME++. In this example, MQZ1CMGR.

Customize the Configuration Manager JCL

Customize the JCL templates that have been copied to the Configuration Manager's component data set to the environment on which they run. The ISPF macro BIPEDIT, which in this example has been renamed to MQZ1EDCM**,** can be used to replace the JCL variables in all of the members of the Configuration Manager's component data set. To use this macro, perform the following steps:

1. Edit the renamed BIPEDIT member and replace the sample parameters with the variables collected in "Collect required system information" on page 201.

For instance, the ++INSTALL++ JCL parameter will look like this in the renamed BIPEDIT member:

```
"change ++INSTALL++ /usr/lpp/mqsi/V6ROMO                    all"
```

Important: Not all the variables in the BIPEDIT member are relevant for the Configuration Manager (for example, the database variables are not required).

2. After the renamed BIPEDIT member has been correctly configured, run it against the members of the component data set to replace the JCL variables in each member. To activate the macro for execution, run the following Time Sharing Option (TSO) command in the ISPF panel, option 6:

   ```
   ALTLIB ACTIVATE APPLICATION(EXEC) DA('++COMPONENTDATASET++')
   ```

   This command is active for the local ISPF session for which it was entered.

   Note: If you have split screen sessions, the other sessions will not be able to use this command. If you use ISPF option 6 to enter the command, use ISPF option 3.4 to edit the data set. This enables you to use the **edit** command.

3. After the ISPF macro is active, replace the JCL variables for each member in the Configuration Manager's component data set. For each member of the component data set (*except* the renamed BIPEDIT macro itself):

   a. View or Edit the member.

      Tip: We recommend that you use View instead of Edit on a member until any problem in the ISPF macro is resolved. Alternatively, you can Cancel the Edit session instead of using Save.

   b. Run the renamed BIPEDIT exec by typing its name on the command line (for example, MQZ1EDCM) to replace the JCL variables with the desired values.

   c. Exit and Save the changes, if they have been successful.

   The JCL members in the component data set are now ready to be run.

Generate the Configuration Manager's ENVFILE

The Configuration Manager requires an ENVFILE file, an environment file in which the Configuration Manager stores all of its environment variables. When the components are started, they read the ENVFILE to configure their

environment. To generate this file, submit the BIPGEN job in the Configuration Manager's component data set. The ENVFILE is then created in the ++HOME++ directory on the HFS.

Make sure that the BIPGEN job run is successful and that the ENVFILE is created in the ++HOME++ directory. If any errors occur, resolve them and run the BIPGEN job again. Continue to run the job and rectify errors until the ENVFILE is successfully created. The Configuration Manager cannot be created until an ENVFILE has been generated by BIPGEN.

Create the Configuration Manager

To create the Configuration Manager component, after completing the steps detailed in the previous section, submit the BIPCRCM job.

Note: The user ID submitting the BIPCRCM command must have the appropriate Configuration Manager ACL entries and WebSphere MQ authorities. For a summary of the z/OS authorities required, refer to the topic **Reference → Operations → z/OS specific information → z/OS customization → Customization tasks and roles → Summary of required access,** on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae14040_.htm

For additional information about security considerations in z/OS, refer to the WebSphere Message Broker Information Center, under the topic **Security → Setting up z/OS security** on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae14030_.htm

Example 11-11 shows the output from the successful creation of a WebSphere Message Broker V6.0 Configuration Manager.

Example 11-11 Output from the WebSphere Message Broker V6.0 Configuration Manager creation

```
ImbMqZosSystemCommand::Connect QueueManager = MQZ1 CompCode = 0
ReasonCode = 0
ImbMqZosSystemCommand::getReplyQueue CompCode = 0 ReasonCode = 0
ImbMqZosSystemCommand::Put CompCode = 0 ReasonCode = 0
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQN205I   COUNT=       2, RETURN=00000000, REASON=00000000
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQ9022I -MQZ1 CSQMAQLC ' DEFINE QLOCAL' NORMAL COMPLETION
```

```
ImbMqZosSystemCommand::Put CompCode = 0 ReasonCode = 0
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQN205I   COUNT=       2, RETURN=00000000, REASON=00000000
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQ9022I -MQZ1 CSQMAQLC ' DEFINE QLOCAL' NORMAL COMPLETION
ImbMqZosSystemCommand::Put CompCode = 0 ReasonCode = 0
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQN205I   COUNT=       2, RETURN=00000000, REASON=00000000
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQ9022I -MQZ1 CSQMAQLC ' DEFINE QLOCAL' NORMAL COMPLETION
ImbMqZosSystemCommand::Put CompCode = 0 ReasonCode = 0
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQN205I   COUNT=       2, RETURN=00000000, REASON=00000000
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQ9022I -MQZ1 CSQMAQLC ' DEFINE QLOCAL' NORMAL COMPLETION
ImbMqZosSystemCommand::Put CompCode = 0 ReasonCode = 0
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQN205I   COUNT=       2, RETURN=00000000, REASON=00000000
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQ9022I -MQZ1 CSQMAQLC ' DEFINE QMODEL' NORMAL COMPLETION
ImbMqZosSystemCommand::Put CompCode = 0 ReasonCode = 0
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQN205I   COUNT=       2, RETURN=00000000, REASON=00000000
ImbMqZosSystemCommand::Get CompCode = 0 ReasonCode = 0
CSQ9022I -MQZ1 CSQMACHL ' DEFINE CHANNEL' NORMAL COMPLETION
BIP8071I: Successful command completion.
ImbMqZosSystemCommand::Disconnect CompCode = 0 ReasonCode = 0
```

Copy the started task JCL

Copy the renamed BIPCMGRP member, in this example MQZ1CMGR, to the SYS1.PROCLIB concatenation. The Configuration Manager is now ready to be started.

Start the Configuration Manager

To start or stop a Configuration Manager, enter the command from System Display and Search Facility (SDSF), NetView®, or other products that provide access to the console. Example 11-12 shows the commands for SDSF; they have the / command prefix, which may be different in other tools.

From SDSF, start the component, for example, /S MQZ1CMGR.

Check the job log for the initiated task to ensure that the component has started successfully, as illustrated in Example 11-12.

Example 11-12 Output from the successfully initiated Configuration Manager

```
+BIP9141I MQZ1CMGR 0 THE COMPONENT WAS STARTED. :
ImbControlService(744)
+BIP9108I MQZ1CMGR 0 BROKER SERVICE VALUE IS IMBSERV.V6ROM00.GOLD.....
: ImbControlService(749)
+BIP2001I MQZ1CMGR 0 THE WEBSPHERE MESSAGE BROKERS SERVICE HAS STARTED
AT VERSION 6000; PROCESS ID 852169. : ImbControlService(773)
+BIP8255I MQZ1CMGR 0 USER NAME SERVICES ARE DISABLED. :
ImbSecClient(101)
+BIP1003I MQZ1CMGR 0 THE CONFIGURATION MANAGER IS AVAILABLE FOR USE. :
ImbConfigMgrAgent(694)
```

However, if the component fails to start correctly, use the error information in the job log to rectify the problem. Ensure that you have performed all of the prerequisite steps in this section and that they were successful.

Note: On distributed platforms, create the Configuration Manager by entering the following command:

```
mqsicreateconfigmgr cm_name -i userid -a password -q cm_qm
```

Refer to step 6 on page 405 in 21.2.1, “Moving the Configuration Manager’s platform after migrating to WebSphere Message Broker V6.0” for detailed instructions.

You can create the new Configuration Manager with the same name as the Configuration Manager being replaced. However, a Configuration Manager cannot have the same name as a broker or another Configuration Manager on the same system.

Consider the name of the queue manager used for the new Configuration Manager. It is important to maintain uniqueness in the WebSphere MQ network for communication among the domain components, so use a different queue manager name from the Configuration Manager that is moved.

Example 11-12 utilizes an existing queue manager, MQZ1, that is shared with an existing broker. As with creating the Configuration Manager in previous versions, it is possible to have a separate queue manager for the Configuration Manager. On z/OS, this must exist before the command is run. On distributed platforms, the command can create the queue manager specified in the command.

Create and start a WebSphere MQ channel initiator with a TCP/IP listener on the Configuration Manager’s queue manager. In the environment that is demonstrated here, the Configuration Manager listener runs on port 1450.

1. On the z/OS system, configure the WebSphere MQ to enable the new WebSphere Message Broker V6.0 Configuration Manager to communicate with the broker domain. Then, on each of the broker systems, configure the broker's queue manager to communicate with the new Configuration Manager. This involves configuring the appropriate channels and transmission queue definitions. Start the channels to confirm that successful communication among the WebSphere MQ components is possible.
2. Copy the Configuration Manager backup archive file created on Windows to an appropriate directory on the z/OS system. In the migration that is demonstrated here, the ConfigMgr_backup.zip file is created in the C:\backup directory and is moved using binary FTP to the z/OS system and stored in the home directory of the Configuration Manager component /u/MQZ1CMGR.

   Attention: You must stop the Configuration Manager before proceeding to restore the backup file into the repository.

   From SDSF, stop the component by entering /P MQZ1CMGR.

3. On the z/OS system, restore the contents of the Windows Configuration Manager repository backup into the empty repository of the newly created Configuration Manager:

   a. Edit the BIPRSCM job to have the correct ++DIRECTORYPATH++ and ++ARCHIVENAME++ variable values.

   b. Run the BIPRSCM job.

   Example 11-13 shows the output from the successful restoration of the Configuration Manager repository.

Example 11-13 Output from the Configuration Manager restore command

```
BIP1079I: Replacing repository for 'MQZ1CMGR' with archive
'/u/mqz1brk/ConfigMgr_backup.zip'...
BIP1171I: Verifying the restored repository...
BIP8071I: Successful command completion.
```

Note: On distributed systems, use the following command to restore the Configuration Manager from a backup:

`mqsirestoreconfigmgr` *`cm_name`* `-d` *`directory`* `-a` *`backup_filename`*

Refer to step 8 on page 407 in 21.2.1, "Moving the Configuration Manager's platform after migrating to WebSphere Message Broker V6.0" for detailed instructions.

4. On the z/OS system, start the Configuration Manager by entering the following command from SDSF:

   ```
   /S MQZ1CMGR
   ```

   Check the job log for the initiated task to ensure that the component has started successfully. When successful, the final entry BIP1003I confirms that the Configuration Manager is available for use, as illustrated in Example 11-11 on page 207.

   Note: Since a broker can be managed only by a single Configuration Manager, change the Configuration Manager on the system you have moved from so that it does not start automatically. After verifying whether the moved Configuration Manager is working correctly, delete it by entering the following command:

   ```
   mqsideleteconfigmgr
   ```

5. In order to associate all the brokers in the domain with the new Configuration Manager, a complete topology deployment is required. This ensures that the brokers publish their status messages to the correct queue manager.

   a. On the z/OS system, edit the BIPDPLY job by entering the following command:

      ```
      mqsideploy -i "" -p 0 -q cm_qm -l -m
      ```

   b. Then, run the BIPDPLY job.

      Alternatively, enter the following console command from SDSF:

      ```
      /F cm_name,DP parms
      ```

Attention: If you enter the **mqsideploy -i** *cm_name* **-p** *port* **-q** *cm_qm* **-l -m** command via the BIPDPLY job, the command attempts to run **mqsideploy** using WebSphere MQ Java client code. This is not allowed on z/OS and the following error occurs:

```
BIP1046E: Unable to connect with the Configuration Manager's
queue manager
```

Depending on the version of WebSphere MQ that is used, the reported reason codes differ:

- On WebSphere MQ V5.3.1:

  ```
  2012  0x000007dc  MQRC_ENVIRONMENT_ERROR
  ```

- On WebSphere MQ V6.0:

  ```
  2298  0x000008fa  MQRC_FUNCTION_NOT_SUPPORTED
  ```

Therefore, connecting to the local Configuration Manager with the parameters **-i ""** and **-p 0** forces the command into a local mode.

After step 5 on page 211 is completed, all the brokers in the domain are managed by the new Configuration Manager on the new system.

Important: WebSphere Message Broker uses ACLs to govern which users and groups can manipulate objects within the Configuration Manager and Message Brokers Toolkit. This is discussed in 11.3.4, “Access Control Lists” on page 224 and documented in the WebSphere Message Broker Information Center, under the topic **Reference** → **Security** → **ACL permissions**, on the Web at:

```
http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp
?topic=/com.ibm.etools.mft.doc/ap12520_.htm
```

When a user creates or starts a Configuration Manager, an entry is made in the ACL to grant this user ID (on any machine) full authority. This is the user ID of the initiated task. In order to list the ACL entries, run the BIPLIACL job. This runs under the TSO user ID of the user submitting the BIPCRACL job, which must be granted appropriate authority by running the BIPCRACL job. Alternatively, use the console command **/F cm_name,LA** that runs under the Configuration Manager’s started task user ID and therefore always has the authority.

Example 11-14 shows the output from a successful topology deployment.

Example 11-14 Output from the topology deployment command

```
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1055I: Deploying pub/sub neighbors (topology) configuration...
BIP1099I: Deployment completed successfully.

All brokers affected by the deployment request have successfully
processed the message and have updated their configurations
accordingly.
```

Attention: If you have stopped the brokers in the domain at step 2 on page 199, start them again by using the **mqsistart** command as soon as the topology deployment is initiated. This allows the deployments to be processed.

6. If it is not possible to stop the brokers in the domain in step 2 on page 199, then the brokers may send status messages to the original Configuration Manager. Any status change events that may occur between stopping the original Configuration Manager in step 3 on page 199 and the complete deployment in step 5 on page 211, are sent to the original Configuration Manager's queue manager's SYSTEM.BROKER.ADMIN.REPLY queue. Therefore, check this queue. If there are messages in it, they must be manually transferred, maintaining the headers, to the *new* Configuration Manager's queue manager's SYSTEM.BROKER.ADMIN.REPLY queue. To complete this manual transfer, use a utility such as SupportPac IH03 or SupportPac MO71. You can find details about SupportPac IH03 on the Web at:

   http://www-1.ibm.com/support/docview.wss?rs=171&uid=swg24000637&loc=en_US&cs=utf-8&lang=en

   For details about SupportPac MO71, go to the Web at:

   http://www-1.ibm.com/support/docview.wss?rs=171&uid=swg24000142&loc=en_US&cs=utf-8&lang=en

11.2.2 Verifying the moved Configuration Manager

After completing the Configuration Manager move to another platform, it is important to connect to it through the Message Brokers Toolkit to verify that the component is running correctly.

In order to connect to a Configuration Manager that is not on the same system as the Message Brokers Toolkit, authorization for the user ID of the Message Brokers Toolkit user is required on the remote system. Create an appropriate ACL entry for the Message Brokers Toolkit user ID on the remote Configuration Manager system. On the z/OS system, edit the BIPCRACL job to enter the following command:

```
mqsicreateaclentry cm_name -u userid -m machine -x F -p
```

Alternatively, enter the following console command from SDSF:

```
/F cm_name,CA parm
```

For example, to grant full authority to all the resources for the Message Brokers Toolkit user ID cz50365 from machine PC2197 on the z/OS Configuration Manager MQZ1CMGR, enter the following command:

```
mqsicreateaclentry MQZ1CMGR -u cz50365 -m PC2197 -x F -p
```

Because a new Configuration Manager has been created on a different system, you must either change the existing domain connection or create a new domain connection. We recommend that you create a new domain connection. Then delete the old one when the original Configuration Manager is deleted. To create a new domain connection, perform the following steps:

1. In the Message Brokers Toolkit, open the Broker Administration perspective. Go to the Domains view, right-click it, and select **New** → **Domain**.
2. In the dialog that appears, enter the Configuration Manager's queue manager Name, Host, and Port, as illustrated in Figure 11-14. Then select **Next**.

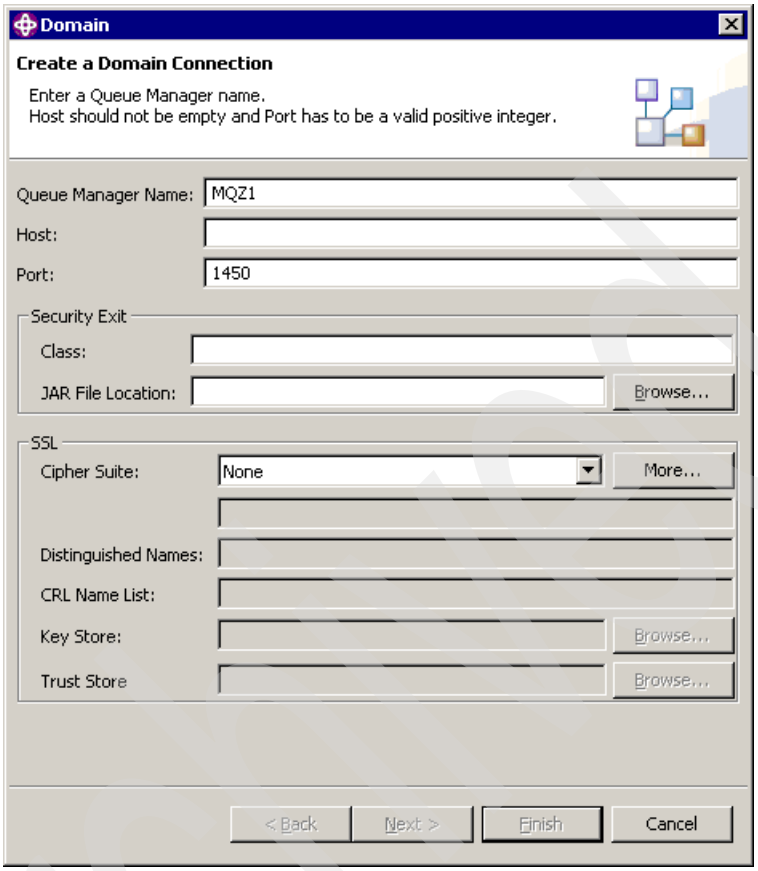


Figure 11-14 Creating a new domain connection to the moved Configuration Manager

3. When the Configuration Manager connection has been established, another dialog appears. Enter the Connection name and select **Finish**. The Domains view shows the new connection to the Configuration Manager, named MQZ1CMGR on the z/OS machine. When the broker topology is expanded (as illustrated in Figure 11-15), the domain of brokers, previously controlled by the Windows Configuration Manager named ConfigMgr, is shown in the state it was prior to stopping the Windows Configuration Manager.

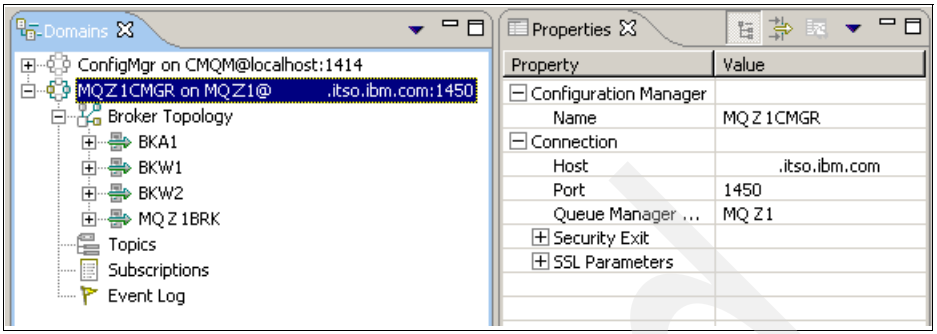


Figure 11-15 Moved Configuration Manager's broker domain in the Message Brokers Toolkit

Troubleshooting

If the Message Brokers Toolkit user ID does not have an appropriate ACL entry on the Configuration Manager system and the user tries to connect to the Configuration Manager, the BIP0889E error appears, as shown in Figure 11-16.

Figure 11-16 Error: Toolkit userID without an appropriate ACL on the Configuration Manager

11.2.3 Migrating the Configuration Manager and moving to another platform in one step

For information about the alternative process of migrating and moving a Configuration Manager in a single step, refer to the WebSphere Message Broker Information Center, under the topic **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 → Migrating components on different computers → Migrating the Configuration Manager → To a different computer that has DB2 installed**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23939_.htm

The WebSphere MQ Integrator V2.1 Configuration Manager component still remains on the original system after migrating and moving the Configuration Manager in a single step. After completing the topology deployment to associate all the brokers in the domain with the WebSphere Message Broker V6.0 Configuration Manager on the new system, you cannot use the WebSphere MQ Integrator V2.1 Configuration Manager to manage the broker domain. Therefore, to remove the WebSphere MQ Integrator V2.1 Configuration Manager component, enter the following command in the Windows command prompt on the original system:

```
mqsideleteconfigmgr
```

11.3 Configuration Manager enhancements

This section briefly discusses some of the enhancements of the WebSphere Message Broker V6.0 Configuration Manager component.

11.3.1 Configuration Manager runtime changes

This section describes the runtime changes in the Configuration Manager.

- The Configuration Manager runtime component is now supported on platforms other than just Windows. It is supported on the same platforms as the broker runtime such as AIX, HP-UX, Linux (x86 platform), Linux (zSeries platform), Solaris, Windows, and z/OS. For details of the WebSphere Message Broker V6.0 system requirements, see the information available at the following Web site:

  http://www.ibm.com/software/integration/wbimessagebroker/requirements/index.html

- There is no longer a restriction to have only a single Configuration Manager on a system. You can create multiple named Configuration Managers. For details about the available command options, refer to the WebSphere Message Broker Information Center, under the topic **Reference** → **Operations** → **Commands** → **Runtime commands** → **mqsicreateconfigmgr,** on the Web at:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an23000_.htm

  The names of the Configuration Managers must be unique within the broker domain. A Configuration Manager cannot have the same name as another Configuration Manager or broker.

  Note: A broker can be managed only by a single Configuration Manager, but you can administer multiple domains from a single Message Brokers Toolkit.

  A Configuration Manager cannot share a host queue manager with another Configuration Manager. However, it can share a host queue manager with one broker in the broker domain.

- The Configuration Manager no longer requires a DB2 Universal Database and now maintains the configuration details in an internal repository.

  Note: Following the migration, the WebSphere MQ Integrator V2.1 Configuration Manager's DB2 Universal Database still exists unmodified. The WebSphere Message Broker V6.0 Configuration Manager uses an internal repository instead of a database. So after a successful validation of the migrated Configuration Manager, you can remove the database.

11.3.2 The Configuration Manager Proxy

The Configuration Manager Proxy is a comprehensive Java interface that allows users to control broker domains programmatically. It can be used to write applications to interact with the Configuration Manager to enquire about and change the status of the broker domain. For more information about the Configuration Manager Proxy, refer to the WebSphere Message Broker Information Center, under the topic **Developing applications** → **Developing applications using the CMP** → **Configuration manager Proxy**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae20620_.htm

On Windows, launch the Java application programming interface documentation from **Start → Programs → IBM WebSphere Message Brokers 6.0 → Java Programming APIs → Configuration Manager Proxy API Documentation**.

For information about developing applications that use the Configuration Manager Proxy Java API, refer to the topic **Developing applications → Developing applications using the CMP** on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae33010_.htm

11.3.3 The Configuration Manager Proxy API Exerciser

The Configuration Manager Proxy API Exerciser is a sample application shipped with WebSphere Message Broker V6.0 . It demonstrates the capabilities of the Configuration Manager Proxy by enabling the user to invoke most of the Configuration Manager Proxy functions from within a GUI.

Environment setup

In order to use the sample application on platforms other than Windows, you must set up the proper environment. Include the following Configuration Manager Proxy and the WebSphere MQ classes for Java jars in the CLASSPATH environment variable:

```
<install_directory>/sample/ConfigManagerProxy/ConfigManagerProxySamples
.jar
<install directory>/classes/ConfigManagerProxy.jar
com.ibm.mq.jar
connector.jar
```

Also, your Java development directory must be on the CLASSPATH. If you want the Exerciser to save the user settings, add the current directory to the CLASSPATH. The user settings are saved to the current directory and are subsequently loaded from the CLASSPATH via Java's ResourceBundle mechanism. For details on configuring the environment, refer to the WebSphere Message Broker Information Center, under the topic **Developing applications → Developing applications using the CMP → Configuring an environment for developing and running Configuration Manager Proxy applications**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae33020_.htm

Using the application

For information about the capabilities of the Configuration Manager Proxy API Exerciser sample application, refer to the WebSphere Message Broker

Information Center, under the topic **Developing applications Developing applications using the CMP → The CMP samples → Running the Configuration Manager Proxy API Exerciser sample**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae33350_.htm

A brief introduction to the basic usage of the sample application, demonstrating topics such as connecting to the Configuration Manager, discovering the attributes of domain objects, and changing the status of the broker domain, is given in the following section.

To launch the application:

- On Windows, select **Start → Programs → IBM WebSphere Message Brokers 6.0 → Java Programming APIs → Configuration Manager API Proxy Exerciser**.
- On other platforms, enter the following command:

  Installation_Directory\sample\ConfigManagerProxy\StartConfigManagerProxyExerciser

When the application starts, right-click the **[Right-click to connect]** option in the top left pane and select the **Connect to Configuration Manager...** option, as illustrated in Figure 11-17.

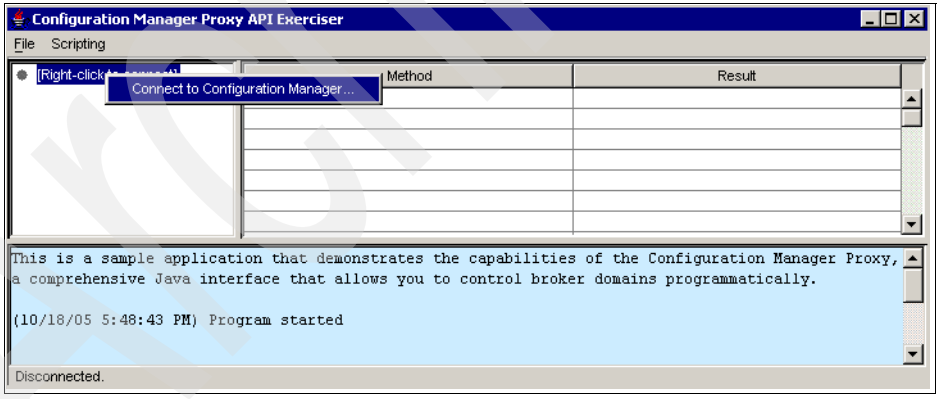


Figure 11-17 Connect to Configuration Manager Proxy API Exerciser sample application

Complete the Configuration Manager connection parameters in the window named Connect to Configuration Manager..., such as the Configuration Manager's host name, port number, and queue manager (as illustrated in Figure 11-18). Then select **Submit**.

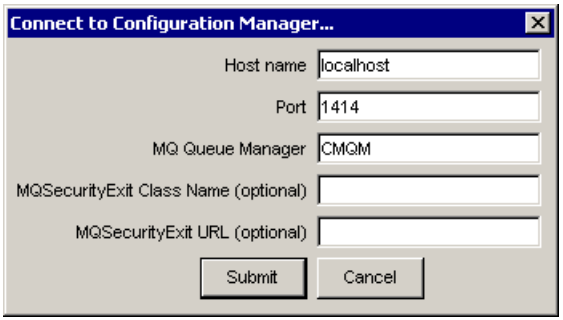


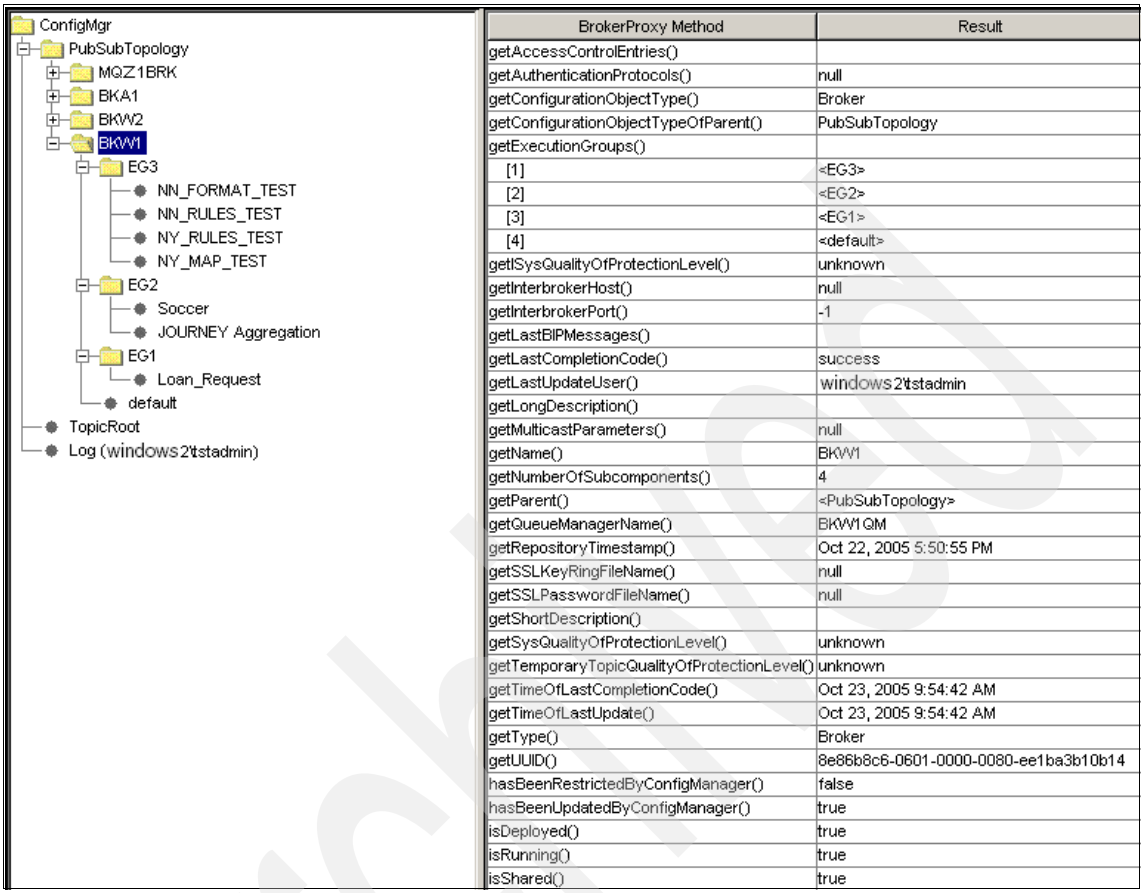
Figure 11-18 Configuration Manager connection parameters

The blue pane at the bottom of the application reports registering for all the Configuration Manager's domain objects such as the brokers, execution groups, and message flows. After the four lines (illustrated in Example 11-15) appear as the output, the Configuration Manager Proxy API Exerciser is ready to be used to manipulate the domain objects.

Example 11-15 Successfully connected to the Configuration Manager Proxy API Exerciser

```
The Configuration Manager's object hierarchy is available for use.
Click on an object in the hierarchy to select it and display its
properties.
Right-click a selected object to manipulate it.
<---- cmp.exerciser.ClassTesterForConfigManagerProxy.testConnect
```

The application now displays a hierarchical view of the domain of brokers that the Configuration Manager manages. Selecting a domain object such as a broker, execution group, or message flow in the top left pane causes the right pane to display the known information about the selected object. To find out the attributes of a broker, select the broker in the left pane and the right pane displays the attributes, including its list of execution groups, queue manager, and status, as illustrated in Figure 11-19.



BrokerProxy Method	Result
getAccessControlEntries()	
getAuthenticationProtocols()	null
getConfigurationObjectType()	Broker
getConfigurationObjectTypeOfParent()	PubSubTopology
getExecutionGroups()	
[1]	<EG3>
[2]	<EG2>
[3]	<EG1>
[4]	<default>
getISysQualityOfProtectionLevel()	unknown
getInterbrokerHost()	null
getInterbrokerPort()	-1
getLastBIPMessages()	
getLastCompletionCode()	success
getLastUpdateUser()	windows2\tstadmin
getLongDescription()	
getMulticastParameters()	null
getName()	BKW1
getNumberOfSubcomponents()	4
getParent()	<PubSubTopology>
getQueueManagerName()	BKW1QM
getRepositoryTimestamp()	Oct 22, 2005 5:50:55 PM
getSSLKeyRingFileName()	null
getSSLPasswordFileName()	null
getShortDescription()	
getSysQualityOfProtectionLevel()	unknown
getTemporaryTopicQualityOfProtectionLevel()	unknown
getTimeOfLastCompletionCode()	Oct 23, 2005 9:54:42 AM
getTimeOfLastUpdate()	Oct 23, 2005 9:54:42 AM
getType()	Broker
getUUID()	8e86b8c6-0601-0000-0080-ee1ba3b10b14
hasBeenRestrictedByConfigManager()	false
hasBeenUpdatedByConfigManager()	true
isDeployed()	true
isRunning()	true
isShared()	true

Figure 11-19 Displays a broker's attributes in the Configuration Manager Proxy API Exerciser sample application

To find out what actions can be performed on a domain object, select the domain object in the left pane of the application and right-click it. A context menu of the available options appears. Figure 11-20 illustrates the context menu options displayed when you right-click a message flow.

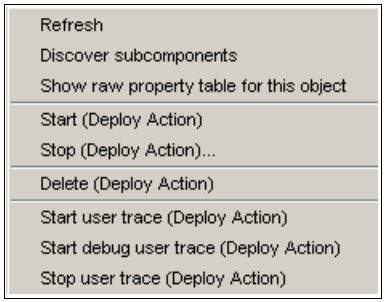


Figure 11-20 Context menu for a message flow in the Configuration Manager Proxy API Exerciser sample application

You can change the status of the broker domain by selecting an action from the menu. For example, if you select **Stop (Deploy Action)...** after right-clicking a message flow, a dialog appears, illustrated in Figure 11-21.

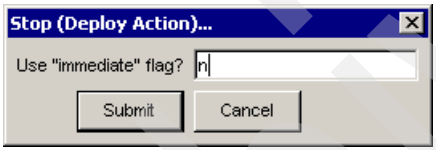


Figure 11-21 Dialog box presented following a Stop (Deploy Action)... command on a message flow in the Configuration Manager Proxy API Exerciser sample application

Selecting **Submit** stops the selected message flow. The right pane of the application displays the attribute `isRunning()` with a result of `false`. The blue pane at the bottom of the application reports the messages sent and received by the Configuration Manager Proxy, as shown in Example 11-16.

Example 11-16 Configuration Manager Proxy messages when a message flow is stopped

```
----> cmp.exerciser.ClassTesterForMiscellaneousActions.testStopMsgFlows(<Loan_Request>, false)
The request was successfully sent to the Configuration Manager.
<---- cmp.exerciser.ClassTesterForMiscellaneousActions.testStopMsgFlows
----> cmp.exerciser.ExerciserAdministeredObjectListener.processActionResponse(...)
affectedObject = Loan_Request
completionCode = success
(Log entry) {message=BIP1522I, detail=BIP1522I: Stop message flow operation successfully
initiated for broker BKW1.
The Configuration Manager has processed the stop message flow request and initiated a stop
message flow operation for the target message flows., source=ConfigMgr, timestamp=Sun Oct 23
10:24:51 BST 2005}
(Reference property) parent.uuid=3f6fd2c6-0601-0000-0080-ee1ba3b10b14
(Reference property) uuid=a8f25e03-0701-0000-0080-b3d73bb07de6
(Reference property) commsmessage.lastinbatch=true
```

```
(Reference property) object.runstate=stopped
(Reference property) configmanagerproxy.osname=Windows 2000
(Reference property) userid=tstadmin
(Reference property) configmanagerproxy.hostname=windows2
(Reference property) commsmessage.configobjecttype=MessageProcessingNodeType
(Reference property) parent.type=ExecutionGroup
(Reference property) commsmessage.operationtype=modify
(Reference property) configmanagerproxy.noeventlog=false
<---- cmp.exerciser.ExerciserAdministeredObjectListener.processActionResponse()
----> cmp.exerciser.ExerciserAdministeredObjectListener.processModify(...)
affectedObject = Loan_Request
(Changed attribute) object.runstate
<---- cmp.exerciser.ExerciserAdministeredObjectListener.processModify()
----> cmp.exerciser.ExerciserAdministeredObjectListener.processModify(...)
affectedObject = Log (windows2\tstadmin)
(New subcomponent) 2056<<BKW1<<2005-10-23 10:24:58<<BKW1<<8e86b8c6-0601-0000-0080-ee1ba3b10b14
<---- cmp.exerciser.ExerciserAdministeredObjectListener.processModify()
----> cmp.exerciser.ExerciserAdministeredObjectListener.processModify(...)
affectedObject = Log (windows2\tstadmin)
(New subcomponent) 4040<<BKW1<<2005-10-23 10:24:58<<EG1<<3f6fd2c6-0601-0000-0080-ee1ba3b10b14
<---- cmp.exerciser.ExerciserAdministeredObjectListener.processModify()
```

In the Message Brokers Toolkit's Domains view, double-click **Event Log** to open the Event Log view. The Configuration Manager Proxy receives back messages confirming the success of the operation. These messages, message identifiers 2056 and 4040, are reported both in the output from the Configuration Manager Proxy (as illustrated in Example 11-16 on page 223) and the Event Log view in the Message Brokers Toolkit (as illustrated in Figure 11-22).

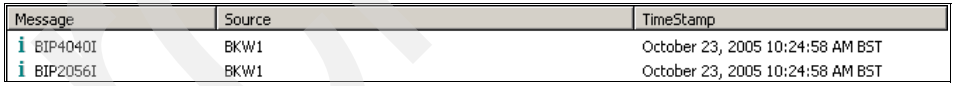


Figure 11-22 Successful configuration change messages received in the Message Brokers Toolkit

The result is also reflected in the Message Brokers Toolkit Domains view and Alerts view.

11.3.4 Access Control Lists

WebSphere Message Broker uses Access Control Lists (ACLs) to govern which users and groups can manipulate objects within the Configuration Manager, from the Message Brokers Toolkit and Configuration Manager Proxy applications. For detailed information on this subject, refer to the WebSphere Message Broker Information Center, under the topic **Reference** → **Security** → **ACL permissions**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ap12520_.htm

A brief introduction to ACL entries and how to manage them is given in the following section.

The ACL entry permissions behave in a hierarchical manner in order to reduce the number required. Therefore, when an ACL entry is made for a given object, permission is granted to that object and inherited by all the objects beneath it in the hierarchy, unless it is overridden by another entry. Figure 11-23 illustrates the ACL hierarchy.

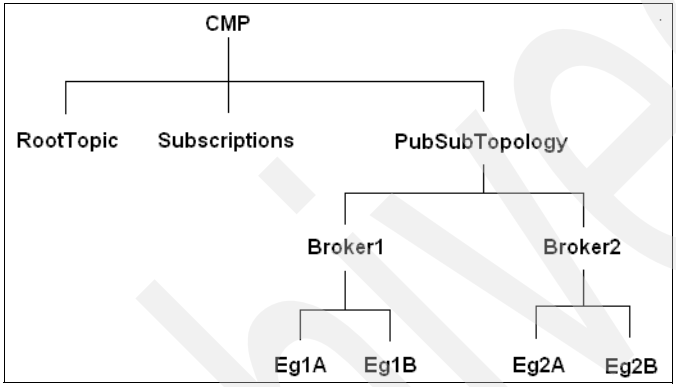


Figure 11-23 ACL entries hierarchy

Managing the Access Control Lists

On the Configuration Manager system, you can manage the ACL entries with a series of commands (as documented in the following section) on the Command Console if the Configuration Manager is on Windows. On other platforms, enter the commands at the command prompt after running the mqsiprofile.

Listing Access Control List entries

To list the ACL entries that have been granted, enter the following command on distributed systems:

```
mqsilistaclentry cm_name
```

On z/OS, run the BIPLIACL job or enter the following console command from SDSF:

```
/F cm_name,LA
```

For details about the other available command options, refer to the WebSphere Message Broker Information Center, under the topic **Reference** →

Operations → **Commands** → **Runtime commands** → **mqsilistaclentry**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an23050_.htm

Example 11-17 illustrates the format of the output that is produced.

Example 11-17 Output from listing the Access Control List entries of a Configuration Manager

```
$ mqsilistaclentry CMA1
BIP1778I: mqbrkrs            -  GROUP  -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
BIP1778I: mqbrdevt           -  GROUP  -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
BIP1778I: mqbrasgn           -  GROUP  -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
BIP1778I: mqbrops            -  GROUP  -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
BIP1778I: <userid>           -  USER   -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
BIP1778I: <domain>\<userid> -  USER   -  F  -  PubSubTopology      -  PubSubTopology
BIP1778I: mqbrtpic           -  GROUP  -  F  -  TopicRoot           -  TopicRoot
BIP1778I: <userid>           -  USER   -  F  -  BKW1                -  Broker
BIP1778I: <userid>           -  USER   -  F  -  BKW1/EG1            -  ExecutionGroup
```

Note: When a user creates or starts a Configuration Manager, an entry is made in the ACL to grant that user ID, on any machine, full authority at the top of the hierarchy, the ConfigManagerProxy object.

Access Control List entries migration

As part of the Configuration Manager migration, the user groups used in previous versions to grant users authorities are addedas ACL entries, as illustrated in the Example 11-17 output.

Deleting Access Control List entries

To delete an ACL entry, enter the following command:

```
mqsideleteaclentry
```

For example, to remove the ACL entry for a user ID from any machine for a particular execution group, enter the following command on distributed systems:

```
mqsideleteaclentry <cm_name> -u <userid> -a -b <broker> -e <exgrp>
```

On z/OS, run the BIPDLACL job. Alternatively, enter the following console command from SDSF:

```
/F <cm_name>,DA <parms>
```

For details about the other available command options, refer to the WebSphere Message Broker Information Center, under the topic **Reference** →

Operations → **Commands** → **Runtime commands** → **mqsideleteaclentry**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an23040_.htm

Creating Access Control List entries

You can grant four different access levels for a user or a group: Full, Deploy, Edit, and View. If the control to be assigned to a user for a particular object should be different from the inherited permission this user will be granted, create a new entry to override the access available to the object and its children. For example, to limit the scope of a user ID on a specific machine to only deploying a specific broker, enter the following command on the distributed systems:

```
mqsicreateaclentry <cm_name> -u <userid> -m <machine_name> -x D -b
<broker>
```

On z/OS, run the BIPCRACL job. Alternatively, enter the following console command from SDSF:

```
/F <cm_name>,CA <parms>
```

Due to the hierarchical nature of the ACL entries, the user ID on the machine specified is implicitly granted deployment access to any execution groups belonging to the broker. The user is also granted a limited form of view access on the Configuration Manager Proxy and Topology objects. As a result of this limited access, the user is able to view the presence of any other brokers in the domain, but is unable to view any details such as brokers' names, or perform any actions on them. For details about the other available command options, refer to the WebSphere Message Broker Information Center, under the topic **Reference** → **Operations** → **Commands** → **Runtime commands** → **mqsicreateaclentry**, on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an23030_.htm

12

Migrating the User Name Server

This chapter describes the process of migrating from WebSphere MQ Integrator V2.1 User Name Server on AIX to WebSphere Message Broker V6.0 User Name Server on AIX. It guides you through the migration of the AIX User Name Server in the scenario described in Chapter 7, “WebSphere MQ Integrator Version 2.1 environment setup” on page 107.

The User Name Server is a component that provides authentication to users and groups performing publish/subscribe operations. It uses Access Control Lists to enable control over the topics. Users can publish/subscribe to particular topics based on the security access provided to them. It requires the WebSphere MQ queue manager, which can be shared with a Configuration Manager and also a broker in the broker domain.

After completing the steps in this chapter, the User Name Server is ready to operate in the migrated domain.

12.1 Migration preparation

This section describes the tasks to be performed before the User Name Server component migration. Figure 12-1 depicts the migration scenario.

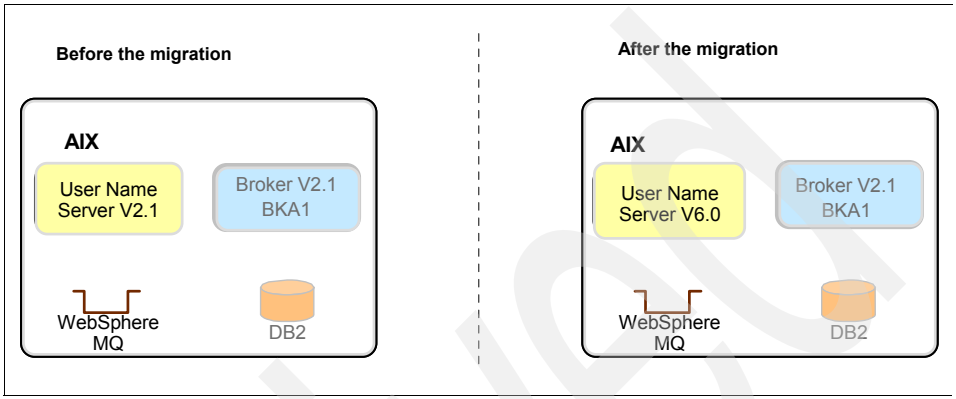


Figure 12-1 User Name Server migration scenario

12.1.1 Reviewing the existing installation

Verify the existing installation for any errors. Before the migration, ensure that the User Name Server is running properly and that there are no error reports.

Reviewing the configuration

To confirm that all important components are running properly, you can use the following commands and check the output.

Log in using the User Name Server user ID and enter the commands from the appropriate platform command prompt:

1. Display the list of queue managers and enter the **dspmq** command. Check the status of all expected queue managers as shown in Example 12-1.

Example 12-1 Displaying the queue managers

```
$ dspmq
QMNAME(UNSQM)
STATUS(Running)
QMNAME(BKA1QM)
STATUS(Running)
```

2. Display the list of Integrator V2.1 components and enter the **mqsilist** command. Check the status of all expected components as shown in Example 12-2.

Example 12-2 Displaying the components

```
$ mqsilist
BIP8099I: BKA1  -  BKA1QM
BIP8099I: UserNameServer  - UNSQM
BIP8071I: Successful command completion.
```

3. Check whether the User Name Server is running as shown in Example 12-3. On AIX, enter the following command:

```
ps -ef | grep UserNameServer
```

Example 12-3 Displaying the User Name Server processes

```
$ ps -ef | grep UserNameServer
tstadmin 24372 60958   0 11:44:13  pts/4  0:00 grep UserNameServer
tstadmin 55140     1   0 11:29:37      -  0:00 bipservice
UserNameServer
tstadmin 59718 55140   0 11:29:37      -  0:00 bipuns UserNameServer
```

Premigration verification

Check the error log files for Integrator V2.1 components, queue manager, and DB2, for any important information or errors.

For details about which error log files to check and how to check them, refer to 16.1.4, “Checking the logs and where to find them” on page 297.

12.1.2 Installing WebSphere Message Broker Version 6.0

Verify or perform the WebSphere Message Broker V6.0 installation.

Perform the installation according to standard installation documentation. The description of the process is outside the scope of this book.

Check installation of the following products:

- Rational Agent Controller (RAC) V6.0.0.1

> **Important:** If you want to install any fix pack for RAC, check the WebSphere Message Broker V6.0 documentation. The WebSphere Message Broker V6.0 requires the correct version of RAC. Any other version is not supported.

- WebSphere Message Broker V6.0 components:

  User Name Server

> **Tip:** We recommend that you install the latest fix pack for WebSphere Message Broker V6.0 and an appropriate RAC version.

12.2 Configuration settings

Some configuration changes have to be made before the component migration starts. This section discusses these changes.

Before you make any changes to the configuration, stop the User Name Server. First, log in with the user ID of the User Name Server. Then enter the following command from the appropriate platform command prompt:

```
mqsistop UserNameServer
```

When the server stops, you see a message like the one in Example 12-4.

Example 12-4 Stopping the User Name Server

```
$ mqsistop UserNameServer
BIP8071I: Successful command completion.
```

12.2.1 Updating environment settings

The environment settings profile for User Name Server user ID changes in Message Broker V6.0, so you must change the user profile as well.

The full path of the new Message Broker sample profile is *broker_install_directory*/6.0/bin/mqsiprofile.

Edit the Message Broker service user ID profile using an editor such as vi:

1. Remove the WebSphere MQ Integrator V2.1 settings. This may be a call to the sample profile, in which case delete the following line:

   ```
   . /usr/opt/mqsi/sample/profiles/profile.aix
   ```

2. Add the line:

   ```
   . broker_install_directory/6.0/bin/mqsiprofile
   ```

The relevant part of the user profile is shown in Example 12-5.

Example 12-5 Sample of the user profile

```
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:/usr
/java131/jre/bin:.

export PATH

. <broker_install_directory>/6.0/bin/mqsiprofile
```

Log off and then log in again as the User Name Server user ID to activate these changes.

Important: If Integrator V2.1 and Message Broker V6.0 need to coexist, then you cannot use both versions of the Message Broker profile in the user profile at the same time. Leave the user profile without them and enter the appropriate version of Message Broker profile manually. You need to use an appropriate version of the profile to start each User Name Server.

12.3 User Name Server migration

This section contains the component User Name Server migration steps for distributed platforms. For more details on how to perform migration on z/OS, refer to the WebSphere Message Broker V6.0 Information Center, under the topic **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 → Migrating components on different computers → Migrating the User Name Server → z/OS** on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23846_.htm

12.3.1 Before starting

Before migrating a User Name Server, ensure that it stops and that there are no errors in the operating system application log.

12.3.2 Migrating the User Name Server

To migrate a User Name Server on distributed operating systems to Message Broker V6.0 at the same location, the following steps are needed.

Log in as the User Name Server user ID and enter the commands from an appropriate platform command prompt:

1. Stop the broker and the Configuration Manager connected to the User Name Server (not mandatory).

   Note: If the broker and the Configuration Manager do not stop during the migration of the User Name Server, the process of their security requests does not take place until the User Name Server migration is complete and the User Name Server restarts.

2. Ensure that the User Name Server is not running.
3. Perform the appropriate environment settings as shown in Example 12-6.

Example 12-6 Running the Message Broker V6.0 profile

```
$ . <broker_install_directory>/6.0/bin/mqsiprofile
```

4. Enter the following Message Broker V6.0 command to perform a premigration check on the User Name Server:

   ```
   mqsimigratecomponents -c UserNameServer
   ```

   Note: This command fails; the defect is expected to be fixed in a future fix pack.

5. Enter the following Message Broker V6.0 command to migrate the User Name Server to Message Broker V6.0:

   ```
   mqsimigratecomponents UserNameServer
   ```

   Example 12-7 contains the complete command output. You must observe a similar successful command output when you migrate your User Name Server.

Example 12-7 Migrating the User Name Server

```
$ mqsimigratecomponents UserNameServer
BIP8675I: UserNameServer 'UserNameServer' (Version 2.1) specified for
migration.
BIP8750I: Starting registry migration for component 'UserNameServer'.
BIP8755I: Copied value 'QueueManagerName' into the new location
BIP8755I: Copied value 'ServiceUserId' into the new location
BIP8755I: Copied value 'ServicePassword' into the new location
BIP8755I: Copied value 'NTSecurityDomainName' into the new location
BIP8755I: Copied value 'RefreshInterval' into the new location
```

```
BIP8755I: Copied value 'AdminAgentPID' into the new location
BIP8755I: Copied value 'BrokerUUID' into the new location
BIP8763I: Deleted value 'QueueManagerName' from the old location
BIP8763I: Deleted value 'ServiceUserId' from the old location
BIP8763I: Deleted value 'ServicePassword' from the old location
BIP8763I: Deleted value 'NTSecurityDomainName' from the old location
BIP8763I: Deleted value 'RefreshInterval' from the old location
BIP8763I: Deleted value 'AdminAgentPID' from the old location
BIP8763I: Deleted value 'BrokerUUID' from the old location
BIP8768I: Finished registry migration for component 'UserNameServer'.
BIP8654I: Moving filesystem artefacts from '/var/mqsi' to '/var/mqsi'
BIP8655I: Removing '/var/mqsi/brokers/UserNameServer/bin/mqsistart'.
BIP8655I: Removing '/var/mqsi/brokers/UserNameServer/bin/mqsistop'.
BIP8655I: Removing '/var/mqsi/brokers/UserNameServer/bin'.
BIP8655I: Removing '/var/mqsi/brokers/UserNameServer'.
BIP8071I: Successful command completion.
```

6. Enter the following command to do the post-migration check of the Message Broker V6.0 User Name Server:

   ```
   mqsimigratecomponents -v UserNameServer
   ```

   A successful command output reports the message BIP8689I, as shown in Example 12-8.

Example 12-8 Verifying the User Name Server migration

```
$ mqsimigratecomponents -v UserNameServer
BIP8675I: UserNameServer 'UserNameServer' (Version 6.0) specified for
migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

12.3.3 Validating the User Name Server migration

Now that the User Name Server migration is complete, your task on the AIX system is nearly complete. Start the User Name Server and then check whether it executes properly.

The commands for managing the User Name Server are the same as in Integrator V2.1.

Starting the User Name Server

Log in as the User Name Server user ID and enter the commands from an appropriate platform command prompt, then:

1. Display the list of WebSphere Message Broker components and enter the **mqsilist -a** command. The command should display the same components as those before migration. The version is displayed for all previous broker versions, as shown in Example 12-9:

   Attention: The previous versions of the broker are displayed only if the **-a** option is used.

Example 12-9 Displaying the components

```
$ mqsilist -a
BIP8099I: Broker: BKA1 (Version 2) -  BKA1QM
BIP8099I: UserNameServer: UserNameServer  -  UNSQM
BIP8071I: Successful command completion.
```

2. Start all brokers and the Configuration Manager connected to the User Name Server (if you stopped them earlier in 12.3.2, “Migrating the User Name Server” on page 233).
3. As Example 12-10 shows, start your User Name Server and enter the following command:

   ```
   mqsistart UserNameServer
   ```

Example 12-10 Starting the User Name Server

```
$ mqsistart UserNameServer
WebSphere MQ queue manager running.
BIP8096I: Successful command initiation, check the system log to ensure
that the component started without problem and that it continues to run
without problem.
```

4. Check whether the User Name Server is running. As shown in Example 12-11, enter the following command on AIX:

   ```
   ps -ef | grep UserNameServer
   ```

Example 12-11 Displaying the User Name Server processes

```
$ ps -ef | grep UserNameServer
tstadmin 27292 43164   0 13:34:03      -  0:00 bipuns UserNameServer
tstadmin 43164     1   0 13:34:03      -  0:00 bipservice
UserNameServer
```

```
tstadmin 49818 43512   3 13:34:42  pts/0  0:00 grep UserNameServer
```

5. Verify the operating system application log for information or errors.

   If the migration fails, then refer to Chapter 16, “Migration verification” on page 287.

   Once the migration of the product is completed and thorough testing has been performed to ensure that the newly migrated components are functionally verified, then the WebSphere MQ Integrator V2.1 product can be uninstalled at your discretion.

Migration verification

Check the error log files after the migration to ensure that there were no errors during migration.

For details about which error log files should be checked and how to check them, refer to 16.1.4, “Checking the logs and where to find them” on page 297.

13

Migrating the Windows Broker

This chapter describes the process of migrating a WebSphere MQ Integrator V2.1 broker, running on Windows with a DB2 data source, to WebSphere Message Broker V6.0.

The goal of this chapter is to guide you through the migration of the Windows broker in the scenario described in Chapter 7, “WebSphere MQ Integrator Version 2.1 environment setup” on page 107. After completing the steps in this chapter, the broker will be ready to operate in the migrated domain.

13.1 Migration preparation

This section describes the tasks that are to be performed before the broker component migration as shown in Figure 13-1.

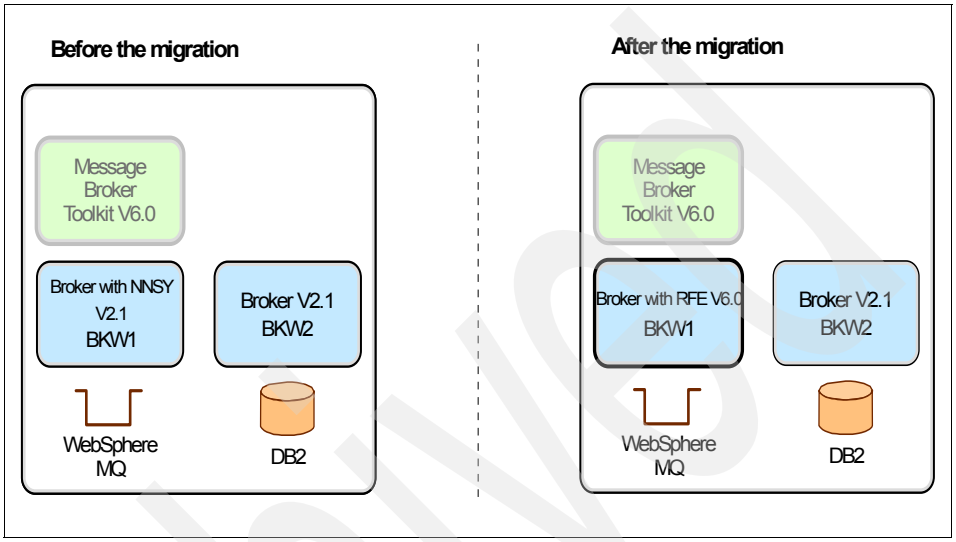


Figure 13-1 Broker migration scenario

13.1.1 Reviewing the existing installation

Check the existing installation for any errors. Ensure that the broker is running properly and that there are no error reports before the migration.

Reviewing the configuration

To confirm that all important components are running properly, you can use the following commands and check the output.

Log in using broker service user ID and launch the commands from the Windows platform command prompt:

1. Display the list of queue managers and issue the **`dspmq`** command. Check the status of all expected queue managers, as shown in Example 13-1.

Example 13-1 Displaying the queue managers

```
dspmq
QMNAME(BKW1QM)
STATUS(Running)
```

```
QMNAME(BKW2QM)
STATUS(Running)
QMNAME(CMQM)
STATUS(Running)
```

2. Display the list of WebSphere MQ Integrator components and issue the **mqsilist** command. Check the status of all expected components, as shown in Example 13-2.

Example 13-2 Displaying the components

```
mqsilist
BIP8099I: BKW1  -  BKW1QM
BIP8099I: BKW2  -  BKW2QM
BIP8099I: ConfigMgr  - CMQM
BIP8071I: Successful command completion.
```

3. Display the detail of each broker and issue the following command:

   `mqsilist` *broker_name*

   Check all execution groups, as shown in Example 13-3.

Example 13-3 Displaying the execution groups

```
mqsilist BKW1
BIP8130I: Execution Group: EG1  -  3784
BIP8130I: Execution Group: default  -  3220
BIP8130I: Execution Group: EG3  -  3872
BIP8130I: Execution Group: EG2  -  3892
BIP8071I: Successful command completion.
mqsilist BKW2
BIP8130I: Execution Group: EG1  -  3988
BIP8130I: Execution Group: EG3  -  4012
BIP8130I: Execution Group: default  -  3912
BIP8130I: Execution Group: EG2  -  4020
BIP8071I: Successful command completion.
```

4. As shown in Example 13-4, check whether the database is available. Enter the **db2 connect to** <*broker_database*> command on a DB2 command window (type the command **db2cmd** in the Windows command prompt).

Example 13-4 Testing the database connection

```
db2 connect to BKW1DB
Database Connection Information
Database server        = DB2/NT 8.2.0
SQL authorization ID   = TSTADMIN
```

```
Local database alias   = BKW1DB
db2 connect to BKW2DB
Database Connection Information
Database server        = DB2/NT 8.2.0
SQL authorization ID   = TSTADMIN
Local database alias   = BKW2DB
```

5. As shown in Example 13-5, reset the DB2 connections with the **db2 disconnect current** command.

Example 13-5 Resetting the database connection

```
db2 disconnect current
DB20000I  The SQL DISCONNECT command completed successfully.
```

Premigration verification

Check the error log files for Integrator V2.1 components, queue manager, and DB2, for any important information or errors.

For details about which error log files to check and how to check them, refer to 16.1.4, “Checking the logs and where to find them” on page 297.

Backing up the broker configuration

Ensure that the broker configuration is successfully backed up.

For details on how to back up the broker configuration, refer to Chapter 9, “Backing up the WebSphere MQ Integrator V2.1 domain” on page 133.

13.1.2 Installing WebSphere Message Broker Version 6.0

Install and verify the WebSphere Message Broker V6.0 installation, if this has not already been done.

Perform the installation according to standard installation documentation. The description of the procedure for the installation is outside the scope of this book. Check installation of the following products:

- Rational Agent Controller (RAC) V6.0.0.1.

  Important: If you want to install any fix pack for RAC, check the WebSphere Message Broker V6.0 documentation. WebSphere Message Broker V6.0 requires the correct version of RAC. Any other version is not supported.

- WebSphere Message Broker V6.0 components:

– Broker
– Transformation Services

Tip: We recommend that you install the latest fix pack for WebSphere Message Broker V6.0 and an appropriate RAC version.

13.2 Configuration settings

Some configuration changes are done before the component migration starts. This section discusses these changes.

Before you make any changes to the configuration, you must stop the broker. First, log in with the user ID of the broker service. Then enter the following command from the appropriate Windows platform command prompt.

```
mqsistop broker_name
```

You will see the following:

```
mqsistop BKW1
BIP8071I: Successful command completion.
```

13.2.1 Updating environment settings

The environment settings for the service user ID have changed in Message Broker V6.0, so some environment variables also have to be changed. This initialization ensures that all the commands you enter, interact with the correct installation of the code.

The new environment variables are set up automatically when opening the WebSphere Message Broker Command Console.

Click **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **Command Console** to open the command console. This opens a command window with the correct environment. You are not restricted to invoking runtime commands in this window. You can enter operating system commands to perform typical actions.

A command console is shown in Figure 13-2.

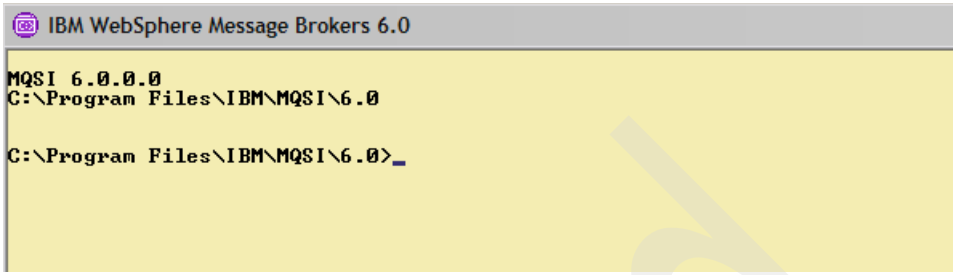


Figure 13-2 IBM WebSphere Message Broker 6.0 Command Console

13.2.2 Updating ODBC settings

The next step is to modify the settings for the Open Database Connectivity (ODBC) driver, first recording what the previous settings were. The new product contains new versions of the ODBC drivers for Oracle and Sybase databases. These databases are outside the scope of this book.

In the DB2 for Windows database there is no change to the ODBC settings.

Tip: The standard documentation describes the configuration changes. Refer to the WebSphere Message Broker 6.0 Information Center under the topics **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 to WebSphere Message Broker Version 6.0 → Migrating components on different computers → Migrating the broker → Distributed operating systems → Changing the ODBC connection and XA resource manager definitions for a migrated broker for details**, on the Web at:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13850_.htm`

13.3 The broker migration

This section describes the component migration steps.

13.3.1 Before starting

Before migrating a broker, ensure that you do not have any aggregations in progress. When migrating a broker to Message Broker V6.0, any live data being stored for aggregations in progress will be lost.

Also make sure that the broker service user ID uses the supported locale and code page.

Tip: Refer to the WebSphere Message Broker 6.0 Information center under the topic **Reference → Installation → National Language Support**, on the Web at:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an08720_.htm`

13.3.2 Migrating the broker

To migrate a WebSphere MQ Integrator V2.1 broker in distributed operating systems to the WebSphere Message Broker V6.0 broker at the same location, you need to take the following actions.

Log in as the broker service user ID and enter the commands from an appropriate platform command prompt:

1. Stop all channels from communicating to the broker queue manager.
2. Open the Command Console using the path **Start → Programs → IBM WebSphere Message Brokers 6.0 → Command Console.**
3. As shown in Example 13-6, enter the following Message Broker V6.0 command to perform a premigration check on the broker:

   `mqsimigratecomponents -c` *`broker_name`*

Example 13-6 Testing the broker migration

```
mqsimigratecomponents -c BKW1
BIP8849I: Broker 'BKW1' (Version 2.1) with Queue Manager 'BKW1QM' and
Data Source 'BKW1DB' specified for migration..
BIP8791I: Duplicate rows check started.
BIP8794I: Table BRMINFO has no duplicated rows.
BIP8794I: Table BRMRTDDEPINFO has no duplicated rows.
BIP8794I: Table BROKERRESOURCES has no duplicated rows.
BIP8794I: Table BRMRTDINFO has no duplicated rows.
BIP8794I: Table BRMWFDINFO has no duplicated rows.
BIP8792I: Duplicate rows check passed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001)
should be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001)
should be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001)
should be redeployed.
```

```
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRKO7EO01)
should be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRKO7CO01)
should be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRKO7EO01)
should be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRKO7CO01)
should be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRKO7EO01)
should be redeployed.
BIP8680I: Pre-migration check succeeded.
BIP8071I: Successful command completion.
```

4. As shown in Example 13-7, enter the following Message Broker V6.0 command to migrate the broker to Message Broker V6.0:

   `mqsimigratecomponents` *`broker_name`*

Example 13-7 Migrating the broker

```
mqsimigratecomponents BKW1
BIP8849I: Broker 'BKW1' (Version 2.1) with Queue Manager 'BKW1QM' and
Data Source 'BKW1DB' specified for migration.
BIP8755I: Copied value 'QueueManagerName' into the new location
BIP8755I: Copied value 'DataSourceName' into the new location
BIP8755I: Copied value 'DataSourceUserId' into the new location
BIP8755I: Copied value 'DataSourcePassword' into the new location
BIP8755I: Copied value 'LilPath' into the new location
BIP8755I: Copied value 'ConfigurationTimeout' into the new location
BIP8755I: Copied value 'ConfigurationDelayTimeout' into the new
location
BIP8755I: Copied value 'MigrationNeeded' into the new location
BIP8755I: Copied value 'MQTrustedQueueManager' into the new location
BIP8755I: Copied value 'UserNameServerQueueManagerName' into the new
location
BIP8755I: Copied value 'BrokerUUID' into the new location
BIP8755I: Copied value 'AdminAgentPID' into the new location
BIP8763I: Deleted value 'QueueManagerName' from the old location
BIP8763I: Deleted value 'DataSourceName' from the old location
BIP8763I: Deleted value 'DataSourceUserId' from the old location
BIP8763I: Deleted value 'DataSourcePassword' from the old location
BIP8763I: Deleted value 'LilPath' from the old location
BIP8763I: Deleted value 'ConfigurationTimeout' from the old location
BIP8763I: Deleted value 'ConfigurationDelayTimeout' from the old
location
BIP8763I: Deleted value 'MigrationNeeded' from the old location
```

```
BIP8763I: Deleted value 'MQTrustedQueueManager' from the old location
BIP8763I: Deleted value 'UserNameServerQueueManagerName' from the old
location
BIP8763I: Deleted value 'BrokerUUID' from the old location
BIP8763I: Deleted value 'AdminAgentPID' from the old location
BIP8768I: Finished registry migration for component 'BKW1'.
BIP8654I: Moving filesystem artefacts from '' to 'C:\Documents and
Settings\AllUsers\Application Data\IBM\MQSI'
BIP8670I: Database migration started
BIP8663I: Creating temporary new tables
BIP8664I: Migrating from existing tables to temporary new tables
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001)
should be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001)
should be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001)
should be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001)
should be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001)
should be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001)
should be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001)
should be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001)
should be redeployed.
BIP8665I: Dropping existing tables
BIP8666I: Creating new tables
BIP8667I: Copying all rows from temporary new tables to new tables
BIP8668I: Dropping temporary new tables
BIP8669I: Database migration successful
BIP8785I: Starting WebSphere MQ queue migration for component 'BKW1'.
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REQUEST'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.CONTROL'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REPLY'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.TIMEOUT'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.UNKNOWN'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.TIMEOUT.QUEUE'
```

```
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue
'SYSTEM.BROKER.INTERBROKER.MODEL.QUEUE'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.WS.INPUT'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.WS.REPLY'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.WS.ACK'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.IPC.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.ADMIN.QUEUE'
BIP8787I: Cleared WebSphere MQ queue
'SYSTEM.BROKER.EXECUTIONGROUP.QUEUE'
BIP8787I: Cleared WebSphere MQ queue
'SYSTEM.BROKER.EXECUTIONGROUP.REPLY'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.IPC.QUEUE'
BIP8789I: Finished WebSphere MQ queue migration for component 'BKW1'.
BIP8071I: Successful command completion.
```

Attention: The output of the previous command shows that you should redeploy all message sets. WebSphere MQ Integrator V2.1 deploys message sets directly to brokers rather than to an execution group. During migration WebSphere Message Broker V6.0 deploys all message sets to all execution groups.

5. Enter the following Message Broker V6.0 command to do the post-migration check of your broker to Message Broker V6.0:

   ```
   mqsimigratecomponents -v broker_name
   ```

   A successful command output reports the message BIP8689I, as shown in Example 13-8.

Example 13-8 Verifying the broker migration

```
mqsimigratecomponents -v BKW1
BIP8849I: Broker 'BKW1' (Version 6.0) with Queue Manager 'BKW1QM' and
Data Source 'BKW1DB' specified for migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

13.3.3 Validating the broker migration

Now that the broker migration is complete, your tasks on the Windows system is nearly complete. Start the broker and then check if it executes properly.

The commands for managing the broker are the same as in WebSphere MQ Integrator V2.1.

Starting the broker

Log in as the broker service user ID and enter the commands from an appropriate platform command prompt:

1. Start all channels communicating to the broker queue manager.
2. Open the Command Console using the path **Start → Programs → IBM WebSphere Message Brokers 6.0 → Command Console**.
3. As shown in Example 13-9, start your broker and enter the following command:

   ```
   mqsistart broker_name
   ```

Example 13-9 Starting the broker

```
mqsistart BKW1
WebSphere MQ queue manager running.
BIP8096I: Successful command initiation, check the system log to ensure
that the component started without problem and that it continues to run
without problem.
```

4. Display the list of WebSphere Message Broker components and enter the **mqsilist -a** command. The command should display the same components as those before migration. The version is also displayed for all the previous version brokers, as shown in Example 13-10.

Tip: The previous versions of the broker are displayed only if the -a option is used.

Example 13-10 Displaying the components

```
mqsilist -a
BIP8099I: Broker: BKW1  -  BKW1QM
BIP8221I: Broker: BKW2 (Version 2) - BKW2QM
BIP8099I: ConfigMgr: ConfigMgr -  CMQM
BIP8071I: Successful command completion.
```

5. Display the details of the broker by entering the following command:

```
mqsilist broker_name
```

6. The command should display the same list of the execution groups as those before migration, as shown in Example 13-11.

Note: To display the details of the previous broker version, use the command and environment settings from WebSphere MQ Integrator V2.1.

Example 13-11 Displaying the execution groups

```
mqsilist BKW1
BIP8130I: Execution Group: EG01 -  2792
BIP8130I: Execution Group: default  -  3888
BIP8130I: Execution Group: EG03 -  3880
BIP8130I: Execution Group: EG02 -  2948
BIP8071I: Successful command completion.
```

7. Currently there is an issue that after migration the message flows need to start, but this is expected to be resolved in a future fix pack. To start all message flows use the **mqsistartmsgflow** command. Enter the command for each execution group, as shown in Example 13-12.

Example 13-12 Starting the message flows

```
mqsistartmsgflow -q CMQM -i localhost -p 1414 -b BKW1 -e EG1
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1027I: Submitting request to the Configuration Manager to start
'BKW1/EG1'...
BIP1029I: Waiting up to 60 second(s) for broker BKW1 to update its
configuration...
BIP1030I: Message flow 'BKW1/EG1/Loan_Request' is reported as started.
BIP8071I: Successful command completion.
```

Migration verification

Verify the error log files, after the migration, to ensure that there were no errors during migration.

For details about which error log files to check and how to check them, refer to 16.1.4, “Checking the logs and where to find them” on page 297.

13.3.4 Migration verification for Rules and Formatter Extension

If WebSphere Message Broker V6.0 is using the Rules and Formatter Extension, then perform the following steps:

1. Check the environment variable NNSY_ROOT in the IBM WebSphere Message Broker V6.0 Command Console. The New Era Of Networks installation automatically updates the NNSY_ROOT variable to the installation directory.

   ```
   NNSY_ROOT=C:\Program Files\IBM\rf41
   ```

2. Check the ODBC database setting in 13.2.2, “Updating ODBC settings” on page 244.

   The nnsyreg.dat sample file of New Era Of Networks is in the following directory:

   ```
   <rfe_install_dir>\rfdt\gui\examples
   ```

   Check the differences between the current nnsyreg.dat file and the Message Broker V6.0 sample. Edit the nnsyreg.dat file and change the NNOT_SHARED_LIBRARY parameter in all sessions.

   Example 13-13 shows a section of the nnsyreg.dat file.

Example 13-13 nnsyreg.dat sample file

```
Session.MQSI_CONFIG
   NNOT_SHARED_LIBRARY    = dvdb41db2
   NNOT_FACTORY_FUNCTION  = NNSesDB2Factory
   NN_SES_SERVER          = NNDB
   NN_SES_USER_ID         = tstadmin
   NN_SES_PASSWORD        = password
```

   New Era Of Networks searches for the current nnsyreg.dat in the following directories:

   a. Current working directory.
   b. The environment variable NN_CONFIG_FILE_PATH specifies the directory.
   c. The environment variable NNSY_ROOT specifies the directory.

   Be sure to use the correct configuration file.

> **Tip:** If the new nnsyreg.dat file is located in the WebSphere MQ Integrator V2.1 product directory, move it to an appropriate directory as indicated previously.

14

Migrating the AIX Broker

This chapter describes how to migrate a WebSphere MQ Integrator V2.1 broker that is running on AIX with a DB2 data source to WebSphere Message Broker V6.0.

It guides you through the migration of the AIX broker in the scenario described in Chapter 7, “WebSphere MQ Integrator Version 2.1 environment setup” on page 107. After you complete the steps given here, the broker will be ready to operate in the migrated topology.

14.1 Migration preparation

This section describes the tasks that must be performed before the broker component migration, as shown in Figure 14-1.

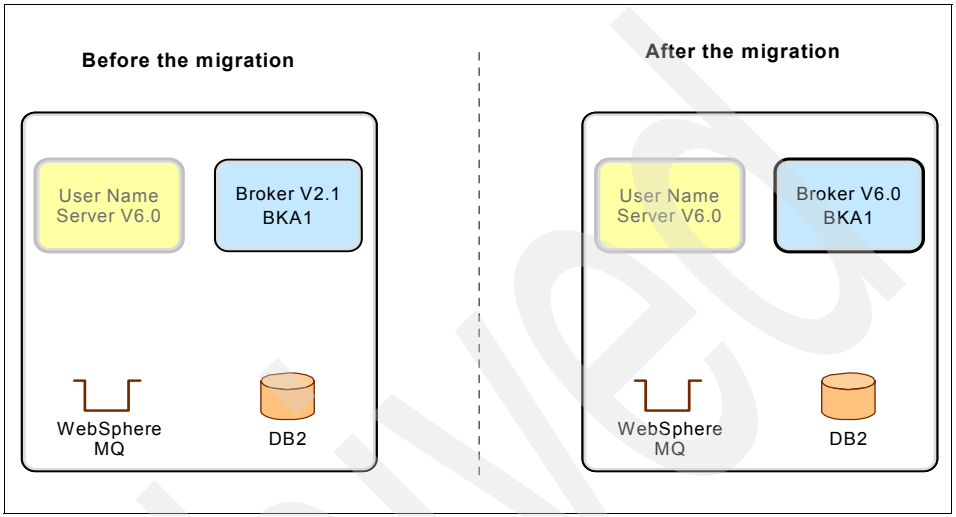


Figure 14-1 Broker migration scenario

14.1.1 Reviewing the existing installation

Verify that the existing installation has no errors. Ensure that the broker is running properly and that there are no error reports before the migration.

Tip: You can use the **`script /tmp/output.txt`** command to make a typescript of everything displayed on your terminal to the specified file (in this case, to /tmp/output.txt) on UNIX systems.

Reviewing the configuration

To confirm that all important components are running properly, you can use the following commands and check the output. Log in as the broker service user ID and enter the commands from an appropriate platform command prompt:

1. To display the list of queue managers and the status of all expected queue managers, enter the **`dspmq`** command as shown in Example 14-1.

Example 14-1 Displaying the queue managers

```
$ dspmq
QMNAME(UNSQM)                                             STATUS(Running)
```

```
QMNAME(BKA1QM)                                  STATUS(Running)
```

2. To display the list of WebSphere MQ Integrator components and the status of all expected components, enter the **mqsilist command** as shown in Example 14-2.

Example 14-2 Displaying the components

```
$ mqsilist
BIP8099I: BKA1  -  BKA1QM
BIP8099I: UserNameServer  -  UNSQM
BIP8071I: Successful command completion.
```

3. To display the detail of your broker, enter the following command:

   `mqsilist broker_name`

4. Check all expected execution groups, as shown in Example 14-3.

Example 14-3 Displaying the execution groups

```
$ mqsilist BKA1
BIP8130I: Execution Group: EG2  -  44036
BIP8130I: Execution Group: EG1  -  50718
BIP8130I: Execution Group: EG3  -  33716
BIP8130I: Execution Group: default  -  9658
BIP8071I: Successful command completion.
```

5. To verify that the database is available, enter the following command as shown in Example 14-4:

   `db2 connect to broker_database`

Example 14-4 Testing the database connection

```
$ db2 connect to BKA1DB user tstadmin using password
Database Connection Information
Database server        = DB2/6000 8.2.0
SQL authorization ID   = TSTADMIN
Local database alias   = BKA1DB
```

6. Reset the DB2 connection and enter the following command:

   `db2 disconnect current`

Premigration verification

Check the error log files for Integrator V2.1 components, queue manager, and DB2 for any important information or errors.

For details about which error log files to check and how to check them, refer to 16.1.4, “Checking the logs and where to find them” on page 297.

Backing up the broker configuration

Ensure that the broker configuration is successfully backed up. For details about how to back up the broker configuration, refer to Chapter 9, “Backing up the WebSphere MQ Integrator V2.1 domain” on page 133.

14.1.2 Installing WebSphere Message Broker Version 6.0

Install or verify the WebSphere Message Broker V6.0 installation, if not already done.

Perform the installation according to standard installation documentation. The description for this procedure is outside the scope of this book. Verify that the following products have been installed:

- Rational Agent Controller (RAC) V6.0.0.1

  Important: To install any fix pack for RAC, check the WebSphere Message Broker V6.0 documentation. WebSphere Message Broker V6.0 requires the exact version of RAC. Any other version is not supported.

- WebSphere Message Broker V6.0 components:
  - Broker
  - Transformation Services

  Tip: We recommend that you install the latest fix pack for WebSphere Message Broker V6.0 and the appropriate RAC version.

14.2 Configuration settings

Some configuration changes must be completed before the component migration starts. This section discusses these changes.

Before you make any changes to the configuration, you must stop the broker as shown in Example 14-5. First log in with the user ID of the broker service. Then enter the following command from an appropriate platform command prompt window:

```
mqsistop broker_name
```

Example 14-5 Stopping the broker

```
$ mqsistop BKA1
BIP8071I: Successful command completion.
```

14.2.1 Updating environment settings

The environment settings for the service user ID changes in Message Broker V6.0, so you must change the user profile as well. This initialization ensures that all the commands you enter will interact with the correct installation of the code.

The full path of the new Message Broker sample profile is:

broker_install_directory/6.0/bin/mqsiprofile

Edit the Message Broker service user ID profile using an editor such as vi:

- Remove the WebSphere MQ Integrator V2.1 settings. This may be a call to the sample profile, in which case delete the following line:

  /usr/opt/mqsi/sample/profiles/profile.aix

- Add the following line:

  broker_install_directory/6.0/bin/mqsiprofile

Example 14-6 shows the relevant part of the user profile.

Example 14-6 Sample of user profile

```
PATH=$PATH:/usr/opt/db2_08_01/das/bin:/usr/java131/jre/bin:/usr/java131/bin

export PATH

. <broker_install_directory>/6.0/bin/mqsiprofile
```

Note: There are a number of new commands available with the WebSphere Message Broker V6.0 Configuration Manager installation, including the following:

- mqsibackupconfigmgr
- mqsirestoreconfigmgr
- mqsistartmsgflow
- mqsistopmsgflow

These utilize the Configuration Manager Proxy API. In order to execute the new commands on UNIX platforms, you need to add the shipped Java 1.4.2 JRE to the PATH environment variable in the mqsiprofile file. The PATH environment variable must have *install_directory*/jre/bin added to it.

Log out and then log back in using the broker service ID again to activate these changes.

Important: If WebSphere MQ Integrator V2.1 and WebSphere Message Broker V6.0 need to coexist, then both versions of the broker profile cannot exist in the user profile at the same time. Leave the user profile without the broker profile and choose the appropriate version of broker profile manually.

14.2.2 Updating ODBC settings

The next step is to modify the settings for the Open Database Connectivity (ODBC) driver—but first take a backup copy of the .odbc.ini.

The new product contains new versions of the ODBC drivers for Oracle and Sybase databases. (However, covering those databases is beyond the scope of this redbook.) A new version of the DataDirect Driver Manager for Linux and UNIX systems is also included.

Change the ODBC connection definition for each DB2 database that the broker accesses. The full path of the new WebSphere Message Broker V6.0 odbc.ini sample is:

broker_install_directory/6.0/merant/odbc.ini

Check the differences between the current .odbc.ini file and the Message Broker V6.0 sample.

Edit the .odbc.ini file and perform the required changes. Enter the following command using an editor such as vi:

```
vi /var/mqsi/odbc/.odbc.ini
```

Example 14-7 shows the relevant part of the .odbc.ini file.

Example 14-7 Sample of the .odbc.ini

```
[BKA1DB]
Driver=/db2data/db2inst1/sqllib/lib/libdb2.a
Description=BKA1DB DB2 ODBC Database
Database=BKA1DB
```

Tip: The configuration changes are documented in the standard documentation. Refer to the WebSphere Message Broker 6.0 Information Center; follow this path **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 to WebSphere Message Broker Version 6.0 → Migrating components on different computers → Migrating the broker → Distributed operating systems → Changing the ODBC connection and XA resource manager definitions for a migrated broker**, available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13850_.htm

14.3 Migrating the broker

This section explains the component migration steps.

14.3.1 Before starting

Before migrating a broker, ensure that you do not have any aggregations in progress because when migrating a broker to Message Broker V6.0, any live data being stored for aggregations in progress is lost.

Also, check that the broker service user ID uses the supported locale and code page.

Tip: Refer the WebSphere Message Broker 6.0 Information center; follow this path **Reference → Installation → National Language Support** on the web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an08720_.htm

14.3.2 Migrating the broker

Follow these steps to migrate a WebSphere MQ Integrator V2.1 broker on a distributed operating system to WebSphere Message Broker V6.0 at the same location.

Log in as the broker service user ID and enter the commands from an appropriate platform command prompt:

1. Stop all channels from communicating to the broker queue manager.
2. Perform the appropriate environment settings, as shown in Example 14-8.

Example 14-8 Running Message Broker V6.0 profile

```
$ . /opt/IBM/mqsi/6.0/bin/mqsiprofile
```

3. Perform a pre-migration on the broker by entering the following Message Broker V6.0 command; also see Example 14-9.

   `mqsimigratecomponents -c` *`broker_name`*

Example 14-9 Testing the broker migration

```
$ mqsimigratecomponents -c BKA1
BIP8849I: Broker 'BKA1' (Version 2.1) with Queue Manager 'BKA1QM' and Data
Source 'BKA1DB' specified for migration.
BIP8791I: Duplicate rows check started.
BIP8794I: Table BRMINFO has no duplicated rows.
BIP8794I: Table BRMRTDDEPINFO has no duplicated rows.
BIP8794I: Table BROKERRESOURCES has no duplicated rows.
BIP8794I: Table BRMRTDINFO has no duplicated rows.
BIP8794I: Table BRMWFDINFO has no duplicated rows.
BIP8792I: Duplicate rows check passed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001) should
be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001) should
be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001) should
be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001) should
be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001) should
be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001) should
be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001) should
be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001) should
be redeployed.
BIP8680I: Pre-migration check succeeded.
BIP8071I: Successful command completion.
```

4. Migrate the broker by entering the following Message Broker V6.0 command; also see Example 14-10.

   `mqsimigratecomponents` *`broker_name`*

Example 14-10 Migrating the broker

```
$ mqsimigratecomponents BKA1
BIP8849I: Broker 'BKA1' (Version 2.1) with Queue Manager 'BKA1QM' and Data
Source 'BKA1DB' specified for migration.
BIP8750I: Starting registry migration for component 'BKA1'.
```

```
BIP8755I: Copied value 'QueueManagerName' into the new location
BIP8755I: Copied value 'DataSourceName' into the new location
BIP8755I: Copied value 'ServiceUserId' into the new location
BIP8755I: Copied value 'ServicePassword' into the new location
BIP8755I: Copied value 'DataSourceUserId' into the new location
BIP8755I: Copied value 'DataSourcePassword' into the new location
BIP8755I: Copied value 'LilPath' into the new location
BIP8755I: Copied value 'ConfigurationTimeout' into the new location
BIP8755I: Copied value 'ConfigurationDelayTimeout' into the new location
BIP8755I: Copied value 'MigrationNeeded' into the new location
BIP8755I: Copied value 'MQTrustedQueueManager' into the new location
BIP8755I: Copied value 'UserNameServerQueueManagerName' into the new location
BIP8755I: Copied value 'AdminAgentPID' into the new location
BIP8755I: Copied value 'BrokerUUID' into the new location
BIP8763I: Deleted value 'QueueManagerName' from the old location
BIP8763I: Deleted value 'DataSourceName' from the old location
BIP8763I: Deleted value 'ServiceUserId' from the old location
BIP8763I: Deleted value 'ServicePassword' from the old location
BIP8763I: Deleted value 'DataSourceUserId' from the old location
BIP8763I: Deleted value 'DataSourcePassword' from the old location
BIP8763I: Deleted value 'LilPath' from the old location
BIP8763I: Deleted value 'ConfigurationTimeout' from the old location
BIP8763I: Deleted value 'ConfigurationDelayTimeout' from the old location
BIP8763I: Deleted value 'MigrationNeeded' from the old location
BIP8763I: Deleted value 'MQTrustedQueueManager' from the old location
BIP8763I: Deleted value 'UserNameServerQueueManagerName' from the old location
BIP8763I: Deleted value 'AdminAgentPID' from the old location
BIP8763I: Deleted value 'BrokerUUID' from the old location
BIP8768I: Finished registry migration for component 'BKA1'.
BIP8654I: Moving filesystem artefacts from '/var/mqsi' to '/var/mqsi'
BIP8655I: Removing '/var/mqsi/brokers/BKA1/bin/mqsistart'.
BIP8655I: Removing '/var/mqsi/brokers/BKA1/bin/mqsistop'.
BIP8655I: Removing '/var/mqsi/brokers/BKA1/bin'.
BIP8655I: Removing '/var/mqsi/brokers/BKA1'.
BIP8670I: Database migration started
BIP8663I: Creating temporary new tables
BIP8664I: Migrating from existing tables to temporary new tables
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001) should
be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001) should
be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001) should
be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001) should
be redeployed.
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRK07C001) should
be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRK07E001) should
be redeployed.
```

```
BIP8688W: Following migration, Message Set 'RequestLoan'(E6H9JRKO7CO01) should
be redeployed.
BIP8688W: Following migration, Message Set 'ResultLoan'(E6H9JRKO7EO01) should
be redeployed.
BIP8665I: Dropping existing tables
BIP8666I: Creating new tables
BIP8667I: Copying all rows from temporary new tables to new tables
BIP8668I: Dropping temporary new tables
BIP8669I: Database migration successful
WebSphere MQ queue manager running.
BIP8785I: Starting WebSphere MQ queue migration for component 'BKA1'.
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REQUEST'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.CONTROL'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REPLY'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.TIMEOUT'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.UNKNOWN'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.TIMEOUT.QUEUE'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.INTERBROKER.MODEL.QUEUE'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.WS.INPUT'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.WS.REPLY'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.WS.ACK'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.IPC.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.ADMIN.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.EXECUTIONGROUP.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.EXECUTIONGROUP.REPLY'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.IPC.QUEUE'
BIP8789I: Finished WebSphere MQ queue migration for component 'BKA1'.
BIP8071I: Successful command completion.
```

Attention: The output of this command shows that you must redeploy all message sets. WebSphere MQ Integrator V2.1 deploys message sets directly to the broker, rather than to an execution group. During migration, WebSphere Message Broker V6.0 deploys all message sets to all execution groups.

5. Perform a post-migration check of your broker to Message Broker V6.0 by issuing the following command:

   ```
   mqsimigratecomponents -v broker_name
   ```

   A successful command output reports the message BIP8689I, as shown in Example 14-11.

Example 14-11 Verifying the broker migration

```
$ mqsimigratecomponents -v BKA1
BIP8849I: Broker 'BKA1' (Version 6.0) with Queue Manager 'BKA1QM' and Data
Source 'BKA1DB' specified for migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

14.3.3 Validating the broker migration

Now that the broker migration is complete, your tasks on the AIX system are also nearly complete. Start the broker and then verify that it executes properly. The commands for managing the broker are the same as in WebSphere MQ Integrator V2.1.

Starting the broker

Log in as the broker service user ID and enter the commands from an appropriate platform command prompt window:

1. Start all channels so as to communicate to the broker queue manager.
2. Start your broker by issuing the following command; also see Example 14-12.

   ```
   mqsistart broker_name
   ```

Example 14-12 Starting the broker

```
$ mqsistart BKA1
WebSphere MQ queue manager running.
BIP8096I: Successful command initiation, check the system log to ensure that
the component started without problem and that it continues to run without
problem.
```

3. Display the list of WebSphere Message Broker components by entering the **mqsilist** command. The command should display the same components as those before migration, as shown in Example 14-13.

Important: Previous versions of the broker are displayed only if the **-a** option is used.

Example 14-13 Displaying the components

```
$ mqsilist
BIP8099I: Broker: BKA1  -  BKA1QM
BIP8099I: UserNameServer: UserNameServer  -  UNSQM
BIP8071I: Successful command completion.
```

4. Display the detail of the broker by entering the following command:

   `mqsilist broker_name`

5. The command should display the same list of the execution groups as before the migration process; see Example 14-14.

Example 14-14 Displaying the execution groups

```
$ mqsilist BKA1
BIP8130I: Execution Group: EG2  -  33802
BIP8130I: Execution Group: EG1  -  49460
BIP8130I: Execution Group: EG3  -  44048
BIP8130I: Execution Group: default  -  44308
BIP8071I: Successful command completion.
```

Important: To display the details of the previous version broker, use the command and environment settings from WebSphere MQ Integrator V2.1.

6. Currently there is an issue that after migration, you need to start the message flows. (This is expected to be resolved in a future fix pack.) To start all message flows, use the **mqsistartmsgflow** command. Issue the command as shown in Example 14-15 for each execution group.

Example 14-15 Starting the message flows

```
$  mqsistartmsgflow -i <cm_host> -p 1414 -q CMQM -b BKA1 -e EG1
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1027I: Submitting request to the Configuration Manager to start
'BKA1/EG1'...
BIP1029I: Waiting up to 60 second(s) for broker BKA1 to update its
configuration...
BIP1030I: Message flow 'BKA1/EG1/Loan_Request' is reported as started.
BIP8071I: Successful command completion.
```

Tip: To run the new WebSphere Message Broker V6.0 commands, such as the **mqsistartmsgflow** command, you must use Java version 1.4.2.

Migration verification

Verify the error log files after the migration to ensure that there were no errors during migration.

For details about which error log files to check and how to check them, refer to 16.1.4, “Checking the logs and where to find them” on page 297.

14.3.4 Migration verification for Rules and Formatter Extension

If the new WebSphere Message Broker V6.0 is using the Rules and Formatter Extension, then follow these steps:

1. Enter **echo** $NNSY_ROOT on the command line to check the environment variable NNSY_ROOT. If it does not already exist, set NNSY_ROOT to point to the root directory where Rules and Formatter Extension is installed. For example, you can add the following line to the user profile:

   ```
   export NNSY_ROOT=/opt/IBM/rf41/rfe
   ```

2. Check the ODBC database setting in 14.2.2, “Updating ODBC settings” on page 258.

   The nnsyreg.dat sample file of New Era Of Networks is in the *rfe_install_dir*/rfe/examples/NNSYRF directory. Check the differences between the current nnsyreg.dat file and the Message Broker V6.0 sample.

   Edit the nnsyreg.dat file and change the NNOT_SHARED_LIBRARY parameter in all sessions.

   Example 14-16 shows a section of the nnsyreg.dat file.

Example 14-16 nnsyreg.dat sample file

```
Session.MQSI_CONFIG
   NNOT_SHARED_LIBRARY    = dvdb41db2
   NNOT_FACTORY_FUNCTION  = NNSesDB2Factory
   NN_SES_SERVER          = NNDB
   NN_SES_USER_ID         = tstadmin
   NN_SES_PASSWORD        = password
```

   New Era Of Networks searches for the current nnsyreg.dat in the following directories:

   – Current working directory
   – Directory specified by the environment variable NN_CONFIG_FILE_PATH
   – Directory specified by the environment variable NNSY_ROOT

   Be sure to use the correct configuration file.

Tip: If the new nnsyreg.dat file is located inside the WebSphere MQ Integrator V2.1 product directory, move it to an appropriate directory, as indicated in the preceding list.

15

Migrating the z/OS Broker

This chapter describes how to migrate a WebSphere MQ Integrator V2.1 Broker running on z/OS to WebSphere Message Broker V6.0.

It describes this migration as part of the larger domain migration detailed in Chapter 7, “WebSphere MQ Integrator Version 2.1 environment setup” on page 107. After you complete the steps explained in this chapter, the z/OS Broker will be ready to operate in the migrated topology.

15.1 Migration scenario

In this chapter, we explain how broker MQZ1BRK is migrated from WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0. Figure 15-1 shows before and after pictures to illustrate this migration.

Note: The Configuration Manager in Figure 15-1 is greyed out because the migration of the Configuration Manager to z/OS is covered in Chapter 11, “Migrating the Configuration Manager” on page 177. This chapter only details the migration of the broker on z/OS.

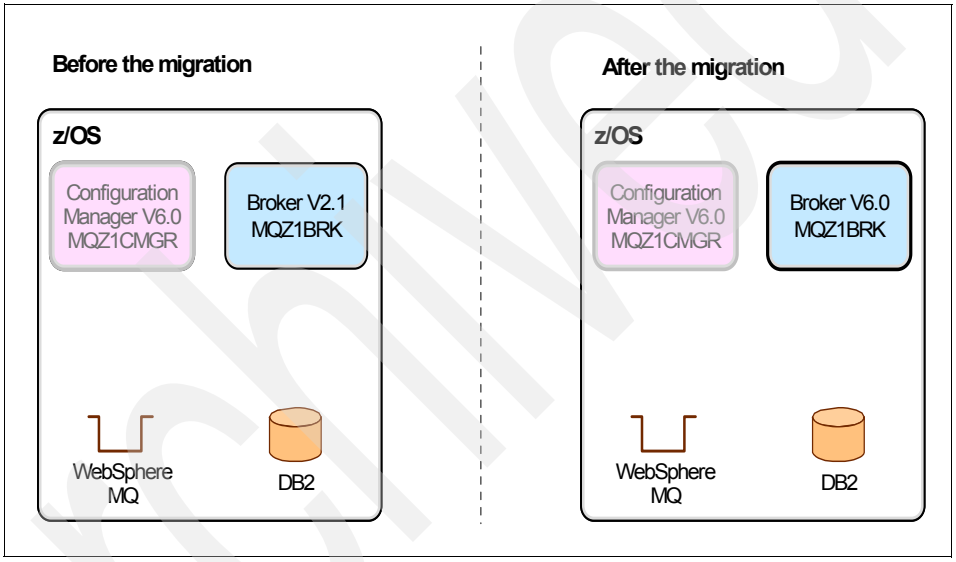


Figure 15-1 z/OS Broker V2.1 before and after migration

This chapter explains the steps you follow to ensure the successful migration of the broker.

15.2 Environment preparation

The steps involved in migrating the broker include preparing the environment for migration, checking the WebSphere Message Broker prerequisites, and backing up the current WebSphere MQ Integrator V2.1 domain.

15.2.1 Verify the current environment

Ensure that the current WebSphere MQ Integrator environment is functioning correctly before you begin the migration process. This will facilitate a smooth migration to WebSphere Message Broker V6.0.

To verify the environment, review the job log for the WebSphere MQ Integrator broker task. In this case, the started task MQZ1BRK was checked to ensure that no errors were present. Also check the DB2 and WebSphere MQ job logs to ensure there are no errors.

Resolve any errors that you encounter before you begin the migration. Any issues with the broker, DB2, or WebSphere MQ at WebSphere MQ Integrator V2.1 cause problems when migrating to WebSphere Message Broker V6.0.

15.2.2 Stop the broker

Stop the broker before backing it up. This ensures that no changes occur between the backup being taken and the migration completion. Also, the broker must be stopped in order for the migration job to run.

15.2.3 Back up the broker

Before you migrate the broker, ensure that the broker database and the broker registry are both successfully backed up. This allows you to restore the broker to the current WebSphere MQ Integrator V2.1 level, if required.

Important: If any changes occur to the broker database or broker registry *after* the backup has taken place (that is, if new subscriptions are made or new resources are deployed), they are lost if the backups are restored.

For details about how to back up the broker on z/OS, refer to Chapter 9, “Backing up the WebSphere MQ Integrator V2.1 domain” on page 133.

15.2.4 Check the system requirements

The system requirements for WebSphere Message Broker V6.0 are different from those of WebSphere MQ Integrator V2.1. You can find the prerequisites that must be installed before migration begins listed at:

`http://www.ibm.com/support/docview.wss?rs=849&uid=swg27006594`

Also, you must check the Preventative Service Planning (PSP), which is referenced in the program directory of the product being installed.

15.2.5 Check user ID authorities

The authorities for the broker administrator user ID and the broker started task user ID should already be configured correctly for the WebSphere MQ Integrator V2.1 domain. The migration will encounter problems if the correct authorities have not been assigned.

Refer to **Security** → **Setting up z/OS security** in the WebSphere Message Broker Information Center, available on the Web, for information about the authorities needed:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae14030_.htm

15.2.6 Install WebSphere Message Broker V6.0

The installation instructions for WebSphere Message Broker V6.0 are not covered here. For detailed instructions for installation on z/OS, select the path **Installing** → **Installation Guide** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah24100_.htm

Verify that the WebSphere Message Broker V6.0 code has been installed according to the installation guide and that the following products have been installed:

1. WebSphere Message Broker V6.0

> **Tip:** We recommend that you install the latest fix pack for WebSphere Message Broker V6.0 and the appropriate Rational Agent Controller version.

You can check the installation by locating the SBIPPROC and SBIPSAMP template PDSEs that are required in later steps (ask your systems administrator for the location of these data sets).

You can also check the hierarchical file system (HFS) for the WebSphere Message Broker V6.0 code. By default, this is located in /usr/lpp/mqsi/V6M0R0 (to determine the exact location, check with the person responsible for installation).

You will need to have both the template PDSEs and the HFS in place before migration can begin.

Note: Ensure that APAR OA11699 has been applied to the system before installation. This allows you to view the output from the WebSphere Message Broker V6.0 broker customization jobs in the job logs.

If APAR OA11699 has not been applied to the system, redirect the output from the broker jobs to the HFS. To do this, refer to customizing the broker's job control language (JCL) in "Redirect STDOUT and STDERR" on page 279.

2. Rational Agent Controller V6.0.0.1

   If the remote debugging of message flows using the Message Brokers Toolkit is required on the z/OS brokers, then install the Rational Agent Controller. By default the Rational Agent Controller is installed to /usr/lpp/IBM/AgentController.

Important: Check the WebSphere Message Broker V6.0 documentation to ensure the correct version of Rational Agent Controller is installed. The WebSphere Message Broker V6.0 requires an exact version of Rational Agent Controller. It does not support any other version.

15.3 Premigration tasks

After the environment has been verified, the migration can begin. Perform the premigration configuration tasks first. Gather information about the current system. This is essential for configuring the broker's JCL in later stages.

Then allocate a new PDSE for the migrated broker's JCL and configuration files. Finally, copy the template JCL provided with WebSphere Message Broker V6.0 to the new data set, and customize it for the local system.

15.3.1 Collect required information for the broker

The mqsicompcif file, used to configure the WebSphere MQ Integrator V2.1 broker, contains most of the necessary information required to fill out Table 15-1 on page 272, Table 15-2 on page 272, and Table 15-3 on page 273. For any missing information, contact the person responsible for that component of the system.

Collect reference information

Table 15-1 on page 272 lists installation information for reference.

Table 15-1 Installation information for your reference

Description	JCL variable	Redbook variables	Your installation variable
Fully qualified name of the product's SBIPPROC data set	N/A	<hlq>.SBIPPROC	
Fully qualified name of the product's SBIPSAMP data set	N/A	<hlq>.SBIPSAMP	

Collect JCL information

Table 15-2 lists the run time information for customizing the JCL.

Table 15-2 Broker run time information for JCL customization

Description	JCL variable	Redbook variables	Your installation variable
File system directory where the product has been installed	++INSTALL++	/usr/lpp/mqsi/V6M0R0	
HFS directory where the broker is to exist	++COMPONENTDIRECTORY++	/var/wmqi/MQZ1BRK	
Broker name	++COMPONENTNAME++	MQZ1BRK	
The broker's user ID HFS home directory	++HOME++	/u/mqz1brk	
mqsicreatebroker options	++OPTIONS++	‘’ Note: No options were specified so two single quotes are specified.	
Locale of environment where commands are run by submitting JCL	++LOCALE++	C	
Time zone of environment where commands are run by submitting JCL	++TIMEZONE++	GMT0BST	
Location of Java installation	++JAVA++	/usr/lpp/java/J1.4.2	

Description	JCL variable	Redbook variables	Your installation variable
LAN emulation (LE) high-level qualifier	++LEHLQ++	PP.ADLE370.ZOS150	
WebSphere MQ high-level-qualifier	++WMQHLQ++	MQM.V531	
Queue Manager associated with the broker	++QUEUEMANAGER++	MQZ1	
The data set where all JCL relevant to the broker is saved	++COMPONENTDATASET++	MQSI6.MQZ1BRK.CNTL	
Name of the Started Task JCL can be a maximum of 8 characters	++STARTEDTASKNAME++	MQZ1BRK	
Profile name	++COMPONENTPROFILE++	BIPBPROF	
Location of the XML toolkit	++XMLTOOLKIT++	/usr/lpp/ixm/IBM/xml4c-5_5	

Important: WebSphere Message Broker V6.0 requires XML Parser V1.8 for z/OS. For more information about obtaining the latest version, refer to the XML Toolkit Web site:

http://www-03.ibm.com/servers/eserver/zseries/software/xml/cparser/install.html

Collect DB2 information

Table 15-3 lists the DB2 information.

Table 15-3 DB2 information

Description	JCL variable	Redbook variables	Your installation variable
Specifies the DB2 converter	++DB2CONVERSION++	SINGLE	
DB2 subsystem identifier	++DB2SUBSYSTEM++	DB8W	

Description	JCL variable	Redbook variables	Your installation variable
DB2 location value of the DB2 subsystem	++DB2LOCATION++	DB8W	
DB2 table owner user ID	++DB2TABLEOWNER++	MQZ1BRK	
DB2 user ID for the component and commands	++DB2CURRENTSQLID++	SDRES01	
DB2 plan name	++DB2DSNACLIPLAN++	DSNACLI	
DB2 high-level qualifier	++DB2HLQ++	DB8W8	
DB2 run library value	++DB2RUNLIB++	DB8WU.RUNLIB.LOAD	
DB2 program value	++DB2SAMPLEPROGRAM++	DSNTEP2	
DB2 plan value	++DB2SAMPLEPROGRAMPLAN++	DSNTEP81	
DB2 broker database	++DB2DATABASE++	DMQZ1BRK	
DB2 storage group	++DB2STORAGEGROUP++	SMQZ1BRK	
DB2 bufferpool	++DB2BUFFERPOOL++	BP0	
DB2 index bufferpool	++DB2INDEXBP++	BP0	
DB2 LOB bufferpool	++DB2LOBBP++	BP0	

You can also find these tables in the WebSphere Message Broker Information Center. For the documentation version, select the path **Configuring the broker domain** → **Configuring broker domain components** → **Creating a broker** → **z/OS** → **Information required to create a broker** in the WebSphere Message Broker Information Center available on the Web*:*

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp

15.3.2 Allocate the broker PDSE

This new data set stores the JCL and configuration members for the migrated broker. The data set must exist so that the template JCL and template configuration members can be copied into it.

Allocate the data set

Each broker requires either a PDSE or a partitioned data set (PDS). However, a PDSE is preferable because free space is available without the need to compress the data set.

Create the broker's component data set, for example by using option 3.2 on ISPF. The name of the PDSE must be the same as the JCL variable ++COMPONENTDATASET++ collected in 15.3.1, "Collect required information for the broker" on page 271.

Allocate a data set with the following:

- Eight directory blocks
- 15 tracks (or one cylinder) of 3390 DASD with a record format of fixed blocked 80
- A suitable block size (for example, 27920)
- Data set type of LIBRARY

For information about creating the broker's PDSE, select the path **Configuring the broker domain** → **Configuring broker domain components** → **Creating a broker** → **z/OS** → **Creating the broker PDSE** in the WebSphere Message Broker Information Center available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae22410_.htm`

The migration example has allocated the following data set for the migrated broker to use.

Example 15-1 Example broker data set used in the migration scenario

```
MQSI6.MQZ1BRK.CNTL
```

Customize the broker data set

Copy the members in the PDSEs SBIPPROC and SBIPSAMP to the new broker data set. Use the data set utility panel (usually option 3.3 in ISPF) to copy the JCL listed on page 276.

Copy the following JCL from <hlq>.SBIPSAMP to +COMPONENTDATASET++:

- `BIPBPROF` (broker profile)
- `BIPDSNAO` (DB2 dsnaoini)

Copy the following members from <hlq>.SBIPPROC to +COMPONENTDATASET++. Note that instead of copying the entire data set, you can choose to copy only the following broker members:

- BIPALDB Job to alter DB2 table spaces and tables
- BIPALMQ Job to alter WebSphere MQ table spaces and tables
- BIPBRWS Job to run the **mqsibrowse** command
- BIPCHBK Job to issue the **mqsichangebroker** command
- BIPCHMS Job to issue the **mqsichangeflowstats** command
- BIPCHPR Job to issue the **mqsichangeproperties** command
- BIPCLMP Job to issue the **mqsiclearmqpubsub** command
- BIPCRBK Job to create a broker
- BIPCRDB Job to create the DB2 storage group, database and table spaces
- BIPDLBK Job to issue the **mqsideletebroker** command
- BIPDLDB Job to delete the DB2 storage group, database and table spaces
- BIPEDIT Job for JCL customization
- BIPFMLG Job to format the log
- BIPGEN Job to generate ENVFILE
- BIPJNMP Job to issue the **mqsijoinmqpubsub** command
- BIPLIST Job to issue the **mqsilist** command
- BIPLSMP Job to issue the **mqsilistmqpubsub** command
- BIPMGTB Job to change DB2 definitions when migrating from V2.1 to V6.0
- BIPRELG Job to read logs
- BIPRPMS Job to issue the **mqsireportflowstats** command
- BIPRPPR Job to issue the **mqsireportproperties** command
- BIPSDBP Job to define a data source, user ID, and password for user data sources
- BIPBRKP (started task)

Important: Customize the following files for your broker:

- Rename BIPEDIT to a unique name that identifies it with the current component, for example, MQZ1EDBK.
- Rename BIPBRKP to the same as the ++STARTEDTASKNAME++. In this example, it is MQZ1BRK.

This process has been replicated from the WebSphere Message Broker Information Center. For the most up-to-date version, select the path **Configuring the broker domain → Configuring broker domain components → Creating a broker → z/OS → Customizing the broker component data set** in the WebSphere Message Broker Information Center available on the Web*:*

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae22450_.htm

15.3.3 Customize the broker JCL

Customize the JCL templates that were copied to the broker's component data set for the environment in which they run. Use the ISPF macro BIPEDIT (which in this example has been renamed to MQZ1EDBK) to replace the JCL variables in all members of the broker's component data set. To use this macro, follow the steps detailed below.

Customize the BIPEDIT member

Edit the renamed BIPEDIT member and replace the example parameters with the variables collected in 15.3.1, "Collect required information for the broker" on page 271.

Example 15-2 shows how the ++INSTALL++ JCL parameter would look in the renamed BIPEDIT member.

Example 15-2 Editing the BIPEDIT member to replace the variables

```
"change ++INSTALL++ /usr/lpp/mqsi/V6R0M0                    all"
```

Important: Update all JCL variables listed in the renamed BIPEDIT member. If any are left blank or incorrectly configured, the JCL does not work correctly.

You can also add a substitution for the job card information at the top of the broker's JCL. For instance, the first line of the BIPCRBK job is //BIPCRBK JOB.

For convenience, you can replace the word JOB with a suitable job card for you system by using the renamed BIPEDIT macro. As shown in Example 15-3, add the following line to the renamed BIPEDIT macro and add any extra job parameters you require.

Example 15-3 Replacing the job card information

```
"change ' JOB ' ' JOB 1,MSGCLASS=H'                          all "
```

Activate the macro for execution

After you correctly configure the renamed BIPEDIT member, run it against the members of the component data set to replace the JCL variables in each member.

To activate the macro for execution, run the following TSO command in the ISPF panel, option 6:

```
ALTLIB ACTIVATE APPLICATION(EXEC) DA('++COMPONENTDATASET++')
```

This command is active for the local ISPF session for which it was issued.

Note: If you have split screen sessions, the other sessions will not be able to use this. If you use ISPF option 6 to issue the command, use ISPF option 3.4 to edit the data set. This enables you to use the edit command.

Replace the JCL variables

When the ISPF macro is active, replace the JCL variables for each member in the broker's component data set. For each member of the component data set, *except* the renamed BIPEDIT macro itself, follow these steps:

1. View or Edit the member.

   Tip: Use View instead of Edit on the first member until any problems in the ISPF macro are resolved. Alternatively, you can cancel the Edit session instead of using Save if you encounter problems.

2. Run the renamed BIPEDIT exec by typing its name on the command line (for example, `MQZ1EDBK`) to replace the JCL variables with the desired values.
3. Exit and save the changes if they have been successful.

   Tip: You can add the word `save` to your macro name to save your changes. For example, you could type or paste the following on the command line to replace the variables and then save the member:

   ```
   MQZ1EDBK;save
   ```

The JCL members in the component data set are now ready to run. However, before you run them, ensure that the final steps listed in the next section are followed.

Update BIPBPROF and BIPDSNAO

Before moving to the next stage, we recommend that you copy any additional changes you made in the WebSphere MQ Integrator V2.1 environment to the new WebSphere Message Broker V6.0 members as follows:

- Copy any changes made to the WebSphere MQ Integrator ENVFILE, to the BIPBPROF member.
- Copy any changes made to the Open Database Connectivity (ODBC) initialization file (dsnaoini) to the BIPDSNAO member.

The broker's component data set is now configured correctly, and it is now possible to migrate the broker.

Redirect STDOUT and STDERR

If you have not applied APAR OA11699 to the system, then you can redirect the output from the broker's jobs to the HFS. To do this, change the //STDOUT and //STDERR data definition (DD) cards to specify the following, as shown in Example 15-4.

Example 15-4 Redirecting output from the broker's jobs to the HFS

```
//STDOUT   DD PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//            PATHMODE=(SIRWXU,SIRWXG),
//            PATH='/u/mqz1brk/redirect.out'

//STDERR   DD PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//            PATHMODE=(SIRWXU,SIRWXG),
//            PATH='/u/mqz1brk/redirect.err'
```

Rename the files in the HFS so that each job creates its own file. This helps when debugging the jobs after they have run.

15.4 Migrate the broker

Before beginning the broker migration, ensure that all the steps in 15.3, "Premigration tasks" on page 271 have been successfully completed.

For the following steps, the JCL members in the broker's component data set should be correctly configured, security privileges should be correctly assigned, prerequisite software should be installed, and the broker should be successfully backed up.

15.4.1 Tasks to verify before migration

Before starting the migration, verify the following tasks.

Aggregations

Before migrating a broker, ensure that you do not have any aggregations in progress. When you migrate a broker to Message Broker V6.0, any live data being stored for aggregations in progress is lost.

Check bipimain

Use the **extattr** command to display the attributes of the object bipimain. For example:

```
extattr /usr/lpp/mqsi/V6ROMO/bin/bipimain
```

It should display `APF authorized = YES`. If it does not, use **extattr +a bipimain** to set this attribute. For example:

```
extattr +a /usr/lpp/mqsi/V6ROMO/bin/bipimain
```

15.4.2 Create the broker's home and component directories

Create the home and component directories in the HFS if they do not already exist. These are defined as ++HOME++ and ++COMPONENTDIRECTORY++ in 15.3.1, "Collect required information for the broker" on page 271.

Ensure that the broker's started task user ID and the broker system administrator's user ID have read, write and execute permissions to both the ++COMPONENTDIRECTORY++ and the ++HOME++ directory.

15.4.3 Generate the broker's ENVFILE

The first step in the broker migration is to create the ENVFILE, which is the environment file in which the broker stores all of its environment variables. When components are started, they read the ENVFILE to configure their environment.

Submit the BIPGEN job in the broker's component data set. This creates the ENVFILE in the ++HOME++ directory on the HFS.

Ensure that the BIPGEN job runs successfully and that the ENVFILE is created in the ++HOME++ directory. If any errors occur, resolve them and run the BIPGEN job again. Continue running the job and rectifying errors until the ENVFILE is successfully created. The migration cannot proceed until an ENVFILE has been generated by BIPGEN.

Attention: If any changes are made to the BIPBPROF member, you must re-run the BIPGEN job to generate a new ENVFILE.

Run the BIPGEN job at the same time as a planned restart of the broker, as this will maintain consistency between the ENVFILE in ++HOME++ and the copy that is being used by the running components.

15.4.4 Prepare DB2

WebSphere Message Broker V6.0 introduces the use of large object (LOB) tablespaces in the broker database on z/OS. They are not used in WebSphere MQ Integrator V2.1, therefore you need to create them at this point.

To make these changes, submit the BIPMGTB job. The user ID submitting this job should have the following file permissions:

- UPDATE access to the component PDSE
- READ/EXECUTE access to the installation directory
- READ/WRITE/EXECUTE access to the broker-specific directory

For DB2, the user ID also needs DBADMIN authority (or similar authority) to run the configuration jobs.

Check the output of this job to ensure that the DB2 changes have been successfully applied. If you encounter any errors during the database changes, you must restore the broker database backups and re-run the job. See 16.2, “Restoring the WebSphere MQ Integrator Version 2.1 environment” on page 301 for more information about how to restore database backups. After the restoration is successful, correct the problem and resubmit the job. Continue this until the BIPMGTB job succeeds. The database changes must be applied successfully before proceeding.

15.4.5 Perform a test migration of the broker

Before running the **`mqsimigratecomponents`** command properly, you can run it in a check mode. This performs a premigration check so you can see whether the command, when run properly, is likely to succeed.

To run this check, edit the BIPMGCMP job in the broker’s data set and alter the command defined at the bottom of the member. Change the command to specify the **`-c`** parameter, as shown in Example 15-5.

Example 15-5 Specifying the -c parameter in the BIPMGCMP job

```
BPXBATCH PGM -
/usr/lpp/mqsi/V6ROMO/bin/-
mqsimigratecomponents -
MQW1BRK -
-c
```

Submit the edited BIPMGCMP and check the output of the job. If successful, the job should display the message as shown in Example 15-6.

Example 15-6 Expected output from the BIPMGCMP job

```
BIP8680I: Pre-migration check succeeded.
```

If the premigration check fails, rectify the cause of the failure and run the migration check again. Continue this process until the premigration check succeeds.

Note: You can run the **mqsimigratecomponents** command with a full set of command line options. For a full list of options, select the path **Reference** → **Operations** → **Commands** → **Runtime Commands** → **mqsimigratecomponents command** on the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an26150_.htm

15.4.6 Migrating the broker's components

After the migration check has succeeded, run the **mqsimigratecomponents** command without the check parameter. This causes the command to perform a full migration of the broker.

You can configure the **mqsimigratecomponents** command to migrate the file system, WebSphere MQ, or the broker database in separate stages. This is useful if the user ID performing the migration does not have the required authority to change WebSphere MQ or the broker's database tables. In this situation, a job can be configured and run by a different user with the correct authorities to migrate that component.

To perform any of the migration steps individually, use the parameters of the **mqsimigratecomponents** command as listed in Table 15-4.

Table 15-4 Commands to migrate individual broker components

Parameter	Migrates component
-1	Does only registry and file system work. Use the -1 parameter before the -2 or -3 parameters.
-2	Does only WebSphere MQ work.
-3	Does only database work.

To perform the actual migration, edit the BIPMGCMP job in the broker's data set and alter the command defined at the bottom of the member. Remove the **-c** option, which was specified in the previous step.

To migrate a single component, use the parameters shown above. To migrate all components, run the command with no parameters except for the broker name.

Tip: Ensure that the continuation character at the end of the previous line is also removed.

Verify that the BIPMGCMP job succeeds and that the components have been successfully migrated. Check for the following messages from the job's output, as shown in Example 15-7.

Example 15-7 Output from the BIPMGCMP job running in non-check mode

```
BIP8768I: Finished registry migration for component 'MQZ1BRK'.
Registry data has been successfully migrated for component 'MQZ1BRK'.
BIP8669I: Database migration successful
BIP8789I: Finished WebSphere MQ queue migration for component 'MQZ1BRK'. Queue
data has been successfully migrated for component 'MQZ1BRK'.
BIP8071I: Successful command completion.
```

Also check the job's output to see if any message flows or message sets require redeployment.

Note: If the BIPMGCMP job fails to run successfully, restore the backup taken before the migration began. This restores the broker to its previous WebSphere MQ Integrator V2.1 level.

See 16.2, "Restoring the WebSphere MQ Integrator Version 2.1 environment" on page 301 for the steps to restore this backup. Once the restoration is complete, re-run the BIPMGCMP job.

15.4.7 Copy the started task JCL

The SYS1.PROCLIB concatenation already contains members for the WebSphere MQ Integrator V2.1 broker's started task. Rename or back up these before copying in the new WebSphere Message Broker V6.0 broker member.

Copy the renamed BIPBRKP member (in this example, MQZ1BRK) to the SYS1.PROCLIB concatenation. The broker is now ready to be started.

15.4.8 Start the migrated broker

You can issue the command to start or stop a broker from SDSF, from NetView, or from or other products that provide access to the console. In the following examples the commands are given for SDSF, and have the / command prefix. The syntax may be different on other tools.

From SDSF, start the component; for example:

```
/S MQZ1BRK
```

Check the log for the started task to ensure that the component started successfully. Example 15-8 shows the expected broker startup messages.

Example 15-8 Expected broker startup messages

```
+BIP9141I MQZ1BRK 0 THE COMPONENT WAS STARTED. : ImbControlService(744)

+BIP9108I MQZ1BRK 0 BROKER SERVICE VALUE IS IMBSERV.V6ROM00.GOLD..... :
ImbControlService(749)

+BIP2001I MQZ1BRK 0 THE WEBSPHERE MESSAGE BROKERS SERVICE HAS STARTED AT
VERSION 6000; PROCESS ID 852085.:
ImbControlService(773)
```

After the broker starts, any execution groups that were migrated also start. Check the logs to ensure they have successfully started. The execution group logs should contain the message shown in Example 15-9.

Example 15-9 Execution group startup message

```
+BIP2201I MQZ1BRK EG1 0 EXECUTION GROUP STARTED: PROCESS

'84738167';
```

This message is followed by more startup information. Example 15-10 shows the final message displayed after the execution group is successfully up and running.

Example 15-10 Execution group startup completion message

```
+BIP2154I MQZ1BRK EG1 26 EXECUTION GROUP FINISHED WITH CONFIGURATION MESSAGE
```

However, if the component fails to start correctly, use the error information in the log to rectify the problem. Ensure that all of the prerequisite steps in this chapter have been performed successfully.

Note: When the main broker task (MQZ2BRK) starts up, it runs the step VFYDB2MQ. This ensures that the broker has the required access to DB2, WebSphere MQ, and the HFS.

15.4.9 Verify the migration

At this stage, the broker and its execution groups have successfully started and the VFYDB2MQ step has run. To verify that the broker has successfully migrated, refer to Chapter 16, “Migration verification” on page 287.

15.4.10 Verifying the migration for Rules and Formatter Extension

If the new WebSphere Message Broker V6.0 is using the Rules and Formatter Extension, then check the environment variable NNSY_ROOT by examining the ENVFILE in the ++HOME++ directory.

If it does not exist in the ENVFILE, add NNSY_ROOT to the BIPBPROF member and generate a new ENVFILE by running the BIPGEN job. For example, add the following to BIPBPROF:

```
NNSY_ROOT=/usr/lpp/mqsi/V6R0M0/nnsy
```

The nnsyreg.dat sample file of New Era Of Networks is in the /usr/lpp/mqsi/V6R0M0/nnsy/rulfmt41/examples directory. Check the differences between the current nnsyreg.dat file and Message Broker V6.0 sample.

Edit the nnsyreg.dat file and change the NNOT_SHARED_LIBRARY parameter in all sessions.

New Era Of Networks searches for the current nnsyreg.dat in the following directories:

- Current working directory
- Directory specified by the environment variable NN_CONFIG_FILE_PATH
- Directory specified by the environment variable NNSY_ROOT

Ensure that the correct configuration file is used.

15.5 Creating a new broker on z/OS

The steps covered in this chapter discuss the migration of a broker from WebSphere MQ Integrator V2.1 to WebSphere Message Broker V6.0.

If you require a new broker on z/OS, then select the path **Configuring the broker domain** → **Configuring broker domain components** → **z/OS** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae22400_.htm

16

Migration verification

This chapter discusses the tasks you need to perform in order to verify a successful migration to WebSphere Message Broker V6.0. It also covers the detailed procedures for restoring the WebSphere MQ Integrator V2.1 environment from backups.

The migration verification and restoration is based on the environment described in Chapter 7, “WebSphere MQ Integrator Version 2.1 environment setup” on page 107.

16.1 Post-migration tasks

After completing the migration to WebSphere Message Broker V6.0, there are a number of verification tasks to perform. These include:

- Verifying the components by listing, starting and stopping:
  - The Configuration Manager
  - The User Name Server
  - Brokers
- Verifying the development environment by:
  - Starting the Message Brokers Toolkit
  - Assigning and deploying resources to brokers
- Testing that the migrated message flows and message sets work as expected
- Reviewing the logs for handling any problems

In case of an unsuccessful migration, you may have to restore the backups of the previous version. The following list includes the components to be restored:

- The WebSphere MQ Integrator V2.1 components and their configurations
- Related databases
- Development resources
- User environment

After the migration verification, you can choose to remove items no longer required; for example:

- The Configuration Manager database
- The Message Repository Manager (MRM) database
- Old versions of products

There may also be post-migration verification tasks specific to your environment. Include anything specifically helpful to your environment. For more details about the tasks that are not covered here, refer to the WebSphere Message Broker V6.0 Information Center available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp`

The following sections cover how to implement the verification, restoration, and cleanup tasks listed above.

16.1.1 Verifying the components

Starting and stopping WebSphere Message Broker V6.0 components without failure is a basic checkpoint to verify that migration has completed successfully.

You can control the WebSphere Message Broker V6.0 components with a set of commands.

Before running any WebSphere Message Broker V6.0 commands, run the following command to set up the environment:

- On Windows platforms, enter the following command from the command prompt:

  broker_install_directory/bin/mqsiprofile.cmd

  Alternatively, launch the Command Console, which runs **mqsiprofile.cmd** automatically at startup, by selecting **Start → IBM WebSphere Message Broker V6.0 → Command Console**

- On UNIX platforms, enter the following command from the command line:

  . *broker_install_directory*/bin/mqsiprofile

If you encounter any problem controlling the components with commands, check the related logs to ensure that the components ran correctly. Refer to 16.1.4, "Checking the logs and where to find them" on page 297 for information about where to find the related logs on each platform.

Important: You can issue commands to control a component on the z/OS platform from System Display and Search Facility (SDSF), Netview or other products that provide access to the console.

In the following examples, the commands are given for SDSF, and have the "/" command prefix. The syntax may be different on other tools.

Verification using the mqsimigratecomponents command

The **mqsimigratecomponents** command is provided in WebSphere Message Broker V6.0 to check, migrate and verify the components. By using this command with **-v** and **-t** parameters, you can check whether:

- The correct database tables and queues exist for the specified version.
- The registry is in the correct format for the specified version.

After migrating the components to WebSphere Message Broker V6.0, we recommend that you enter the following command from the command line on distributed platforms:

```
mqsimigratecomponents -v -t 6 component_name
```

Example 16-1 shows a successful command output.

Example 16-1 Verification using the mqsimigratecomponents command

```
C:\>mqsimigratecomponents -v -t 6 BKW1
BIP8849I: Broker 'BKW1' (Version 6.0) with Queue Manager 'BKW1QM' and Data
Source 'BKW1DB' specified for migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

On z/OS, the BIPMGCMP job is provided. Run this job in the broker's component data set. Specify the correct parameters at the bottom of this job for the **mqsimigratecomponents** command.

Listing the components

Run the appropriate commands for each platform, as explained here:

- To list the components, use the **mqsilist** command on the distributed platforms.
- On z/OS, enter the **list** command as follows:

  ```
  /F broker_name, list
  ```

  Substitute your broker name for *broker_name*.

> **Note:** Use the **list** command on z/OS to list all the execution groups defined to a specific broker, or all the message flows contained in a named execution group on a specific broker. This is different from the **mqsilist** command on the distributed platforms, which lists all components.
>
> For more information about the **list** command on z/OS, select the path **Reference** → **Operations** → **Commands** → **Runtime commands** → **mqsilist** in the WebSphere Message Broker V6.0 Information Center available on the Web:
>
> http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an07250_.htm

Starting and stopping the Configuration Manager

In WebSphere Message Broker V6.0, the Configuration Manager can run on all supported run-time platforms. Run the appropriate commands for each system, as explained here.

Distributed platforms

On the distributed platforms:

- Start Configuration Manager by entering this command on the command line:

```
mqsistart configuration_manager_name
```

- Stop Configuration Manager by entering this command on the command line:

```
mqsistop configuration_manager_name
```

Substitute your Configuration Manager name for *configuration_manager_name*.

z/OS system

On z/OS:

- Start Configuration Manager by entering this command on the command line:

```
/S configuration_manager_name
```

- Stop Configuration Manager by entering this command on the command line:

```
/P configuration_manager_name
```

Substitute your Configuration Manager name for *configuration_manager_name*.

For more information about the start and stop command on z/OS, select the path **Administering the broker domain → Starting and stopping a Configuration Manager → z/OS** in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae24130_.htm

Starting and stopping the User Name Server

Run the appropriate commands for each system, as explained here.

Distributed platforms

On the distributed platforms:

- Start the User Name Server by entering this command on the command line:

```
mqsistart UserNameServer
```

- Stop the User Name Server by entering this command on the command line:

```
mqsistop UserNameServer
```

z/OS system

On z/OS:

- Start the User Name Server by entering this command:

  ```
  /S user_name_server_name
  ```

- Stop the User Name Server by entering this command:

  ```
  /P user_name_server_name
  ```

Substitute your User Name Server name for *user_name_server_name*.

For more information about the start and stop command on z/OS, select the path **Administering the broker domain → Starting and stopping a User Name Server → z/OS** in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an04019_.htm

Starting and stopping a broker

Run the appropriate commands for each system as explained below.

Distributed platforms

On the distributed platforms:

- Start a broker by entering this command on the command line:

  ```
  mqsistart broker_name
  ```

- Stop a broker by entering this command on the command line:

  ```
  mqsistop broker_name
  ```

Substitute your broker name for *broker_name*.

z/OS system

On z/OS:

- Start a broker by entering this command:

  ```
  /S broker_name
  ```

- Stop a broker by entering this command:

  ```
  /P broker_name
  ```

Substitute your broker name for *broker_name*.

For more information about the start and stop command on z/OS, select the path **Administering the broker domain → Starting and stopping a broker → z/OS**

in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an04016_.htm

16.1.2 Verifying the development environment

In WebSphere Message Broker V6.0, you can install the Message Brokers Toolkit on Windows or Linux (x86 platform). To ensure that the new version of Message Brokers Toolkit is installed and the broker resources are migrated correctly, start the Message Brokers Toolkit and attempt to develop and deploy applications.

Launching the Message Brokers Toolkit

The following sections give an overview of how to launch the V6.0 Message Brokers Toolkit on each platform. If there is any problem launching and working with the Message Brokers Toolkit, check the related logs. Refer to 16.1.4, "Checking the logs and where to find them" on page 297 for information about where you can find the related logs on each platform.

Windows

On Windows, you need administrator authority for installation verification. You should complete verification with the same user ID that was used to complete installation. To launch the Message Brokers Toolkit, perform one of the following operations:

- Enter the following command from the command line:

  toolkit_install_directory\wmbt.exe

- Select **Start** → **IBM WebSphere Message Broker 6.0** → **WebSphere Message Brokers Toolkit** from the Windows Start menu.

Linux

On Linux, you do not need root authority to launch the Message Brokers Toolkit. Log in with any user ID *except* root. To launch the Message Brokers Toolkit, perform one of the following operations:

- Enter the following command from the command line:

  toolkit_install_directory/wmbt.bin

- Select **WebSphere Message Brokers Toolkit** from the Linux main menu.

Tip: Do not log in to a UNIX or Linux system using a root account unless necessary. On Linux, any authorized user except root can be used for WebSphere Message Broker V6.0 application development.

Configuring the location of a workspace

When you first launch the Message Brokers Toolkit, you will be prompted to specify the location of your workspace. Refer to the following checkpoints:

- You can accept the default path, or you can specify your choice by locating it using the **Browse** button.
- Select the check box `Use this as the default and do not ask again` to inhibit the display of the workspace dialog next time you launch the Message Brokers Toolkit, as shown in Figure 16-1.

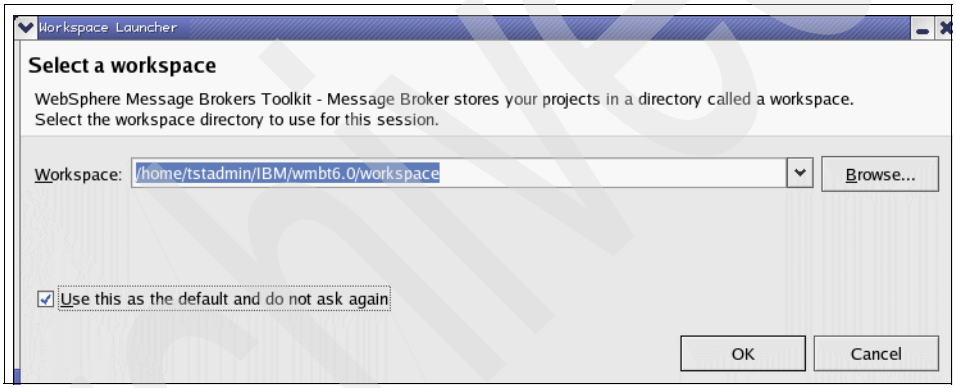


Figure 16-1 Selecting a workspace

Testing the development environment

The following sections discuss how you can test development and deployment using the new Message Brokers Toolkit. If you encounter any problem in the Message Brokers Toolkit or while deploying to the broker, check the related logs. Refer to 16.1.4, “Checking the logs and where to find them” on page 297 for information about where you can find the related logs on each platform.

Details about importing, developing and deploying the broker resources are not covered here. For more details about developing, refer to **Developing applications** in the WebSphere Message Broker V6.0 Information Center available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac18710_.htm`

The general steps are as follows:

1. Launch the Message Brokers Toolkit.
2. Develop or import the WebSphere Message Broker resources in the Broker Application Development perspective.
3. Create a broker archive file and add the resources you want to deploy to the broker.
4. Switch to the Broker Administration perspective and connect to the WebSphere Message Broker domain.
5. Drag and drop your broker archive to the execution group in your broker.
6. Verify that the deployment is successful.

Deploying the migrated applications

The new development environment and the migration process can be verified by deploying your applications, migrated from the previous version, to the new version of the broker.

Note: Before deploying anything to the broker, ensure that you have migrated your previous version of broker applications according to the procedures described in Chapter 10, “Migrating the WebSphere MQ Integrator V2.1 Control Center resources” on page 145.

Using the samples

We recommend that you use the samples provided with WebSphere Message Broker V6.0 to verify your development environment. Review the following samples in the Samples Gallery:

- WebSphere Message Brokers Getting Started samples provide:
  - Pager
  - Scribble
  - Soccer result
- WebSphere Message Brokers Application samples provide:
  - Airline reservations
  - Coordinated request replies
  - Data warehouses
  - Error handlers
  - Large messaging
  - Message routing
  - User-defined extensions
  - Video rental message sets

For the details about running each sample, perform the following operations in the Message Brokers Toolkit:

1. Select **Help** → **Samples Gallery** from the menu.
2. In the window that appears, browse to one of the following locations in the left pane:
   - **Application samples** → **Message Brokers(Getting started)**
   - **Application samples** → **Message Brokers**

The features that each sample demonstrates are documented in the WebSphere Message Broker Information Center under the topic **Samples**, available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ax20230_.htm

16.1.3 Verifying the applications

You can run the applications in the same way as in the previous environment. To verify the migration using the samples provided with WebSphere Message Broker V6.0, and for information about how you can run the sample applications, refer to the topic **Samples** in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ax20230_.htm

To test the environment, include a broad range of function by including all or a subset of the following functions in your verification scenario:

- WebSphere MQInput/MQOutput/Reply
- Compute/Mapping/Filter/XMLTransformation
- Database
- Aggregation
- Pub/Sub
- Subflows
- Realtime
- HTTPInput/Reply/Request
- New Era Of Networks
- Message Sets

If there is a problem running the applications, check the related logs to ensure that the components run correctly. Refer to 16.1.4, “Checking the logs and where to find them” on page 297 for information about where to find the related logs on each platform.

16.1.4 Checking the logs and where to find them

If you have any problem controlling the components, working with the Message Brokers Toolkit and running your applications, check the logs to ensure that your migration or configuration is correct. The following sections explain where to find the appropriate logs for each WebSphere Message Broker V6.0 task.

Local error logs for WebSphere Message Broker components

If you encounter any problems while running WebSphere Message Broker V6.0 components such as brokers, check the local error log.

Windows systems: Event Viewer

On Windows, the application log contains events logged by applications or programs. To view the application log, enter **`eventvwr`** from the command line or select **Start** → **Control Panel** → **Administrative Tools** → **Event Viewer**.

This opens the Windows Event Viewer. In the left pane, click the Application Log. All events that have been logged by applications or programs—including WebSphere Message Broker V6.0—are displayed there.

UNIX systems: syslog

Before checking the syslog file, make sure that you have configured the syslog daemon properly to redirect the related syslog messages to a separate file. On UNIX systems, all WebSphere Message Broker messages are sent to the syslog.

The entry that indicates where all WebSphere Message Broker messages are redirected to must be added to the /etc/syslog.conf file. For example:

```
user.info /var/log/user.log
```

In this case, the WebSphere Message Broker messages are found in /var/log/user.log.

z/OS: console log and job log

On z/OS, the broker writes messages to the appropriate z/OS system log and job logs.

The broker writes all of its messages to the z/OS system console log. Messages from all address spaces running on the z/OS system are written in this log. It is easy to identify jobs associated with the broker in the console log because of the naming of broker address spaces. Using the console log, you can see the order of event reporting for different products. This is helpful for cross-product problem determination.

The broker control address space and each of the execution group address spaces has its own job log. Selecting the job log for the appropriate broker

address space allows you to see all messages relating to that address space. This is helpful in a busy system where the system console log might have many messages from different products, obscuring the information in which you are interested.

Message Brokers Toolkit logs

In Message Brokers Toolkit, review the domain's Event Log after deploying anything to the broker domain, and review the Eclipse log if there is a problem executing the Message Brokers Toolkit.

Message Brokers Toolkit Event Log

After deployment to the broker domain, we recommend that you check the domain's Event Log to ensure that the deployment is successful. Follow these steps in the Message Brokers Toolkit:

1. Switch to the Broker Administration perspective.
2. Move to the Domains view where there will be an entry Event Log with a flag icon.
3. Double-click the Event Log icon.

Eclipse log

If there is a problem with the Message Brokers Toolkit, review the Eclipse log to determine what the problem is. To view the Eclipse error log, perform the following steps in the Message Brokers Toolkit:

1. From the menu, select **Window** → **Open Perspective** → **Other**.
2. When a dialog box appears, check **Show all** and select **Plug-in Development** in the list, as illustrated in Figure 16-2.

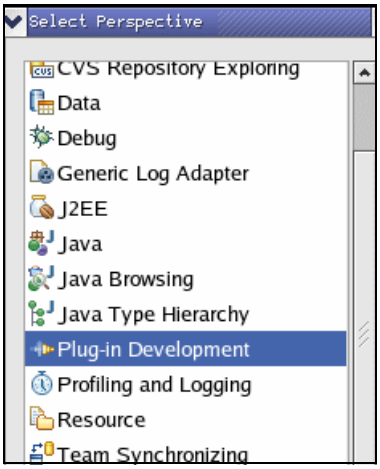


Figure 16-2 Opening the Plug-in Development perspective

3. When the Plug-in Development perspective is launched, in the bottom right pane of the Tasks view, click the Error Log tab. The error log is displayed, as illustrated in Figure 16-3.

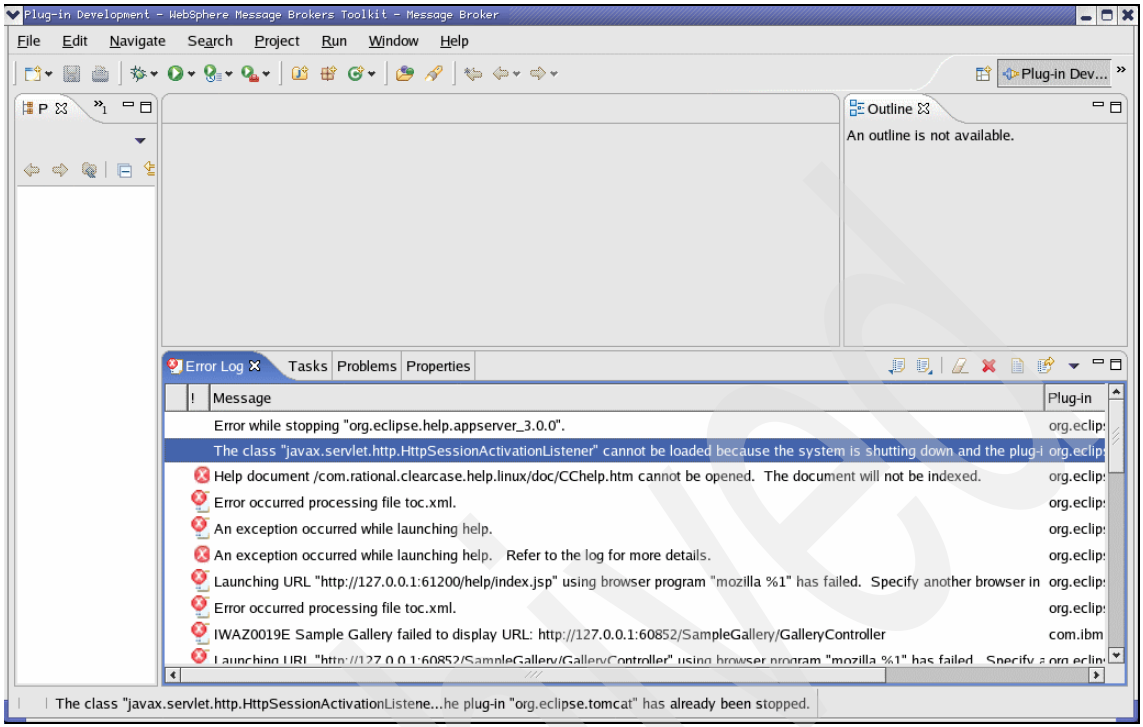


Figure 16-3 Eclipse error log view

4. If an error displays a plus sign (+), it means it is a complex problem and there are a number of errors contributing to it. Click the plus sign to view the individual errors.
5. To see the details of a particular problem, double-click the entry in the Tasks view. A separate window is displayed showing more details of the error, as illustrated in Figure 16-4.

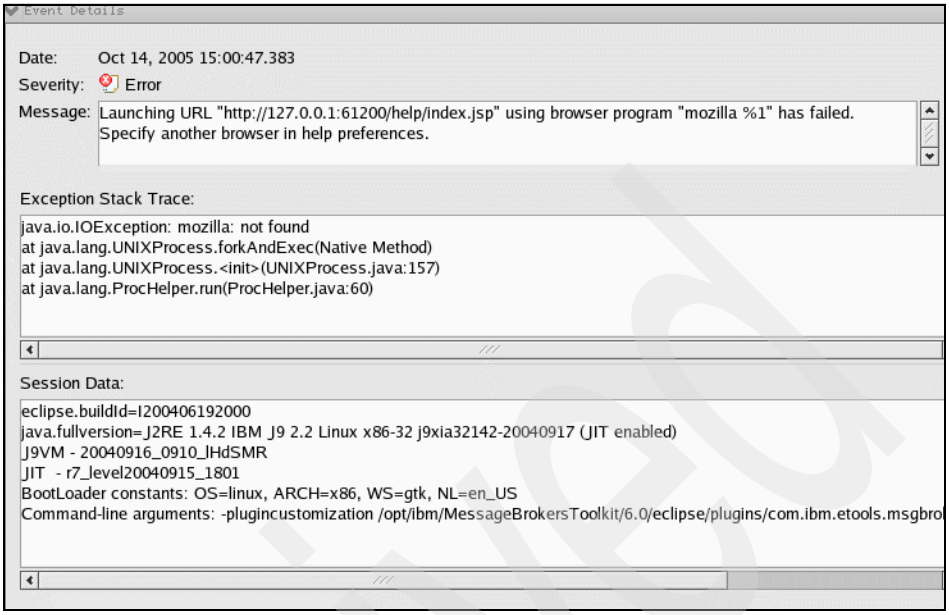


Figure 16-4 Eclipse event details view

WebSphere MQ logs

For solutions to any problems related to WebSphere MQ, refer to the topic **Problem Determination** in the WebSphere MQ Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmqv6/v6r0/topic/com.ibm.mq.amqzag.doc/mxprobl.htm

DB2 Universal Database logs

For solutions to any problems related to DB2 Universal Database, refer to the topic **Analyzing db2diag.log files using db2diag** in the DB2 Universal Database Information Center at available on the Web:

http://publib.boulder.ibm.com/infocenter/db2help/topic/com.ibm.db2.udb.pd.doc/pd/c0020701.htm

16.2 Restoring the WebSphere MQ Integrator Version 2.1 environment

You can restore the WebSphere MQ Integrator V2.1 environment using the backups taken in Chapter 9, “Backing up the WebSphere MQ Integrator V2.1

domain" on page 133. You must restore the following components to return to the previous level:

- The Configuration Manager
- Brokers
- The User Name Server
- Workspace
- User profile and Open Database Connectivity (ODBC) connection file

The following sections discuss the detailed restoration procedure for each component.

Important: You cannot restore components to their WebSphere MQ Integrator V2.1 level with the **mqsimigratecomponents** command provided in WebSphere Message Broker V6.0.

This command restores WebSphere Message Broker V6.0 components only to WebSphere Business Integration Message Broker V5.0 level. Any changes made in the WebSphere Message Broker V6.0 environment since migration are lost when you restore to the previous level.

We recommend that you carefully review the impact of restoring before embarking on the changes.

Restoring the components

To restore the WebSphere MQ Integrator V2.1 components, perform the following steps and refer to the appropriate product documentation if you need assistance with the specific commands:

1. Ensure that all deployment actions are completed and no Message Brokers Toolkit is connected to the V6.0 broker domain.
2. Stop the brokers, the Configuration Manager, and the User Name Server by using the **mqsistop** command or the z/OS equivalent.
3. Delete WebSphere Message Broker V6.0 brokers, Configuration Managers, and the User Name Server. Consider the following options:
   - To delete a broker, enter the **mqsideletebroker** command with an option **-w** to delete any trace files in the working path.

     For the z/OS equivalent, select the path **Configuring the broker domain** → **Configuring broker domain components** → **Deleting a broker** → **z/OS** in the WebSphere Message Broker V6.0 Information Center available on the Web:

     http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae12970_.htm

– To delete the Configuration Manager, enter the **mqsideleteconfigmgr** command with options **-w -n** to delete any trace files in the working path and its repository.

  For the z/OS equivalent, select the path **Configuring the broker domain** → **Configuring broker domain components** → **Deleting a Configuration Manager** → **z/OS** in the WebSphere Message Broker V6.0 information available on the Web:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae24480_.htm

4. If required, recreate the databases for the brokers, the Configuration Manager, and the Message Repository Manager.

   For the DB2 Universal Database command on the distributed platforms, refer to Example 16-2.

Example 16-2 The DB2 Universal Database command for creating a database

```
$ db2 CREATE DB BKDB
DB20000I  The CREATE DATABASE command completed successfully.
```

5. Recreate WebSphere MQ Integrator V2.1 brokers, the Configuration Manager, and the User Name Server as follows:
   – On the distributed platforms, use the **mqsicreatebroker**, **mqsicreateconfigmgr**, and **mqsicreateusernameserver** commands, respectively.
   – For the z/OS equivalent, refer to WebSphere MQ Integrator V2.1 documentation.
6. Follow the steps in “Restoring the DB2 Universal Database” on page 304 to restore the databases for the brokers, the Configuration Manager, and the Message Repository Manager.
7. From the backup, restore the Universally Unique Identifier (UUID) of each broker by performing one of the following operations:
   – On Windows, import the previously exported Windows registry file (*.reg) in the regedit tool by selecting **File** → **Import**.

     Or, change the value of HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphereMQIntegrator*broker_name*\BrokerUUID to the value you have recorded before the migration.
   – On UNIX, copy the previously backed-up registry file to $MQSI_REGISTRY/registry/*broker_name*/BrokerUUID.

     Or, change the value within the BrokerUUID file to the value you have recorded before the migration.

– On z/OS, copy the previously backed-up registry file to $MQSI_REGISTRY/registry/*broker_name*/CurrentVersion/BrokerUUID.

  Or, change the value within the BrokerUUID file to the value you have recorded before the migration.

8. Start the brokers, the Configuration Manager, and the User Name Server using the **mqsistart** command. Refer to WebSphere MQ Integrator V2.1 documentation for the z/OS equivalent. If there is a problem starting these components, check the related log.
9. Launch the Control Center and select the Topology view. To redeploy the domain configuration, select **File** → **Deploy** → **Complete configuration (all types)** → **Normal** to ensure that the configuration across the broker domain is consistent.

Restoring the DB2 Universal Database

Restoring the WebSphere MQ Integrator V2.1 components may require an additional database restoration procedure. Before restoring databases, make sure that these have been backed up correctly. Refer to 9.2, “Database backup” on page 134, for details about how to back up a database.

The method of backing up and restoring a database depends on the platform used, as described here.

UNIX and Windows systems

To restore the databases required for the WebSphere MQ Integrator V2.1 components on the distributed platforms, follow these steps:

1. Open an appropriate DB2 Universal Database command line.
2. Enter the appropriate commands to restore a database, as shown in Example 16-3.

Example 16-3 DB2 Universal Database commands for database restoration example on the distributed systems

```
db2 CONNECT TO BKA1DB
db2 QUIESCE DATABASE IMMEDIATE FORCE CONNECTIONS
db2 PRUNE HISTORY 9999 WITH FORCE OPTION
db2 CONNECT RESET
db2 RESTORE DATABASE BKA1DB FROM "/home/tstadmin/backup/db2_backup" TAKEN AT
<time stamp noted from backup, like 20051010142634> WITH 2 BUFFERS BUFFER 1024
REPLACE EXISTING PARALLELISM 1 WITHOUT ROLLING FORWARD WITHOUT PROMPTING
db2 CONNECT TO BKA1DB
db2 UNQUIESCE DATABASE
db2 CONNECT RESET
```

This example sequence of operations gives you a 2359 warning message, but you can ignore it because it means that an existing database is being replaced with a new one.

3. Verify that your database has been restored successfully. For example, try to connect to the database and list tables.

z/OS system

Example 16-4 shows the job code language (JCL) for restoring the database.

Example 16-4 An example of JCL for restoring the database

```
//DBRECOVER JOB NOTIFY=&SYSUID,MSGLEVEL=(1,1),MSGCLASS=T
//PROCS JCLLIB ORDER=(XXXXXX.PROCLIB)
/*JOBPARM SYSAFF=XXXX
//UTIL EXEC DSNUPROC,SYSTEM=XXXX,UID='XXXXXXXX',UTPROC=''
//*********************************************
//*
//* JCL FOR THE IMAGE COPY UTILITY
//* (c)Copyright IBM Corp. 2002
//* DATE: 13/10/05
//* CREATED BY: DEANEJ
//* UPDATED BY: Mark Hiscock
//*
//* COPY THIS MEMBER INTO YOUR BROKER PDSE AND
//* EDIT THIS JOB AND UPDATE THE FOLLOWING:
//*
//* 1. The proclib for your DB2 (line 2)
//* 2. The system name for the job to run on (line 3)
//* 3. The DB2 sub system and user ID (line 4)
//* 4. The Dataset names (lines 30 and 32)
//* 5. The database name (line 40)
//* 6. Most IMPORTANTLY the LRSN value (line 41)
//*
//* THE JOBS SHOULD BE RUN IN THE FOLLOWING ORDER
//*
//* 1. BACKUP DB WITH DBBACKUP
//* 2. GET LRSN VALUE FOR FIC WITH DBREPORT
//* 3. RECOVER DB WITH DBRESTOR
//*
//*********************************************
//*
// SYSREC   DD DSN=WMQI.DMQW1BRK.STEP1.SYSREC,DISP=(MOD,DELETE,CATLG),
//          UNIT=SYSDA,SPACE=(8000,(20,20),,,ROUND)
// SYSUT1   DD DSN=WMQI.DMQW1BRK.STEP1.SYSUT1,DISP=(MOD,DELETE,CATLG),
//          UNIT=SYSDA,SPACE=(8000,(20,20),,,ROUND)
//*
//* TOLOGPPINT USES THE HEX VALUE OF THE LRSN FOR THE FULL IMAGE COPY
//* OF YOUR CHOICE. YOU CAN OBTAIN THIS BY RUNNING THE REPORT RECOVERY
```

```
//* JOB DBREPORT.
//*
//DSNUPROC.SYSIN DD *
LISTDEF COMPONENT INCLUDE TABLESPACES DATABASE XXXXXXXX ALL
RECOVER LIST COMPONENT REUSE TOLOGPOINT X'LRSNXXXXXVALUE'
REBUILD INDEX LIST COMPONENT REUSE
//
/*
```

This JCL sample can be downloaded from the Web. See Appendix C, “Additional material” on page 555 for instructions.

Restoring the workspace

Before restoring to the WebSphere MQ Integrator V2.1 Control Center environment, ensure that you have backed up your workspace resources correctly. For detailed information about how to back up your workspace resources, refer to 9.2, “Database backup” on page 134.

Using the workspace file

Start the Control Center and select **File** → **Open Workspace**. You can then browse to open the backed up copy of your WebSphere MQ Integrator V2.1 workspace XML file.

Importing the resources manually

If the migration was not successful and you still want to migrate the broker resources into the Configuration Manager repository and Message Repository, then you can migrate the message flows and the message sets manually. Here we explain how to import the message flows, the Topology, and the Topic.

In case the previously exported files include information about the Topology or the Topic of the broker domain, check the Topology or the Topic in the Control Center:

1. Select **File** → **Import to Workspace**.
2. The Import Resources dialog box appears. Choose what you want to import. Check **Replace resources in configuration manager which are not locked** if you want to overwrite the resources that are already in the repository and being shared, as shown in Figure 16-5.

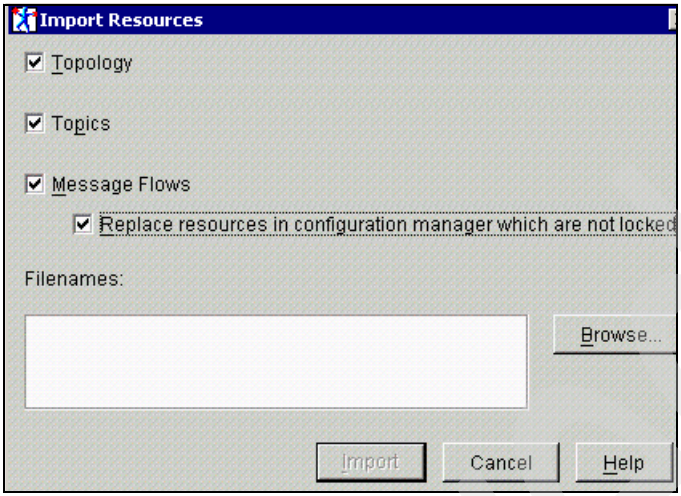


Figure 16-5 Setting import options

3. Click **Browse** to open the previously exported files.
4. Click **Import** to complete the import. A dialog box appears indicating what has been imported into your workspace.
5. If the resource is newly created, you will see the blue box icon beside it. You can check the resource by selecting **File** → **Check In** → **All(Save to Shared)**.

To import the message sets manually, enter the **`mqsiimpexpmsgset`** command from the command line, as shown in Example 16-5.

Example 16-5 A mqsiimpexpmsgset command

```
mqsiimpexpmsgset -i -n MRMDB -u tstadmin -p r3db0Ok -f RequestLoan.mrp
BIP8071I: Successful command completion
```

If the message set migration is successful, you can add it to your workspace. Switch to Message Sets view, right-click the Message Sets folder, and select **Add to Workspace - Message Set**. You can then browse through the list of message sets that appears.

Restoring the user profile and the ODBC connection file

In case the user profile and the ODBC connections file have been changed for the WebSphere Message Broker V6.0 environment on UNIX, restore them to their previous levels.

If your user profile has been modified to run the WebSphere Message Broker V6.0 profile, correct it to point to the path of the previous version. You can find the samples of the V2.1 profile in the following path:

```
broker_install_directory/sample/profiles
```

You can change or run your user profile as follows:

```
. broker_install_directory/sample/profiles/profile.aix
```

If the ODBCINI system environment variable has been set to point to the new ODBC driver from WebSphere Message Broker V6.0, reset it to point to the previous version of the file path; for example:

```
export ODBCINI=/var/mqsi/odbc/.odbc.ini
```

16.3 Additional post-migration tasks

This section discusses additional post-migration tasks that are optional. These include:

- Cleaning up obsolete database or tables in the Configuration Manager and Message Repository Manager (MRM) databases
- Removing the previous version of WebSphere MQ Integrator V2.1

Dropping the unnecessary database or tables

In WebSphere Message Broker V6.0, the Configuration Manager does not use an external database to store the domain configuration. It uses an internal repository instead.

Therefore, you can drop the Configuration Manager database. To drop the database, enter the command from the DB2 Universal Database command line as shown in Example 16-6.

Example 16-6 Dropping a database

```
db2 DROP DB database_name
```

If you prefer not to drop the database, you can choose to drop only the unnecessary tables. To do this on distributed platforms, enter the command from the DB2 Universal Database command line as shown in Example 16-7.

Example 16-7 Dropping a specific table from a database

```
db2 CONNECT TO database_name
db2 DROP TABLE table_name
```

The Configuration Manager does not use the following tables in a V6.0 environment:

- CBROKER
- CBROKERCEG
- CCOLLECTIVE
- CCOLLECTIVECBROKER
- CDELETE
- CEG
- CEGCMSGFLOW
- CEGCMSGPROJECT
- CLOG
- CMSGFLOW
- CMSGPROJECT
- CNEIGHBOURS
- COUTSTANDING
- CSUBSCRIBE
- CTOPIC
- CTOPICCTOPIC
- CTOPOLOGY
- CTRACE
- CUUIDLOCKS

In WebSphere Message Broker V6.0, message set resources are stored in the Message Brokers Toolkit's workspace and not a database. Therefore, you can also drop the Message Repository Manager (MRM) database.

Uninstalling WebSphere MQ Integrator V2.1

WebSphere Message Broker V6.0 allows coexistence with previous versions of WebSphere Message Broker products. Therefore, removing WebSphere MQ Integrator V2.1 and its prerequisite software products after the migration is optional.

Before uninstalling any previous versions of products, ensure that the migration is successful and has been completely verified. Refer to the related product documentation for details about how to uninstall the products.

Part 3

Migrating from WebSphere BI Message Broker V5.0

Part 3 book covers the migration of all components of WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0. It illustrates the coexistence of Message Broker V5.0 with Message Broker V6.0, and provides details about how to migrate the components to WebSphere Message Broker V6.0.

It also covers the migration steps specific to Message Broker V5.0, using the business scenario established for the purposes of the redbook.

17

WebSphere Business Integration Message Broker Version 5.0 environment setup

This chapter describes the environment used for the migration from WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0. It covers the environment prior to the migration, as well as the logical and physical topology of the environment to be migrated.

It also discusses the logical and physical topology of the environment and the changes to the environment following the migration. The differences between Message Broker V5.0 and Message Broker V6.0 are explored from the migration perspective.

17.1 Assumptions made during migration

- The installation and configuration of the WebSphere Business Integration Message Broker V5.0 environment was completed according to the installation guide.
- The installation of WebSphere Message Broker V6.0 components has been completed according to the installation guide that comes with the product. You can find this guide on the Web:

  http://www.elink.ibmlink.ibm.com/public/applications/publications/cgibin/pbi.cgi?CTY=GB&%2O%2OCTY=US&FNC=SRX&PBL=GC34-6621-O0&
- WebSphere MQ V5.3 was used on all platforms.

  However, the migration to WebSphere MQ V6.0 is not covered in this publication. It can be completed after WebSphere Message Broker migration, as a separate step.
- DB2 Universal Database V8.1.7 was used both before and after the migration.
- The latest available fix pack versions were used for WebSphere Business Integration Message Broker V5.0, WebSphere Message Broker V6.0, and WebSphere MQ V5.3.
- The decision was made to not migrate one AIX broker.

  This demonstrates the coexistence with a previous version of the broker, and that a Message Broker V6.0 Configuration Manager can control a heterogeneous domain of brokers.

17.2 The environment before migration

In this section we describe the WebSphere Business Integration Message Broker V5.0 environment, focusing on the following topics:

- The logical topology before migration
- The components that are planned to be migrated
- The physical topology before migration

17.2.1 The logical topology

Figure 17-1 on page 315 shows the logical topology of the Message Broker V5.0 environment before the migration:

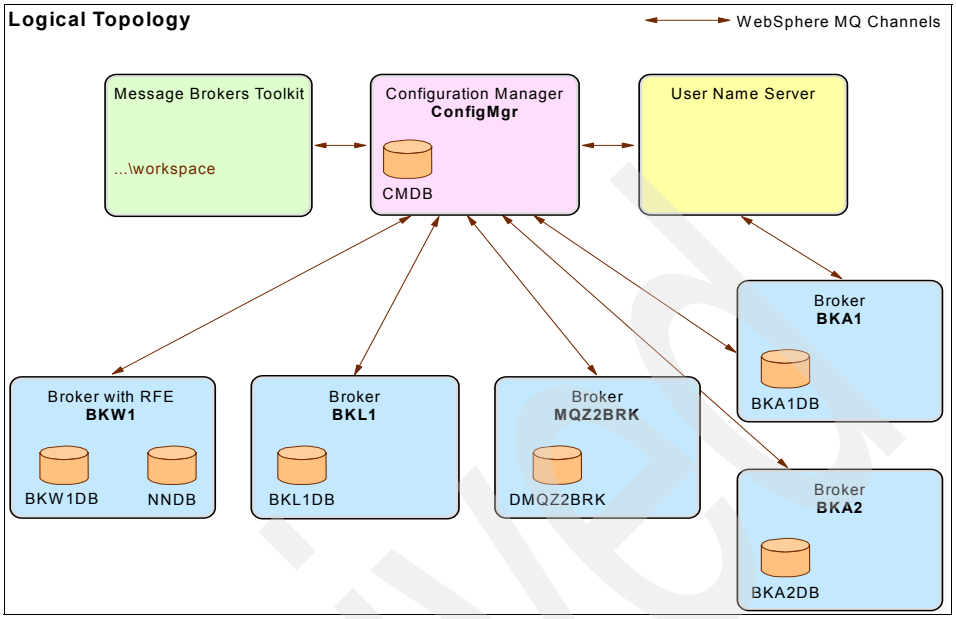


Figure 17-1 The logical topology of Message Broker V5.0 scenario before the migration

The environment consists of one broker domain, with a Configuration Manager, a Message Brokers Toolkit, a User Name Server and five brokers. All components and their required resources are listed in Table 17-1.

Table 17-1 The domain components before the migration, V5.0 scenario

Component type	Component name	Queue manager	Database
Message Brokers Toolkit	N/A	N/A	N/A
Configuration Manager	ConfigMgr	CMQM	CMDB
User Name Server	UserNameServer	UNSQM	N/A
Broker with RFE	BKW1	BKW1QM	BKW1DB, NNDB
Broker	BKA1	BKA1QM	BKA1DB[a]
Broker	BKA2	BKA2QM	BKA2DB[a]
Broker	MQZ2BRK	MQZ2	DMQZ2BRK
Broker	BKL1	BKL1QM	BKL1DB

a. TCP/IP loopback communication was used instead of shared memory to enable more than 10 sessions from the broker into the database in the same time.

Note: Each component uses its own queue manager in this scenario even if more than one component is on a single machine.

(There are user databases for message flow interactions on each broker as well, however, this is not documented for reasons of simplicity in the table and figures.)

For the message flows used in this scenario, refer to Appendix A, “Message flows for WebSphere Business Integration Message Broker V5.0 scenario”.

Components to be migrated

In our scenario, all components migrate except broker BKA2. Broker BKA2 remains at Message Broker V5.0 to demonstrate the coexistence of the two versions of the products in an environment.

17.2.2 The physical topology

Figure 17-2 shows the physical topology of the Message Broker V5.0 environment before the migration.

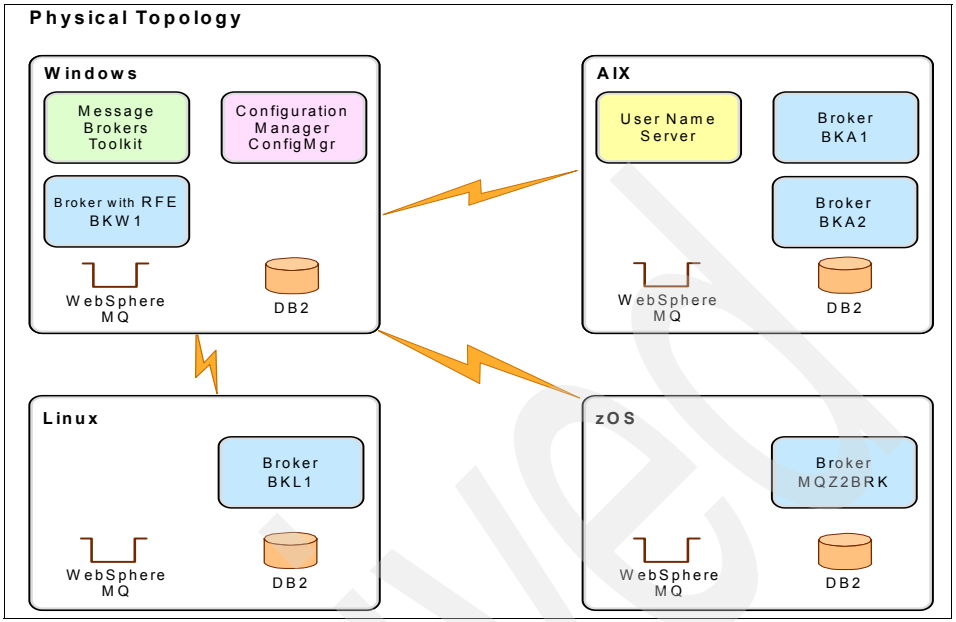


Figure 17-2 Physical topology of Message Broker V5.0 scenario before migration

The physical environment consists of four machines, which run Windows, AIX, z/OS, and Linux operating systems. Note the following:

- There is a Message Brokers Toolkit, a Configuration Manager and one broker on the Windows machine.
- There is a User Name Server and two brokers on the AIX machine.
- There is one broker on the z/OS machine.
- There is one broker on the Linux machine.
- Network communication is not needed between the AIX machine and the z/OS machine, or between the Linux machine and the z/OS machine, because there is no interaction between the components on these machines.

All software products used and the versions before the migration are listed in the following tables.

Important: Refer to table footnotes 1 to 7 for information regarding migration level.

Table 17-2 shows the Windows machine before the migration in the V5.0 scenario.

Table 17-2 Windows machine before migration - V5.0 scenario

Software	Installed level	Minimum level for migration
Operating system	Windows 2000 Service Pack 4[1]	Windows Server 2003 or Windows XP Professional[2]
WebSphere Business Integration Message Broker with Rules and Formatter Extension	V5.0 fix pack 6	V5.0 fix pack 4[3]
WebSphere MQ	V5.3 fix pack 11	V5.3 fix pack 1[4]
DB2	V8.1 fix pack 7	V8.2[5,6]
RAC	V5.2.0.1	V5.2.0.1
JRE	V1.4.2 (only for RAC)	V1.4.1 (only for RAC)

1. Windows 2000 is no longer a supported environment with WebSphere Message Broker V6.0.
2. Windows XP Professional is supported for development and test purposes only
3. Fix pack 6 is required for the Message Broker V5.0 broker to coexist in a Message Broker V6.0 domain.
4. WebSphere MQ Version 5.3 fix pack 10 is required for Real-time support and JMS MQ Optimized node. If you develop message flows that use WebSphere MQ Real-time Transport with Multicase PGM support, you must install WebSphere MQ Version 6.0 or later on each affected broker system. For more information, refer to the WebSphere Message Broker Information Center under ***Reference*** → ***Installation*** → ***Software requirements*** → ***Additional required products*** available on the Web:
   `http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah09088_.htm`
5. DB2 Universal Database V8.2 is equivalent to DB2 Universal Database V8.1 with fix pack 7.
6. XA coordination for messages in the MRM domain on Windows requires DB2 Version 8.2 fix pack 10 or later.

Table 17-3 shows the AIX machine before migration in the V5.0 scenario.

Table 17-3 AIX machine before migration - V5.0 scenario

Software	Installed level	Minimum level for migration
Operating system	V5.3 Maintenance Level 03	V5.3 Maintenance Level 02
WebSphere Business Integration Message Broker	V5.0 fix pack 6	V5.0 fix pack 4[3]
WebSphere MQ	V5.3 fix pack 11	V5.3 fix pack 1[4]
DB2	V8.1 fix pack 7	V8.2[5]
RAC	V5.2.0.1	V5.2.0.1
JRE	V1.4.2 (1.4.2.10 only for RAC)	V1.4.1 (only for RAC)

3. Fix pack 6 is required for the Message Broker V5.0 broker to coexist in a Message Broker V6.0 domain.
4. WebSphere MQ Version 5.3 fix pack 10 is required for Real-time support and JMS MQ Optimized node. If you develop message flows that use WebSphere MQ Real-time Transport with Multicase PGM support, you must install WebSphere MQ Version 6.0 or later on each affected broker system. For more information, refer to the WebSphere Message Broker Information Center under ***Reference*** → ***Installation*** → ***Software requirements*** → ***Additional required products*** available on the Web:
   `http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah09088_.htm`
5. DB2 Universal Database V8.2 is equivalent to DB2 Universal Database V8.1 with fix pack 7.

Table 17-4 shows the z/OS machine before migration in the V5.0 scenario.

Table 17-4 z/OS machine before migration - V5.0 scenario

Software	Installed level	Minimum level for migration
Operating system	V1R6 RSU0508	V1R5 at RSU0507 plus PTF for OA11699
WebSphere Business Integration Message Broker	V5.0 RSU0508	V5.0 fix pack 4[3]
WebSphere MQ	V5.3.1 RSU0508	V5.3.1 plus PTF for PQ80677

Software	Installed level	Minimum level for migration
DB2	V8.1 RSU0508	V8.1 plus PTF for PQ84976
RAC	V5.2.0.1	V5.2.0.1
JRE	V1.4.2 SR3	V1.4.2 SR2

3. Fix pack 6 is required for the Message Broker V5.0 broker to coexist in a Message Broker V6.0 domain.

Table 17-5 shows the Linux machine before migration in the V5.0 scenario.

Table 17-5 Linux machine before migration - V5.0 scenario

Software	Installed level	Minimum level for migration
Operating system	Intel IA32 Red Hat Enterprise AS V3.2.3-47	Intel IA32 Red Hat Enterprise AS V3.0 plus Update 2
WebSphere Business Integration Message Broker	V5.0 fix pack 6	V5.0 fix pack 4[3]
WebSphere MQ	V5.3 fix pack 11	V5.3 fix pack 1[4,7]
DB2	V8.1 fix pack 7	V8.2[4]
RAC	V5.2.0.1	V5.2.0.1
JRE	V1.4.2 (only for RAC)	V1.4.1 (only for RAC)

3. Fix pack 6 is required for the Message Broker V5.0 broker to coexist in a Message Broker V6.0 domain.
4. WebSphere MQ Version 5.3 fix pack 10 is required for Real-time support and JMS MQ Optimized node. If you develop message flows that use WebSphere MQ Real-time Transport with Multicase PGM support, you must install WebSphere MQ Version 6.0 or later on each affected broker system. For more information, refer to the WebSphere Message Broker Information Center under ***Reference*** → ***Installation*** → ***Software requirements*** → ***Additional required products*** available on the Web:

```
http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=
/com.ibm.etools.mft.doc/ah09088_.htm
```

7. To exploit the Native POSIX Threading Library (NPTL) performance gains on Linux systems (available with Kernel 2.6 and above), which support multiple threads running within one process, you must install WebSphere MQ Version 6.0. Performance gains might be achieved where you have multiple message flow assigned to a single execution group.

Products to be migrated

The only component selected for the migration in this scenario is WebSphere Business Integration Message Broker V5.0. All other components, including the operating system, remain at the same level. The Configuration Manager will be moved from the Windows machine to the AIX machine.

An additional Message Brokers Toolkit will be used on the Linux machine to demonstrate the availability of the component on a new platform.

17.3 The environment after migration

In this section we describe the WebSphere Message Broker environment after the migration, focusing on the following topics:

- The logical topology after migration
- The physical topology after migration
- The changes after migration

17.3.1 The logical topology

Figure 17-3 shows the logical topology of the Message Broker V6.0 environment after the migration process.

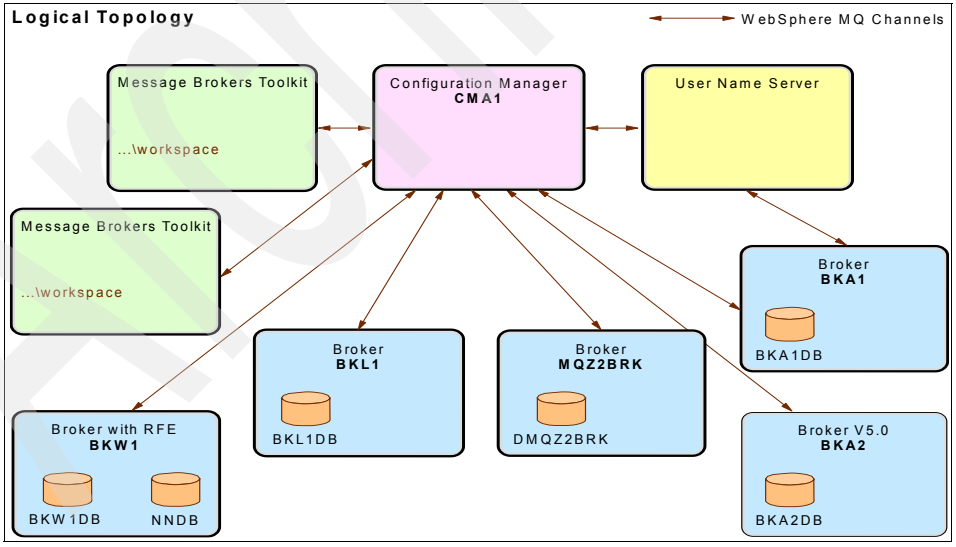


Figure 17-3 Logical topology of Message Broker V5.0 scenario after migration

Note that the environment after migration still contains a single broker domain. All components and their resources after the migration are listed in Table 17-6 on page 322.

Table 17-6 Domain components after migration - V5.0 scenario

Component type	Component name	Queue manager	Database
Message Brokers Toolkit	N/A	N/A	N/A
Configuration Manager	CMA1	CMQM	N/A
User Name Server	UserNameServer	UNSQM	N/A
Broker with RFE	BKW1	BKW1QM	BKW1DB, NNDB
Broker	BKA1	BKA1QM	BKA1DB
Broker at V5.0 level	BKA2	BKA2QM	BKA2DB
Broker	MQZ2BRK	MQZ2	DMQZ2BRK
Broker	BKL1	BKL1QM	BKL1DB

The changes after the migration

The changes, as illustrated by Figure 17-3 on page 321 and Table 17-6 on page 322, involve the following:

- The DB2 Universal Database for the Configuration Manager was removed because it is no longer needed for Message Broker V6.0, which uses an internal repository.
- Broker BKA2 remains at Message Broker V5.0 to demonstrate the coexistence of two versions of the products in a heterogeneous domain.

17.3.2 The physical topology

Figure 17-4 shows the physical topology of the Message Broker V6.0 environment after the migration process.

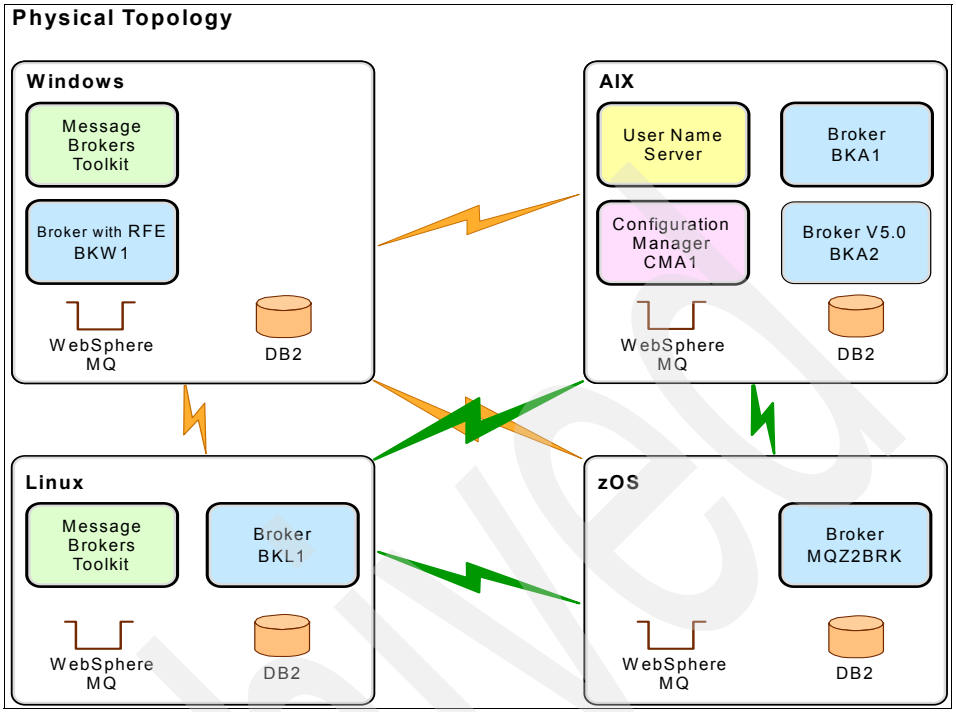


Figure 17-4 Physical topology of Message Broker V5.0 scenario after migration

The physical environment after the migration still contains four machines, which run Windows, AIX, z/OS, and Linux operating systems.

All software products used and the versions after the migration are listed in the following tables.

Table 17-7 Windows machine after migration - V5.0 scenario

Software	Installed level
Operating system	Windows 2000 Service Pack 4[a]
WebSphere Message Broker with Rules and Formatter Extension	V6.0.0.0
WebSphere MQ	V5.3 fix pack 11
DB2	V8.1 fix pack 7 (which is equivalent to V8.2)
RAC	V5.2.0.1
JRE	V1.4.2 (only for RAC)

a. Windows 2000 is no longer a supported environment with WebSphere Message Broker V6.0. Following a migration to WebSphere Message Broker V6.0, the operating system should be upgraded to a supported version.
Details of this upgrade are beyond the scope of this publication; for information about supported environments for WebSphere Message Broker V6.0, refer to:
http://www.ibm.com/software/integration/wbimessagebroker/requirements/index.html

Table 17-8 shows the AIX machine after migration in the V5.0 scenario.

Table 17-8 AIX machine after migration - V5.0 scenario

Software	Installed level
Operating system	V5.2 Maintenance Level 07
WebSphere Business Integration Message Broker	V5.0 fix pack 6
WebSphere Message Broker	V6.0.0.0
WebSphere MQ	V5.3 fix pack 11
DB2	V8.1 fix pack 7 (which is equivalent to V8.2)
RAC	V5.2.0.1
JRE	V1.4.2 (1.4.2.10, only for RAC)

Table 17-9 shows the z/OS machine after migration in the V5.0 scenario.

Table 17-9 z/OS machine after migration - V5.0 scenario

Software	Installed level
Operating system	V1R6 RSU0508
WebSphere Message Broker	V6.0 RSU0508
WebSphere MQ	V5.3.1 RSU0508
DB2	V8.1 RSU0508
RAC	V5.2.0.1
JRE	V1.4.2 SR3
XML Toolkit	V1.8

Table 17-10 shows the Linux machine after migration in the V5.0 scenario.

Table 17-10 Linux machine after migration - V5.0 scenario

Software	Installed level
Operating system	Intel IA32 Red Hat Enterprise AS V3.2.3-47
WebSphere Message Broker	V6.0.0.0
WebSphere MQ	V5.3 fix pack 11
DB2	V8.1 fix pack 7 (which is equivalent to V8.2)
RAC	V5.2.0.1
JRE	V1.4.2 (only for RAC)

The changes after the migration

The changes, as illustrated by Figure 17-4 on page 323 and Table 17-7 through Table 17-10, involve the following:

- WebSphere Message Broker V6.0 was installed on all four machines.
- The XML Toolkit V1.8 was installed on the z/OS machine.
- The Configuration Manager was moved from the Windows machine to the AIX machine, and the WebSphere MQ environment was configured with new channels and transmission queues as appropriate.
- An additional Message Brokers Toolkit was installed on the Linux machine to demonstrate the availability of the component on a new platform.

17.4 Differences between Message Broker V5.0 and Message Broker V6.0

This section discusses the main differences between Message Broker V5.0 and Message Broker V6.0, from the migration perspective.

Note: You can find more detailed information about the version differences in Chapter 4, “New features in WebSphere Business Integration Message Broker V5.0” on page 55 and on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp`

The major differences include the following:

- There is a new version of Message Brokers Toolkit, built on the Rational Application Developer Platform.
- The Message Brokers Toolkit can now be installed on Linux (x86 platform).
- There are changes in console commands, including the availability of new commands and of new options for existing commands. For more information, visit the WebSphere Message Broker 6.0 Information Center, under ***Reference*** → ***Operations*** → ***Commands***, available on the Web:

  `http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an07060_.htm`

  Tip: The commands **`mqsistartmsgflow`** and **`mqsistopmsgflow`** can be used for starting and stopping message flows, as an alternative to using the Message Brokers Toolkit.

- The Configuration Manager now uses its own internal repository, so the DB2 Universal Database is no longer needed.

  Tip: The command **`mqsibackupconfigmgr`** can be used to back up a Configuration Manager.

- The Configuration Manager can now be created on all supported platforms: AIX, HP-UX, Linux (x86 platform), Linux (zSeries platform), Solaris, Windows, and z/OS.
- The Configuration Manager can be administered by the Configuration Manager Proxy (CMP) API. This is a Java application programming interface for the Configuration Manager. It enables programs to be written to administer broker domains, as an alternative to using the Message Brokers Toolkit.

  Tip: A Configuration Manager Proxy API Exerciser sample application is supplied to demonstrate the capabilities of the Configuration Manager Proxy (CMP) API in a graphical use interface.

- Multiple named Configuration Managers can be created on the same system in Message Broker V6.0.

  Note: A broker can still only be managed by a single Configuration Manager, but multiple domains can be managed from a single system.

18

WebSphere Business Integration Message Broker V5.0 migration steps

This chapter discusses the high level steps needed to migrate a WebSphere Business Integration Message Broker V5.0 domain to WebSphere Message Broker V6.0.

The chapter covers the following scenarios:

- Migrating a single domain
- Migrating multiple domains (development, test, and production)

18.1 Pre-migration planning

Before beginning the migration, read Chapter 5, “Migration considerations” on page 75 and Chapter 6, “Planning for migration” on page 87 to ensure that all pre-migration tasks have been completed, including the following:

- Minimum software levels for the current WebSphere Business Integration Message Broker V5.0 are correctly configured.
- WebSphere Message Broker V6.0 pre-requisite software and hardware levels are correct.
- New migration features, such as coexistence, are understood.

This planning will help ensure that your WebSphere Business Integration Message Broker V5.0 domain is correctly configured for migration.

18.2 Migrating a single domain

All WebSphere Business Integration Message Broker V5.0 components can interact with components running at WebSphere Message Broker V6.0. This means that the order of migration is flexible.

The order as presented in the following sections is the approach taken in our migration scenario. The migration of the Message Brokers Toolkit, Configuration Manager, and User Name Server all occur before the migration of the broker. This chronological approach ensures that small changes are made to the environment at each step. Problems you encounter can thus be rectified before the migration continues.

The following steps illustrate the migration of a single WebSphere Business Integration Message Broker V5.0 domain to WebSphere Message Broker V6.0. For information about migrating multiple domains (for example, a development, test, and production environment) see 18.3, “Migrating multiple domains” on page 330.

For each step in the following section we provide a high level overview of what is required to migrate a WebSphere Business Integration Message Broker V5.0 domain to WebSphere Message Broker V6.0. The steps also contain a link to the appropriate chapter for details about performing the step.

18.2.1 Back up the domain

Before migrating any components, you need to back up the current domain. Refer to Chapter 19, “Backing up the WebSphere Business Integration Message Broker V5.0 domain” on page 339 for details of this task.

Important: Performing a backup of the domain allows you to restore your current WebSphere Business Integration Message Broker V5.0 environment if you encounter problems during migration.

18.2.2 Migrate the Message Brokers Toolkit and Configuration Manager

After successfully backing up the WebSphere Business Integration Message Broker V5.0 domain, you can begin the migration process by migrating the Message Brokers Toolkit and the Configuration Manager.

1. Message Brokers Toolkit and resources

   The first stage in the migration process is to migrate the Message Brokers Toolkit and any resources, such as message flows, message sets, and user-defined nodes. Chapter 20, “Migrating WebSphere Message Broker V5.0 and V5.1 tools and resources” on page 353 discusses the steps needed for this migration.

2. Configuration Manager

   After successfully migrating the Message Brokers Toolkit, message flows and message sets, migrate the Configuration Manager. Chapter 21, “Migrating the WebSphere Business Integration Message Broker V5.0 Configuration Manager” on page 387 discusses the steps needed for this migration.

18.2.3 Migrate the User Name Server

Migrate the User Name Server before migrating any broker. Chapter 22, “Migrating the User Name Server” on page 427 discusses the steps needed for this migration.

18.2.4 Migrate the brokers

You can migrate the brokers after all other WebSphere Business Integration Message Broker V5.0 components have been migrated. The steps, which cover how to migrate a WebSphere Business Integration Message Broker V5.0 broker

to WebSphere Message Broker V6.0 on the stated platform, are discussed in the following chapters:

- Chapter 23, “Migrating a Windows broker” on page 437
- Chapter 24, “Migrating an AIX broker” on page 447
- Chapter 25, “Migrating the z/OS broker” on page 461
- Chapter 26, “Migrating the Linux broker” on page 481

The chapters also cover the migration of New Era Of Networks components.

18.2.5 Ensure a successful migration

After a broker has been migrated to WebSphere Message Broker V6.0, follow the steps discussed in Chapter 27, “Verifying migration to WebSphere Message Broker V6.0” on page 495 to ensure that the migration has been successful.

18.2.6 Single domain migration checklist

Table 18-1 summarizes the migration steps for a single domain. Follow these steps in the order presented.

Table 18-1 Checklist for migrating a single broker domain

Step	Completed
Read the considerations and planning chapters.	
Prepare the domain for migration.	
Back up the current domain.	
Migrate the Message Brokers Toolkit and resources.	
Migrate the Configuration Manager.	
Migrate the User Name Server (if present).	
Take a current backup of the brokers, so that any changes are backed up.	
Migrate the brokers.	
Check for successful migration.	

18.3 Migrating multiple domains

In 18.2, “Migrating a single domain” on page 328, we provide a high level overview of migrating a single broker domain. In a given organization, however,

there may be multiple domains that require migration. Here we discuss the migration of three domains: development, test, and production. Before you start migrating multiple domains, read 18.1, “Pre-migration planning” on page 328.

Note that, during the migration to WebSphere Message Broker V6.0, you may be required to keep the production brokers operational. Therefore the development and test domains can run WebSphere Business Integration Message Broker V5.0 and WebSphere Message Broker V6.0 in parallel. This strategy allows critical WebSphere Business Integration Message Broker V5.0 changes to be made, tested, and deployed to production while the migration is in progress.

We recommend that you freeze the application code during migration. This reduces the number of steps required to migrate to WebSphere Message Broker V6.0. However, it is not always possible to freeze code changes for this length of time.

So if changes *must* be made to WebSphere Business Integration Message Broker V5.0 code during migration, then they must also be migrated to WebSphere Message Broker V6.0. This can be done on a daily or weekly basis as the migration is in progress.

18.3.1 Coexistence in the development domain

In the development domain, message flow developers create message flows and unit test them in a “sandbox” environment. The brokers in the domain are not responsible for handling business-critical data. Here, WebSphere Message Broker V6.0 should be installed alongside WebSphere Business Integration Message Broker V5.0, and both should be run in parallel until the production domain has been successfully migrated. This allows developers to run both versions of the tooling on their machines, and to migrate their message flows and message sets to WebSphere Message Broker V6.0.

Important: Use care when having brokers coexist on a single machine. If applications running in the broker are not designed to share resources (such as database tables), they may encounter problems when running in parallel.

New WebSphere Message Broker V6.0 brokers and a new Configuration Manager should be created alongside their WebSphere Business Integration Message Broker V5.0 counterparts. These new components should form a new broker domain and should *not* interoperate with the existing WebSphere Business Integration Message Broker V5.0 domain.

Developers should use this new WebSphere Message Broker V6.0 environment to unit test their migrated message flows and message sets. After the resources have been successfully migrated and unit tested, deploy them to the test domain.

Important: New WebSphere Message Broker V6.0 functionality (such as new nodes in message flows) should *not* be used until all production domain brokers have been fully migrated.

18.3.2 Coexistence in the test domain

In the test domain, message flows from the development domain are tested against test data in a realistic broker configuration.

As with the development domain, the test domain should exploit the coexistence feature of WebSphere Message Broker V6.0 to allow multiple brokers and Configuration Managers on a single computer.

The WebSphere Business Integration Message Broker V5.0 brokers and the WebSphere Message Broker V6.0 brokers should be kept in separate domains and administered from different Configuration Managers. This parallel approach allows the WebSphere Business Integration Message Broker V5.0 components to be easily uninstalled once WebSphere Message Broker V6.0 has been successfully configured in the production environment.

After the migrated application has been successfully tested, the migration of the production environment can begin.

18.3.3 Migrate the production domain

When message flows in the test domain are deemed to be robust enough, they are promoted to the production domain. This is the domain responsible for performing actual business transactions and message flows in the domain work with live data.

At this point, the development and test environments have verified that the migrated applications work correctly on the new WebSphere Message Broker V6.0 brokers. Therefore, a staged migration of the production domain can begin.

Message Brokers Toolkit and Configuration Manager

The first task in migrating the production domain is to migrate both the Message Brokers Toolkit and Configuration Manager to WebSphere Message Broker V6.0.

Before beginning this migration, back up your Configuration Manager and Message Brokers Toolkit resources, as described in Chapter 19, “Backing up the WebSphere Business Integration Message Broker V5.0 domain” on page 339.

User Name Server

After the Control Center and Configuration Manager have been successfully migrated, you must migrate the User Name Server.

(If the User Name Server is not used in the broker domain, ignore this task.)

Brokers

Once the production domain brokers are being controlled by the WebSphere Message Broker V6.0 Configuration Manager, you can migrate the brokers. These should be migrated one at a time to ensure that existing WebSphere Business Integration Message Broker V5.0 brokers are still running to process business transactions.

Note: Brokers that share a database schema must be migrated *simultaneously*.

After the first broker has been successfully migrated, and has run for a period of time without error, the next broker can be migrated. Migrate the brokers one at a time until all brokers in the production domain are running at the WebSphere Message Broker V6.0 level.

The message flows and message sets that are running in the WebSphere Business Integration Message Broker V5.0 broker continue to run after the broker has been migrated. For this reason, a new deployment of the migrated resources is not required immediately.

Deploy migrated resources

After all brokers have been successfully migrated and have been running for a period of time, you can deploy the migrated message flows and message sets to them. (This task is not required if no functional changes have occurred in the applications.)

Use the Message Brokers Toolkit to deploy the migrated resources to a single broker. After the migrated applications have been running successfully for a period of time, deploy the migrated applications to the remaining brokers one at a time.

At this point, developers can use new WebSphere Message Broker V6.0 functionality in their message flows because all production brokers are running at WebSphere Message Broker V6.0 and are therefore able to support it.

Important: Any change to applications, including the use of new WebSphere Message Broker V6.0 nodes, should be tested in the development and test domains before deployment to production.

Uninstall WebSphere Business Integration Message Broker V5.0

After the production domain has been running at WebSphere Message Broker V6.0 for a period of time, the WebSphere Business Integration Message Broker V5.0 components in the development and test environments can be uninstalled.

18.3.4 Multiple domain migration checklists

Table 18-2 through Table 18-4 present checklists of the steps required to migrate development, test, and production environments. Follow these steps in the order presented.

Table 18-2 shows the checklist for migrating a development broker domain.

Table 18-2 Migrating a development broker domain - checklist

Development steps	Domain	Completed
Read the considerations and planning chapters.	Test/Dev/Prod	
Install V6 Message Brokers Toolkit alongside V5 Message Brokers Toolkit, exploiting coexistence.	Dev	
Migrate the V5 resources to V6.	Dev	
Install V6 Configuration Manager and brokers along side V5 counterparts.	Dev	
Create V6 brokers and Configuration Managers.	Dev	
Connect the Message Brokers Toolkit to the V6 Configuration Manager.	Dev	
Add the V6 brokers to a new domain and deploy the migrated resources to test them.	Dev	

Table 18-3 shows the checklist for migrating the test broker domain.

Table 18-3 Migrating a test broker domain - checklist

Test steps	Domain	Completed
Install V6 Message Brokers Toolkit alongside V5 Message Brokers Toolkit, exploiting coexistence.	Test	
Import the migrated V6 resources into the V6 Message Brokers Toolkit.	Test	
Install V6 Configuration Manager and brokers alongside V5 counterparts.	Test	
Create V6 brokers and Configuration Managers.	Test	
Connect the V6 Message Brokers Toolkit to the V6 Configuration Manager.	Test	
Add the V6 brokers to a new domain and deploy the migrated resources to them.	Test	

Table 18-4 shows the checklist for migrating a production broker domain.

Table 18-4 Migrating a production broker domain - checklist

Production steps	Domain	Completed
Install V6 on the Configuration Manager and Message Brokers Toolkit machines.	Prod	
Back up the Configuration Manager.	Prod	
Migrate the Configuration Manager and Message Brokers Toolkit to V6.	Prod	
Import the migrated resources into the Message Brokers Toolkit.	Prod	
Migrate the User Name Server to WebSphere Message Broker V6.0	Prod	
Back up your brokers.	Prod	
Migrate a single broker to WebSphere Message Broker V6.0, leaving the others running.	Prod	
Leave the broker running for a period of time to ensure success of the migration.	Prod	
When successful, back up and migrate the remaining brokers, one at a time.	Prod	

Production steps	Domain	Completed
After all brokers in the domain are stable, connect to the Configuration Manager and deploy the migrated resources to a single broker.	Prod	
Run the migrated resources in the broker for a period of time to ensure they run successfully.	Prod	
Deploy the migrated resources to the remaining brokers, one at a time.	Prod	
After all brokers are successfully migrated, remove the V5 components from the development and test environments.	Dev/Test	
After all brokers are running at WebSphere Message Broker V6.0, new product features can be used.	Dev/Test/Prod	

If you encounter any problems during this migration, you can use the backups to restore components back to their WebSphere Business Integration Message Broker V5.0 level. For more information, see Chapter 27, “Verifying migration to WebSphere Message Broker V6.0” on page 495.

18.4 Best practices for a successful migration

Following these best practices will help you to reduce the possibility of encountering a problem or suffering an outage during migration.

Use redundant brokers for high availability

If the domain to be migrated contains a single broker, then the migration process impacts the availability of the broker, because it must be stopped. Therefore, an ideal broker domain has a minimum of three brokers processing workload. This allows one broker to be migrated while the remaining two brokers provide failover for each other.

Test the migration

By exploiting new features of WebSphere Message Broker V6.0, you can test the migration of a component by using the following command:

```
mqsimigratecomponents -c
```

This ensures that the component migrates successfully when the actual command is run. The use of this command is discussed in the broker migration chapters.

After migrating a component, let it run at the new WebSphere Message Broker V6.0 level for a period of time. That way, any problems you encounter can be resolved before you migrate the next component.

Use a stepwise approach

Performing migration steps in parallel is highly discouraged because, if a problem occurs, it is extremely difficult to know which new component caused the problem.

Therefore, the process described in this chapter emphasizes a chronological approach—that is, migrate one component after another. Also, old WebSphere Business Integration Message Broker V5.0 components should be removed only after the new WebSphere Message Broker V6.0 components are functioning correctly.

19

Backing up the WebSphere Business Integration Message Broker V5.0 domain

This chapter describes basic backup strategies for the WebSphere Business Integration Message Broker V5.0 domain, including the Message Brokers Toolkit workspace and databases. Backup should be successfully completed before you begin the migration to WebSphere Message Broker V6.0.

The purpose of backup is to ensure business continuity, high availability and planning disaster recovery, and in this chapter we illustrate a simple example. However, in your installation there will be many environment-specific backup requirements you will need to keep in mind.

The major points covered here provide an insight into backing up your environment successfully.

19.1 Backup overview

Following are the recommended backup tasks to be completed before migrating to WebSphere Message Broker V6.0.

- Back up databases:
  - Broker database
  - Configuration Manager database
  - New Era Of Networks database
- Back up Universally Unique ID (UUID)
- Back up Message Brokers Toolkit:
  - Workspace
  - User-defined nodes/parsers

19.2 Backing up components that use the DB2 Universal Database

In this section we only deal with the steps for backing up WebSphere Business Integration Message Broker components that use the DB2 Universal Database. Backup information regarding other databases is beyond the scope of this publication.

Perform DB2 backup of Configuration Manager, New Era Of Networks and the Broker databases.

1. Choose a safe storage location, like an external disk drive or DVD, to store the backup information:
   - For z/OS, create a sequential non-VSAM data set.
   - For Windows or UNIX, create a directory or file system.
2. As illustrated in Figure 19-1 on page 341, check for applications with active database connections by entering the following command in the DB2 command window:

```
DB2 list applications global
```

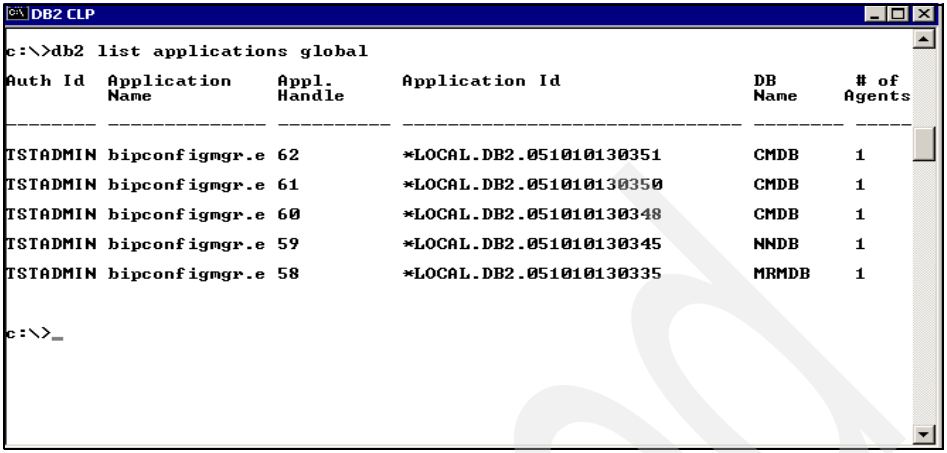


Figure 19-1 Check active DB2 applications

3. Stop all WebSphere Business Integration Message Broker components on the machine where you are performing the backup by typing the **mqsistop** *broker_name* command on Windows or UNIX and by issuing the **/P** *broker_name* command from SDSF on z/OS. For example, type:

   ```
   mqsistop BKW1
   ```

   Stop all WebSphere Business Integration Message Broker components on the machine where you are performing the backup.

   – On Windows or UNIX, type:

     ```
     mqsistop BKW1
     ```

   – On z/OS, from SDSF, type:

     ```
     /P UserNameServer
     ```

   To stop the Configuration Manager, on Windows type:

   ```
   mqsistop ConfigMgr
   ```

   Ensure that the components are stopped by checking the operating system application user/error log.

   Important: After stopping all domain components, back up each component in stages. Restart the components only after they all have been backed up. This avoids inconsistency between broker backups and Configuration Manager backups.

4. As illustrated in Figure 19-2, check whether all applications are stopped by typing the following command in the DB2 command window:

```
DB2 list applications global
```

If some applications are still running, type the following command in the DB2 command window:

```
DB2 force application all
```

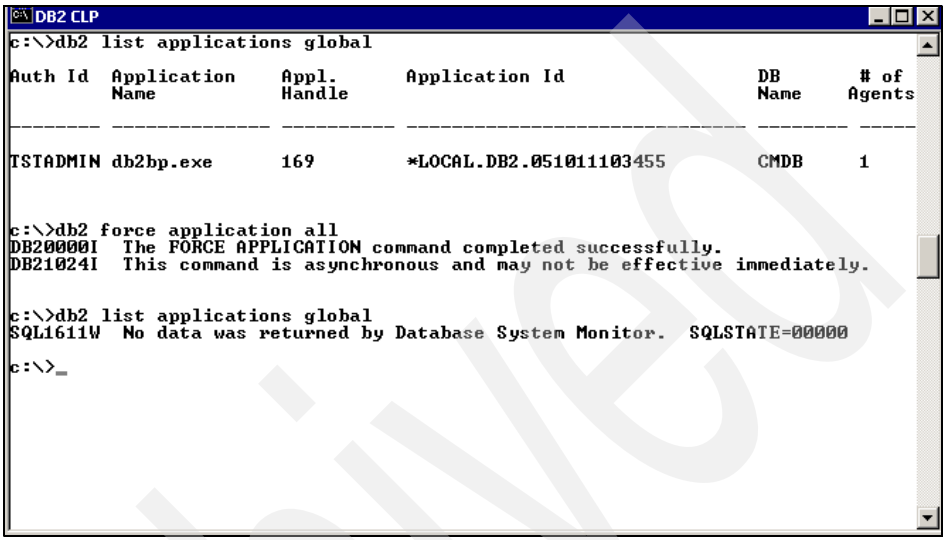


Figure 19-2 Force DB2 applications

5. Back up the Configuration Manager database on a Windows machine by using the DB2 command window:

   ```
   DB2 connect to <database_name>
   ```

   ```
   DB2 quiesce database immediate force connections
   ```

   ```
   DB2 connect reset
   ```

   ```
   DB2 backup database <database_name> to <directory> with 2 buffers
   buffer 1024 parallelism 1 without prompting
   ```

   ```
   DB2 connect to <database_name>
   ```

   ```
   DB2 unquiesce database
   ```

   ```
   DB2 connect reset
   ```

Attention: Make a note of the time stamp displayed after successful completion of a backup, as illustrated in Figure 19-3.

The reason for this is as follows: The time stamp is unique for each backup, and is used as a parameter for the command to restore the database. If a value for the time stamp is not known when a restore is performed, then only one backup image must be available on the source media. So noting the time stamp ensures that the correct backup is used in a restore operation, if required at a later stage.

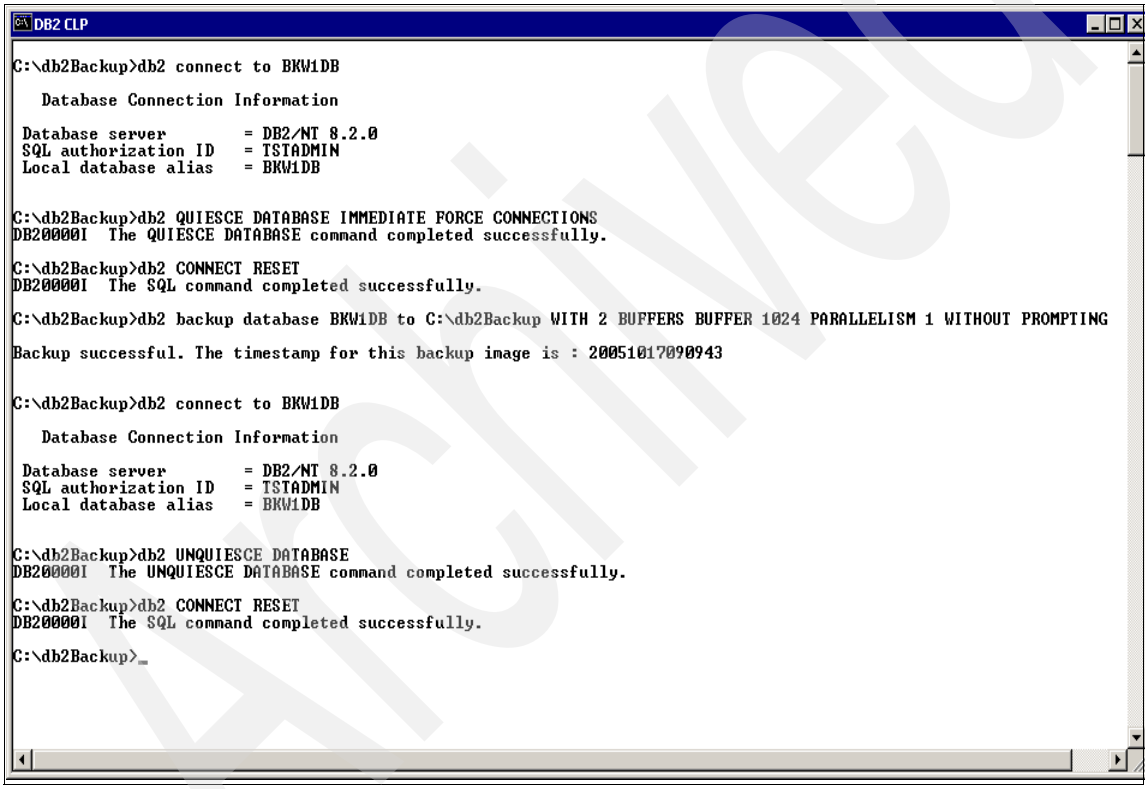


Figure 19-3 DB2 backup with time stamp

6. Back up the Broker and New Era Of Networks databases on Windows or UNIX machines using the same commands documented in step 5.

   On z/OS systems, use an Image Copy Utility (DB2BACKUP) job. Obtain the LRSN value for this backup using a DBREPORT job, as illustrated in the following examples.

The JCL samples can be downloaded from the Web; refer to Appendix C, “Additional material” on page 555 for instructions.

Example 19-1 shows sample DB2BACKUP JCL.

Example 19-1 DB2BACKUP JCL sample

```
/DBBACKUP JOB MSGLEVEL(1,1),MSGCLASS=T,NOTIFY=&SYSUID
//PROCS JCLLIB ORDER=(XXXXXX.PROCLIB)
/*JOBPARM SYSAFF=XXXX
//*
//UTIL EXEC DSNUPROC,SYSTEM=DBXX,UID='XXXXXXX',UTPROC=''
//*
//*********************************************
//*
//* JCL FOR THE IMAGE COPY UTILITY
//* DATE: 13/10/05
//* CREATED BY: DEANEJ
//* UPDATED BY: Mark Hiscock
//* COPY THIS MEMBER INTO YOUR BROKER PDSE AND
//* EDIT THIS JOB AND UPDATE THE FOLLOWING:
//*
//*
//* 1. The proclib for your DB2 (line 1)
//* 2. The system name for the job to run on (line 2)
//* 3. The DB2 sub system and user ID (line 4)
//* 4. The broker database name for the LISTDEF command
//* 5. The broker name for the DSN template (not required but consistent)
//*
//* THE JOBS SHOULD BE RUN IN THE FOLLOWING ORDER
//*
//* 1. BACKUP DB WITH DBBACKUP
//* 2. GET LRSN VALUE FOR FIC WITH DBREPORT
//* 3. RECOVER DB WITH DBRESTOR
//*
//*********************************************
//*
//*
//* INCLUDE INDEXSPACES DATABASE DMQXXBRK ALL
//* DOES NOT WORK AS OUR INDEX SPACES DO NOT
//* HAVE THE COPY YES ATTRIBUTE. SPECIFYING THIS
//* RESULTS IN MESSAGE DSNU425I.
//*
//*
//DSNUPROC.SYSIN DD *
LISTDEF COMPONENT INCLUDE TABLESPACES DATABASE DMQXXBRK ALL
TEMPLATE MQXXCOPY
  DSN('WMQI.&DB..&TS..D&DATE..T&TIME..IC')
  UNIT(SYSALLDA) VOLUMES(TOTDCN) SPACE(300,300) TRK
  DISP(NEW,CATLG,DELETE)
```

```
COPY LIST COMPONENT COPYDDN(MQXXCOPY)
  FULL YES
/*
```

Example 19-2 shows sample DBREPORT JCL.

Example 19-2 DBREPORT JCL sample

```
//DBREPORT JOB MSGLEVEL(1,1),MSGCLASS=T,NOTIFY=&SYSUID
//PROCS JCLLIB ORDER=(XXXXX.PROCLIB)
/*JOBPARM SYSAFF=XXXX
//*
//UTIL EXEC DSNUPROC,SYSTEM=XXXXX,UID='XXXXX',UTPROC=''
//*
//*******************************************
//*
//* JCL FOR THE REPORT UTILITY
//* DATE: 13/10/05
//* CREATED BY: DEANEJ
//* UPDATED BY: Mark Hiscock
//* COPY THIS MEMBER INTO YOUR BROKER PDSE AND
//* EDIT THIS JOB AND UPDATE THE FOLLOWING:
//*
//* 1. The proclib for your DB2 (line 1)
//* 2. The system name for the job to run on (line 2)
//* 3. The DB2 sub system and user ID (line 4)
//* 4. The broker database name for the LISTDEF command
//*
//* THE JOBS SHOULD BE RUN IN THE FOLLOWING ORDER
//*
//* 1. BACKUP DB WITH DBBACKUP
//* 2. GET LRSN VALUE FOR FIC WITH DBREPORT
//* 3. RECOVER DB WITH DBRESTOR
//*
//*******************************************
//*
//DSNUPROC.SYSIN DD *
LISTDEF COMPONENT INCLUDE TABLESPACES DATABASE DMQXXBRK ALL
QUIESCE LIST COMPONENT
//
/*
```

Example 19-3 shows DBREPORT JCL output.

Example 19-3 Sample output from DBREPORT JCL

```
-DB8W DSNUQUIA - QUIESCE AT RBA 000004290EC2 AND AT LRSN 000004290EC2
```

Tip: For New Era Of Networks components, we also recommend that you export the formats and rules using NNRie and NNFie utilities, though this is not mandatory.

19.3 Universally Unique ID backup

Each WebSphere Business Integration Message Broker component has its own unique ID. During the first deployment to a WebSphere Business Integration Message Broker, the Configuration Manager exchanges this ID with the broker to guarantee consistency.

We recommend that you back up the UUID (registry information) located on the following:

- For Unix machines

  In the /var/mqsi directory, issue the command **`tar cvf mqsiconfig.tar /var/mqsi`**. This creates a tar file called mqsiconfig.tar that contains the contents of the /var/mqsi directory structure.

- For z/OS machines

  Look at the value of the variable MQSI_REGISTRY in environment file ENVFILE of the broker service userID and issue the **`tar`** command. For example, if MQSI_REGISTRY is /var/wmqi, issue the command **`tar cvf mqsiconfig.tar /var/wmqi.`** This creates a tar file called mqsiconfig.tar that contains the contents of the /var/wmqi directory structure.

- For Windows machines

  Run the command **`regedit`** on a command line. Navigate to the BrokerUUID key by expanding **HKEY_LOCAL_MACHINE** → **SOFTWARE** → **IBM** → **WebSphereMQIntegrator** → **2** → ***<BrokerName>***

  <BrokerName> is the name of the broker.

  Select the BrokerUUID key as shown in Figure 19-4 on page 347. Right-click **BrokerUUID** and select **Modify**.

  Select the contents of “Value data:” from the Edit String dialog box and copy these contents into a text file, or make a note of them.

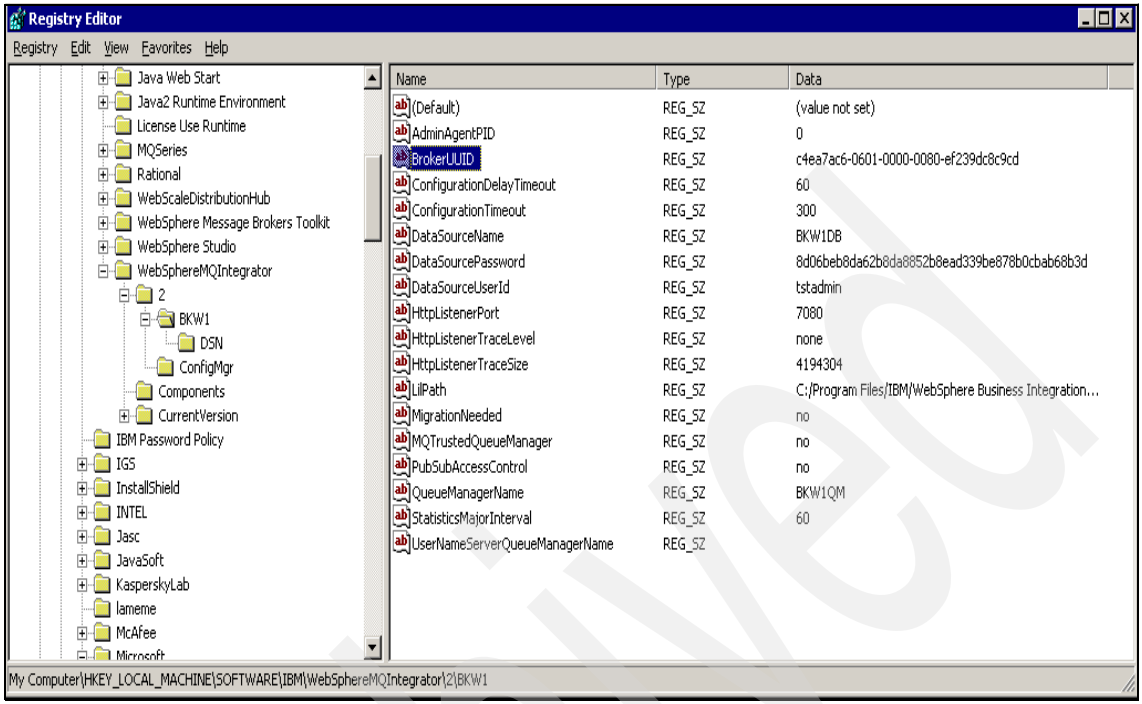


Figure 19-4 UUID key in the Registry Editor

19.4 Message Brokers Toolkit workspace backup

Now back up the Message Brokers Toolkit workspace resources. If users are using a code repository, then they must back up the repository. Otherwise, the Message Brokers Toolkit workspace holds all WebSphere Business Integration Message Broker resources (message flows, message sets, Plug In Node Projects, configuration for the broker connections and broker archive files).

Attention: To ensure that the external projects are also saved, we recommend that you back up the Message Brokers Toolkit workspace using the Message Brokers Toolkit.

1. As illustrated in Figure 19-5, select all projects in the Message Brokers Toolkit.

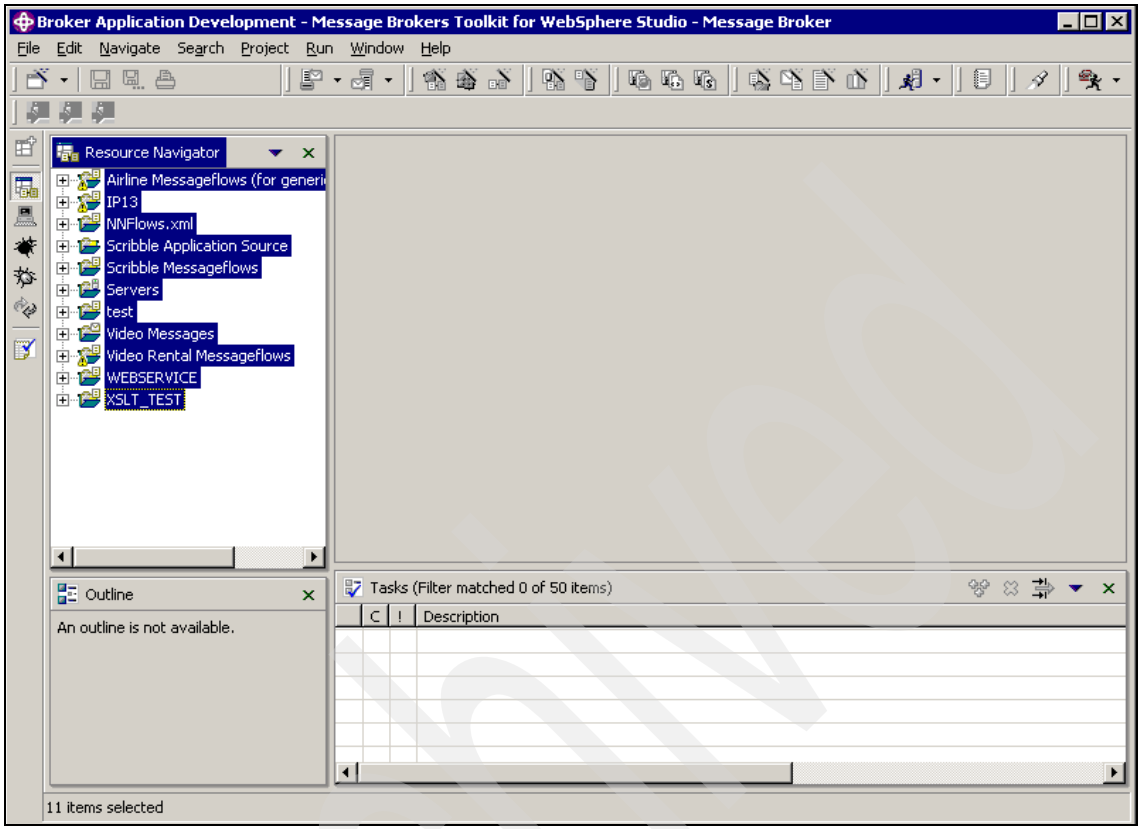


Figure 19-5 Selecting all projects

2. Select **File** → **Export** and choose the **Zip file** option from the menu, as illustrated in Figure 19-6 on page 349. Then click Next.

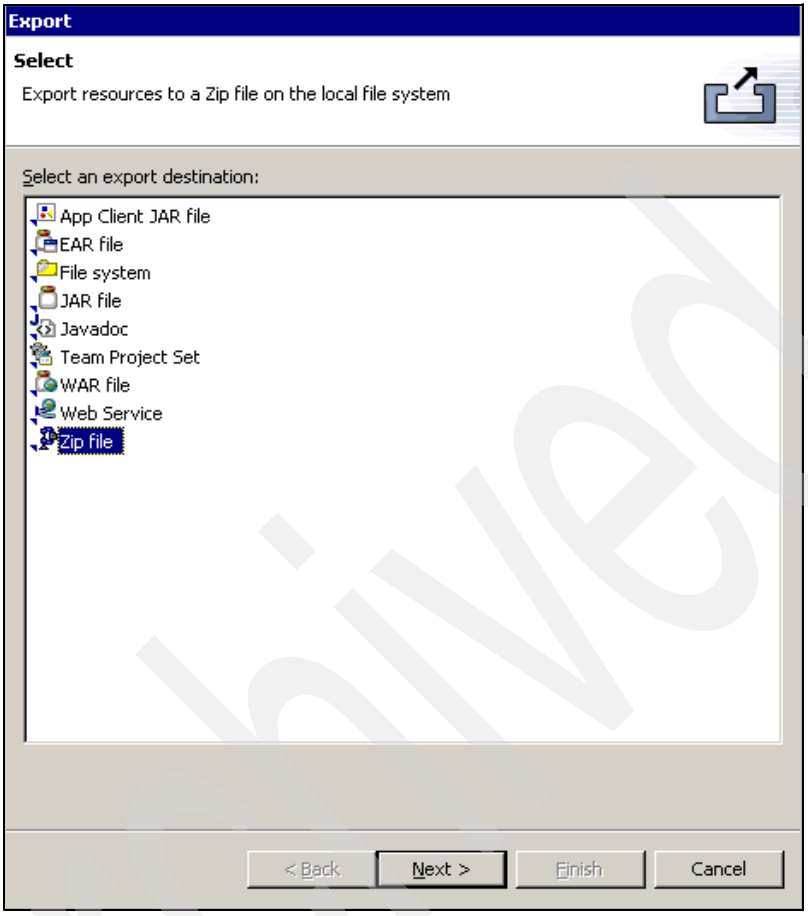


Figure 19-6 Choosing the Zip file option

3. Click Browse and select the export destination (for example, c:\WSBackup\TK_Workspace_V5.zip, as illustrated in Figure 19-7 on page 350). Click Save to save the file.

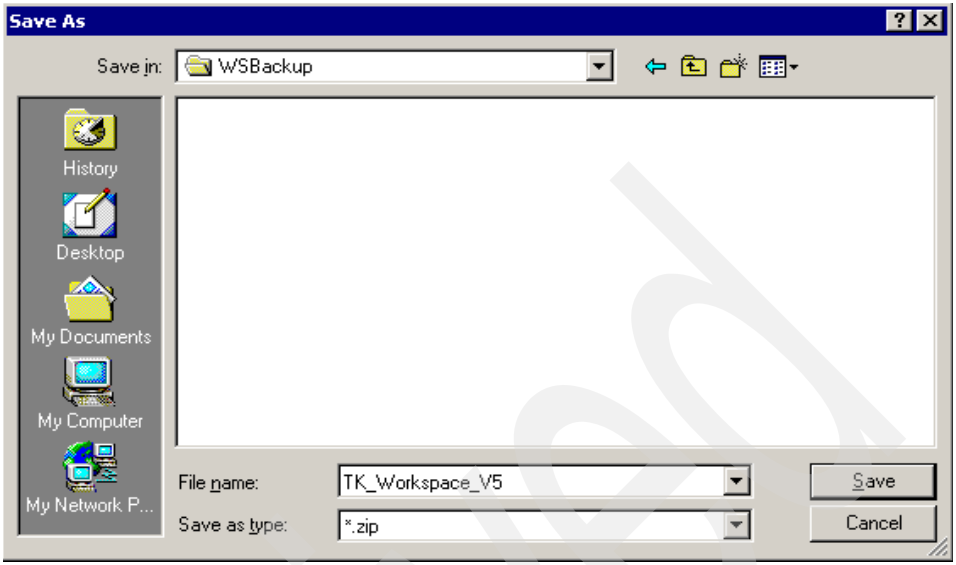


Figure 19-7 Choosing directory destination and zip file name

4. Ensure that the radio button `Create directory structure for files` is selected, as illustrated in Figure 19-8 on page 351. Then click Finish.

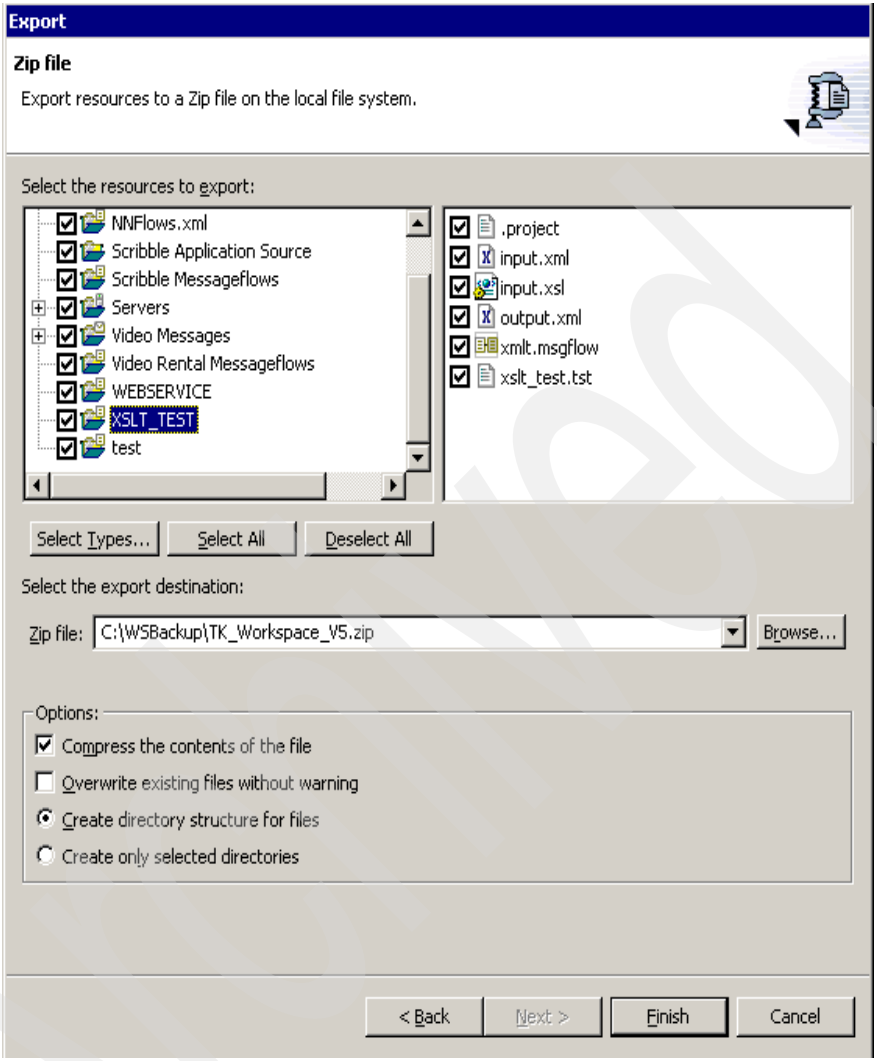


Figure 19-8 Create directory structure

Note: After taking all the backups, we recommend that you copy all the backup data to a secure external source location.

19.4.1 User-defined nodes/parsers

Follow these steps:

1. Determine whether the broker is using the user-defined nodes or parsers.
2. Find out the install directory for defined nodes or parsers by reviewing the registry entry LilPath, as described here:
   - On UNIX machines, check the directories specified in the registry entry */var/mqsi/registry/<broker_name>/LilPath.*
   - On z/OS machines, look at the value of variable *MQSI_REGISTRY* in environment file ENVFILE of broker service userID and check the directories at subdirectory *registry/<broker_name>/LilPath* inside this directory.

     For example, if *MQSI_REGISTRY* is */var/wmqi*, and broker name is *MQZIBRK*, check the directories specified in the entry */var/wmqi/MQZIBRK/LilPath*.
   - On Windows machines, check the directories specified in registry entry *HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphereMQIntegrator\<broker_name>\LilPath.*
3. Save all *.jar, *.zip and *.lil files from these directories to a user directory.

20

Migrating WebSphere Message Broker V5.0 and V5.1 tools and resources

This chapter describes how to migrate WebSphere Business Integration Message Broker V5.0 Toolkit and WebSphere Business Integration Message Broker V5.1 Toolkit resources to WebSphere Message Broker V6.0 Toolkit.

It also discusses using the WebSphere Message Broker V6.0 Toolkit on Linux (x86 platform).

20.1 The WebSphere Message Broker V6.0 Toolkit

It is assumed that the WebSphere Message Broker V6.0 Toolkit has been installed as described in the installation guide into a different location on the same machine that has either WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 Toolkit installed.

Note: If the WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 with Rules and Formatter Extension Toolkit was used and the Rules and Formatter Extension resources are being migrated to WebSphere Message Broker V6.0, then install the WebSphere Message Broker V6.0 with Rules and Formatter Extension Toolkit.

Note the following:

- The new WebSphere Message Broker V6.0 Toolkit is based on IBM Eclipse SDK Version 3.01 and Rational Application Developer Version 6.0.0.1.
- It can coexist on the same system with either WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 Toolkit.
- It is available on both Windows and Linux (x86 platform).

20.2 Migration preparation of the tooling resources to WebSphere Message Broker V6.0

This section discusses the planning considerations and preparation necessary before performing the migration of the tooling resources to WebSphere Message Broker V6.0. It also explains the tasks you need to perform in order to migrate the resource.

Figure 20-1 illustrates the Message Brokers Toolkit before and after migration scenario of WebSphere Business Integration Message Broker V5.0 Toolkit resources.

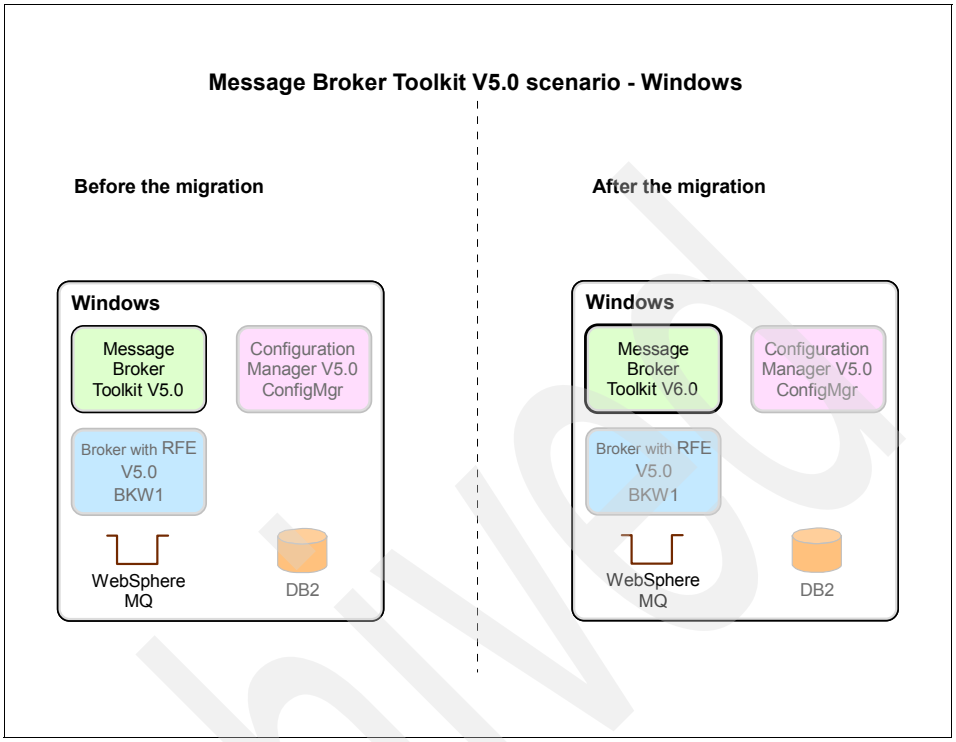


Figure 20-1 Environment before and after the migration of WebSphere Business Integration Message Broker V5.0 Toolkit resources

20.2.1 Preparing workspace resources for migration

The Message Brokers Toolkit workspace holds all the broker development and deployment resources, including message flow files, ESQL files, message set files, mapping files, broker archive files and domain connections.

Important: After WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 Toolkit resources have been opened and saved in the WebSphere Message Broker V6.0 Toolkit, they can no longer be used in previous Message Brokers Toolkits.

Therefore, it is assumed that WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 resources have been backed up following the recommended backup strategy as described in 19.4, "Message Brokers Toolkit workspace backup" on page 347 to produce a zip file of the workspace contents.

This ensures that projects in the workspace that have been imported from existing projects held elsewhere on the file system are included in the workspace backup.

User-defined nodes, SupportPacs and third party resources

Before a message flow using a user-defined node written in the WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 Toolkit can be deployed on the WebSphere Message Broker V6.0 Toolkit, the user-defined node must be migrated to the WebSphere Message Broker V6.0 Toolkit. Therefore, in preparation for migration, it is essential to identify if any message flows utilize user-defined nodes and ensure that the user-defined node project is available to be migrated.

If any broker development resource utilizes a SupportPac or other third party resource, it is essential that the user investigate the availability of the resource at WebSphere Message Broker V6.0 before embarking on the migration. If a SupportPac or third party resource has not been updated to support WebSphere Message Broker V6.0, the user must determine:

- Whether the functionality has been incorporated into WebSphere Message Broker V6.0

  New WebSphere Message Broker V6.0 functionality since WebSphere Business Integration Message Broker V5.0 is described in Chapter 4., "New features in WebSphere Business Integration Message Broker V5.0" on page 55, and is also documented in the WebSphere Message Broker information center; select the path **Product overview** → **Introduction** → **What's new in Version 6.0?**, available on the Web:

  `http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ab00025_.htm`

- Whether the migration of the affected broker development resources needs to be delayed until WebSphere Message Broker V6.0 support is available.
- Whether the broker development resource needs to be redesigned.

20.3 Migrating the tooling resources to WebSphere Message Broker V6.0

This section discusses the migration of the tooling resources from WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0.

Migrating the Message Brokers Toolkit from WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 to WebSphere Message Broker V6.0 is documented in the WebSphere Message Broker Information Center; select the path **Migrating** → **Migrating from Version 5.0 products** → **Migrating from WebSphere Business Integration Message Broker Version 5.0** → **Migrating the Message Brokers Toolkit**, available on the Web:

```
http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23850_.htm
```

The WebSphere Message Broker V6.0 Toolkit workspace can use the same location as the WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 Toolkit workspace. Starting the WebSphere Message Broker V6.0 Toolkit with a Version 5.0 or 5.1 workspace causes the workspace resources to be automatically rebuilt.

In the migration of the Message Brokers Toolkit resources demonstrated in this section, the WebSphere Business Integration Message Broker V5.0 Toolkit workspace is decoupled from the WebSphere Message Broker V6.0 Toolkit workspace by populating the Message Broker V6.0 Toolkit workspace with the contents of the workspace backup zip file, rather than directly using the old workspace directory.

This strategy allows the location of the WebSphere Message Broker V6.0 Toolkit workspace to be different from the previous version, and enables the resources in the workspace of the WebSphere Message Broker V6.0 Toolkit to be saved without affecting the resources used by the previous Message Brokers Toolkit.

If the workspace resources are stored in a version control code repository, an alternative strategy could be to produce an additional code stream based on existing resources.

1. Extract the workspace backup zip file into a suitable location on the file system. Then start the Message Brokers Toolkit by selecting **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **WebSphere Message Brokers Toolkit**.

   Or, in a Windows command prompt, change to the Message Brokers Toolkit installation directory and type: `wmbt.exe`.

2. After the WebSphere Message Broker V6.0 Toolkit is started, a dialog prompts the user to enter a workspace location, as illustrated in Figure 20-2.

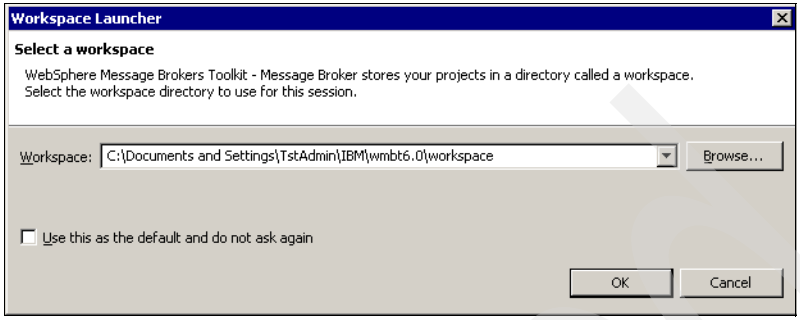


Figure 20-2 Selecting the workspace location

Specify the location of the directory into which the workspace backup zip file was extracted and select OK.

3. When the Message Brokers Toolkit is launched for the first time, the welcome page illustrated in Figure 20-3 appears. Hover over the icons to reveal their information, from listing what is new to the samples available.

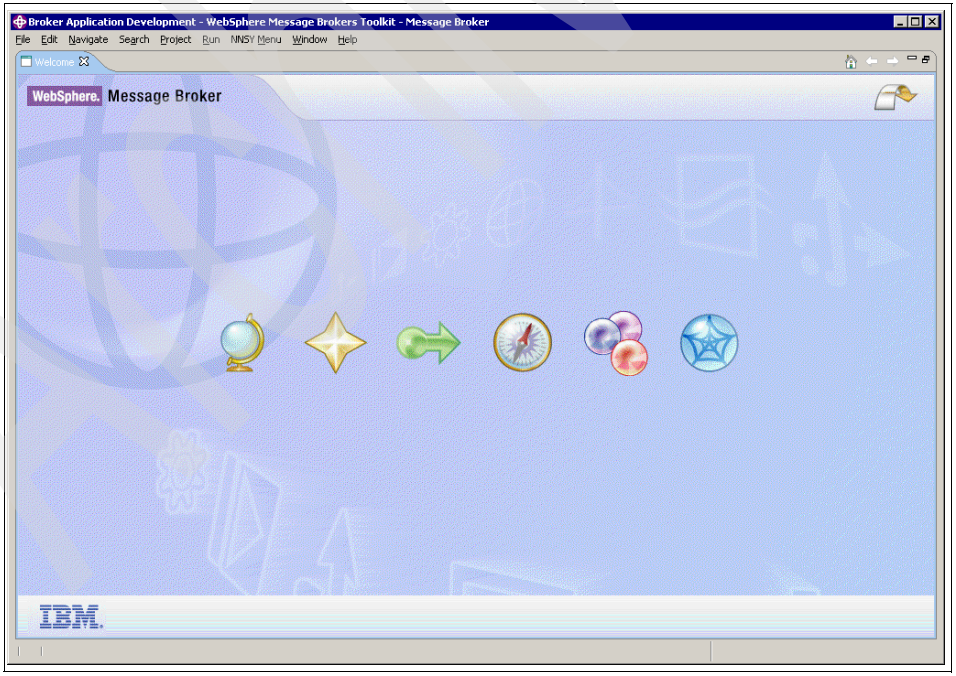


Figure 20-3 WebSphere Message Broker V6.0 Toolkit welcome screen

To close the welcome page and go to the workbench, select the curved arrow in the top right corner of the page, as illustrated in Figure 20-3. The Message Brokers Toolkit opens in the Broker Application Development perspective.

Tip: The icons used to switch between perspectives in WebSphere Message Broker V6.0 are now located at the top right of the workbench.

20.3.1 Migrating message flows and message sets

With the backed-up workspace from WebSphere Business Integration Message Broker V5.0 extracted into the WebSphere Message Broker V6.0 workspace directory, the existing projects need to be imported into the workbench to be seen.

1. In the Broker Application Development perspective, select **File** → **Import...** from the menu.
2. In the dialog box that appears, select option **Existing Project into Workspace**, then select Next.
3. In the dialog box that appears next, select Browse and choose one of the existing projects from the workspace directory, then choose OK. As shown in Figure 20-4, the Project name is automatically detected from the existing project selected.

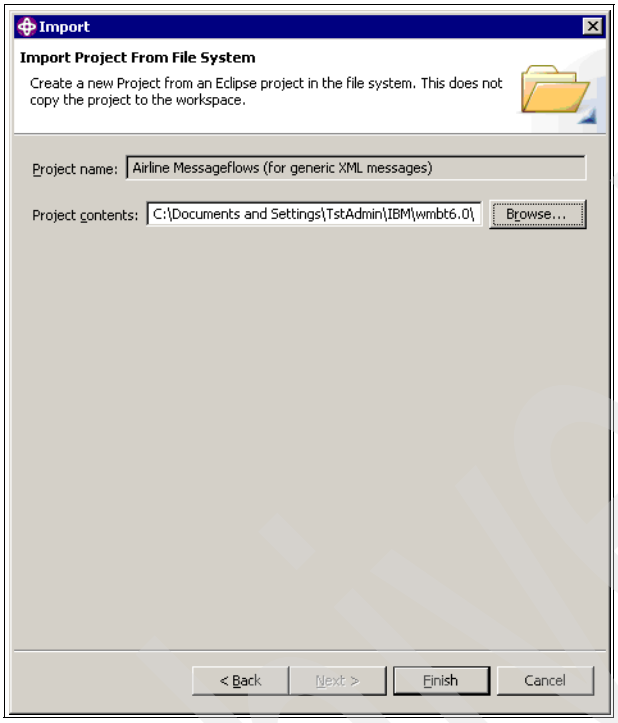


Figure 20-4 Importing an existing project from backed-up workspace extracted into WebSphere Message Broker V6.0 workspace directory

Select Finish; the project is displayed in the Resource Navigator view.

4. Repeat the process for each project.
5. After all projects can be seen in the workbench, clean and rebuild the workspace. Select **Project** → **Clean...** from the menu and the dialog box shown in Figure 20-5 appears.

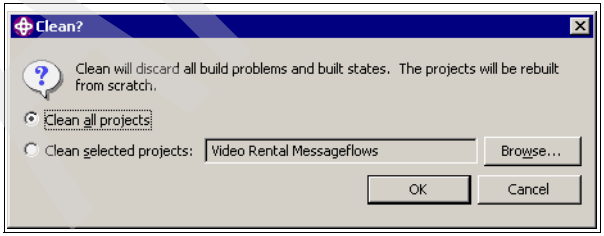


Figure 20-5 Clean all projects

Select the option `Clean all projects`, then select OK.

The resources in each of the projects is rebuilt. Depending on the number and complexity of the projects, the cleaning process may take some time. The progress of the process is indicated by a progress bar located at the bottom right of the workbench; see Figure 20-6.

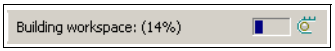


Figure 20-6 The progress of the Clean all projects process

Attention: Be aware that there have been some changes in the behavior of some of the properties of message sets, as explained in the WebSphere Message Broker Information Center; select the path **Reference → Migration and upgrade → Message set migration notes**, available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah20250_.htm`

After the WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 Toolkit resources have been opened and saved in the WebSphere Message Broker V6.0 Toolkit, they can no longer be used in the previous Message Brokers Toolkits.

Message flows containing XSL Transformation nodes

WebSphere Message Broker V6.0 has been modified to allow stylesheets and XML files to be deployed, which can be used by the XSL Transformation node. This option simplifies the user deployment process, as well as maintenance.

The WebSphere Message Broker Information Center's topic **Reference → Migration and upgrade → Message flow migration notes → Migrating style sheets, XML files and their references** documents the modifications necessary for message flows utilizing the XSL Transformation node to use the new deployment feature.

- If the XSL Transformation node references its principal stylesheet with a fully-qualified or absolute path, the message flow remains working as at WebSphere Business Integration Message Broker V5.0 without modification.
- If the XSL Transformation node references its principal stylesheet with a relative reference and does not utilize the new deployment feature, the relatively referenced stylesheet and its relatively referenced descendants must be moved to the correct directory structure under the *<broker_work_path>/XSL/external* directory.

  All dependent XML files must also be moved to the correct directory structure under the <broker_work_path>/XML/external directory.

Note: The location of the <broker_work_path> is defined by the *MQSI_WORKPATH* environment variable set by the mqsiprofile.

On Windows, mqsiprofile.cmd is run as a result of starting the Command Console.

- WebSphere Message Broker V6.0 treats stylesheets embedded in a message in the same way as a file loaded from the root of the broker's deployed storage.

  Therefore, any descendant stylesheet or XML files must either be added to a broker archive file manually, or moved to the appropriate <broker work path>/{XSL|XML}/external directory structure. If this is not possible, the embedded stylesheet must reference its descendants with fully-qualified or absolute path references.

Attention: If the new deployment method is going to be utilized, the XSL Transformation node must use relative references to the principal stylesheet and the stylesheets and XML files to be deployed must be imported into a project in the workspace.

If the stylesheets are not in the same message flow project, the project in which they exist must be referenced.

- Utilizing the Browse... mechanism of specifying the principal stylesheet when setting the XSL Transformation node's properties causes the file to be automatically pulled into the broker archive file when the message flow is added. Any dependent files are also added, as long as they are in a referenced project in the workspace.

Attention: Message flows that are running must be restarted in order to pick up a stylesheet that has been deployed for the first time; this is working as designed.

The stylesheet editor

The WebSphere Message Broker V6.0 Toolkit offers users a suite of tools provided by the Rational workbench, and these include XML tools. The default action when a stylesheet (.xsl or .xslt) file is opened in the Message Brokers Toolkit is to open the file in the default Web browser.

By following the instructions listed here to enable XML Developer capabilities, stylesheet files can be opened with an XSL editor within the Toolkit, allowing the

use of facilities such as command assist. To enable XML Developer capabilities in the WebSphere Message Broker V6.0 Toolkit, so the following:

1. From the menu select **Window** → **Preferences...**.
2. In the Preferences dialog that appears, expand the Workbench entry in the left pane, then select the Capabilities option. In the right pane, check the option `XML Developer`, as illustrated in Figure 20-7.

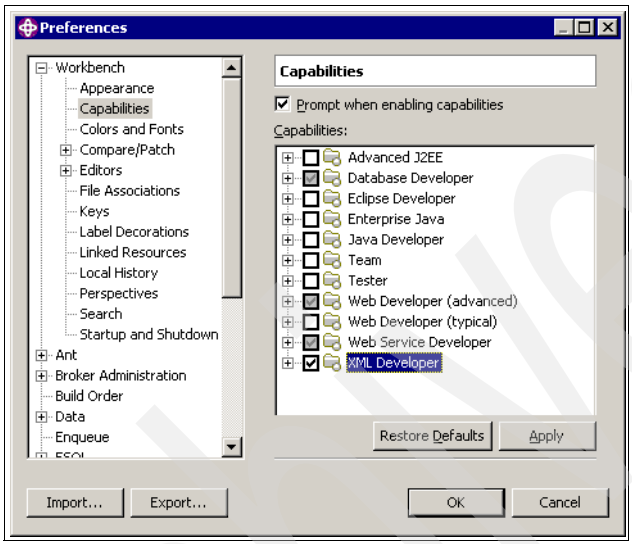


Figure 20-7 Setting the workbench capabilities

Select Apply, then OK.

3. Double-click a stylesheet (.xsl or .xslt) file in a project in the workbench; the file opens in the XSL editor.

Message flows containing WebSphere MQ Everyplace nodes

The support for connecting WebSphere MQ Everyplace device clients to WebSphere Message Broker has changed in WebSphere Message Broker V6.0.

- The use of MQeInput and MQeOutput nodes is deprecated in WebSphere Message Broker V6.0. The nodes still exist on the Message Brokers Toolkit's node palette, allowing message flows to be imported or created containing them. Note the following:
  - When a message flow containing them is deployed to a previous version broker, there is no change.
  - When a message flow containing them is deployed to a WebSphere Message Broker V6.0 broker, the runtime interprets the broker archive file differently.

This behavior provides a route for migration and allows for deployment to previous brokers.

- It is only possible to connect WebSphere MQ Everyplace device clients to WebSphere Message Broker V6.0 through a WebSphere MQ Everyplace gateway.
- WebSphere Message Broker V6.0 no longer installs a version of WebSphere MQ Everyplace; a version must be installed separately if required.

The WebSphere Message Broker Information Center fully documents the changes required; follow the path **Reference** → **Migration and upgrade** → **Message flow migration notes** → **Migrating a flow containing WebSphere MQ Everyplace nodes**, available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac34700_.htm

Message flows containing promoted properties

In the migration scenario being demonstrated, a WebSphere Business Integration Message Broker V5.0 message flow containing promoted properties was migrated to the WebSphere Message Broker V6.0 Toolkit. The Problems view displayed the following error for the message flow:

```
Property 'URLSpecifier' promoted from node 'HTTP Request' does not match
original property editor
```

To resolve the issue, the promoted property reporting the error must be removed and re-added to the promoted properties, as follows:

1. Open the affected message flow in the message flow editor by double-clicking it in the Resource Navigator.
2. Right-click in a blank space on the message flow editor and select the option **Properties...** from the context menu that appears. Note the value of the entry `Web Service URL`.
3. Right-click the affected node and select **Promote Property...**.
4. In the dialog box that appears, the promoted properties for this node are listed in the right pane and the remaining properties available for promotion are in the left pane.
   a. Note the group that the entry `Web Service URL` belongs to, then select the property from the pane on the right and click Remove.
   b. Select the *Web Service URL* entry from the pane on the left and select **Promote...**.
   c. In the *T*arget Selection dialog that appears, select the group (as noted in step a), and select OK.

d. In the Promote Property dialog, select OK to apply the changes and close the dialog box.

5. Right-click in a blank space on the message flow editor and select the option **Properties...** from the context menu that appears. Complete the entry `Web Service URL` with the value noted in step 2. Select OK to apply the changes and close the dialog box.
6. When you save the message flow changes, the error indicated in the Problems view is removed.

Depending on the types of resources in the projects, the Problems view may still show a number of errors, as mapping files and user-defined nodes need to be migrated, as discussed in the following sections.

20.3.2 Migrating mappings

The mapping editor provides a way to describe transformations in message flows in a graphical user interface rather than through ESQL. The mapping editor in WebSphere Message Broker V6.0 has been improved to provide improved XML schema support, additional brokering scenarios (such as splitting batch messages and mapping message headers), debugging support, and support for XPath expression grammar.

As a result, the format of the mapping files has changed and the WebSphere Message Broker V6.0 Toolkit's mapping editor only recognizes .msgmap file format, so double-clicking a .mfmap file only opens a text editor.

An error is reported for a project with a migrated message flow that references a WebSphere Business Integration Message Broker V5.0 .mfmap file. The Resource Navigator view displays a red cross against the project, the affected message flows, and the Problems view has an entry explaining that the map is unable to be located in the project or a referenced project.

To resolve the errors, WebSphere Business Integration Message Broker V5.0 .mfmap files need to be migrated using the **`mqsimigratemfmaps`** command documented in the WebSphere Message Broker Information Center; follow the path **Reference** → **Operations** → **Commands** → **Message broker Toolkit commands** → **mqsimigratemfmaps**, available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an26140_.htm`

1. Close Message Brokers Toolkit.

2. In a command prompt, change to the Message Brokers Toolkit installation directory's eclipse directory. With the default installation directory, this would be `C:\Program Files\IBM\MessageBrokersToolkit\6.0\eclipse`.

   – To check all projects in the workspace for .mfmap files to be migrated, issue the following command:

     mqsimigratemfmaps -all -data *<workspace_directory>*

   – To specify a specific project to have its .mfmap files migrated, issue the following command:

     mqsimigratemfmaps -p *<project>* **-data** *<workspace_directory>*

   > **Tip:** If the project name or workspace directory path contains spaces, put double quotes around the name or location.
   >
   > For example, specify a project named My Project as `-p "My Project"` in the command.

   Example 20-1 demonstrates issuing the **mqsimigratemfmaps** command to migrate a specific project.

Example 20-1 Migrating the .mfmap files for a specified project in the workspace

```
C:\Program Files\IBM\MessageBrokersToolkit\6.0\eclipse>mqsimigratemfmaps -p
"Video Rental Messageflows" -data "C:\Documents and
Settings\TstAdmin\IBM\wmbt6.0\workspace"

Migrating workspace project: Video Rental Messageflows
Migrating map file: Video_Test_Mapping.mfmap

Migration completed with the following statistics:
        Affected projects: 1 out of 1
        Affected files:    1 out of 1
        Errors: 0, Warnings: 0, Infos: 0
Refer to report file C:\Program
Files\IBM\MessageBrokersToolkit\6.0\eclipse\mqsimigratemfmaps.report.txt
for details.
After restarting the workbench, please refresh and rebuild all workspace
projects.
```

3. Confirm the success of the command by reviewing the `mqsimigratemfmaps.report.txt` file.

   By default, this file is located in the directory <toolkit_installation_directory>\eclipse unless the command is issued with an alternative directory and filename specified using the **-log** command option.

Example 20-2 illustrates the contents of the `mqsimigratemfmaps.report.txt` file after a successful migration.

Example 20-2 Contents of the mqsimigratemfmaps.report.txt file after a successful .mfmap migration

```
################################################################################
# Mapping Migration 6.0.0, Wed Oct 12 18:02:00 BST 2005
################################################################################
================================================================================
Migrating workspace project: Video Rental Messageflows
--------------------------------------------------------------------------------
Migrating map file: Video_Test_Mapping.mfmap
================================================================================
Migration completed with the following statistics:
   Affected projects: 1 out of 1
   Affected files:    1 out of 1

Errors: 0, Warnings: 0, Infos: 0
```

Because mappings are migrated by a tool, it is important to test their functionality carefully to verify that they work as expected. The WebSphere Business Integration Message Broker V5.0 broker archive files containing message flows with mappings can continue to be used while verifying the migrated resources.

Attention: The WebSphere Message Broker Information Center documents the mapping functions that are no longer supported in WebSphere Message Broker V6.0. (Mappings containing these functions cause errors in the mqsimigratemfmaps.report.txt file.) Follow the path **Reference → Migration and upgrade → Message flow migration notes → Migrating message mappings from Version 5.0** , available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ar25253_.htm`

View the error report to see details of the functions that could not be migrated, and create a new node that executes the non-migrated functions. Alternatively, you can recreate the mapping function using another node such as a Java Compute node.

4. Reopen Message Brokers Toolkit with the same workspace specified and switch to the Broker Application Development perspective. The project containing the message flow referencing a mapping still shows an error. Right-click the affected project and select Refresh from the context menu, as shown in Figure 20-8.

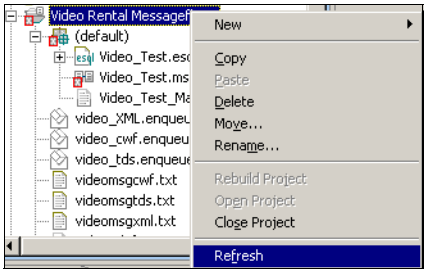


Figure 20-8 Refresh the message flow project

Refreshing the project to causes the additional .msgmap file to appear in the Resource Navigator.

Note: The .mfmap files are deliberately left in the workspace so that migration can be run again. However, they are not recognized as valid WebSphere Message Broker development artifacts and are not included in builds.

The .mfmaps files can be deleted at the discretion of the user.

5. To associate the new .msgmsp file with the message flow and remove the errors, select the project and then **Project** → **Clean...**. In the dialog box that follows, select option `Clean selected projects` and then select OK.

6. Double-click the .msgmap file to launch the mapping editor, where the mapping can be edited.

Restriction: There are certain scenarios where the migration of mfmap files is not supported. The WebSphere Message Broker Information Center explains why migration is not automatic in these situations, and provides instructions for how to complete a successful migration; follow the path **Reference** → **Migration and upgrade** → **Message flow migration notes** → **Migrating message mappings from Version 5.0** → **Restrictions on migrating message mappings**, available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ar25255_.htm

If a .mfmap file, references a namespace prefix defined in an .esql file, the **mqsimigratemfmaps** command produces an error like the following example:

```
"Error:134: The MFMAP does not define a prefix for namespace '<namespace>'
of the XSD component '<xsd_component>'. Please correct this problem manually
after migration completes."
```

It is not possible to correct this in the mapping editor. This issue should be resolved in a future fix pack.

20.3.3 Migrating user-defined nodes

Migrating a user-defined node is documented in the WebSphere Message Broker Information Center; follow the path **Reference** → **Migration and upgrade** → **Message flow migration notes** → **Migrating user-defined nodes from Version 5.0**, available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/as10009_.htm

For a migration of the user-defined node Message Brokers Toolkit resources, follow these steps:

1. Import the user-defined node project into the WebSphere Message Broker V6.0 Toolkit workspace.
2. From the menu, select **Project** → **Clean...**, then choose option `Clean selected projects`.

User-defined nodes from the WebSphere Business Integration Message Broker V5.1 Toolkit are now migrated.

The user-defined nodes from the WebSphere Business Integration Message Broker V5.0 Toolkit need the following extra steps to complete the migration:

1. Open the *plugin.xml* file in the root of the user-defined node project.
2. Modify the *<requires>* element to match Example 20-3.

Example 20-3 Modification required for the <requires> element of a WebSphere Business Integration Message Broker V5.0 user defined node's plugin.xml file

```
<requires>
 <import match="greaterOrEqual" plugin="com.ibm.etools.mft.api" version="6.0.0"/>
</requires>
```

3. Modify the *org.eclipse.help.contexts* extension to match Example 20-4.

Example 20-4 Modification required for the org.eclipse.help.contexts extension of a WebSphere Business Integration Message Broker V5.0 user defined node's plugin.xml file

```
<extension point="org.eclipse.help.contexts">
 <contexts file="HelpContexts.xml"/>
</extension>
```

20.4 Using the WebSphere Message Broker V6.0 Toolkit

After you migrate WebSphere Business Integration Message Broker V5.0 Toolkit resources, they can be used in the WebSphere Message Broker V6.0 Toolkit. As in the WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 Toolkit, you open resources in the appropriate editor by double-clicking the files.

- The Broker Application Development perspective should be used for development resources such as message flows, message sets or mappings.

  Tip: When the message flow editor is first opened, the node palette is minimized on the left of the message flow editor.

  To maximize the node palette to see the available nodes, click the small blue triangle above the word `Palette`.

- The Broker Administration perspective should be used for deployment resources such as a broker archive files and domain connections available from the Broker Administration Navigator.

20.4.1 Coexisting with WebSphere Business Integration Message Broker V5.0 resources

As with WebSphere Business Integration Message Broker V5.0 and WebSphere Business Integration Message Broker V5.1 Toolkits, WebSphere Message Broker V6.0 Toolkits can connect to multiple domains concurrently.

The Message Brokers Toolkits can connect to either a WebSphere Business Integration Message Broker V5.0, or to WebSphere Message Broker V6.0 Configuration Manager.

Connecting to the domain

Having imported the server projects from the WebSphere Business Integration Message Broker V5.0 Toolkit, the Broker Administration perspective navigator's Domain Connections and Broker Archives containers contain the server projects with the existing connections and broker archive files.

In the Broker Administration perspective's Domains view, right-click the existing domain connection and select **Connect**.

Figure 20-9 illustrates the coexistence of WebSphere Message Broker V6.0 Toolkit with a WebSphere Business Integration Message Broker V5.0 Configuration Manager managing a domain of WebSphere Business Integration Message Broker V5.0 brokers.

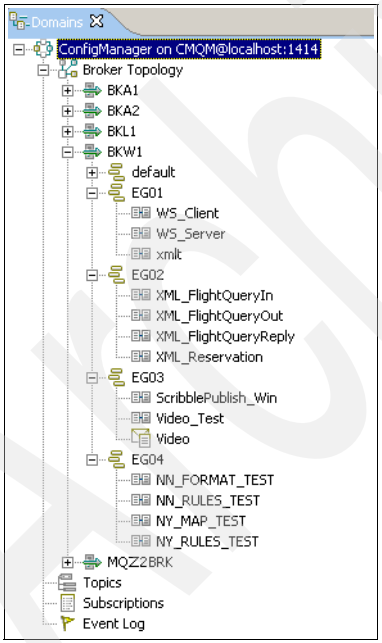


Figure 20-9 WebSphere Message Broker V6.0 Toolkit connected to a WebSphere Business Integration Message Broker V5.0 Configuration Manager

Coexistence with components from previous versions provides an opportunity to conduct a flexible, staged migration process. The WebSphere Message Broker Information Center documents which WebSphere Message Broker V6.0

components can coexist with components from previous versions; follow the path **Migrating** → **Coexistence**, available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23930_.htm

The WebSphere Business Integration Message Broker V5.0 domain can be administered in the WebSphere Message Broker V6.0 Toolkit. For example, deployed resources such as message flows can be started, stopped, and removed from execution groups in the Domains view of the Broker Administration perspective.

Deploying the migrated resources

Broker archive files can be deployed from WebSphere Message Broker V6.0 Toolkit to WebSphere MQ Integrator V2.1, WebSphere Business Integration Message Broker V5.0, and WebSphere Message Broker V6.0 brokers through both WebSphere Business Integration Message Broker V5.0 and WebSphere Message Broker V6.0 Configuration Manager domains.

Important: The existing broker archive files migrated from either the WebSphere Business Integration Message Broker V5.0 or WebSphere Business Integration Message Broker V5.1 Toolkit can be deployed in the WebSphere Message Broker V6.0 Toolkit.

When creating new broker archive files in the WebSphere Message Broker V6.0 Toolkit, it is important to remember:

- All migrated development resources can be deployed to a runtime at the same or later version of WebSphere Message Broker than developed at.
- Development resources developed in later versions of the product can be deployed to earlier broker runtimes as long as the resources are not using functionality only available in later versions.

For example, a migrated WebSphere Business Integration Message Broker V5.0 message flow which references an ESQL file can always be deployed to a WebSphere Business Integration Message Broker V5.0 and a WebSphere Message Broker V6.0 broker.

However, it can only be deployed to a WebSphere MQ Integrator V2.1 broker if it only uses nodes and ESQL functions available at WebSphere MQ Integrator V2.1.

- The WebSphere Message Broker Information Center documents the conditions applicable for a WebSphere Business Integration Message Broker V5.0 broker participating in a WebSphere Message Broker V6.0 domain;

follow the path **Migrating → Coexistence → Conditions for a Version 5.0 broker participating in a Version 6.0 broker domain,** available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13705_.htm

- Table 20-1 from the WebSphere Message Broker Information Center summarizes the conditions applicable for a WebSphere MQ Integrator V2.1 broker participating in a WebSphere Message Broker V6.0 domain. You can follow this path **Migrating → Coexistence → Conditions for a Version 2.1 broker participating in a Version 6.0 broker domain**, available on the Web:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13700_.htm

Table 20-1 Conditions applicable for a WebSphere MQ Integrator V2.1 broker participating in a WebSphere Message Broker V6.0 domain

Configuration data	Can be deployed to a V2.1 broker?	Conditions
Message flows	Yes	Follow this path for topic ah13830_ **Migrating → Coexistence → Conditions for a Version 2.1 broker participating in a Version 6.0 broker domain → Conditions for deploying a message flow to a Version 2.1 broker**, available on the Web: http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13830_.htm
Message sets	Yes	Follow this path for topic ah13840_ **Migrating → Coexistence → Conditions for a Version 2.1 broker participating in a Version 6.0 broker domain → Conditions for deploying a message set to a Version 2.1 broker**, available on the Web: http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13840_.htm
Topology	Yes	No restrictions.
Topics	Yes	A WebSphere MQ Integrator V2.1 broker uses only the Access Control List for the topic and ignores any multicast or quality of protection (QoP) settings for the topic.
Broker properties	No	

Important: When adding message flows to a broker archive file that is going to be deployed to a WebSphere MQ Integrator V2.1 broker, select the `Compile ESQL for broker Version 2.1` check box in the broker archive editor. This is required because the WebSphere MQ Integrator V2.1 product does not understand the ESQL generated at later versions, which implements modules and schemas. However, a broker archive file created using this option can be deployed to a WebSphere Business Integration Message Broker V5.0 or WebSphere Message Broker V6.0 broker.

Having added resources to a broker archive file, select the `Details>>` button on the dialog box `Adding to Broker Archive File` that appears. This reports messages about the process of adding the resources to the broker archive.

For example, adding a migrated WebSphere Business Integration Message Broker V5.0 message set could output the following warning:

```
BIP0177W This dictionary requires a Version 5 (or later) broker.
This message set has namespace support enabled. Namespaces are not supported
by brokers earlier than Version 5.
```

Runtime versioning

At WebSphere Message Broker V6.0, all deployable resources can be tagged with a version and other useful information in addition to the standard compiled time and deployment time attributes.

The WebSphere Message Broker Information Center describes how keywords can be defined to be embedded in the deployable resources. Follow this path to the topic **Reference** → **Message flows** → **Description properties** → **Guidance for defining keywords**), available on the Web:

```
http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac35480_.htm
```

The syntax for defining a keyword is:

```
$MQSI Keyword = Value MQSI$
```

Version information is set using the *Version* property for message flows and message sets.

- In the message flow editor, right-click in a blank space and select **Properties...**. The version information can be entered in the Version field and additional user defined keywords specified in the Long Description field, as illustrated in Figure 20-10.

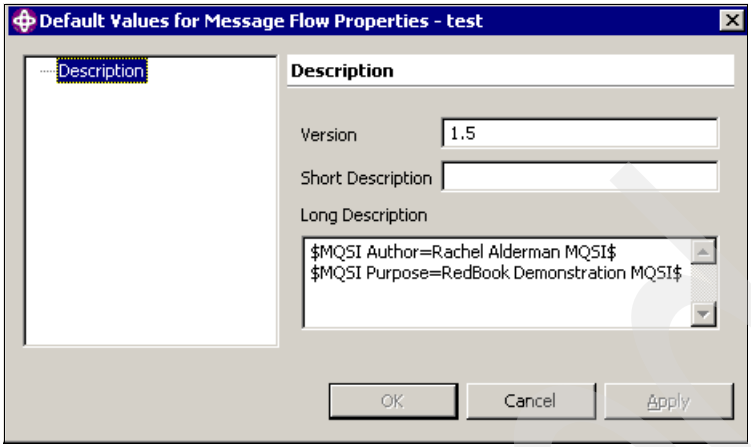


Figure 20-10 Setting message flow keywords

- In the message set editor's Documentation view, the Version field can be set along with additional user-defined keywords specified in the Documentation field.

For further detail on embedding keywords for each of the deployable resources, refer to the appropriate WebSphere Message Broker Information Center's topic as listed in Table 20-2.

Table 20-2 Topics on embedding keywords in deployable resources

Deployable resource	**WebSphere Message Broker Information Center topic**
Message flows	ac26550_ **Developing applications → Developing message flow applications → Managing message flows → Version and keywords**, available on the Web: http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac26550_.htm
ESQL	ac35430_ **Developing applications → Developing message flow applications → Developing ESQL → Writing ESQL → Adding keywords**, available on the Web: http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac25430_.htm

Deployable resource	WebSphere Message Broker Information Center topic
Subflows	ac35460_ **Developing applications → Developing message flow applications → Designing a message flow → Using subflows → Adding keywords, available on the Web**: http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac35460_.htm
Message sets	ad10326_ **Developing applications → Developing message models → Message modeling overview → The message model → Message sets → Versions and keywords**, available on the Web: http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ad10326_.htm
XSL stylesheets	ac35450_ **Reference → Message flows → Built-in nodes → XMLTransformation node → Adding keywords to XSL stylesheets**, available on the Web: http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac35450_.htm
JAR files	ac35440_ **Developing applications → Developing message flow applications → Developing Java → Writing Java → Adding keywords**, available on the Web: http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac35440_.htm

When the resources are added to a broker archive file and deployed, the information is displayed in the Broker Administration perspective. As illustrated in Figure 20-11, the version of the message flow appears in the broker archive file's Version column for the .cmf entry.

Also, by selecting the deployed message flow in the Domains view, the Properties view displays the properties and their values, as illustrated in Figure 20-11.

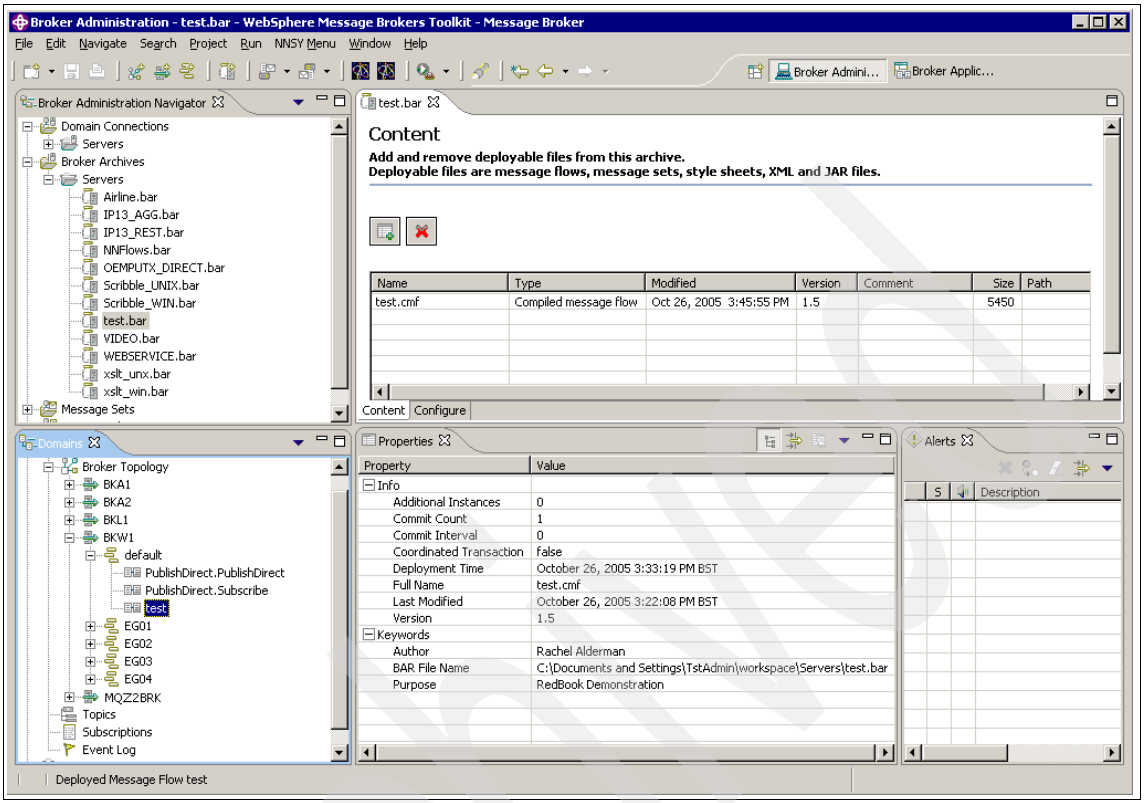


Figure 20-11 Displaying the runtime versioning in the Broker Administration perspective

This additional administration information makes it easy to see which resources have been deployed to which systems.

20.4.2 Sharing broker resources

The workspace of the WebSphere Message Broker V6.0 Toolkit is file-based, therefore we recommend that you use a code repository to share the resources with multiple concurrent users. The WebSphere Message Broker Information Center provides information on the repositories that are supported; follow this path **Product overview** → **Technical overview** → **Client environment** → **Development repository**, available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ab00240_.htm`

The use of a version control code repository provides the following benefits:

- Concurrent users of the Message Brokers Toolkit can share the resources without the risk of overwriting each other's work.
- Each Message Brokers Toolkit resource can be versioned, producing an archive of previous versions that can be accessed as necessary. For example, it would be possible to revert to a known working version of a message flow if the latest version has problems as the result of further development.
- Version control repositories can maintain a history of the evolution of the resource. This history can be used to track the changes to a resource, for example, which user made the change, the date and time of the change, and a description of the change.

Any repository with an Eclipse 3.0 client is supported. WebSphere Message Broker V6.0 Toolkit provides two perspectives to interact with code repositories.

- The Team Synchronizing perspective provides a view of the project sharing status of a resource, as well as tasks to synchronize project resources with supported repositories. Figure 20-12 illustrates the Team Synchronizing perspective from the Message Brokers Toolkit.

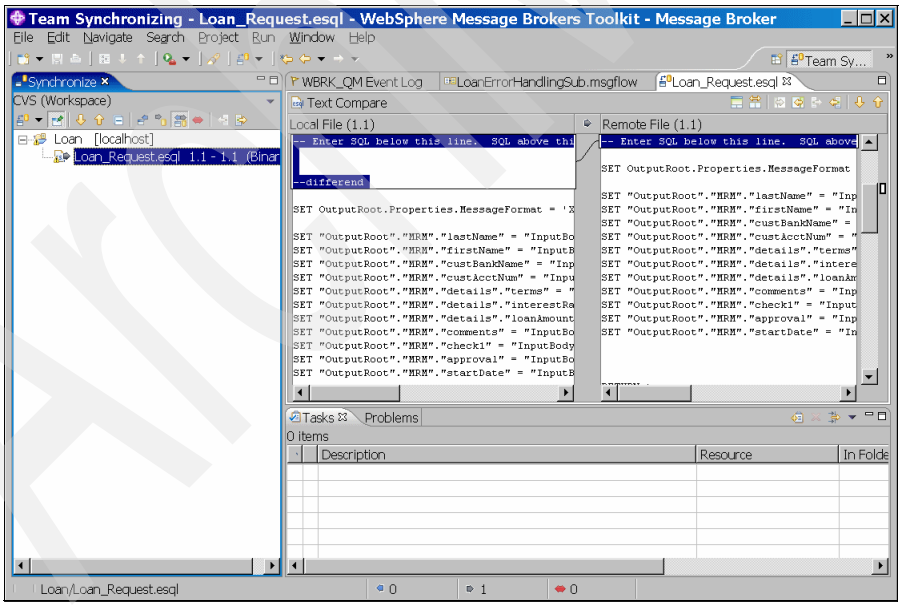


Figure 20-12 The Team Synchronizing perspective

- The CVS Repository perspective shows the content of a CVS based repository. Figure 20-13 illustrates the CVS Repository perspective from the Message Brokers Toolkit.

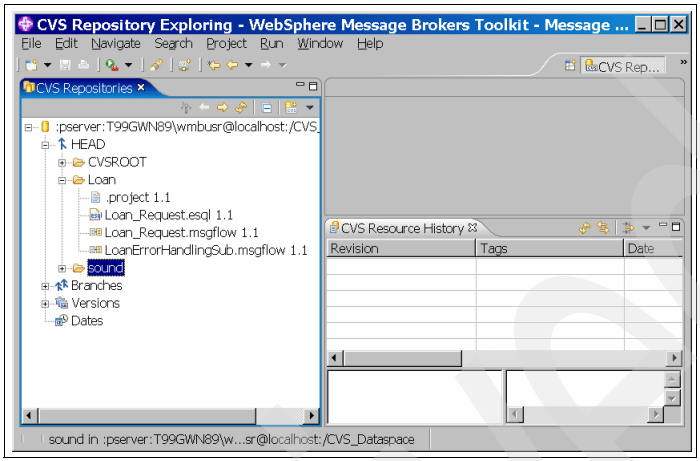


Figure 20-13 The CVS Repository perspective

The WebSphere Message Broker Information Center documents how to configure CVS to run with the Message Brokers Toolkit; follow this path **Configuring the broker domain** → **Configuring the workbench** → **Configuring CVS**, available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac26530_.htm`

To share a project in the Message Brokers Toolkit with a code repository, right-click the project and select **Team** → **Share Project...** from the context menu that appears. Use the wizard to add the project to a repository.

Having added the project to a repository, the Broker Application Development perspective's Resource Navigator view indicates the projects are managed by a code repository by changing the icons.

In Figure 20-14, the RequestLoan message set project and JOURNEY message flow project have been shared with a code repository. The > symbol on the JOURNEY message flow project and its ESQL file indicate that the project's ESQL file, which is at version 1.1 in the repository, has been changed in the workbench and needs to be synchronized with the repository.

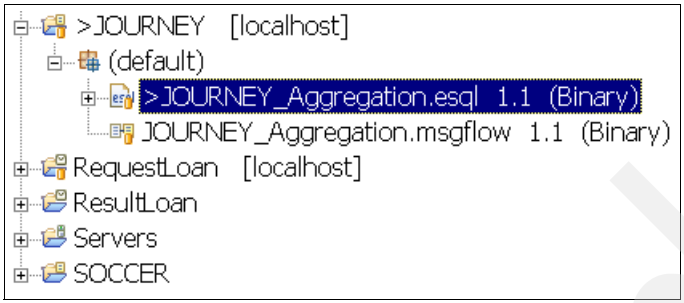


Figure 20-14 Projects in the Broker Application Development perspective shared with a code repository

20.4.3 Using the Linux (x86 platform) Message Brokers Toolkit

The Linux (x86 platform) Message Brokers Toolkit offers exactly the same graphical user interface and functionality as the Windows Message Brokers Toolkit. This section demonstrates the use of the WebSphere Message Broker V6.0 Toolkit on Linux (x86 platform) as an additional Toolkit within the migration environment, as illustrated in Figure 20-15.

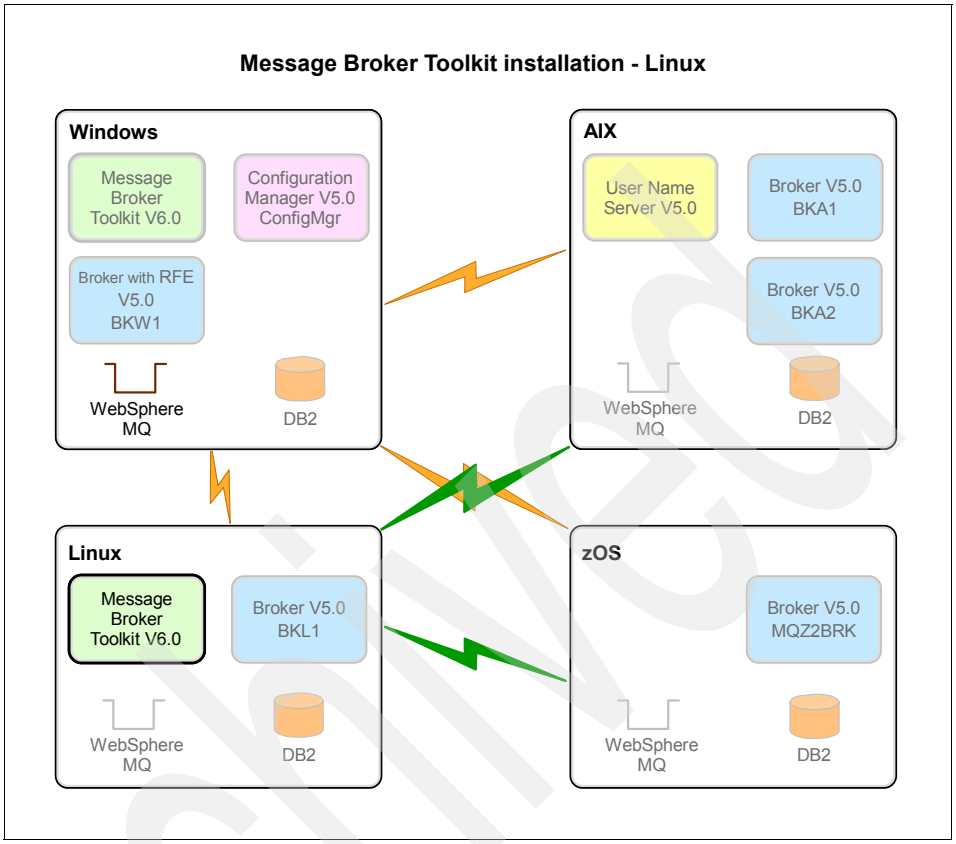


Figure 20-15 Environment using a WebSphere Message Broker V6.0 Toolkit on both Windows and Linux (x86 platform) systems

For the purposes of this chapter, to demonstrate the Linux (x86 platform) Message Brokers Toolkit, the successfully migrated Toolkit resources produced in 20.3, “Migrating the tooling resources to WebSphere Message Broker V6.0” on page 357 are extracted from the Windows WebSphere Message Broker V6.0 Toolkit and used to populate the Linux (x86 platform) Message Brokers Toolkit’s workspace, as explained in the following steps.

1. On the Windows system, start the Message Brokers Toolkit by selecting **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **WebSphere Message Brokers Toolkit**.
2. In the Broker Application Development perspective, select **File** → **Export...** from the menu. In the dialog box that appears, select the Project Interchange option and then select Next.

3. Click the Select All button and specify the location and file name of the zip file in the `To zip file` field, then select Finish.
4. Copy the workspace archive file created to an appropriate directory on the Linux (x86 platform) system. In the migration being demonstrated, workspace_v6.zip is moved using binary FTP to the Linux (x86 platform) system and stored in the user's home directory. Extract the workspace archive file into a suitable location on the file system,
5. Log onto the graphical desktop environment of the Linux (x86 platform) system. In the migration scenario demonstrated, this was using Red Hat Enterprise Linux 3.2.3 with KDE environment.

Note: It is assumed that the WebSphere Message Broker V6.0 Toolkit for Linux (x86 platform) has been installed, as described in the installation guide.

6. Start the Message Brokers Toolkit by selecting **Main Menu** → **Programming** → **WebSphere Message Brokers Toolkit**. Or, in a terminal window, change to the Message Brokers Toolkit installation directory and type: `wmbt.bin`.

   The default location for the Message Brokers Toolkit installation is /opt/ibm/MessageBrokersToolkit.
7. When the WebSphere Message Broker V6.0 Toolkit is started, a dialog prompts the user to enter a workspace location, as illustrated in Figure 20-16.

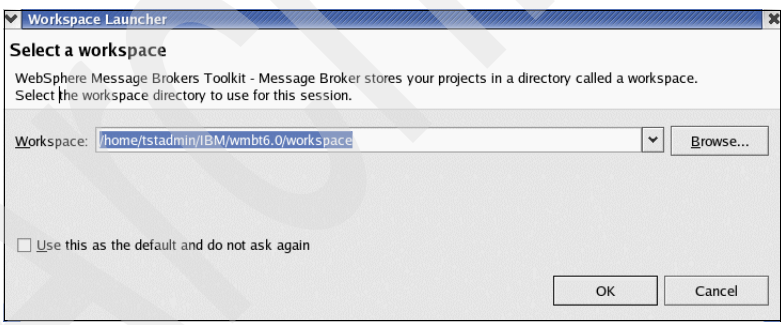


Figure 20-16 Selecting the workspace location

The default location is under the user's home directory. Select **Browse...** to specify the location of the directory into which the Windows workspace archive file was extracted, then select OK.

When the Message Brokers Toolkit is launched for the first time, the welcome page appears. To close the welcome page and go to the workbench, select the curved arrow in the top right corner of the page. The Message Brokers

Toolkit opens in the Broker Application Development perspective, as illustrated in Figure 20-17.

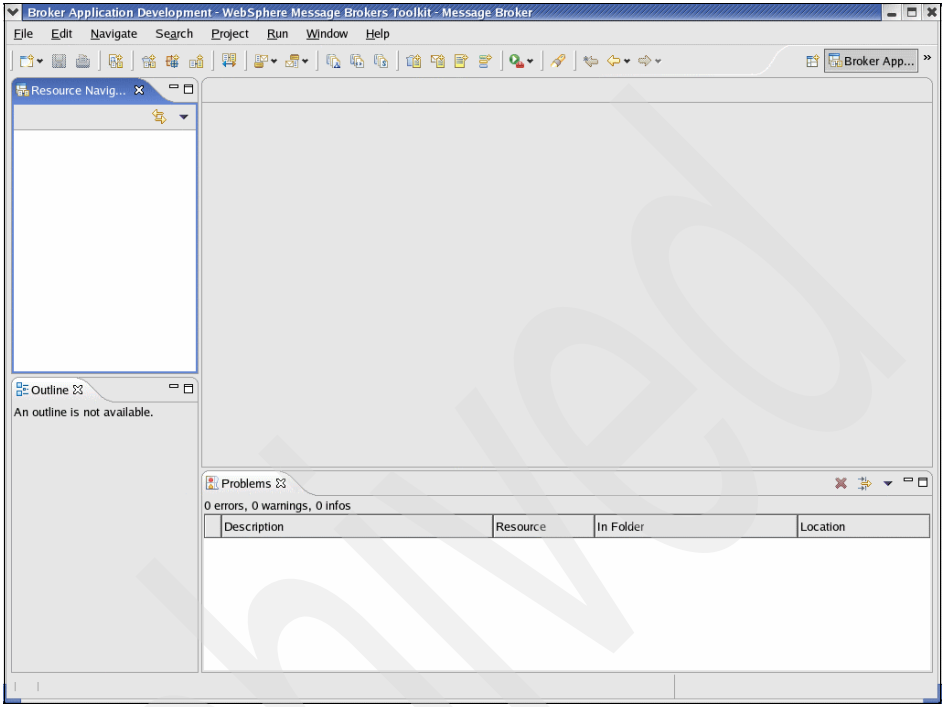


Figure 20-17 The Broker Application Development perspective on Linux (x86 platform)

8. With the extracted Windows workspace extracted into the Linux (x86 platform) workspace directory, the existing projects need to be imported into the workbench to be seen.

   a. In the Broker Application Development perspective, select **File** → **Import...** from the menu.

   b. In the dialog box that appears, select the option `Existing Project into Workspace`, then select `Next`.

   c. In the dialog box that appears, select `Browse` and choose one of the existing projects from the workspace directory, then select OK. As shown in Figure 20-18, the Project name is automatically detected from the existing project selected.

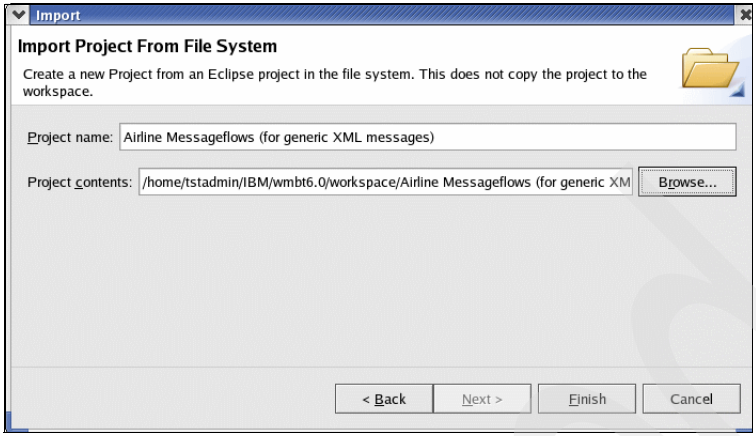


Figure 20-18 Importing an existing project from the backed up workspace extracted into the Linux (x86 platform) workspace directory

When you select `Finish`, the project is displayed in the Resource Navigator view. Repeat the process for each project.

9. After all projects have been imported into the workbench, it is good practice to clean the workspace. Select **Project** → **Clean...** from the menu. In the following dialog, select the option `Clean all projects` and then select OK.

With all WebSphere Message Broker V6.0 Toolkit resources imported into the Linux (x86 platform) workspace, the resources can be developed in the same way as on a Windows Message Brokers Toolkit.

It is now possible to connect to the broker domain. Having imported the server projects from the Windows Message Brokers Toolkit, the Broker Administration perspective navigator's Domain Connections and Broker Archives containers contain the server projects with the existing connections and broker archive files.

1. In the Broker Administration perspective's navigator, double-click the .configmgr file to open it. Modify the domain connection parameters as necessary.

   In the migration scenario being demonstrated, the Host value for the Configuration Manager needed to be changed from localhost to the hostname of the Windows system hosting the Configuration Manager. Save any changes and close the domain connection editor.

2. In the Domains view, right-click the existing domain connection and select Connect.

The Linux (x86 platform) Message Brokers Toolkit is connected concurrently to the same Configuration Manager as the Windows Message Brokers Toolkit. As

in WebSphere Business Integration Message Broker V5.0, multiple concurrent users administering the same broker domain can see the status updated in their Message Brokers Toolkit Domains view.

For example, having used the Linux (x86 platform) Message Brokers Toolkit to deploy a broker archive file imported from the Windows workspace:

- On the Linux (x86 platform) system, double-click the Event Log in the Domains view. The successful event messages BIP2056I and BIP4040I are reported.

  Note: The event messages reporting the status of an administration task are only reported to the Event Log of the Message Brokers Toolkit that issued the command.

  If command line commands such as **mqsideploy** are used to administer the domain, all Message Brokers Toolkits connected to the broker domain have the event messages reported in their Event Logs.

- The Domains view on all Message Brokers Toolkits connected to the same broker domain is updated, regardless of the platform on which they are running, showing the deployed resources.

21

Migrating the WebSphere Business Integration Message Broker V5.0 Configuration Manager

This chapter describes the process of migrating the WebSphere Business Integration Message Broker V5.0 Configuration Manager to WebSphere Message Broker V6.0.

It also provides details about changing the platform of the Configuration Manager to Advanced Interactive Executive (AIX), and enhancing the WebSphere Message Broker V6.0 Configuration Manager.

21.1 Migrating Configuration Manager to WebSphere Message Broker V6.0

WebSphere Business Integration Message Broker V5.0 Configuration Manager maintains the broker domain configuration, interfacing between the WebSphere Message Brokers Toolkit and a set of executing brokers.

The Configuration Manager acts as the central repository for broker domain components, communicating with other components in the domain using WebSphere MQ. It also deploys to the broker domain as a result of actions initiated through the WebSphere Message Brokers Toolkit, and reports the resultant status messages back to the Event Log.

Note: We recommend the following:

- Ensure that the WebSphere Business Integration Message Broker V5.0 domain is backed up as described in Chapter 19, “Backing up the WebSphere Business Integration Message Broker V5.0 domain” on page 339, in case there is a need to reinstall the environment.
- Ensure the WebSphere Business Integration Message Broker V5.0 Toolkit resources are migrated to the WebSphere Message Broker V6.0 Toolkit, as described in Chapter 20, “Migrating WebSphere Message Broker V5.0 and V5.1 tools and resources” on page 353.

There are several options for migrating a Configuration Manager. You can migrate it in situ on the same computer, migrate it to another computer where the DB2 Universal Database has not been installed, or migrate it to another computer where the DB2 Universal Database has been installed. We recommend that you select the scenario most appropriate to your environment.

For details about the various options available, refer to the topic “Migrating a Configuration Manager” by selecting the path **Migrating → Migrating from Version 5.0 products → Migrating from WebSphere Business Integration Message Broker Version 5.0 → Migrating a Configuration Manager** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23820_.htm

Whatever option you choose, ensure that the Configuration Manager is stopped prior to migration.

21.1.1 Migrating the Configuration Manager in situ

For documentation on the process of migrating the Configuration Manager on the same system, refer to the topic “To the same computer” by selecting the path **Migrating → Migrating from Version 5.0 products → Migrating from WebSphere Business Integration Message Broker Version 5.0 → Migrating a Configuration Manager → To the same computer** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23824_.htm

This section describes how to replace the WebSphere Business Integration Message Broker V5.0 Configuration Manager with a WebSphere Message Broker V6.0 Configuration Manager in situ. Figure 21-1 illustrates the environment before and after the migration of the Configuration Manager.

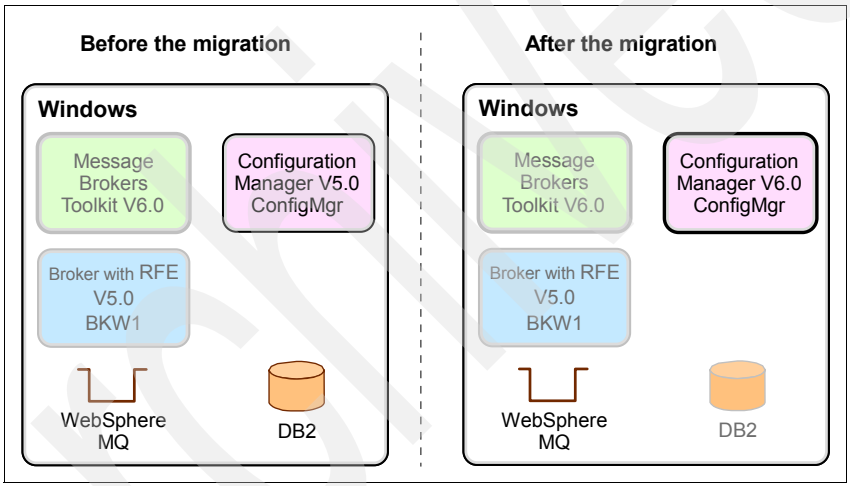


Figure 21-1 Environment before and after the migration of the Configuration Manager in situ

The environment shown in Figure 21-1 illustrates the coexistence of the WebSphere Message Broker V6.0 components with brokers at WebSphere Business Integration Message Broker V5.0. It is possible for a migrated Configuration Manager to manage a heterogeneous set of brokers.

The WebSphere Message Broker Information Center’s topic “Coexistence” in the path **Migrating → Coexistence** documents which WebSphere Message Broker V6.0 components can coexist with components from previous versions. It also provides links to topics explaining the conditions under which the components of previous versions can participate in the WebSphere Message Broker V6.0 domain.

This information is available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23930_.htm

Important: We assume, as described in the installation guide, that the WebSphere Message Broker V6.0 Configuration Manager component has been installed on a machine which has the WebSphere Business Integration Message Broker V5.0 Configuration Manager, but in a different location.

The installation guide is available on the Web:

http://www.elink.ibmlink.ibm.com/public/applications/publications/cgibin/pbi.cgi?¶¶CTY=US&FNC=SRX&PBL=GC34-6621-00

Follow these steps to migrate the Configuration Manager in situ:

1. If the WebSphere Message Broker V6.0 Toolkit is open, disconnect the Configuration Manager from the domain. In the Broker Administration perspective's Domains view, right-click the **Configuration Manager** connection and select **Disconnect**.
2. In the Windows command prompt, stop the Configuration Manager by entering the following command:

   ```
   mqsistop ConfigMgr
   ```

3. Start the Event Viewer by selecting **Start** → **Programs** → **Administrative Tools** → **Event Viewer Application Log** from the menu.

   When the Configuration Manager stops, a warning event message is displayed with source identifier BIPV500I and event identifier 2002.

   Important: We recommend that you stop the Configuration Manager before attempting the migration process.

4. Open a WebSphere Message Broker V6.0 Command Console and select **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **Command Console**. Enter the following command:

   ```
   mqsilist -a
   ```

   This lists all components that have been created on the system.

   Prior to migration, the Configuration Manager entry indicates that it is a Version 5 component. In the environment demonstrating the migration, the Configuration Manager and Windows broker get configured in the system, as illustrated in Figure 21-1 on page 389 and Example 21-1 on page 391.

Example 21-1 Output from mqsilist -a prior to the Configuration Manager migration

```
C:\Program Files\IBM\MQSI\6.0>mqsilist -a
BIP8221I: Broker: BKW1 (Version 5)  -  BKW1QM
BIP8221I: ConfigMgr: ConfigMgr (Version 5)  - CMQM

BIP8071I: Successful command completion.
```

5. Before migrating the Configuration Manager, perform a premigration check to determine whether the component can be safely migrated by entering the following command on the Command Console:

   ```
   mqsimigratecomponents ConfigMgr -c
   ```

   On successful completion of the premigration check, an output similar to that shown in Example 21-2 is displayed.

Example 21-2 Output from a premigration check of the Configuration Manager

```
C:\Program Files\IBM\MQSI\6.0>mqsimigratecomponents ConfigMgr -c
BIP8674I: Configuration Manager 'ConfigMgr' (Version 5.0) specified for
migration.
BIP8680I: Pre-migration check succeeded.
BIP8071I: Successful command completion.
```

Important: The premigration check does not verify whether the Configuration Manager's database can be accessed.

Therefore, even if a premigration check is successful, the migration may fail if the data source user ID and the password combination is incorrect.

6. To migrate the Configuration Manager to WebSphere Message Broker V6.0, enter the following command on the Command Console:

   ```
   mqsimigratecomponents ConfigMgr
   ```

   This causes the Configuration Manager's queues, registry, and file system information to migrate from WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0. The output from a successful in situ Configuration Manager migration is shown in Example 21-3.

Example 21-3 Output from a successful in situ Configuration Manager migration

```
C:\Program Files\IBM\MQSI\6.0>mqsimigratecomponents ConfigMgr
BIP8674I: Configuration Manager 'ConfigMgr' (Version 5.0) specified for
migration.
BIP8755I: Copied value 'QueueManagerName' into the new location
BIP8755I: Copied value 'DataBaseName' into the new location
BIP8755I: Copied value 'NTDomainAware' into the new location
BIP8755I: Copied value 'DataBaseUserId' into the new location
```

```
BIP8755I: Copied value 'DataBasePassword' into the new location
BIP8755I: Copied value 'JDBCDriverName' into the new location
BIP8755I: Copied value 'JDBCConnectionName' into the new location
BIP8755I: Copied value 'NTSecurityDomainName' into the new location
BIP8755I: Copied value 'UserNameServerQueueManagerName' into the new location
BIP8755I: Copied value 'AdminAgentPID' into the new location
BIP8763I: Deleted value 'QueueManagerName' from the old location
BIP8763I: Deleted value 'DataBaseName' from the old location
BIP8763I: Deleted value 'NTDomainAware' from the old location
BIP8763I: Deleted value 'DataBaseUserId' from the old location
BIP8763I: Deleted value 'DataBasePassword' from the old location
BIP8763I: Deleted value 'JDBCDriverName' from the old location
BIP8763I: Deleted value 'JDBCConnectionName' from the old location
BIP8763I: Deleted value 'NTSecurityDomainName' from the old location
BIP8763I: Deleted value 'UserNameServerQueueManagerName' from the old location
BIP8763I: Deleted value 'AdminAgentPID' from the old location
BIP8768I: Finished registry migration for component 'ConfigMgr'.
BIP8654I: Moving filesystem artefacts from 'C:\Documents and Settings\All
Users\Application Data\IBM\MQSI' to 'C:\Documents and Settings\All
Users\Application Data\IBM\MQSI'
BIP8071I: Successful command completion.
```

7. After migrating the Configuration Manager, carry out a post-migration check to ensure that the migration completed successfully and to confirm that correct queues exist and the registry is in the correct format.

   To carry out a post-migration check of the Configuration Manager, enter the following command on the Command Console:

   ```
   mqsimigratecomponents ConfigMgr -v
   ```

   The output from a successful post-migration check is shown in Example 21-4.

Example 21-4 Output from a post-migration check of the Configuration Manager

```
C:\Program Files\IBM\MQSI\6.0>mqsimigratecomponents ConfigMgr -v
BIP8674I: Configuration Manager 'ConfigMgr' (Version 6.0) specified for
migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

8. To complete the migration process, start the Configuration Manager by entering the following command on the Command Console:

   ```
   mqsistart ConfigMgr
   ```

   When you start the Configuration Manager for the first time after completing the migration process, it detects the domain information held in the DB2 Universal Database of the WebSphere Business Integration Message Broker

V5.0 Configuration Manager and migrates it to the internal repository of the WebSphere Message Broker V6.0 Configuration Manager.

In this way, WebSphere Message Broker V6.0 Configuration Manager thus controls the same broker domain configuration that WebSphere Business Integration Message Broker V5.0 Configuration Manager used to control.

Note: When you start the Configuration Manager for the first time after migration, the configuration data held in the database is copied into the internal repository. As a result, Configuration Manager takes longer than usual to start.

21.1.2 Validating the Configuration Manager migration

This section describes several methods for validating Configuration Manager migration.

Application Log messages

Start the Event Viewer by selecting **Start** → **Programs** → **Administrative Tools** → **Event Viewer** → **Application Log**.

When Configuration Manager starts successfully for the first time after migration, seven information event messages are logged, as illustrated in Figure 21-2.

Type	Date	Time	Source	Category	Event
Information	14/10/2005	10:20:16	WebSphere Broker v6000	None	1003
Information	14/10/2005	10:20:14	WebSphere Broker v6000	None	8280
Information	14/10/2005	10:20:14	WebSphere Broker v6000	None	8255
Information	14/10/2005	10:17:56	WebSphere Broker v6000	None	1228
Information	14/10/2005	10:17:34	WebSphere Broker v6000	None	1224
Information	14/10/2005	10:17:34	WebSphere Broker v6000	None	1223
Information	14/10/2005	10:17:29	WebSphere Broker v6000	None	2001

Figure 21-2 Event messages logged after migration

Subsequent successful Configuration Manager starts log only four information event messages: event identifiers 2001, 8255, 8280, 1003.

Note: The source identifier of a WebSphere Message Broker V6.0 component event message is WebSphere Broker v6000 (not BIPV500).

BIPV500 identifies a WebSphere Business Integration Message Broker V5.0 component event message.

Troubleshooting

If a problem occurs when you start Configuration Manager for the first time after migration, error event messages are logged in the system's local error log. For details about some common issues you may encounter while migrating resources, refer to the topic "Problems when importing or migrating resources" by selecting the path **Troubleshooting and support → Dealing with problems → Problems when importing or migrating resources** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/au16660_.htm

The issues encountered may depend on the size and complexity of the domain. You may encounter errors as a result of a scenario having thousands of persistent subscriptions in each of the brokers.

As an example, the Windows Application Log shown in Figure 21-3 demonstrates the result of the WebSphere Business Integration Message Broker V5.0 Configuration Manager's DB2 Universal Database parameter, which required tuning to deal with the size of the domain to be migrated.

Type	Date	Time	Source	Category	Event
Error	14/10/2005	10:14:55	WebSphere Broker v6000	None	1007
Error	14/10/2005	10:14:55	WebSphere Broker v6000	None	1205
Information	14/10/2005	10:14:38	WebSphere Broker v6000	None	1228
Information	14/10/2005	10:14:18	WebSphere Broker v6000	None	1224
Information	14/10/2005	10:14:18	WebSphere Broker v6000	None	1223
Information	14/10/2005	10:14:12	WebSphere Broker v6000	None	2001

Figure 21-3 Configuration Manager failed to migrate successfully due to a DB2 tuning issue

Error event message 1205 indicates the need to tune the DB2 Universal Database APP_CTL_HEAP_SZ parameter, as illustrated in Figure 21-4.

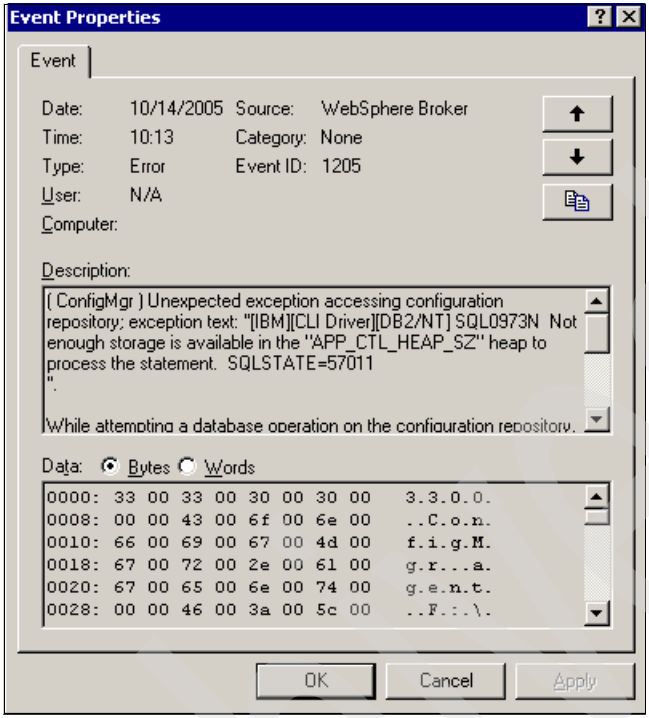


Figure 21-4 Error event 1205: need to tune the DB2 APP_CTL_HEAP_SZ parameter

This set of event messages is repeated until the Configuration Manager is stopped. The repetition occurs because the error encountered is an unexpected exception and the Configuration Manager is designed to retry the operation after a short delay, as explained in error event message 1007.

This issue arises because of the size of the broker domain that Configuration Manager manages. To address the problem, carry out the following tasks:

1. Stop the Configuration Manager by entering the following command on the Command Console:

   ```
   mqsistop ConfigMgr
   ```

   When the Configuration Manager has stopped, a warning event message is shown in the system's local error log, displaying source identifier WebSphere Broker v6000 and event identifier 2002.

2. At the DB2 Universal Database command prompt, do the following:

   a. Connect to the Configuration Manager database by entering the following command:

      ```
      db2 connect to cm_db user db_userid using db_password
      ```

b. Discover the current value of the APP_CTL_HEAP_SZ parameter for the Configuration Manager database by entering the following command:

```
db2 get db cfg for cm_db | find "APP_CTL_HEAP_SZ"
```

Figure 21-5 illustrates the output.

Example 21-5 Output from the database configuration command

```
Max appl. control heap size (4KB)     (APP_CTL_HEAP_SZ) = 128
```

c. Increase the value of the APP_CTL_HEAP_SZ parameter for the Configuration Manager database by entering the following command:

```
db2 update db cfg for cm_db using APP_CTL_HEAP_SZ higher_value
```

d. Enter the following command to force all applications to disconnect from the database and allow the changes to become effective:

```
db2 force application all
```

e. Stop and restart the DB2 Universal Database, to enable the changes to become effective immediately, by entering the following commands:

- `db2stop`
- `db2start`

3. Start the Configuration Manager by entering the following command on the Command Console:

```
mqsistart ConfigMgr
```

When the Configuration Manager starts, a series of seven information event messages are present in the system's local error log. They contain the source identifier WebSphere Broker v6000 and event identifiers 2001, 1223, 1224, 1228, 8255, 8280, and 1003 (see Figure 21-2 on page 393).

If the system's error log still repeatedly contains the set of six event identifiers 2001, 1223, 1224, 1228, 1205, and 1007 (shown in Figure 21-3 on page 394), then increase the value assigned to APP_CTL_HEAP_SZ and repeat step 1 to step 3.

Listing the components

List all the components that are on the system by entering the **`mqsilist -a`** command on the Command Console.

After migration the Configuration Manager entry does not display a version, thereby indicating that it is at the latest level (WebSphere Message Broker V6.0). In the environment being migrated, the Configuration Manager and a Windows broker reside on the same system.

The output shown in Example 21-6 illustrates that although the Configuration Manager has been migrated, the broker is unaffected and remains in WebSphere Business Integration Message Broker V5.0 as it did before the Configuration Manager migration (as shown in Example 21-1 on page 391).

Example 21-6 Output from mqsilist -a following the Configuration Manager migration

```
C:\Program Files\IBM\MQSI\6.0>mqsilist -a
BIP8221I: Broker: BKW1 (Version 5)  -  BKW1QM
BIP8099I: ConfigMgr: ConfigMgr  -  CMQM

BIP8071I: Successful command completion.
```

Connecting to the domain in the Message Broker V6.0 Toolkit

Start the WebSphere Message Broker V6.0 Toolkit by selecting **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **WebSphere Message Brokers Toolkit** and open the Broker Administration perspective.

In the Domains view, right-click the **Configuration Manager** connection and select **Connect**, as illustrated in Figure 21-5.

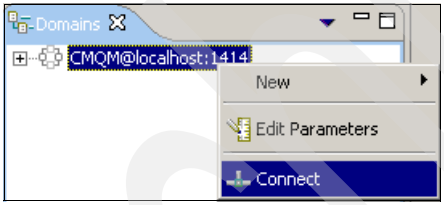


Figure 21-5 Connecting to an existing domain connection in the Domains view

An information pop-up, illustrated in Figure 21-6, appears while trying to establish connection for the first time.

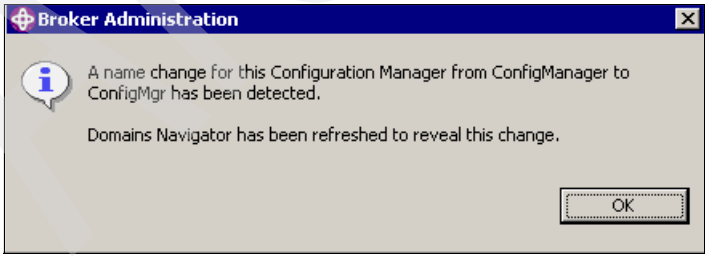


Figure 21-6 Information pop-up during initial connection to the migrated Configuration Manager

Only a single Configuration Manager named *ConfigMgr* was allowed in WebSphere Business Integration Message Broker V5.0, so the Domains Properties view did not display a Configuration Manager name property until now.

However, because WebSphere Message Broker V6.0 allows there to be multiple named Configuration Managers on a system, the name property of the Configuration Manager is now relevant enough to be displayed.

Note: If your previous configuration relied on the MCA user ID being set on the SYSTEM.BRK.CONFIG WebSphere MQ channel to the Configuration Manager, you may face problems when connecting to the migrated Configuration Manager from WebSphere Message Broker Toolkit V6.0.

The Version 6.0 Toolkit now passes additional information to Configuration Manager regarding the user and the machine domain that is used to run the Toolkit.

To resolve this problem, configure additional security in the form of Access Control List (ACL) entries on the Configuration Manager system for user IDs running the Toolkit. For details about how to configure ACL entries, refer to 21.3.4, “Access Control Lists” on page 423.

If the broker topology is expanded as illustrated in Figure 21-7, the domain of the brokers previously controlled by the WebSphere Business Integration Message Broker V5.0 Configuration Manager are displayed in the state they were in when Configuration Manager was stopped prior to the migration.

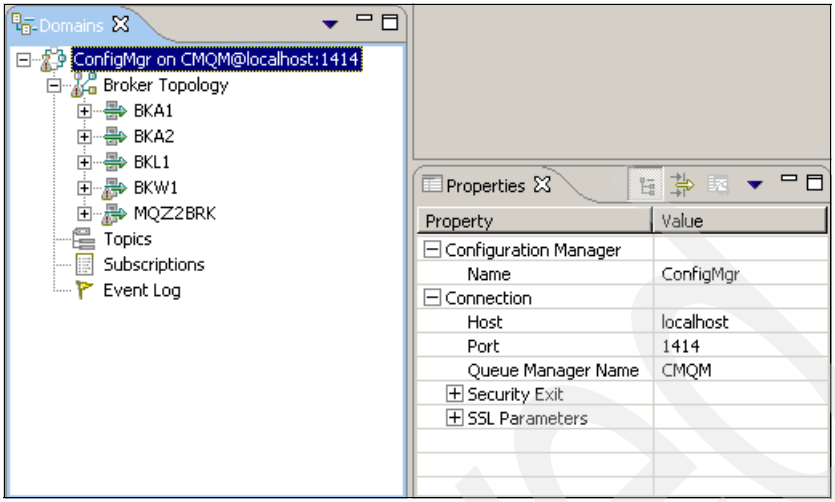


Figure 21-7 Expanded Broker Topology

To confirm that persistent subscriptions are retained after the migration, double-click **Subscriptions** in the Domains view. In the Subscriptions editor that opens, click **Query** as indicated in Figure 21-8.

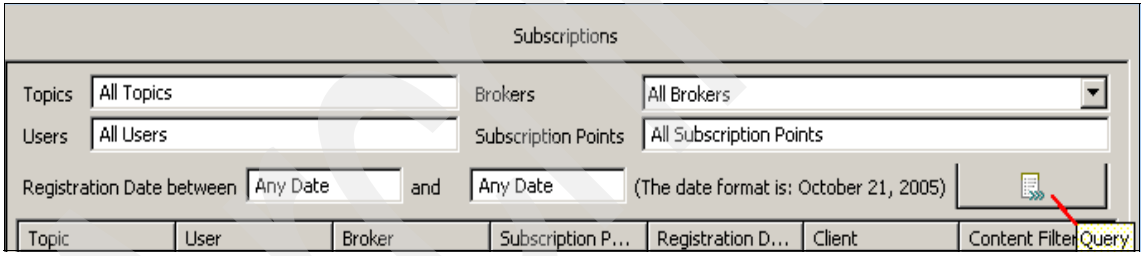


Figure 21-8 The Subscriptions editor

Note: In a domain with a large number of active subscriptions, filter the Subscriptions by topic, user, or broker categories before clicking **Query**.

Manipulating the domain

After connecting to the Configuration Manager in the WebSphere Message Broker Toolkit, you can administer the domain from the Broker Administration perspective—for example, to deploy broker archive files, and use the Domains view to start and stop message flows.

Notes:

- Both WebSphere Business Integration Message Broker V5.0 and WebSphere Message Broker V6.0 broker archive files can be deployed to WebSphere MQ Integrator V2.1, WebSphere Business Integration Message Broker V5.0, and WebSphere Message Broker V6.0 brokers as long as the functionality deployed is applicable to the runtime.

  For example, while developing new resources in the WebSphere Message Broker V6.0 Toolkit, ensure that any new functionality that is available only in WebSphere Message Broker V6.0 is *not* deployed to Message Broker V5.0 or WebSphere MQ Integrator V2.1 brokers.

- When new broker archive files are created in WebSphere Message Broker V6.0 from existing broker resources, validate the runtime results to ensure that the migrated resources are working as expected.

There are also a series of new and enhanced command line commands that offer an alternative way to administer a domain. The results of these commands are reflected in the Message Broker Toolkit Domains view, Alerts view, and the Event Log, along with the BIP2056I and BIP4040I successful response messages.

The following inserts are used in the examples of the commands:

<cn_host>	This is the IP address or host name of the Configuration Manager.
<cm_port>	This is the port number of the Configuration Manager.
<cm_qm>	This is the Queue Manager of the Configuration Manager.
<broker>	This is the name of the broker to deploy to.
<exgrp>	This is the name of the execution group to deploy to.
<bar>	This is the location (relative or absolute reference) and name of the broker archive file to be deployed.
<deployed_objects>	This is a colon-separated list of the deployed objects to be removed from the execution group.
<msgflow>	This is the name of the message flow to be stopped.

Examples of the commands used for administering the domain from the command line include the following:

- You can perform an incremental deployment of a broker archive file to an execution group by entering the following command on the Command Console:

  ```
  mqsideploy -i cm_host -p cm_port -q cm_qm -b broker -e exgrp -a bar
  ```

  Example 21-7 demonstrates how to deploy the test.bar broker archive file from the workspace directory to the default execution group of the Windows Broker BKW1, and the response messages.

Example 21-7 Deploying a broker archive file using the mqsideploy command

```
mqsideploy -i localhost -p 1414 -q CMQM -b BKW1 -e default -a "C:\Documents
and Settings\TstAdmin\IBM\wmbt6.0\workspace\Servers\test.bar"
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1039I: Deploying BAR file 'C:\Documents and
Settings\TstAdmin\IBM\wmbt6.0\workspace\Servers\test.bar' to broker 'BKW1
' (execution group 'default') ...
BIP1092I: Broker BKW1 successfully processed the deployment request.
```

- You can stop a message flow in an execution group by entering the following command on the Command Console:

  ```
  mqsistopmsgflow -i cm_host -p cm_port -q cm_qm -b broker -e exgrp -m
  msgflow
  ```

  Example 21-8 demonstrates how to stop the test message flow in the default execution group of the Windows Broker BKW1, and the response messages.

Example 21-8 Stopping a message flow in an execution group using the mqsistopmsgflow command

```
mqsistopmsgflow -i localhost -p 1414 -q CMQM -b BKW1 -e default -m test
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1028I: Submitting request to the Configuration Manager to stop
'BKW1/default/test'...
BIP1029I: Waiting up to 60 second(s) for broker BKW1 to update its
configuration...
BIP1031I: Message flow 'BKW1/default/test' is reported as stopped.
BIP8071I: Successful command completion.
```

- You can remove a message flow or message set from an execution group by entering the following command on the Command Console:

  ```
  mqsideploy -i cm_host -p cm_port -q cm_qm -b broker -e exgrp -d
  deployed_objects
  ```

Example 21-9 demonstrates how to remove the test message flow and the video message set from the default execution group of the Windows Broker BKW1, and the response messages.

Example 21-9 Removing a message flow and message set from an execution group using the mqsideploy command

```
mqsideploy -i localhost -p 1414 -q CMQM -b BKW1 -e default -d
test.cmf:Video.dictionary
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1062I: Removing the following objects from execution group 'default':
          test.cmf Video.dictionary
BIP1092I: Broker BKW1 successfully processed the deployment request.
```

Restriction: If a message flow is running in two brokers in the same named execution group and you right-click the message flow in the Message Broker Toolkit, and then select **Stop** from the context menu, the message flow in the other broker may stop. The same is true for the Start command.

To avoid this problem, use the following commands:

```
mqsistopmsgflow
mqsistartmsgflow
```

Post-migration issues

Following migration, WebSphere Business Integration Message Broker V5.0 Configuration Manager's DB2 Universal Database remains unmodified.

WebSphere Message Broker V6.0 Configuration Manager uses an internal repository *instead* of a database—so following a successful validation of the migrated Configuration Manager, you can remove the database if desired.

Should you need to restore the WebSphere Business Integration Message Broker V5.0 environment, you can restore the database if you followed the backup procedures described in Chapter 19, "Backing up the WebSphere Business Integration Message Broker V5.0 domain" on page 339.

However, any changes made to a migrated domain will not be reflected in the restored database, because it is not possible to roll back a changed domain to the DB2 Universal Database™.

Important: Before removing a Configuration Manager database, confirm that the Configuration Manager has started successfully after migration, because this is when the configuration data held in the database is copied into the internal repository.

21.2 Changing the Configuration Manager platform to Advanced Interactive Executive (AIX)

In WebSphere Message Broker V6.0, the Configuration Manager runtime component is supported on the same platforms as the broker runtime. Refer to the WebSphere Message Broker system requirements for details about the environment needed:

http://www-306.ibm.com/software/integration/wbimessagebroker/requirements/index.html

Therefore, you may wish to move the Configuration Manager's platform, as well as migrate it to WebSphere Message Broker V6.0.

21.2.1 Moving the Configuration Manager's platform after migrating to WebSphere Message Broker V6.0

The two-step process of migrating the Configuration Manager on the same system and then moving the migrated Configuration Manager to another system is documented in the topic "Migrating a Configuration Manager to a different computer that does not have DB2 installed". To access this topic, select the path **Migrating → Migrating from Version 5.0 products → Migrating from WebSphere Business Integration Message Broker Version 5.0 → Migrating a Configuration Manager → To a different computer that does not have DB2 installed** in the WebSPhere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23823_.htm

The steps to follow when moving a Configuration Manager to a new queue manager are detailed in the topic "Moving the Configuration Manager to a new queue manager". To access this topic, select the path **Configuring the broker domain → Configuring broker domain components → Modifying a Configuration Manager → Moving the Configuration Manager to a new queue manager** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae20240_.htm

In 21.1.1, "Migrating the Configuration Manager in situ" on page 389, we describe how to migrate the Configuration Manager in situ. Here, we describe how to move the Configuration Manager from a Windows system to a new queue

manager on the AIX system. Figure 21-9 illustrates how the environment looks after completing the Configuration Manager's move to the AIX system.

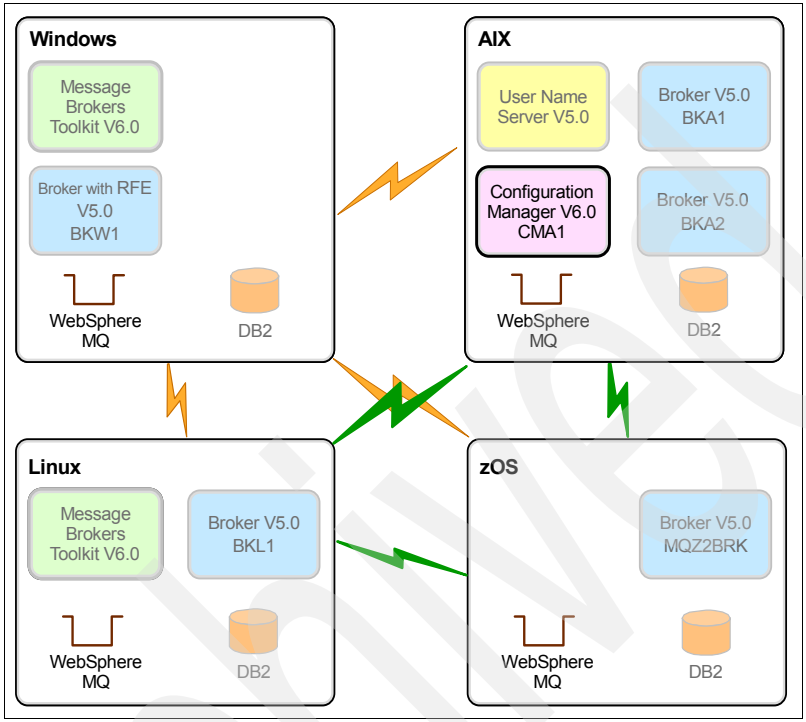


Figure 21-9 Environment after moving the Configuration Manager from Windows to AIX

Begin this process as follows.

1. If any Message Broker Toolkits are open, disconnect from the domain:

   In the Broker Administration perspective's Domains view, right-click the **Configuration Manager** connection and select **Disconnect**.

2. If possible, stop all brokers in the domain by issuing the **`mqsistop`** command.

   Attention: If it is not possible to stop the brokers in the domain because it is a high availability system, carry out step 11 on page 410.

3. On the Windows system, open a WebSphere Message Broker V6.0 Command Console by selecting **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **Command Console**. Next, stop the Configuration Manager by entering the following command:

   ```
   mqsistop cm_name
   ```

Start the Event Viewer by selecting **Start** → **Programs** → **Administrative Tools** → **Event Viewer** → **Application Log**. When the Configuration Manager stops successfully, a warning event message is displayed with source identifier WebSphere Broker v6000 and event identifier 2002.

Important: It is essential that the Configuration Manager be stopped *before* you attempt to back up the Configuration Manager's internal repository.

4. To create a backup archive of the Configuration Manager's internal repository, enter the following command in the Command Console on the Windows system:

   ```
   mqsibackupconfigmgr cm_name -d directory -a backup_filename
   ```

   Note: The directory you specify for the **-d** command option (to write the backup archive to) must exist *before* the command is issued.

   The output from a successful Configuration Manager repository backup is shown in Example 21-10.

Example 21-10 Output from the Configuration Manager backup command

```
C:\Program Files\IBM\MQSI\6.0>mqsibackupconfigmgr ConfigMgr -d C:\backup -a
ConfigMgr_backup
BIP1075I: Creating backup archive for Configuration Manager 'ConfigMgr' in
directory 'C:\backup' ...
BIP1017I: A backup archive called 'ConfigMgr_backup.zip' was created
successfully.
BIP8071I: Successful command completion.
```

5. Copy the Configuration Manager backup archive file that was created into an appropriate directory on the AIX system. In the migration being demonstrated, a ConfigMgr_backup.zip file is created in the C:\backup directory. It is then moved, using binary FTP, to the AIX system and stored in a directory named "backup" in your home directory.
6. On the AIX system, run the appropriate **mqsiprofile** command to set the WebSphere Message Broker V6.0 environment. Create a named WebSphere Message Broker V6.0 Configuration Manager by entering the following command:

   ```
   mqsicreateconfigmgr cm_name -i userid -a password -q cm_qm
   ```

Notes:

- You can give a new Configuration Manager the same name as the Configuration Manager being replaced. However, a Configuration Manager *cannot* have the same name as a broker or another Configuration Manager on the same system.
- As for the queue manager used for the new Configuration Manager—it is important to maintain uniqueness in the WebSphere MQ network for communication among the domain components, so use a different queue manager name from the Configuration Manager being moved.

The output from the successful creation of a WebSphere Message Broker V6.0 Configuration Manager, with the User Name Server's queue manager specified, is shown in Example 21-11.

Example 21-11 Output from the WebSphere Message Broker V6.0 Configuration Manager creation

```
$ mqsicreateconfigmgr CMA1 -i <userid> -a <password> -q CMA1QM -s UNSQM
WebSphere MQ queue manager created.
Creating or replacing default objects for CMA1QM.
Default objects statistics : 31 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
WebSphere MQ queue manager 'CMA1QM' started.
The setmqaut command completed successfully.
The setmqaut command completed successfully.
The setmqaut command completed successfully.
The setmqaut command completed successfully.
The setmqaut command completed successfully.
The setmqaut command completed successfully.
The setmqaut command completed successfully.
BIP8071I: Successful command completion.
```

Note: On z/OS, the Configuration Manager is created by running the BIPCRCM job. Refer to "Creating a Configuration Manager on z/OS" on page 200 in Chapter 11, "Migrating the Configuration Manager" on page 177 for detailed instructions.

Be aware that it is possible to use a queue manager that was created previously, and to share a queue manager with a broker. However, using the command as shown in Example 21-11, the new queue manager specified is created during the creation of the Configuration Manager.

A WebSphere MQ listener must be created and started for the Configuration Manager. In the environment being demonstrated, the Configuration Manager's WebSphere MQ listener is running on port 1414.

Confirm the successful creation of the component by entering a **mqsilist -a** command to list all components created on the system. In the environment demonstrating the migration, the system has two brokers and a User Name Server in Message Broker V5.0. Following the Configuration Manager creation, there is also a Configuration Manager called CMA1, as illustrated in Example 21-12.

Example 21-12 Output from mqsilist -a, following Configuration Manager creation on the AIX system

```
$ mqsilist -a
BIP8099I: Broker: BKA1 (Version 5) -  BKA1QM
BIP8221I: Broker: BKA2 (Version 5)  -  BKA2QM
BIP8099I: UserNameServer: UserNameServer (Version 5) -  UNSQM
BIP8099I: ConfigMgr: CMA1  -  CMA1QM

BIP8071I: Successful command completion.
```

7. On the AIX system, configure the WebSphere MQ to allow the new WebSphere Message Broker V6.0 Configuration Manager to communicate with the broker domain.

   Then, on each of the broker systems, configure the broker's queue manager to communicate with the new Configuration Manager. This involves configuring the appropriate channels and transmitting queue definitions. Start the channels to confirm successful communication among the WebSphere MQ components.

   Important: The Configuration Manager must be stopped before you restore the backup file into the repository.

8. On the AIX system, enter the following command to overwrite the contents of the empty repository of the newly created Configuration Manager with the Windows Configuration Manager repository backup:

   ```
   mqsirestoreconfigmgr cm_name -d directory -a backup_filename
   ```

Note: A number of new commands are available in the WebSphere Message Broker V6.0 Configuration Manager installation. These include:

- `mqsibackupconfigmgr`
- `mqsirestoreconfigmgr`
- `mqsistartmsgflow`
- `mqsistopmsgflow`

To execute the new commands on UNIX platforms, add the shipped Java 1.4.2 JRE to the PATH environment variable in the mqsiprofile file:

install_directory/jre/bin

The output from the successful restoration of the Configuration Manager repository is shown in Example 21-13.

Example 21-13 Output from the Configuration Manager restore command

```
$ mqsirestoreconfigmgr CMA1 -d /home/<userid>/backup -a ConfigMgr_backup
BIP1079I: Replacing repository for 'CMA1' with archive
'/home/<userid>/backup/ConfigMgr_backup.zip'...
BIP1171I: Verifying the restored repository...
BIP8071I: Successful command completion.
```

Note: On z/OS, the Configuration Manager is restored from a backup by running the BIPRSCM job.

For detailed instructions, refer to step 3 on page 210 in Chapter 11, “Migrating the Configuration Manager” on page 177.

9. On the AIX system, start the Configuration Manager by entering the following command:

   `mqsistart` *cm_name*

   Check the system log to verify that Configuration Manager started successfully; the final entry in the system log should be BIP1003I, confirming that Configuration Manager is available for use.

Important: Because a broker can still only be managed by a single Configuration Manager, the Configuration Manager on the system you have moved from should be changed to *not* start automatically.

After verifying that the moved Configuration Manager is working correctly, at your discretion you can delete the Configuration Manager on the system you have moved from by using the **`mqsideleteconfigmgr`** command.

Note: After the next step is completed, all brokers in the domain will be managed by the new Configuration Manager on the new system.

10. In order to associate all brokers in the domain with the new Configuration Manager, which will enable the brokers to publish their status messages to the correct queue manager, a complete topology deployment is required.

    On the AIX system, enter the following command:

    ```
    mqsideploy -i cm_host -p port -q cm_qm -l -m
    ```

    Important: WebSphere Message Broker uses Access Control Lists (ACLs) to govern which users and groups can manipulate objects within the Configuration Manager and the Message Broker Toolkit. This is discussed in 21.3.4, “Access Control Lists” on page 423 and documented in the topic ACL permissions in the WebSphere Message Broker Information Center. The topic can be accessed by selecting the path **Reference** → **Security** → **ACL permissions** on the Web at:

    http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ap12520_.htm

    When you create or start a Configuration Manager, an entry is made in the ACL to grant that user ID full authority on any machine. However, there is currently a problem (which will be addressed in a future fix pack) with correct permission being granted to the user ID for a full topology deployment if there are existing broker or execution group entries. The following error message appears in the system log *BIP1780E: Unable to validate specified user*.

    If you encounter this problem, enter the following command to grant the user ID explicit topology authority on the system:

    ```
    mqsicreateaclentry cm_name -u userID -m machine_name -x F -t
    ```

    The output from a successful topology deployment is shown in Example 21-14.

Example 21-14 Output from the topology deployment command

```
$ mqsideploy -i localhost -p 1414 -q CMA1QM -l -m
BIP1044I: Connecting to the Configuration Manager's queue manager...
BIP1045I: Connecting to the Configuration Manager...
BIP1055I: Deploying pub/sub neighbors (topology) configuration...
BIP1099I: Deployment completed successfully.

All brokers affected by the deployment request have successfully processed the
message and have updated their configurations accordingly.
```

Attention: If the brokers in the domain were stopped at step 2 on page 404, start them again by using the **mqsistart** command as soon as the topology deploy is initiated, in order to allow the deployments to be processed.

11. If you could not stop the brokers in the domain (at step 2 on page 404, by entering the **mqsistop** command), it means that the brokers may have sent status messages to the original Configuration Manager.

    In such a case, any status change events that occurred between the time the original Configuration Manager was stopped (step 3 on page 404) and the time complete deployment took place (step 8 on page 407) are sent to the original Configuration Manager's queue manager's SYSTEM.BROKER.ADMIN.REPLY queue.

    Therefore, check the original Configuration Manager's queue manager's SYSTEM.BROKER.ADMIN.REPLY queue. If there are messages in this queue, transfer them manually to the new Configuration Manager's queue manager's SYSTEM.BROKER.ADMIN.REPLY queue, even while maintaining the headers.

    You can complete this manual transfer by using a utility such the SupportPac IH03 or the SupportPac MO71. For details about these utilities, refer to the following Web sites:

    http://www-1.ibm.com/support/docview.wss?rs=171&uid=swg24000637&loc=en_US&cs=utf-8&lang=en
    http://www-1.ibm.com/support/docview.wss?rs=171&uid=swg24000142&loc=en_US&cs=utf-8&lang=en

21.2.2 Verifying the Configuration Manager that has been moved

After moving the Configuration Manager to another platform, it is important that you connect to it through the WebSphere Message Broker Toolkit to verify that the component is running correctly.

To connect to a Configuration Manager that is not on the same system as the WebSphere Message Broker Toolkit, authorize the user ID of the WebSphere Message Broker Toolkit user on the remote system, as follows:

1. Create a user on the remote system with the same user ID used to run the Toolkit.

Note: Creating a user on the remote system *before* adding the ACL entry means that the **mqsicreateaclentry** command created the appropriate WebSphere MQ authorities.

2. Create an appropriate ACL entry for the Message Broker Toolkit user ID on the remote Configuration Manager system by entering the **mqsicreateaclentry** command.

   For example, to grant full authority to all the resources for the Message Broker Toolkit user ID cz50365 from the machine PC2197 on the Configuration Manager CMA1, enter the following command:

   ```
   mqsicreateaclentry CMA1 -u cz50365 -m PC2197 -x F -p
   ```

Now, because you created a new Configuration Manager on a different system, you must either change the existing domain connection or create a new domain connection. We recommend that you create a new domain connection and delete the old one after the original Configuration Manager is deleted.

To create a new domain connection, follow these steps:

1. In the Message Broker Toolkit, open the Broker Administration perspective and go to the Domains view. Right-click and select **New** → **Domain**.

   In the dialog box that appears, type the necessary information in the Configuration Manager's queue manager Name, Host, and Port fields, as illustrated in Figure 21-10. Then click **Next**.

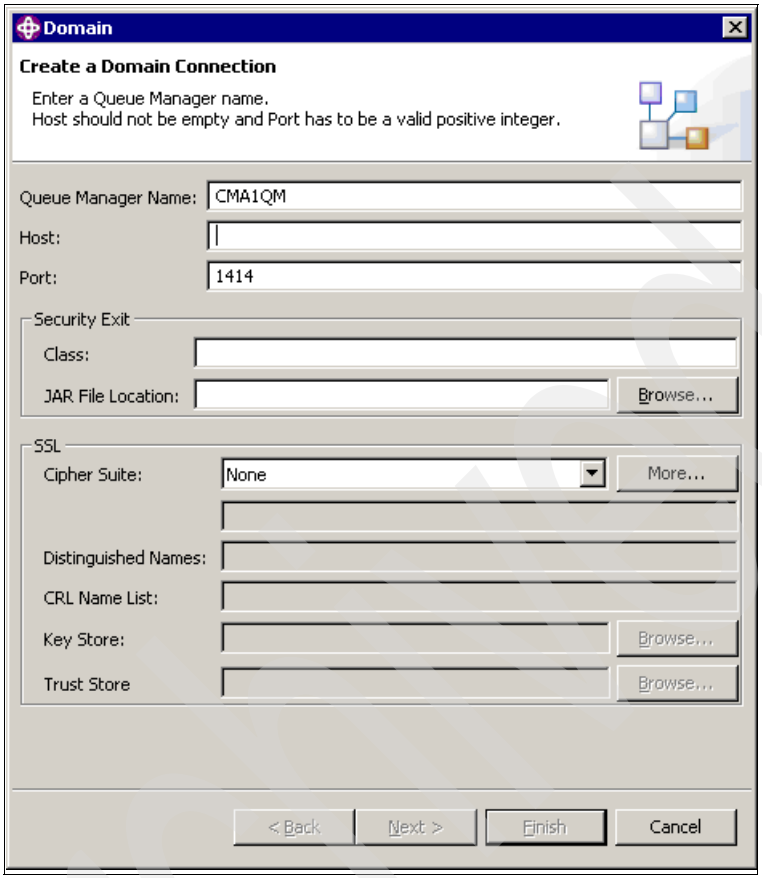


Figure 21-10 Creating a new domain connection to the moved Configuration Manager

2. After the Configuration Manager connection is established, another dialog box appears. Type the Connection Name and click **Finish**. The Domains view shows the new connection to the Configuration Manager, named CMA1 on the AIX machine.

   If the broker topology is expanded (see Figure 21-11), the domain of the brokers previously controlled by the Windows Configuration Manager *ConfigMgr* is now shown in the state it was in prior to the Windows Configuration Manager being stopped.

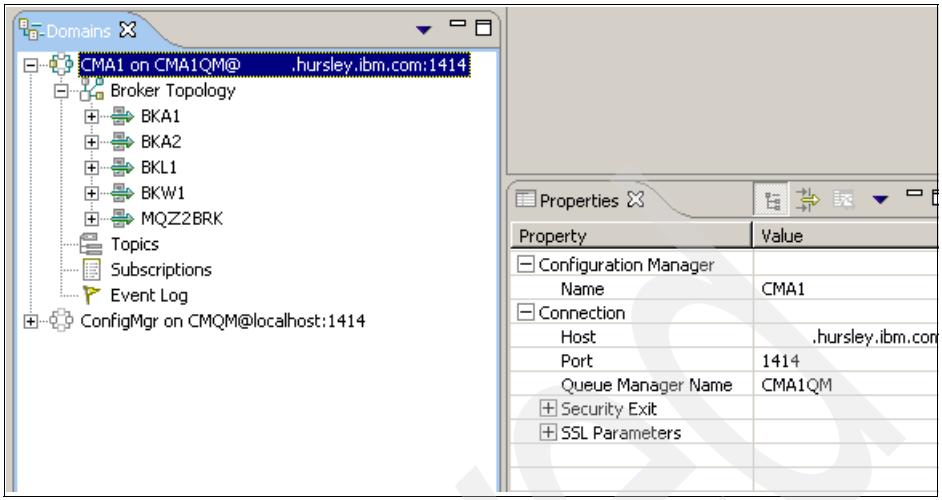


Figure 21-11 Moved Configuration Manager's broker domain in Message Brokers Toolkit

Troubleshooting

When connecting to Configuration Manager, you may encounter some of the following problems:

- If the Message Brokers Toolkit user ID has not been created on the Configuration Manager system and a user tries to connect to the Configuration Manager, error BIP0915E appears, as shown in Figure 21-12.

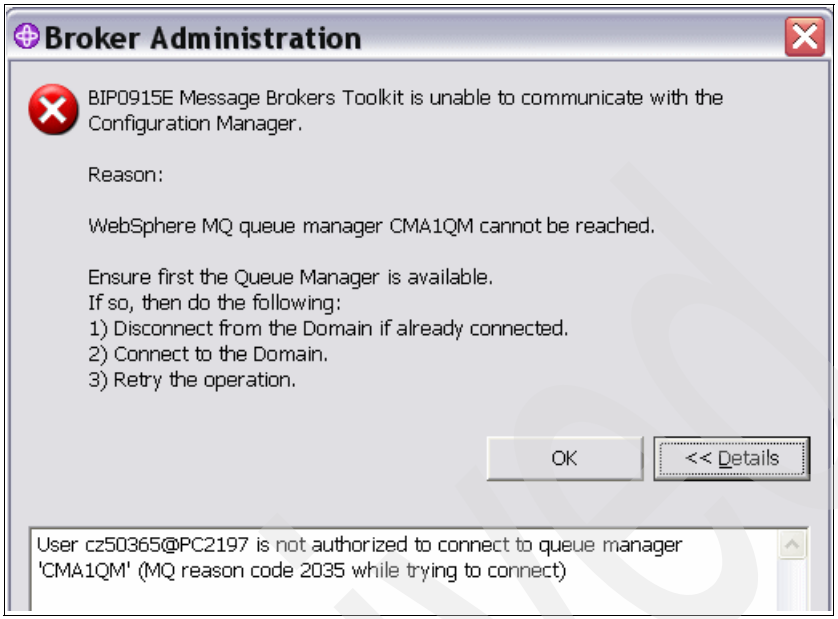


Figure 21-12 Error: Toolkit user ID does not exist on the Configuration Manager system

- If the Message Brokers Toolkit user ID does not have an appropriate ACL entry on the Configuration Manager system and a user tries to connect to the Configuration Manager, error BIP0889E appears, as shown in Figure 21-13.

Figure 21-13 Error: Toolkit user ID without an appropriate ACL on the Configuration Manager

21.2.3 Migrating the Configuration Manager and moving to another platform in one step

The alternative process of migrating and moving a Configuration Manager in a single step is documented in the topic “Migrating a Configuration Manager to a different computer that has DB2 installed” in the WebSphere Message Broker Information Center. This information can be accessed by selecting the path **Migrating → Migrating from Version 5.0 products → Migrating from WebSphere Business Integration Message Broker Version 5.0 → Migrating a Configuration Manager → To a different computer that has DB2 installed**, available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah23822_.htm`

Note that after you complete this process, the WebSphere Business Integration Message Broker V5.0 Configuration Manager component still exists on the original system. After finishing a complete topology deployment (to associate all brokers in the domain with the WebSphere Message Broker V6.0 Configuration Manager on the new system), you can no longer use the WebSphere Business

Integration Message Broker V5.0 Configuration Manager to manage the broker domain.

If you choose to remove the WebSphere Business Integration Message Broker V5.0 Configuration Manager component, enter the **mqsideleteconfigmgr** command in the Windows command prompt on the original system.

21.3 Configuration Manager enhancements

This section details some of the enhancements to the WebSphere Message Broker V6.0 Configuration Manager component.

21.3.1 Configuration Manager runtime changes

The enhancements made to the Configuration Manager include the following:

- The Configuration Manager runtime component is now supported on the AIX, Hewlett Packard-UX, Linux (x86 platform), Linux (zSeries platform), Solaris, Windows, and the z/OS platforms, the same platforms as the broker runtime. For details about the WebSphere Message Broker V6.0 system requirements, refer to the following Web site:

  http://www.ibm.com/software/integration/wbimessagebroker/requirements/index.html

- There is no longer a restriction that a system may have only one Configuration Manager. Multiple Configuration Managers can now be created by entering the following command on the Command Console:

  mqsicreateconfigmgr *cm_name* -i *service_userid* -a *password* -q *cm_qm*

  Details about the other command options available are provided in the topic “mqsicreateconfigmgr” in the WebSphere Message Broker Information Center. Access this information by selecting the path **Reference** → **Operations** → **Commands** → **Runtime commands** → **mqsicreateconfigmgr**, available on the Web at:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an23000_.htm

Notes:

- A broker can still only be managed by a single Configuration Manager. However, multiple domains can be administered from a single Message Brokers Toolkit.
- Names of Configuration Managers must be unique within the broker domain; a Configuration Manager cannot have the same name as another Configuration Manager or broker.

  A Configuration Manager cannot share a host queue manager with another Configuration Manager,although it can share a host queue manager with one broker in the broker domain.

- The Configuration Manager no longer requires a DB2 Universal Database because it maintains the configuration details in an internal repository.

Important: Even after migration, the WebSphere Business Integration Message Broker V5.0 Configuration Manager's DB2 Universal Database remains unmodified.

However, the WebSphere Message Broker V6.0 Configuration Manager uses an *internal repository* instead of a database.

Therefore, a database can be removed, following successful validation of the migrated Configuration Manager.

21.3.2 The Configuration Manager Proxy

The Configuration Manager Proxy is a comprehensive Java interface. It allows you to control broker domains programmatically by using it to write applications to interact with the Configuration Manager to enquire about and change the status of the broker domain.

For more information about the Configuration Manager Proxy, refer to the topic "Configuration manager Proxy" by selecting the path **Developing applications → Developing applications using the CMP → Configuration manager Pyroxene** the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae20620_.htm

On Windows, launch the Java application programming interface documentation from **Start > Programs → IBM WebSphere Message Brokers 6.0 → Java Programming APIs → Configuration Manager Proxy API Documentation**.

For information about developing applications that use the Configuration Manager Proxy Java application programming interface, refer to topic "Developing applications using the CMP" by selecting the path **Developing applications → Developing applications using the CMP** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae33010_.htm

21.3.3 The Configuration Manager Proxy API Exerciser

The Configuration Manager Proxy API Exerciser is a sample application shipped along with WebSphere Message Broker V6.0. It demonstrates the capabilities of the Configuration Manager Proxy by enabling you to invoke most Configuration Manager Proxy functions from within a GUI.

Setting up the correct environment

To use the sample application on platforms other than Windows, set up the correct environment. Include the following Configuration Manager Proxy and WebSphere MQ classes for Java jars in the CLASSPATH environment variable:

```
<install_directory>/sample/ConfigManagerProxy/ConfigManagerProxySamples.jar
<install_directory>/classes/ConfigManagerProxy.jar
com.ibm.mq.jar
connector.jar
```

Also ensure that your Java development directory is on the CLASSPATH. If you want the Exerciser to save user settings, add the current directory to the CLASSPATH. The user settings are saved to the current directory and subsequently loaded from the CLASSPATH via Java's ResourceBundle mechanism.

For details about configuring the environment, refer to topic "Configuring an environment for developing and running Configuration Manager Proxy applications" by selecting the path **Developing applications → Developing applications using the CMP → Configuring an environment for developing and running Configuration Manager Proxy applications** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae33020_.htm

Using the application

For information about what the Configuration Manager Proxy API Exerciser sample application can do, refer to topic “Running the Configuration Manager Proxy API Exerciser sample” by selecting the path **Developing applications** → **Developing applications using the CMP** → **The CMP samples** → **Running the Configuration Manager Proxy API Exerciser sample** in the WebSphere Message Broker Information Center available on the Web at:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae33350_.htm`

Following is a brief introduction to the basic usage of the sample application, demonstrating activities such as connecting to the Configuration Manager, discovering attributes of domain objects, and changing the status of the broker domain.

To launch an application:

- On Windows, select **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **Java Programming APIs** → **Configuration Manager API Proxy Exerciser**.
- On other platforms, enter the following command:

  Installation_Directory`\sample\ConfigManagerProxy\StartConfigManagerProxyExerciser`

When the application starts, right-click the option **[Right-click to connect]** in the top left corner and select the option **Connect to Configuration Manager...**, as illustrated in Figure 21-14.

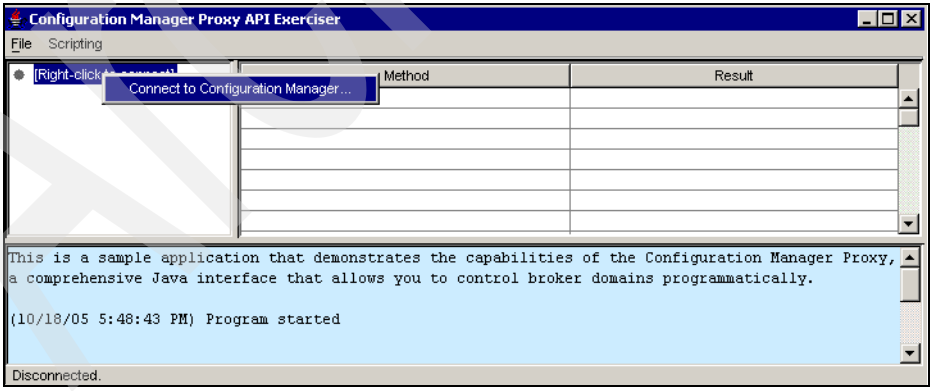


Figure 21-14 Connect to the Configuration Manager Proxy API Exerciser sample application

Type the Configuration Manager connection parameters in the window titled Connect to Configuration Manager.... The parameters include the Configuration Manager's Host name, Port number, and MQ queue manager, as illustrated in Figure 21-15. Then click **Submit**.

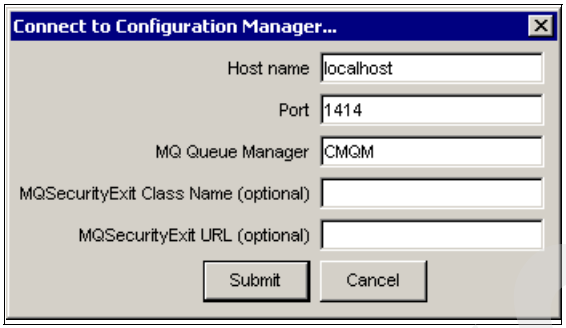


Figure 21-15 Configuration Manager connection parameters

The blue pane at the bottom of the application reports registration for all Configuration Manager domain objects, such as brokers, execution groups, and message flows. After the four lines shown in Figure 21-15 are displayed, the Configuration Manager Proxy API Exerciser is ready to manipulate the domain objects.

Example 21-15 Successful connection to the Configuration Manager Proxy API Exerciser

```
The Configuration Manager's object hierarchy is available for use.
Click on an object in the hierarchy to select it and display its properties.
Right-click on a selected object to manipulate it.
<---- cmp.exerciser.ClassTesterForConfigManagerProxy.testConnect
```

The application now displays a hierarchical view of the domain of brokers that the Configuration Manager controls. When you select a domain object such as a broker, an execution group, or a message flow in the top left pane, the right pane displays information about the selected object.

For example, to find the attributes of a broker, select the broker in the left pane. The right pane then displays the attributes, including the list of execution groups, queue manager, and status, as illustrated in Figure 21-16.

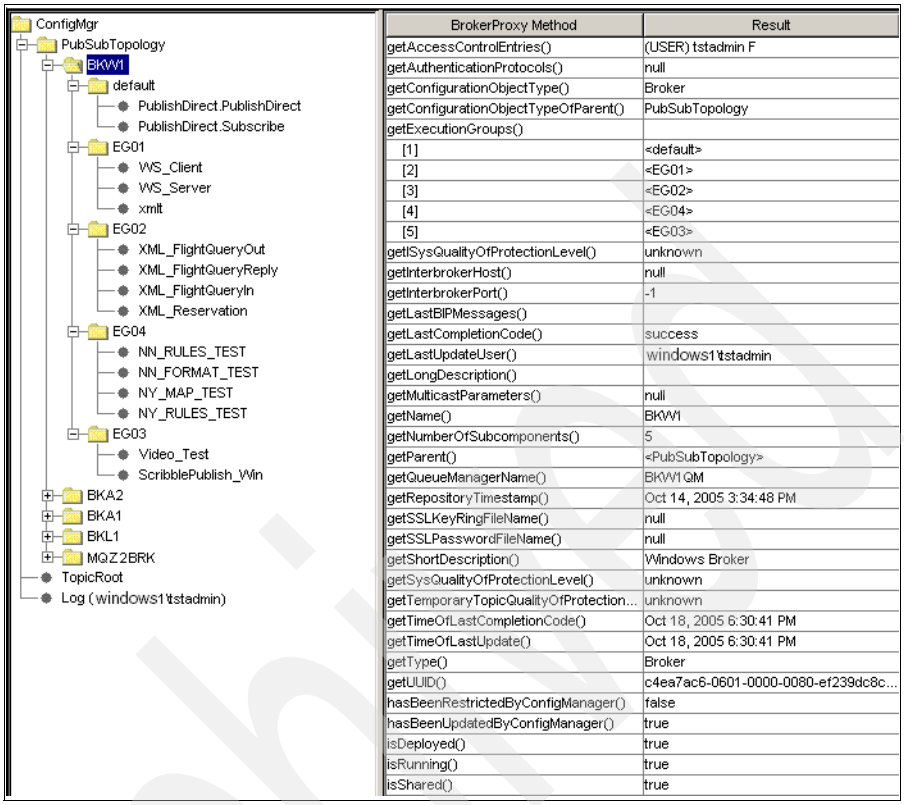


Figure 21-16 Displaying a broker's attributes in the Configuration Manager Proxy API Exerciser sample application

To determine which actions can be performed on a domain object, select the object in the left pane of the application and right-click. A context menu appears which displays the available options, as shown in Figure 21-17.

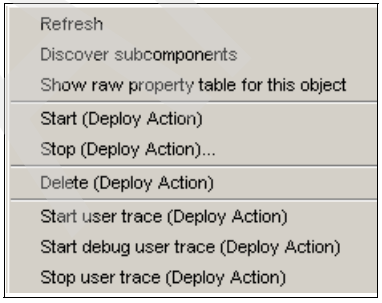


Figure 21-17 Context menu for a message flow

You can change the status of the broker domain by selecting an action from the menu. For example, by right-clicking a message flow and selecting **Stop (Deploy Action)...** from the menu, a dialog box is displayed as shown in Figure 21-18.

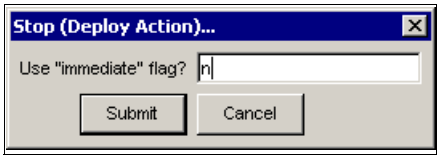


Figure 21-18 Dialog box

To stop the selected message flow, select **Submit**. The right pane of the application displays the attribute *isRunning()* with a result of *false*. The blue pane at the bottom of the application displays the messages sent and received by the Configuration Manager Proxy, as shown in Example 21-6.

Example 21-16 Messages sent and received by the Configuration Manager Proxy

```
----> cmp.exerciser.ClassTesterForMiscellaneousActions.testStopMsgFlows(<WS_Client>, false)
The request was successfully sent to the Configuration Manager.
<---- cmp.exerciser.ClassTesterForMiscellaneousActions.testStopMsgFlows
----> cmp.exerciser.ExerciserAdministeredObjectListener.processActionResponse(...)
affectedObject = WS_Client
completionCode = success
(Log entry) {message=BIP1522I, detail=BIP1522I: Stop message flow operation successfully
initiated for broker BKW1.
The Configuration Manager has processed the stop message flow request and initiated a stop
message flow operation for the target message flows., source=ConfigMgr, timestamp=Sun Oct 23
10:38:04 BST 2005}
(Reference property) parent.uuid=e8eb9bdb-0601-0000-0080-ef239dc8c9cd
(Reference property) uuid=c3e09ddb-0601-0000-0080-e36abdf61734
(Reference property) commsmessage.lastinbatch=true
(Reference property) object.runstate=stopped
(Reference property) configmanagerproxy.osname=Windows 2000
(Reference property) userid=tstadmin
(Reference property) configmanagerproxy.hostname=windows1
(Reference property) commsmessage.configobjecttype=MessageProcessingNodeType
(Reference property) parent.type=ExecutionGroup
(Reference property) commsmessage.operationtype=modify
(Reference property) configmanagerproxy.noeventlog=false
<---- cmp.exerciser.ExerciserAdministeredObjectListener.processActionResponse()
----> cmp.exerciser.ExerciserAdministeredObjectListener.processModify(...)
affectedObject = WS_Client
(Changed attribute) object.runstate
<---- cmp.exerciser.ExerciserAdministeredObjectListener.processModify()
----> cmp.exerciser.ExerciserAdministeredObjectListener.processModify(...)
affectedObject = Log (windows1\tstadmin)
(New subcomponent) 2056<<BKW1<<2005-10-23 10:38:57<<BKW1<<c4ea7ac6-0601-0000-0080-ef239dc8c9cd
```

```
<---- cmp.exerciser.ExerciserAdministeredObjectListener.processModify()
----> cmp.exerciser.ExerciserAdministeredObjectListener.processModify(...)
affectedObject = Log (windows1\tstadmin)
(New subcomponent) 4040<<BKW1<<2005-10-23 10:38:58<<EG01<<e8eb9bdb-0601-0000-0080-ef239dc8c9cd
<---- cmp.exerciser.ExerciserAdministeredObjectListener.processModify()
```

In the Message Brokers Toolkit Domains view, double-click **Event Log** to open the Event Log view. The messages received by the Configuration Manager Proxy confirming the success of the operation (message identifiers 2056 and 4040) are reported in the output from the Configuration Manager Proxy (see Example 21-16) and the Event Log view (see Figure 21-19).

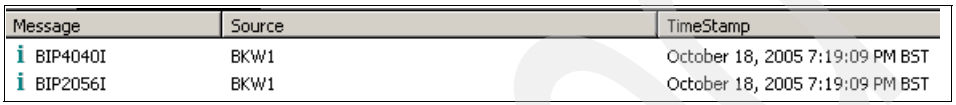


Figure 21-19 Successful configuration change messages received in Message Brokers Toolkit

The result is also reflected in the Domains view and the Alerts view.

21.3.4 Access Control Lists

WebSphere Message Broker uses Access Control Lists (ACLs) to control which users and groups can manipulate objects within the Configuration Manager, including the Message Brokers Toolkit and the Configuration Manager Proxy applications. For more details, refer to topic “ACL permissions” by selecting the path **Reference** → **Security** → **ACL permissions** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ap12520_.htm

Following is a brief discussion of ACL entries and ways to manage them.

ACL entry permissions behave in a hierarchical manner, in order to reduce the number required. Therefore, when an ACL entry is made for a given object, that permission is granted to that object and inherited by all objects beneath it in the hierarchy, unless overridden by another entry. Figure 21-20 illustrates the ACL hierarchy.

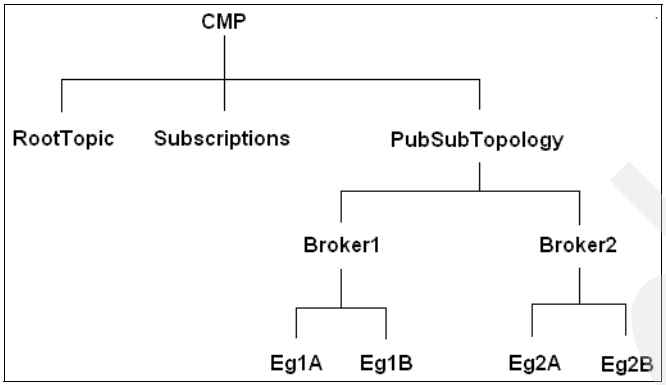


Figure 21-20 Access Control List hierarchy

Managing the ACLs

On the Configuration Manager system, you can manage the ACL entries with a series of commands—on the Command Console, if the Configuration Manager is on Windows—and at the command prompt, after running mqsiprofile on other platforms.

Listing ACL entries

To list the ACL entries that have been granted, enter the following command:

```
mqsilistaclentry cm_name
```

On z/OS, run the BIPLIACL job or enter the following console command from SDSF:

```
/F cm_name,LA
```

For details about other command options available, refer to topic "mqsilistaclentry" by selecting the path **Reference** → **Operations** → **Commands** → **Runtime commands** → **mqsilistaclentry** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an23050_.htm

Example 21-7 shows the format of the output that is produced.

Example 21-17 Output from listing the ACL entries of a Configuration Manager

```
$ mqsilistaclentry CMA1
BIP1778I: mqbrkrs          -  GROUP  -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
BIP1778I: mqbrdevt         -  GROUP  -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
BIP1778I: mqbrasgn         -  GROUP  -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
BIP1778I: mqbrops          -  GROUP  -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
```

```
BIP1778I: <userid>          -  USER   -  F  -  ConfigManagerProxy  -  ConfigManagerProxy
BIP1778I: <domain>\<userid> -  USER   -  F  -  PubSubTopology      -  PubSubTopology
BIP1778I: mqbrtpic          -  GROUP  -  F  -  TopicRoot           -  TopicRoot
BIP1778I: <userid>          -  USER   -  F  -  BKW1                -  Broker
BIP1778I: <userid>          -  USER   -  F  -  BKW1/EG01           -  ExecutionGroup
```

Note: When a user creates or starts a Configuration Manager, an entry is made in the ACL to grant that user ID, on any machine, full authority at the top of the hierarchy, the ConfigManagerProxy object.

ACL entries migration

As part of Configuration Manager migration, user groups used in previous versions to grant users authorities have been added as ACL entries, as illustrated in Example 21-17. Furthermore, ACL entries made in WebSphere Business Integration Message Broker V5.0 have been migrated.

In WebSphere Business Integration Message Broker V5.0, whenever a user created a broker or execution group, an access control entry was created to grant that user ID, on any machine, full authority to that object, as illustrated in Example 21-7. However, because ACL entries are now hierarchical, these additional entries are no longer required and can be deleted.

Deleting ACL entries

To delete an ACL entry, enter the **mqsideleteaclentry** command. For example, to remove the ACL entry for a user ID from any machine for a particular execution group, enter the following command:

```
mqsideleteaclentry cm_name -u userid -a -b broker -e exgrp
```

On z/OS, run the BIPDLACL job, or enter the following console command from SDSF:

```
/F cm_name,DA parms
```

For details about other command options available, refer to topic “mqsideleteaclentry” by selecting the path **Reference** → **Operations** → **Commands** → **Runtime commands** → **mqsideleteaclentry** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an23040_.htm

Creating ACL entries

Four different access levels can be granted for a user or group: Full; Deploy; Edit; View. If the control given to a particular user for an object should be different from the inherited permission that user would be granted, you can

create a new entry to override the access available to the object and its children. For example, to limit a user ID on a specific machine to only being able to deploy to a specific broker, enter the following command:

```
mqsicreateaclentry cm_name -u userID -m machine_name -x D -b broker
```

On z/OS, run the BIPCRACL job, or enter the following console command from SDSF:

```
/F cm_name,CA parms
```

Because of the hierarchical nature of ACL entries, the user ID on the machine specified would also implicitly be granted deployment access to any execution groups belonging to the broker, as well as to a limited form of view access on the Configuration Manager Proxy and Topology objects. With this limited access, the user would be able to view the presence of other brokers in the domain. However, the user would be unable to view details such as brokers' names, or perform any actions on them.

For details about other commands available, refer to topic "mqsicreateaclentry" by selecting the path **Reference** → **Operations** → **Commands** → **Runtime commands** → **mqsicreateaclentry** in the WebSphere Message Broker Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an23030_.htm

22

Migrating the User Name Server

This chapter describes the process of migrating a User Name Server on AIX to WebSphere Message Broker V6.0 User Name Server on AIX. It guides you through the migration of the AIX User Name Server in the scenario described in Chapter 17, “WebSphere Business Integration Message Broker Version 5.0 environment setup” on page 313.

The User Name Server is a component that authenticates users and groups performing publish/subscribe operations. It uses access control lists (ACLs) to enable control over the topics. The users can publish/subscribe to particular topics based on the security access provided to them. It requires WebSphere MQ queue manager, which can be shared with a Configuration Manager, a broker, or both in the broker domain.

After you complete the steps in this chapter, the User Name Server will be ready to operate in the migrated domain.

22.1 Migration preparation

This section describes the tasks that you have to perform before the User Name Server component migration. Figure 22-1 illustrates the scenario before and after the migration of the AIX User Name Server V5.0.

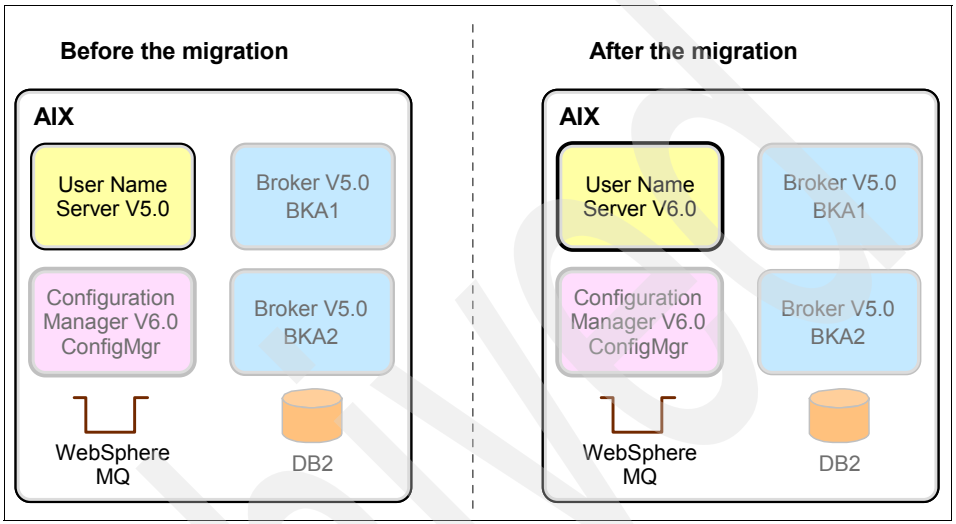


Figure 22-1 User Name Server migration scenario

22.1.1 Reviewing the existing installation

Check the existing installation for errors. Prior to migrating, ensure that the User Name Server is running properly and that no errors are reported.

Reviewing the configuration

To confirm that all the important components are running properly, we recommend that you execute the following commands and check the output.

Log in using the User Name Server user ID and launch the commands from the appropriate platform command prompt.

Perform the following steps:

1. Display the list of queue managers by entering the **dspmq** command. Check the status of all expected queue managers as shown in Example 22-1.

Example 22-1 Displaying the queue managers

```
$ dspmq
QMNAME(BKA1QM)                                            STATUS(Running)
QMNAME(BKA2QM)                                            STATUS(Running)
QMNAME(UNSQM)                                             STATUS(Running)
```

2. Display the list of Message Broker V5.0 components by entering the **mqsilist** command. Check the status of all expected components as shown in Example 22-2.

Example 22-2 Displaying the components

```
$ mqsilist
BIP8099I: BKA1  -  BKA1QM
BIP8099I: BKA2  -  BKA2QM
BIP8099I: UserNameServer  - UNSQM
BIP8071I: Successful command completion.
```

3. Verify that the User Name Server is running. On AIX, enter the following command:

   ```
   ps -ef | grep UserNameServer
   ```

   Example 22-3 illustrates the User Name Server processes.

Example 22-3 Displaying the User Name Server processes

```
$ ps -ef | grep UserNameServer
tstadmin 24372 60958   0 11:44:13  pts/4  0:00 grep UserNameServer
tstadmin 55140     1   0 11:29:37      -  0:00 bipservice UserNameServer
tstadmin 59718 55140   0 11:29:37      -  0:00 bipuns UserNameServer
```

Premigration verification

Check the error log files for Message Broker V5.0 components, queue manager, and DB2 for any important information or errors.

For details about which error log files you need to check and how to check them, refer to 27.1.4, “Checking the logs and where to find them” on page 505.

22.1.2 Installing WebSphere Message Broker Version 6.0

Install and verify the WebSphere Message Broker V6.0 installation, if this has not already been done.

When performing the installation, follow the standard installation documentation. Verify that the following products are installed:

- Rational Agent Controller V6.0.0.1
- WebSphere Message Broker V6.0 component, User Name Server

Important: To install any fix pack for Rational Agent Controller, refer to WebSphere Message Broker V6.0 documentation.

WebSphere Message Broker V6.0 requires the correct version of Rational Agent Controller. Any other version is not supported.

22.2 Configuration settings

Before starting the component migration, you must make some changes to the configuration, as explained in the following sections.

Before you make any changes to the configuration, you must stop the User Name Server. First, log in using the User Name Server user ID. Then launch the following command from the appropriate platform command prompt:

```
mqsistop UserNameServer
```

When the server is stopped, you see a message like the one in Example 22-4.

Example 22-4 Stopping the User Name Server

```
$ mqsistop UserNameServer
BIP8071I: Successful command completion.
```

22.2.1 Updating the environment settings

The environment settings profile for the User Name Server user ID has changed in Message Broker V6.0, therefore you must change the user profile, if it is not yet changed.

The full path of the new Message Broker sample profile is *broker_install_directory*/6.0/bin/mqsiprofile.

Edit the Message Broker service user ID profile using an editor such as `vi`:

1. Remove the WebSphere Business Integration Message Broker V5.0 settings. This may be a call to the sample profile, in which case delete the line:

   ```
   . /usr/opt/mqsi/sample/profiles/profile.aix
   ```

2. Add the line:

   `. broker_install_directory/6.0/bin/mqsiprofile`

   Example 22-5 shows the relevant path of the user profile.

Example 22-5 Sample of the user profile

```
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:/usr/java131
/jre/bin:.

export PATH

. broker_install_directory/6.0/bin/mqsiprofile
```

3. Log off and log in again using the User Name Server user ID to activate these changes.

Important: If Message Broker V5.0 and Message Broker V6.0 must coexist, do not use both versions of the Message Broker profile in the user profile at the same time.

Enter the appropriate version of the Message Broker profile manually. Use the appropriate version of the Message Broker profile to start each User Name Server.

22.3 User Name Server migration

This section describes the component User Name Server migration steps for distributed platforms. For more details about how to perform the migration on z/OS, refer to the WebSphere Message Broker Information Center, under the topic **Migrating → Migrating from WebSphere Business Integration Event Broker Version 5.0 → Migrating a User Name Server → z/OS** available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.eb.doc/ah23844_.htm

22.3.1 Before starting

Before migrating a User Name Server, ensure that there are no errors in the operating system application log and that the User Name Server is stopped.

22.3.2 Migrating the User Name Server

To migrate a User Name Server on distributed operating systems to Message Broker V6.0 at the same location, log in using the User Name Server user ID and launch the commands from the appropriate platform command prompt. Perform the following steps:

1. Stop all the brokers and the Configuration Manager connected to the User Name Server. (This is not mandatory.)

   Note: If the broker and the Configuration Manager are not stopped during the migration of the User Name Server, then their security requests are not processed until the User Name Server migration is complete and the User Name Server has been restarted.

2. Ensure that the User Name Server is not running.
3. Perform the appropriate environment settings as shown in Example 22-6.

Example 22-6 Running Message Broker V6.0 profile

```
-$ . <broker_install_directory>/6.0/bin/mqsiprofile
```

4. Perform a premigration check on the User Name Server by entering the Message Broker V6.0 command:

   ```
   mqsimigratecomponents -c UserNameServer
   ```

   Note: This command fails and the defect is expected to be fixed in a future fix pack.

5. Migrate the User Name Server to Message Broker V6.0 by entering the Message Broker V6.0 command:

   ```
   mqsimigratecomponents UserNameServer
   ```

   Example 22-7 contains the complete command output.

Example 22-7 Migrating the User Name Server

```
$ mqsimigratecomponents UserNameServer
BIP8675I: UserNameServer 'UserNameServer' (Version 2.1) specified for
migration.
BIP8750I: Starting registry migration for component 'UserNameServer'.
BIP8755I: Copied value 'QueueManagerName' into the new location
BIP8755I: Copied value 'ServiceUserId' into the new location
BIP8755I: Copied value 'ServicePassword' into the new location
BIP8755I: Copied value 'NTSecurityDomainName' into the new location
BIP8755I: Copied value 'RefreshInterval' into the new location
```

```
BIP8765W: Found registry value 'GroupsInDataSource' that is not valid in the
source version; left unchanged.
BIP8765W: Found registry value 'AuthProtocolDataSource' that is not valid in
the source version; left unchanged.
BIP8755I: Copied value 'AdminAgentPID' into the new location
BIP8755I: Copied value 'BrokerUUID' into the new location
BIP8763I: Deleted value 'QueueManagerName' from the old location
BIP8763I: Deleted value 'ServiceUserId' from the old location
BIP8763I: Deleted value 'ServicePassword' from the old location
BIP8763I: Deleted value 'NTSecurityDomainName' from the old location
BIP8763I: Deleted value 'RefreshInterval' from the old location
BIP8763I: Deleted value 'AdminAgentPID' from the old location
BIP8763I: Deleted value 'BrokerUUID' from the old location
BIP8768I: Finished registry migration for component 'UserNameServer'.
BIP8654I: Moving filesystem artefacts from '/var/mqsi' to '/var/mqsi'
BIP8655I: Removing '/var/mqsi/brokers/UserNameServer/bin/mqsistart'.
BIP8655I: Removing '/var/mqsi/brokers/UserNameServer/bin/mqsistop'.
BIP8655I: Removing '/var/mqsi/brokers/UserNameServer/bin'.
BIP8655I: Removing '/var/mqsi/brokers/UserNameServer'.
BIP8071I: Successful command completion.
```

When you migrate your User Name Server, a similar successful command output must be observed.

6. Run the postmigration check of the Message Broker V6.0 User Name Server by entering the following command:

   ```
   mqsimigratecomponents -v UserNameServer
   ```

   A successful command output reports the message BIP8689I as shown in Example 22-8.

Example 22-8 Verifying the User Name Server migration

```
$ mqsimigratecomponents -v UserNameServer
BIP8675I: UserNameServer 'UserNameServer' (Version 6.0) specified for
migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

22.3.3 Validating the User Name Server migration

After completing the User Name Server migration, your tasks on the AIX system are nearly complete. Start the User Name Server and verify that it executes properly.

The commands for managing the User Name Server are the same as in Message Broker V5.0.

Starting the User Name Server

Log in using the User Name Server user ID and launch the commands from the appropriate platform command prompt. Then perform the following steps:

1. Display the list of WebSphere Message Broker components by entering the command:

   ```
   mqsilist -a
   ```

   The command should display the same components as before the migration. The version is also displayed for all the previous versions of the broker, as shown in Example 22-9.

   Important: The previous versions of the broker are displayed only if the **-a** option is used.

Example 22-9 Displaying the components

```
$ mqsilist -a
BIP8099I: Broker: BKA1 (Version 5)  -  BKA1QM
BIP8221I: Broker: BKA2 (Version 5)  -  BKA2QM
BIP8099I: UserNameServer: UserNameServer  -  UNSQM
BIP8071I: Successful command completion.
```

2. Start all the brokers and the Configuration Manager connected to the User Name Server (if they were stopped in step 1 in 22.3.2, “Migrating the User Name Server”).
3. Start your User Name Server by entering the following command:

   ```
   mqsistart UserNameServer
   ```

   Example 22-10 illustrates how to start the User Name Server.

Example 22-10 Starting the User Name Server

```
$ mqsistart UserNameServer
WebSphere MQ queue manager running.
BIP8096I: Successful command initiation, check the system log to ensure that
the component started without problem and that it continues to run without
problem.
```

4. Check if the User Name Server is running. On AIX, enter the following command:

   ```
   ps -ef | grep UserNameServer
   ```

Example 22-11 illustrates how to check the User Name Server processes.

Example 22-11 Displaying the User Name Server processes

```
$ ps -ef | grep UserNameServer
tstadmin 19554     1   0 12:35:22      -  0:00 bipservice UserNameServer
tstadmin 25506 51718   0 12:36:50  pts/2  0:00 grep UserNameServer
tstadmin 59204 19554   0 12:35:22      -  0:00 bipuns UserNameServer
```

5. Verify the operating system application log for information or errors.

   If the migration fails, then refer to Chapter 27, “Verifying migration to WebSphere Message Broker V6.0” on page 495 for information.

   After the migration of the product is completed and a thorough testing has been performed to ensure that the newly migrated components are functionally verified, the WebSphere Business Integration Message Broker V5.0 product can be uninstalled at the user’s discretion.

Migration verification

Verify the error log files after the migration to ensure that there were no errors during the migration.

For details about which error log files you need to check and to learn how to check them, refer to 27.1.4, “Checking the logs and where to find them” on page 505.

23

Migrating a Windows broker

This chapter describes the process of migrating a WebSphere Business Integration Message Broker V5.0 broker running on Windows with a DB2 data source to WebSphere Message Broker V6.0.

It guides you through the migration of the Windows broker in the scenario described in Chapter 17, “WebSphere Business Integration Message Broker Version 5.0 environment setup” on page 313. After completing the steps in this chapter, the broker is ready to operate in the migrated domain.

23.1 Migration preparation

This section describes the tasks that you have to perform before the broker component migration. Figure 23-1 illustrates the scenario before and after the migration of the Windows Broker V5.0.

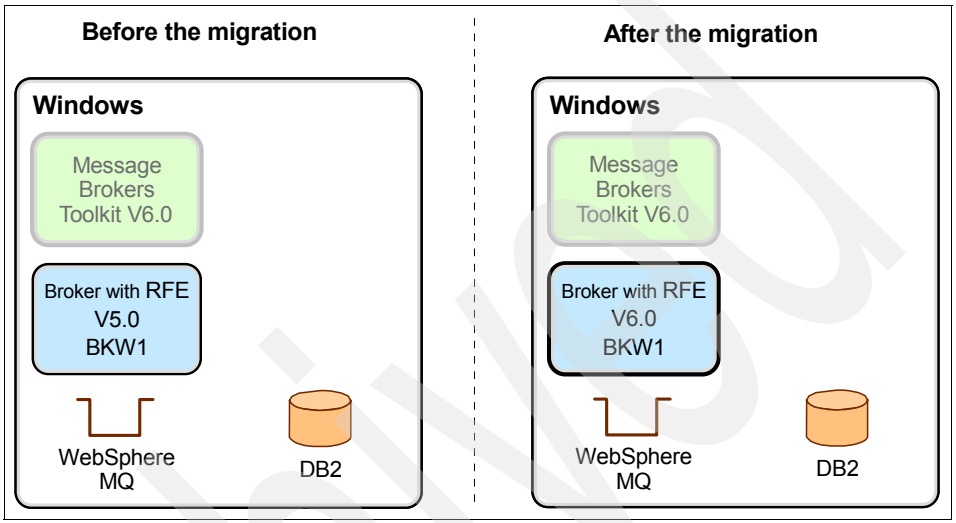


Figure 23-1 Broker migration scenario

23.1.1 Reviewing the existing installation

Check the existing installation for errors. Prior to migrating, ensure that the broker is running properly and that no errors are reported.

Reviewing the configuration

To confirm that all the important components are running properly, we recommend that you use the following commands and check the output.

Log in using the Message Broker service user ID and launch the commands from the Windows command prompt. Perform the following steps:

1. Display the list of queue managers by entering the **`dspmq`** command. Check the status of all expected queue managers as shown in Example 23-1.

Example 23-1 Displaying the queue managers

```
dspmq
QMNAME(BKW1QM)        STATUS(Running)
QMNAME(CMQM)        STATUS(Running)
```

2. Display the list of WebSphere Business Integration Message Broker components by entering the **mqsilist** command. Check the status of all the expected components as shown in Example 23-2.

Example 23-2 Displaying the components

```
mqsilist
BIP8099I: Broker: BKW1 -  BKW1QM
BIP8099I: ConfigMgr: CMMQ -  CMQM
```

3. Display the details of each broker by entering the following command:

   `mqsilist broker_name`

   Check all expected execution groups as shown in Example 23-3.

Example 23-3 Displaying the execution groups

```
mqsilist BKW1
BIP8130I: Execution Group: EG1 -  5884
BIP8130I: Execution Group: EG2 -  5887
BIP8130I: Execution Group: EG3 -  5857
BIP8130I: Execution Group: EG4 -  4537
BIP8071I: Successful command completion.
```

4. Verify that the database is available by entering the following command in a DB2 command window as shown in Example 23-4:

   `db2 connect to broker_database`

   In the Windows command prompt, enter the **db2cmd** command.

Example 23-4 Testing the database connection

```
db2 connect to BKW1DB
Database Connection Information
Database server        = DB2/NT 8.2.0
SQL authorization ID   = TSTADMIN
Local database alias   = BKW1DB
```

5. Reset the DB2 connection by entering the following command:

   `db2 disconnect current`

Premigration verification

Check the error log files for Message Broker V5.0 components, queue manager, and DB2 for any important information or errors.

For details about which error log files you need to check and to learn how to check them, refer to 27.1.4, “Checking the logs and where to find them” on page 505.

Backing up the broker configuration

Ensure that you have successfully backed up the broker configuration.

For details about how to back up the broker configuration, refer to Chapter 19, “Backing up the WebSphere Business Integration Message Broker V5.0 domain” on page 339.

23.1.2 Installing WebSphere Message Broker Version 6.0

Install and verify the WebSphere Message Broker V6.0 installation, if this has not already been done.

When performing the installation, we recommend that you follow the standard installation documentation. Verify that the following products are installed:

- Rational Agent Controller V6.0.0.1
- WebSphere Message Broker V6.0 components:
  - Broker
  - Transformation Services

Tip: We recommend that you install the latest fix pack for WebSphere Message Broker V6.0 and the appropriate Rational Agent Controller version.

To install any fix pack for Rational Agent Controller, refer to WebSphere Message Broker V6.0 documentation.

WebSphere Message Broker V6.0 requires the correct version of Rational Agent Controller. Any other version is not supported.

23.2 Configuration settings

Before starting the component migration, you must make some changes to the configuration, as explained in the following sections.

Before you make any changes to the configuration, you should stop the broker. First, log in using the Message Broker service user ID. Then launch the following command from the Windows command prompt:

`mqsistop` *broker_name*

When the broker is stopped, you see a message like the one shown in Example 23-5.

Example 23-5 Stopping the broker

```
mqsistop BKW1
BIP8071I: Successful command completion.
```

23.2.1 Updating the environment settings

The environment settings for the service user ID have changed in Message Broker V6.0, therefore you must change the environment variables. This initialization ensures that all commands entered interact with the correct installation of the code.

The new environment variables are set up automatically when opening the WebSphere Message Broker Command Console.

Select **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **Command Console** to open the command console. This opens a command window with the correct environment. Note that you are not restricted to invoking run-time commands in this window, and you can perform typical actions by entering operating systems commands.

Figure 23-2 shows a command console.

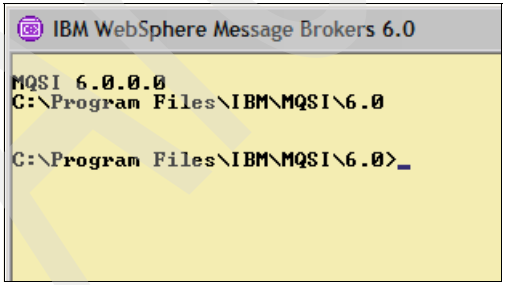


Figure 23-2 IBM WebSphere Message Brokers 6.0 Command Console

23.2.2 Updating the Open Database Connectivity settings

The next step is to modify the settings for the Open Database Connectivity (ODBC) driver, after first recording what the previous settings were. The new product contains new versions of the ODBC drivers for Oracle and Sybase databases. (Detailed information concerning these databases is beyond the scope of this book.)

In the DB2 for Windows database, there is no change in the ODBC settings.

Tip: At the time of writing, the information about ODBC settings is not referenced from the Message Broker V5.0 migration topics.

However, the ODBC setting details in the WebSphere Message Broker 6.0 Information Center are relevant to WebSphere Business Integration Message Broker V5.0, as well.

At the Information Center, refer to the topic **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 to WebSphere Message Broker Version 6.0 → Migrating components on different computers → Migrating the broker → Distributed operating systems → Changing the ODBC connection and XA resource manager definitions for a migrated broker**. This information is available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13850_.htm`

23.3 The broker migration

This section describes the component migration steps.

23.3.1 Before starting

Before migrating a broker, ensure that you do not have any aggregations in progress. When migrating a broker to Message Broker V6.0, any live data that is stored for aggregations in progress is lost.

Also, check if the Message Broker service user ID uses the supported locale and code page.

Tip: Refer to the WebSphere Message Broker V6.0 Information Center under the topic **Reference** → **Installation** → **National Language Support**, available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an08720_.htm

23.3.2 Migrating the broker

To migrate a Message Broker V5.0 broker on distributed operating systems to Message Broker V6.0 at the same location, log in using the Message Broker service user ID and launch the commands from the appropriate platform command prompt. Then perform the following steps:

1. Stop all the channels communicating to the broker queue manager.
2. Open the Command Console using the path **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **Command Console**.
3. Perform a premigration check on the broker by entering the Message Broker V6.0 command:

   ```
   mqsimigratecomponents -c broker_name
   ```

   Example 23-6 illustrates how to test the broker migration.

Example 23-6 Testing the broker migration

```
mqsimigratecomponents -c BKW1
BIP8849I: Broker 'BKW1' (Version 5.0) with Queue Manager 'BKW1QM' and Data
Source 'BKW1DB' specified for migration.
BIP8680I: Pre-migration check succeeded.
BIP8071I: Successful command completion.
```

4. Migrate the broker to Message Broker V6.0 by entering the Message Broker V6.0 command:

   ```
   mqsimigratecomponents broker_name
   ```

   Example 23-7 illustrates how to migrate the broker.

Example 23-7 Migrating the broker

```
mqsimigratecomponents BKW1 -q -s 5.0.1.6 -t 6
BIP8849I: Broker 'BKW1' (Version 5.0) with Queue Manager 'BKA1QM' and Data
Source 'BKW1DB' specified for migration.
BIP8750I: Starting registry migration for component 'BKW1'.
...
...
BIP8768I: Finished registry migration for component 'BKW1'.
...
```

```
...
BIP8669I: Database migration successful
...
...
BIP8789I: Finished WebSphere MQ queue migration for component 'BKW1'.
BIP8071I: Successful command completion.
```

5. Perform the postmigration check of the Message Broker V6.0 broker by entering the Message Broker V6.0 command:

   `mqsimigratecomponents -v broker_name`

   A successful command output reports the message BIP8689I, as shown in Example 23-8.

Example 23-8 Verifying the broker migration

```
mqsimigratecomponents -v BKW1
BIP8849I: Broker 'BKW1' (Version 6.0) with Queue Manager 'BKW1QM' and Data
Source 'BKW1DB' specified for migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

23.3.3 Validating the broker migration

After completing the broker migration, your tasks on the Windows system are nearly complete. Start the broker and then verify that it executes properly.

The commands for managing the broker are the same as in Message Broker V5.0.

Starting the broker

Log in using the Message Broker service user ID and launch the commands from the appropriate platform command prompt. Then perform the following steps:

1. Start all the channels communicating to the broker queue manager.
2. Open the Command Console using the path **Start** → **Programs** → **IBM WebSphere Message Brokers 6.0** → **Command Console**.
3. Start the broker by entering the following command:

   `mqsistart broker_name`

   Example 23-9 shows how to start the broker.

Example 23-9 Starting the broker

```
mqsistart BKW1
WebSphere MQ queue manager running.
BIP8096I: Successful command initiation, check the system log to ensure that
the component started without problem and that it continues to run without
problem.
```

4. Display the list of WebSphere Message Broker components by entering the following command:

   ```
   mqsilist -a
   ```

   The command should display the same components as before the migration. The version is also displayed for all the previous versions of the broker, as shown in Example 23-10.

   Important: The previous versions of the broker is displayed only if the **-a** option is used.

Example 23-10 Displaying the components

```
mqsilist
BIP8099I: Broker: BKW1 -  BKW1QM
BIP8099I: ConfigMgr: CMMQ -  CMQM
BIP8071I: Successful command completion.
```

5. Display the details of the broker by entering the following command:

   ```
   mqsilist broker_name
   ```

   The command should display the same list of execution groups as before the migration, as shown in Example 23-11.

Example 23-11 Displaying the execution groups

```
mqsilist BKW1
BIP8130I: Execution Group: EG1 -  5884
BIP8130I: Execution Group: EG2 -  5887
BIP8130I: Execution Group: EG3 -  5857
BIP8130I: Execution Group: EG4 -  4537
BIP8071I: Successful command completion.
```

Migration verification

Verify the error log files after the migration to ensure that there were no errors during the migration.

For details about which error log files you need to check and to learn how to check them, refer to 27.1.4, “Checking the logs and where to find them” on page 505.

23.3.4 Verifying the migration for Rules and Formatter Extension

If the new WebSphere Message Broker V6.0 uses the Rules and Formatter Extension, then perform the following steps:

1. Check the environment variable NNSY_ROOT inside the IBM WebSphere Message Broker V6.0 Command Console. The NNSY_ROOT variable is updated automatically by the New Era Of Networks installation to the installation directory:

   ```
   NNSY_ROOT=C:\Program Files\IBM\rf41
   ```

2. Check the ODBC database setting in 23.2.2, “Updating the Open Database Connectivity settings” on page 442.

   The nnsyreg.dat sample file of New Era Of Networks is in the *rfe_install_dir*\rfdt\gui\examples directory. Check the differences between the current nnsyreg.dat file and the Message Broker V6.0 sample.

   Edit the nnsyreg.dat file and change the NNOT_SHARED_LIBRARY parameter in all sessions.

   Tip: If the nnsyreg.dat file is located inside the WebSphere Business Integration Message Broker V5.0 product directory, move it to an appropriate directory.

   Example 23-12 shows a section of the nnsyreg.dat file.

Example 23-12 nnsyreg.dat sample file

```
Session.MQSI_CONFIG
   NNOT_SHARED_LIBRARY    = dvdb41db2
   NNOT_FACTORY_FUNCTION  = NNSesDB2Factory
   NN_SES_SERVER          = NNDB
   NN_SES_USER_ID         = tstadmin
   NN_SES_PASSWORD        = password
```

   New Era Of Networks searches for the current nnsyreg.dat file in the following directory order:

   a. Current working directory
   b. Directory specified by the environment variable NN_CONFIG_FILE_PATH
   c. Directory specified by the environment variable NNSY_ROOT

   Ensure that you use the correct configuration file.

24

Migrating an AIX broker

This chapter describes the migration of a WebSphere Business Integration Message Broker V5.0 broker running on AIX with a DB2 data source to WebSphere Message Broker V6.0.

It guides you through the migration of the AIX broker in the scenario described in Chapter 17, “WebSphere Business Integration Message Broker Version 5.0 environment setup” on page 313. After completing the steps in this chapter, the broker is ready to operate in the migrated domain.

24.1 Migration preparation

This section describes the tasks to be performed before the broker component migration. Figure 24-1 illustrates the AIX broker V5.0 scenario before and after migration.

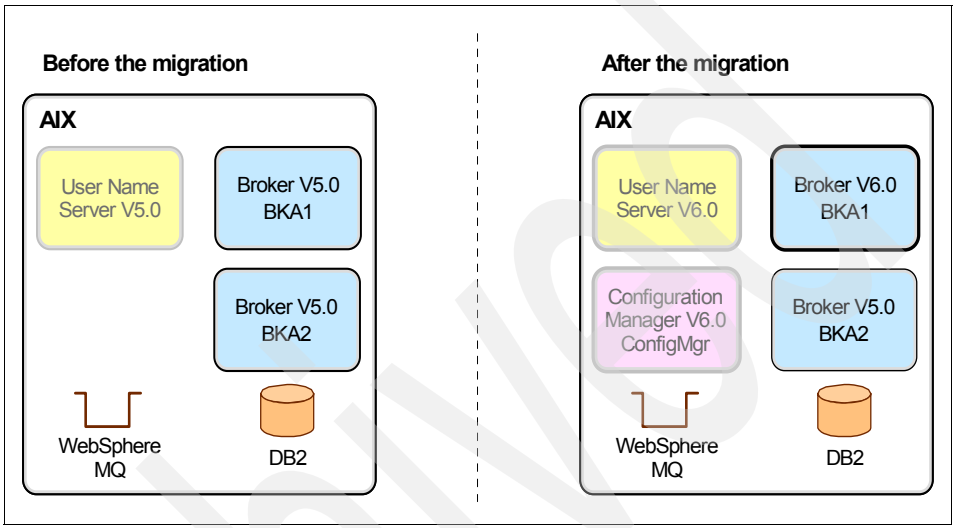


Figure 24-1 AIX broker V5.0 migration scenario

24.1.1 Reviewing the existing installation

Check the existing installation for errors. Prior to migrating, ensure that the broker is running properly and that no errors are reported.

Tip: On UNIX systems, use the **`script /tmp/output.txt`** command to make a typescript of everything displayed on your terminal to a specified file (/tmp/output.txt, in this case).

Reviewing the configuration

To confirm that all important components are running properly, perform the following steps and check the output.

Log in with the Message Broker service user ID and launch the commands from the appropriate platform command prompt, as explained in the following steps.

1. Display the list of queue managers by issuing the **`dspmq`** command. Check the status of all expected queue managers, as shown in Example 24-1 on page 449.

Example 24-1 Displaying the queue managers

```
$ dspmq
QMNAME(BKA1QM)                                           STATUS(Running)
QMNAME(BKA2QM)                                           STATUS(Running)
QMNAME(UNSQM)                                            STATUS(Running)
```

2. Display the list of Message Broker V5.0 components by issuing the **mqsilist** command. Check the status of all expected components as shown in Example 24-2.

Example 24-2 Displaying the components

```
$ mqsilist
BIP8099I: BKA1  -  BKA1QM
BIP8099I: BKA2  -  BKA2QM
BIP8099I: UserNameServer  -
BIP8071I: Successful command completion.
```

3. Display the details of your broker by issuing the following command:

   `mqsilist` *`broker_name`*

   Check all expected execution groups as shown in Example 24-3.

Example 24-3 Displaying the execution groups

```
$ mqsilist BKA1
BIP8130I: Execution Group: EG02  -  33924
BIP8130I: Execution Group: EG01  -  58480
BIP8130I: Execution Group: EG03  -  29906
BIP8130I: Execution Group: default  -  43814
BIP8071I: Successful command completion.
$ mqsilist BKA2
BIP8130I: Execution Group: EG02  -  69272
BIP8130I: Execution Group: default  -  46340
BIP8130I: Execution Group: EG03  -  38296
BIP8130I: Execution Group: EG01  -  65750
BIP8071I: Successful command completion.
```

4. Verify that the database is available by issuing the following command:

   `db2 connect to` *`broker_database`*

   Check the database connection information, as shown in Example 24-4 on page 450.

Example 24-4 Testing the database connection

```
$ db2 connect to BKA1DB user tstadmin using password
 Database Connection Information
 Database server        = DB2/6000 8.2.0
 SQL authorization ID   = TSTADMIN
 Local database alias   = BKA1DB
$ db2 connect to BKA2DB user tstadmin using password
 Database Connection Information
 Database server        = DB2/6000 8.2.0
 SQL authorization ID   = TSTADMIN
 Local database alias   = BKA2DB
```

5. To reset the DB2 connection, enter the following command:

```
db2 disconnect current
```

Premigration verification

Check the error log files for Message Broker V5.0 components, queue manager and DB2 for any important information or errors.

For details about which error log files to check and how to check them, refer 27.1.4, “Checking the logs and where to find them” on page 505.

Backing up the broker configuration

Ensure that the broker configuration is successfully backed up.

To learn how to back up the broker configuration refer to Chapter 19, “Backing up the WebSphere Business Integration Message Broker V5.0 domain” on page 339.

24.1.2 Installing WebSphere Message Broker Version 6.0

Install and verify WebSphere Message Broker V6.0 installation, if this has not already been done.

Perform the installation according to standard installation documentation. The description of the same is outside the scope of this book. Check whether the following products are installed:

- Rational Agent Controller V6.0.0.1
- WebSphere Message Broker V6.0 components:
  - Broker
  - Transformation Services

Tip: We recommend that you install the latest fix pack for WebSphere Message Broker V6.0 and the appropriate Rational Agent Controller version.

To install any fix pack for Rational Agent Controller, refer to WebSphere Message Broker V6.0 documentation.

WebSphere Message Broker V6.0 requires the correct version of Rational Agent Controller. Any other version is not supported.

24.2 Configuration settings

Before starting the migration, you must make some changes to the configuration as explained in the following sections.

Before you make any changes to the configuration, you must stop the broker. First, log in with the Message Broker service user ID. Then enter the following command from the appropriate platform command prompt:

```
mqsistop broker_name
```

When the broker is stopped, you see a message like the one in Example 24-5.

Example 24-5 Stopping the broker

```
$ mqsistop BKA1
BIP8071I: Successful command completion.
```

24.2.1 Updating environment settings

The environment settings for the service user ID have changed in Message Broker V6.0, so you have to change the user profile. This initialization ensures that all commands that issued interact with the correct installation of the code.

The full path of the new Message Broker sample profile is *broker_install_directory*/6.0/bin/mqsiprofile.

To edit the Message Broker service user ID profile using an editor such as vi, do the following:

- Remove the WebSphere Business Integration Message Broker V5.0 settings. This may be a call to the sample profile, in which case delete the following line:

  ```
  . /usr/opt/mqsi/sample/profiles/profile.aix
  ```

- Add the following line:

  `. broker_install_directory/6.0/bin/mqsiprofile`

  Example 24-6 shows the relevant part of the user profile.

Example 24-6 Sample of the user profile

```
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:/usr/java131
/jre/bin:.

export PATH

. broker_install_directory/6.0/bin/mqsiprofile
```

> **Note:** There are a number of new commands available with the WebSphere Message Broker V6.0 Configuration Manager installation. These include:
>
> - `mqsibackupconfigmgr`
> - `mqsirestoreconfigmgr`
> - `mqsistartmsgflow`
> - `mqsistopmsgflow`
>
> These commands utilize the Configuration Manager Proxy application programming interface (API).
>
> Therefore, in order to execute the new commands on UNIX platforms, you should add the shipped Java 1.4.2 Java Runtime Environment (JRE) to the PATH environment variable in the mqsiprofile file. You must add *install_directory*/jre/bin to the PATH environment.

- Log out and log in with the Message Broker service user ID again to activate these changes.

> **Important:** If WebSphere Business Integration Message Broker V5.0 and WebSphere Message Broker V6.0 have to coexist, then both versions of the broker profile cannot be in the user profile at the same time. Leave the user profile without the broker profile and manually enter the appropriate version of the broker profile.
>
> Use the appropriate version of the Message Broker profile for starting each broker.

24.2.2 Updating ODBC settings

The next step is to modify the settings for the Open Database Connectivity (ODBC) driver. Before you make any modifications, you must first make a backup copy of the .odbc.ini file.

For DB2, be sure to verify the ODBC settings. The new product contains new versions of the ODBC drivers for Oracle and Sybase.

The full path of the new WebSphere Message Broker V6.0 odbc.ini sample is *broker_install_directory*/6.0/merant/odbc.ini. Check the differences between the current .odbc.ini file and Message Broker V6.0 sample.

Using an editor such as vi, edit the .odbc.ini file and make the required changes by entering the following command:

```
vi /var/mqsi/odbc/.odbc.ini
```

Example 24-7 shows the relevant part of the .odbc.ini file.

Example 24-7 Sample of the .odbc.ini

```
[BKA1DB]
Driver=/db2data/db2inst1/sqllib/lib/libdb2.a
Description=Migration DB
Database=BKA1DB
[BKA2DB]
Driver=/db2data/db2inst1/sqllib/lib/libdb2.a
Description=Migration DB
Database=BKA2DB
```

Note: For more information about the ODBC setting details, select the path **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 to WebSphere Message Broker Version 6.0 → Migrating components on different computers → Migrating the broker → Distributed operating systems → Changing the ODBC connection and XA resource manager definitions for a migrated broker** in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13850_.htm

24.3 Migrating the broker

This section describes the steps to follow for the component migration.

24.3.1 Before starting

Before migrating a broker, ensure that you do not have any aggregations in progress. When migrating a broker to Message Broker V6.0, any live data being stored for aggregations in progress is lost.

Also, verify that the Message Broker service user ID uses the supported locale and code page.

Tip: For more information, select the path **Reference** → **Installation** → **National Language Support** in the WebSphere Message Broker 6.0 Information Center available on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an08720_.htm`

24.3.2 Beginning the migration

To migrate a Message Broker V5.0 broker on distributed operating systems to Message Broker V6.0 at the same location, follow these steps.

Log in with the Message Broker service user ID and launch the commands from an appropriate platform command prompt as explained here:

1. Stop all channels communicating to the broker queue manager.
2. Run the appropriate environment settings, as shown in Example 24-8.

Example 24-8 Running Message Broker V6.0 profile

```
$ . broker_install_directory/6.0/bin/mqsiprofile
```

3. Run a premigration check on the broker by entering the following Message Broker V6.0 command:

   ```
   mqsimigratecomponents -c broker_name
   ```

   Example 24-9 on page 455 shows a successful premigration check on the broker.

Example 24-9 Testing the broker migration

```
$ mqsimigratecomponents -c BKA1
BIP8849I: Broker 'BKA1' (Version 5.0) with Queue Manager 'BKA1QM' and Data
Source 'BKA1DB' specified for migration.
BIP8680I: Pre-migration check succeeded.
BIP8071I: Successful command completion.
```

4. Migrate the broker to Message Broker V6.0 by entering the following command:

   `mqsimigratecomponents` *`broker_name`*

   Example 24-10 shows the successful migration of the broker.

Example 24-10 Migrating the broker

```
$ mqsimigratecomponents BKA1
BIP8849I: Broker 'BKA1' (Version 5.0) with Queue Manager 'BKA1QM' and Data
Source 'BKA1DB' specified for migration.
BIP8750I: Starting registry migration for component 'BKA1'.
BIP8755I: Copied value 'QueueManagerName' into the new location
BIP8755I: Copied value 'DataSourceName' into the new location
BIP8755I: Copied value 'ServiceUserId' into the new location
BIP8755I: Copied value 'ServicePassword' into the new location
BIP8755I: Copied value 'DataSourceUserId' into the new location
BIP8755I: Copied value 'DataSourcePassword' into the new location
BIP8755I: Copied value 'LilPath' into the new location
BIP8755I: Copied value 'ConfigurationTimeout' into the new location
BIP8755I: Copied value 'ConfigurationDelayTimeout' into the new location
BIP8755I: Copied value 'StatisticsMajorInterval' into the new location
BIP8762I: Copied registry value from 'HttpListenerPort' to HTTP value 'port';
set to '7080'
BIP8755I: Copied value 'MigrationNeeded' into the new location
BIP8755I: Copied value 'MQTrustedQueueManager' into the new location
BIP8755I: Copied value 'UserNameServerQueueManagerName' into the new location
BIP8755I: Copied value 'PubSubAccessControl' into the new location
BIP8755I: Copied value 'AdminAgentPID' into the new location
BIP8755I: Copied value 'BrokerUUID' into the new location
BIP8762I: Copied registry value from 'HttpListenerTraceLevel' to HTTP value
'traceLevel'; set to 'none'
BIP8762I: Copied registry value from 'HttpListenerTraceSize' to HTTP value
'traceSize'; set to '4194304'
BIP8763I: Deleted value 'QueueManagerName' from the old location
BIP8763I: Deleted value 'DataSourceName' from the old location
BIP8763I: Deleted value 'ServiceUserId' from the old location
BIP8763I: Deleted value 'ServicePassword' from the old location
BIP8763I: Deleted value 'DataSourceUserId' from the old location
BIP8763I: Deleted value 'DataSourcePassword' from the old location
BIP8763I: Deleted value 'LilPath' from the old location
BIP8763I: Deleted value 'ConfigurationTimeout' from the old location
```

```
BIP8763I: Deleted value 'ConfigurationDelayTimeout' from the old location
BIP8763I: Deleted value 'StatisticsMajorInterval' from the old location
BIP8763I: Deleted value 'HttpListenerPort' from the old location
BIP8763I: Deleted value 'MigrationNeeded' from the old location
BIP8763I: Deleted value 'MQTrustedQueueManager' from the old location
BIP8763I: Deleted value 'UserNameServerQueueManagerName' from the old location
BIP8763I: Deleted value 'PubSubAccessControl' from the old location
BIP8763I: Deleted value 'AdminAgentPID' from the old location
BIP8763I: Deleted value 'BrokerUUID' from the old location
BIP8763I: Deleted value 'HttpListenerTraceLevel' from the old location
BIP8763I: Deleted value 'HttpListenerTraceSize' from the old location
BIP8768I: Finished registry migration for component 'BKA1'.
BIP8654I: Moving filesystem artefacts from '/var/mqsi' to '/var/mqsi'
BIP8655I: Removing '/var/mqsi/brokers/BKA1/bin/mqsistart'.
BIP8655I: Removing '/var/mqsi/brokers/BKA1/bin/mqsistop'.
BIP8655I: Removing '/var/mqsi/brokers/BKA1/bin'.
BIP8655I: Removing '/var/mqsi/brokers/BKA1'.
BIP8670I: Database migration started
BIP8663I: Creating temporary new tables
BIP8664I: Migrating from existing tables to temporary new tables
BIP8665I: Dropping existing tables
BIP8666I: Creating new tables
BIP8667I: Copying all rows from temporary new tables to new tables
BIP8668I: Dropping temporary new tables
BIP8669I: Database migration successful
WebSphere MQ queue manager running.
BIP8785I: Starting WebSphere MQ queue migration for component 'BKA1'.
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REQUEST'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.CONTROL'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REPLY'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.TIMEOUT'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.UNKNOWN'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.TIMEOUT.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.ADMIN.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.EXECUTIONGROUP.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.EXECUTIONGROUP.REPLY'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.WS.INPUT'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.WS.REPLY'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.WS.ACK'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.IPC.QUEUE'
BIP8789I: Finished WebSphere MQ queue migration for component 'BKA1'.
BIP8071I: Successful command completion
```

5. Run a post-migration check of your broker to Message Broker V6.0 by entering the following command:

   ```
   mqsimigratecomponents -v broker_name
   ```

   Example 24-11 shows a successful command output after running a post-migration check.

Example 24-11 Verifying the broker migration

```
$ mqsimigratecomponents -v BKA1
BIP8849I: Broker 'BKA1' (Version 6.0) with Queue Manager 'BKA1QM' and Data
Source 'BKA1DB' specified for migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

24.3.3 Validating the broker migration

Now that the broker migration is complete, your tasks on the AIX system are nearly complete, too. Start the broker and verify that it executes properly.

The commands for managing the broker are the same as in Message Broker V5.0.

Starting the broker

Log in with the Message Broker service user ID and enter the commands from the appropriate platform command prompt, as explained here:

1. Start all channels communicating to the broker queue manager.
2. Start your broker and enter the following command:

   ```
   mqsistart broker_name
   ```

   When the broker is started, you see a message like the one in Example 24-12.

Example 24-12 Starting the broker

```
$ mqsistart BKA1
WebSphere MQ queue manager running.
BIP8096I: Successful command initiation, check the system log to ensure that
the component started without problem and that it continues to run without
problem.
```

3. Display the list of WebSphere Message Broker components and enter the **`mqsilist -a`** command. The command should display the same components

as before migration. The version is also displayed for all previous version brokers, as shown in Example 24-13.

Example 24-13 Displaying the components

```
$ mqsilist -a
BIP8099I: Broker: BKA1  -  BKA1QM
BIP8221I: Broker: BKA2 (Version 5)  -  BKA2QM
BIP8099I: UserNameServer: UserNameServer  -  UNSQM
BIP8071I: Successful command completion.
```

Important: The previous versions of the broker are displayed only if the **-a** option is used.

4. Display the details of your broker and enter the following command:

   `mqsilist` *`broker_name`*

   The command should display the same list of the execution groups as before migration, as Example 24-14 shows.

Example 24-14 Displaying the execution groups

```
$ mqsilist BKA1
BIP8130I: Execution Group: EG02  -  51934
BIP8130I: Execution Group: EG01  -  43826
BIP8130I: Execution Group: EG03  -  48866
BIP8130I: Execution Group: default  -  58482
BIP8071I: Successful command completion.
```

Important: To display the details of the previous version brokers, use the command and environment settings from Message Broker V5.0.

5. Log out and log in with the Message Broker service user ID again. Set up an appropriate profile for Message Broker V5.0 and enter the following command:

   `mqsilist` *`broker_name`*

   Example 24-15 on page 459 shows the execution group for the previous version broker.

Example 24-15 Displaying the execution group for previous version broker

```
$ . /usr/opt/mqsi/sample/profiles/profile.aix
$ mqsilist BKA2
BIP8130I: Execution Group: EG02  -  69272
BIP8130I: Execution Group: default  -  46340
BIP8130I: Execution Group: EG03  -  38296
BIP8130I: Execution Group: EG01  -  65750
```

Verifying the migration

Verify the error log files after the migration to ensure that there were no errors during migration.

For details about which error log files should be checked and how to check them, refer to 27.1.4, “Checking the logs and where to find them” on page 505.

24.3.4 Verifying the migration for Rules and Formatter Extension

If the new WebSphere Message Broker V6.0 is using the Rules and Formatter Extension, then perform the following steps:

1. Check the environment variable NNSY_ROOT by entering the following command in the command line:

   ```
   echo $NNSY_ROOT
   ```

   If it does not already exist, set NNSY_ROOT to point to the root directory where the Rules and Formatter Extension is installed. For example, add the following line to the user profile:

   ```
   export NNSY_ROOT=/opt/IBM/rf41/rfe
   ```

2. Check the ODBC database settings; refer to 24.2.2, “Updating ODBC settings” on page 453.

   The nnsyreg.dat sample file of New Era Of Networks is provided in the examples directory; for example:

   rfe_install_dir/rfe/examples/NNSYRF

   Check the differences between the current nnsyreg.dat file and the Message Broker V6.0 sample.

   Edit the nnsyreg.dat file and change the NNOT_SHARED_LIBRARY parameter in all sessions.

   Example 24-16 shows a section of the nnsyreg.dat file.

Example 24-16 nnsyreg.dat sample file

```
Session.MQSI_CONFIG
   NNOT_SHARED_LIBRARY    = dvdb41db2
   NNOT_FACTORY_FUNCTION  = NNSesDB2Factory
   NN_SES_SERVER          = NNDB
   NN_SES_USER_ID         = tstadmin
   NN_SES_PASSWORD        = password
```

New Era Of Networks searches for the current nnsyreg.dat file in the following directories:

- Current working directory
- Directory specified by the environment variable NN_CONFIG_FILE_PATH
- Directory specified by the environment variable NNSY_ROOT

Ensure that the correct configuration file is used.

25

Migrating the z/OS broker

This chapter describes the migration of WebSphere Business Integration Message Broker V5.0, running on z/OS, to WebSphere Message Broker V6.0.

The migration of the broker is described as part of a larger domain migration that is detailed in Chapter 17, “WebSphere Business Integration Message Broker Version 5.0 environment setup” on page 313. After you complete the steps described in this chapter, the z/OS broker will be ready to operate in a migrated topology.

25.1 Migration scenario

Here we explain how to migrate broker MQZ2BRK from WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0.

Figure 25-1 illustrates the broker scenario before migration and after migration.

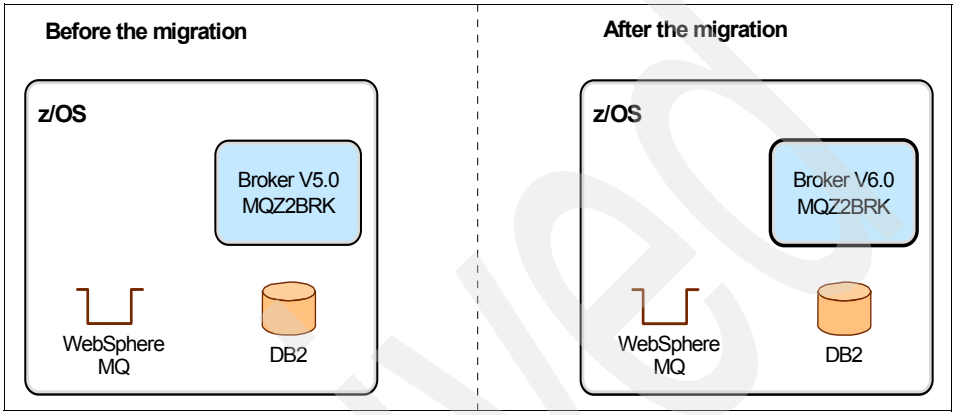


Figure 25-1 How the z/OS broker looks after the migration

Note: This chapter explains the migration of the broker in a number of steps. Follow these steps in order to ensure a successful migration.

25.2 Environment preparation

Before beginning the migration process, you must perform several steps. These steps involve preparing the environment for migration, checking the WebSphere Message Broker prerequisites, and backing up the current WebSphere Business Integration Message Broker V5.0 domain. In the following sections, we explain these tasks in more detail.

25.2.1 Verify the current environment

Ensure that the current environment of WebSphere Business Integration Message Broker V5.0 is functioning correctly, as this will assist in a smooth migration to WebSphere Message Broker V6.0.

To check that the environment is running properly, review the job log for the WebSphere Business Integration Message Broker V5.0 task. In the example this chapter presents, we check the started task MQZ2BRK to ensure that no errors

are present. We also check the WebSphere MQ job logs to ensure that there are no errors.

You must resolve any errors you encounter before the migration begins. Any problems with the broker, DB2, or WebSphere MQ at WebSphere Business Integration Message Broker V5.0 causes problems when migrating to WebSphere Message Broker V6.0.

25.2.2 Stop the broker

You must stop the broker before you create a backup. This ensures that no changes occur between the backup taken and the migration completion.

Also, you must stop the broker in order for the migration job to run.

25.2.3 Back up the broker

Ensure that the broker database and the broker registry are successfully backed up before the broker is migrated. This allows the broker to get restored to the current WebSphere Business Integration Message Broker V5.0 level, if required.

Important: Any change made to the broker database or the broker registry after the backup has taken place (such as new subscriptions that are made or new resources that are deployed) is lost if the backup is restored.

For details about how to back up the broker on z/OS, refer to Chapter 19, “Backing up the WebSphere Business Integration Message Broker V5.0 domain” on page 339.

25.2.4 Check the system requirements

The system requirements for WebSphere Message Broker V6.0 are different from those for WebSphere Business Integration Message Broker V5.0. Before you begin the migration you must review the system requirements, as outlined on the following Web site, and install any necessary prerequisites:

http://www.ibm.com/support/docview.wss?rs=849&uid=swg27006594

Also, you must check the Preventive Service Planning (PSP) that is referred to in the Program Directory of the product that is installed.

25.2.5 Check user ID authorities

The authorities for the broker administrator user ID and the broker started task user ID should already be configured correctly for the WebSphere Business Integration Message Broker V5.0 domain. The migration will encounter problems if the correct authorities are not assigned.

To ensure the authority that the user ID requires is available, you must refer to the WebSphere Message Broker information center under the topic **Security** → **Setting up z/OS security**, available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae14030_.htm

25.2.6 Install WebSphere Message Broker Version 6.0

This chapter does not cover the installation instructions for WebSphere Message Broker V6.0. You can find detailed instructions for installation on z/OS in the online documentation in the WebSphere Message Broker Information Center under the topic **Installing** → **Installation Guide**, available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah24100_.htm

However, be sure to check that WebSphere Message Broker V6.0 code has been installed according to the installation guide instructions. Also, verify that the following products have been installed:

- WebSphere Message Broker V6.0

> **Tip:** We recommend installation of the latest fix pack for WebSphere Message Broker V6.0 and the appropriate Rational Agent Controller (RAC) version.

Locate the SBIPPROC and the SBIPSAMP template Partitioned Data Set Extended (PDSE), which are required in later steps, to check the installation of the broker. Ask the systems administrator for the location of these data sets.

You can check the hierarchical file system (HFS) for the WebSphere Message Broker V6.0 code. By default this is located in /usr/lpp/mqsi/V6M0R0. However, to determine the exact location, check with the person responsible for installation.

The template PDSEs and HFS installation are required before migration can begin.

Note: Ensure that APAR OA11699 is applied to the system before installation. This allows you to view the output from WebSphere Message Broker V6.0 customization jobs in the joblogs.

If APAR OA11699 is not applied to the system, you can redirect the output from broker jobs to the HFS to help work around the problem. For more information, refer to “Redirect STDOUT and STDERR” on page 473 for more information.

- Rational Agent Controller V6.0.0.1

  If the remote debugging of message flows using the Message Brokers Toolkit is required on the z/OS broker, then you must install the Rational Agent Controller (RAC). The RAC, by default, installs to /usr/lpp/IBM/AgentController.

Important: Refer to WebSphere Message Broker V6.0 documentation to ensure installation of the correct version of Rational Agent Controller.

WebSphere Message Broker V6.0 requires an exact version of Rational Agent Controller. Any other version is not supported.

25.3 Premigration tasks

After you verify the environment, perform the premigration configuration tasks. Begin by gathering information about the current system. (This is essential for configuring the broker JCL in later stages.)

Next, allocate a new PDSE for the migrated broker JCL and the configuration files. Finally, copy the template JCL provided with WebSphere Message Broker V6.0 to the new dataset and customize it for the local system.

25.3.1 Collect the required information for the broker

The mqsicompcif file used to configure the WebSphere Business Integration Message Broker V5.0 contains most of the necessary information required to fill out the tables that follow. For any missing information, contact the person responsible for that component of the system.

Collect reference information

Table 25-1 lists the installation information required for your reference.

Table 25-1 Installation information for your reference

Description	JCL variable	The redbook variables	Your installation variable
Fully qualified name of the product's SBIPPROC dataset	N/A	<hlq>.SBIPPROC	
Fully qualified name of the product's SBIPSAMP dataset	N/A	<hlq>.SBIPSAMP	

Collect JCL information

Table 25-2 lists the broker runtime information required for JCL customization.

Table 25-2 Broker runtime information for JCL customization

Description	JCL variable	The redbook variables	Your installation variable
File system directory where the product has been installed	++INSTALL++	/usr/lpp/mqsi/V6M0R0	
HFS directory where the broker is to exist	++COMPONENTDIRECTORY++	/var/wmqi/MQZ2BRK	
Broker name	++COMPONENTNAME++	MQZ2BRK	
The broker user ID HFS home directory	++HOME++	/u/mqz2brk	
The mqsicreatebroker options	++OPTIONS++	‘’ **Note:** No options were specified, so two single quotes are specified.	
Locale of environment where commands are run by submitting JCL	++LOCALE++	C	
Time zone of environment where commands are run by submitting JCL	++TIMEZONE++	GMT0BST	
Location of Java installation	++JAVA++	/usr/lpp/java/J1.4.2	

Description	JCL variable	The redbook variables	Your installation variable
LE high-level-qualifier	++LEHLQ++	PP.ADLE370.ZOS150	
WebSphere MQ high-level-qualifier	++WMQHLQ++	MQM.V531	
Queue manager associated with the broker	++QUEUEMANAGER++	MQZ2	
Dataset where all JCL relevant to the broker is saved	++COMPONENTDATASET++	MQSI6.MQZ2BRK.CNTL	
Name of the Started Task JCL that can be a maximum of 8 characters	++STARTEDTASKNAME++	MQZ2BRK	
Profile name	++COMPONENTPROFILE++	BIPBPROF	
Location of the XML toolkit	++XMLTOOLKIT++	/usr/lpp/ixm/IBM/xml4c-5_5	

Important: WebSphere Message Broker V6.0 requires version 1.8 of the XML Parser for z/OS. To obtain the latest version, refer to the XML Toolkit Web site:

http://www-03.ibm.com/servers/eserver/zseries/software/xml/cparser/install.html

Collect DB2 information

Table 25-3 lists the DB2 information to be collected.

Table 25-3 DB2 information

Description	JCL variable	The redbook variables	Your installation variable
Specifies the DB2 converter	++DB2CONVERSION++	SINGLE	
DB2 subsystem identifier	++DB2SUBSYSTEM++	DB8W	
DB2 location value of the DB2 subsystem	++DB2LOCATION++	DB8W	

Description	JCL variable	The redbook variables	Your installation variable
DB2 table owner user ID	++DB2TABLEOWNER++	MQZ2BRK	
DB2 user ID for the component and commands	++DB2CURRENTSQLID++	SDRES02	
DB2 plan name	++DB2DSNACLIPLAN++	DSNACLI	
DB2 high-level qualifier	++DB2HLQ++	DB8W8	
DB2 run library value	++DB2RUNLIB++	DB8WU.RUNLIB.LOAD	
DB2 program value	++DB2SAMPLEPROGRAM++	DSNTEP2	
DB2 plan value	++DB2SAMPLEPROGRAMPLAN++	DSNTEP81	
DB2 broker database	++DB2DATABASE++	DMQZ2BRK	
DB2 storage group	++DB2STORAGEGROUP++	SMQZ2BRK	
DB2 bufferpool	++DB2BUFFERPOOL++	BP0	
DB2 index bufferpool	++DB2INDEXBP++	BP0	
DB2 LOB bufferpool	++DB2LOBBP++	BP0	

These tables are taken from the WebSphere Message Broker Information Center and are replicated here for your convenience. The documentation version is found under the topic **Configuring the broker domain → Configuring broker domain components → Creating a broker → z/OS → Information required to create a broker**, available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp

25.3.2 Allocate the broker PDSE

This new data set stores the JCL and the configuration members for the migrated broker. The data set must exist so that you can copy the template JCL and the template configuration members into it.

Allocate the data set

Each broker requires either a PDSE or a partitioned data set (PDS). However, a PDSE is preferable to a PDS because free space is available without the need to compress the data set.

Create the broker component dataset using option 3.2 on Interactive System Productivity Facility (ISPF). The name of the PDSE must be the same as the JCL variable ++COMPONENTDATASET++ collected in a previous step in 25.3.1, “Collect the required information for the broker” on page 465.

Allocate a data set with the following:

- Eight directory blocks
- 15 tracks (or 1 cylinder) of 3390 DASD with a record format of fixed block 80
- A suitable block size (for example, 27920)
- Data set type of LIBRARY

The information for creating the broker's PDSE was taken from the WebSphere Message Broker Information Center under the topic **Configuring the broker domain → Configuring broker domain components → Creating a broker → z/OS → Creating the broker PDSE**, available on the Web at:

```
http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae22410_.htm
```

In our migration scenario, we allocated the data set shown in Example 25-1 for the migrated broker to use.

Example 25-1 Example broker data set used in the migration scenario

```
MQSI6.MQZ2BRK.CNTL
```

Customize the broker data set

Copy the members in PDSE SBIPPROC and PDSE SBIPSAMP to the new broker data set. Use the Data Set Utility panel (usually option 3.3 in ISPF) to copy the following members.

Copy the JCL from the following data set:

```
<hlq>.SBIPSAMP to ++COMPONENTDATASET++
```

Copy the following JCL from the previous data set:

- BIPBPROF (broker profile)
- BIPDSNAO (DB2 dsnaoini)

Copy the following members from:

```
<hlq>.SBIPPROC to ++COMPONENTDATASET++
```

Note that instead of copying the entire contents of the data set, you may choose to copy only the following broker members:

- BIPALDB: Job to alter DB2 table spaces and tables
- BIPALMQ: Job to alter WebSphere MQ table spaces and tables
- BIPBRWS: Job to run the **mqsibrowse** command
- BIPCHBK: Job to enter the **mqsichangebroker** command
- BIPCHMS: Job to enter the **mqsichangeflowstats** command
- BIPCHPR: Job to enter the **mqsichangeproperties** command
- BIPCLMP: Job to enter the **mqsiclearmqpubsub** command
- BIPCRBK: Job to create a broker
- BIPCRDB: Job to create the DB2 storage group, database, and table spaces
- BIPDLBK: Job to enter the **mqsideletebroker** command
- BIPDLDB: Job to delete the DB2 storage group, database, and table spaces
- BIPEDIT: JCL customization
- BIPFMLG: Job to format the log
- BIPGEN: Generate ENVFILE
- BIPJNMP: Job to enter the **mqsijoinmqpubsub** command
- BIPLIST: Job to enter the **mqsilist** command
- BIPLSMP: Job to enter the **mqsilistmqpubsub** command
- BIPMGTB: Job to change DB2 definitions when migrating from V5.0 to V6.0
- BIPRELG: Job to read logs
- BIPRPMS: Job to enter the **mqsireportflowstats** command
- BIPRPPR: Job to enter the **mqsireportproperties** command
- BIPSDBP: Job to define a data source, user ID, and password for user data sources
- BIPBRKP (started task)

Important: Customize the following files for your broker:

- Rename BIPEDIT to a unique name that identifies it to the current component (for example, MQZ2EDBK).
- Rename BIPBRKP to the same as the ++STARTEDTASKNAME++ (it is MQZ2BRK in the example shown).

This process has been taken from the WebSphere Message Broker Information Center topic and replicated for convenience. The most up-to-date version is found under the topic **Configuring the broker domain → Configuring broker domain components → Creating a broker → z/OS → Customizing the broker component data set**, available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae22450_.htm

25.3.3 Customize the broker JCL

The copy of the JCL templates in the broker component dataset require customization for the environment in which they run. To replace the JCL variables in all the members of the broker component dataset, you can use the ISPF macro BIPEDIT that, in this example, is renamed to MQZ2EDBK. To use this macro, perform the following steps.

Customize the BIPEDIT member

Edit the renamed BIPEDIT member and replace the example parameters with the variables collected in 25.3.1, “Collect the required information for the broker” on page 465.

For instance, the ++INSTALL++ JCL parameter would look like Example 25-2 in the renamed BIPEDIT member.

Example 25-2 Editing the BIPEDIT member to replace the variables

```
"change ++INSTALL++ /usr/lpp/mqsi/V6R0M0                    all"
```

Important: You must update all JCL variables listed in the renamed BIPEDIT member. If any are left blank or are incorrectly configured, the JCL will not work correctly.

You may also add a substitution for the job card information at the top of the broker JCL. For instance, the first line of the BIPCRBK job is //BIPCRBK JOB.

You can use the renamed BIPEDIT macro to replace the word JOB with a suitable job card for your system. For example, add the following line to the renamed BIPEDIT macro and add any extra job parameters you require as shown in Example 25-3.

Example 25-3 Replacing the job card information

```
"change ' JOB ' ' JOB 1,MSGCLASS=H'                    all "
```

Activate the macro for execution

After the renamed BIPEDIT member is correctly configured, it can run against the members of the component dataset to replace the JCL variables in each member.

To activate the macro for execution, run the following Time Sharing Option (TSO) command in the ISPF panel called option 6:

```
ALTLIB ACTIVATE APPLICATION(EXEC) DA('++COMPONENTDATASET++')
```

This command is active for the local ISPF session for which it was issued.

Note: If you have split-screen sessions, the other sessions are not able to use this. If you use ISPF option 6 to issue the command, use ISPF option 3.4 to edit the data set. This enables you to use the edit command.

Replace the JCL variables

Now that the ISPF macro is active, it is possible to replace the JCL variables for each member in the broker component dataset.

For each member of the component dataset (except the renamed BIPEDIT macro itself), follow these steps:

1. View or Edit the member.

   Tip: It is advisable to use View instead of Edit on the first member until you resolve all problems in the ISPF macro. Alternatively, it is possible to use Cancel on the Edit session instead of using Save, if you encounter problems.

2. Type the name of the member on the command line (for example, `MQZ1EDBK`) to run the renamed BIPEDIT.exec to replace the JCL variables with the values you desire.
3. Exit and save the changes if they have been successful.

Tip: You can add the word Save to your macro name to save your changes. For example, you could type (or paste) `MQZ2EDBK;save` on the command line to replace the variables, and then save the member.

The JCL members in the component dataset are now ready to run. However, before they are run, ensure that you perform the following final steps.

Update BIPBPROF and BIPDSNAO

Before moving onto the next stage, we recommend that you copy any additional changes that you have made from the WebSphere Business Integration Message Broker V5.0 environment to the new WebSphere Message Broker V6.0 members.

Perform the following steps:

- Copy any changes made to the WebSphere Business Integration Message Broker V5.0 ENVFILE into the BIPBPROF member.
- Copy any changes made to the ODBC initialization file dsnaoini into the BIPDSNAO member.

The configuration of the broker component dataset is now correct. It is now possible to migrate the broker.

Redirect STDOUT and STDERR

If APAR OA11699 has not been applied to the system, then it is possible to redirect the output from the broker jobs to the HFS.

To do this, change the //STDOUT and //STDERR DD cards as shown in Example 25-4.

Example 25-4 Redirecting output from the broker jobs to the HFS

```
//STDOUT   DD PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//            PATHMODE=(SIRWXU,SIRWXG),
//            PATH='/u/mqz1brk/redirect.out'
//STDERR   DD PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//            PATHMODE=(SIRWXU,SIRWXG),
//            PATH='/u/mqz1brk/redirect.err'
```

Rename the files in the HFS so that each job creates its own file. This helps you to debug jobs after they have run.

25.4 Migrate the broker

Before beginning the broker migration, ensure that all the steps in 25.3, “Premigration tasks” on page 465 are successfully completed. The following steps require that the JCL members in the broker component dataset are all correctly configured, security privileges are correctly assigned, prerequisite software is installed, and the broker is successfully backed up.

25.4.1 Before starting

This section lists tasks that need verification before starting the migration.

Aggregations

Before migrating a broker, ensure that you do not have any aggregations in progress. When you migrate a broker to Message Broker V6.0, any live data stored for aggregations that are in progress are lost.

Check bipimain

Use the **extattr** command to display the attributes of the object `bipimain`, for example:

```
extattr /usr/lpp/mqsi/V6ROMO/bin/bipimain
```

It should show APF authorized = YES. If it does not, use **extattr +a bipimain** to set this attribute; for example:

```
extattr +a /usr/lpp/mqsi/V6ROMO/bin/bipimain
```

25.4.2 Create the broker home and component directories

If they do not already exist, create the home and component directories in the HFS. They are defined as ++HOME++ and ++COMPONENTDIRECTORY++ in 25.3.1, “Collect the required information for the broker” on page 465.

Ensure that the broker start task user ID and the broker system administrator’s user ID have read, write, and execute permissions to both the ++COMPONENTDIRECTORY++ and the ++HOME++ directory.

25.4.3 Generate the broker ENVFILE

The first step in the broker migration is to create the ENVFILE (an environment file that the broker stores all its environment variables in). When you start the components, they read the ENVFILE to configure their environment.

Next, submit the BIPGEN job in the broker component dataset. This creates the ENVFILE in the ++HOME++ directory on the HFS.

Ensure that the BIPGEN job runs successfully and that you create the ENVFILE in the ++HOME++ directory. If any errors occur, resolve them and then run the BIPGEN job again. Continue running the job and rectifying errors until you successfully create the ENVFILE. The migration cannot proceed until BIPGEN generates an ENVFILE.

Important: If any changes are made to the BIPBPROF member, then you must rerun the BIPGEN job to generate a new ENVFILE.

The running of the BIPGEN job is done at the same time as a planned restart of the broker. This maintains consistency between the ENVFILE in ++HOME++ and the running components that use the copy.

25.4.4 Perform a test migration of the broker

Before the **mqsimigratecomponents** command is run properly, you can execute it in check mode. This performs a premigration check to see if the command, when run properly, is likely to succeed.

To run this check, edit the BIPMGCMP job in the broker dataset and alter the command defined at the bottom of the member. You should change the command to specify the -c parameter, as shown in Example 25-5.

Example 25-5 Specifying the -c parameter in the BIPMGCMP job

```
BPXBATCH PGM -
/usr/lpp/mqsi/V6ROMO/bin/-
mqsimigratecomponents -
MQW1BRK -
-c
```

Submit the edited BIPMGCMP and check the output of the job. If successful, the job should contain the message shown in Example 25-6.

Example 25-6 Expected output from the BIPMGCMP job

```
BIP8680I: Pre-migration check succeeded.
```

If the migration check fails, rectify the cause of the failure and then run the migration check again. Continue this process until the migration check succeeds.

Note: You can run the **mqsimigratecomponents** command with a full set of command line options. For a full list of options, refer to the WebSphere Message Broker Information Center under the topic **Reference** → **Operations** → **Commands** → **Runtime Commands** → **mqsimigratecomponents command**, available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an26150_.htm

25.4.5 Migrate the broker components

After the success of the migration check, the **mqsimigratecomponents** command is run without the check parameter. This causes the command to perform a full migration of the broker.

You can configure the **mqsimigratecomponents** command to migrate the file system, WebSphere MQ, or the broker database in separate stages. This is useful if the user ID performing the migration does not have the required authority to change WebSphere MQ or the broker database tables. In this situation, a different user with the correct authorities to migrate that component can configure and run a job.

To perform any of the migration steps individually, use the parameters of the **mqsimigratecomponents** command, as listed in Table 25-4.

Table 25-4 Commands to migrate individual broker components

Parameter	Migrates component
-1	Do only registry and file system work. Use the -1 parameter before the -2 or -3 parameters.
-2	Do only WebSphere MQ work.
-3	Do only database work.

To perform the actual migration, edit the BIPMGCMP job in the broker dataset and alter the command defined at the bottom of the member. Remove the -c option that is specified in the previous step. To migrate a single component, use the parameters shown in Table 25-4. To migrate all components, run the command with no parameters except for the broker name.

Tip: Ensure that you also remove the continuation character at the end of the previous line.

Ensure that the BIPMGCMP job has succeeded and that the components have successfully migrated. Check for the following messages from the job's output, as shown in Example 25-7.

Example 25-7 Output from BIPMGCMP job running in non check mode

```
BIP8768I: Finished registry migration for component 'MQZ2BRK'.
Registry data has been successfully migrated for component 'MQZ2BRK'.
BIP8669I: Database migration successful
BIP8789I: Finished WebSphere MQ queue migration for component 'MQZ2BRK'. Queue
data has been successfully migrated for component 'MQZ2BRK'.
BIP8071I: Successful command completion.
```

Also check the job's output to see if any message flows or message sets require redeployment.

Note: If the BIPMGCMP job fails to run succesfully, then you should restore the backup taken before the migration began. This restores the broker to its previous WebSphere Business Integration Message Broker V5.0 level. Refer to 27.2, "Restoring the WebSphere Business Integration Message Broker Version 5.0 environment" on page 509 for the steps to restore this backup.

After the restoration is complete, rerun the BIPMGCMP job.

25.4.6 Copy the started task JCL

The SYS1.PROCLIB concatenation already contains members for the WebSphere Business Integration Message Broker V5.0 started task. Rename or back these up before you copy the new WebSphere Message Broker V6.0 member in.

Copy the renamed BIPBRKP member (in this example, MQZ2BRK) to the SYS1.PROCLIB concatenation. The broker is now ready to start.

25.4.7 Start the migrated broker

You can issue the command to start or stop a broker from SDSF, NetView, or other products that provide access to the console. As shown in the following examples, the commands are given for SDSF, and have the forward slash (/) command prefix. This may be different on other tools.

From SDSF, start the component (for example, /S MQZ2BRK).

Check the log for the started task to ensure that the component starts succesfully. The log should display messages as shown in Example 25-8.

Example 25-8 Expected broker startup messages

```
+BIP9141I MQZ2BRK 0 THE COMPONENT WAS STARTED. : ImbControlService(744)

+BIP9108I MQZ2BRK 0 BROKER SERVICE VALUE IS IMBSERV.V6ROM00.GOLD..... :
ImbControlService(749)

+BIP2001I MQZ2BRK 0 THE WEBSPHERE MESSAGE BROKERS SERVICE HAS STARTED AT
VERSION 6000; PROCESS ID 852085. :
ImbControlService(773)
```

After the broker starts, any execution groups that were migrated also start. Check logs to ensure they succesfully start. The execution group logs should contain the information shown in Example 25-9.

Example 25-9 Execution group start message

```
+BIP2201I MQZ2BRK EG1 0 EXECUTION GROUP STARTED: PROCESS
'84738167';
```

More startup information follows this message. After the execution group is successfully up and running, the final message is as shown in Example 25-10.

Example 25-10 Execution group startup completion message

```
+BIP2154I MQZ2BRK EG1 26 EXECUTION GROUP FINISHED WITH CONFIGURATION
MESSAGE
```

However, if the component fails to start correctly, use the error information in the log to rectify the problem. Ensure that you perform all of the prerequisite steps in this chapter and that these steps are successful.

Note: When the main broker task (MQZ2BRK) starts up, it executes the step VFYDB2MQ. This ensures that the broker has access to DB2, WebSphere MQ, and the HFS.

25.4.8 Verify the migration

At this stage, the broker and its execution groups have succesfully started and the VFYDB2MQ step has been executed. To verify that the broker has succesfully migrated, refer to Chapter 27, “Verifying migration to WebSphere Message Broker V6.0” on page 495.

25.4.9 Migration verification for Rules and Formatter Extension

If the new WebSphere Message Broker V6.0 is using the Rules and Formatter Extension, then examine the ENVFILE in the ++HOME++ directory to check the environment variable NNSY_ROOT.

If it does not exist in the ENVFILE, add NNSY_ROOT to the BIPBPROF member and execute the BIPGEN job to generate a new ENVFILE. For example, add the following line to BIPBPROF:

```
NNSY_ROOT=/usr/lpp/mqsi/V6R0M0/nnsy
```

You can find the sample file nnsyreg.dat of New Era Of Networks in the /usr/lpp/mqsi/V6R0M0/nnsy/rulfmt41/examples directory. Check the differences between the current nnsyreg.dat file and the Message Broker V6.0 sample.

Edit the nnsyreg.dat file and change the NNOT_SHARED_LIBRARY parameter in all sessions.

New Era Of Networks searches for the current nnsyreg.dat file in the following directories:

1. Current working directory
2. Directory specified by the environment variable NN_CONFIG_FILE_PATH
3. Directory specified by the environment variable NNSY_ROOT

Be sure the correct configuration file is used.

25.5 Creating a new broker on z/OS

The steps this chapter covers, discuss the migration of a broker from WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0. If a new broker is required on z/OS then you should consult the WebSphere Message Broker Information Center under the topic **Configuring the broker domain > Configuring broker domain components → z/OS**, available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae22400_.htm

26

Migrating the Linux broker

This chapter describes the migration of a WebSphere Business Integration Message Broker V5.0 broker running on Linux with a DB2 data source to WebSphere Message Broker V6.0. It guides you through the migration of the Linux broker in the scenario described in Chapter 17, “WebSphere Business Integration Message Broker Version 5.0 environment setup” on page 313. After completing the steps in this chapter, the broker is ready to operate in the migrated domain.

26.1 Migration preparation

This section describes the tasks to be performed before the broker component migration. Figure 26-1 illustrates the Linux broker V5.0 scenario before and after the migration.

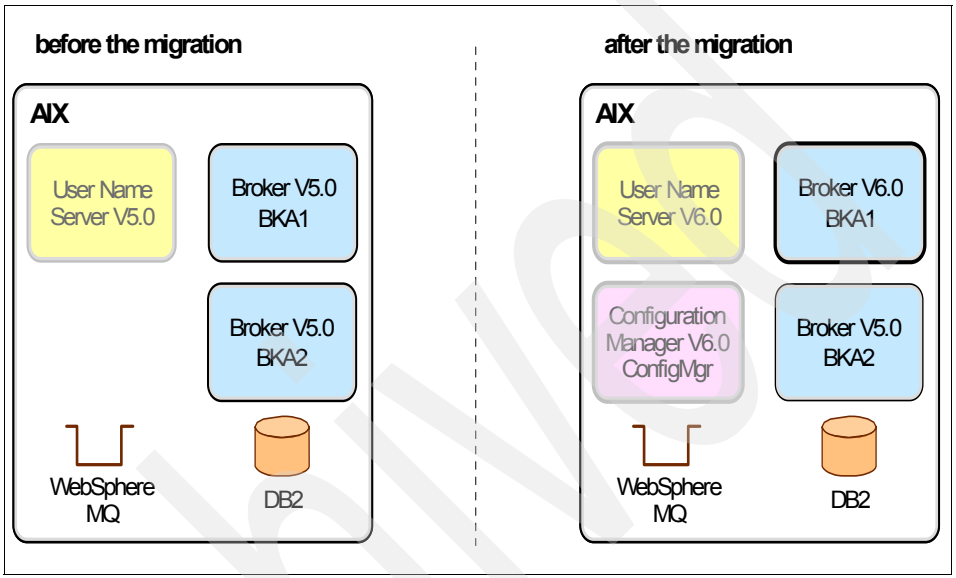


Figure 26-1 Linux broker V5.0 migration scenario

26.1.1 Reviewing the existing installation

Check the existing installation for errors. Prior to migrating, ensure that the broker is running properly and that no errors are reported.

Tip: On the Linux platform, use the `script /tmp/output.txt` command to make a typescript of everything displayed on your terminal to a specified file (to /tmp/output.txt, in this case).

Reviewing the configuration

To confirm that all important components are running properly, perform the following steps and check the output.

Log in with the Message Broker service user ID and enter the commands from the command line as explained in the following steps:

1. Display the list of queue managers and enter the **dspmq** command. Check the status of all expected queue managers, as shown in Example 26-1.

Example 26-1 Displaying the queue managers

```
$ dspmq
QMNAME(BKL1QM)                                            STATUS(Running)
```

2. Display the list of Message Broker V5.0 components and enter the **mqsilist** command. Check the status of all expected components, as shown in Example 26-2.

Example 26-2 Displaying the components

```
$ mqsilist
BIP8099I: BKL1  -  BKL1QM
BIP8071I: Successful command completion.
```

3. Display the detail of your broker and enter the following command:

   `mqsilist broker_name`

   Check all expected execution groups, as shown in Example 26-3.

Example 26-3 Displaying the execution groups

```
$ mqsilist BKL1
BIP8130I: Execution Group: EG03  -  2359
BIP8130I: Execution Group: default  -  2286
BIP8130I: Execution Group: EG01  -  2363
BIP8130I: Execution Group: EG02  -  2365
BIP8071I: Successful command completion.
```

4. Check if the database is available and enter the following command:

   `db2 connect to broker_database`

   Check the database connection information, as shown in Example 26-4.

Example 26-4 Testing the database connection

```
$ db2 connect to BKL1DB
Database Connection Information
Database server        = DB2/LINUX 8.2.0
SQL authorization ID   = TSTADMIN
Local database alias   = BKL1DB
```

5. Reset the DB2 connection and enter the following command:

   `db2 disconnect current`

Premigration verification

Check the error log files for Message Broker V5.0 components, queue manager, and DB2 for any important information or errors.

For details about which error log files should be checked and how to check them, refer to 27.1.4, “Checking the logs and where to find them” on page 505.

Backing up the broker configuration

Ensure that the broker configuration is successfully backed up.

To learn how to back up the broker configuration refer to Chapter 19, “Backing up the WebSphere Business Integration Message Broker V5.0 domain” on page 339.

26.1.2 Installing WebSphere Message Broker Version 6.0

Install and verify the WebSphere Message Broker V6.0 installation (if this has not already been done). Perform the installation according to standard installation documentation.

Verify that the following products are installed:

- Rational Agent Controller V6.0.0.1
- WebSphere Message Broker V6.0 components:
  - Broker
  - Transformation Services

> **Tip:** We recommend that you install the latest fix pack for WebSphere Message Broker V6.0 and the appropriate Rational Agent Controller version (refer to WebSphere Message Broker V6.0 documentation for the correct version of Rational Agent Controller, as it does not support any other version).

26.2 Configuration settings

Before starting the component migration you must make some changes to the configuration, as explained in the following sections. Before making changes to the configuration, however, you must first stop the broker.

To stop the broker, log in with the Message Broker service user ID. Then enter the following command from the appropriate platform command prompt:

```
mqsistop broker_name
```

When the broker is stopped, you will receive a message like the one shown in Example 26-5.

Example 26-5 Stopping the broker

```
$ mqsistop BKL1
BIP8071I: Successful command completion.
```

26.2.1 Updating environment settings

The environment settings for the service user ID have changed in Message Broker V6.0, so you have to change the user profile. This initialization ensures that all the commands issued interact with the correct installation of the code.

Refer to the following steps for updating the user profile for WebSphere Message Broker V6.0 on Linux:

1. Find and open the .bash_profile in the home directory of the service user ID with an editor such as vi. If there are lines like those shown in Example 26-6, you can work with .bashrc instead.

   Example 26-6 shows the relevant part of the user profile.

Example 26-6 Running .bashrc from .bash_profile

```
if [ -f ~/.bashrc ]; then
        . ~/.bashrc
fi
```

2. Change broker_install_directory to the directory in which you installed WebSphere Message Broker V6.0 (for example, /opt/ibm/mqsi/6.0). Add the following line to the .bash_profile or .bashrc at the end of the file:

   ```
   . broker_install_directory/bin/mqsiprofile
   ```

3. Log out and log in with the Message Broker service user ID again.

4. Ensure that the environment settings are correct. You can see some environment variables by pointing to the path where WebSphere Message Broker V6.0 is installed. Example 26-7 shows the environment variables.

Example 26-7 Checking the environment settings

```
$ env | grep -i mqsi
MQSI_REGISTRY=/var/mqsi
MQSI_VERSION=6.0.0.0
LD_LIBRARY_PATH=/opt/mqm/java/lib:/opt/ibm/mqsi/6.0/lib:/opt/ibm/mqsi/6.0/bin:/
opt/ibm/mqsi/6.0/merant/lib:/opt/ibm/mqsi/6.0/jre/bin/classic:/opt/ibm/mqsi/6.0
/jre/bin:/opt/ibm/mqsi/6.0/xml4c/lib:/home/tstadmin/sqllib/lib
MQSI_WORKPATH=/var/mqsi
MQSI_CATALINA_HOME=/opt/ibm/mqsi/6.0/catalina
NLSPATH=/opt/ibm/mqsi/6.0/messages/%L/%N:/opt/ibm/mqsi/6.0/messages/En_US/%N:
PATH=/opt/ibm/mqsi/6.0/bin:/opt/ibm/MessageBrokersToolkit/6.0/eclipse:/usr/kerb
eros/bin:/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin:/home/tstadmin/sqllib/bin:
/home/tstadmin/sqllib/adm:/home/tstadmin/sqllib/misc:/home/tstadmin/bin:/usr/lo
cal/mozilla
MQSI_FAD=3
MQSI_JARPATH=/opt/ibm/mqsi/6.0/classes:/opt/ibm/mqsi/6.0/messages
MQSI_DEVELOPMENT=/var/mqsi/registry
MQSI_FILEPATH=/opt/ibm/mqsi/6.0
MQSI_LILPATH=/opt/ibm/mqsi/6.0/lil:/opt/ibm/mqsi/6.0/jplugin
ODBCINI=/var/mqsi/odbc/.odbc.ini
CLASSPATH=/opt/ibm/mqsi/6.0/classes/derby.jar:/opt/ibm/mqsi/6.0/classes/ConfigM
anagerProxy.jar:/opt/ibm/mqsi/6.0/classes/configutil.jar:/opt/mqm/java/lib/com.
ibm.mq.jar:/opt/mqm/java/lib/connector.jar:/opt/ibm/mqsi/6.0/messages:/home/tst
admin/sqllib/java/db2java.zip:/home/tstadmin/sqllib/java/db2jcc.jar:/home/tstad
min/sqllib/java/sqlj.zip:/home/tstadmin/sqllib/function:/home/tstadmin/sqllib/j
ava/db2jcc_license_cisuz.jar:/home/tstadmin/sqllib/java/db2jcc_license_cu.jar:.
MQSI_EXMLTCONFIGPATH=/opt/ibm/mqsi/6.0/exmltConfig
DISTHUB_PATH=/opt/ibm/mqsi/6.0
```

Note: There are a number of new commands available with the WebSphere Message Broker V6.0 Configuration Manager installation. These include:

- mqsibackupconfigmgr
- mqsirestoreconfigmgr
- mqsistartmsgflow
- mqsistopmsgflow

They utilize the Configuration Manager Proxy application programming interface (API). Therefore, in order to execute the new commands on UNIX platforms, you should add the shipped Java 1.4.2 Java Runtime Environment (JRE) to the PATH environment variable in the mqsiprofile file.

You must add install_directory/jre/bin to the PATH environment variable.

26.2.2 Updating ODBC settings

The next step is to modify the settings for the Open Database Connectivity (ODBC) driver. Before you make any modifications, you must first make a backup copy of the .odbc.ini file. For DB2, be sure to verify the ODBC settings. The new product contains new versions of the ODBC drivers for Oracle and Sybase. Example 26-8 shows a sample odbc.ini file for the ODBC driver configuration for Linux.

Example 26-8 A sample odbc.ini file content

```
[ODBC Data Sources]
BKL1DB=IBM DB2 ODBC Driver
RESERVDB=IBM DB2 ODBC Driver
USERDB=IBM DB2 ODBC Driver
MQSIBKDB=IBM DB2 ODBC Driver
MYDB=IBM DB2 ODBC Driver
ORACLEDB=DataDirect 5.0 Oracle
SQLSERVERDB=DataDirect 5.0 SQL Server Wire Protocol
SYBASEDB=DataDirect 5.0 Sybase Wire Protocol
INFORMIXDB=DataDirect 5.0 Informix Wire Protocol

[BKL1DB]
Driver=/home/tstadmin/sqllib/lib/libdb2.so
Description=BKL1DB DB2 ODBC Database
Database=BKL1DB

[RESERVDB]
Driver=/home/tstadmin/sqllib/lib/libdb2.so
Description=RESERVDB DB2 ODBC Database
Database=RESERVDB

[USERDB]
Driver=/home/tstadmin/sqllib/lib/libdb2.so
[USERDB]
Driver=/home/tstadmin/sqllib/lib/libdb2.so
Description=USERDB DB2 ODBC Database
Database=USERDB

[MQSIBKDB]
Driver=/opt/IBM/db2/V8.1/lib/libdb2.so
Description=MQSIBKDB DB2 ODBC Database
Database=MQSIBKDB

[MYDB]
Driver=/home/tstadmin/sqllib/lib/libdb2.so
Description=MYDB DB2 ODBC Database
Database=MYDB

[ORACLEDB]
Driver=/opt/mqsi/merant/lib/UKor820.so
```

```
Description=DataDirect 5.0 Oracle
EnableDescribeParam=1
OptimizePrepare=1
ServerName=Oracle host
WorkArounds=536870912
WorkArounds2=2

[SYBASEDB]
Driver=/opt/mqsi/merant/lib/UKase20.so
Description=DataDirect 5.0 Sybase Wire Protocol
Database=sybasedb
ServerName=YourServerName
EnableDescribeParam=1
OptimizePrepare=1
SelectMethod=0
NetworkAddress=10.30.14.72,5000
SelectUserName=1

[SQLSERVERDB]
Driver=/opt/mqsi/merant/lib/UKmsss20.so
Description=DataDirect 5.0 SQL Server Wire Protocol
Address=SQLServer host,SQLServer server port
AnsiNPW=Yes
Database=db
LogonID=UID
Password=PWD
QuotedId=No

[INFORMIXDB]
Driver=/opt/mqsi/merant/lib/UKifcl20.so
Description=DataDirect 5.0 Informix Wire Protocol
ApplicationUsingThreads=1
CancelDetectInterval=0
Database=db
HostName=Informix host
LogonID=UID
Password=PWD
PortNumber=Informix server port
ServerName=Informix server
TrimBlankFromIndexName=1

[ODBC]
Trace=0
TraceFile=/var/mqsi/odbc/odbctrace.out
TraceDll=/opt/mqsi/merant/lib/odbctrac.so
InstallDir=/opt/mqsi/merant
UseCursorLib=0

IANAAppCodePage=4
```

Note: For more information about the ODBC setting details, select the path **Migrating → Migrating from Version 2.1 products → Migrating from WebSphere MQ Integrator Broker Version 2.1 to WebSphere Message Broker Version 6.0 → Migrating components on different computers → Migrating the broker → Distributed operating systems → Changing the ODBC connection and XA resource manager definitions for a migrated broker** in the WebSphere Message Broker V6.0 information center, available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah13850_.htm

26.3 Migrating the broker

This section details the steps for the component migration.

26.3.1 Before starting

Before migrating a broker, ensure that you do not have any aggregations in progress. When migrating a broker to Message Broker V6.0, any live data being stored for aggregations in progress is lost.

Also, check if the Message Broker service user ID uses the supported locale and code page.

Tip: For more information, select the path **Reference → Installation → National Language Support** in the WebSphere Message Broker V6.0 Information Center available on the Web at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an08720_.htm

26.3.2 Migrating the broker

To migrate a Message Broker V5.0 broker on distributed operating systems to Message Broker V6.0 at the same location, perform the following steps.

Log in with the Message Broker service user ID and launch the commands from an appropriate platform command prompt as follows:

1. Stop all channels communicating to the broker queue manager.

2. Run a premigration check on the broker by entering the Message Broker V6.0 command:

   ```
   mqsimigratecomponents -c broker_name
   ```

   Example 26-9 shows a successful premigration check on the broker.

Example 26-9 Testing the broker migration

```
$ mqsimigratecomponents -c BKL1
BIP8849I: Broker 'BKL1' (Version 5.0) with Queue Manager 'BKL1QM' and Data
Source 'BKL1DB' specified for migration.
BIP8680I: Pre-migration check succeeded.
BIP8071I: Successful command completion.
```

3. Migrate the broker to Message Broker V6.0 by entering the following command:

   ```
   mqsimigratecomponents broker_name
   ```

   Example 26-10 shows a successful command output.

Example 26-10 Migrating the broker

```
$ mqsimigratecomponents -s 5.0.0.6 -t 6 BKL1
BIP8849I: Broker 'BKL1' (Version 5.0) with Queue Manager 'BKL1QM' and Data
Source 'BKL1DB' specified for migration.
BIP8750I: Starting registry migration for component 'BKL1'.
BIP8755I: Copied value 'QueueManagerName' into the new location
BIP8755I: Copied value 'DataSourceName' into the new location
BIP8755I: Copied value 'ServiceUserId' into the new location
BIP8755I: Copied value 'ServicePassword' into the new location
BIP8755I: Copied value 'DataSourceUserId' into the new location
BIP8755I: Copied value 'DataSourcePassword' into the new location
BIP8755I: Copied value 'LilPath' into the new location
BIP8755I: Copied value 'ConfigurationTimeout' into the new location
BIP8755I: Copied value 'ConfigurationDelayTimeout' into the new location
BIP8755I: Copied value 'StatisticsMajorInterval' into the new location
BIP8755I: Copied value 'MigrationNeeded' into the new location
BIP8755I: Copied value 'MQTrustedQueueManager' into the new location
BIP8755I: Copied value 'UserNameServerQueueManagerName' into the new location
BIP8755I: Copied value 'PubSubAccessControl' into the new location
BIP8755I: Copied value 'AdminAgentPID' into the new location
BIP8755I: Copied value 'BrokerUUID' into the new location
BIP8762I: Copied registry value from 'HttpListenerTraceLevel' to HTTP value
'traceLevel'; set to 'none'
BIP8762I: Copied registry value from 'HttpListenerTraceSize' to HTTP value
'traceSize'; set to '4194304'
BIP8762I: Copied registry value from 'HttpListenerPort' to HTTP value 'port';
set to '7080'
BIP8763I: Deleted value 'QueueManagerName' from the old location
```

```
BIP8763I: Deleted value 'DataSourceName' from the old location
BIP8763I: Deleted value 'ServiceUserId' from the old location
BIP8763I: Deleted value 'ServicePassword' from the old location
BIP8763I: Deleted value 'DataSourceUserId' from the old location
BIP8763I: Deleted value 'DataSourcePassword' from the old location
BIP8763I: Deleted value 'LilPath' from the old location
BIP8763I: Deleted value 'ConfigurationTimeout' from the old location
BIP8763I: Deleted value 'ConfigurationDelayTimeout' from the old location
BIP8763I: Deleted value 'StatisticsMajorInterval' from the old location
BIP8763I: Deleted value 'MigrationNeeded' from the old location
BIP8763I: Deleted value 'MQTrustedQueueManager' from the old location
BIP8763I: Deleted value 'UserNameServerQueueManagerName' from the old location
BIP8763I: Deleted value 'PubSubAccessControl' from the old location
BIP8763I: Deleted value 'AdminAgentPID' from the old location
BIP8763I: Deleted value 'BrokerUUID' from the old location
BIP8763I: Deleted value 'HttpListenerTraceLevel' from the old location
BIP8763I: Deleted value 'HttpListenerTraceSize' from the old location
BIP8763I: Deleted value 'HttpListenerPort' from the old location
BIP8768I: Finished registry migration for component 'BKL1'.
BIP8654I: Moving filesystem artefacts from '/var/mqsi' to '/var/mqsi'
BIP8655I: Removing '/var/mqsi/brokers/BKL1/bin/mqsistart'.
BIP8655I: Removing '/var/mqsi/brokers/BKL1/bin/mqsistop'.
BIP8655I: Removing '/var/mqsi/brokers/BKL1/bin'.
BIP8655I: Removing '/var/mqsi/brokers/BKL1'.
BIP8670I: Database migration started
BIP8663I: Creating temporary new tables
BIP8664I: Migrating from existing tables to temporary new tables
BIP8665I: Dropping existing tables
BIP8666I: Creating new tables
BIP8667I: Copying all rows from temporary new tables to new tables
BIP8668I: Dropping temporary new tables
BIP8669I: Database migration successful
WebSphere MQ queue manager running.
BIP8785I: Starting WebSphere MQ queue migration for component 'BKL1'.
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REQUEST'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.CONTROL'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REPLY'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.TIMEOUT'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.AGGR.UNKNOWN'
The setmqaut command completed successfully.
BIP8786I: Created WebSphere MQ queue 'SYSTEM.BROKER.TIMEOUT.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.ADMIN.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.EXECUTIONGROUP.QUEUE'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.EXECUTIONGROUP.REPLY'
```

```
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.WS.INPUT'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.WS.REPLY'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.WS.ACK'
BIP8787I: Cleared WebSphere MQ queue 'SYSTEM.BROKER.IPC.QUEUE'
BIP8789I: Finished WebSphere MQ queue migration for component 'BKL1'.
BIP8071I: Successful command completion.
```

4. Run a post-migration check on your broker to Message Broker V6.0 by entering the following command:

   ```
   mqsimigratecomponents -v broker_name
   ```

   Example 26-11 shows a successful command output.

Example 26-11 Verifying the broker migration

```
$ mqsimigratecomponents -v BKL1
BIP8849I: Broker 'BKL1' (Version 6.0) with Queue Manager 'BKL1QM' and Data
Source 'BKL1DB' specified for migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

26.3.3 Validating the broker migration

Now that the broker migration is complete, your tasks on the Linux system are nearly complete, too. Start the broker and verify that it executes properly.

The commands for managing the broker are the same as in Message Broker V5.0.

Starting the broker

Log in with the Message Broker service user ID and launch the commands from the appropriate platform command prompt as follows:

1. Start all channels communicating to the broker queue manager.
2. Start your broker and enter the following command:

   ```
   mqsistart broker_name
   ```

   When the broker is started, you see a message like the one shown in Example 26-12.

Example 26-12 Starting the broker

```
$ mqsistart BKL1
WebSphere MQ queue manager running.
BIP8096I: Successful command initiation, check the system log to ensure that
the component started without problem and that it continues to run without
problem.
```

3. Display the list of WebSphere Message Broker components and enter the **mqsilist** command. The command should display the same components as before migration, as Example 26-13 shows.

Example 26-13 Displaying the components

```
$ mqsilist
BIP8099I: Broker: BKL1  -  BKL1QM
BIP8071I: Successful command completion.
```

> **Important:** The previous versions of the broker are displayed only if the -a option is used.

4. Display the details of your broker and enter the command:

   `mqsilist` *broker_name*

   The command should display the same list of the execution groups as before migration, as Example 26-14 shows.

Example 26-14 Displaying the execution groups

```
$ mqsilist BKL1
BIP8130I: Execution Group: EG03  -  6866
BIP8130I: Execution Group: default  -  6802
BIP8130I: Execution Group: EG01  -  6864
BIP8130I: Execution Group: EG02  -  6865
BIP8071I: Successful command completion.
```

> **Tip:** To display the details of the previous broker version, use the command and environment settings from Message Broker V5.0.

Verifying the migration

Verify the error log files after the migration to ensure that there were no errors during migration.

For details about which error log files should be checked and how to check them, refer to 27.1.4, “Checking the logs and where to find them” on page 505.

26.3.4 Verifying the migration for Rules and Formatter Extension

If the new WebSphere Message Broker V6.0 is using the Rules and Formatter Extension, then perform the following steps:

1. Check the environment variable NNSY_ROOT by typing the following command in the command line:

   ```
   echo $NNSY_ROOT
   ```

   Set NNSY_ROOT to point to the root directory where the Rules and Formatter Extension is installed, if it does not already exist. For example, add the following line to the user profile:

   ```
   export NNSY_ROOT=/opt/IBM/rf41/rfe
   ```

2. Check the ODBC database settings. Refer to 26.2.2, “Updating ODBC settings” on page 487.

   The nnsyreg.dat sample file of New Era Of Networks is provided in the examples directory; for example:

   rfe_install_dir/rfe/examples/NNSYRF

   Check the differences between the current nnsyreg.dat file and the Message Broker V6.0 sample.

   Edit the nnsyreg.dat file and change the NNOT_SHARED_LIBRARY parameter in all sessions.

   Example 26-15 shows a section of the nnsyreg.dat file.

Example 26-15 nnsyreg.dat sample file

```
Session.MQSI_CONFIG
   NNOT_SHARED_LIBRARY    = dvdb41db2
   NNOT_FACTORY_FUNCTION  = NNSesDB2Factory
   NN_SES_SERVER          = NNDB
   NN_SES_USER_ID         = tstadmin
   NN_SES_PASSWORD        = password
```

   The New Era Of Networks searches for the current nnsyreg.dat in the following directories:

   – Current working directory
   – Directory specified by the environment variable NN_CONFIG_FILE_PATH
   – Directory specified by the environment variable NNSY_ROOT

   Ensure that the correct configuration file is used.

27

Verifying migration to WebSphere Message Broker V6.0

This chapter discusses the tasks that verify whether the migration to WebSphere Message Broker V6.0 has been successful. It also details the procedures to be performed to restore the WebSphere Business Integration Message Broker V5.0 environment from backups.

The verification and restoration of migration are based on the environment described in Chapter 17, “WebSphere Business Integration Message Broker Version 5.0 environment setup” on page 313.

27.1 Post-migration tasks

After completing the migration to WebSphere Message Broker V6.0, perform verification tasks such as:

- Verifying the components by listing, starting, and stopping the following:
  - Configuration Manager
  - User Name Server
  - Brokers
- Verifying the development environment by:
  - Starting the Message Brokers Toolkit
  - Assigning and deploying the resources to brokers
- Checking if the migrated message flows and message sets are working as expected
- Reviewing the logs for problems

If the migration process was unsuccessful, restore the backups of the previous version. The following list includes the components that should be restored:

- The WebSphere Business Integration Message Broker V5.0 components and their configuration
- The related databases
- The development resources
- The user environment

After verifying, remove or uninstall the items that are no longer required. These include:

- The Configuration Manager database
- The old versions of products

There may also be post-migration verification tasks specific to your environment that should be performed. For more details about the tasks that are not covered here, refer to the WebSphere Message Broker V6.0 Information Center on the Web:

`http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp`

The following sections describe how to implement these verification, restoration, and clean-up tasks.

27.1.1 Verifying the components

To verify that the migration was successful, you can perform a basic checkpoint by starting and stopping the WebSphere Message Broker V6.0 components without failure. The WebSphere Message Broker V6.0 components can be controlled with a set of commands.

However, before you run any WebSphere Message Broker V6.0 commands, enter the appropriate command to set up your environment, as described here.

- On Windows platforms, enter the following command from the command prompt:

  broker_install_directory/bin/mqsiprofile.cmd

  Alternatively, you can launch the command console, which runs mqsiprofile.cmd automatically at startup, by selecting **Start → IBM WebSphere Message Broker V6.0 → Command Console**.

- On UNIX and Linux platforms, enter the following command on the command line:

  broker_install_directory/bin/mqsiprofile

If a problem arises while controlling the components with commands, check the related logs to ensure that the components are running correctly. Refer to 27.1.4, “Checking the logs and where to find them” on page 505 for information about the related logs on each platform.

Note: To control a component on the z/OS platform, issue commands from the System Display and Search Facility (SDSF), NetView, or other products that provide access to the console. In the examples that follow, the commands are given for the SDSF, and have the forward slash (**/**) command prefix. This may be different for other tools.

Verifying the use of the mqsimigratecomponents command

The **mqsimigratecomponents** command is provided in the WebSphere Message Broker V6.0 for checking, migrating, and verifying the components. By using this command with the -v and -t parameters, you can check whether:

- The correct database tables and queues exist for the specified version
- The registry is in the correct format for the specified version

After migrating the components to WebSphere Message Broker V6.0, enter the following command in the command line on the distributed platforms:

```
mqsimigratecomponents -v -t 6 component_name
```

Example 27-1 illustrates the use of this command.

Example 27-1 Using the mqsimigratecomponents command

```
$ mqsimigratecomponents -v -t 6 BKL1
BIP8849I: Broker 'BKL1' (Version 6.0) with Queue Manager 'BKL1QM' and Data
Source 'BKL1DB' specified for migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

On z/OS, the BIPMGCMP job is provided. Run this job in the broker's component data set. Specify the correct parameters at the bottom of this job for the **mqsimigratecomponents** command.

Listing the components

Run the appropriate commands for each system, as explained here.

- To list the components, use the **mqsilist** command on the distributed platforms. Enter the following command on the command line:

  ```
  mqsilist
  ```

- On z/OS, enter the **list** command:

  ```
  /F broker_name list
  ```

  Type your broker name in place of *broker_name.*

> **Note:** The **list** command on z/OS is used to list all the execution groups defined to a specific broker, or to list all the message flows contained in a named execution group on a specific broker.
>
> It is different from the **mqsilist** command on the distributed platforms, which lists all the components.
>
> For more information about the **list** command on z/OS, refer to the topic "mqsilist" by selecting the path **Reference** → **Operations** → **Commands** → **Runtime commands** → **mqsilist** in the WebSphere Message Broker V6.0 Information Center available on the Web:
>
> http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an07250_.htm

Starting and stopping the Configuration Manager

In WebSphere Message Broker V6.0, the Configuration Manager can now run on all supported run time platforms. Run the appropriate commands for each system, as explained in the following sections.

Distributed platforms

You can start and stop the Configuration Manager on the distributed platforms as follows:

- To start the Configuration Manager, enter the following command on the command line:

  ```
  mqsistart configuration_manager_name
  ```

- To stop the Configuration Manager, enter the following command on the command line:

  ```
  mqsistop configuration_manager_name
  ```

In both of these commands, type your Configuration Manager's name in place of *configuration_manager_name*.

z/OS system

You can start and stop the Configuration Manager on z/OS as follows:

- To start the Configuration Manager, enter the following command:

  ```
  /S configuration_manager_name
  ```

- To stop the Configuration Manager, enter the following command:

  ```
  /P configuration_manager_name
  ```

In both of these commands, type your Configuration Manager's name in place of *configuration_manager_name*.

For more information about the start and stop command on z/OS, refer to the topic "z/OS" by selecting the path **Administering the broker domain** → **Starting and stopping a Configuration Manager** → **z/OS** in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae24130_.htm

Starting and stopping the User Name Server

Run the appropriate commands for each system as explained in the following sections.

Distributed platforms

You can start and stop the User Name Server on the distributed platforms as follows:

- To start the User Name Server, enter the following command on the command line:

  ```
  mqsistart UserNameServer
  ```

- To stop the User Name Server, enter the following command on the command line:

  ```
  mqsistop UserNameServer
  ```

z/OS system

You can start and stop the User Name Server on z/OS as follows:

- To start the User Name Server, enter the following command on the command line:

  ```
  /S user_name_server_name
  ```

- To stop the User Name Server, enter the following command on the command line:

  ```
  /P user_name_server_name
  ```

In both of these commands, type your User Name Server Name in place of *user_name_server_name*.

For more information about starting and stopping commands on z/OS, refer to the topic “z/OS” by selecting the path **Administering the broker domain** → **Starting and stopping a User Name Server** → **z/OS**) in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an04019_.htm

Starting and stopping a broker

Run the appropriate commands for each system as explained in the following sections.

Distributed platforms

To start and stop a broker on the distributed platforms, perform these tasks:

- To start a broker, enter the following command on the command line:

  ```
  mqsistart broker_name
  ```

- To stop a broker, enter the following command on the command line:

  ```
  mqsistop broker_name
  ```

In both of these commands, type your broker name in place of *broker_name*.

z/OS system

To start and stop a broker on z/OS, perform these tasks:

- To start a broker, enter the following command on the command line:

  ```
  /S broker_name
  ```

- To stop a broker, enter the following command on the command line:

  ```
  /P broker_name
  ```

In both of these commands, type your broker's name in place of *broker_name*.

For more information about the start and stop command on z/OS, refer to the topic "z/OS" by selecting the path **Administering the broker domain** → **Starting and stopping a broker** → **z/OS** in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/an04016_.htm

27.1.2 Verifying the development environment

In WebSphere Message Broker V6.0, the Message Brokers Toolkit can be installed on a Windows or a Linux (x86 platform). To ensure that the new version of the Message Brokers Toolkit is installed and the broker resources are migrated correctly, start the Message Brokers Toolkit before developing and deploying the applications.

Launching the Message Brokers Toolkit

The following sections explain how to launch the WebSphere V6.0 Message Brokers Toolkit on each platform. If you encounter problems while launching and working with the Message Brokers Toolkit, check the related logs. Refer to 27.1.4, "Checking the logs and where to find them" on page 505 for details about where to find the related logs on each platform.

Windows

On Windows, you need administrator authority to verify the installation. Therefore, it is advisable to complete the verification with the same user ID that you used for completing the installation.

To launch the Message Brokers Toolkit:

- Enter the following command on the command line:

  ```
  toolkit_install_directory/wmbt.exe
  ```

- Select **Start** → **IBM WebSphere Message Broker 6.0** → **WebSphere Message Brokers Toolkit** from the Windows Start menu.

Linux

On Linux, do *not* use root authority to launch the Message Brokers Toolkit. Log in using any user ID *except* root.

To launch the Message Brokers Toolkit:

- Enter the following command on the command line:

  toolkit_install_directory/wmbt.bin

- Select **WebSphere Message Brokers Toolkit** from the Linux main menu.

Tip: We recommend that you do not log into a UNIX or Linux system using a root account unless necessary. All authorized users, except root, can be used for WebSphere Message Broker V6.0 application development on Linux.

The reason for this is as follows: The root userid is generally reserved for system administration tasks because it has overall authority on all system resources. Therefore, it is general practice on UNIX and Linux systems to log on with a user ID that has the authorities needed to perform the tasks required and no more, in order to avoid accidental damage to other resources.

Configuring the location of a workspace

When you first launch the Message Brokers Toolkit, you will be prompted to specify the location of your workspace. You can either accept the default path, or specify your choice by locating it using the **Browse** button.

To inhibit the display of the workspace dialog the next time you launch the Message Brokers Toolkit, select the check box **Use this as the default and do not ask again**, as shown in Figure 27-1.

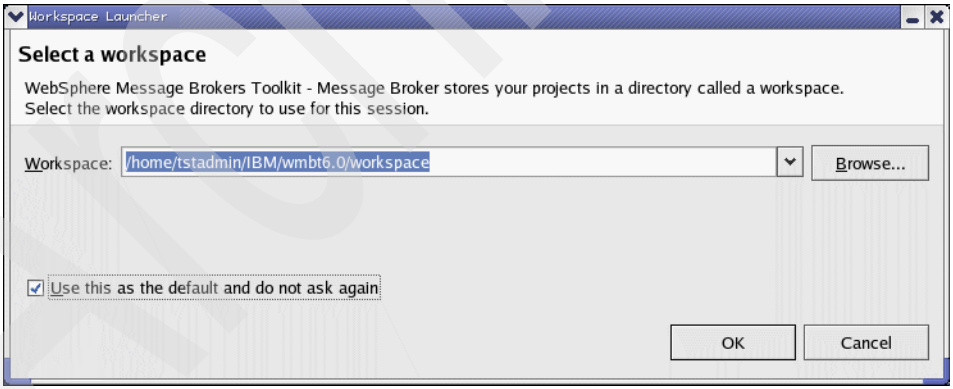


Figure 27-1 Inhibiting the subsequent display of the workspace dialog

Testing the development environment

The following sections provide information about testing the development and deployment processes with the help of the new Message Brokers Toolkit. If you encounter a problem with the Message Brokers Toolkit or while deploying to a broker, check the related logs. Refer to 27.1.4, “Checking the logs and where to

find them” on page 505 for details about where to find the related logs on each platform.

To test the development environment:

1. Launch the Message Brokers Toolkit.
2. Develop or import the WebSphere Message Broker resources in the Broker Application Development perspective.
3. Create a broker archive file and add the resources you want to deploy to the broker.
4. Switch to the Broker Administration perspective and connect to the WebSphere Message Broker domain.
5. Drag and drop the broker archive to the execution group in the broker.
6. Check if the deployment is successful.

For information about how to import, develop, and deploy the broker resources, refer to the topic “Developing applications” in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ac18710_.htm

Deploying the migrated applications

You can verify both the new development environment and the migration at the same time by deploying the applications that you migrated from the previous version of the broker to the new version.

Before you deploy anything to the broker, however, make sure that the previous version of the broker applications have been migrated according to the procedures described in Chapter 20, “Migrating WebSphere Message Broker V5.0 and V5.1 tools and resources” on page 353.

Using the samples

We recommend that you use the samples provided in WebSphere Message Broker V6.0 to verify the development environment, including:

- The WebSphere Message Broker Getting Started samples that provide:
  - Pager
  - Scribble
  - Soccer result
- The WebSphere Message Broker Application samples that provide:
  - Airline reservations
  - Coordinated request reply
  - Data warehouse

- Error handler
- Large messaging
- Message routing
- User-defined extension
- Video rental message set

For details about running each of these samples, in the Message Brokers Toolkit:

1. Select **Help** → **Samples Gallery** from the menu.
2. In the window that appears, browse and select one of the following locations in the left panel:
   - **Application samples** → **Message Brokers (Getting started)**
   - **Application samples** → **Message Brokers**

The features that each sample demonstrates are documented in the topic Samples, in the WebSphere Message Broker V 6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ax20230_.htm

27.1.3 Verifying the applications

To verify the migration using the samples provided with WebSphere Message Broker V6.0, and to run the sample applications, refer to the Samples topic in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ax20230_.htm

To test the environment, include all (or a subset) of the following functions in the verification scenario:

- WebSphere MQInput/MQOutput/Reply
- Compute/Mapping/Filter/XMLTransformation
- Database
- Aggregation
- Pub/Sub
- Subflows
- Realtime
- HTTP Input/Reply/Request
- New era of networks
- Message sets

If a problem occurs while running the applications, check the related logs to ensure that the components are running correctly. Refer to 27.1.4, “Checking the

logs and where to find them" on page 505 for details about where to find the related logs on each platform.

27.1.4 Checking the logs and where to find them

If you encounter problems while controlling the components, working with the Message Brokers Toolkit, and running the applications, check the logs to ensure that the migration or configuration is correct. The following sections explain where to find the appropriate logs for each WebSphere Message Broker V6.0 task.

Local error logs for WebSphere Message Broker components

If problems occur while running the WebSphere Message Broker V6.0 components such as brokers, check the local error log.

Windows systems: Event Viewer

On Windows, the application log contains events logged by applications or programs. To view the application log, enter the **eventvwr** command on the command line or select **Start** → **Control Panel** → **Administrative Tools** → **Event Viewer**.

This opens the Windows Event Viewer. In the left panel of this window, click **Application Log**. All events that have been logged by applications or programs, including WebSphere Message Broker V6.0, are displayed here.

Linux and UNIX systems: syslog

Before checking the syslog file, ensure that the syslog daemon is configured correctly to redirect the related syslog messages to a separate file. On the Linux and UNIX systems, all WebSphere Message Broker messages are sent to the syslog.

The entry indicating where all the WebSphere Message Broker messages are redirected to must be added to the /etc/syslog.conf file; for example:

```
user.info /var/log/user.log
```

In this case, the WebSphere Message Broker messages are found in /var/log/user.log.

z/OS: console log and joblog

On z/OS, the broker writes messages to the appropriate z/OS system log and joblogs.

The broker writes all its messages to the z/OS system console log, including messages from all address spaces running on the z/OS system. It is easy to

identify jobs associated with the broker in the console log because of the naming of broker address spaces.

Using the console log, you can see the order of event reporting for different products. This is helpful in cross-product problem determination.

The broker control address space, and each of the execution group address spaces, has its own joblog. Selecting the joblog for the appropriate broker address space allows you to see all the messages relating to that address space. This is helpful in a busy system where the system console log may have numerous messages from different products, thereby obscuring the relevant information.

Message Brokers Toolkit logs

In the Message Brokers Toolkit, the domain's event log should be reviewed after each deployment to the broker domain. The Eclipse log should be reviewed if problems arise while executing the Message Brokers Toolkit.

Message Brokers Toolkit event log

After the deployment to the broker domain, we recommend that you check the domain's event log to ensure that the deployment was successful. In the Message Brokers Toolkit:

1. Switch to the Broker Administration perspective.
2. Move to the Domains view and find the entry Event Log (with a flag icon).
3. Double-click the **Event Log** icon.

Eclipse log

If you encounter a problem with the Message Brokers Toolkit, review the Eclipse log to determine what the problem is. To view the Eclipse error log, in the Message Brokers Toolkit:

1. Select **Window** → **Open Perspective** → **Other** from the menu.
2. When a dialog box appears, check **Show all** and select **Plug-in Development** in the list, as illustrated in Figure 27-2.

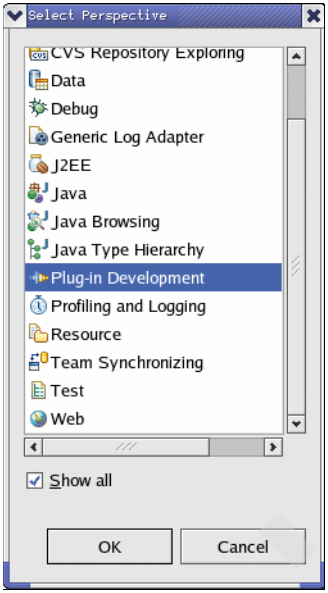


Figure 27-2 Opening the Plug-in Development perspective

3. When the Plug-in Development perspective is launched, in the Tasks view, click the **Error Log** tab. An error log is displayed, as illustrated in Figure 27-3.

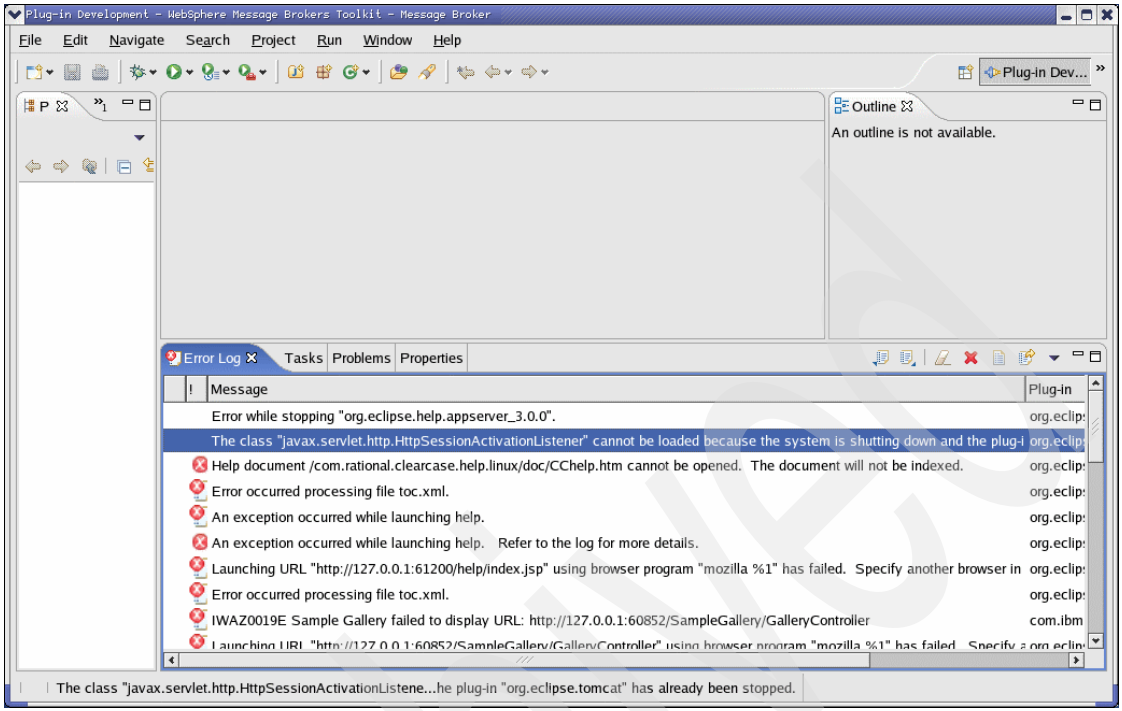


Figure 27-3 Eclipse error log view

4. If a plus (+) sign precedes an error, it means that it is a complex problem, and that there are a number of errors contributing to it. Click the **+** sign to view the individual errors.
5. To view the details pertaining to a particular problem, double-click the corresponding entry in the Tasks view. A separate window showing more details about the error is displayed, as illustrated in Figure 27-4.

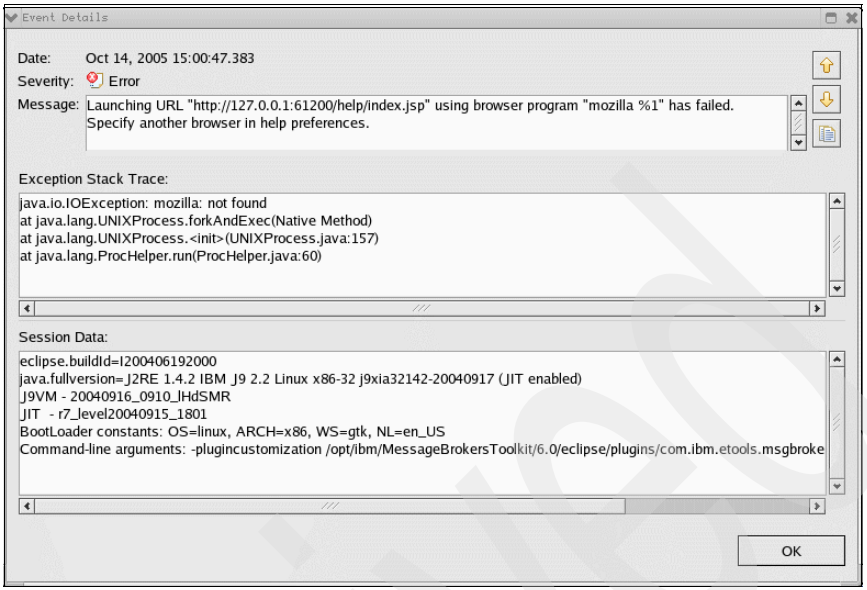


Figure 27-4 Eclipse Event Details view

WebSphere MQ logs

If you encounter problems relating to WebSphere MQ, refer to the topic “Problem Determination” in the WebSphere MQ Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmqv6/v6r0/topic/com.ibm.mq.amqzag.doc/mxprobl.htm

DB2 Universal Database logs

If you encounter problems relating to the DB2 Universal Database, refer to the topic “Analyzing db2diag.log files using db2diag” in the DB2 Universal Database Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/db2help/topic/com.ibm.db2.udb.pd.doc/pd/c0020701.htm

27.2 Restoring the WebSphere Business Integration Message Broker Version 5.0 environment

Restore the WebSphere Business Integration Message Broker V5.0 environment by using the backups taken earlier (refer to Chapter 19, “Backing up the WebSphere Business Integration Message Broker V5.0 domain” on page 339).

The following list displays the components that must be restored to return to the previous level:

- The Configuration Manager
- The Brokers
- The User Name Server
- The Workspace
- The User profile and the Open DataBase Connectivity (ODBC) connection file

In the sections that follow, the detailed restoration procedures for each component is discussed.

Important: Restoring the components to the previous level causes any changes made in the WebSphere Message Broker V6.0 environment since migration to be lost.

We recommend that you review the impact of the restoration before embarking on the changes.

27.2.1 Restoring the components

To restore the WebSphere Business Integration Message Broker V5.0 components, carry out the following tasks after referring to the appropriate produce documentation.

1. Ensure that all the deployment actions are complete and that the Message Brokers Toolkit is connected to the V6.0 broker domain.
2. Stop the brokers, the Configuration Manager, and the User Name Server by using the **mqsistop** command or the z/OS equivalent.
3. Delete the WebSphere Message Broker V6.0 brokers, Configuration Managers, and the User Name Server using the following options:
   a. To delete a broker, enter the **mqsideletebroker** command with an option **-w** to delete any trace files in the working path.

      For the z/OS equivalent, refer to the topic “z/OS” by selecting the path **Configuring the broker domain** → **Configuring broker domain components** → **Deleting a broker** → **z/OS** in the WebSphere Message Broker V6.0 Information Center available on the Web:

      http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae12970_.htm

   b. To delete the Configuration Manager, enter the **mqsideleteconfigmgr** command with the options **-w -n** to delete any trace files in the working path and its repository.

For the z/OS equivalent, refer to the topic “z/OS” by selecting the path **Configuring the broker domain** → **Configuring broker domain components** → **Deleting a Configuration Manager** → **z/OS** in the WebSphere Message Broker V6.0 Information Center available on the Web:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ae12970_.htm

4. If necessary, recreate the databases for the brokers and the Configuration Manager. For the DB2 Universal Database command on the distributed platforms, refer to Example 27-2.

Example 27-2 The DB2 Universal Database command for creating a database

```
$ db2 CREATE DB BKDB
DB20000I  The CREATE DATABASE command completed successfully.
```

5. Recreate V5.0 of the brokers, the Configuration Manager, and the User Name Server as follows:
   a. On the distributed platforms, use the **mqsicreatebroker**, **mqsicreateconfigmgr**, and **mqsicreateusernameserver** commands respectively.
   b. For the z/OS equivalent, refer to the V5.0 documentation.
6. Restore the databases for the brokers and the Configuration Manager according to the steps outlined in“Restoring the DB2 Universal Database” on page 512.
7. From the backup, restore the UUID of each broker by performing the appropriate operation for your platform:
   – On Windows, import the previously exported Windows registry file (*.reg) in the regedit tool by selecting **File** → **Import**.

     As an alternative, you can change the value of HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphereMQIntegrator\<broker_name>\BrokerUUID to the value you recorded before the migration.
   – On Linux and UNIX, copy the previously backed-up registry file to $MQSI_REGISTRY/registry/<broker_name>/BrokerUUID.

     As an alternative, you can change the value within the BrokerUUID file to the value you recorded before the migration.
   – On z/OS, copy the previously backed up registry file to $MQSI_REGISTRY/registry/<broker_name>/CurrentVersion/BrokerUUID.

     As an alternative, you can change the value within the BrokerUUID file to the value you recorded before the migration.

8. Start the brokers, the Configuration Manager, and the User Name Server using the **mqsistart** command. For the z/OS equivalent, refer to the V5.0 documentation. Check the related log if a problem occurs while starting these components.
9. Redeploy the configuration across the broker domain by launching the V 5.0 Message Brokers Toolkit and perform these tasks:
   a. Switch to the Broker Administration perspective.
   b. Right-click the **Domain** icon and select **Connect** in the Domain view if it is not already connected.
   c. Right-click **Broker Topology** and select **Deploy Topology Configuration → Complete.**

Restoring the DB2 Universal Database

Restoring the WebSphere Business Integration Message Broker V5.0 components may require an additional database restoration procedure. Before restoring the databases, ensure that they have been backed up correctly. Refer to 19.2, “Backing up components that use the DB2 Universal Database” on page 340 for information about how to back up the database.

The backup and restoration method to be used depends on your platform; following are the platform-specific examples.

UNIX, Linux, and Windows systems

To restore the databases required for the WebSphere Business Integration Message Broker V5.0 components on the distributed platforms:

1. Open an appropriate DB2 Universal Database command line.
2. Enter the appropriate commands to restore a database, as illustrated in Example 27-3.

Example 27-3 DB2 Universal Database commands for database restoration on the distributed systems

```
db2 CONNECT TO BKL1DB
db2 QUIESCE DATABASE IMMEDIATE FORCE CONNECTIONS
db2 PRUNE HISTORY 9999 WITH FORCE OPTION
db2 CONNECT RESET
db2 RESTORE DATABASE BKL1DB FROM "/home/tstadmin/backup/db2_backup" TAKEN AT
<time stamp noted from backup, like 20051010142634> WITH 2 BUFFERS BUFFER 1024
REPLACE EXISTING PARALLELISM 1 WITHOUT ROLLING FORWARD WITHOUT PROMPTING
db2 CONNECT TO BKL1DB
db2 UNQUIESCE DATABASE
db2 CONNECT RESET
```

This sequence of operations gives a 2359 warning message. However, this can be ignored because it means that an existing database is being replaced with a new one.

3. Verify whether the database is restored successfully by connecting to the database and the list tables.

z/OS system

Example 27-4 displays an example of JCL for restoring the database. This JCL sample can be downloaded from the Internet as described in Appendix C, “Additional material” on page 555.

Example 27-4 An example of JCL for restoring the database

```
//DBRECOVER JOB NOTIFY=&SYSUID,MSGLEVEL=(1,1),MSGCLASS=T
//PROCS JCLLIB ORDER=(XXXXXX.PROCLIB)
/*JOBPARM SYSAFF=XXXX
//UTIL EXEC DSNUPROC,SYSTEM=XXXX,UID='XXXXXXXX',UTPROC=''
//*********************************************
//*
//* JCL FOR THE IMAGE COPY UTILITY
//* (c)Copyright IBM Corp. 2002
//* DATE: 13/10/05
//* CREATED BY: DEANEJ
//* UPDATED BY: Mark Hiscock
//*
//* COPY THIS MEMBER INTO YOUR BROKER PDSE AND
//* EDIT THIS JOB AND UPDATE THE FOLLOWING:
//*
//* 1. The proclib for your DB2 (line 2)
//* 2. The system name for the job to run on (line 3)
//* 3. The DB2 sub system and user ID (line 4)
//* 4. The Dataset names (lines 30 and 32)
//* 5. The database name (line 40)
//* 6. Most IMPORTANTLY the LRSN value (line 41)
//*
//* THE JOBS SHOULD BE RUN IN THE FOLLOWING ORDER
//*
//* 1. BACKUP DB WITH DBBACKUP
//* 2. GET LRSN VALUE FOR FIC WITH DBREPORT
//* 3. RECOVER DB WITH DBRESTOR
//*
//*********************************************
//*
// SYSREC   DD DSN=WMQI.DMQW1BRK.STEP1.SYSREC,DISP=(MOD,DELETE,CATLG),
//          UNIT=SYSDA,SPACE=(8000,(20,20),,,ROUND)
// SYSUT1   DD DSN=WMQI.DMQW1BRK.STEP1.SYSUT1,DISP=(MOD,DELETE,CATLG),
//          UNIT=SYSDA,SPACE=(8000,(20,20),,,ROUND)
//*
```

```
//* TOLOGPPINT USES THE HEX VALUE OF THE LRSN FOR THE FULL IMAGE COPY
//* OF YOUR CHOICE. YOU CAN OBTAIN THIS BY RUNNING THE REPORT RECOVERY
//* JOB DBREPORT.
//*
//DSNUPROC.SYSIN DD *
LISTDEF COMPONENT INCLUDE TABLESPACES DATABASE XXXXXXXX ALL
RECOVER LIST COMPONENT REUSE TOLOGPOINT X'LRSNXXXXXVALUE'
REBUILD INDEX LIST COMPONENT REUSE
//
/*
```

27.2.2 Restoring the workspace

Before restoring the workspace to the V5.0 Message Brokers Toolkit environment, ensure that the workspace resources have been stored or backed up correctly. For detailed information about how to back up the workspace resources, refer to 19.4, “Message Brokers Toolkit workspace backup” on page 347.

If the projects from the Message Brokers Toolkit were exported to a single zip file as described in 19.4, “Message Brokers Toolkit workspace backup” on page 347, the workspace backup will also be a zip file.

To restore the workspace:

1. Unzip the previously zipped file into a temporary directory. These could be directories containing projects that have to be imported again.
2. Launch the V5.0 Message Brokers Toolkit.
3. Repeat the following steps for every project that has to be restored:
   a. Launch the V5.0 Message Brokers Toolkit.
   b. Select **File** → **Import** in the menu.
   c. Select **Existing Project into Workspace** in the list displayed in the dialog box that appears, as illustrated in Figure 27-5, and click **Next**.

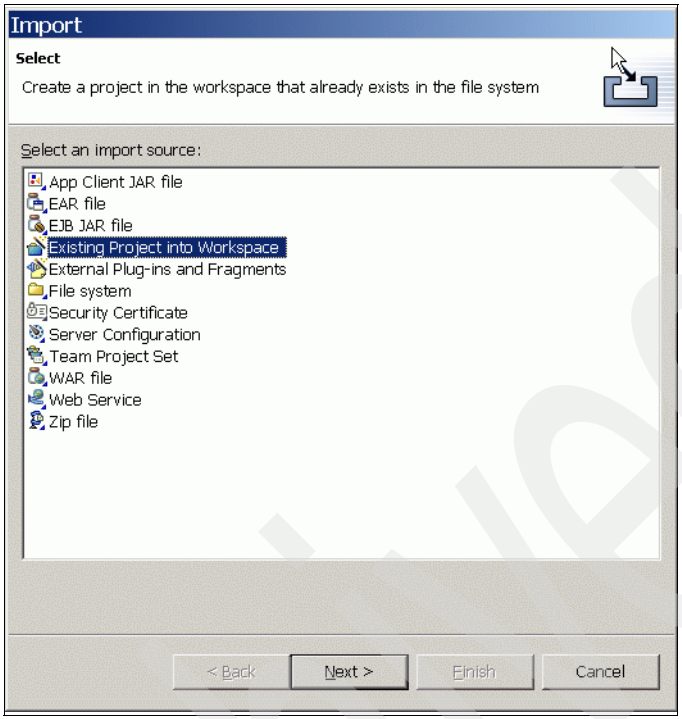


Figure 27-5 Selecting a source

d. In the dialog box that appears, click **Browse**, then nd go to the root directory of each project under the temporary directory and click **OK**.

e. Click **Finish** to complete the import, as shown in Figure 27-6.

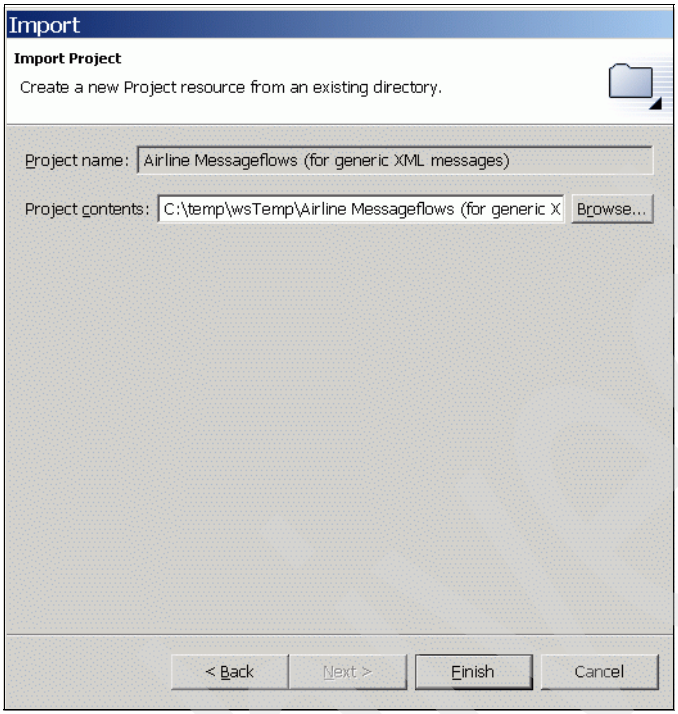


Figure 27-6 Browsing the source directory

4. Switch to the Broker Application Development perspective and review the resources.

27.2.3 Restoring the user profile and the ODBC connection file

If the user profile and the ODBC connection file have been changed for the WebSphere Message Broker V6.0 environment on UNIX and Linux, they also must be restored to their previous levels.

If the user profile has been modified to run the WebSphere Message Broker V6.0 profile, change it in such a way that it points to the previous version's path. The samples of the V5.0 profile is found in the following path:

broker_install_directory/sample/profiles

Change or run the user profile as follows:

broker_install_directory/sample/profiles/profile.aix

If the ODBCINI system environment variable has been set to point to the new ODBC driver from the WebSphere Message Broker V6.0, reset it to point to the previous version of the file path as displayed here.

```
export ODBCINI=/var/mqsi/odbc/.odbc.ini
```

27.2.4 Restoring the components using the mqsimigratecomponents command

The **mqsimigratecomponents** command can also be used to restore the components to their previous level, either fully or in part. The steps outlined in the following list detail the tasks to be performed with the **mqsimigratecomponents** command in order to restore the components.

1. Check one or more components without making any changes to ensure that the components are suitable for migration.
2. Move one or more components to a different version, either in full or in part.
3. Undo, that is, reverse a successful move from one version to another, either in full or in part.
4. Verify that the move is successful.

On z/OS, the BIPMGCMP job is provided. Run this job in the broker's component data set. Specify the correct parameters at the bottom of this job for the **mqsimigratecomponents** command.

Preparing to restore

Carry out the steps outlined in the following list to prepare the components for restoration:

1. Open the command line environment of WebSphere Message Broker V6.0.
2. Stop the brokers, the Configuration Manager, and the User Name Server.
3. Ensure that all the deployments to the broker or the broker domain are completed.
4. The source level **-s** and the target level **-t** must be presented with the check **-c** parameter. Verify that the components are suitable for restoration by entering the following command:

   ```
   mqsimigratecomponents -c -s 6.0.0.0 -t 5.0.0.6 component_name
   ```

   On the distributed platforms, the output of the command appears as shown in Example 27-5:

Example 27-5 The output of the premigration check of the mqsimigratecomponents command

```
$ mqsimigratecomponents -c -s 6.0.0.0 -t 5.0.0.6 BKL1
BIP8849I: Broker 'BKL1' (Version 6.0) with Queue Manager 'BKL1QM' and Data
Source 'BKL1DB' specified for migration.
BIP8680I: Pre-migration check succeeded.
BIP8071I: Successful command completion.
```

Restoring to the previous level fully

To restore to the previous level fully, the source level **-s** and the target level **-t** must be presented, and the following command must be entered in the command line:

```
mqsimigratecomponents -s 6.0.0.0 -t 5.0.0.6 component_name
```

On the distributed platforms, the output of the command appears as shown in Example 27-6:

Example 27-6 The output of the mqsimigratecomponents command

```
$mqsimigratecomponents -s 6.0.0.0 -t 5.0.0.6 BKL1
BIP8849I: Broker 'BKL1' (Version 6.0) with Queue Manager 'BKL1QM' and Data
Source 'BKL1DB' specified for migration.
BIP8670I: Database migration started
BIP8663I: Creating temporary new tables
BIP8664I: Migrating from existing tables to temporary new tables
BIP8665I: Dropping existing tables
BIP8666I: Creating new tables
BIP8667I: Copying all rows from temporary new tables to new tables
BIP8668I: Dropping temporary new tables
BIP8669I: Database migration successful
WebSphere MQ queue manager running.
BIP8785I: Starting WebSphere MQ queue migration for component 'BKL1'.
BIP8788I: Deleted WebSphere MQ queue 'SYSTEM.BROKER.TIMEOUT.QUEUE'
BIP8788I: Deleted WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REQUEST'
BIP8788I: Deleted WebSphere MQ queue 'SYSTEM.BROKER.AGGR.CONTROL'
BIP8788I: Deleted WebSphere MQ queue 'SYSTEM.BROKER.AGGR.REPLY'
BIP8788I: Deleted WebSphere MQ queue 'SYSTEM.BROKER.AGGR.TIMEOUT'
BIP8788I: Deleted WebSphere MQ queue 'SYSTEM.BROKER.AGGR.UNKNOWN'
BIP8789I: Finished WebSphere MQ queue migration for component 'BKL1'.
BIP8750I: Starting registry migration for component 'BKL1'.
BIP8755I: Copied value 'QueueManagerName' into the new location
BIP8755I: Copied value 'DataSourceName' into the new location
BIP8755I: Copied value 'ServiceUserId' into the new location
BIP8755I: Copied value 'ServicePassword' into the new location
BIP8755I: Copied value 'DataSourceUserId' into the new location
BIP8755I: Copied value 'DataSourcePassword' into the new location
BIP8755I: Copied value 'LilPath' into the new location
```

```
BIP8755I: Copied value 'ConfigurationTimeout' into the new location
BIP8755I: Copied value 'ConfigurationDelayTimeout' into the new location
BIP8755I: Copied value 'StatisticsMajorInterval' into the new location
BIP8755I: Copied value 'MigrationNeeded' into the new location
BIP8755I: Copied value 'MQTrustedQueueManager' into the new location
BIP8755I: Copied value 'UserNameServerQueueManagerName' into the new location
BIP8755I: Copied value 'PubSubAccessControl' into the new location
BIP8755I: Copied value 'AdminAgentPID' into the new location
BIP8755I: Copied value 'BrokerUUID' into the new location
BIP8767W: Found registry value 'FADLevel' that is not valid in the target
version; not copied.
BIP8762I: Copied registry value from 'traceLevel' to HTTP value
'HttpListenerTraceLevel'; set to 'none'
BIP8762I: Copied registry value from 'traceSize' to HTTP value
'HttpListenerTraceSize'; set to '4194304'
BIP8762I: Copied registry value from 'port' to HTTP value 'HttpListenerPort';
set to '7080'
BIP8763I: Deleted value 'port' from the old location
BIP8763I: Deleted value 'traceLevel' from the old location
BIP8763I: Deleted value 'traceSize' from the old location
BIP8763I: Deleted value 'QueueManagerName' from the old location
BIP8763I: Deleted value 'DataSourceName' from the old location
BIP8763I: Deleted value 'ServiceUserId' from the old location
BIP8763I: Deleted value 'ServicePassword' from the old location
BIP8763I: Deleted value 'DataSourceUserId' from the old location
BIP8763I: Deleted value 'DataSourcePassword' from the old location
BIP8763I: Deleted value 'LilPath' from the old location
BIP8763I: Deleted value 'ConfigurationTimeout' from the old location
BIP8763I: Deleted value 'ConfigurationDelayTimeout' from the old location
BIP8763I: Deleted value 'StatisticsMajorInterval' from the old location
BIP8763I: Deleted value 'MigrationNeeded' from the old location
BIP8763I: Deleted value 'MQTrustedQueueManager' from the old location
BIP8763I: Deleted value 'UserNameServerQueueManagerName' from the old location
BIP8763I: Deleted value 'PubSubAccessControl' from the old location
BIP8763I: Deleted value 'AdminAgentPID' from the old location
BIP8763I: Deleted value 'BrokerUUID' from the old location
BIP8763I: Deleted value 'FADLevel' from the old location
BIP8768I: Finished registry migration for component 'BKL1'.
BIP8654I: Moving filesystem artefacts from '/var/mqsi' to '/var/mqsi'
BIP8657I: Creating directory '/var/mqsi/brokers/BKL1'
BIP8657I: Creating directory '/var/mqsi/brokers/BKL1/bin'
BIP8653I: Creating a symbolic link with target '/opt/mqsi/template/mqsistart'
and link name '/var/mqsi/brokers/BKL1/bin/mqsistart'
BIP8653I: Creating a symbolic link with target '/opt/mqsi/template/mqsistop'
and link name '/var/mqsi/brokers/BKL1/bin/mqsistop'
BIP8071I: Successful command completion.
```

Restoring to the previous level partially

If the migration to the WebSphere Message Broker V6.0 environment was performed in stages utilizing the **-1**, **-2**, and **-3** parameters with the **mqsimigratecomponents** command, then the components can be restored to the previous version in the same manner.

The command should be specified with the **-u** parameter along with the source level **-s** and the target level **-t** options. The following list describes the meanings of these options:

-u Reverses a failed migration step. The **-u** parameter must also specify at least one of the **-1**, **-2**, or **-3** parameters. This option should be used only when migration has failed, as has auto recover.

-1 Restores only the registry and the file system to the previous version. The **-1** parameter should be used after the **-2** or **-3** parameters.

-2 Restores only the WebSphere MQ to the previous version.

-3 Restores only the database to the previous version.

Verifying the restoration

The restoration can be verified by using the **-v** parameter with the **mqsimigratecomponents** command. Only the target level **-t** must be specified.

Enter the following command in the command line:

```
mqsimigratecomponents -v -t 5.0.0.6 component_name
```

On the distributed platforms, the output of this command appears as shown in Example 27-7.

Example 27-7 The output of the mqsimigratecomponents command with the verification option

```
$ mqsimigratecomponents -v -t 5.0.0.6 BKL1
BIP8849I: Broker 'BKL1' (Version 5.0) with Queue Manager 'BKL1QM' and Data
Source 'BKL1DB' specified for migration.
BIP8689I: The source and target version are compatible, no migration is
necessary.
BIP8071I: Successful command completion.
```

27.3 Additional post-migration tasks

This section discusses additional post-migration tasks that are optional, including:

- Cleaning up the obsolete database or tables in the Configuration Manager database
- Removing the previous version of WebSphere Business Integration Message Broker V5.0

27.3.1 Dropping the unnecessary database or tables

In WebSphere Message Broker V6.0, the Configuration Manager no longer uses an external database to store the domain configuration. It uses an internal repository instead. This means that the Configuration Manager database can be dropped. To drop the database, enter the following command in the DB2 Universal Database command line:

```
db2 DROP DB database_name
```

If you do not want to drop the database, you can choose to drop only the unnecessary tables. To do this on distributed platforms, enter the following command in the DB2 Universal Database command line:

```
db2 CONNECT TO database_name
db2 DROP TABLE table_name
```

The following tables are not used by the Configuration Manager in the V6.0 environment:

- CACLGROUPS
- CBROKER
- CBROKERCEG
- CCOLLECTIVE
- CCOLLECTIVECBROKER
- CDELETE
- CEG
- CEGCMSGFLOW
- CEGCMSGPROJECT
- CLOG
- CMSGFLOW
- CMSGPROJECT
- CNEIGHBOURS
- COUTSTANDING
- CPROXY
- CPROXYCRESOURCE
- CSUBSCRIBE

- CTOPIC
- CTOPICCTOPIC
- CTOPOLOGY
- CTRACE
- CUUIDLOCKS

27.3.2 Uninstalling the WebSphere Business Integration Message Broker V5.0

The WebSphere Message Broker V6.0 allows coexistence with the previous versions of the WebSphere Message Broker products. Removing the WebSphere Business Integration Message Broker V5.0 and its prerequisite software products after the migration is, therefore, optional.

Before uninstalling any previous versions of products, ensure that the migration is successful and has been verified thoroughly. Refer to the related product documentation for details about uninstalling the products.

Part 4

Appendixes

This part of the book provides appendices that include the sample code used for migration verification in this redbook, discusses integrating the WebSphere MQ V6.0 Explorer perspective into Message Brokers Toolkit V6.0 and lists the softcopy samples available to download.

The following topics are covered:

- All message flows and message sets migrated from WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 products to WebSphere Message Broker V6.0 product
- WebSphere MQ V6.0 Explorer integration into Message Brokers Toolkit V6.0
- Web material associated with this redbook available in softcopy

A

Message flows used in the migration scenarios

This appendix provides information about all message flows and message sets migrated from WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0.

In the scenarios in this redbook, some message flows were deployed with different functionality, summarized in Table A-1 on page 526.

The sample code and other required material can be downloaded from the Web. See Appendix C, “Additional material” on page 555 for instructions.

Message flows for WebSphere MQ Integrator V2.1 scenario

The tables here show a summary of message flows and nodes used in the operating systems.

Table A-1 Message flows in the AIX environment

Flow name	Nodes used	Figure
Soccer	MQInput Publication	See “Soccer” on page 528
Loan_Request[a]	MQInput Database Compute Filter MQOutput LoanErrorHandlingSub	See “Loan_Request” on page 528
LoanErrorHandlingSub	Input ResetContentDescriptor Trace MQOutput	See “LoanErrorHandlingSub” on page 529
JOURNEY_Aggregation	MQInput AggregateControl Compute MQOutput AggregateRequest AggregateReply MQReply	See “JOURNEY_Aggregation” on page 529

a)The message flow Loan_Request requires the message sets: RequestLoan (CWF physical type) and ResultLoan (XML physical type).

Table A-2 Message flows in the Windows environment

Flow name	Nodes used	Figure
Soccer	MQInput Publication	See “Soccer” on page 528
Loan_Request[b]	MQInput Database Compute Filter MQOutput LoanErrorHandlingSub	See “Loan_Request” on page 528

Flow name	Nodes used	Figure
LoanErrorHandlingSub	Input ResetContentDescriptor Trace MQOutput	See “LoanErrorHandlingSub” on page 529
JOURNEY_Aggregation	MQInput AggregateControl Compute MQOutput AggregateRequest AggregateReply MQReply	See “JOURNEY_Aggregation” on page 529
NN_FORMAT_TEST	MQInput NeonFormatter MQOutput	See “NN_FORMAT_TEST” on page 530
NN_RULES_TEST	MQInput NeonRules MQOutput	See “NN_RULES_TEST” on page 530
NY_MAP_TEST	MQInput NEONMap MQOutput	See “NY_MAP_TEST” on page 531
NY_RULES_TEST	MQInput NEONRulesEvaluation MQOutput RouteToLabel Label	See “NY_RULES_TEST” on page 531

b)The message flow Loan_Request requires the message sets: RequestLoan (CWF physical type) and ResultLoan (XML physical type).

Table A-3 Message flows in the z/OS environment

Flow name	Nodes used	Figure
DB2U	MQInput Compute Database MQOutput	See “DB2U” on page 532
FANIN	MQInput Compute Filter AggregateReply MQOutput	See “FANIN” on page 532

Flow name	Nodes used	Figure
FANOUT_plus_original_msg	MQInput Compute AggregateControl MQOutput AggregateRequest	See “FANOUT_plus_original_msg” on page 532
REQUEST_to_REPLY	MQInput MQOutput	See “REQUEST_to_REPLY” on page 533
ONE2ONE	MQInput MQOutput	See “ONE2ONE” on page 533

Message flow figures

This section presents the figures of all message flows for the WebSphere MQ Integrator V2.1 scenario.

Soccer

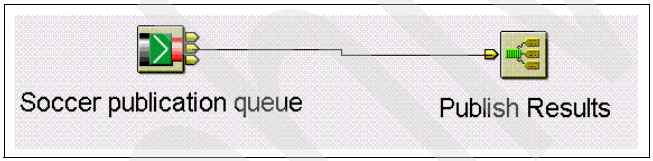


Figure A-1 Soccer message flow

Loan_Request

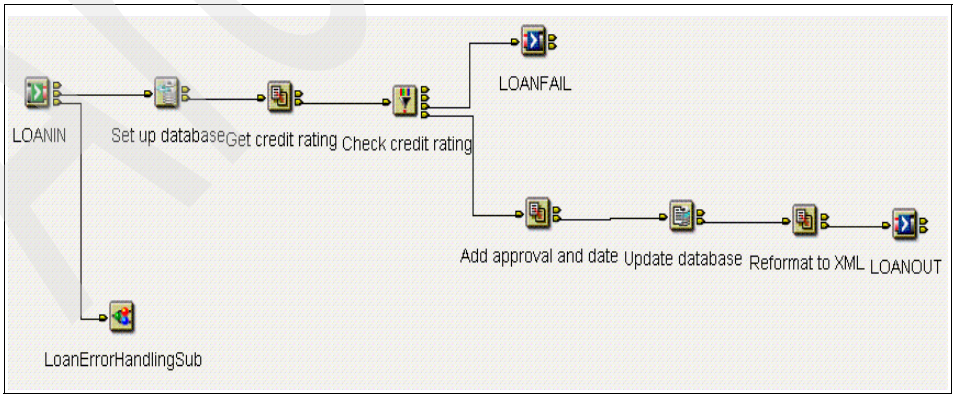


Figure A-2 Loan_Request message flow

LoanErrorHandlingSub

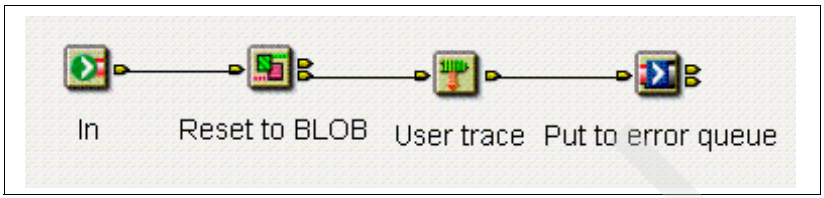


Figure A-3 LoanErrorHandlingSub subflow

JOURNEY_Aggregation

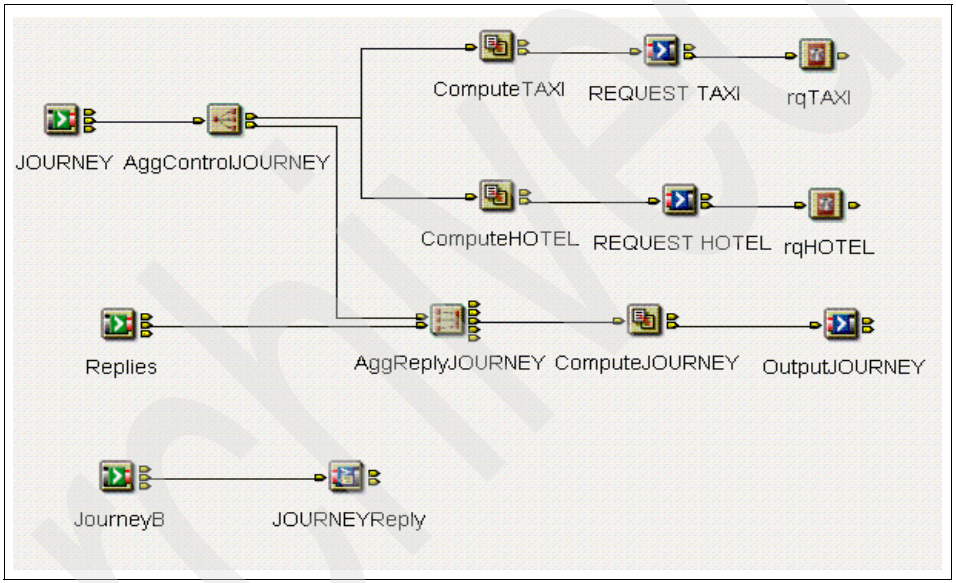


Figure A-4 JOURNEY_Aggregation message flow

NN_FORMAT_TEST

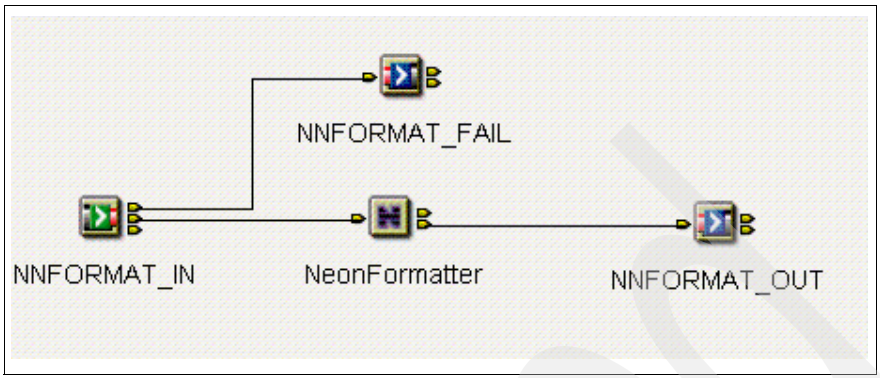


Figure A-5 NN_FORMAT_TEST message flow

NN_RULES_TEST

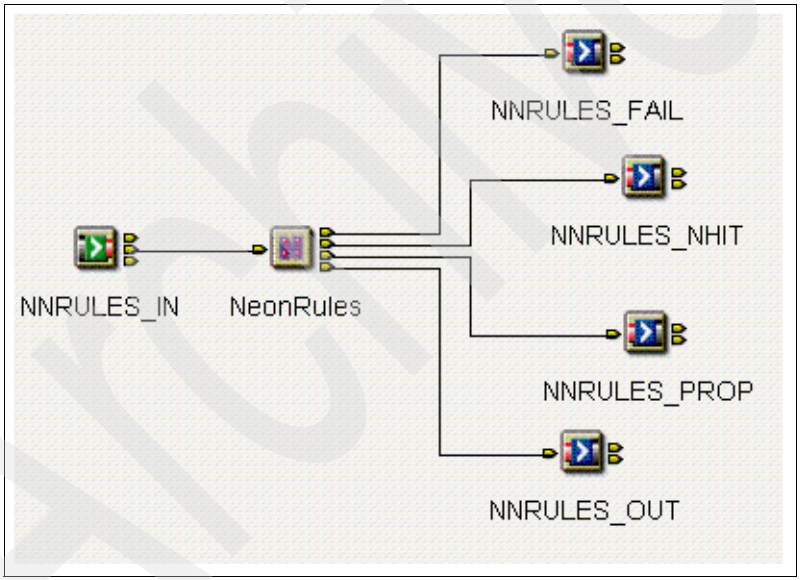


Figure A-6 NN_RULES_TEST message flow

NY_MAP_TEST

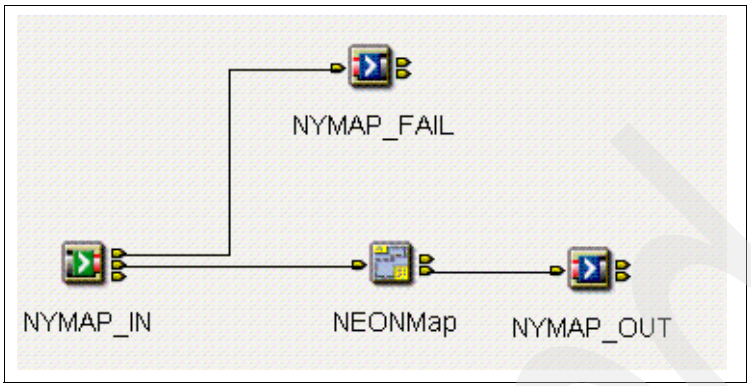


Figure A-7 NY_MAP_TEST message flow

NY_RULES_TEST

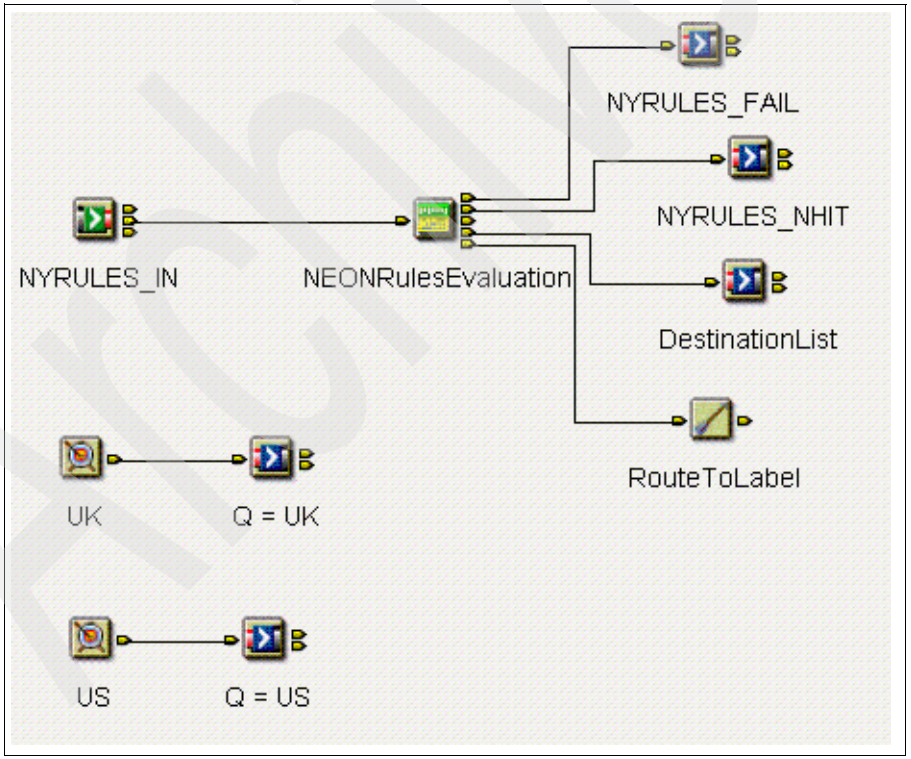


Figure A-8 NY_RULES_TEST message flow

DB2U

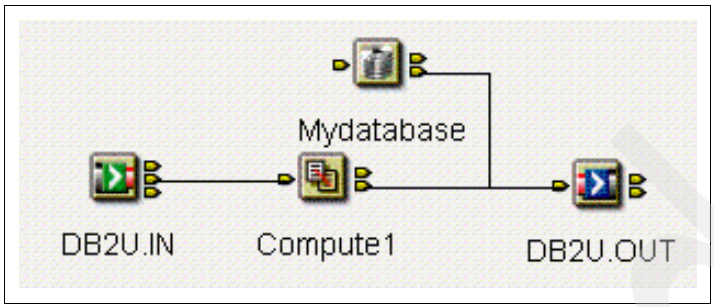


Figure A-9 DB2U message flow

FANIN

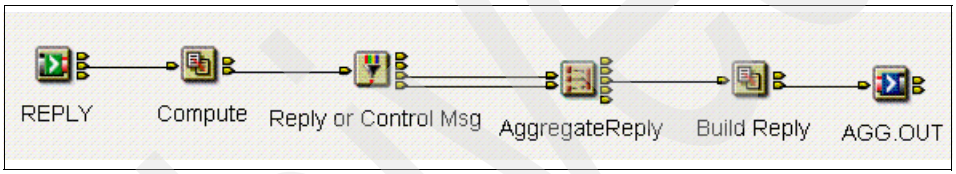


Figure A-10 FANIN message flow

FANOUT_plus_original_msg

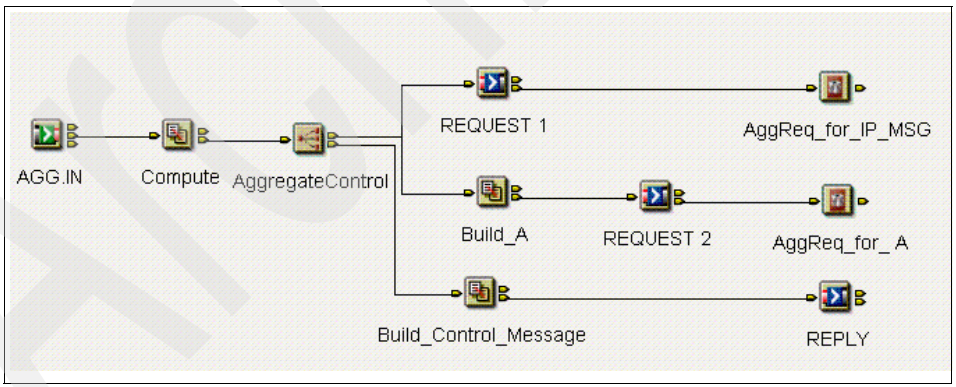


Figure A-11 FANOUT_plus_original_msg message flow

REQUEST_to_REPLY

Figure A-12 REQUEST_to_REPLY message flow

ONE2ONE

Figure A-13 ONE2ONE message flow

Message flows for WebSphere Business Integration Message Broker V5.0 scenario

The following tables present a summary of message flows and nodes used in the various operating systems.

Table A-4 Message flows in the AIX and Linux environments

Flow Name	Node used	Figure
WS_Server	HTTPInput Compute Trace HTTPReply	See “WS_Server” on page 538
WS_Client	MQInput HTTPRequest Compute MQOutput	See “WS_Client” on page 538
xmlt	MQInput XMLTransformation MQOutput	See “xmlt” on page 538

Flow Name	Node used	Figure
XML_FlightQueryIn	MQInput AggregateReply Compute MQOutput	See “XML_FlightQueryIn” on page 539
XML_FlightQueryOut	MQInput AggregateControl Compute MQOutput AggregateRequest	See “XML_FlightQueryOut” on page 539
XML_FlightQueryReply	MQInput Compute MQOutput MQReply	See “XML_FlightQueryReply” on page 540
XML_Reservation	MQInput Database MQOutput Trace Compute XML_BuildReplyMessage	See “XML_Reservation” on page 540
XML_BuildReplyMessage	Input Filter Compute Output Throw	See “XML_BuildReplyMessage” on page 541
ScribblePublish_Unix	MQInput Filter Compute Publication	See “ScribblePublish_Unix” on page 541
Video_Test[c]	MQInput MQOutput Mapping	See “Video_Test” on page 542

c)The Message flow Video_Test requires message set Video that uses the physical types CWF, XML and TDS.

Table A-5 ***Message flows on Windows environment***

Flow Name	Node used	Figure
WS_Server	HTTPInput Compute Trace HTTPReply	See “WS_Server” on page 538
WS_Client	MQInput HTTPRequest Compute MQOutput	See “WS_Client” on page 538
xmlt	MQInput XMLTransformation MQOutput	See “xmlt” on page 538
XML_FlightQueryIn	MQInput AggregateReply Compute MQOutput	See “XML_FlightQueryIn” on page 539
XML_FlightQueryOut	MQInput AggregateControl Compute MQOutput AggregateRequest	See “XML_FlightQueryOut” on page 539
XML_FlightQueryReply	MQInput Compute MQOutput MQReply	See “XML_FlightQueryReply” on page 540
XML_Reservation	MQInput Database MQOutput Trace Compute XML_BuildReplyMessage	See “XML_Reservation” on page 540
XML_BuildReplyMessage	Input Filter Compute Output Throw	See “XML_BuildReplyMessage” on page 541

Flow Name	Node used	Figure
ScribblePublish_Win	JMSInput MQInput Filter Compute Publication	See “ScribblePublish_Win” on page 542
Video_Test[d]	MQInput MQOutput Mapping	See “Video_Test” on page 542
NN_FORMAT_TEST	MQInput NeonFormatter MQOutput	See “NN_FORMAT_TEST” on page 543
NN_RULES_TEST	MQInput NeonRules MQOutput	See “NN_RULES_TEST” on page 543
NY_MAP_TEST	MQInput NEONMap MQOutput	See “NY_MAP_TEST” on page 544
NY_RULES_TEST	MQInput NEONRulesEvaluation MQOutput RouteToLabel Label	See “NY_RULES_TEST” on page 544

d)The Message flow Video_Test requires message set Video that uses the physical types CWF, XML and TDS.

Table A-6 Message flows on z/OS environment

Flow name	Nodes used	Figure
DB2U	MQInput Compute MQReply	See “DB2U” on page 545
FANIN	MQInput Compute Filter AggregateReply MQOutput	See “FANIN” on page 545

Flow name	Nodes used	Figure
FANOUT_plus_original_msg	MQInput Compute AggregateControl MQOutput AggregateRequest	See “FANOUT_plus_original_msg” on page 545
HTTP	MQInput Compute HTTPRequest MQReply Trace	See “HTTP” on page 546
ONE2ONE	MQInput MQReply	See “ONE2ONE” on page 546
PUBSUB	MQInput Trace Compute Publication	See “PUBSUB” on page 546
REQUEST_to_REPLY	MQInput MQReply	See “REQUEST_to_REPLY” on page 546
TOPUBSUB	MQInput Compute Trace MQOutput	See “TOPUBSUB” on page 547
Web_service	HTTPInput HTTPReply	See “Web_service” on page 547
XSLT	MQInput XMLTransformation Trace MQOutput MQReply	See “XSLT” on page 547
WS_Server	HTTPInput Compute Trace HTTPReply	See “WS_Server” on page 538
WS_Client	MQInput HTTPRequest Compute MQOutput	See “WS_Client” on page 538

Message flow figures

This section presents the figures of all message flows for WebSphere Business Integration Message Broker V5.0.

WS_Server

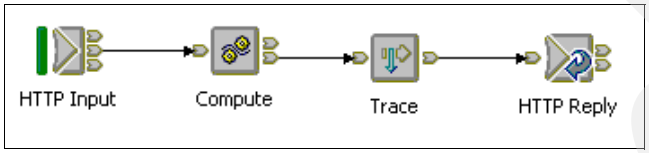


Figure A-14 WS_Server message flow

WS_Client

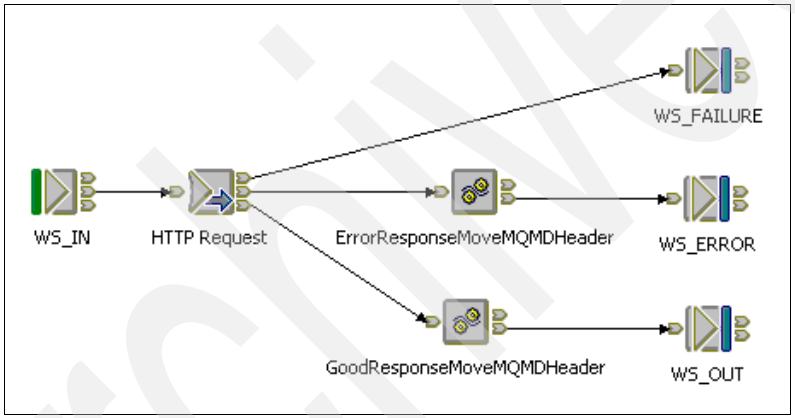


Figure A-15 WS_Client message flow

xmlt

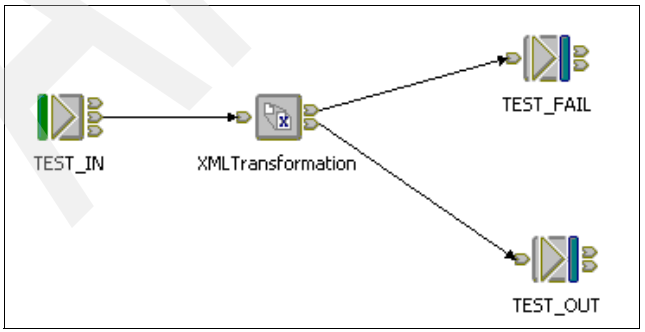


Figure A-16 xmlt message flow

XML_FlightQueryIn

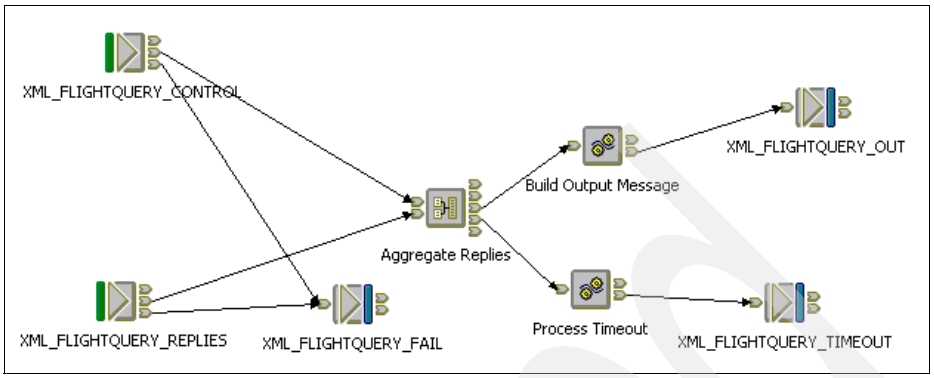


Figure A-17 XML_FlightQueryIn message flow

XML_FlightQueryOut

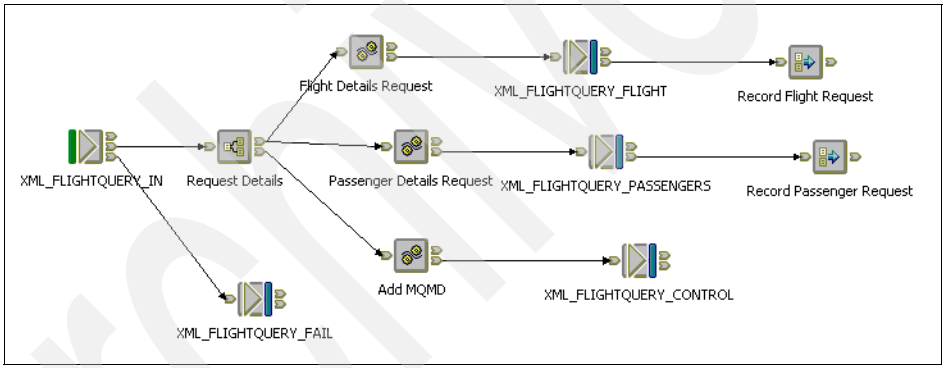


Figure A-18 XML_FlightQueryOut message flow

XML_FlightQueryReply

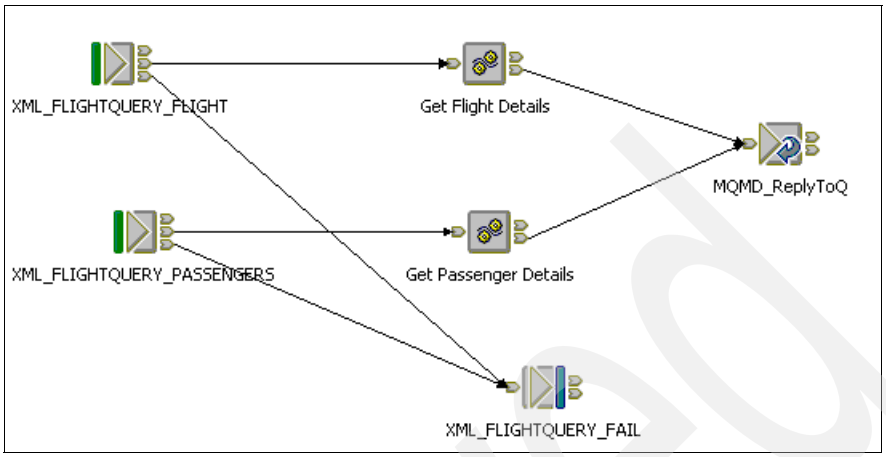


Figure A-19 XML_FlightQueryReply subflow

XML_Reservation

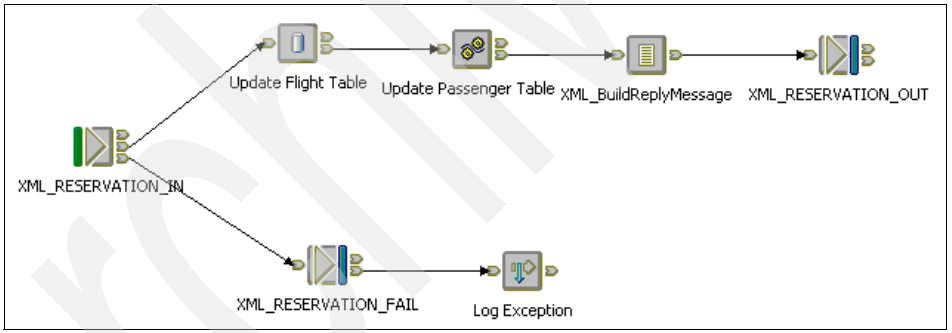


Figure A-20 XML_Reservation message flow

XML_BuildReplyMessage

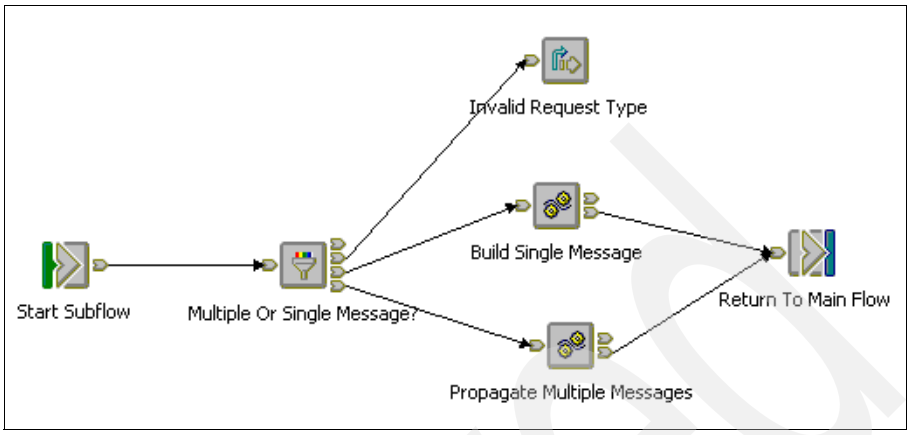


Figure A-21 XML_BuildReplyMessage message flow

ScribblePublish_Unix

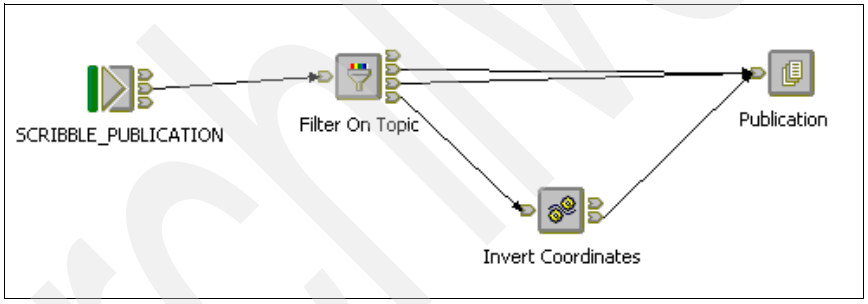


Figure A-22 ScribblePublish_Unix message flow

Video_Test

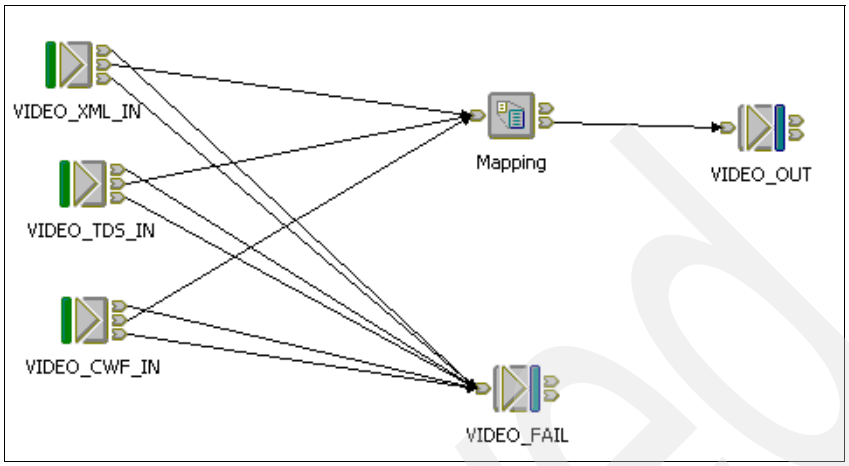


Figure A-23 Video_Test message flow

ScribblePublish_Win

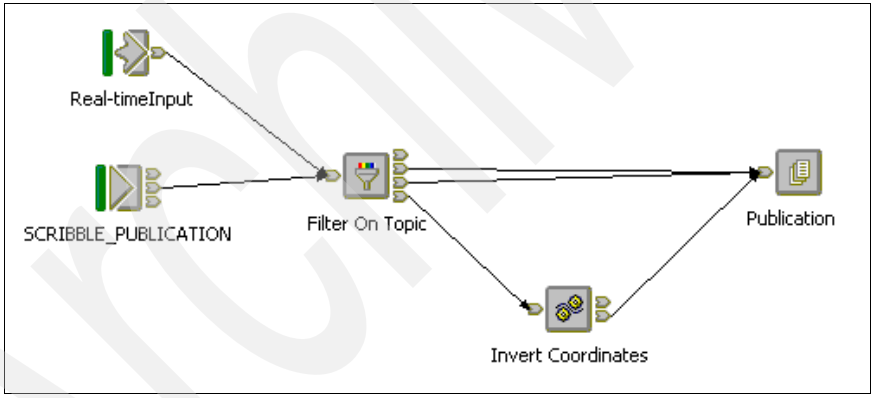


Figure A-24 ScribblePublish_Win message flow

NN_FORMAT_TEST

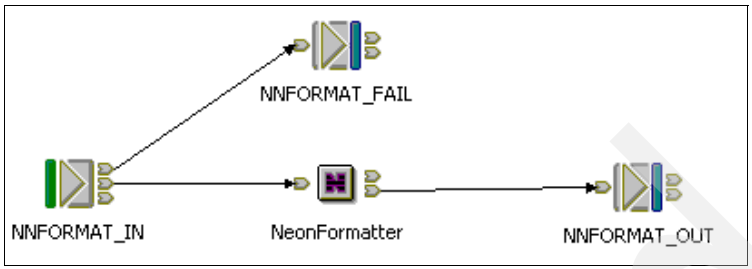


Figure A-25 NN_FORMAT_TEST message flow

NN_RULES_TEST

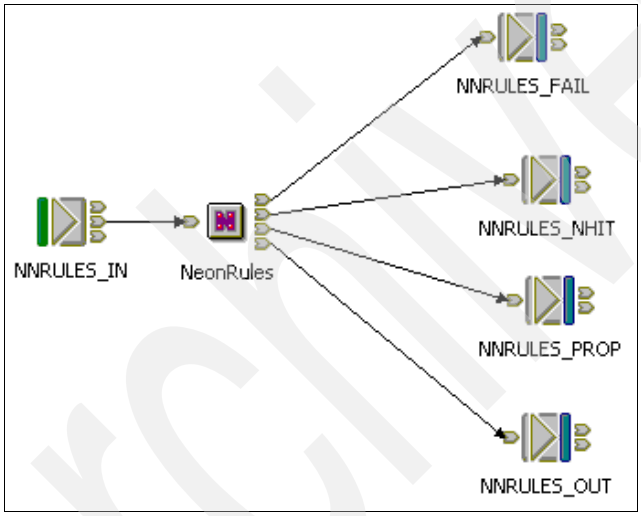


Figure A-26 NN_RULES_TEST message flow

NY_MAP_TEST

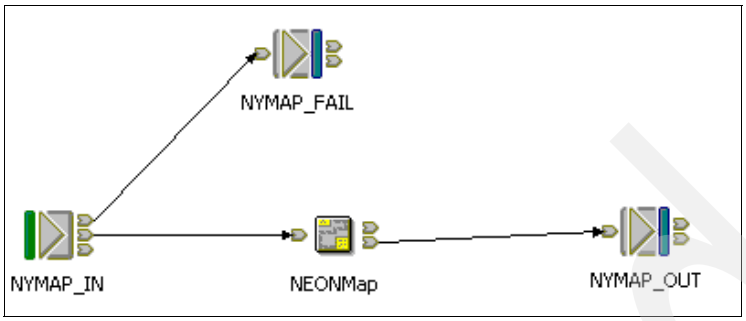


Figure A-27 NY_MAP_TEST message flow

NY_RULES_TEST

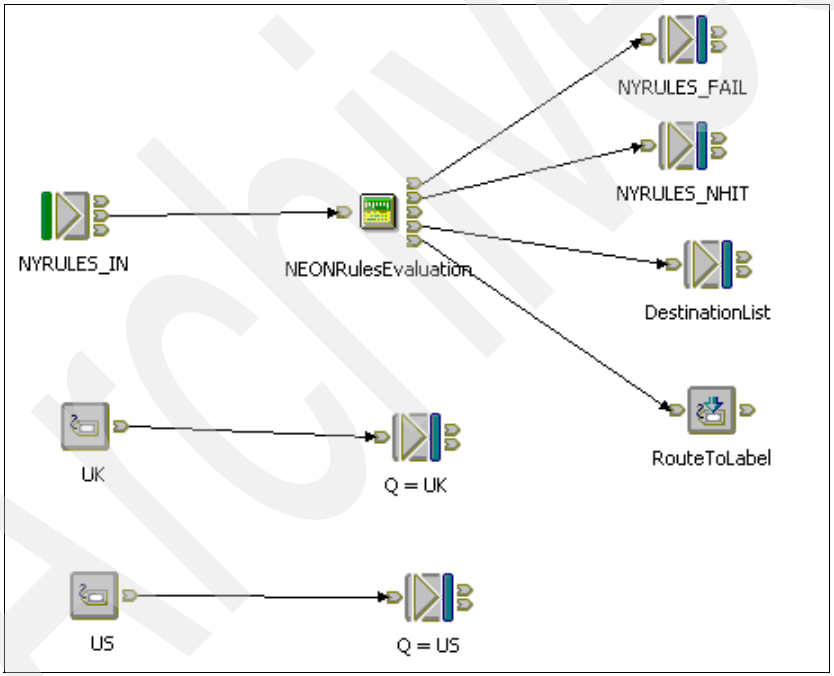


Figure A-28 NY_RULES_TEST message flow

DB2U

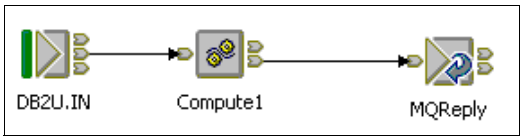


Figure A-29 DB2U message flow

FANIN

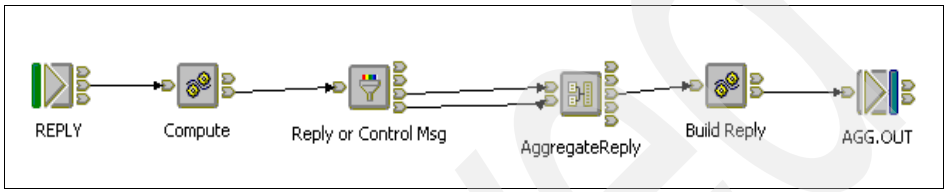


Figure A-30 FANIN message flow

FANOUT_plus_original_msg

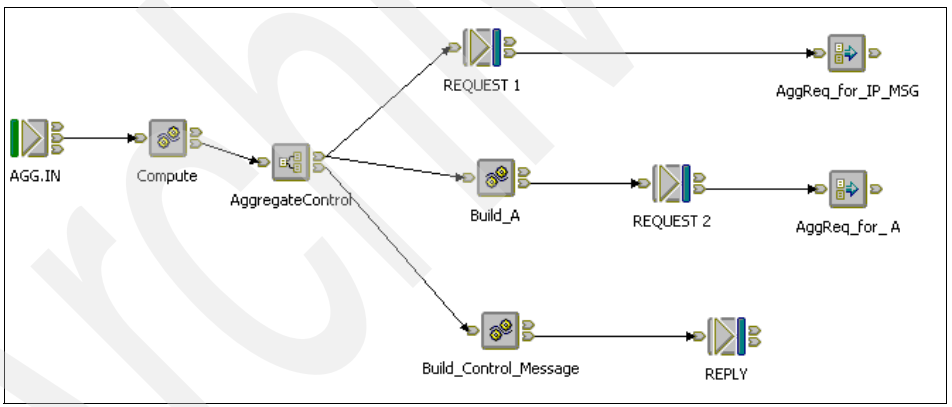


Figure A-31 FANOUT_plus_original_msg message flow

HTTP

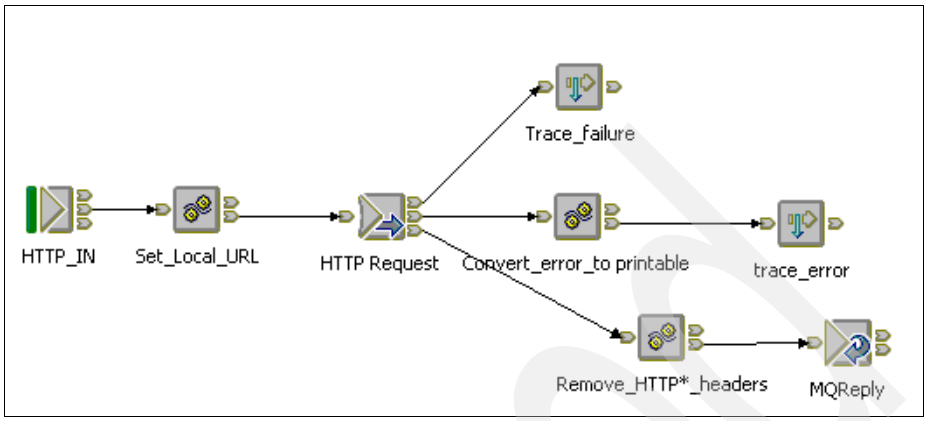


Figure A-32 HTTP message flow

ONE2ONE

Figure A-33 ONE2ONE message flow

PUBSUB

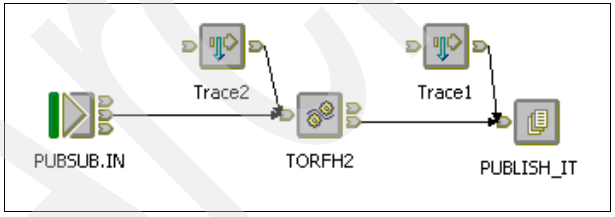


Figure A-34 PUBSUB message flow

REQUEST_to_REPLY

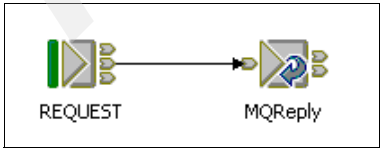


Figure A-35 REQUEST_to_REPLY message flow

TOPUBSUB

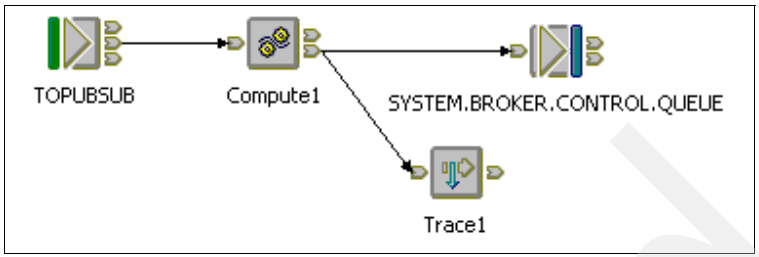


Figure A-36 TOPUBSUB message flow

Web_service

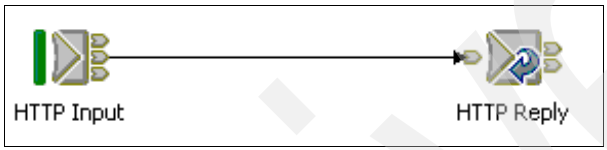


Figure A-37 Web_service message flow

XSLT

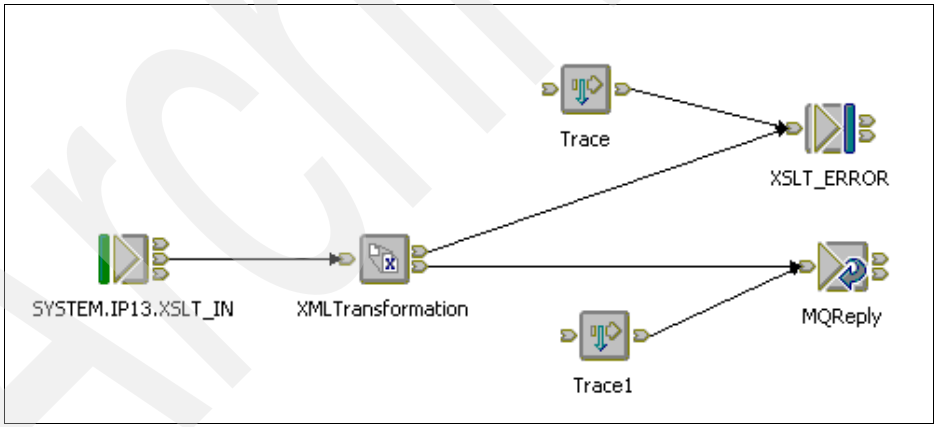


Figure A-38 XSLT message flow

B

Integrating WebSphere MQ Version 6.0 Explorer into Message Brokers Toolkit Version 6.0

This appendix discusses integrating the WebSphere MQ V6.0 Explorer perspective into Message Brokers Toolkit V6.0 and provides an example of the integration steps on Windows.

Integrating the WebSphere MQ Explorer perspective into the Message Brokers Toolkit

WebSphere MQ V6.0 Explorer is available as an Eclipse perspective and can be integrated into Message Brokers Toolkit Version 6.0.

The WebSphere MQ Explorer can be integrated from the regular WebSphere MQ installation, or the WebSphere MQ installation media can be used instead (the regular WebSphere MQ installation is not required in this case).

The Eclipse Update Manager is used to install WebSphere MQ V6.0 Explorer plugins into Message Brokers Toolkit V6.0. The minimum level for this installation is WebSphere MQ V6.0 with Refresh Pack 6.0.1.0.

In the Message Brokers Toolkit, do the following:

1. Select **Help** → **Software Updates** → **Find and Install**.
2. Select the **Search for new features to install** option and click **Next**.
3. Select **New Local Site** and add *<WebSphere MQ installation or sources directory>*`/eclipse` and click **OK**.
4. Select the entry added in the step above, in “Sites to include in search pane” and click **Next**.
5. Figure B-1 on page 551 illustrates the available WebSphere MQ V6.0 Explorer features. In the “Select the features to install pane”, select the **IBM WebSphere MQ Explorer, Version 6.0.1.0** entry and click **Next**.

   Optionally, the WebSphere MQ Explorer help entry can also be selected.

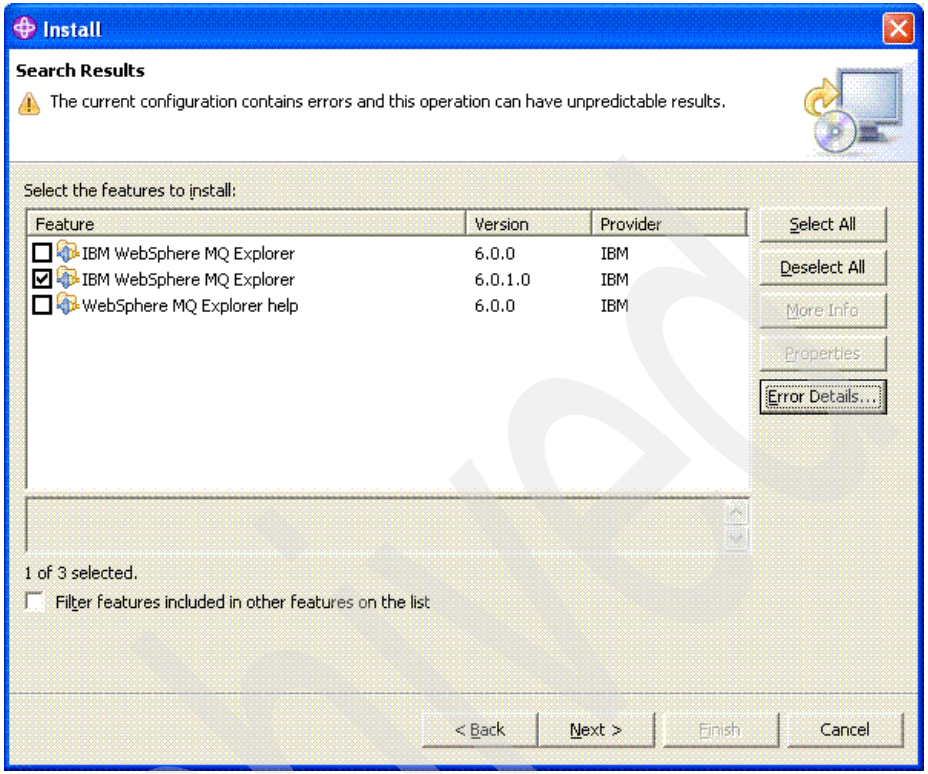


Figure B-1 WebSphere MQ V6.0.1.0 Explorer feature

6. Read the WebSphere MQ V6.0 license agreement and select the **I accept the terms in the license agreements** option, then click **Next**.
7. As illustrated in Figure B-2, select the **<WebSphere Message Broker V6.0 installation directory>/eclipse** entry in the “Available sites” pane and click **Finish**.

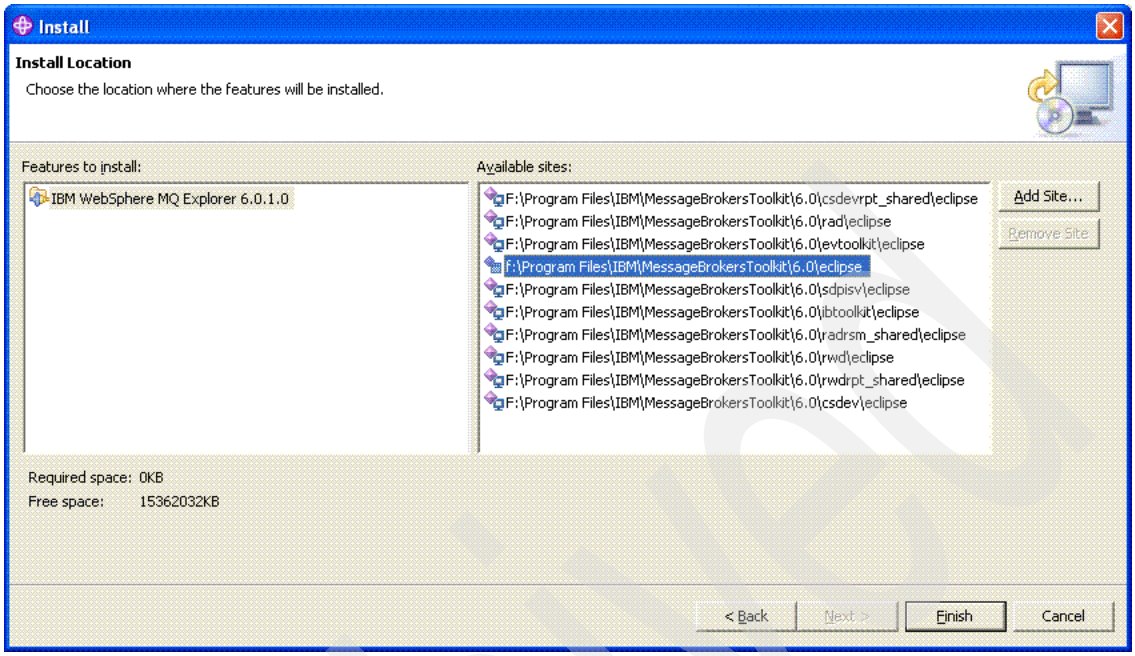


Figure B-2 Install locations for the WebSphere MQ V6.0.1.0 feature

8. The Message Brokers Toolkit prompts for the workbench to be restarted in order for the changes to take effect; select **Yes**.
9. When the Message Brokers Toolkit has restarted, select **Window** → **Open Perspective** → **Other**. In the dialog that appears, select **WebSphere MQ Explorer**, as illustrated in Figure B-3, then click **OK**.

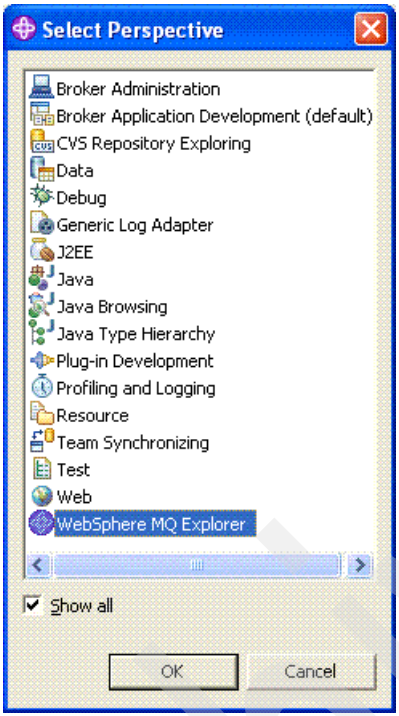


Figure B-3 Opening the WebSphere MQ V6.0 Explorer perspective

The WebSphere MQ Explorer perspective, as shown in Figure B-4, opens in the Message Brokers Toolkit.

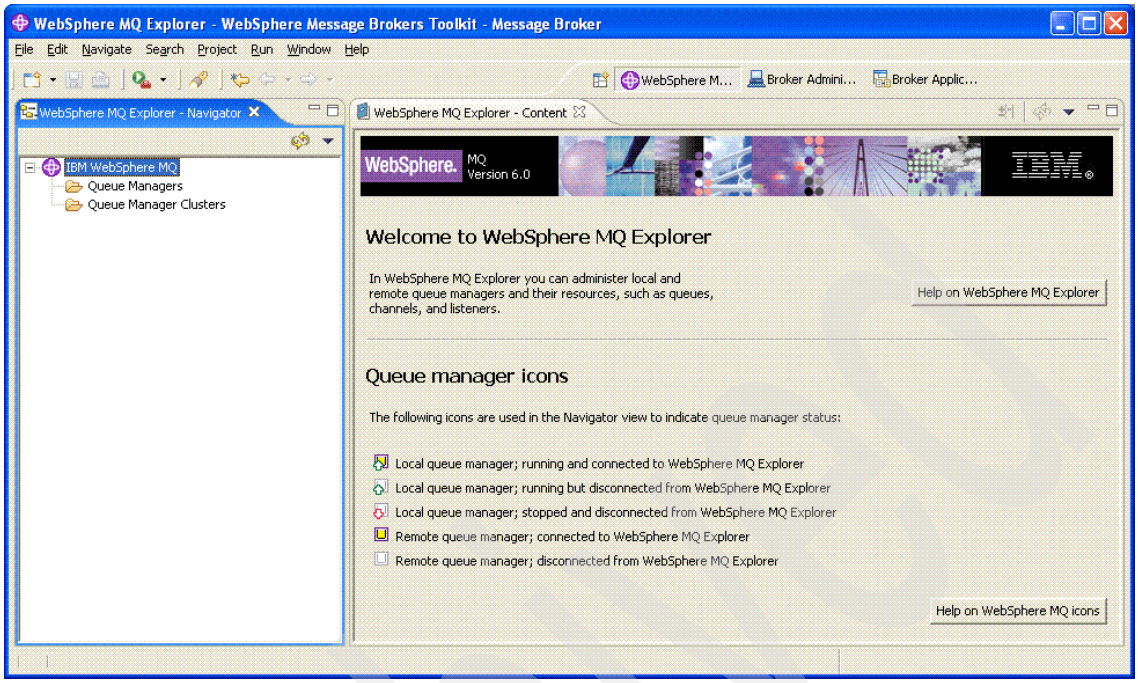


Figure B-4 WebSphere MQ perspective in the Message Brokers Toolkit

Attention: WebSphere MQ at Refresh Pack 6.0.1.0 is the minimum required level for installation via Eclipse Update Manager. This resolves the following WebSphere MQ defects that previously affected the integration of the WebSphere MQ perspective into the Message Brokers Toolkit:

- 96325 - Refer to:

  `http://www.ibm.com/software/integration/mqfamily/support/readme/all60_read.html#mqupdatemanager`

- IY76118 - Refer to:

  `http://www.ibm.com/support/docview.wss?rs=171&context=SSFKSJ&dc=DB550&uid=swg1IY76118&loc=en_US&cs=utf-8&lang=en`

C

Additional material

This redbook refers to additional material that can be downloaded from the Internet as described below.

Locating the Web material

The Web material associated with this redbook is available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser to:

ftp://www.redbooks.ibm.com/redbooks/SG247198

Alternatively, you can go to the IBM Redbooks Web site at:

ibm.com/redbooks

Select the **Additional materials** and open the directory that corresponds with the redbook form number, SG247198.

Using the Web material

The additional Web material that accompanies this redbook includes the following SG247198.*zip* file.

The downloadable *SG247198.zip* file contains three different folders, with the sub-folders listed below:

1. WebSphere MQ Integrator V2.1 message flows named *WMQI2.1.zip*.
2. WebSphere Business Integration Message Broker V5.0 message flows named *WBIMB5.0.zip*.
3. DB2 Universal Database for z/OS JCL examples named *DB2JCL.zip*.

SG247198.zip

This file contains samples used for Part 2 (WebSphere MQ Integrator V2.1) and Part 3 (WebSphere Business Integration Message Broker V5.0) validation scenarios in this redbook:

File name	*Description*
WMQI2.1.zip	Zipped code samples for Part 2 (WebSphere MQ Integrator V2.1).
WBIMB5.0.zip	Zipped code samples for Part 3 (WebSphere Business Integration Message Broker V5.0).
DB2JCL.zip	Zipped code samples for chapters 9,16,19 and 27 (DB2 Universal Database for z/OS backup and restore jobs).

WMQI2.1.zip

This file contains all samples used in WebSphere MQ Integrator V2.1 with sub-folders:

File name	*Description*
ZOSv2.zip	Zipped Code Samples for all message flows used on z/OS environment.
TRAVELv2.zip	Zipped Code Samples for JOURNEY_Aggregation message flow.
SOCCERv2.zip	Zipped Code Samples for Soccer message flow.
NNSYv2.zip	Zipped Code Samples for all New Era Of Networks message flows.
LOANv2.zip	Zipped Code Samples for Loan_Request message flow.

WBIMB5.0.zip

This file contains all samples used in WebSphere Business Integration Message Broker V5.0 with sub-folders:

File name	*Description*
ZOSv5.zip	Zipped Code Samples for all message flows used on z/OS environment.
AIRLINEv5.zip	Zipped Code Samples for XML_Flight message flows and XML_Reservation message flow.
SCRIBBLEv5.zip	Zipped Code Samples for Scribble message flows.

NNSYv5.zip	Zipped Code Samples for all New Era Of Networks message flows.
VIDEOv5.zip	Zipped Code Samples for Video_Test message flow.
WEBSERVICESv5.zip	Zipped Code Samples for WS_Server and WS_Client message flows.
XSLTv5.zip	Zipped Code Samples for XSLT message flow.

DB2JCL.zip

This file contains the sample jobs used in WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 to backup and restore DB2 Universal Database for z/OS.

How to use the Web material

Create a subdirectory (folder) on your workstation, and unzip the contents of the Web material zip file into this folder.

Glossary

AggregateControl node. A node in a message flow used to mark the beginning of a fan-out of requests that are part of an aggregation. It sends a control message that is used by the AggregateReply node to match the different requests that have been made.

AggregateReply node. A node in a message flow used to mark the end of an aggregation fan-in. This node collects replies and combines them into a single aggregated message.

AggregateRequest node. A node in a message flow used to record the fact that request messages for an aggregation have been sent. This node also collects information that helps the AggregateReply node to construct the aggregated message.

Application Log. A log viewable in the Windows Event Viewer that displays event messages from software on a machine.

Bar file. Abbreviation for message broker archive file, used to store compiled message flows, message sets and other code to be deployed to an execution group.

BIP message. An event message produced by WebSphere Message Broker, identifiable by its BIP number, for example BIP1003.

Breakpoint. Used as a point to stop the flow of a message in a message flow when the flow debugger is attached.

Broker. A broker is a set of execution processes that host and run message flows.

Broker Administration perspective. The perspective in the Message Brokers Toolkit that is used for administering and monitoring objects in the broker domain. This perspective is also used for changing the configuration and message flow deploy operations.

Broker Application Development perspective. The perspective in the Message Brokers Toolkit in which message flows and message sets can be developed.

Broker database. A database that stores configuration for a broker. Multiple brokers can share the same database.

Broker Domain. A group of brokers that share a common configuration and managed by a single Configuration Manager.

Broker topology editor. An editor in the Message Brokers Toolkit for configuring the properties of brokers in the domain.

Command console. This is a command line interface that sets up a suitable environment for running WebSphere Message Broker commands.

Compute node. A node in a message flow for processing messages using ESQL. Usually used for message transformation.

Configuration Manager. Stores the configuration data for the broker domain that it manages, and performs the deployment operations between the Message Broker Toolkit and the brokers in the domain.

Configuration Manager Proxy API. TA programming interface for performing administration operations on WebSphere Message Broker components.

Database node. A node in a message flow used to interact with a database in the specified ODBC data source, for performing database operations using ESQL.

DataDelete node. A node in a message flow that uses message mappings to delete data in a database based on the contents of an input message.

DataInsert node. A node in a message flow that uses message mappings to insert data in a database from the contents of an input message.

DataUpdate node. A node in a message flow that uses message mappings to update data in a database from the contents of an input message.

DB2 Enterprise Server. A database which is supported for use as a broker database and is supplied with WebSphere Message Broker.

Dead letter queue. A WebSphere MQ queue that holds messages that were put back onto an input queue by a message flow.

Debug perspective. The perspective in the Message Brokers Toolkit used for debugging message flows and the Java, ESQL, or mapping code associated with them.

Default Configuration Wizard. Creates a simple broker domain for verifying a WebSphere Message Broker installation. This configuration can be used for test purposes and for using the samples.

Domain connection. A reference to a broker domain in the Message Brokers Toolkit.

Enqueue file. A file in the Message Brokers Toolkit used to put simple messages on to a WebSphere MQ queues.

ESQL. ESQL is Extended Structured Query Language and is used in the transformation of messages in message flows. It is also used to perform database operations such as querying or updating a database.

ESQL editor. An editor within the Message Brokers Toolkit for creating and editing ESQL.

Event Log. An editor in the Message Brokers Toolkit showing event messages generated as a result of deployment operations and changes to the broker domain configuration.

Event messages. Messages produced by software on a machine indicating a specific event or error.

Execution groups. An execution group represents a collection of message flows within a broker.

Filter node. A node in a message flow used to route a message according to message content.

HTTPInput node. A node in a message flow that is used to receive a Web Services request. Must be used in conjunction with an HTTPReply node.

HTTPReply node. A node in a message flow that returns a response from a message flow to a Web service client.

HTTPRequest node. A node in a message flow that constructs a Web Service request using the contents of an input message. The response is parsed for inclusion in the output tree.

Input node. A node in a message flow that provides an in terminal for a subflow.

Java. An object-oriented programming language used for programming the JavaCompute node or user-defined nodes in the Message Brokers Toolkit.

Java editor. An editor in the Message Brokers Toolkit for developing Java code. Used in association with a JavaCompute node.

JavaCompute node. A node in a message flow for transforming and routing messages using Java. It is also used for performing database operations using Java.

JMSInput node. A node in a message flow that receives messages from JMS destinations accessed through a connection to a JMS provider.

JMSMQTransform node. A node in a message flow that transforms a message with a JMS message tree into a message compatible with the a WebSphere MQ JMS provider.

JMSOutput node. A node in a message flow that sends messages to JMS destinations using the Java Message Service Specification.

Label node. A node in a message flow used in combination with a RouteToLabel node to dynamically determine the route that a message takes through the message flow, based on its content..

Mapping. A method of message transformation using drag and drop from references to message definitions and database defintions.

Mapping node. A node in a message flow that uses message mappings to construct an output message using other messages or information from database tables.

Message Brokers Toolkit. A graphical user interface for performing the development and debugging of message flow applications. It is also used for administering WebSphere Message Broker components and deploying message flow applications.

Message Definition editor. An editor in the Message Brokers Toolkit for defining the logical and physical structure of messages.

Message Domain. The Message Domain is a property that can be set on an input node to indicate the type of message that the flow expects to process, and selects the appropriate parser for the flow to use. Examples are XML and MRM.

Message flow debugger. A tool for tracing the path of messages through a message flow and viewing the changing content of the message as it is processed by the flow.

Message flow editor. An editor within the Message Brokers Toolkit for creating message flows by adding and connecting nodes on a canvas.

Message flows. Message flows provide the logic used by the broker to process messages. Message flows are built from nodes programmed with basic logic.

Message mapping editor. An editor in the Message Brokers Toolkit for defining mapping relationships between a source and target message or database.

Message Set editor. An editor in the Message Brokers Toolkit for setting the logical and physical properties of a message set.

Message sets. Message sets contain definitions of messages to be processed by the broker. These message definitions contain information about the logical and physical structure of the messages.

MQGet node. A node in a message flow that can be used anywhere in a flow to retrieve a message from a WebSphere MQ queue.

MQInput node. A node in a message flow used to receive messages from clients that connect to the broker using the WebSphere MQ Enterprise Transport, and that use the MQI and AMI application programming interfaces. The MQInput node receives message input to a message flow from a WebSphere MQ message queue.

MQJMSTranform node. A node in a message flow that receives messages that have a WebSphere MQ JMS provider message tree format, and transform them into messages for JMS destinations.

MQOptimised node. A node in a message flow that is used to publish a persistent JMS message to a single subscriber to improve performance over ordinary publish/subscribe message flows.

MQOutput node. A node in a message flow used to send messages to clients that connect to the broker using the WebSphere MQ Enterprise Transport and that use the MQI and AMI application programming interfaces. The MQOutput node delivers an output message from a message flow to a WebSphere MQ queue.

MQReply node. A node in a message flow used to send a response to the originator of the input message. The MQReply node is a specialized form of the MQOutput node that puts the output message to the WebSphere MQ queue identified by the ReplyToQ field of the input message header.

mqsideploy. A command run in the command console for deploying message broker archive files.

mqsilist. A command run in the command console to list all the WebSphere Message Broker components on the machine.

mqsistart. A command run in the command console to start a component such as a broker or configuration manager.

mqsistop. A command run in the command console to stop a component such as a broker or configuration manager.

NEONFormatter node. A Rules and Formatter Extension node in a message flow that transforms a message from a known input format to a specified output format.

NEONMap node. A Rules and Formatter Extension node in a message flow that has the same functionality as the NEONTransform node, except output operations associated with the target format are not applied to the output message.

NEONRules node. A Rules and Formatter Extension node in a message flow that provides an encapsulation of the New Era of Networks Rules engine.

NEONRulesEvaluation node. A Rules and Formatter Extension node in a message flow that implements the processing and execution of rules defined using New Era of Networks Rules. These rules transform and redirect incoming messages based on the format, application group, and message content.

NEONTransform node. A Rules and Formatter Extension node in a message flow that transforms a message from a known input format to a specified output format.

ODBC drivers for Cloudscape. Open Database Connectivity drivers for the embedded Derby database used when the Derby database is used as a broker database.

Output node. A node in a message flow that provides an out terminal for a subflow.

Passthrough node. A node in a message flow that is used to provide a label to indicate the subflow it is contained in. Message are not processed; they only pass through this node.

Promoted Property. A message flow node property that has been promoted to the level of the message flow in which it is included.

Publication node. A node in a message flow used to support publish/subscribe applications by filtering output messages from a message flow and transmitting them to subscribers who have registered an interest in a particular set of topics.

Publish/subscribe. An alternative style of messaging using topics. Messages published on a topic are sent to all applications which subscribe to that topic.

Queue manager. A system program that provides queuing services to applications. It is used to enable communication between the WebSphere Message Broker components. Each component requires access to a queue manager.

Rational Agent Controller. Used for message flow debugging in the Message Brokers Toolkit. It must be installed on the same machine as the broker being debugged.

Real-timeInput node. A node in a message flow that receives messages from clients using WebSphere MQ Real-time Transport or WebSphere MQ Multicast Transport and JMS applications.

Real-timeOptimizedFlow node. A node in a message flow for high performance publish/subscribe using WebSphere MQ Real-time Transport or WebSphere MQ Multicast Transport and JMS.

ResetContentDescriptor node. A node in a message flow that enables the parser to be changed for a message mid flow.

RouteToLabel node. A node in a message flow used in combination with one or more Label nodes to dynamically determine the route that a message takes through the message flow, based on its content. The RouteToLabel node interrogates the LocalEnvironment of the message to determine the identifier of the Label node to which to route the message.

Rules and Formatter Extension. An extension from New Era of Networks providing Rules and Formatter nodes and the associated runtime elements to maintain functionality supplied by earlier releases.

Runtime version information. Information added to message flows and message sets to provide information about the version. This information is visible in deployed resources through the Message Brokers Toolkit.

Sample Deploy Wizard. A wizard for use with the WebSphere Message Broker samples which can import the sample files, create WebSphere MQ and database resources and deploy the sample to a default configuration.

Subflow. An embedded message flow, represented by a node, within another message flow. A subflow can be used in more than one message flow, hence allows common function such as error handling to be maintainable and reusable.

Subscriptions. A subscription is a registration of an applications interest in a particular topic in publish/subscribe.

System Log. A log viewable in the Windows Event Viewer that displays information about software running as Windows services including WebSphere Message Broker components.

Terminal. Each node in a message flow has a number of terminals. Messages are output to different terminals on a node, depending upon the results of processing in the node.

Throw node. A node in a message flow used to throw an exception within a message flow.

TimeoutControl node. A node in a message flow that used in conjunction with a TimeoutNotification node. This node receives an input message that contains a time out request.

TimeoutNotification node. A node in a message flow that processes time out request messages that are set by its associated TimeoutControl node and propagates copies or parts of the message to the next node in the message flow.

Topic. Used in publications and subscriptions to control the routing of publish/subscribe messages. A publication is about a particular topic.

Trace node. A node in a message flow used to generate trace records that can incorporate text, message content, and date and time information, to help monitor the behavior of a message flow. The records can be written to either the user trace file, another file, or the local error log (which contains error and information messages written by all other WebSphere Message Broker components).

User Name Server. Used to provide authentication and security for publish/subscribe in a broker domain.

Validate node. A node in a message flow that checks a message matches a given message structure definition. If it does it is output to the match terminal, else it is propagated to the failure terminal.

WebSphere Event Broker. Used for the distribution and routing of messages from disparate applications. Often used for publish/subscribe messaging.

WebSphere Message Broker. Provides storage, transformation and enrichment of data in addition to the functionality provided by WebSphere Event Broker.

WebSphere MQ. A messaging application which enables the Message Brokers Toolkit, Configuration Manager, and brokers to communicate. WebSphere MQ provides many of the available transport protocols between business applications and message flows.

WebSphere MQ Explorer. A graphical user interface for WebSphere MQ for administering WebSphere MQ components such as queue managers, channels and queues.

Windows Event Viewer. A Windows tool for viewing the contents of the Application and System logs.

XMLTransformation node. A node in a message flow that transforms an XML message to another form of XML message, according to the rules provided by an XSL (eXtensible Stylesheet Language) style sheet.

Abbreviations and acronyms

ACL	Access Control List
API	Application Programming Interface
BAR	broker archive file
CMP	Configuration Manager Proxy
ESB	enterprise service bus
ESQL	Extended Structured Query Language
HFS	Hierarchical File System
HTTP	Hypertext Transfer Protocol
IBM	International Business Machines Corporation
ISPF	Interactive System Productivity Facility
ITSO	International Technical Support Organization
MCA	Message Channel Agent
MIME	Multipurpose Internet Mail Extensions
MOM	Message-Oriented Middleware
MRM	Message Repository Manager
NPTL	Native POSIX Threading Library
ODBC	Open Database Connectivity
PSP	Preventative Service Planning
QoP	quality of protection
RAC	Rational Agent Controller
RAM	Random Access Memory
RDB	Relational Database
SLES	SUSE Linux Enterprise Server
SOA	service-oriented architecture
SOAP	Simple Object Access Protocol(SOAP)
SSL	Secure Sockets Layer
TSO	Time Sharing Option
URI	Uniform Resource Identifier
USS	UNIX System Services
UUID	Universally Unique ID
WSDL	Web Services Description Language
XML	eXtensible Markup Language
XML DTD	eXtensible Markup Language Document Type Definition
XMLNS	eXtensible Markup Language Namespace
XSL	eXtensible Stylesheet Language

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 568. Note that some of the documents referenced here may be available in softcopy only.

- *WebSphere MQ V6.0 Fundamentals*, SG24-7128
- *WebSphere Message Broker Basics*, SG24-7137

Online resources

These Web sites and URLs are also relevant as further information sources:

- Product documentation in the Information Center:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp

- This is the online version of WebSphere Message Broker V6.0:

  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp
  http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp?topic=/com.ibm.etools.mft.doc/ah24100_.htm

- High availability document for message broker:

  http://www-128.ibm.com/developerworks/websphere/library/techarticles/0403_humphreys/0403_humphreys.html

- Message broker requirements:

  http://www-306.ibm.com/software/integration/wbimessagebroker/requirements/

- WebSphere MQ documentation library

  http://www-306.ibm.com/software/integration/wmq/library/

- DB2 Information Center

  http://publib.boulder.ibm.com/infocenter/db2help/index.jsp

- Information about Rational Application Developer

  http://www-306.ibm.com/software/awdtools/developer/application/

- WebSphere Message Broker roadmap

  http://www-128.ibm.com/developerworks/websphere/zones/businessintegration/roadmaps/wsmb/

- WebSphere Message Broker support downloads

  http://www-1.ibm.com/support/docview.wss?rs=171&uid=swg27006367

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

Migrating to WebSphere Message Broker Version 6.0

(1.0” spine)
0.875”<->1.498”
460 <-> 788 pages

Migrating to WebSphere Message Broker Version 6.0

Coexistence of versions and migrations on multiple platforms

V2.1 migration to V6.0

V5.0 migration to V6.0

This IBM Redbook will help you to migrate WebSphere MQ Integrator V2.1 and WebSphere Business Integration Message Broker V5.0 to WebSphere Message Broker V6.0. It begins with an overview of the WebSphere Message Broker V6.0 solution, which plays a key role in the integration of disparate applications and platforms by providing functional and transport capabilities to support and facilitate enterprise-level business integration. Then it explains the new features and enhanced capabilities in WebSphere Message Broker V6.0, and describes the migration path from the previous supported versions.

The book provides extensive guidance and instruction about the various activities involved in migration including planning, backing up resources, migration steps, verification, coexistence, and restoration. It covers the migration of development resources, Configuration Manager, User Name Server, and brokers on multiple operating systems. It also shows you how to move the Configuration Manager component and development resources to other platforms.

The book covers both generic and platform-specific considerations. It also demonstrates the migration process, with the help of coexistence, on several major platforms using two scenarios, one for WebSphere MQ Integrator V2.1 and the other for WebSphere Business Integration Message Broker V5.0. The information included in this redbook complements but does not replace product documentation.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-7198-00 ISBN 0738494453