



Nastel Navigator Components

Installation Guide

Version 10

Document Number: NAV/COMP 101.018

Title: Nastel Navigator Components Installation Guide

Document Release Date: November 2022

Document Number: NAV/COMP 101.018

Published By:

R&D Department

Nastel Technologies, Inc.

88 Sunnyside Blvd, Suite 101

Plainview, NY 11803

Copyright © 2017-2022 by Nastel Technologies, Inc. All rights reserved. No part of the contents of this document may be produced or transmitted in any form, or by any means without the written permission of Nastel Technologies.

Confidentiality Statement: The information within this media is proprietary in nature and is the sole property of Nastel Technologies, Inc. All products and information developed by Nastel are intended for limited distribution to authorized Nastel employees, licensed clients, and authorized users. This information (including software, electronic and printed media) is not to be copied or distributed in any form without the expressed written permission from Nastel Technologies, Inc.

Acknowledgements: The following terms are trademarks of Nastel Technologies Corporation in the United States or other countries or both: AutoPilot/IT, APMW, AutoPilot Enterprise, AutoPilot/Web, M6 for Middleware, AutoPilot M6, MQControl, Navigator, XRay

The following terms are trademarks of the IBM Corporation in the United States or other countries or both: IBM, MQ, WIN-OS/2, AS/400, OS/2, DB2, AND AIX

The following terms are trademarks of Hewlett-Packard in the United States or other countries or both: OpenView, HP-UX

Compaq, the Compaq logo, AlphaServer, Compaq Insight Manager, CDA, DEC, DECnet, TruCluster, ULTRIX, and VAX Registered in U.S. Patent and Trademark Office. Alpha and Tru64 are trademarks of Compaq Information Technologies Group, L.P in the United States and other countries

SNMPc, SNMPc WorkGroup, and SNMPc Enterprise are Trademarks of Castle Rock Computing in the United States or other countries, or both.

Java and the Java Logos are trademarks of Sun Microsystems Inc. in the United States or other countries, or both.

InstallAnywhere is a registered trademark of ZeroG Software in the United States or other countries, or both.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). The Jakarta Project" and "Tomcat" and the associated logos are registered trademarks of the Apache Software Foundation

Intel, Pentium and Intel486 are trademarks or registered trademarks of Intel Corporation in the United States, or other countries, or both

Microsoft, Windows, Windows, Windows XP, the Windows logos, and SQL Server are registered trademarks of the Microsoft Corporation.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

"Linux" and the Linux Logos are registered trademarks of Linus Torvalds, the original author of the Linux kernel. All other titles, applications, products, and so forth are copyrighted and/or trademarked by their respective authors.

SCO CUSA, SCO Doctor, SCO Doctor for Networks, SCO Doctor Lite, SCO Global Access, SCO MPX, SCO MultiView, SCO Nihongo OpenServer, SCO OK, the SCO OK logo, SCO OpenServer, SCO Open Server, SCO Portfolio, SCO POS System, SCO ToolWare, and The World Never Stops are trademarks or registered trademarks of Caldera International, Inc. in the U.S.A. and other countries. All Rights Reserved.

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

Table of Contents

CHAPTER 1:	INTRODUCTION.....	9
1.1	HOW THIS GUIDE IS ORGANIZED.....	9
1.2	HISTORY OF THIS DOCUMENT.....	10
1.3	RELATED DOCUMENTS.....	11
1.4	RELEASE NOTES.....	11
1.5	INTENDED AUDIENCE.....	11
1.6	SYSTEM REQUIREMENTS.....	12
1.7	TERMS AND ABBREVIATIONS.....	12
1.8	TECHNICAL SUPPORT.....	12
1.9	NASTEL NAVIGATOR INSTALLATION SUPPORT (32-BIT/64-BIT).....	12
CHAPTER 2:	INSTALLING AND UPGRADING.....	14
2.1	INSTALLING COMPONENTS.....	14
2.2	UPGRADING COMPONENTS.....	14
CHAPTER 3:	WINDOWS INSTALLATION.....	16
3.1	NASTEL NAVIGATOR COMPONENTS FOR WINDOWS.....	16
3.2	INSTALLATION REQUIREMENTS.....	17
3.3	PACKAGE INSTALLATION.....	18
3.4	NASTEL NAVIGATOR ENVIRONMENT VARIABLES.....	27
3.5	USING CLIENT APPLICATIONS IN WINDOWS.....	28
3.6	UNINSTALLING NASTEL NAVIGATOR.....	38
CHAPTER 4:	UNIX INSTALLATION.....	40
4.1	NASTEL NAVIGATOR COMPONENTS FOR UNIX.....	40
4.2	UNIX INSTALLATION REQUIREMENTS.....	41
4.3	INSTALLATION.....	41
4.4	CONFIGURATION.....	42
4.5	STARTING AND STOPPING COMPONENTS.....	44
CHAPTER 5:	Z/OS INSTALLATION.....	53
5.1	NASTEL NAVIGATOR COMPONENTS FOR Z/OS.....	53
5.2	SYSTEM REQUIREMENTS.....	53
5.3	LIMITATIONS OF NASTEL NAVIGATOR ON Z/OS.....	53
5.4	UPGRADING NASTEL NAVIGATOR ON Z/OS.....	54
5.5	INSTALLING NASTEL NAVIGATOR.....	54
5.6	CONFIGURATION REQUIREMENTS FOR Z/OS.....	59
5.7	OPERATING NASTEL NAVIGATOR COMPONENTS VIA JCL PROCEDURES.....	66
5.8	STARTING IBM MQ AGENT.....	66
5.9	STOPPING THE IBM MQ AGENT.....	69
5.10	STARTING THE MESSAGE SERVER.....	71
5.11	STOPPING THE MESSAGE SERVER.....	72
5.12	STARTING THE EVENT PUBLISH-SUBSCRIBER.....	73
5.13	STOPPING THE EVENT PUBLISH-SUBSCRIBER.....	76
5.14	STOPPING MULTIPLE NASTEL NAVIGATOR COMPONENTS.....	77
5.15	UNINSTALLING NASTEL NAVIGATOR ON Z/OS.....	79
CHAPTER 6:	OS/400 INSTALLATION.....	81

6.1	NASTEL NAVIGATOR COMPONENTS FOR OS/400	81
6.2	SYSTEM REQUIREMENTS	81
6.3	SECURITY FOR OS/400.....	81
6.4	INSTALLING NASTEL NAVIGATOR.....	81
6.5	POST-INSTALLATION TASKS.....	87
6.6	STARTING THE IBM MQ AGENT.....	88
6.7	STOPPING THE IBM MQ AGENT	90
6.8	STARTING THE MESSAGE SERVER FOR IBM MQ	91
6.9	STOPPING THE MESSAGE SERVER FOR IBM MQ.....	92
6.10	STARTING THE PUBLISH SUBSCRIBER FOR IBM MQ	92
6.11	STOPPING THE PUBLISH SUBSCRIBER FOR IBM MQ.....	93
6.12	NASTEL NAVIGATOR UTILITIES ON OS/400	93
6.13	GETTING A SPOOL FILE FROM THE AS/400	101
6.14	UNINSTALLING NASTEL NAVIGATOR ON OS/400.....	102
CHAPTER 7: HP OPENVMS INSTALLATION		103
7.1	NASTEL NAVIGATOR COMPONENTS FOR OPENVMS	103
7.2	SYSTEM REQUIREMENTS	104
7.3	BEFORE YOU INSTALL.....	104
7.4	UPGRADING NASTEL NAVIGATOR ON VMS.....	104
7.5	INSTALLING NASTEL NAVIGATOR	105
7.6	VERIFYING FILES AND DIRECTORIES	106
7.7	SETTING NASTEL NAVIGATOR LOGICALS AND FOREIGN COMMANDS	109
7.8	CONFIGURING THE AGENT	109
7.9	VERIFYING THE AGENT UCX CONFIGURATION.....	111
7.10	REGISTERING THE OPENVMS NODE WITH A WORKGROUP SERVER.....	112
7.11	CONFIGURING THE MMF SERVER.....	112
7.12	STARTING THE MMF SERVER.....	113
7.13	VERIFYING THE MMF SERVER CONFIGURATION.....	114
7.14	STOPPING THE MMF SERVER.....	114
7.15	CONFIGURING THE MQ EVENT PUBLISHER.....	114
7.16	STARTING THE MQ EVENT PUBLISHER	116
7.17	STOPPING THE MQ EVENT PUBLISHER.....	116
7.18	SETTING NASTEL NAVIGATOR COMPONENT FOR AUTOMATIC STARTUP	116
7.19	USING THE UTILITY PROGRAMS	116
7.20	PERIODIC MAINTENANCE PROCEDURES	117
7.21	TROUBLESHOOTING PROCEDURES.....	117
7.22	UNINSTALLING NASTEL NAVIGATOR ON VMS.....	118
CHAPTER 8: LOG ADAPTER INSTALLATION		120
8.1	LOG ADAPTER.....	120
8.2	PROCESSING OF EVENTS.....	120
8.3	MANUAL INSTALLATION OF UNIX NASTEL NAVIGATOR LOG ADAPTER.....	120
8.4	START THE LOG ADAPTER USING THE SPI	122
8.5	STOP THE LOG ADAPTER ON A WINDOWS NODE USING THE SPI.....	122
8.6	START THE LOG ADAPTER ON A UNIX NODE	122
8.7	STOP THE LOG ADAPTER ON A UNIX NODE.....	125
8.8	START THE LOG ADAPTER FROM THE WINDOWS COMMAND LINE	125
8.9	TROUBLESHOOTING THE LOG ADAPTER	126
8.10	REASON CODE AND REASON QUALIFIER LOCATIONS	127

CHAPTER 9:	CONNECTION MANAGER FOR IBM MQ.....	129
9.1	PLATFORMS.....	129
9.2	PREREQUISITES	129
9.3	CONFIGURATION STEPS	130
9.4	REMOTE QUEUE MANAGER CONFIGURATION IN NASTEL NAVIGATOR	131
9.5	START CONNECTION MANAGER	133
9.6	ADD NASTEL NAVIGATOR IBM MQ OBJECTS	135
9.7	COMMAND-LINE ARGUMENTS	136
9.8	SETUP FOR CONNECTION MANAGER FOR IBM MQ WITH SSL.....	138
CHAPTER 10:	JAVA AGENT/CONNECTION MANAGER FOR IBM MQ.....	141
10.1	REQUIREMENTS	141
10.2	STARTING THE JAVA AGENT ON LINUX	141
10.3	NSQCOMM.PROPERTIES FILE STRUCTURE	141
CHAPTER 11:	CONNECTION MANAGER FOR TIBCO EMS	159
11.1	INSTALLING TIBCO EMS FOR UNIX AND WINDOWS.....	159
11.2	REQUIREMENTS	159
11.3	STARTING THE EMS CONNECTION MANAGER	159
11.4	NSQCMEMS.PROPERTIES FILE STRUCTURE	160
11.5	REMOTE EMS MANAGERS CONFIGURATION IN NASTEL NAVIGATOR.....	162
11.6	ADDING NASTEL NAVIGATOR EMS CONNECTION MANAGER NODES.....	164
11.7	ADD NASTEL NAVIGATOR EMS OBJECTS	165
11.8	RUNNING NSQCMEMS WITH SIMULTANEOUS CONNECTIONS TO MORE THAN 1 WGS.....	166
CHAPTER 12:	CONNECTION MANAGER FOR KAFKA	167
12.1	INSTALLING KAFKA FOR UNIX AND WINDOWS	167
12.2	STARTING THE KAFKA CONNECTION MANAGER	167
12.3	KAFKA NODE PROPERTIES FILE STRUCTURE	168
12.4	REMOTE KAFKA MANAGERS CONFIGURATION IN NASTEL NAVIGATOR.....	170
12.5	ADD NASTEL NAVIGATOR KAFKA OBJECTS	171
12.6	KAFKA CONFIGURATION VALUES.....	172
12.7	KAFKA CONNECTION CONFIGURATION EXAMPLES.....	172
CHAPTER 13:	CONNECTION MANAGER FOR IBM IIB/ACE.....	175
13.1	INSTALLING IIB/ACE FOR UNIX AND WINDOWS	175
13.2	STARTING THE IIB/ACE CONNECTION MANAGER	175
13.3	IIB/ACE PROPERTIES FILE STRUCTURE	175
13.4	REMOTE IIB/ACE MANAGERS CONFIGURATION IN NASTEL NAVIGATOR	177
13.5	ADDING NASTEL NAVIGATOR CONNECTION MANAGER NODES	180
13.6	ADD NASTEL NAVIGATOR IIB/ACE OBJECTS.....	181
CHAPTER 14:	CONNECTION MANAGER FOR SOLACE	182
14.1	INSTALLING SOLACE FOR UNIX AND WINDOWS.....	182
14.2	STARTING THE SOLACE CONNECTION MANAGER	182
14.3	SOLACE PROPERTIES FILE STRUCTURE	182
14.4	REMOTE SOLACE MANAGERS CONFIGURATION IN NASTEL NAVIGATOR.....	184
14.5	ADDING NASTEL NAVIGATOR CONNECTION MANAGER NODES	186
14.6	ADD NASTEL NAVIGATOR SOLACE OBJECTS.....	188

CHAPTER 15: NASTEL NAVIGATOR COMPONENT INSTALLATION IN MULTI-INSTANCE QUEUE MANAGER SETUP	189
15.1 NASTEL NAVIGATOR QUEUES UNIX.....	189
15.2 INSTALLING NASTEL NAVIGATOR COMPONENTS FOR UNIX.....	189
15.3 INSTALLING NASTEL NAVIGATOR COMPONENTS FOR WINDOWS.....	189
CHAPTER 16: Z/OS ACCOUNTING AND STATISTICS (ZAS) COLLECTOR	191
16.1 CONFIGURINGNSQZAS.....	193
16.2 STARTINGNSQZAS.....	196
16.3 STOPPING/RESTARTINGNSQZAS.....	196
16.4 MAPPING BETWEEN ACCOUNTING AND STATISTICS PCF MESSAGES ON Z/OS AND DISTRIBUTED PLATFORMS 196	
16.5 BEST PRACTICES FOR ZAS COLLECTION.....	218
16.6 INITIALIZING SMF DATA SETS.....	220
16.7 SMF COMMANDS.....	221
16.8 USEFUL WMQ COMMANDS.....	224
16.9 UTILITIES.....	225
16.10 SMF EXIT JCL LISTING.....	235
CHAPTER 17: NSQPRID UTILITY	237
17.1 DESCRIPTION.....	237
17.2 COMMAND.....	237
17.3 SAMPLE OUTPUT.....	237
CHAPTER 18: NSQMIGR UTILITY	239
18.1 PARAMETERS.....	239
18.2 GET / PUT FILE.....	240
18.3 IMPORTING/EXPORT WORKGROUP SERVER DATABASES.....	240
18.4 IMPORT/EXPORT REMOTE QUEUE MANAGERS.....	243
18.5 IMPORT/EXPORT REMOTE EMS SERVERS.....	246
APPENDIX A: NASTEL AND IBM DOCUMENTATION	249
A.1 NASTEL DOCUMENTATION.....	249
A.2 IBM DOCUMENTATION.....	249
APPENDIX B: NASTEL NAVIGATOR COMMAND LINES BY COMPONENT	251
B.1 IBM MQ AGENT.....	251
B.2 MMF SERVER.....	254
B.3 MQ EVENT PUBLISHER.....	256
B.4 MQSC UTILITY.....	259
B.5 NSQPING (PING) UTILITY.....	260
B.6 NSQCODE UTILITY.....	261
B.7 LOG ADAPTER.....	262
B.8 MESSAGE MANAGEMENT UTILITY.....	264
B.9 QUEUE MONITOR UTILITY.....	265
B.10 MESSAGE GET/PUT UTILITY.....	265
B.11 REMOTE SHUTDOWN UTILITY.....	273
B.12 TRIGGER MESSAGE UTILITY.....	275
B.13 IBM MQ EVENT ADAPTER.....	276

APPENDIX C: QUEUE NAMING CONVENTIONS AND QUALIFIERS.....	277
APPENDIX D: NASTEL NAVIGATOR IBM MQ OBJECTS	279
GLOSSARY	282
INDEX	287

Chapter 1: Introduction

Welcome to the *Nastel Navigator Components Installation Guide*. This guide explains how to install Nastel Navigator related components: platform-dependent agents, agentless connection manager, and utilities; a complete management solution for IBM MQ, IIB/ACE, TIBCO EMS and Kafka on all supported platforms.



NOTE

This book contains instructions for installing agents, connection managers and utilities. If you need to install workgroup server components, please read the [Nastel Navigator Workgroup Server Expert V10 – Installation Guide](#) instead.

1.1 How this Guide is Organized

[Chapter 1:](#) Introductory information about this document and Nastel Navigator.

[Chapter 2:](#) Information on installing and upgrading Nastel Navigator components.

[Chapter 3:](#) MS Windows installation instructions and information.

[Chapter 4:](#) UNIX installation instructions and information.

[Chapter 5:](#) z/OS installation instructions and information.

[Chapter 6:](#) OS/400 installation instructions and information.

[Chapter 7:](#) HP OpenVMS installation instructions and information.

[Chapter 8:](#) Log Adapter installation instructions and information.

[Chapter 9:](#) Instructions on how to set up the Connection Manager for IBM MQ.

[Chapter 10:](#) Instructions on how to set up version 10.x Java Agent / Connection Manager.

[Chapter 11:](#) Instructions on how to set up the Connection Manager for TIBCO EMS.

[Chapter 12:](#) Instructions on how to set up the Connection Manager for Kafka.

[Chapter 13:](#) Instructions on how to set up the Connection Manager for IBM IIB/ACE.

[Chapter 14:](#) Instructions on how to set up the Connection Manager for Solace.

[Chapter 15:](#) Instructions and information for Nastel Navigator agent installation in multi-instance queue manager setup.

[Chapter 16:](#) Information on the Z/OS Accounting and Statistics (ZAS) Collector.

[Chapter 17:](#) Information on using the nsqprid Utility, which creates a specific user/password for a workgroup server.

[Chapter 18:](#) Information on using the nsqmigr Utility, which is used to export and import workgroup server and node information in bulk.

[Appendix A:](#) Detailed list of all reference information required for the operation of Nastel Navigator.

[Appendix B:](#) Summary of features across the supported platforms.

[Appendix C:](#) Information on qualifiers and queue naming conventions.

[Appendix D:](#) Listing of Nastel Navigator and IBM MQ objects that are created by Nastel Navigator components.

[Glossary:](#) Listing of unique and common acronyms, words, and definitions.

[Index:](#) Alphanumeric listing of all topics and subjects of importance.

1.2 History of this Document

Table 1-1. Document History

Release Date	Document Number	Version	Summary
July 2020	NAV/COMP 101.001	10	Initial release
July 2020	NAV/COMP 101.002	10	Updates to first paragraph of Section 5.6.1.
August 2020	NAV/COMP 101.003	10	Add Chapters 13 and 14.
September 2020	NAV/COMP 101.004	10	Update Figure 1-1.
December 2020	NAV/COMP 101.005	10	Update package names throughout. Update table 5-1. Remove HP-UX. Update section 9.2 (add +a info).
March 2021	NAV/COMP 101.006	10	Update sections 6.5 and 7.8.
March 2021	NAV/COMP 101.007	10	Update sections 10.4 and 11.2 – add information on multiple connections.
April 2021	NAV/COMP 101.008	10	Add Chapter 12: IIB/ACE Connection Manager. Add sections B.10.3 (Options to Get Messages from Queue to File) and B.10.4 (Options to Put Messages from File to Queue).
May 2021	NAV/COMP 101.009	10	Update section 15.1 (nsqmigr parameters) – added -k flag.
May 2021	NAV/COMP 101.010	10	Update information on +a in section 9.2 (Connection Manager for IBM MQ Prereqs).
May 2021	NAV/COMP 101.011	10	Update step 3 in section 4.5.1.1 (Starting IBM MQ Agent).
May 2021	NAV/COMP 101.012	10	Update section 6.12.2 (Message File (nsqmsgf) Utility). Updated usage display text.
May 2021	NAV/COMP 101.013	10	-Updates to Table 3-1 (Custom Installation Components). Update sections 5.8.1 and 5.9.

Table 1-1. Document History

Release Date	Document Number	Version	Summary
May 2021	NAV/COMP 101.014	10	Add section 11.2: Starting the Kafka Connection Manager. Add section 12.2: Starting the IIB/ACE Connection Manager. Update section 10.3.
June 2021	NAV/COMP 101.015	10	Update #4 in section 6.5 - Post-Installation Tasks
June 2021	NAV/COMP 101.016	10	Update Figure 1-1 (Nastel Navigator Installation Support (32-bit/64-bit))
January 2022	NAV/COMP 101.017	10	Update sections 6.12.3 and 6.12.4. Added ZAS Collector information as chapter 14. Updated some online references. Added Nastel Navigator product name and procedures where applicable.
November 2022	NAV/COMP 101.018	10	Added chapters for Solace and Java Agent/Connection Manager for IBM MQ. Added cross-references to chapters and sections of Navigator guide to end of instructions for each component.

1.2.1 User Feedback

Nastel encourages all users and administrators of Nastel Navigator to submit comments, suggestions, corrections, and recommendations for improvement for all Nastel Navigator documentation. Please send comments via e-mail to: support@nastel.com. You will receive a response, along with status of any proposed change, update, or correction.

1.3 Related Documents

Complete listings of related and referenced documents related to Nastel Navigator can be found in [Appendix A](#).

1.4 Release Notes

See README.HTM file and README_INSTALL.txt on installation media or Workgroup Server Expert installation directory. Release notes and updates are also available through the [Nastel Resource Center](#).

1.5 Intended Audience

The installation guide is intended for:

System administrators who manage IBM MQ, IIB/ACE, TIBCO EMS, Kafka or other networks.

Installers of Nastel Navigator on IBM MQ, IIB/ACE, TIBCO EMS, Kafka or other networks.

Readers should have a general understanding of middleware, network protocols, and systems management.

1.6 System Requirements

System requirements vary by component and platform. For information about your platform, see the appropriate platform chapter in this guide, and included in the installation downloadable from our FTP site.

1.7 Terms and Abbreviations

A list of Terms and Abbreviations used in this document is in the [Glossary](#).

1.8 Technical Support

For technical support, visit the Nastel Resource Center at <https://customers.nastel.com/>.

To contact Nastel technical support by telephone, call **800-963-9822**. If you are calling from outside the United States, dial **1-516-801-2100**.

To contact Nastel technical support by e-mail, send a message to support@nastel.com. You can also contact Nastel support via the support website (user ID/Password required), go to: <http://support.nastel.com/>.

1.9 Nastel Navigator Installation Support (32-bit/64-bit)

See [Figure 1-1](#) for Nastel Navigator Installation Support. This platform is for the workgroup server and databases supported. Agents may be available for other versions of operating systems and middleware versions. Contact your support representative if your platform is not listed.

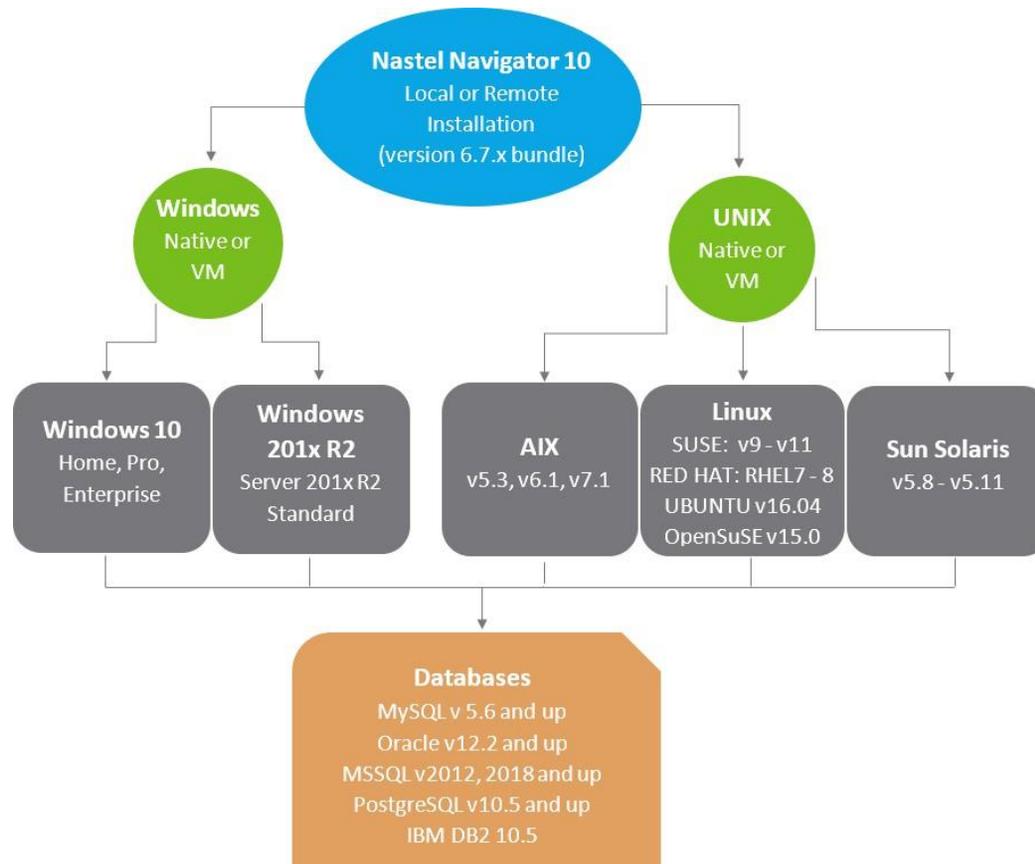


Figure 1-1. Nastel Navigator Installation Support (32-bit/64-bit)

Chapter 2: Installing and Upgrading

2.1 Installing Components

This book contains instructions for installing agents, connection managers and utilities. If you need to install workgroup server components, please read the [Nastel Navigator Workgroup Server Expert V10 - Installation Guide](#) instead.

Instructions on installing Nastel Navigator components are in the proceeding chapters. Please refer to the chapter related to your platform:

- [Chapter 3: Windows](#)
- [Chapter 4: UNIX](#)
- [Chapter 5: z/OS](#)
- [Chapter 6: OS/400](#)
- [Chapter 7: HP OpenVMS Installation](#)

2.2 Upgrading Components



These instructions apply if prior version was 6.5.0 or higher. If you have a version earlier than 6.5.0 installed, you must uninstall the pre 6.5.0 version and re-install the 6.6.0 base version.

2.2.1 E-Fixes

Software changes come in two forms:

- An e-fix is a cumulative update to a base version. For example, the e-fix 6.5.1 is a cumulative update to version 6.5.
- A hot fix contains specific files required to fix an issue with a specific e-fix version. For example, hot fix 6.5.1.2 would contain fixes to e-fix 6.5.1.

The packaging is the same for both; only the content is different. When applying an e-fix, only the latest is required. For example, 6.5.2 includes changes from 6.5.0 and 6.5.1, so you could apply e-fix 6.5.2 to either 6.5.0 or 6.5.1. Hot fixes can only be applied to their respective e-fix version. That is, hot fix 6.5.1.2 is specific to e-fix 6.5.1, so hot fix 6.5.1.2 cannot be applied to e-fix 6.5.2. The release notes, linked from the README files contained with the fixes, will describe any special considerations.

2.2.2 Windows Component Upgrade

Nastel provides a zip file which replaces your existing installation files. An example of an e-fix file is `Nav_Components_10.1.5_WIn_x86.zip`.

1. Stop all services and processes.
 2. Backup the existing program files in case you need to roll back the changes.
 3. Open the e-fix zip file.
-

4. Copy all the files.
5. Paste the files over your existing installation, which is probably in a directory like `c:\nastel\apwmq`. When prompted to copy and replace an existing file, select **Yes**.

2.2.3 Unix/Linux Component Upgrade

Nastel provides a compressed or tar file which replaces your existing installation files. An example of an e-fix file is `Nav_Components_10.1.5_Linux_Rhel5_x86_64.tar.gz`.

1. Stop all services and processes.
2. Backup the existing program files in case you need to roll back the changes.
3. Decompress the file if required.
4. Change Directory to the folder containing Nastel Navigator, for example, `/opt/nastel/apwmq`
5. Untar the files replacing the existing files
(`tar -xvf Nav_Components_10.1.5_Linux_Rhel5_x86_64.tar`)

Chapter 3: Windows Installation

This chapter defines installation of Nastel Navigator on the supported Windows platforms.

3.1 Nastel Navigator Components for Windows

The following Nastel Navigator components can be installed on Windows:

Clients

- Nastel Navigator Explorer
- Message Explorer
- Workgroup Statistics

Utilities

Software Development Kit (SDK)

Books

3.1.1 Custom Installation Components

The following table lists the Nastel Navigator components necessary for a custom installation. Note that a custom *Client only* installation is only available for Windows installations.

Workgroup Server V6 has been replaced by V10. Go to the [Nastel Navigator Workgroup Server Expert V10 - Installation Guide](#) book to install it.

Table 3-1. Custom Installation Components		
	Nastel Navigator Components	Binaries (Executables)
Agents only	IBM MQ Agent	APWMQ_HOME\bin\nsqmq
	Message Manager for IBM MQ	APWMQ_HOME\bin\nsqmsg
	Publish Subscriber for IBM MQ	APWMQ_HOME\bin\nsqpub
	Connection Manager IBM MQ	APWMQ_HOME\bin\nsqcm
Workgroup server only	Workgroup Server	APWMQ_HOME\bin\nsqmgr
Client only	WMQ Explorer	APWMQ_HOME\bin\nsqexplr.exe
	Message Explorer	APWMQ_HOME\bin\nsqmsgex.exe
	Workgroup Statistics	APWMQ_HOME\bin\nsqclnex.exe

3.2 Installation Requirements

The requirements outlined in the paragraphs below specify minimum operating standards for Nastel Navigator.

3.2.1 Hardware Requirements:

- Intel-based PC (compatible) capable of supporting Windows operating systems
- TCP/IP communication hardware
- 60-100MB free disk space for workspace

3.2.1.1 Agents

IBM MQ 7.0 and higher requirements are sufficient to run lightweight agents.
For systems running IBM MQ 6.x, you should install and use the V6.1 AutoPilot for IBM MQ Agents.

3.2.1.2 Clients

Minimum:

- 1 GHZ or higher CPU
- 512 MB RAM
- 60 MB free disk space.

Recommended:

- 1 GHZ or higher CPU
- 1 GB RAM
- 100 MB free disk space.

3.2.2 System Requirements

3.2.2.1 Minimum Software Requirements

- IBM MQ 7.0.1.2
- Microsoft Windows 2008 and above
- Windows



These and other Microsoft updates are available from Microsoft web site at: <http://www.microsoft.com>
When installing Nastel Navigator on Windows 2008, it should be installed as user Administrator.
Otherwise, all Nastel Navigator Windows services and some of the Start > Program > Nastel APWMQ icons will not get installed.

3.3 Package Installation

There are two types of installations available for Nastel Navigator on Windows:

New Installation – no previous installation of Nastel Navigator exists.

Upgrade Installation – a previous installation exists that is prior to V6.5.

The Nastel Navigator installation media automatically detects which type of installation you have and will navigate you to the correct installation screens.



1. The installation procedure below covers a new installation. If you are upgrading your installation, non-required screens will not be displayed.
2. The msc scripts nastel.tst, pubsub.tst and mmf.tst are automatically installed using this procedure.

To install Nastel Navigator, do the following:

1. If Nastel Navigator previously installed, stop all Nastel Navigator services and components.
2. Run **setup.exe**:

Run as... and select user **Administrator**. (Refer to note in [section 3.2.2.1.](#))



Name	Type	Compressed size
ISSetupPrerequisites	File folder	
Installation Guide.pdf	PDF File	5,937 KB
README_INSTALL.txt	Text Document	2 KB
setup.exe	Application	87,718 KB
SILENTINSTALL.txt	Text Document	6 KB

Figure 3-1. Setup.exe

3. The *Welcome* screen is displayed. When the wizard setup is complete, click **Next** to continue.



Figure 3-2. Installation Wizard Welcome Screen

4. When the *License Agreement* screen is displayed, click *I accept the terms in the license agreement* to continue the installation. Click **Next** to continue. If you do not accept the terms, you cannot proceed.

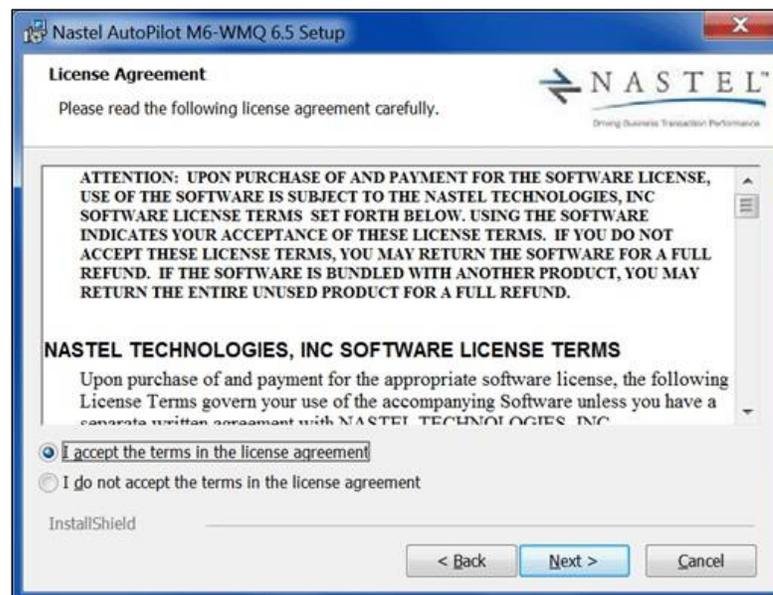


Figure 3-3. License Acceptance

5. When the *Customer Information* screen is displayed enter the following:
User Name: System assigned default name of Windows user
Organization: System assigned default name of Windows user

Anyone who uses this computer (all users): When selected, the windows programs menu will display shortcuts for all users. This is the default.

Only for me: When selected, only the user specified would have shortcuts installed during the installation process. Nastel Navigator can be manually applied by all users after installation.

Click **Next** to continue.

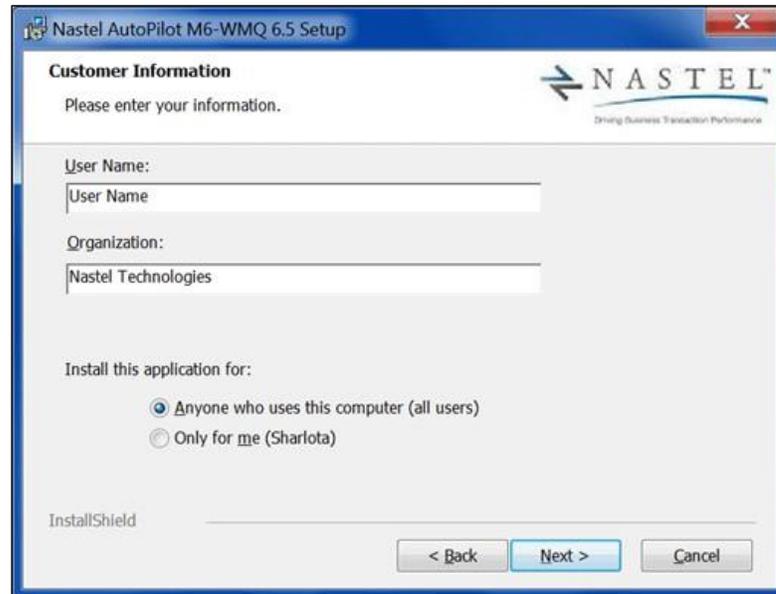


Figure 3-4. Customer Information

6. When the *M6-WMQ Setup Type* screen is displayed, select one of the following:

Full: Do not use this option.

Partial: Installs only the categories selected from the following screen. All features associated with the selected category will also be installed. This is the default.

Custom: Installs only the features selected from the following screen.

Click **Next** to continue.



Figure 3-5. Installation Setup

7. If **Partial** installation was selected in previous step, continue with step 9. If **Custom** installation was selected in previous step, continue with next step (step 8).
8. If **Custom** installation was selected in previous step, you will need to select components and options to be installed on this machine.
 - a. Use selection options for each feature or sub-feature as needed using the options depicted in the Figure below. (See [Table 3-1](#) above for a listing of the Nastel Navigator components necessary for a custom installation. See [Table 3-2](#) below for a complete list of installation options and a brief description.) All features are selected for installation to local hard drive by default. Select one of the installation options in the installation menu for each feature or sub-feature as required. Click on down arrow next to feature to select installation option for each feature installed.

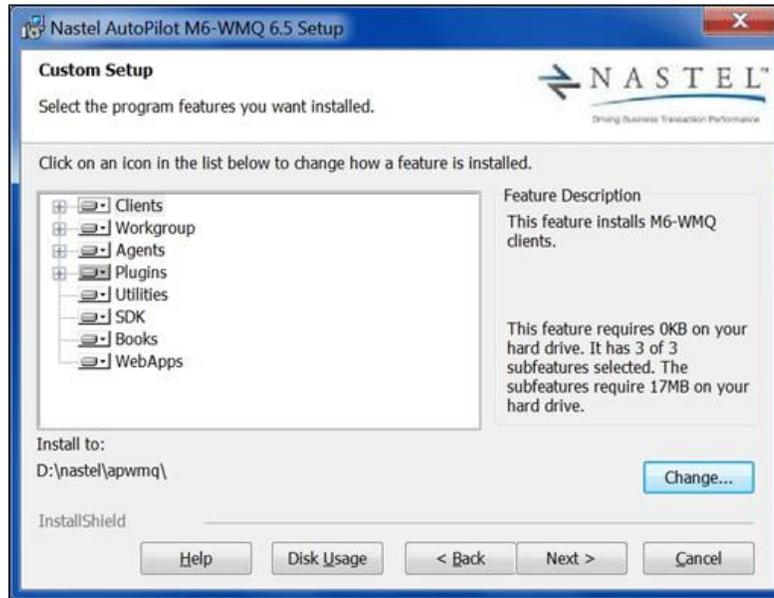


Figure 3-6. Custom Installation Options

Table 3-2. Nastel Navigator Features			
Feature	Sub-feature	Description	Disk Space Required
Clients	Installs Nastel Navigator clients		
	Nastel Navigator Explorer	Provides visual interface for managing IBM MQ objects.	12MB
	Message Explorer	Enables IBM MQ Message Management interface.	4364KB
	Workgroup Statistics	Enables workgroup communication interface.	946KB
Workgroup (no longer used)	Provides management of registered IBM MQ nodes.		1708KB
	Workgroup Server	Monitors IBM MQ nodes.	617KB
Agents	Provides agents to monitor and manage local and remote IBM MQ environments.		
	IBM MQ Agent	Provides agent to monitor local IBM MQ queue managers.	1008KB
	Message Manager for IBM MQ	Provides server component for IBM MQ message management.	660KB
	Publish Subscriber for IBM MQ	Provides IBM MQ event publishing to registered subscribers and dead letter queue monitoring.	472KB
	Connection Manger for IBM MQ	Provides the agent to administer remote IBM MQ queue managers.	589KB
Plugins			

Table 3-2. Nastel Navigator Features			
Feature	Sub-feature	Description	Disk Space Required
ITO		Provides the adapter to integrate with HPOV (ITO) Enterprise Management solution.	
	ITO Adapter	Interfaces workgroup server with HPOV (ITO).	464KB
Tivoli		Provides the adapter to integrate with Tivoli Enterprise Management solution.	
	Tivoli Adapter	Interfaces workgroup server with Tivoli.	444KB
Utilities		Provides various IBM MQ and Nastel Navigator utilities.	572KB
SDK		Provides Nastel Navigator API, including a set of libraries and program samples.	488KB
WebApps		Provides the Nastel Navigator and Nastel Navigator Security Manager war files to be deployed to an AppServer.	1500KB

- b. The default destination for the installation is C:\nastel\apwmq. Change as needed by clicking **Change** and enter the new folder name and address.

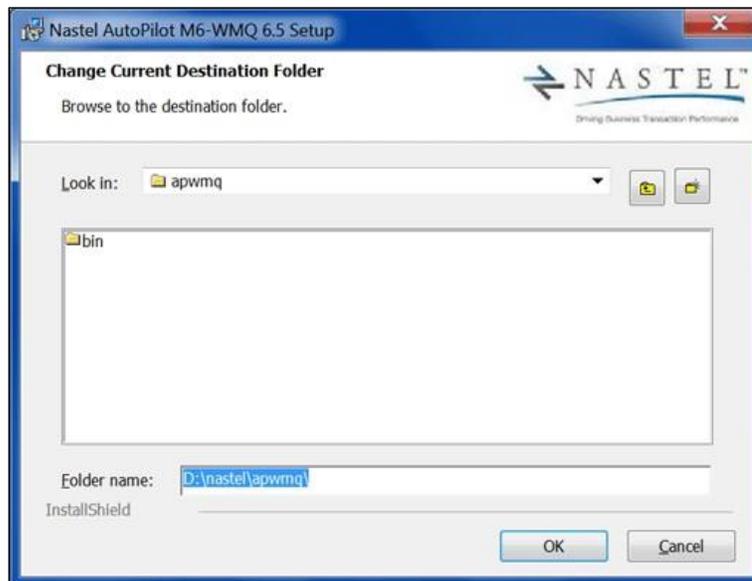


Figure 3-7. Change Destination Folder

- c. Click **Next** and continue with step 10, configuring IBM MQ Agents.
9. If **Partial** installation was selected in step 6, you will need to select the category of features to be installed on this machine. See [Table 3-2](#) above for a complete list of installation categories and a brief description. All categories are selected for installation to local hard drive by default.

Click **Next** to continue.



Figure 3-8. Partial Installation Feature Categories

10. Configure IBM MQ Agents as follows:

- **M6-WMQ Agent port:** The default port is 5010, change as required.
- **Connection Manager port:** The default port is 5020, change as required.
- **Install agents as Windows services:** This option is selected by default and is recommended.

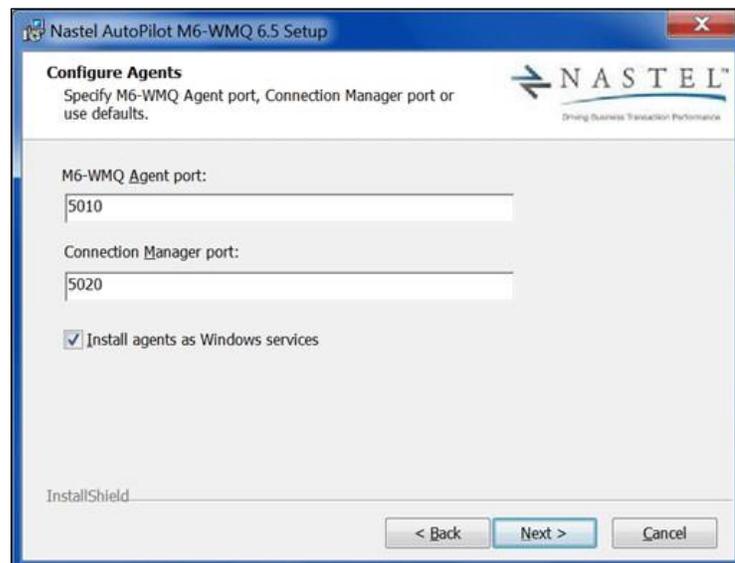


Figure 3-10. Configuring Agents

11. **For Full installation or Custom installation and Partial installation if plug-ins option was selected:** Configure plug-ins as required. If any sub-feature option is selected to not be installed, this screen will appear as shown, deselect the option.

Log Adapter (ITO or Tivoli):

Install Log Adapter as Windows service: This option is selected by default and is recommended. Remain checked to install as Windows service.

Click **Next** to continue.



Figure 3-11. Configure Plug-ins

12. Ensure you have reviewed your installation options before starting installation and click **Back** if any changes are required. Click **Install** to begin installation.

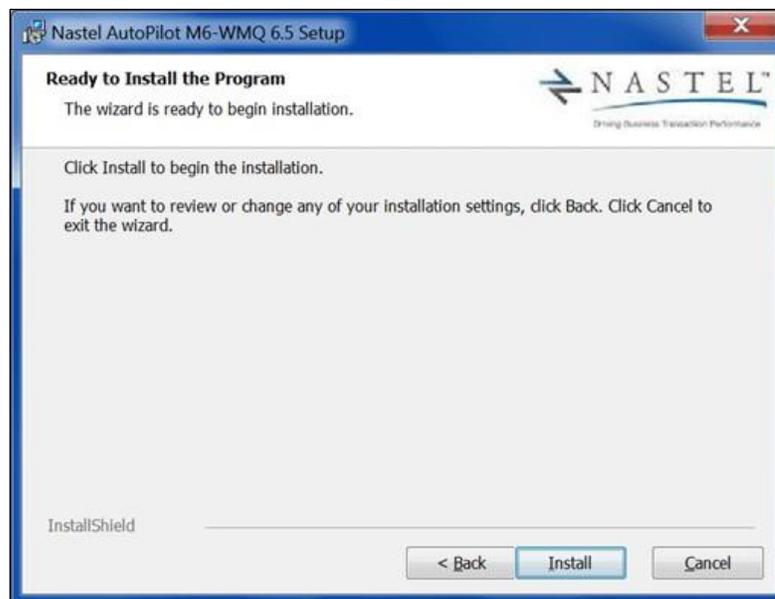


Figure 3-12. Ready to Install

13. Installation progress is monitored in the status bar. You have the option to cancel at any time during the installation process.



Figure 3-13. Installation Progress

14. When the installation is complete, the following screen will be displayed. Click **Finish**.



Figure 3-14. Completing the Installation

15. When Setup is complete you may be instructed to restart your machine before you can use Nastel Navigator, however Windows typically handles this automatically and restart is required.



Figure 3-15. Restarting Your Machine

3.4 Nastel Navigator Environment Variables

In Windows, the following environment variables are automatically defined and exported:

- APWMQ_HOME – points to Nastel Navigator installation directory.
- PATH – includes path to Nastel Navigator bin directory.

The following environment variables are optional:



Environment variables with the `_<wgrp>` appended to them are only processed when they run as a windows service. They have no effect when the processes are started from a command prompt.

- APWMQ_LS_OPTIONS – contains Local Agent startup options.
- *If running multiple Local Agent services and each instance requires different options, you can define environment variables of the form APWMQ_LS_OPTIONS_<wgrp>, where <wgrp> is the name of the workgroup that the Workgroup Server is processing. These settings will override those defined in APWMQ_LS_OPTIONS.*
- APWMQ_MS_OPTIONS – contains Nastel Navigator Message Server startup options.
- APWMQ_EMS_HOME – contains the location of the Connection Manager for EMS property file (`nsqcmems.properties`). Please note, does not support <wgrp> mentioned in the NOTE above.
- APWMQ_EVT_OPTIONS – contains Nastel Navigator Event Adapter startup options.
- *If running multiple Event Adapter services and each instance requires different options, you can define environment variables of the form APWMQ_EVT_OPTIONS_<wgrp>, where <wgrp> is the name of the workgroup that the Workgroup Server is processing. These settings will override those defined in APWMQ_EVT_OPTIONS.*
- APWMQ_LA_OPTIONS – contains Nastel Navigator Log Adapter startup options.
- *If running multiple Log Adapter services and each instance requires different options, you can define environment variables of the form APWMQ_LA_OPTIONS_<wgrp>, where <wgrp> is the name of the workgroup that the Workgroup Server is processing. These settings will override those defined in APWMQ_LA_OPTIONS.*
- APWMQ_PS_OPTIONS – contains Nastel Navigator Publish-Subscriber startup options.
- APWMQ_CM_OPTIONS – contains Nastel Navigator Connection Manager startup options.

- If running multiple Connection Manager services and each instance requires different options, you can define environment variables of the form `APWMQ_CM_OPTIONS_<wkgrp>`, where `<wkgrp>` is the name of the workgroup that the Workgroup Server is processing. These settings will override those defined in `APWMQ_CM_OPTIONS`.

3.5 Using Client Applications in Windows

The following defines the procedures for verifying installation in Window environments.

3.5.1 Starting Nastel Navigator in Windows Environments



Please note that Nastel recommends using the web version of Nastel Navigator, accessible at http://%3Cmachine_name%3E:8080/navigator/.

1. Click **Start, Programs**, click **AutoPilot-WMQ** to open the Nastel Navigator program menu.
2. Click **M6-WMQ Explorer**.

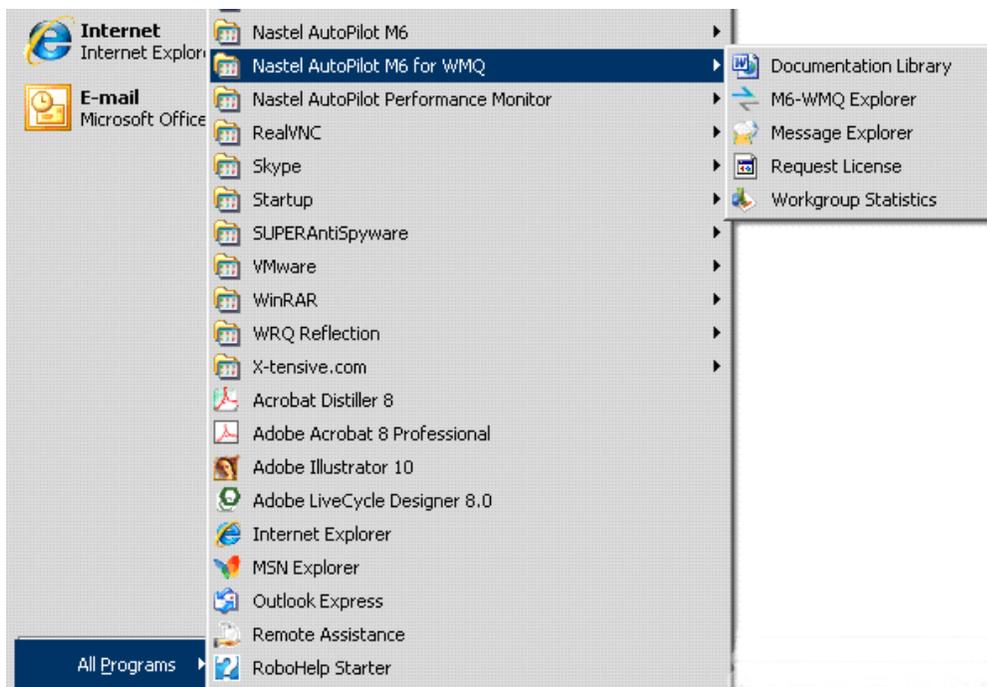


Figure 3-16. Windows Program Group

The Nastel Navigator splash and Logon screens are automatically displayed.



Figure 3-17. Nastel Navigator Splash

3. On the **General** tab, use the down arrow to select the Security Realm you will be using to log on. This will depend on the security method used. Please contact your system administrator for specifics.

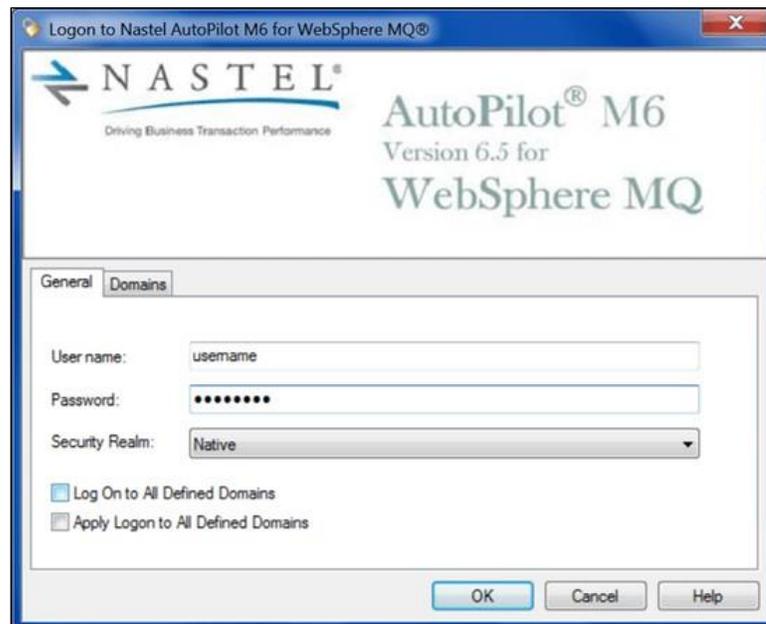


Figure 3-18. User Login - General Tab

4. Enter **User name** and **Password** (**Admin** is the default user name, **admin** is the default password).
5. To select the workgroup server to log on to:
 - a. If you want to log on to all workgroups defined under the **Domains** tab, select *Log On to All Defined Domains* checkbox (domains must have been previously defined). Click **OK** and exit.

- b. If you want the logon information on this screen to be used for all domains being logged in to, select *Apply Logon to All Defined Domains* checkbox. This will override any previous user name or security realm defined for that domain. Click **OK**.
- c. If you want to log onto one or more workgroups, but not all of them, select the **Domains** tab. The following screen displays workgroups that have already been defined in M6-WMQ Explorer.

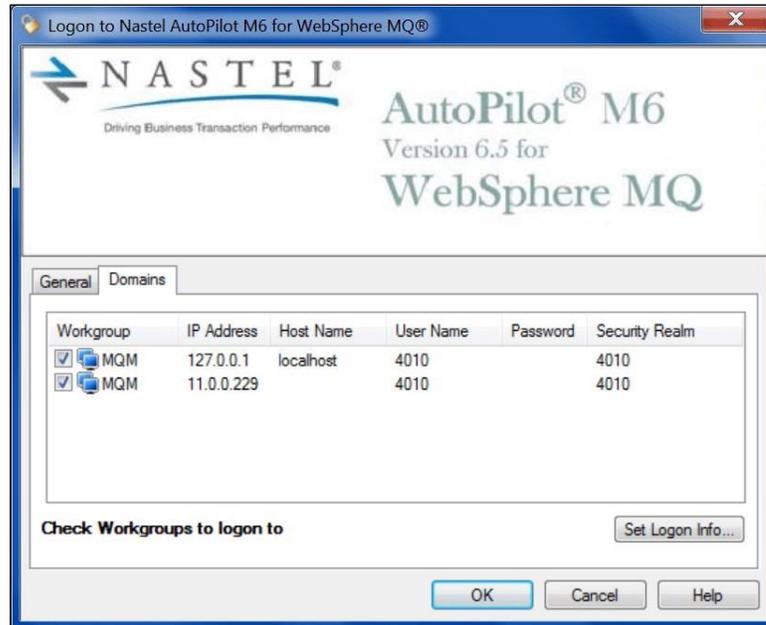


Figure 3-19. User Login - Domains Tab

- d. To set User name, Password, and Security Realm for a workgroup, highlight the workgroup and click **Set Logon Info** to display the following screen:



Figure 3-20. Login Information for Workgroup

- e. Enter information in all three fields and click **OK** to be returned to the User Workgroup Login screen. Repeat for all workgroups as necessary.
- f. Select the checkbox(s) next to the workgroup(s) that you want to logon to. Click **OK** to be returned to the **General** tab of the logon screen.

6. Click **OK**.

NOTE By default, the User ID is not authenticated via domain security service. The Windows user ID will be used for IBM MQ object authorization. To enable User ID authentication via Nastel Navigator workgroup server, start-up options must be changed. See [Appendix B](#) for additional details.

The default *M6-WMQ Explorer* screen will be displayed.

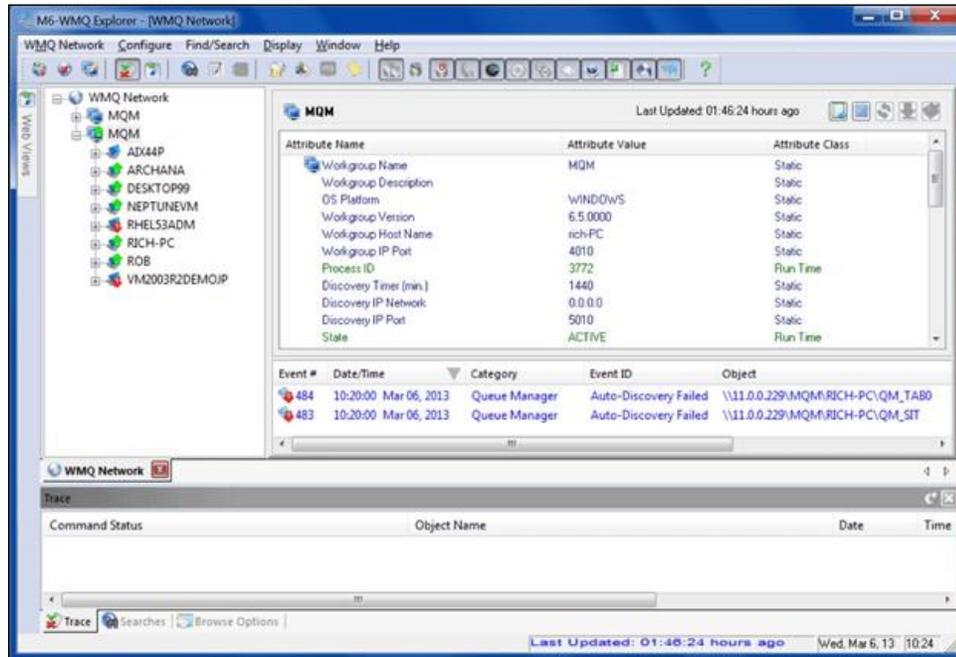


Figure 3-21. M6-WMQ Explorer Default Screen

3.5.2 Starting the Message Explorer for IBM MQ

1. Click **Start, Programs**, click **AutoPilot-WMQ** to open Nastel Navigator program menu.
2. Click **Message Explorer**.

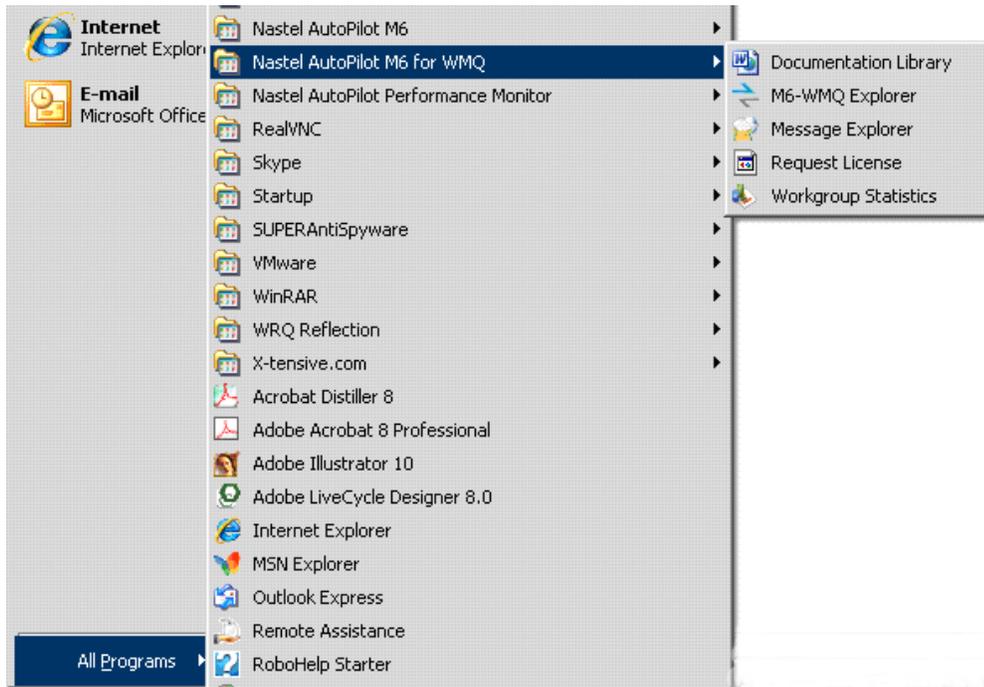


Figure 3-22. Windows Program Group

The Nastel Navigator Splash and Logon screens are automatically displayed.



Figure 3-23. Nastel Navigator Splash

3. On the **General** tab, use the down arrow to select the Security Realm you will be using to log on. This will depend on the security method used. Please contact your system administrator for specifics.

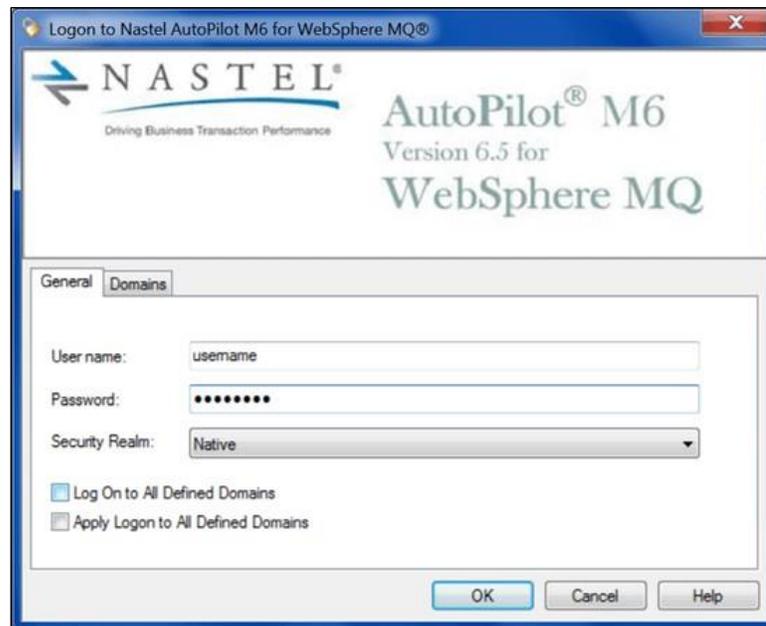


Figure 3-24. User Authentication for Message Explorer

4. Enter User name and Password (admin is the default user name).
5. To select the workgroup server to log on to:
 - a. If you want to log on to all workgroups defined under the **Domains** tab, select *Log On to All Defined Domains* checkbox. (Domains must have been previously defined.) Click **OK** and exit this procedure.
 - b. If you want the logon information on this screen to be used for all domains being logged in to, select *Apply Logon to All Defined Domains* checkbox. This will override any previous user name or security realm defined for that domain. Click **OK**.

- c. If you want to log onto one or more workgroups, but not all of them, select the **Domains** tab. The following screen displays workgroups that have already been defined in M6-WMQ Explorer.

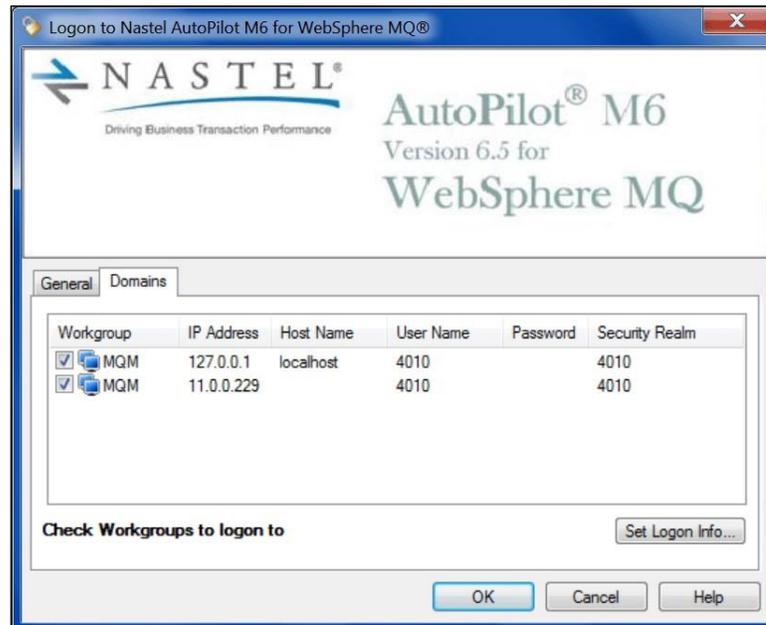


Figure 3-25. User Workgroup Login

- i. To set User name, Password, and Security Realm for a workgroup, highlight the workgroup and click **Set Logon Info** to display the following screen:

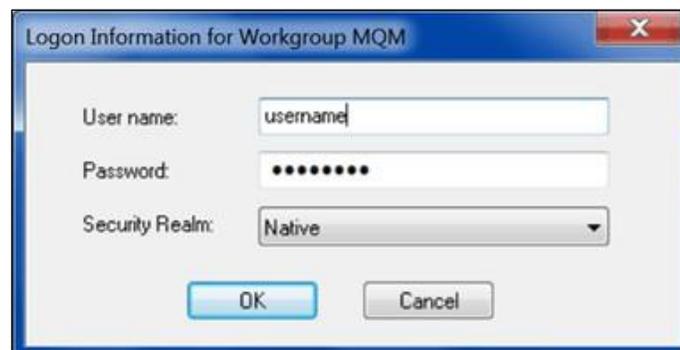


Figure 3-26. Login Information for Workgroup

- ii. Enter information in all three fields and click **OK** to be returned to the User Workgroup Login screen. Repeat for all workgroups as necessary.
- iii. Select the checkbox(s) next to the workgroup(s) that you want to logon to. Click **OK** to be returned to the **General** tab of the logon screen.
6. Click **OK**.



By default, the User ID is not authenticated via domain security service. The Windows user ID will be used for IBM MQ object authorization. To enable User ID authentication via Nastel Navigator workgroup server, start-up options must be changed. See [Appendix B](#) for additional details.

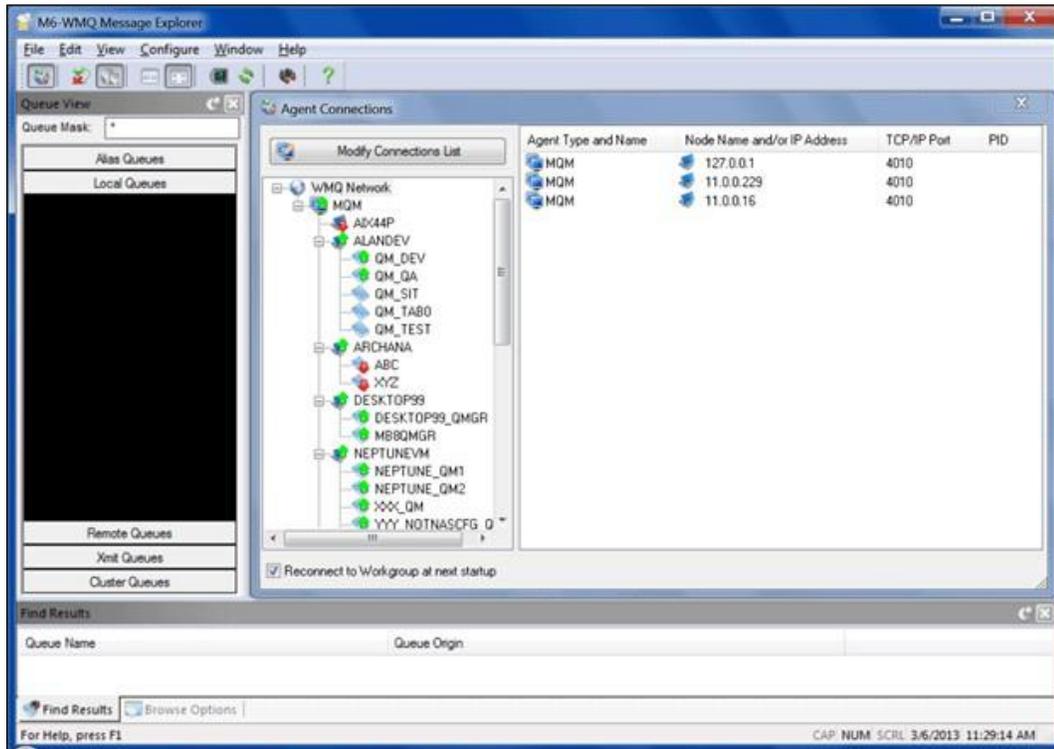


Figure 3-27. Message Explorer

7. To establish the queue manager to which to connect:

- a. On the tool bar, click the **Agent Connections** button  to open the *Agent Connections* dialogue box.

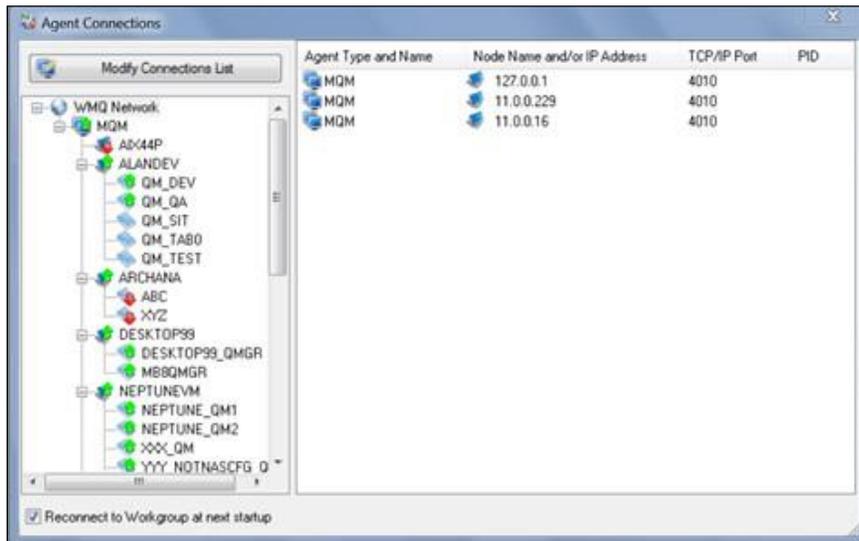


Figure 3-28. Agent Connections

- b. On the *Agent Connections* dialogue box, click **Modify Connections List** to open the *Configure Connections* dialog box.

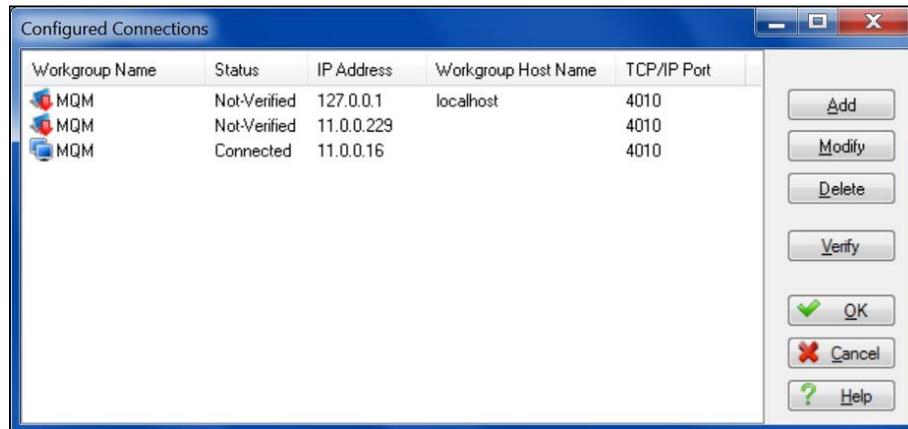


Figure 3-29. Configured Connections

- c. Select your workgroup and click **Verify** then **OK**.
- d. Expand the IBM MQ Network tree, the workgroup, and the desired agent node.
- e. Double-click the desired node. The dialog box closes and a panel with the available queues (alias, local, remote, xmit, cluster) are displayed.
- f. Either right-click a queue to select an available option or select an action from the tool bar.
- g. To work with another queue manager, click the **Agent Connections** button .
- h. Select a workgroup (perhaps a different one), log on (as done in step 5.c.i.), and select the desired node and queue manager.

3.5.3 Starting IBM MQ Workgroup Statistics

1. Click **Start, Programs**, click **AutoPilot-WMQ** to open Nastel Navigator program menu.
2. Click **Workgroup Statistics**.

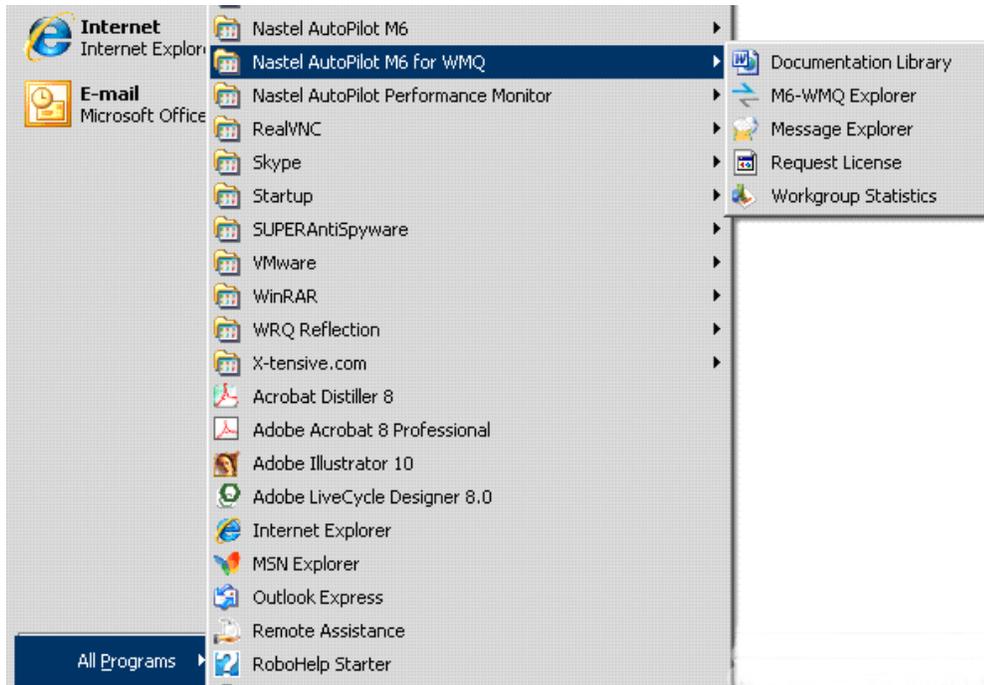


Figure 3-30. Windows Program Group

3. The *Workgroup Statistics* screen is displayed.

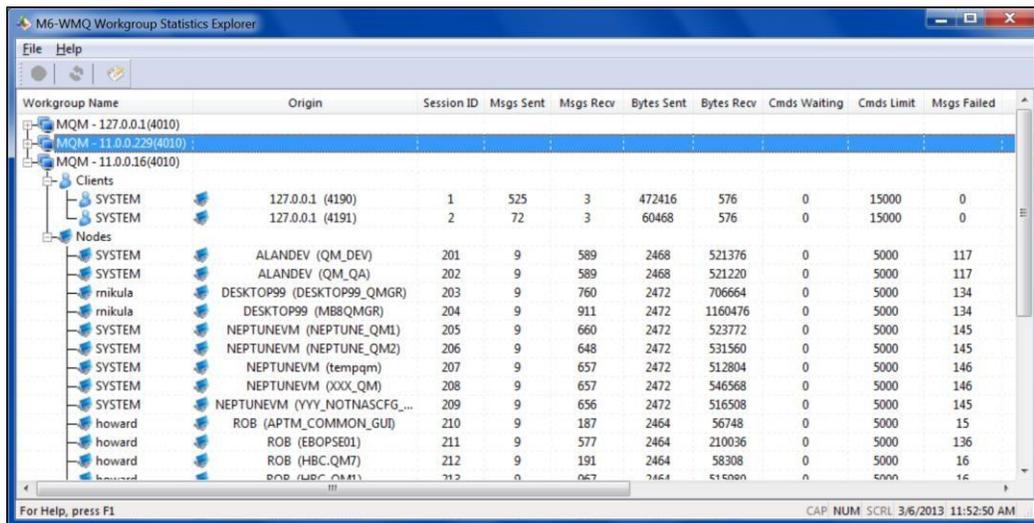


Figure 3-31. Workgroup Statistics

3.6 Uninstalling Nastel Navigator

This section provides the information required to uninstall Nastel Navigator.



NOTE

1. Prior to uninstalling Nastel Navigator or M6 Web, ensure all Nastel Navigator components have been stopped.
2. Uninstaller will only remove those files installed by Nastel Navigator. Files and directories that were User installed, created, or renamed (for example, experts, and business views) will require manual removal.
3. Uninstall procedures outlined in this section reflect Windows 2000; however, procedures for all operating systems/platforms are similar.

1. Copy any logs, business views, or other files you want to retain after you remove Nastel Navigator.
2. Run the install program. The *Welcome* screen is displayed. Click **Next**.

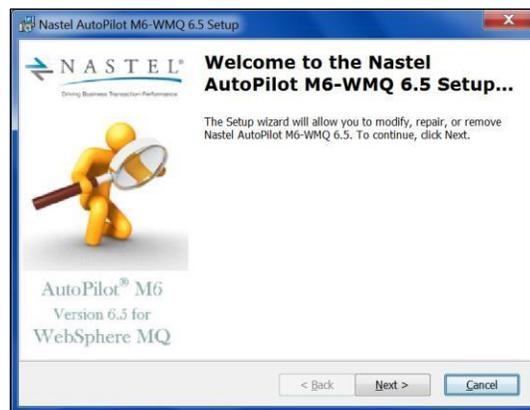


Figure 3-32. Removing Nastel Navigator

3. The *Program Maintenance* screen is displayed. Select **Remove** and click **Next**.



Figure 3-33. Remove Nastel Navigator

- The *Uninstalling* status screen is displayed. You can cancel the uninstall while this screen is displayed.

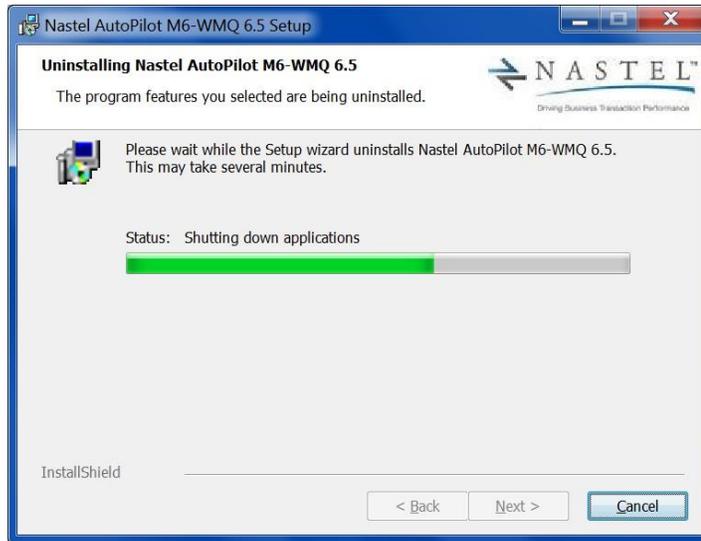


Figure 3-34. Uninstall Status

- When the uninstall is complete, a confirmation screen is displayed. Click **Finish**. All installed and system-generated files have been removed. User-defined files or logs may require manual removal.



Figure 3-35. Uninstall Complete

- You must restart your machine for the uninstall to take effect. The following screen is displayed. Click **Yes** to restart now or **No** to restart later.

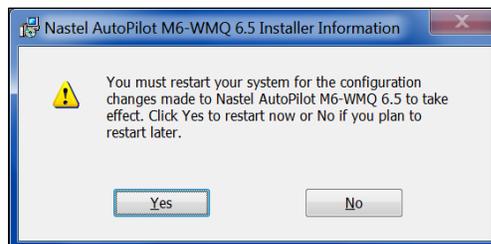


Figure 3-36. Restart Notice

Chapter 4: UNIX Installation

This chapter explains how to install Nasted Navigator in UNIX environment. Nasted Navigator can be installed in the following UNIX environment:

- AIX
- Solaris
- Linux
- Linux on zSeries s390x platform
- Linux on PowerPC

The installation package is distributed in the form of compressed archive file (tar.Z). The following table identifies package names by platform and edition:

Table 4-1. Compressed Archives by Platform	
OS Platform	Archive
AIX	Nav_Components_10.1.5_AIX_6.1_tar.Z
Solaris	M6WMQ_version_SUN_SOLARIS_osversion[.architecture].tar.Z
Linux	Nav_Components_10.1.5_Linux_Rhel5_x86_64.tar.gz

4.1 Nasted Navigator Components for UNIX

The Nasted Navigator installation for UNIX installs the following components on each machine:

- MQ Agent
- Message Server for IBM MQ
- WMQ Event Publish-Subscriber
- WMQ utilities

4.1.1 Custom Installation Components

The following table lists the Nasted Navigator components necessary for a custom installation.

Table 4-2. Custom Installation Components		
If Installing	Nasted Navigator Components	Binaries (Executables)
Agents only	IBM MQ Agent	<i>APWMQ_HOME/bin/nsqmq</i>
	Message Manager for IBM MQ	<i>APWMQ_HOME/bin/nsqmsg</i>
	Publish Subscriber for IBM MQ	<i>APWMQ_HOME/bin/nsqpub</i>
	Connection Manager for IBM MQ	<i>APWMQ_HOME/bin/nsqcm</i> (Agentless ONLY)

4.2 Unix Installation Requirements

Unix installation requirements are shown below:

Approximate Package Size:

1. Nastel Navigator – 50 MB; completely installed – 150 MB
2. Nastel Navigator Agents require about 20 MB of disk space.

4.2.1 Hardware Requirements

SunOS installation requires:

- Sun SPARC processor
- Solaris 8 or higher

AIX installation requires:

- AIX 5.3
- AIX 6.1
- AIX 7.1

LINUX installation requires:

- Red Hat V4 or higher on x86 platform

zLINUX installation requires:

- Red Hat V4 or higher on zSeries s390x platform

4.3 Installation

To install Nastel Navigator on UNIX system, obtain the compressed archive file (tar) for the corresponding system. The appropriate archive can be found on the Nastel Navigator distribution media in the Nastel Navigator directory.

4.3.1 Installation Considerations

Nastel Navigator could be installed on one or more systems. The administrator should consider the following:

MQ Agents must be installed on the IBM MQ nodes. Please make sure that IBM MQ installation requirements are satisfied. By default, IBM MQ Agent uses TCP/IP port 5010. To change it, specify `-s` command line option (`-s5010`).

Connection Manager can be run with the workgroup server or on its own servers. See [Connection Manager for TIBCO EMS](#), [Connection Manager for Kafka](#), [Connection Manager for IBM MQ](#), [Connection Manager for IBM IIB/ACE](#), and [Connection Manager for Solace](#) for more information.

4.3.2 Installing the Package

1. Download compressed archive file to the installation directory. For example: /opt.



Ensure you have a minimum of 200MB available in the installation directory. When installing IBM MQ Agents ensure your user account is part of the “mqm” group.

2. While in the installation directory, uncompress Nastel Navigator package. For example:

for UNIX: `uncompress M6WMQ_6.5.0_SUN_Solaris10_sparc.tar.Z`

for Linux: `gunzip Nav_Components_10.x.x_Linux_Rhel5_x86_64.tar.gz`

3. Untar the package. For example:

`tar -xvf Nav_Components_10.x.x_Linux_Rhel5_x86_64.tar.gz`

4. Verify installation by displaying directory structure. For example:

`ls -l`

4.4 Configuration

After Nastel Navigator is installed, follow the instructions given in the [Nastel Navigator Workgroup Server Expert V10 - Installation Guide](#) to configure the workgroup server. Refer to the next section, [Configuring the Agent on the IBM MQ Server](#), to configure the agent on an IBM MQ server.

4.4.1 Configuring Agent on the MQServer

4.4.1.1 Nastel Navigator Queues

Nastel Navigator local agents require the following queues to be created on IBM MQ queue managers. These names can be changed see [Appendix C](#) to change them.

MQ Agent:

Model Queue:

NASTEL.MODEL.QUEUE (default persistence, DEFPSIST, must be set to NO).

Local Queue:

NASTEL.EVENT.QUEUE (required only if Event Publish-Subscriber is not used).

Alias Queues:

3. SYSTEM.ADMIN.QMGR.EVENT (base queue name, TARGQ, must be set to NASTEL.EVENT.QUEUE)
4. SYSTEM.ADMIN.CHANNEL.EVENT (base queue name, TARGQ, must be set to NASTEL.EVENT.QUEUE)
5. SYSTEM.ADMIN.PERFM.EVENT (base queue name, TARGQ, must be set to NASTEL.EVENT.QUEUE)
6. SYSTEM.ADMIN.CONFIG.EVENT (base queue name, TARGQ, must be set to NASTEL.EVENT.QUEUE).
7. SYSTEM.ADMIN.COMMAND.EVENT (base queue name, TARGQ, must be set to

NASTEL.EVENT.QUEUE)

All above queues could be created by executing MQSC script, nastel.tst, located in the [Install_Dir]/mqsc directory. Example:

```
runmqsc QMGR < $APWMQ_HOME/mqsc/nastel.tst
```

Publish-Subscriber for IBM MQ

Local Queue:

NASTEL.PUBSUB.EVENT.QUEUE

Alias Queues:

8. SYSTEM.ADMIN.QMGR.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)
9. SYSTEM.ADMIN.CHANNEL.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)
10. SYSTEM.ADMIN.PERFM.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)
11. SYSTEM.ADMIN.CONFIG.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE).
12. SYSTEM.ADMIN.COMMAND.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)

These queues could be created by executing MQSC script, pubsub.tst, located in the [Install_Dir]/mqsc directory. Example:

```
runmqsc QMGR < $APWMQ_HOME/mqsc/pubsub.tst
```

Message Server for IBM MQ

Local Queues:

NASTEL.MMF.ADMIN.COMMAND.QUEUE
NASTEL.MMF.AUDIT.QUEUE (required only if MMF auditing is needed).

These queues could be created by executing MQSC script, mmf.tst, located in the [Install_Dir]/mqsc directory. Example:

```
runmqsc QMGR < $APWMQ_HOME/mqsc/mmf.tst
```

4.4.1.2 Configuring mqgroup.ini File

By default, mqgroup.ini file consists of the following:

```
Group::MQM=MQM  
MQM::Node=localhost  
MQM::Service=4010
```

If Workgroup name, TCP/IP address, or TCP/IP port changed, make sure the corresponding entries in the file are also changed. For example: If your Workgroup name is

MQM_IRELAND and Workgroup is running on the node with TCP/IP address 11.0.0.112, port – 4020. The configuration should look like:

```
Group::MQM_IRELAND=MQM_IRELAND
```

```
MQM_IRELAND::Node=11.0.0.112 (or fully qualified host name)
```

```
MQM_IRELAND::Service=4020
```

4.5 Starting and Stopping Components

This section contains instruction for starting and stopping Nastel Navigator components.

4.5.1 Starting and Stopping the IBM MQ Agent

This section contains instructions for starting and stopping the IBM MQ Agent.



The agent has a default auto-discovery interval of 10 minutes. This value may be changed using the Nastel Navigator once the agent is running.

4.5.1.1 Starting IBM MQ Agent

Start IBM MQ Agents as follows:

1. Ensure Workgroup configuration file (mqgroup.ini) is configured. It must have entries to resolve workgroup server to correct IP address and port:

```
Group::MQM=MQM
```

```
MQM::Node=localhost
```

```
MQM::Service=4010
```

2. Ensure the following environment variables are defined and exported:
 - a. *APWMQ_HOME* - must point to Nastel Navigator installation directory
 - b. *PATH* - must include path to Nastel Navigator bin directory

(opt/nastel/apwmq/bin or \$APWMQ_HOME/bin).

- c. *LD_LIBRARY_PATH* (Solaris and LINUX) – must include path to Nastel Navigator lib directory (opt/nastel/apwmq/lib or \$APWMQ_HOME/lib).
- d. *SHLIB_PATH* (HPUX) – must include path to Nastel Navigator lib directory

(opt/nastel/apwmq/lib or \$APWMQ_HOME/lib).

- e. *LIBPATH* (AIX) - must include path to Nastel Navigator lib directory

(opt/nastel/apwmq/lib or \$APWMQ_HOME/lib).

3. The following environment variables are optional:

- a. *APWMQ_MQS_HOME*– points to an alternate IBM MQ configuration file (mqs.ini) directory, mqs.ini or default to /var/mqm



A method is provided to allow managing a subset of the queue managers on your system, using an "alternate mqs.ini" file. See section 6.5, [Managing a Subset of Queue Managers](#).

- b. `APWMQ_ADP`– points to authorization audit directory or default to `$APWMQ_HOME`
- c. `APWMQ_LS_OPTIONS` – should consist of IBM MQ Agent startup options, space or semicolon delimited (`-mMQM; -u; -a`). These options overridden by options specified on the command line.

Possible problem with GUI command View Error Log

The agent can be configured to run with a subset of its queue managers by use of the environment variable `APWMQ_MQS_HOME`, pointing to some directory containing an abbreviated, altered copy of `mqs.ini`. However, the Nastel Navigator (and Explorer) GUI will not display output for the command View Error Log unless the alternate `mqs.ini` is configured correctly.

This requires, that in addition to copying the minimal queue manager lines, such as these,

QueueManager:

Name=HBC_QM8

you must also copy the lines that determine where the error log files are located, as in this example (for Windows, and same would apply for Unix agents):

AllQueueManagers:

```
#####
#* The path to the qmgrs directory, within which queue manager data *#
#* is stored *#
#####
DefaultPrefix=C:\Program Files (x86)\IBM\WebSphere MQ
```

LogDefaults:

LogDefaultPath=C:\Program Files (x86)\IBM\WebSphere MQ\log

QueueManager:

Name=HBC_QM8

Prefix=C:\Program Files (x86)\IBM\WebSphere MQ

Directory=HBC_QM8

InstallationName=Installation2

4. Ensure queue manager(s) configured for Nastel's monitoring. Next queues required for monitoring:

Model Queue:

NASTEL.MODEL.QUEUE (default persistence, DEFPSIST, must be set to NO).

Local Queue:

NASTEL.EVENT.QUEUE (required only if Event Publish-Subscriber is not used).

Alias Queues:

SYSTEM.ADMIN.QMGR.EVENT (base queue name, TARGQ, must be set to NASTEL.EVENT.QUEUE)

SYSTEM.ADMIN.CHANNEL.EVENT (base queue name, TARGQ, must be set to NASTEL.EVENT.QUEUE)

SYSTEM.ADMIN.PERFM.EVENT (base queue name, TARGQ, must be set to NASTEL.EVENT.QUEUE)

SYSTEM.ADMIN.CONFIG.EVENT (base queue name, TARGQ, must be set to NASTEL.EVENT.QUEUE).

All above queues could be created by executing MQSC script, `nastel.tst`, located in the `$APWMQ_HOME/mqsc` directory.

- Execute `nsqmq` program located in the `$APWMQ_HOME/bin` directory with following options:

```
./nsqmq -mMQM -u -a.
```

where MQM is the name of the workgroup with which IBM MQ Agent must register.

(See [Appendix B](#) for an explanation of these startup options.)

MQ Agent could also be started by executing `nsqmq_start.sh` script located in the `$APWMQ_HOME/sbin` directory. Note that this script must be updated to correspond to correct Nastel Navigator installation directory.

4.5.1.2 Stopping IBM MQ Agents

Stop IBM MQ Agent as follows:

- Ensure Workgroup configuration file (`mqgroup.ini`) is configured. It must have entries to resolve workgroup server to correct IP address and port:

```
Group::MQM=MQM
MQM::Node=localhost
MQM::Service=4010
```

- Ensure the following environment variables are defined and exported:
 - `APWMQ_HOME` - must point to Nastel Navigator installation directory.
 - `LD_LIBRARY_PATH` (Solaris and LINUX) - must include path to Nastel Navigator lib directory (`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - `SHLIB_PATH` (HPUX) – must include path to Nastel Navigator lib directory (`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - `LIBPATH` (AIX) - must include path to Nastel Navigator lib directory (`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).

3. Ensure, if authorization is used, user who executes script has authorization to execute EXCMD_SHUTDOWN_MANAGER command – 10110.
4. Execute nsqstop program located in the \$APWMQ_HOME/bin directory with the following options:

```
./nsqstop -mMQM -n<NodeName>
```

where MQM is the name of the workgroup and <NodeName> is identical to the IBM MQ Agent registration name in the Workgroup.

Below is a list of stop options for IBM MQ Agent:

Table 4-3. IBM MQ Agent Stop Options (using nsqstop utility)		
Option	Description	Default
-m<GroupName>	Workgroup name; required	None
-n<NodeName>	Node where IBM MQ Agent should stop	None
-i<IP Address>	TCP/IP address of node	127.0.0.1
-s<IP Port>	MQ Agent TCP/IP listening port number	5010

nsqshutdown

You can also stop IBM MQ Agents by executing the \$APWMQ_HOME/bin/nsqshutdown script with '-lm' or '-all' option. This shutdown procedure is not graceful, because it uses the command 'kill -9 pid' (SIGKILL signal). Version 6.5.1 or later removes any residual shared memory segments (4-byte size and 0 attached processes) left by the agent and publisher processes when they stop abnormally. Running out of shared memory segments could otherwise prevent the agent from starting.

You can also stop publish-subscribe by executing the \$APWMQ_HOME/bin/nsqshutdown script with '-ps' or '-all' option. This shutdown procedure is not graceful, because it uses the command 'kill -1 pid' (hangup signal SIGHUP). It removes any residual shared memory segments (4-byte size and 0 attached processes) left by the agent and publisher processes when they stop abnormally. Running out of shared memory segments could otherwise prevent publish-subscribe from starting.

nsqpub_stop.sh

Publish-subscribe could also be stopped by executing nsqpub_stop.sh script located in the \$APWMQ_HOME/sbin directory. Note that this script must be edited to correspond to the correct

Nastel Navigator installation directory

4.5.2 Starting and Stopping the Message Server for IBM MQ

This section contains instructions for starting and stopping the Message Server for IBM MQ.

4.5.2.1 Starting the Message Server for IBM MQ

Start the message server as follows:

1. Ensure the following environment variables defined and exported:
 - a. `APWMQ_HOME` - must point to Nastel Navigator home directory
 - b. `PATH` - must include path to Nastel Navigator bin directory
(`opt/nastel/apwmq/bin` or `$APWMQ_HOME/bin`).
 - c. `LD_LIBRARY_PATH` (Solaris and LINUX) - must include path to Nastel Navigator lib directory (`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - d. `SHLIB_PATH` (HPUX) - must include path to Nastel Navigator lib directory (`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - e. `LIBPATH` (AIX) - must include path to Nastel Navigator lib directory
(`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
2. The following environment variable is optional:
`APWMQ_MS_OPTIONS` - should consist of Message Server startup options, space or semicolon delimited (`-mQMGR1;-r1440;-g300`). These options are superseded by the option specified on the command line.
3. Ensure queue manager(s) configured for Nastel's message management. The next queues are required for Message Management Facility (MMF).

Local Queues:

```
NASTEL.MMF.ADMIN.COMMAND.QUEUE
```

```
NASTEL.MMF.AUDIT.QUEUE (required only if MMF auditing is needed).
```

These queues can be created by executing MQSC script, `mmf.tst`, located in `$APWMQ_HOME/mqsc` directory.

4. Execute `nsqmsg` program located in the `$APWMQ_HOME/bin` directory with the following options:

```
./nsqmsg -m<YOUR_QMGR_NAME>
```

 where `<YOUR_QMGR_NAME>` is the name of the queue manager where MMF is required.

(See [Appendix B](#) for an explanation of these startup options.)

Message Server could also be started by executing `nsqmsg_start.sh` script located in the `$APWMQ_HOME/sbin` directory. Note that this script must be updated to correspond to the correct Nastel Navigator installation directory and local queue managers.

4.5.2.2 Stopping the Message Server for IBM MQ

To stop the Message Server for IBM MQ:

1. Ensure the following environment variables are defined and exported:
 - a. `APWMQ_HOME` - must point to Nastel Navigator home directory
 - b. `PATH` - must include path to Nastel Navigator bin directory
(`opt/nastel/apwmq/bin` or `$APWMQ_HOME/bin`).
 - c. `LD_LIBRARY_PATH` (Solaris and LINUX) - must include path to Nastel Navigator lib directory
(`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - d. `SHLIB_PATH` (HPUX) - must include path to Nastel Navigator lib directory
(`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - e. `LIBPATH` (AIX) - must include path to Nastel Navigator lib directory
(`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
2. Ensure the queue manager is active and available
3. Execute `endmcs` program located in the `$APWMQ_HOME/bin` directory with next options:


```
./endmcs <YOUR_QMGR_NAME>
```

where `<YOUR_QMGR_NAME>` is name of queue manager where message server is required to be stopped.

Below is a list of stop options for Message Server:

Table 4-4. Message Server Stop Options (using <code>endmcs</code> utility)		
Option	Description	Default
<code><QmgrName></code>	Queue manager name (required)	*
<code><CmdQName></code>	Command queue name	<code>NASTEL.MMF.ADMIN.COMMAND.QUEUE</code>

It is also possible to stop the message server for IBM MQ by executing `$APWMQ_HOME/bin/nsqshutdown` script with `'-ms'` or `'-all'` option. This shutdown procedure is not graceful and requires certain shared memory cleanup after use; therefore, it must be used with caution. The Message Server for IBM MQ could also be stopped by executing `nsqmsg_stop.sh` script located in the `$APWMQ_HOME/sbin` directory. Note that this script must be updated to correspond to correct Nastel Navigator installation directory and local queue managers.

nsqshutdown

You can also stop Nastel Navigator workgroup servers by executing `$APWMQ_HOME/bin/nsqshutdown` script with `'-gm'` or `'-all'` option. This shutdown procedure is not graceful, because it uses the command `'kill -1 pid'` (hangup signal `SIGHUP`). Version 6.5.1 or later removes any residual shared memory segments (4-byte size and 0 attached processes) left by the agent and publisher processes when they stop abnormally.

nsqmsg_stop.sh

The message server could also be stopped by executing `nsqmsg_stop.sh` script located in the `$APWMQ_HOME/sbin` directory. Note that this script must be edited to correspond to the correct

Nastel Navigator installation directory.

4.5.3 Starting and Stopping the Publish-Subscriber for IBM MQ

This section contains instructions for starting and stopping the Nastel Navigator publish-subscriber. These utilities must be started by an `mqm` user.

4.5.3.1 Starting IBM MQ Publish-Subscriber

Start IBM MQ publish-subscriber as follows:

1. Ensure the following environment variables defined and exported:
 - a. `APWMQ_HOME` – must point to Nastel Navigator home directory.
 - b. `PATH` – must include path to Nastel Navigator bin directory
(`opt/nastel/apwmq/bin` or `$APWMQ_HOME/bin`).
 - c. `LD_LIBRARY_PATH` (Solaris and LINUX) – must include path to Nastel Navigator lib directory (`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - d. `SHLIB_PATH` (HPUX) – must include path to Nastel Navigator lib directory (`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - e. `LIBPATH` (AIX) – must include path to Nastel Navigator lib directory
(`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
2. The following environment variables are optional:
 - a. `APWMQ_MQS_HOME`: Points to an alternate IBM MQ configuration file (`mqs.ini`) directory, `mqs.ini` or default to `/var/mqm`.



A method is provided to allow managing a subset of the queue managers on your system, using an "alternate `mqs.ini`" file. See section 6.5, [Managing a Subset of Queue Managers](#).

- b. `APWMQ_PS_OPTIONS`: Should consist of IBM MQ Publish-Subscriber startup options, space or semicolon delimited (`-fpermsub.ini;-w60`). These options are overridden by options specified on the command line.
3. Ensure queue manager(s) configured for Nastel's Publish-Subscriber. The following queues are required for Publish-Subscriber:

Local Queue:

`NAstel.PUBSUB.EVENT.QUEUE`

Alias Queues:

SYSTEM.ADMIN.QMGR.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)

SYSTEM.ADMIN.CHANNEL.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)

SYSTEM.ADMIN.PERFM.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)

SYSTEM.ADMIN.CONFIG.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)

SYSTEM.ADMIN.COMMAND.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)

These queues could be created by executing MQSC script, `pubsub.tst`, located in the `[Install_Dir]/mqsc` directory.

4. Execute `nsqpub` program located in the `$APWMQ_HOME/bin` directory with the following option: `./nsqpub`
(See [Appendix B](#) for an explanation of the startup options.)

The Publish-Subscriber can also be started by executing `nsqpub_start.sh` script located in the `$APWMQ_HOME/sbin` directory. Note that this script must be updated to correspond to relevant Nastel Navigator installation directory.

4.5.3.2 Stopping IBM MQ Publish-Subscriber

Stop IBM MQ publish-subscribers as follows:

1. Ensure the following environment variables are defined and exported:
 - a. `APWMQ_HOME`: Must point to Nastel Navigator home directory
 - b. `PATH` - must include path to Nastel Navigator bin directory
(`opt/nastel/apwmq/bin` or `$APWMQ_HOME/bin`).
 - c. `LD_LIBRARY_PATH` (Solaris and LINUX) - must include path to Nastel Navigator lib directory (`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - d. `SHLIB_PATH` (HPUX) - must include path to Nastel Navigator lib directory
(`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
 - e. `LIBPATH` (AIX) - must include path to Nastel Navigator lib directory
(`opt/nastel/apwmq/lib` or `$APWMQ_HOME/lib`).
2. Execute `pubend` program located in the `$APWMQ_HOME/bin` directory with the following option: `./pubend`.

Below is a list of stop options for Publish-Subscriber for IBM MQ:

Table 4-5. Publish-Subscriber Stop Options (using pubend utility)

Option	Description	Default
<code>-r<UDP Retry></code>	UDP message retry	5
<code>-u</code>	Reuse UDP address	No
<code>-l<UDP Port></code>	Set shutdown utility UDP port	6000
<code>-s<UDP Port></code>	Set Publish-Subscriber UDP port	5500
<code>-h<Host name></code>	Set Publish-Subscriber host name	localhost
<code>-i<Host IP address></code>	Set Publish-Subscriber IP address	127.0.0.1
<code>-t</code>	Enable message trace	No
<code>-?</code>	Displays usage information.	

You can also stop the Nastel Navigator publish-subscriber by executing the following script:

```
$APWMQ_HOME/bin/nsqshutdown with '-ps' or '-all' option.
```

OR

```
nsqpub_stop.sh located in the $APWMQ_HOME/sbin directory.
```

nsqshutdown

You can also stop publish-subscribe by executing the `$APWMQ_HOME/bin/nsqshutdown` script with `'-ps'` or `'-all'` option. This shutdown procedure is not graceful, because it uses the command `'kill -1 pid'` (hangup signal `SIGHUP`). It removes any residual shared memory segments (4-byte size and 0 attached processes) left by the agent and publisher processes when they stop abnormally. Running out of shared memory segments could otherwise prevent publish-subscribe from starting.

nsqpub_stop.sh

Publish-subscribe could also be stopped by executing `nsqpub_stop.sh` script located in the `$APWMQ_HOME/sbin` directory. Note that this script must be edited to correspond to the correct

Nastel Navigator installation directory.

Chapter 5: z/OS Installation

This chapter defines the installation requirements and procedures for Nistel Navigator on Z Series Operating System (z/OS).

5.1 Nistel Navigator Components for z/OS

The Nistel Navigator installation for z/OS installs the following components on each machine:

- MQ Agent
- MMF Server
- MQ Event Publisher

5.2 System Requirements

An Nistel Navigator installation for z/OS requires:

- z/OS V2.1 or higher
- IBM MQ V7 or higher
- TCP/IP Services
- Runtime support for the C programming language (Language Environment level 1.7 or higher)



On z/OS, Nistel Navigator utilizes IBM Multitasking Facility (MTF). MTF only supports load modules that are stored in a standard Partition Data Set (PDS). MTF does not support load modules stored in a PDS Extended (PDSE). When installing Nistel Navigator, ensure that parallel load modules for IBM MQ Agent (`NSQCSUB`), MQ Event Publisher (`NSQPSUB`), and Message Server (`NSQMSRV`) are stored in a standard PDS.

5.3 Limitations of Nistel Navigator on z/OS

Nistel Navigator has certain limitations when running on z/OS.

5.3.1 Limitations on IBM MQ Agents

On z/OS, IBM MQ Agents cannot be used to:

- Start, stop, create, or delete queue managers
- Start or stop Job Control Language (JCL) jobs
- Execute the Nistel Navigator commands `setmqaut` (set MQ authority) or `dspmqaut` (display MQ authority)



Although Nistel Navigator for z/OS cannot execute `setmqaut` and `dspmqaut` commands, it can issue full range of MQSC commands via Nistel's Nistel Navigator application. Nistel Navigator includes support for z/OS-specific commands, including `ALTER SECURITY`, `DISPLAY SECURITY`, `REFRESH SECURITY`, `VERIFY SECURITY`, `DISPLAY STGCLASS`, and so on.

5.3.2 Limitations on MMF Server

Single instance of MMF Server cannot manage multiple queue managers on z/OS.

5.4 Upgrading Nastel Navigator on z/OS



This section does not apply for first-time installation of Nastel Navigator.

When migrating from earlier versions of Nastel Navigator to a more advanced one, it is recommended to save existing files. This enables reapplication of original configuration to the new Nastel Navigator installation or, if needed, reverting to the previous version of Nastel Navigator without loss of configuration settings.

To upgrade Nastel Navigator on z/OS:

1. Save your original Nastel Navigator under a new (backup) name and rename the defined started task procedures.
2. Install a new version of Nastel Navigator as described in the following section.
3. Review the README file for changes which may be required for the PARM strings or to `dd:MQSINI`, `dd:MQGROUP`, `dd:PERMITS`, `dd:SUBSCRIB` or `dd:SEVERITY`.

5.5 Installing Nastel Navigator

Nastel Navigator is installed from installation media or FTP site where installation files are organized by platform.

5.5.1 The z/OS Installation Files

On the CD, navigate to the z/OS folder. Then navigate to the appropriate folder for your version of z/OS and then Nastel Navigator. In that folder you will find a zip file, which in turn contains two files named `load.xmi` and `jcl.xmi`. The xmi files are binary images of data sets created by the TSO TRANSMIT command and contain the data sets `M6WMQ.LOAD` and `M6WMQ.JCL` respectively. The first data set, `M6WMQ.LOAD`, contains the AutoPilot M6 load modules. The contents of `M6WMQ.LOAD` are listed in table 5-1.

Table 5-1. Contents of the M6WMQ.LOAD Data Set

Member	Description
NSQMQ	MQ Agent executable module.
NSQCSUB	MQ Agent run-time parallel load module.
NSQMSG	MMF server executable module.
NSQMSRV	MMF server's run-time parallel load module.
NSQPUB	MQ Event Publisher executable module.
NSQPSUB	MQ Event Publisher server's run-time parallel load module.
ENDMCS	Executable module used to stop the MMF server.

Table 5-1. Contents of the M6WMQ.LOAD Data Set

Member	Description
PUBEND	Executable module used to stop the IBM MQ Agent or the MQ Event Publisher.
NSQMMFCL	Executable module used to run command line tools. Give access to MMF from command line.
NSQMASC	Executable module used to run command line tools. Give access to queue managers from command line.
NSQMSGF	Executable module used to run command line tools dump messages from queue to file and file to queue.
NSQSTOP	Executable module to stop the agent.
M6\$xyz	Product version identifier. Example: M6\$612 means 6.1.2.
NSQZAS	Accounting and statistics collector program.
NSQXU29	SMF exit program to trigger SMF dump and NSQZAS execution.
MQCDUMP	IBM utility to print SMF type 115/116 accounting and statistics records
AMQSMON	IBM utility to browse/print accounting/statistics messages.

The second data set, `M6WMQ.JCL`, contains sample JCL used to run or stop Nastel Navigator components. `M6WMQ.JCL` also contains Nastel Navigator's default queue definitions and examples of Nastel Navigator configuration files. The contents of `M6WMQ.JCL` are listed in table 5-2.

Table 5-2. Contents of the M6WMQ.JCL Data Set

Member	Description
NSQMQ	JCL for running IBM MQ Agent.
NSQMSG	JCL for running MMF server.
NSQPUB	JCL for running MQ Event Publisher.
LMSTOP	JCL for stopping IBM MQ Agent.
MSSTOP	JCL for stopping MMF server.
PBSTOP	JCL for stopping MQ Event Publisher.
MQMGM	List queue definition required by Nastel Navigator.
MQSINI	Example of Nastel Navigator configuration file that defines all the queue managers.
MQGROUP	Example of Nastel Navigator configuration file that lists Nastel Navigator groups.
PERMITS	File which specifies the permits.
SUBSCRIB	Example of the subscriber permanent configuration file for the MQ Event Publisher.

Table 5-2. Contents of the M6WMQ.JCL Data Set	
SEVERITY	Example of the severity filter configuration, needed when the -W option is used with MQ Event Publisher.
MQMGMREM	Example to remove list queue definition required by Nastel Navigator.
NSQCNFG	Example to run the configuration command using CSQUTIL.
NSQLOAD	Example to restore datasets.
NSQTEMP1	Example to allocate the jcl.xmi for transfer datasets.
NSQTEMP2	Example to allocate the load.xmi for transfer datasets.
README	Release notes and change history.
AMQSMON	JCL for running AMQSMON
M6\$xyz	Product version identifier. Example: M6\$612 means 6.1.2.
MQMZAS	JCL to configure queues used by NSQZAS.
NSQSMF	JCL called by NSQXU29 assembler exit program.
NSQSMFDP	JCL to use the SMF dump utility to extract MQ records into a sequential file.
NSQZAS	JCL to dump SMF data set and start NSQZAS accounting/statistics collector; used for testing.
RUNCDMP	JCL for running MQCDUMP.
RUNZAS	JCL to run the NSQZAS accounting/statistics collector to process a given SMF dump file. Used for stand-alone testing of NSQZAS. This JCL is a variant of NSQZAS.
SMFCLEAN	JCL to invoke the SMF dump utility to clear all trace records from an SMF data set.
SMFGDG	JCL to define GDS (Generation Data Set) for MQ SMF 115/116 data.

5.5.2 Transferring and Restoring Data Sets

When installing Nastel Navigator on z/OS, transfer and restore the data sets from the CD to the mainframe.

Note that in this section:

- Userid: Represents the TSO user ID
- jcl.xmi and load.xmi: Represent the names chosen for temporary transmit/receive data sets
- yourjcllibrary.jcl and yourloadlibrary.load: Represent the names chosen for restored data sets.

Minimum dataset allocations are summarized in the following table:

Table 5-3. Minimum Dataset Allocations											
	DSORG	RECFM	LRECL	BLKSIZE	Member	DirBlks	Blocks	3390 Trks	3390 Cyls	3380 Trks	3380 Cyls

jcl.xmi	PS	FB	80	3120	N/A	N/A	70	1	1	1	1
load.xmi	PS	FB	80	3120	N/A	N/A	2640	105	7	106	8
M6WMQ.JCL	PO	FB	80	27920	34	3	18	30	2	36	3
M6WMQ.LOAD	PO	U	0	27998	17	2	410	205	16	248	16

The current release of IBM's Multi-Tasking Facility (MTF) only supports load modules stored in a PDS and Nastel Navigator for z/OS using MTF to manage its subtasks. If you are planning to copy the load modules in M6WMQ.LOAD to another runtime dataset, please make sure that the target dataset is a PDS and not a PDSE(LIBRARY).

xmi files are binary images of datasets created by the TSO TRANSMIT command. When transferred from a PC to the mainframe that translation must be done in binary mode, with no carriage return or character set translation in effect. The TSO RECEIVE command will fail if the upload is not binary.

The TSO RECEIVE command will also fail if the intermediate dataset is not RECFM=FB and LRECL=80. Failure to pre-allocate these datasets will result in either RECFM=VB or RECFM=U, causing the RECEIVE to fail.

With z/OS and TSO, rather than code up these JCL samples and submit to batch processing, it could be more convenient to use ISPF 3.2, or the TSO ALLOC command to pre-allocate the datasets, unzip, and binary transfer the xmi files to the mainframe, then enter the RECEIVE command from the TSO command prompt for the two xmi files:

```
receive indsn('jcl.xmi')
dsn('yourjcllibrary.jcl')
receive indsn('load.xmi')
dsn('yourloadlibrary.load')
```

Enter the `dsn()` option in response to the prompt from RECEIVE for parameters.

All RECFM=FB datasets may be specified with larger block sizes for efficiency or to conform to local requirements. The single RECFM=U library may be specified with a larger block size, but this will not have any benefit unless the load modules are reformatted.

Transfer Data Sets:

1. On the mainframe, allocate the `jcl.xmi` dataset, by running the following job, either using the TSO ALLOC command, or using ISPF 3.2:

```
//NSQTEMP1 JOB (xxxx,xxxx), 'xxxxxxxx', CLASS=A, MSGCLASS=A,
// MSGLEVEL=(1,1), NOTIFY=userid
//S1 EXEC PGM=IEFBR14
//DD1 DD DSN=jcl.xmi, DISP=(,CATLG),
// UNIT=SYSALLDA, SPACE=(3120, (10,5)),
//* UNIT=3390, SPACE=(TRK, (1,1)),
//* UNIT=3380, SPACE=(TRK, (1,1)),
// DCB=(DSORG=PS, BLKSIZE=3120, LRECL=80, RECFM=FB)
```

This intermediate dataset will not be needed again once the JCL library is restored from it.

2. On your workstation, mount the distribution media and navigate to the Nastel Navigator directory and then to the z/OS subdirectory. Refer to the README.TXT file for guidelines as to which distribution library is appropriate for your z/OS and IBM MQ product versions. Unzip the contents of this file to a temporary location; the names of the unzipped files will be *jcl.xmi* and *load.xmi*. (These are the files to be uploaded to the mainframe in the subsequent steps.)
3. On your workstation, launch an FTP utility that will let you transfer the data sets from the CD to the mainframe. This may be the FTP command or the file transfer facility of your 3270 terminal emulator program. Using FTP will usually be much quicker than using the 3270 emulator.
4. Using the FTP utility, transfer the JCL data set from the PC to the mainframe. If using TCP/IP FTP, use the `put` subcommand. From the PC, the local file is *M6WMQ.JCL*; the remote file is *jcl.xmi*. Specify that the transfer type is binary.
5. On the mainframe, allocate the *load.xmi* dataset by the following job, either using the TSO ALLOC command, or using ISPF 3.2:

```
//NSQTEMP2 JOB (xxxx,xxxx), 'xxxxxxxx', CLASS=A, MSGCLASS=A,  
//          MSGLEVEL=(1,1), NOTIFY=userid  
//S2 EXEC PGM=IEFBR14  
//DD2      DDDSDNAME=load.xmi, DISP=(,CATLG),  
// UNIT=SYSALLDA, SPACE=(3120, (1500, 500)),  
//* UNIT=3390, SPACE=(TRK, (150, 50)),  
//* UNIT=3380, SPACE=(TRK, (200, 50)),  
// DCB=(DSORG=PS, BLKSIZE=3120, LRECL=80, RECFM=FB)
```

This is an intermediate dataset that will not be needed again once the LOAD library is restored from it.

6. Using the FTP utility, transfer the LOAD data set from the PC to the mainframe. If you are using TCP/IP FTP, you will use the `put` subcommand. From the PC, the local file is *M6WMQ.LOAD*; the remote file is *load.xmi*. Specify that the transfer type is binary.

Restore Data Sets:

1. On the mainframe, create a sequential data set with the name:
userid.APWMQ.RECEIVE.CLIST

Enter the following CLIST into the data set:

```

PROC 0
CONTROL MSG LIST PROMPT NOFLUSH
ERROR RETURN
RECEIVE INDSNAME('jcl.xmi') LOGDATASET('NULLFILE')
DATA PROMPT
DSN('yourjcllibrary.jcl')
ENDDATA
RECEIVE INDSNAME('load.xmi') LOGDATASET('NULLFILE')
DATA PROMPT
DSN('yourloadlibrary.load')
ENDDATA

```

2. Run the following job, modifying the allocation parameters as appropriate for the installation requirements:

```

//NSQLOAD      JOB (xxxx,xxxx), 'xxxxxx', CLASS=A, MSGCLASS=A,
//             MSGLEVEL=(1,1), NOTIFY=userid
//PREALLOC     EXEC PGM=IEFBR14
//JCL          DD DSNAME=yourjcllibrary.jcl, DISP=(,CATLG),
//             UNIT=SYSALLDA, SPACE=(3120, (10, 5, 5)),
//*           UNIT=3390, SPACE=(TRK, (5, 2, 5)),
//*           UNIT=3380, SPACE=(TRK, (6, 2, 5)),
//             DCB=(BLKSIZE=3120, LRECL=80, RECFM=FB)
//LOAD         DD DSNAME=yourloadlibrary.load, DISP=(,CATLG),
//             UNIT=SYSALLDA, SPACE=(6144, (1000, 200, 5)),
//*           UNIT=3390, SPACE=(TRK, (150, 50, 5)),
//*           UNIT=3380, SPACE=(TRK, (200, 50, 5)),
//             DCB=(BLKSIZE=6144, RECFM=U)
//RECEIVE      EXEC PGM=IKJEFT01, DYNAMNBR=128, REGION=0M
//SYSTSPRT     DD SYSOUT=*
//SYSTSIN      DD *
PROFILE PREFIX(userid)
EXEC APWMQ.RECEIVE
/*
//

```

When NSQLOAD job has successfully completed, *jcl.xmi*, *load.xmi* and *userid.APWMQ.RECEIVE.CLIST* are no longer needed and may be deleted.

5.6 Configuration Requirements for z/OS

On z/OS, Nastel Navigator components must have certain queues and aliases defined, as described below. Note that these components must have sufficient access authority to retrieve messages from, and place messages onto queues listed in this section.

The MQSC input commands in this section are in *yourjcllibrary.jcl* library in member MQMGM. Review MQMGM and this section and make any required changes before

submitting it for execution. Changes will consist of uncommenting the MMF and/or Publication Subscriber configuration commands for those components that will be run.

Run the configuration commands using CSQUTIL as in the following example:

```
//NSQCNFG JOB (xxxx,xxxx), 'xxxxxxxx', CLASS=A, MSGCLASS=A,  
//          MSGLEVEL=(1,1), NOTIFY=userid  
//MQSC EXEC PGM=CSQUTIL, PARM='queue manager'  
//STEPLIB DD DSN=thlqual.SCSQANLE, DISP=SHR  
//          DD DSN=thlqual.SCSQAUTH, DISP=SHR  
//SYSPRINT DD SYSOUT=*  
//MQSCIN DD DSN=yourjcllibrary.jcl(MQMGM), DISP=SHR  
//SYSINDD *  
COMMAND DDNAME(MQSCIN)  
/*  
//
```

Normally this will work without problem, other than error messages for attempts to delete queues that have not yet been defined. However, if some tasks (such as another IBM MQ monitor) have the event queues open, the DELETE commands for those queues will fail. You may not be able to identify or stop the task holding the queues open, in which case it may be necessary to link the MQMGM member to the CSQINP2 dd statement of your queue manager and restart said queue manager to successfully configure your queue manager for Nastel Navigator.

The configuration in this section will need to be done just once for every queue manager. It is not normally necessary to remove the changes even when Nastel Navigator is not in use; the queues and aliases defined will not harm your IBM MQ system. However, if you do wish to remove these changes, you may delete the queues and aliases created in this section and then recreate the event queues. Member MQMGMREM may be used for this purpose.

5.6.1 External Security Manager Requirements for Nastel Navigator (z/OS)

The security structure for IBM-MQ can be extremely complex depending on the various levels and distribution of authority. Below is a short summary of how authorization works from the Nastel Navigator perspective and a summary of which resources Nastel Navigator requires access to. The following high-level explanation of security checking assumes the external security manager is RACF, CA-ACF2, or CA-Top Secret.

Nastel Navigator connects to queue managers, accesses and updates queues, and issues administrative commands. IBM-MQ validates the authority of Nastel Navigator to perform these operations via a System Authorization Facility (SAF) call. SAF then issues a call to the underlying External Security Manager (ESM), which returns a suggested action whether to allow the operation to continue. The suggestion is based on a lookup of the Resource Name and Class Descriptor combination. That suggestion is returned to IBM-MQ in the

form of a return and reason code; IBM-MQ enforces the suggestion as appropriate. There are scenarios where IBM-MQ will issue more than one security call as it processes a request. For example, a request to alter a queue definition will require Connection authority (Resource Class = MQCONN) to connect to the queue manager, Command authority (Resource Class = MQCMDSD) to issue the alter command, and Administrative authority (Resource Class= MQADMIN or MXADMIN) to the queue name.

Nastel Navigator provides a UserID and for actionable requests, optionally, an Alternate UserID with each request. These UserIDs need appropriate authority. It is suggested that Nastel Navigator be granted a high level of authority via a UserID which cannot be logged on to and is only for Nastel Navigator use. Individual administrators and/or users should be granted authority within Nastel Navigator to view and alter the various objects as appropriate.

Table 5-4 summarizes the security requirements for Nastel Navigator. “*qmgr*” is the name of the queue manager or queue sharing group; it could be generically specified. “*queuename*” refers to the fully or generically qualified queue name. “*objecttype*” refers to any value defined by IBM-MQ as valid object type. “*objectname*” refers to the fully or generically qualified object name. In the example below, “***” indicates any number of qualifiers of any length limited by the maximums supported by the ESM and IBM-MQ. **It is highly recommended that “*queuename*” and “*objecttype.objectname*” be specified as “***” to give Nastel Navigator full authority and limit the individual user’s authority via the Nastel Navigator authorization mechanism.** The Class Descriptor below is for an uppercase resource name. As appropriate and when permitted, the equivalent mixed case class or group class can be used. Refer to the documentation for the ESM you are using.

Table 5-4. Security Requirements for Nastel Navigator				
Security Type	Resource Definition	Class Descriptors	Access	Notes
Connection Security	<i>qmgr</i> .SYSTEM.DEF.SVRCONN	MQCONN	READ	1
	<i>qmgr</i> .BATCH	MQCONN	READ	1,2
Queue Security	<i>qmgr</i> .NASTEL.**	MQQUEUE	UPDATE	3
	<i>qmgr.queuename**</i>	MQQUEUE	ALTER	4,8,10
Context Security	<i>qmgr</i> .CONTEXT.**	MQADMIN	CONTROL	4,8,10
Administrative Authority	<i>qmgr.objecttype.objectname**</i>	MQADMIN	ALTER	7,10
Command Security	<i>qmgr.**</i>	MQCMDSD	ALTER	5,10
	<i>qmgr.**</i>	MQADMIN	ALTER	9,10
Alternate User Security	<i>qmgr</i> .ALTERNATE.USER.**	MQADMIN	UPDATE	6,10

Notes:

1. Allows the connection to complete in an agentless environment.
2. Allows the Nastel z/OS resident Agent to connect to MQ.
3. Allows reading, writing, and deleting of data on the queue.
4. Allow reading, writing, and deleting of “production queues” which M6 WMQ may need to manipulate messages as directed by queue administration operations staff. Specification of “qmgr.**” will allow access to any queue on the queue manager and is preferred.
5. Allow Nastel Navigator to issue commands to the Queue Manager.
6. Allows the primary user to specify an alternate user.
7. Combined with Command Security, defines the level of authority the user has to administer a given object by Object Type and Object Name.
8. Consider system queues such as the Dead Letter Queue, command queues, model and dynamic queues, alias queues, transmit queues and receive queues.
9. In conjunction with command authority required to issue a command, M6 WMQ needs administrative authority to the target object including but not limited to defining and/or deleting it. (See the full definition of Administrative Authority in the table entry and description above. Not all commands (e.g., display) require administrative authority.)
10. It is suggested that Nastel Navigator be given high level of authority as described above. Specifying “qmgr.**” as the resource name will define the authority to all objects and commands. The example above shows how the resource name is configured. More granular control can be accomplished by specifying a more specific resource name and granting access to those resources on a user or group basis either by primary or alternate UserID. The means of specifying wildcard characters in the resource name is defined by the ESM in use.

Table 5-5 shows some examples.

Table 5-5. ESM Resource Definition Examples			
Resource	Class	Access	Result
qmgr.**	MQCMDS	ALTER	Allow any command
qmgr.ALTER.**	MQCMDS	ALTER	Allow the ALTER command
qmgr.ALTER.QLOCAL.PAY**	MQCMDS	ALTER	Allow the ALTER against local queues where the queue name starts with “PAY”
qmgr.DISPLAY.**	MQCMDS	READ	Allow any display command
qmgr.DISPLAY.QLOCAL.**	MQCMDS	READ	Only allow display of local queues

Table 5-5. ESM Resource Definition Examples

Resource	Class	Access	Result
qmgr.DISPLAY.QLOCAL.PAY**	MQCMDS	READ	Only allow display of local queue names that start with "PAY"
qmgr.DEFINE.**	MQCMDS	ALTER	Allow any define command
qmgr.**	MQADMIN	ALTER	Allow any full administration any object type
qmgr.QUEUE.**	MQADMIN	ALTER	Allow any full administration of object type "QUEUE"
qmgr.CHANNEL.BANK1**	MQADMIN	ALTER	Allow any full administration of object type "CHANNEL" where the channel name starts with the string "BANK1"
qmgr.**	MXQUEUE	ALTER	Allow full access to the contents of any queue and the ability to change, add, or delete any message on the queue.
qmgr.PAY**	MXQUEUE	ALTER	Allow full access to the contents of any queue whose name starts with "PAY" and the ability to change, add, or delete any message on the queue.
qmgr.ALTERNATE.USER.**	MQADMIN	UPDATE	Allow the specification of any UserID in the alternate user field
qmgr.ALTERNATE.USER.PAY**	MQADMIN	UPDATE	Allow the specification of any UserID starting with "PAY" in the alternate user field
qmgr.ALTERNATE.USER.JOHN	MQADMIN	UPDATE	Only allow the specification of UserID "JOHN" in the alternate user field

5.6.2 IBM MQ Agent

Each z/OS node must have at least one IBM MQ Agent. Each IBM MQ Agent can control multiple IBM MQ queue managers.

All system event queues must be defined as alias queues. The base queue to which all system event queues should point is `NASTEL.EVENT.QUEUE`.

For each queue manager, a queue named `SYSTEM.ADMIN.COMMAND.QUEUE` must be defined. The commands on this queue are processed by the PCF Command Server. This queue should have been defined when the queue manager was created; normally no action on your part will be required except to make sure that it exists.

For the IBM MQ Agent to create dynamic queues, a model queue must be defined with the name `NASTEL.MODEL.QUEUE`. Run the following MQSC command to define this model:

```
DEFINE QMODEL(NASTEL.MODEL.QUEUE) +
MSGDLVSQ(PRIORITY) DEFPSIST(NO) REPLACE
```

For the IBM MQ Agent to process IBM MQ events, all system event queues must be defined as alias queues. The base queue to which all system event queues should point is `NASTEL.EVENT.QUEUE`. Run the following MQSC commands:



You can use the MQMGM JCL with CSQUTIL to create all the required queues for agent, message server, and optionally, the event publisher. Review the related selections in this document, then edit MQMGM if required and use it with CSQUTIL.

```
DEFINE QLOCAL(NASTEL.EVENT.QUEUE) REPLACE

DELETE QLOCAL(SYSTEM.ADMIN.QMGR.EVENT) PURGE
DELETE QLOCAL(SYSTEM.ADMIN.CHANNEL.EVENT) PURGE
DELETE QLOCAL(SYSTEM.ADMIN.PERFM.EVENT) PURGE

DEFINE QALIAS(SYSTEM.ADMIN.QMGR.EVENT) +
  TARGQ(NASTEL.EVENT.QUEUE) REPLACE
DEFINE QALIAS(SYSTEM.ADMIN.CHANNEL.EVENT) +
  TARGQ(NASTEL.EVENT.QUEUE) REPLACE
DEFINE QALIAS(SYSTEM.ADMIN.PERFM.EVENT) +
  TARGQ(NASTEL.EVENT.QUEUE) REPLACE
DEFINE QALIAS(SYSTEM.ADMIN.CONFIG.EVENT) +
  TARGQ(NASTEL.EVENT.QUEUE) REPLACE
```

5.6.3 MMF Server (Optional)

The MMF Server (also known as the Message Manager or Message Server) is an optional component that provides display, editing, creating, and deleting of messages on IBM MQ queues. If there are no local requirements to use the MMF Server, do not configure as outlined in this paragraph.

For each queue manager for which it is to be used, an instance of the MMF Server must be running. In addition, the MMF Server requires the IBM MQ Agent to be running with the PCF Command Server active.

To run an instance of the MMF Server, you must define the following queues:

NASTEL.MMF.ADMIN.COMMAND.QUEUE – a local queue that is used by MMF Server as an inbound command queue

NASTEL.MMF.AUDIT.QUEUE – a local queue that is used for auditing purpose

To define these queues, run the following MQSC commands:

```
DEFINE QLOCAL (NASTEL.MMF.ADMIN.COMMAND.QUEUE) REPLACE
DEFINE QLOCAL (NASTEL.MMF.AUDIT.QUEUE) REPLACE
```

5.6.4 MQ Event Publisher

MQ Event Publisher is an optional component used for:

- Management fault tolerance by publishing events to multiple workgroup servers
- Co-existence with third-party management solutions and
- Publishing DLQ events.

On each z/OS node where it is to be used, an instance of the MQ Event Publisher must be running. One MQ Event Publisher can serve multiple queue managers. If you are not planning to use this facility, the configuration in this section is not required.

If you are running MQ Event Publisher on z/OS, you must define all system event queues as alias queues (these replace the aliases made above for the IBM MQ Agent). Each of these system event alias queues must point to a base queue named:

```
NASTEL.PUBSUB.EVENT.QUEUE333
```

Run the following MQSC commands to define these aliases:

```
DEFINE QLOCAL (NASTEL.PUBSUB.EVENT.QUEUE) REPLACE

DELETE QLOCAL (SYSTEM.ADMIN.QMGR.EVENT) PURGE
DELETE QLOCAL (SYSTEM.ADMIN.CHANNEL.EVENT) PURGE
DELETE QLOCAL (SYSTEM.ADMIN.PERFM.EVENT) PURGE
DELETE QLOCAL (SYSTEM.ADMIN.CONFIG.EVENT) PURGE

DEFINE QALIAS (SYSTEM.ADMIN.QMGR.EVENT) +
TARGQ (NASTEL.PUBSUB.EVENT.QUEUE) REPLACE
DEFINE QALIAS (SYSTEM.ADMIN.CHANNEL.EVENT) +
TARGQ (NASTEL.PUBSUB.EVENT.QUEUE) REPLACE
DEFINE QALIAS (SYSTEM.ADMIN.PERFM.EVENT) +
TARGQ (NASTEL.PUBSUB.EVENT.QUEUE) REPLACE
DEFINE QALIAS (SYSTEM.ADMIN.CONFIG.EVENT) +
TARGQ (NASTEL.PUBSUB.EVENT.QUEUE) REPLACE
```

5.7 Operating Nastel Navigator Components via JCL Procedures

The following sections present sample jobs for starting, restarting, and stopping Nastel Navigator components. They also include information on converting these jobs into started task procedures.

To execute these jobs as started tasks from an operator console, place them in a started task procedure library such as `SYS1.PROCLIB`. Make changes to the sample JCL, removing the `JOB` statement and any ending null (`//`) statement. Optionally, you may add a `PROC` statement and add symbolic substitution variables for operational flexibility.

From the console, the operator may then issue the following commands:

- S NSQMQ:** To start the IBM MQ Agent as a started task
- P NSQMQ:** To stop the IBM MQ Agent
- S LMSTOP:** An alternative way to stop the IBM MQ Agent
- S NSQMSG:** To start the MMF Server as a started task
- P NSQMSG:** To stop the MMF Server
- S MSSTOP:** An alternative way to stop the MMF Server
- S NSQPUB:** To start the MQ Event Publisher as a started task
- P NSQPUB:** To stop the MQ Event Publisher
- S PBSTOP:** An alternative way to stop the MQ Event Publisher
- S MQCSTOP:** To stop all Nastel Navigator components (see section 5.14, [Stopping Multiple Nastel Navigator Components](#), for more information).

The order in which agents are started is not critical; they may be started or stopped in any order and will operate correctly. The following is the logical startup order:

1. MQ Event Publisher
2. MQ Agent
3. MMF Server

Logical shutdown would be the reverse.

5.8 Starting IBM MQ Agent

Before starting the IBM MQ Agents on z/OS the following system components must be running:

- IBM MQ queue manager
- IBM MQ command server
- TCP/IP.

5.8.1 Starting IBM MQ Agent

To start the IBM MQ Agents, run a job based on the one below. See NSQMQ in data set M6WMQ.JCL. Before starting, read all the related subsections below.

```
//NSQMQ JOB //(xxxx,xxxx),'xxxxxx', CLASS=A,MSGCLASS=A,
//MSGLEVEL=(1,1),NOTIFY=userid
//*****
//* ++hlq++ is a high level qualifier – must be changed
//* ++mqser++ replace with the high level qualifier of the WMQ
library
//* -mMQM specifies the name of the Workgroup Server
//* -s5010 specifies port on which WMQ Agent listens
//* use parameter +a to enable authentication mode
//* use parameter -a to disable authentication mode (default)
//* use parameter +u to enable user authorization checks
//* use parameter -u to disable user authorization checks (default)
//* use parameter -c20 to increase subtask count
//*****
//S1 EXEC PGM=NSQMQ,REGION=4M,TIME=1440,
// PARM='-mMQM -s5010'
//STEPLIB DD DSN=++hlq++.LOAD,DISP=SHR
// DD DSN=++mqser++.SCSQANLE,DISP=SHR
// DD DSN=++mqser++.SCSQAUTH,DISP=SHR
// DD DSN=++mqser++.SCSQLOAD,DISP=SHR
//EDCMTF DD DSN=++hlq++.LOAD,DISP=SHR
//*SYSTCPD DD DSN=++tcpd++,DISP=SHR
//MQSINI DD DSN=++hlq++.MQS.INI,DISP=SHR
//MQGROUP DD DSN=++hlq++.MQGROUP.INI,DISP=SHR
//PERMITS DD DSN=++hlq++.PERMITS.INI,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//
```

5.8.1.1 Value of REGION

If REGION=4M is specified, the z/OS default will provide 4MB below the line and 32MB above the line. The load module and most of its working storage is taken from above the line, and the 32MB limit should be adequate to support up to our maximum of ten queue managers. However, it is possible that the z/OS storage defaults were changed during the installation process. If the storage defaults were changed, it may be necessary to specify some locally specific option to ensure sufficient storage. Consult the local system programming staff for more information.

Nastel recommends setting REGION size in JCL PARM to REGION=0M. Setting REGION=0M does not mean that the job will consume the total amount of memory of virtual storage. It simply sets the upper limit, leaving the option of whether to designate that amount to the application. If REGION=0M is not acceptable, you can use REGION=16M as a starting point, increasing in the unlikely event of storage problems.

If queue managers are defined in the system, you may need to specify the -c parameter to increase the number of service subtasks activated by the IBM MQ agent.

`-c<subtasks>` (z/OS only)

Specifies the maximum number of z/OS MTF subtasks. Nastel recommends using `-c30` for the `nsqmq` task.

5.8.1.2 Running NSQMQ as a Started Task

NSQMQ may be run as a started task from an operator console. To run NSQMQ, place the sample JCL into a started task procedure library such as `SYS1.PROCLIB`. Ensure some changes to the sample JCL, removing the `JOB` statement and any ending null (`//`) statement. Optionally you may add a `PROC` statement and symbolic substitution variables for operational flexibility. For example:

```
//NSQMQ          PROC
//S1             EXEC PGM=NSQMQ,REGION=4M,TIME=1440,
//              PARM=' -mMQM -s5010 '
//STEPLIB       DD   DSN=++hlq++.LOAD,DISP=SHR
//              DD   DSN=++mqser++.SCSQANLE,DISP=SHR
//              DD   DSN=++mqser++.SCSQAUTH,DISP=SHR
//              DD   DSN=++mqser++.SCSQLOAD,DISP=SHR
//EDCMTF        DD   DSN=++hlq++.LOAD,DISP=SHR
//*SYSTCPD      DD   DSN=++tcpd++,DISP=SHR
//MQSINI        DD   DSN=++hlq++.MQS.INI,DISP=SHR
//MQGROUP       DD   DSN=++hlq++.MQGROUP.INI,DISP=SHR
//PERMITS       DD   DSN=++hlq++.PERMITS.INI,DISP=SHR
//SYSPRINT      DD   SYSOUT=*
//SYSOUT        DD   SYSOUT=*
```

5.8.1.3 PARM Field Conventions

The `PARM` field of the `EXEC` statement must use the following conventions:

- Each parameter must be coded with a hyphen, a key symbol and then the value of the parameter (if applicable), as in the example `-mMQM`.
- Each parameter must be separated by blanks.



All parameters in the `PARM` field are optional, but you should be aware of their default values. For a full list of parameters and their default values, see the appendices.

5.8.1.4 The MQSINI Configuration File

The `MQSINI` file is a configuration file required by the IBM MQ Agent. It lists the queue managers that are managed by the IBM MQ Agent. For each queue manager, the file must contain information in the following case-sensitive format:

```
QueueManager:
  Name=CSQ1
  Qualifier=+MQM1
```

`Name` represents the name of the queue manager. `Qualifier` represents the Command Prefix String (CPF) for that queue manager.

When there are multiple queue managers, each descriptive section must begin with the Queue Manager: heading, as in this example:

```
QueueManager:
  Name=CSQ1
  Qualifier=+MQM1
QueueManager:
  Name=CSQ2
  Qualifier=+MQM2
```

If a large number of queue managers are defined in this file, you may need to specify the `-c` parameter to increase the number of service subtasks activated by the IBM MQ Agent. For more information, see [Appendix B](#) for arguments applicable to the IBM MQ Agent.

5.8.1.5 The MQGROUP Configuration File

The MQGROUP file is another configuration file required by the IBM MQ Agent. It lists all Nastel Navigator workgroup servers under which the z/OS IBM MQ Agent will register.

The MQGROUP file contains information in the following format:

```
Group: :MQEVENT=MQM
Group: :MQM=MQM
MQM: :Node=11.0.0.28
MQM: :Service=4005
```

5.8.1.6 The PERMITS Configuration File

The PERMITS file is another configuration file required by the IBM MQ Agent. It is used to grant system privileges to Nastel Navigator users.

The PERMITS file contains information in the following format:

```
UserAuthority: :*:: $SYSTEM$ \*=1
UserAuthority: :*=1
```

5.8.2 Restarting the IBM MQ Agent

If the IBM MQ Agent is stopped for any reason, you can restart it by resubmitting the NSQMQ job.

5.9 Stopping the IBM MQ Agent

There are a few ways to stop IBM MQ Agents. Use any of the methods below:

Use the Shutdown command on your Navigator console.

Enter the z/OS P (STOP) command naming the IBM MQ Agent address space.

By running a job based on the sample below. See LMSTOP in data set M6WMQ.JCL.

Before starting the job, read all the related subsections below.

```
//LMSTOP JOB //(xxxx,xxxx), 'xxxxxx', CLASS=A, MSGCLASS=A,
//MSGLEVEL=(1,1), NOTIFY=userid
```

```

//*****
//* ++hlq++ is a high level qualifier – must be changed
//* ++mqser++ replace with the high level qualifier of WMQ library
//* -s5010 specifies port on which WMQ Agent listens
//* use parameter +a to enable authentication mode
//* use parameter -a to disable authentication mode (default)
//* -nZOS specifies the host computer name
//*****
//S1 EXEC          PGM=NSQSTOP,REGION=4M,PARM='-s5010 -nZOS'
//STEPLIB         DD   DSN=++hlq++.LOAD,DISP=SHR
//               DD   DSN=++mqser++.SCSQAUTH,DISP=SHR
//               DD   DSN=++mqser++.SCSQLOAD,DISP=SHR
//               DD   DSN=++mqser++.SCSQANLE,DISP=SHR
//*SYSTCPD        DD   DSN=++tcpd++,DISP=SHR
//SYSPRINT        DD   SYSOUT=*
//SYSOUT          DD   SYSOUT=*
//

```

5.9.1 Value of REGION

You should set the value of REGION to 4M.

5.9.2 Running the LMSTOP Job as a Started Task

LMSTOP may be run as a started task from an operator console. Place the sample JCL into a started task procedure library such as SYS1.PROCLIB. Make some changes to the sample JCL, removing the JOB statement and any ending null (//) statement. Optionally, add a PROC statement and symbolic substitution variables for operational flexibility. For example:

```

//LMSTOP          PROC
//LMSTOP          EXEC  PGM=NSQSTOP,PARM='-s5010'
//STEPLIB         DD   DSN=++hlq++.LOAD,DISP=SHR
//               DD   DSN=++mqser++.SCSQAUTH,DISP=SHR
//               DD   DSN=++mqser++.SCSQLOAD,DISP=SHR
//               DD   DSN=++mqser++.SCSQANLE,DISP=SHR
//*SYSTCPD        DD   DSN=++tcpd++,DISP=SHR
//SYSPRINT        DD   SYSOUT=*
//SYSOUT          DD   SYSOUT=*

```

5.9.3 Conventions Used in the PARM Field

The PARM field of the EXEC statement must use the following conventions:

Each parameter must be coded with a hyphen, a key symbol and then the value of the parameter (if applicable), as in the example –s5010.

Each parameter must be separated by blanks.



All parameters in the PARM field are optional, but you should be aware of their default values. For a full list of parameters and their default values, see [Appendix B](#).

5.10 Starting the Message Server

Before you run the MMF Server on z/OS, you must start the following system components:

- IBM MQ queue manager
- IBM MQ command server
- PCF Command Server

5.10.1 Start the Message Server

To start the MMF Server, you need to run a JCL job based on the one below. See NSQMSG in data set M6WMQ.JCL. Prior to starting the job, read all the related subsections below.

```
//NSQMSG JOB //(xxxx,xxx), 'xxxxxx', CLASS=A, MSGCLASS=A,
//MSGLEVEL=(1,1), NOTIFY=userid
//*****
/* ++hlq++ is a high level qualifier – must be changed
/* ++mqser++ replace with the high level qualifier of the WMQ
library
/* ++qmgr++ specifies the name of the queue manager
//*****
//S1 EXEC PGM=NSQMSG, REGION=4M, TIME=1440,
// PARM=' -m++qmgr++'
//STEPLIB DD DSN=++hlq++.LOAD, DISP=SHR
// DD DSN=++mqser++.SCSQAUTH, DISP=SHR
// DD DSN=++mqser++.SCSQLOAD, DISP=SHR
// DD DSN=++mqser++.SCSQANLE, DISP=SHR
//EDCMTF DD DSN=++hlq++.LOAD, DISP=SHR
//*SYSTCPD DD DSN=++tcpd++, DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//
```

5.10.1.1 Value of REGION

You should set the value of REGION to 4M.

5.10.1.2 Running the NSQMSG Job as a Started Task

NSQMSG may be run as a started task from an operator console. Place the sample JCL into a started task procedure library such as SYS1.PROCLIB. You must also make some changes to the sample JCL, removing the JOB statement and any ending null (//) statement. Optionally, add a PROC statement and symbolic substitution variables for operational flexibility. For example:

```
//NSQMSG PROC
//S1 EXEC PGM=NSQMSG, REGION=4M, TIME=1440,
// PARM=' -m++qmgr++'
//STEPLIB DD DSN=++hlq++.LOAD, DISP=SHR
// DD DSN=++mqser++.SCSQAUTH, DISP=SHR
// DD DSN=++mqser++.SCSQLOAD, DISP=SHR
// DD DSN=++mqser++.SCSQANLE, DISP=SHR
```

```
//EDCMTF          DD      DSN=++hlq++.LOAD,DISP=SHR
//*SYSTCPD        DD      DSN=++tcpd++,DISP=SHR
//SYSPRINT        DD      SYSOUT=*
//SYSOUT          DD      SYSOUT=*
```

5.10.1.3 PARM Field Conventions

The PARM field of the EXEC statement must use the following conventions:

Each parameter must be coded with a hyphen, a key symbol, and the value of the parameter (if applicable), as in the example `-mQMGR`.

Each parameter must be separated by blanks.



Only one parameter in the PARM field is required: `-m++qmgr++`, where `++qmgr++` should be replaced with the queue manager name. All other parameters in the PARM field are optional, but you should be aware of their default values. For a full list of parameters and their default values, refer to the appendices.

5.10.1.4 Restarting the Message Server

If the MMF Server is stopped for any reason, you can restart it by resubmitting the NSQMSG job.

5.11 Stopping the Message Server

To stop the MMF Server, enter the z/OS P (STOP) command naming the MMF Server address space. Alternatively, run a job based on the one below. See MSSTOP in data set M6WMQ.JCL. Before starting the job, read all the related subsections below.

```
//MSSTOP  JOB  //(xxxx,xxxx), 'xxxxxx', CLASS=A,MSGCLASS=A,
//MSGLEVEL=(1,1),NOTIFY=userid
//*****
//* ++hlq++ is a high level qualifier
//* ++mqser++ replace with the high level qualifier of the WMQ
library
//* ++qmgr++ specifies the name of the queue manager
//*****
//S1 EXEC          PGM=ENDMCS,PARM='++qmgr++'
//STEPLIB          DD      DSN=++hlq++.LOAD,DISP=SHR
//                 DD      DSN=++mqser++.SCSQAUTH,DISP=SHR
//                 DD      DSN=++mqser++.SCSQLOAD,DISP=SHR
//                 DD      DSN=++mqser++.SCSQANLE,DISP=SHR
//SYSPRINT         DD      SYSOUT=*
//SYSOUT           DD      SYSOUT=*
//
```

5.11.1 Value of REGION

You should set the value of REGION to 4M.

5.11.2 Running MSSTOP as a Started Task

MSSTOP may be run as a started task from an operator console. Place the sample JCL into a started task procedure library such as `SYS1.PROCLIB`. Make changes to the sample JCL, removing the JOB statement and any ending null (`//`) statement. Optionally, add a PROC statement and symbolic substitution variables for operational flexibility. For example:

```
//MSSTOP          PROC
//S1              EXEC PGM=ENDMCS, PARM='++qmgr++'
//STEPLIB        DD DSN=++hlq++.LOAD, DISP=SHR
//               DD DSN=++mqser++.SCSQAUTH, DISP=SHR
//               DD DSN=++mqser++.SCSQLOAD, DISP=SHR
//               DD DSN=++mqser++.SCSQANLE, DISP=SHR
//SYSPRINT       DD SYSOUT=*
//SYSOUT         DD SYSOUT=*
```

5.11.3 PARM Field Conventions

Each parameter in the PARM field of the EXEC statement must be separated by blanks.



One parameter is required: `++qmgr++`, where `++qmgr++` is replaced by the queue manager name.

5.12 Starting the Event Publish-Subscriber

Before running the MQ Event Publisher on z/OS, you must start the following system components:

- IBM MQ queue manager
- IBM MQ command server
- TCP/IP
- PCF Command Server
- MQ Agent

5.12.1 Starting the Event Publish-Subscriber

To start the MQ Event Publisher, run a JCL job based on the one below. See NSQPUB in data set M6WMQ.JCL. Before starting the job, read all the related subsections below.

```
//NSQPUB JOB //(xxxx,xxxx), 'xxxxxx', CLASS=A, MSGCLASS=A,
//MSGLEVEL=(1,1), NOTIFY=userid
//*****
//* ++hlq++ is a high level qualifier – must be changed
//* ++mqser++ replace with high level qualifier of the WMQ library
//*****
//S1 EXEC PGM=NSQPUB, REGION=4M, TIME=1440,
// PARM=' -dSYSTEM.DEAD.LETTER.QUEUE'
//STEPLIB DD DSN=++hlq++.LOAD, DISP=SHR
// DD DSN=++mqser++.SCSQAUTH, DISP=SHR
// DD DSN=++mqser++.SCSQLOAD, DISP=SHR
```

```
// DD DSN=++mqser++.SCSQANLE,DISP=SHR
//EDCMTF DD DSN=++hlq++.LOAD,DISP=SHR
//*SYSTCPD DD DSN=++tcpd++,DISP=SHR
//MQSINI DD DSN=++hlq++.MQS.INI,DISP=SHR
//SUBSCRIB DD DSN=++hlq++.SUBSCRIB.INI,DISP=SHR
//*PARM DD DSN=++hlq++.PARM,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//
```

5.12.1.1 Value of REGION

Set the value of REGION to 4M.

5.12.1.2 Running the NSQPUB Job as a Started Task

NSQPUB may be run as a started task from an operator console. Place the sample JCL into a started task procedure library such as `SYS1.PROCLIB`. Make changes to the sample JCL, removing the JOB statement and any ending null (//) statement. Optionally add a PROC statement and symbolic substitution variables for operational flexibility.

For example:

```
//NSQPUB PROC
//S1 EXEC PGM=NSQPUB,REGION=4M,TIME=1440,
// PARM=' -dSYSTEM.DEAD.LETTER.QUEUE'
//STEPLIB DD DSN=++hlq++.LOAD,DISP=SHR
// DD DSN=++mqser++.SCSQAUTH,DISP=SHR
// DD DSN=++mqser++.SCSQLOAD,DISP=SHR
// DD DSN=++mqser++.SCSQANLE,DISP=SHR
//EDCMTF DD DSN=++hlq++.LOAD,DISP=SHR
//*SYSTCPD DD DSN=++tcpd++,DISP=SHR
//MQSINI DD DSN=++hlq++.MQS.INI,DISP=SHR
//SUBSCRIB DD DSN=++hlq++.SUBSCRIB.INI,DISP=SHR
//*PARM DD DSN=++hlq++.PARM,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
```

5.12.1.3 PARM Field Conventions

The PARM field of the EXEC statement must use the following conventions:

Each parameter must be coded with a hyphen, a key symbol and then the value of the parameter (if applicable). Example: `-s5010`.

Each parameter must be separated by blanks.



All parameters in the PARM field are optional, but you should be aware of their default values. For a full list of parameters and their default values, see the appendices.

5.12.1.4 Use of a PARM File

In the z/OS environment a DD statement with a name PARM can be included in the JCL stream to define the run time parameters of the MQ Event Publisher. All parameters must be listed with a delimiter “;” (semicolon) or “,” (comma). The file can consist of one or more records. MQ Event Publisher scans the file for parameters until it reaches the end of the file. For example, the file can have the following parameters:

```
-t;-w50000;-dDEAD.QUEUE  
-h10000; -g500000
```

The user must use this file if it is necessary to code parameters with a total length that exceed the maximum allowed by z/OS. If PARM field of the EXEC statement and PARM DD statements are coded, the parameters from both sources will be used.

If a parameter is coded in both PARM field and in PARM file, the value from file will override value from PARM field.

5.12.1.5 The MQSINI Configuration File

The MQSINI file is a configuration file required by MQ Event Publisher and the IBM MQ Agent. It lists the queue managers that are managed by the IBM MQ Agent. For each queue manager, the file must contain information in this format:

```
QueueManager:  
  Name=CSQ1  
  Qualifier=+MQM1
```

Name represents the name of the queue manager. Qualifier represents the command prefix string (CPF) for that queue manager.

When there are multiple queue managers, each descriptive section must begin with the QueueManager: heading, as in this example:

```
QueueManager:  
  Name=CSQ1  
  Qualifier=+MQM1  
QueueManager:  
  Name=CSQ2  
  Qualifier=+MQM2
```

5.12.1.6 The SUBSCRIB Configuration File

The configuration file SUBSCRIB is required by MQ Event Publisher. The file defines permanent subscribers that receive events. Information about subscribers is stored in the following format:

```
qmgr_name  
QueueName  
EventTypeMask
```

For example, imagine a subscriber named SubOne which has the following characteristics:

It is under a queue manager named CSQ1

It receives event messages via a queue named SUBSCRIB.QUEUE1

It has an EventTypeMask of -1 (meaning that it should receive all events)

In the SUBSCRIB.INI file, SubOne would be defined like this:

```
CSQ1
SUBSCRIB.QUEUE1
-1
```

If there are multiple subscribers, they can be defined in sequence, like this:

```
CSQ1
SUBSCRIB.QUEUE1
-1
CSQ2
SUBSCRIB.QUEUE2
-1
```



The SUBSCRIB configuration file should not have numbering in positions 72-80.

5.12.1.7 SEVERITY Configuration File

The SEVERITY file is a configuration file which is required only when the -W option of the MQ Event Publisher is used. It specifies the events for which WTO messages will be issued and the severity assigned. For example, to issue WTOs for EXRC_DEAD_LETTER_MESSAGE (20016) with severity 3, MQRC_Q_DEPTH_HIGH (2224) with severity 2 and MQRC_Q_DEPTH_LOW (2225) with severity 1, you could use the following:

```
20016=3
2224=2
2225=1
```

When restarted, the IBM MQ Agent NSQ0801I messages will be issued for dead letter queue messages, queue depth high events and queue depth low events.

5.12.2 Restarting the Event Publish-Subscriber

If MQ Event Publisher is stopped for any reason, you can restart it by resubmitting the NSQPUB job.

5.13 Stopping the Event Publish-Subscriber

To stop the MQ Event Publisher, you may enter the z/OS P (STOP) command naming the MQ Event Publisher address space. Alternatively, you may run a job based on the one below. Before you start the job, make sure you read all the related subsections below.

```
//PBSTOP JOB //(xxxx,xxxx), 'xxxxxx', CLASS=A,MSGCLASS=A,
//MSGLEVEL=(1,1),NOTIFY=userid
//*****
/* ++hlq++ is a high level qualifier
/* ++mqser++ replace with the high level qualifier of the WMQ
library
//*****
//S1 EXEC PGM=PUBEND,REGION=4M
//STEPLIB DD DSN=++hlq++.LOAD,DISP=SHR
```

```
//          DD      DSN=++mqser++.SCSQAUTH,DISP=SHR
//          DD      DSN=++mqser++.SCSQLOAD,DISP=SHR
//          DD      DSN=++mqser++.SCSQANLE,DISP=SHR
//*SYSTCPD  DD      DSN=++tcpd++,DISP=SHR
//SYSPRINT DD      SYSOUT=*
//SYSOUT   DD      SYSOUT=*
//
```



To stop the MQ Event Publisher, do not specify +a.

5.13.1 Value of REGION

You should set the value of REGION to 4M.

5.13.2 Running the PBSTOP Job as a Started Task

PBSTOP may be run as a started task from an operator console. To do this, you must place the sample JCL into a started task procedure library such as SYS1.PROCLIB. You must also make some changes to the sample JCL, removing the JOB statement and any ending null (//) statement. Optionally you may add a PROC statement and symbolic substitution variables for operational flexibility. For example:

```
//PBSTOP      PROC
//S1          EXEC  PGM=PUBEND,REGION=4M,PARM='-s5500'
//STEPLIB    DD      DSN=++hlq++.LOAD,DISP=SHR
//           DD      DSN=++mqser++.SCSQAUTH,DISP=SHR
//           DD      DSN=++mqser++.SCSQLOAD,DISP=SHR
//           DD      DSN=++mqser++.SCSQANLE,DISP=SHR
//*SYSTCPD   DD      DSN=++tcpd++,DISP=SHR
//SYSPRINT   DD      SYSOUT=*
//SYSOUT     DD      SYSOUT=*
```

5.13.3 Conventions used in the PARM Field

The PARM field of the EXEC statement for the PBSTOP job must use the same conventions as in [5.12.1.3](#).

5.14 Stopping Multiple Nastel Navigator Components

To stop multiple Nastel Navigator components at one time, you may run a job based on the one below. Before you start the job, make sure you read all the related subsections above. Customization will be required so that a job step is present to shut down every Nastel Navigator function that may have been started.

```
//MQCSTOP  JOB  //(xxxx,xxxx), 'xxxxxxx',CLASS=A,MSGCLASS=A,
//MSGLEVEL=(1,1),NOTIFY=userid
//*****
//* ++hlq++ is a high level qualifier
```

```

/* ++mqser++ replace with the high level qualifier of the
/*                               WebSphere MQ library
/* ++qmgr++ is the name of the queue manager
/******
//LMSTOP      EXEC PGM=NSQSTOP,PARM='-s5010'
//STEPLIB    DD   DSN=++hlq++.LOAD,DISP=SHR
//           DD   DSN=++mqser++.SCSQAUTH,DISP=SHR
//           DD   DSN=++mqser++.SCSQLOAD,DISP=SHR
//           DD   DSN=++mqser++.SCSQANLE,DISP=SHR
//*SYSTCPD   DD   DSN=++tcpd++,DISP=SHR
//SYSPRINT   DD   SYSOUT=*
//SYSOUT     DD   SYSOUT=*
/*
//MSSTOP     EXEC PGM=ENDMCS,PARM='++qmgr++'
//STEPLIB    DD   DSN=++hlq++.LOAD,DISP=SHR
//           DD   DSN=++mqser++.SCSQAUTH,DISP=SHR
//           DD   DSN=++mqser++.SCSQLOAD,DISP=SHR
//           DD   DSN=++mqser++.SCSQANLE,DISP=SHR
//SYSPRINT   DD   SYSOUT=*
//SYSOUT     DD   SYSOUT=*
/*
//PBSTOP     EXEC PGM=PUBSTOP,PARM='-s5500 -a'
//STEPLIB    DD   DSN=++hlq++.LOAD,DISP=SHR
//           DD   DSN=++mqser++.SCSQAUTH,DISP=SHR
//           DD   DSN=++mqser++.SCSQLOAD,DISP=SHR
//           DD   DSN=++mqser++.SCSQANLE,DISP=SHR
//*SYSTCPD   DD   DSN=++tcpd++,DISP=SHR
//SYSPRINT   DD   SYSOUT=*
//SYSOUT     DD   SYSOUT=*
//

```

5.14.1 Running MQCSTOP Job as a Started Task

MQCSTOP may be run as a started task from an operator console. Place the sample JCL into a started task procedure library such as SYS1.PROCLIB. Make changes to the sample JCL, removing the JOB statement and any ending null (//) statement. Optionally, add a PROC statement and symbolic substitution variables for operational flexibility.

For example:

```

//MQCSTOP    PROC
//LMSTOP     EXEC PGM=PUBSTOP,PARM='-s5010 -a'
//STEPLIB    DD   DSN=++hlq++.LOAD,DISP=SHR
//           DD   DSN=++mqser++.SCSQAUTH,DISP=SHR
//           DD   DSN=++mqser++.SCSQLOAD,DISP=SHR
//           DD   DSN=++mqser++.SCSQANLE,DISP=SHR
//*SYSTCPD   DD   DSN=++tcpd++,DISP=SHR
//SYSPRINT   DD   SYSOUT=*
//SYSOUT     DD   SYSOUT=*
/*
//MSSTOP     EXEC PGM=ENDMCS,PARM='++qmgr++'
//STEPLIB    DD   DSN=++hlq++.LOAD,DISP=SHR
//           DD   DSN=++mqser++.SCSQAUTH,DISP=SHR
//           DD   DSN=++mqser++.SCSQLOAD,DISP=SHR

```

```
// DD DSN=++mqser++.SCSQANLE, DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//*
//PBSTOP EXEC PGM=PUBSTOP, PARM=' -s5500'
//STEPLIB DD DSN=++hlq++.LOAD, DISP=SHR
// DD DSN=++mqser++.SCSQAUTH, DISP=SHR
// DD DSN=++mqser++.SCSQLOAD, DISP=SHR
// DD DSN=++mqser++.SCSQANLE, DISP=SHR
//*SYSTCPD DD DSN=++tcpd++, DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
```



After you have completed installation of Nastel Navigator and started any of the components, verify that the queues created match the ones summarized in the appendices.

5.15 Uninstalling Nastel Navigator on z/OS

Removing Nastel Navigator from z/OS Systems:

1. Delete all Nastel Navigator datasets, which include:
 - The load library you installed the product into
 - The group dataset used for `dd:MQGROUP`
 - The *ini* dataset used for `dd:MQSINI`
 - The permit dataset used for `dd:PERMITS`
 - The permanent subscriber file used for `dd:SUBSCRIB`
 - The event severity file used for `dd:SEVERITY` (if present)
2. Remove any RACF *userid*s setup to execute the `NSQMQL`, `NSQMVS`, `NSQMSG` and `NSQPUB` started tasks. For any *userid*s that are not removed, remove accesses to IBM MQ objects that were applied for Nastel Navigator.
3. Delete the following Nastel Navigator queues and model queue:
 - `NASTEL.EVENT.QUEUE`
 - `NASTEL.PUBSUB.EVENT.QUEUE`
 - `NASTEL.MMF.ADMIN.COMMAND.QUEUE`
 - `NASTEL.MMF.AUDIT.QUEUE`
 - `NASTEL.MODEL.QUEUE`



You can use the `MQMGMREM` JCL with `CSQUTIL` to delete the Nastel Navigator queues for agent, message server, and optionally the event publisher and restore the original event queue definitions. Review the related sections in this document, then edit `MQMGMREM` if required and use it with `CSQUTIL`.

4. Redefine your original system queues according to your original IBM MQ system installation:
 - `SYSTEM.ADMIN.QMGR.EVENT`

- `SYSTEM.ADMIN.CHANNEL.EVENT`
- `SYSTEM.ADMIN.PERFM.EVENT`
- `SYSTEM.ADMIN.CONFIG.EVENT`

**NOTE**

Since these queues are in use when IBM MQ is running, you should redefine them at IBM MQ restart. However, it is not necessary to redefine these queues: the system will work just fine with the Nastel aliases, even when the Nastel software is not present.

Chapter 6: OS/400 Installation

This chapter explains how to install Nastel Navigator on OS/400 (the original operating system name; now called IBM i and formerly called i5/OS).

6.1 Nastel Navigator Components for OS/400

The Nastel Navigator installation for OS/400 installs the following components on each machine:

- Nastel Navigator Agent
- Nastel Navigator Message Server
- Nastel Navigator Publish-Subscriber
- Nastel Navigator Utilities

6.2 System Requirements

- *An Nastel Navigator installation for OS/400 requires:*
 - i5/OS V5R3 or i5/OS V5R4
 - TCP/IP services
 - IBM MQ for iSeries, Version 7 or higher
 - Approximately 22 MB of hard disk space; in temporary library, 10 MB space for save file.

6.3 Security for OS/400

In the OS/400 version of Nastel Navigator, only users with the *ALLOBJ (all objects) authority can perform the actions below:

- Ping channel
- Change channel
- Create channel
- Copy channel
- Delete channel
- Reset channel
- Start channel
- Stop channel

The users QPGMR and QSYSOPR are assigned *ALLOBJ authority by default. Other users must obtain *ALLOBJ authority from the system administrator.

6.4 Installing Nastel Navigator

This section walks you through the installation procedures for OS/400. There are two types of installation available for Nastel Navigator on OS/400:

- **New Installation** – no previous installation procedures for OS/400 ([section 6.4.1](#))
-

- **Upgrade Installation** – a previous installation of Nastel Navigator exists; that is prior to V6.5 ([section 6.4.2](#)).

6.4.1 New Installation of Nastel Navigator

Follow these instructions if this is a first-time installation.

1. Log on as a user, such as QSECOFR, with *ALLOBJ authority.
2. The installation procedure will use Integrated File System (IFS). Ensure it is available and that you have the authority to use it.
3. On the installation CD, navigate to the OS/400 folder and load the ZIP file

`M6WMQ[version]_OS400_osver.zip` to a Windows platform,

where *version* is the Nastel Navigator product version and *osver* is the OS/400 system version.release.

Example: `M6WMQ_6.0.24_V5R3.zip`

4. Unzip the file to a temporary folder on the Windows system. The ZIP file contains an OS/400 save file named `M6WMQosver.SAVF`.

Example: `M6WMQV5R3.SAVF`

5. On the OS/400, create a temporary library and save file with *ALL authority for the product installation. Example:

`QSYS/CRTLIB LIB (NASTEMP) TEXT (Temp lib for Nastel Navigator installation)`

`QSYS/CRTSAVF FILE (NASTEMP/M6WMQV5R3) AUT (*ALL)`

You can also perform step 5 remotely from the PC while using the FTP client as follows:

```
ftp> quote rcmd crtlib lib (NASTEMP) TEXT (Temp lib for Nastel Navigator installation)
```

```
ftp> quote rcmd crtsavf file (NASTEMP/M6WMQV5R3) AUT (*ALL)
```

If you have an existing save file you want to reuse, you can clear it remotely from the FTP client as follows:

```
ftp> quote rcmd clrsavf file (NASTEMP/M6WMQV5R3)
```

6. FTP, in binary mode, the save file from Windows machine to the temporary save file you created on the OS/400 in step 5.

```
ftp> cd NASTEMP
```

```
ftp> binary
```

```
ftp> put M6WMQV5R3.SAVF M6WMQV5R3
```

7. You can display the contents of the save file and determine the library from which the objects were saved before restoring it:

```
DSPSAVF FILE (NASTEM333P/M6WMQV5R3)
```

A saved file will be displayed similar to the figure below:

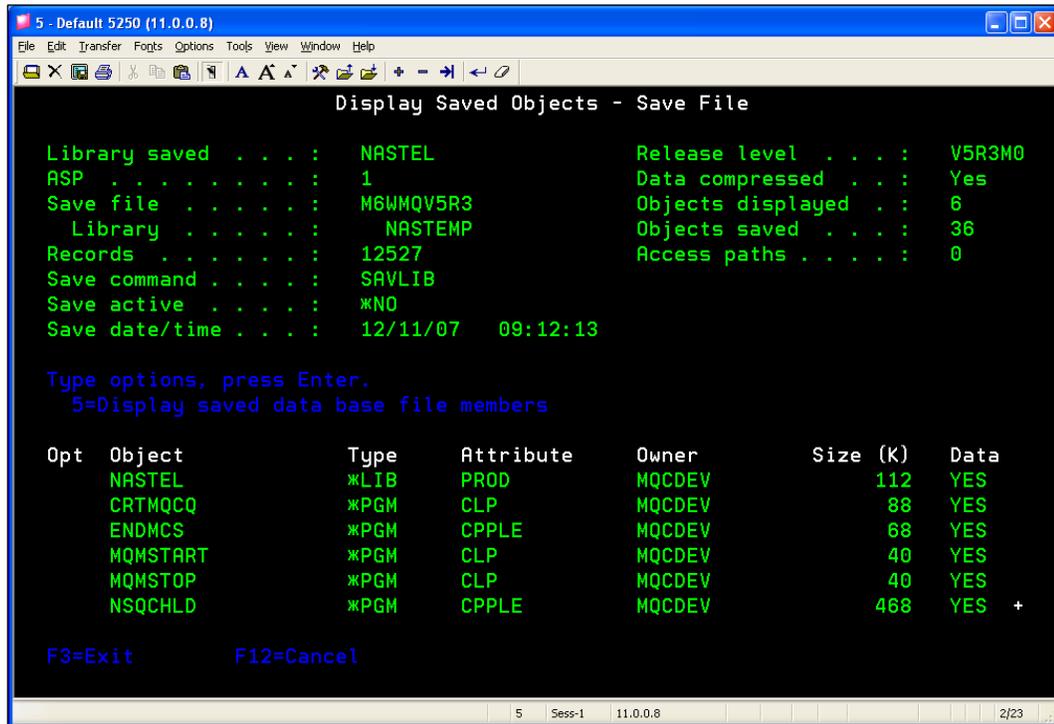


Figure 7-1. Displayed Save File for Nastel Navigator

- On the OS/400, use the following command to restore the save file to the library from which it was created (NASTEL as shown in Figure 7-1) by specifying the library name in the SAVLIB parameter.

```
RSTLIB SAVLIB(libraryname) DEV(*SAVF) SAVF(savefilename)
Example: RSTLIB SAVLIB(NASTEL) DEV(*SAVF) SAVF(NASTEMP/M6WMQV5R3)
```

- Verify that the library has been created. In this example the library name is NASTEL. Use the WRKLIB LIB(libraryname) command. The library should contain the members listed in Table 6-1:

Table 6-1. Members in the NASTEL Library		
Member	Type	Description
CRTMQCQ	*CMD	Create Nastel Navigator queues
RMVMQCQ	*CMD	Remove Nastel Navigator queues
CRTMQCQ	*PGM	Create Nastel Navigator queues
ENDMCS	*PGM	Stop Nastel Navigator Message Server
MQMSTART	*PGM	QMGR start program. Called from the Nastel Navigator agent.
MQMSTOP	*PGM	QMGR stop program. Called from the Nastel Navigator agent.
NSQCHLD	*PGM	Nastel Navigator agent handler. This job is started when NSQM is running and the node is managed from the Nastel Navigator Explorer.
NSQCL	*PGM	Client utility

NSQCODE	*PGM	Reason code lookup utility
NSQMQ	*PGM	Nastel Navigator Agent executable
NSQMSG	*PGM	Nastel Navigator Message Server
NSQMMFCL	*PGM	Message Client utility
NSQMSGF	*PGM	Message file utility
NSQPING	*PGM	Ping utility
NSQPUB	*PGM	Nastel Navigator Event Publish-Subscriber
NSQEVQQLSR	*PGM	Publish-Subscriber Event Listener
NSQDLQLSR	*PGM	Publish-Subscriber Dead Letter Queue Listener
NSQSTOP	*PGM	Stop IBM MQ Agent
PUBEND	*PGM	Stop MQ Event Publish-Subscriber
RMVMQCQ	*PGM	Remove Nastel Navigator queues
LIBNSQAPI	*SRVPGM	Common API library
LIBNSQAUM	*SRVPGM	Security API library
LIBNSQAUTH	*SRVPGM	Authorization API library
LIBNSQCLNX	*SRVPGM	Communication API library
LIBNSQMCTX	*SRVPGM	IBM MQ interface API library
LIBNSQSIGN	*SRVPGM	Authentication API library
README	*FILE	Readme files
INI	*FILE	Nastel Navigator agent initialization files. Contains source members MQGROUP, NSQRC.INI and PERMITS.INI. A member having the local queue manager name is dynamically created by NSQPUB at runtime to hold subscriber data.
MQSC	*FILE	MQSC scripts to configure queues for Nastel Navigator Agent, Message Server, and Event Publish Subscriber.
MQSCTEMP	*FILE	Temporary MQSC work file used by Nastel Navigator agent.
SCRIPTS	*FILE	Event action scripts
NSQHELP	*PNLGRP	Nastel Navigator help panel

10. Make sure that at least one IBM MQ queue manager and command server are running.

11. Add Nastel library to the libraries list:

```
ADDLIBLE LIB(NASTEL) [ POSITION (*LAST) ]
```

12. Before you create the Nastel Navigator administrative queues, you must decide:

- a. Do you want to see IBM MQ events in Nastel Navigator? This will require changing the `SYSTEM.ADMIN.xxx.EVENT` local queues to alias queues that point to a Nastel queue.
- b. Do you want to publish the events to multiple workgroup servers?
- c. Do you want to see dead-letter queue events in Nastel Navigator?

If your answer to (a) is no (and your answers to (b) and (c) would also be no), edit the "NASTEL" MQSC script by commenting out all the commands that relate to aliasing the `SYSTEM.ADMIN.xxx.EVENT` queues to the publish-subscribe event queue, `NASTEL.PUBSUB.EVENT.QUEUE`:

```
EDTF NASTEL/MQSC (NASTEL)
```

Execute the `CRTMQCQ` command below **without** the `PUBQ` event publisher option.

If your answer to (b) or (c) is yes, then no change is required to the MQSC scripts. Execute the `CRTMQCQ` command below **with** the `PUBQ` event publisher option.

Create the Nastel Navigator administrative local and alias queues for the agent (option `MQCQ`), message server (option `MMFQ`) and, optionally based on your decisions above, for the event publisher (option `PUBQ`):

```
NASTEL/CRTMQCQ QMNAME(qmgr_name) MQCQ(*YES) MMFQ(*YES)
[PUBQ(*YES)]
```



If the IBM MQ event queues have been previously defined as alias queues, you may receive error messages when the script is executing. You may safely ignore these messages.

13. Verify Nastel Navigator queues were created:

MQ Agent:

Model Queue:

`NASTEL.MODEL.QUEUE` (default persistence, `DEFPSIST`, must be set to `NO`).

Local Queue:

`NASTEL.EVENT.QUEUE` (required only if Event Publish-Subscriber is not used).

Alias Queues:

13. `SYSTEM.ADMIN.QMGR.EVENT` (base queue name, `TARGQ`, must be set to `NASTEL.EVENT.QUEUE`)
14. `SYSTEM.ADMIN.CHANNEL.EVENT` (base queue name, `TARGQ`, must be set to `NASTEL.EVENT.QUEUE`)
15. `SYSTEM.ADMIN.PERFM.EVENT` (base queue name, `TARGQ`, must be set to `NASTEL.EVENT.QUEUE`)

Nastel Navigator Publish-Subscriber

Local Queue:

NASTEL.PUBSUB.EVENT.QUEUE

Alias Queues:

SYSTEM.ADMIN.QMGR.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)

SYSTEM.ADMIN.CHANNEL.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)

SYSTEM.ADMIN.PERFM.EVENT (base queue name, TARGQ, must be set to NASTEL.PUBSUB.EVENT.QUEUE)

Nastel Navigator Message Server

Local Queues:

NASTEL.MMF.ADMIN.COMMAND.QUEUE

NASTEL.MMF.AUDIT.QUEUE (required only if Nastel Navigator Message Server auditing is needed).

Verify that a NASTEL directory was created under IFS and that it contains the following:

```
/nastel
/nastel/apwmq
/nastel/apwmq/config
/nastel/apwmq/config/groups
/nastel/apwmq/config/groups/nsqrc.ini
/nastel/apwmq/mqsc
```

The Nastel directory is a part of the installation and cannot be deleted or moved to another location.

6.4.2 Upgrade Installation of Nastel Navigator

Follow these instructions if an upgrade of the Nastel Navigator agent is being installed on a system where an earlier version of Nastel Navigator is already installed.

1. Stop the APWMQ active jobs. (Refer to sections [6.7](#), [6.9](#), and [6.11](#).)
2. Backup the NASTEL library to a backup library (e.g., NASTELBK) if it is being used to run the MQ agent. This is necessary to preserve the NASTEL library contents because the upgrade installs to that same library.
3. Install the upgraded agent as if a new installation is being done. (Refer to [section 6.4.1](#).) It will install to library NASTEL.
4. Copy the original configuration members from file INI(MQGROUP and PERMITS) in the backup library to the same file\members in the <new_install_lib>.

5. Re-start/re-submit the APWMQ jobs. Refer to sections [6.6](#), [6.8](#), and [6.10](#) for the SBMJOB commands.

6.5 Post-Installation Tasks

6.5.1 Define Environment Variable

Before running individual Nastel Navigator components, you need to define a few environment variables.

1. **APWMQ_HOME** Required. This defines the fully qualified NASTEL installation or home library.

Example:

```
ADDENVVAR ENVVAR (APWMQ_HOME) VALUE ('/QSYS.LIB/NAstel.LIB') LEVEL (*SYS)
ADDENVVAR ENVVAR (APWMQ_HOME) VALUE ('/QSYS.LIB/NAstel.LIB') LEVEL (*JOB)
```

2. **APWMQ_ADP** Optional. Define it if running the agent with option +u, authorization on, and an authorization audit trail is desired. (Refer to Table 6-2 below.) It should point to a directory created under Integrated File System (IFS).

Example:

```
ADDENVVAR ENVVAR (APWMQ_ADP) VALUE ('/nastel/apwmq[/audit]') LEVEL (*SYS)
ADDENVVAR ENVVAR (APWMQ_ADP) VALUE ('/nastel/apwmq[/audit]') LEVEL (*JOB)
```

The IFS directory **/nastel/apwmq** already exists. If you want to use a separate 'audit' directory, create it.

3. **APWMQ_MQS_HOME** Optional. Define it if you want to monitor/manage a subset of the installed queue managers. It should point to an IFS directory, such as **/nastel/apwmq/config**, where an alternate mqs.ini file is located.

For example, see below [Managing a Subset of Queue Managers](#), step 2.

The file has a format similar to the actual mqs.ini file, with one entry for each queue manager to be managed.

Example:

```
QueueManager:
Name=QM1
QueueManager:
Name=QM2
```

4. **QIBM_MULTI_THREADED** (Start multi-thread capable processes) Required. It must be defined with value "Y" to allow child processes started by the Q Shell (QSH) to create multiple threads. The agent uses QSH to run runmqsc, a multi-threaded application, to process MQSC commands.

You can define or verify the environment variables using the command WRKENVVAR. Use the F4 key to be prompted for the *JOB or *SYS levels.

6.5.2 Managing a Subset of Queue Managers

A method is provided to allow managing a subset of the queue managers on your system, using an "alternate mqs.ini" file. Do the following before starting any APMQ jobs.

1. Choose an IFS directory where the alternate mqs.ini file will be located, such as **/nastel/apwmq/config**.
2. Make a copy of the real mqs.ini file (found typically in directory **/QIBM/UserData/mqm**) into the 'config' directory. Edit it by removing the QmgrName stanzas for all queue managers that you do not want to manage. Example to manage ISERIES.QM2, one of two existing queue managers:

```
AllQueueManagers:
#*****#
#* The path to the qmgrs directory, below which queue manager data *#
#* is stored *#
#*****#

DefaultPrefix=/QIBM/UserData/mqm

LogDefaults:
LogReceiverSize=100000

QueueManager: <=== remove this stanza
Name=ISERIES.QM1
Prefix=/QIBM/UserData/mqm
Library=QMISERIES.
Directory=ISERIES.QM1

QueueManager: <=== keep this stanza
Name=ISERIES.QM2
Prefix=/QIBM/UserData/mqm
Library=QMISERIES0
Directory=ISERIES.QM2
DataPath=/nastel/QIBM/UserData/mqm/qmgrs/ISERIES.QM2
```

3. Using command ADDENVVAR or WRKENVAR, define a *JOB level environment variable APWMQ_MQS_HOME=/nastel/apwmq/config

6.6 Starting the IBM MQ Agent

This section explains how to start the IBM MQ Agent. Before starting IBM MQ Agent, verify that IBM MQ command server is running, so that the IBM MQ Agent can discover IBM MQ objects.

1. Log on as a user with *ALLOBJ authority.
2. Start your workgroup server (usually on a Windows or a UNIX machine) if it is not already running.
3. Manually (step 3.a) or Automatically (step 3.b) register MQ node as follows:
 - a. Manually create and register the MQ node. Select **Create > Node...** from the workgroup server's action menu within the *Workgroup servers* viewlet. Fill out the dialog and click **OK**.

- b. Automatically register MQ node. To register OS/400 MQ node automatically, edit MQGROUP member of NASTEL/INI file. Use the edit file command to edit the database file:

```
EDTF FILE (NASTEL/INI) MBR (MQGROUP)
```

By default, the member looks like this:

```
Group::MQEVENT=MQM
Group::MQM=MQM
MQM::Node=localhost
MQM::Service=4010
```

- i. Ignore the first line – `Group::MQEVENT=MQM`; only applicable to Windows platforms.
- ii. Replace “MQM” with your workgroup server name. Do this only if you defined a workgroup server name other than the default “MQM” name at the time of configuration.
- iii. Change the node name “localhost” to the name or IP address of the node where the workgroup server is installed. You can use a node name if a DNS (domain name server) or the host file is accessible to resolve node name to IP address.
- iv. Change the port number 4010 to match the TCP listening port number used by the workgroup server. Do this only if you defined a port number other than the default value 4010 when you configured the workgroup server.
- v. For each additional workgroup server to which the IBM MQ Agent must register, define the group name, the node name, the TCP listening port of the workgroup server, and the security enablement, as in the example below:

```
Group::ACCT=ACCT
ACCT::Node=Pluto
ACCT::Service=4010
```

where ACCT is your new workgroup server and Pluto is your node name.

4. Make sure that the number of allowed jobs in the subsystem where Nastel agents expected to run allow to continually run for at least $3+4*QMGR_NUM$ jobs ($QMGR_NUM$ – number of queue managers that require monitoring). This formula is based on requirement to run IBM MQ Agent, Nastel Navigator Event Publish-Subscriber, and Nastel Navigator Message Server.
5. Start the IBM MQ Agent by issuing the command:

```
SBMJOB CMD(CALL PGM(NASTEL/NSQMQ) PARM(['-sPort_Number']
['-mGroup_Name'] ['-u'])) [JOB(Job_Name)] CPYENVVAR(*YES)
ALWMLTTHD*(YES)
```

where valid command-line arguments are described in Table 6-2.

6. The job can be submitted to any desired job queue and subsystem. Specify correct parameters for SBMJOB call.

Table 6-2. Command-line Arguments for Starting the IBM MQ Agent

Option	Description
<code>-mGroup_Name</code>	Specifies the name of the workgroup server to which the IBM MQ Agent should register (optional). Deleting this parameter or using the wildcard character (*) allows for registration with all workgroup servers that are manually configured to connect to this IBM MQ Agent, or those workgroup servers specified in the INI(MQGROU)file. Using a specific workgroup server name will allow registration only with that one workgroup server (remember that MQM is the default workgroup server name when Nastel Navigator is installed).
<code>-sPort_Number</code>	Specifies the TCP/IP port number that the IBM MQ Agent can bind to. Default: 5010.
<code>+u/-u</code>	Enables/disables authorization checking (optional). When disabled (-u), all commands submitted by any Nastel Navigator user will be executed on the queue manager as if the user had full *ALLOBJ authority. With authorization checking on (+u), the Nastel Navigator User ID will be checked against the authorities in the user profile. Default value is authorization checking off (-u)
<code>-t</code>	Enables trace for PCF commands and responses (optional). Default: trace off (no argument).
<code>job_name</code>	The job name of the Nastel Navigator process (optional). For example, NSQMQ.
<code>-nAliasName</code>	Assigns a new name to be used when registering with workgroup servers. This name will override the name of the local host.
<code>-e</code>	Forces the IBM MQ Agent to bind to a requested port specified in the -s parameter.
<code>-rNumber</code>	Port bind retry count. Default: 1



Single quotation marks (') must be used when giving arguments. Also, a space is required between arguments.

- Verify that the job was submitted successfully by issuing the command: `wrkactjob`
If system error RC 3429 or RC 3448 are received, ensure the TCP Interface is active.

6.7 Stopping the IBM MQ Agent

This section explains how to stop the IBM MQ Agent.

CALL PGM(NASTEL/NSQSTOP) PARM ([*options*]) CPYENVVAR(*YES)

where [*options*] are command-line arguments described in Table 6-3.

Example: CALL PGM(NASTEL/NSQSTOP) PARM ('-mMQM' '-nOS400') CPYENVVAR(*YES)

Table 6-3. Command-line Arguments for Stopping the IBM MQ Agent

Option	Description
<code>-mGroupName</code>	The name of the workgroup server under which the IBM MQ Agent resides.
<code>-nNodeName</code>	The name of the IBM MQ Agent to be shut down
<code>CPYENVVAR (*YES)</code>	Copy the environmental variables, of which there is only APWMQ_HOME
Or (without connecting to the Workgroup)	
<code>-nNodeName</code>	The name of the IBM MQ Agent to be shut down
<code>-iIPAddress</code>	MQ Agent IP address. Def: 127.0.0.1
<code>-sIPPort</code>	MQ Agent listening port number. Def: 5010



If you terminate the IBM MQ Agent manually for any reason, you should do the following: end the job using *IMMEDIATE modes, delete its spool files, and remove job from job queue. If you just end job and resubmit it, it may not start and error message "IME_0001 Failed to create TCP/IP server: RC (3420)" will appear in the spool file. This is an indication that IBM MQ Agent listening port is in use and cannot be reused.

6.8 Starting the Message Server for IBM MQ

This section explains how to start the Message Server for IBM MQ.

1. Log on as a user with *ALLOBJ authority.
2. Enter the command:

```
SBMJOB CMD(CALL PGM(NASTEL/NSQMSG) PARM ('-mqmgr_name' ['option'
'option' ]...) [JOB(job_name)]) CPYENVVAR(*YES)ALWMLTTHD(*YES)
```

where the command-line arguments are listed in Table 6-4:

Table 6-4. Command-line Arguments for Starting the Message Server for IBM MQ

Option	Description
<code>-mqmgr_name</code>	The queue manager whose local queues are to be accessed
Optional Arguments	
<code>job_name</code>	The job name of the Message Server for IBM MQ, for example, NSQMSG1.
<code>-r</code>	Queue manager disconnect retries
<code>-g</code>	Time to wait between reconnect tries, in seconds
<code>-a</code>	Enable command auditing
<code>-b</code>	Enable binary representation of message body
<code>-c</code>	Enable conversion on MQGET
<code>-t</code>	Turns on the trace option, for debugging purposes. The trace captures details about each request to the Message Server for IBM MQ, including the finite state machine (FSM) state name, the request received, the

Table 6-4. Command-line Arguments for Starting the Message Server for IBM MQ

Option	Description
	action taken, and the rejection reason code (if any). This parameter is optional; the default value is trace off (no argument).



Each parameter must be surrounded by single quotes.

6.9 Stopping the Message Server for IBM MQ

To stop the Message Server for IBM MQ, type the command:

```
CALL PGM(NASTEL/ENDMCS) PARM ('QmgrName'
['CommandQName'])
```

where *QmgrName* is the name of the queue manager on which the Message Server for IBM MQ is running and *CommandQName* is the name of the command queue. By default, the name of the command queue is NASTEL.MMF.MSG.COMMAND.QUEUE.

6.10 Starting the Publish Subscriber for IBM MQ

This section explains how to start the Publish Subscriber for IBM MQ. Note that before you start the Publish Subscriber, you should verify its administrative queues, as described below. Using the DSPMQMQ command, verify that the following alias queues have been created:

```
DSPMQMQ QNAME(SYSTEM.ADMIN.QMGR.EVENT)
DSPMQMQ QNAME(SYSTEM.ADMIN.CHANNEL.EVENT)
DSPMQMQ QNAME(SYSTEM.ADMIN.PERFM.EVENT)
```

These alias queues should point to NASTEL.PUBSUB.EVENT.QUEUE.

Start the IBM MQ Event Publisher. Log on as QPGMR, QSYSOPR, or as a user with *ALLOBJ authority.

Enter the command:

```
SBMJOB CMD(CALL PGM(NASTEL/NSQPUB) [PARM ('-xvalue'...)])
[JOB(pub_job_name)] CPYENVVAR(*YES) ALWMLTTHD(*YES)
```

where command-line arguments for starting the Publish Subscriber for IBM MQ are described in Table 6-5.

Table 6-5. Command-line Arguments for Stopping the Publish Subscriber for IBM MQ

Option	Description
PARM <i>-xvalue</i>	Stands for one or more optional parameters, which are listed in Appendix B . <i>-xvalue</i> is the option letter and value of the selected parameter.

Table 6-5. Command-line Arguments for Stopping the Publish Subscriber for IBM MQ

Option	Description
<i>pub_job_name</i>	The job name of the Publish Subscriber for IBM MQ (optional). For example, NSQPUB.



Each parameter must be surrounded by single quotes.

For the easiest startup of the Publish Subscriber for IBM MQ, no arguments are necessary.

6.11 Stopping the Publish Subscriber for IBM MQ

To stop the Publish Subscriber for IBM MQ, type the command:

```
CALL PGM(NASTEL/PUBEND) PARM ([options])
```

where *[options]* are command-line arguments summarized in [Appendix B](#).

6.12 Nastel Navigator Utilities on OS/400

The following utilities are available on OS/400:

Table 6-6. Command-line Arguments for Stopping Utilities on OS/400

Option	Description
nsqcl	Client program used to debug PCF commands
nsqstop	Stops Workgroup server or IBM MQ Agent
endmcs	Stops Message Server for IBM MQ
pubend	Stops the Publish Subscriber for IBM MQ
nsqcode	Reason code lookup utility
nsqmsgf	Message file utility

6.12.1 Ping Utility (nsqping)

Use the ping utility to verify performance of IBM MQ.

The format of the nsqping command is:

```
SBMJOB CMD(CALL PGM(NASTEL/NSQPING) PARM (options) [JOB(msg_job_name)])
CPYENVVAR(*YES) ALWMLTTHD(*YES)
```

where *(options)* are command-line arguments listed in Table 6-7:



When nsqping runs, it will publish the ping data as facts to AutoPilot if options `-h` and `-p` are provided. You should have an AutoPilot process wrapper configured and deployed to receive these facts. If it is not deployed, an error message will appear near the end of the nsqping output:

```
ERROR: failed to write to host(service)=localhost(6000), sock type 2, RC(3427)
```

where 3427 is an errno (error number) meaning “[EDESTADDRREQ] Operation requires destination address. A destination address has not been associated with the socket.”

Table 6-7. Command-line Arguments for the Ping Utility

Argument	Description	Default
Required Arguments		
-mQMgrName	Name of the local queue manager	N/A
-qQueueName	Queue name of any type	N/A
Optional Arguments		
-dInterval	Generate ping every interval seconds (0=once)	N/A
-hHost	Host name of the target AutoPilot host process wrapper	localhost
-pPort	Port number of the target AutoPilot Process Wrapper	6000
-bBatch	Number of messages in a batch	10
-sBytes	Generate messages of Number bytes	512
-wWait	Wait interval, in milliseconds, before timeout	60000
-eExpiry	Messages expire after Expiry time	Unlimited
-uAltUser	Issue ping on behalf of another user	1/10 sec
-ccoA	Generate confirm-on-arrival report	False
-ccod	Generate confirm-on-delivery report	False
-cexp	Generate confirm-on-expiry report	False
-cexc	Generate confirm-on-exception report	False
-cdlq	Generate messages that will go onto DLQ if undelivered	False
-v0 1 2	Set verbose mode to one of the following: 0 – summary; 1 – reports; 2 – all.	0

6.12.2 Message File (nsqmsgf) Utility

The message file utility is used to dump messages from queue to file and vice-versa.

The format of the nsqmsgf command is:

```
SBMJOB CMD(CALL PGM(NASTEL/NSQMSGF) PARM (options)
[JOB(msg_job_name)]) CPYENVVAR(*YES) ALWMLTTHD(*YES)
```

Enter nsqmsgf.exe ? on a command line to display all options:

```
E:\nastel\apwmq\bin>nsqmsgf.exe ?
```

(C)Copyright 1995 - 2022, Nastel Technologies Inc.
 Reproduction, adaptation, or distribution without prior
 written permission is prohibited, except as allowed under
 the copyright laws.

NSQMSGF Version: 6.7.0004

Options to GET Messages from Queue to File:

```
-m<Queue manager name>
-q<Queue name>
-g<0,1> -- 0=Browse, 1=Destructive get. Default: browse
[-w<Wait interval, secs>] -- MQGET wait time. Default: 1 sec
[-f<File name>] -- File to save the messages. Default: queue_name.msg
[-s<1,2>] -- Strip WMQ headers, 1=MD, 2=ALL. Default: none
[-d<1,2>] -- Message separator, 1=NL, 2=@@. Default: 2
[-b<Message cursor>] -- value 1 - 999999999. Default: 1
[-n<Message count>] -- value 1 - 999999999. Default: -1 (all)
[-c] -- Enable message conversion. Default: disabled
    The message data will be converted by the queue
    manager for built-in formats when the message is
    read, and the MQMD will be converted to the local
    queue manager (native) integer encoding.
[-r] -- Generate reply. Default: none
```

To filter messages by MQMD fields use the following syntax:

To filter messages by the name of application that put the message:
 PutAppName <Operator> <Value>

To filter messages by the date and time when the message was put:
 PutTimestamp <Operator> <Value>

To filter messages between two timestamps:
 PutTimestamp <Operator> <Value> PutTimestamp <Operator> <Value>

The operators are:

```
LT - Less than
GT - Greater than
EQ - Equal to
NE - Not equal to
LE - Less than or equal to
GE - Greater than or equal to
CN - Contains a specified substring
EX - Does not contain a specified substring
```

Value:

The format used for the timestamp: YYYYMMDDhhmssth

where the characters represent:

```
YYYY - year (four numeric digits)
MM - month of year (01 through 12)
DD - day of month (01 through 31)
hh - hours (00 through 23)
```

```
mm - minutes (00 through 59)
ss - seconds (00 through 59)
t - tenths of a second (0 through 9)
h - hundredths of a second (0 through 9).
Greenwich Mean Time (GMT) is used for the PutTimestamp fields
```

Options to PUT Messages from File to Queue:

```
-m<Queue manager name>
-q<Queue name>
-p
[-f<File name>]
[-i<1,2,3>] -- import file type:
    i1 - Import file as 1 message;
    i2 - Import text file of 1 line text messages as multiple messages;
    i3 - Import XML file as multiple messages.
[-h] -- Generate header (Message Descriptor) with user, originator, and
identity
        context from original message, else a new MD is written.
[-d<1,2>] -- Message separator, 1=NL, 2=@@ ; default: 2
[-c] -- Enable message conversion. Default: disabled
        The MQMD from the message file will be converted
        to the local queue manager (native) integer encoding
        before the message is put to the queue, if the MQMD
        encoding is not the native encoding.
[-o] -- Generate message with zero-length application data
[-U<UserName>] -- User ID to be used in queue manager authentication
[-P<Password>] -- Password to be used in queue manager authentication
```

The command-line options and parameters in square brackets are optional; all others are required.



Either `-g` or `-p` parameter can be present. Default is `-g` (get message).

6.12.3 Client Test Utility (nsqcl)

Enter on command line:

```
call pgm(nsqcl)
```

If you are having problems with a WMQ Agent, it may be malfunctioning, or it may lack sufficient privileges to carry out the tasks you are requesting.

The nsqcl utility is used to verify operation of the IBM MQ Agent. It prompts for the location of the IBM MQ Agent; host name, service (port) and mode of connection (secure or non-secure). Use secure mode only if the workgroup server is running in secure mode. The output of nsqcl command is in PCF format.

6.12.3.1 Verifying WMQ Agent Operation

1. Use the nsqcl utility program. This utility prompts you for the location of the M6-WMQ Local Agent—enter the host name and port number. Use conversational mode only. The utility prompts you for the Application Name—enter a random name if the host is running WebSphere MQ Level 2 systems. On WebSphere MQ Level 1 systems the Application Name should correspond to any available queue defined on the queue manager. The output of nsqcl command is in PCF format.
2. Once the connection has been established, the utility prompts you for PCF commands. Before entering any PCF commands, do the following (example is for Windows environments):

```
C:\>nsqcl USER_NAME("john")
nsqcl V6.7.0004 (Apr 28 2021)
```

```
(C)Copyright 1995 - 2021, Nastel Technologies Inc.
Reproduction, adaptation, or distribution without prior
written permission is prohibited, except as allowed under
the copyright laws.
```

```
Sizeof MSHDR = 136 Manager =
Sysname("Windows_NT")
NodeName("NEPTUNE-PC")
Release("V6.7.0004 (Apr 28 2021)")
User ID("jennifer")
```

```
Enter host name (localhost):
```

```
Enter service name (5010): 5020
```

```
Mode of connection ((S)ecure/(N)Secure) (N):
```

```
Connection has been established ok!
```

```
Enter Command Name:
```

To see the available PCF commands, enter "help":

```
Enter Command Name: help
```

```
Supported PCF commands:
```

```
EXCMD_MQ_OPEN
EXCMD_OPEN_EVENT
EXCMD_START_Q_MGR
EXCMD_STOP_Q_MGR
EXCMD_START_MCA
EXCMD_STOP_MCA
EXCMD_INQUIRE_Q_MGR_NAMES
EXCMD_INQUIRE_MANAGER_NAMES
EXCMD_INQUIRE_ACTIVE_MANAGER
etc.
```

3. Issue EXCMD_INQUIRE_Q_MGR_NAMES command to test the discovery of queue managers on the host system, both active and inactive. For the object name, enter an asterisk (*). A list of all the queue managers is returned. For example:

```
Enter Command Name: EXCMD_INQUIRE_Q_MGR_NAMES
```

```
Enter Object Name: *
```

```
Command (EXCMD_INQUIRE_Q_MGR_NAMES) has been issued
PCF RESPONSE FROM THE SERVER -->
```

```
PCF Header (MQCFH):
```

```
->Type(2), StrucLength(36), Version(1), Command(10007)
->MsgSeqNum(1), Control(1), ParameterCount(3)
->CompCode(0), Reason(0)
```

```
PCF String List (MQCFSL):
```

```
->Type(6), StrucLength(120), Parameter (2015), Count(2)
->StringLength(48), CharSetId(0)
->String List
{
    (1) - "NAS_QM
"
    (2) - "NASOAM_QM
"
```

```
};
```

```
PCF String (MQCFST):
```

```
->Type(4), StrucLength(24), Parameter(2015), CharSet(0)
```

```
->StringLength(1), String("*")
```

```
PCF Integer (MQCFIN):
```

```
->Type(3), StrucLength(16), Parameter (20119), Value(7)
```

```
THIS RECORD (LAST)
```

4. Issue EXCMD_MQ_OPEN command. For the object-name, enter the name of the remote queue manager whose operations you would like to test. This command opens a connection to the local queue manager on the remote machine. For example:

```
Enter Command Name: EXCMD_MQ_OPEN
```

```
Enter Object Name: NASOAM_QM
```

```
Command (EXCMD_MQ_OPEN) has been issued
```

```
PCF RESPONSE FROM THE SERVER -->
```

```
PCF Header (MQCFH):
```

```
->Type(2), StrucLength(36), Version(1), Command(10008)
```

```
->MsgSeqNum(1), Control(1), ParameterCount(2)
```

```
->CompCode(0), Reason(0)
```

```
PCF String (MQCFST):
```

```
->Type(4), StrucLength(32), Parameter(2015), CharSet(0)
```

```
->StringLength(9), String("NASOAM_QM")
```

```
PCF Integer (MQCFIN):
```

```
->Type(3), StrucLength(16), Parameter (20119), Value(7)
```

```
THIS RECORD (LAST)
```

5. Issue the EXCMD_OPEN_EVENT command that enables the WMQ Agent to attach to its event queue and send events to the nsqcl utility program. For example:

```
Enter Command Name: EXCMD_OPEN_EVENT
```

```
Enter Object Name: NASTEL.EVENT.QUEUE
```

6. Ensure all the above commands succeed. Commands were successful if the CompCode and the Reason codes of all responses are zero. If the CompCode response is warning

(1) or failure (2), write down the CompCode and Reason values and consult IBM's PCF Reference for details on the command and the reason of the failure.

7. Issue any of the MQCMD_XXX commands described in the IBM's PCF Reference manual to verify operation of any IBM-supported PCF command. For example:

```
Enter Command Name: MQCMD_INQUIRE_Q_NAMES
Enter Object Name: MQ*
Command (MQCMD_INQUIRE_Q_NAMES) has been issued
PCF RESPONSE FROM THE SERVER -->
```

PCF Header (MQCFH):

```
->Type(2), StrucLength(36), Version(1), Command(18)
->MsgSeqNum(1), Control(1), ParameterCount(5)
->CompCode(0), Reason(0)
```

PCF String List (MQCFSL):

```
->Type(6), StrucLength(216), Parameter (3011), Count(4)
->StringLength(48), CharSetId(0)
->String List
{
    (1) - "NASTEL.MMF.AUDIT.QUEUE"
    (2) - "NASTEL.MMF.ADMIN.COMMAND.QUEUE"
    (3) - "NASTEL.REPLY.xxx1999031119025854"
    (4) - "NASTEL.EVENT.QUEUE"
};
```

PCF Integer (MQCFIN):

```
->Type(3), StrucLength(16), Parameter (20126), Value(10)
```

PCF String (MQCFST):

```
->Type(4), StrucLength(24), Parameter(2016), CharSet(0)
->StringLength(3), String("MQ*")
```

PCF Integer (MQCFIN):

```
->Type(3), StrucLength(16), Parameter (20), Value(1001)
```

PCF Integer (MQCFIN):

```
->Type(3), StrucLength(16), Parameter (20119), Value(7)
```

THIS RECORD (LAST)



Some commands will require attributes that NSQCL does not supply, and the command will fail.

8. When a WMQ Agent is functioning properly, all PCF commands should succeed with reason code equal to zero. Note that this rule does not apply to PCF commands that *start/stop* channels or other components, since their failure may be attributed to the WebSphere MQ configuration and authorization.

6.12.4 Reason Code Lookup Utility (nsqcode)

Enter on command line:

```
call pgm(nsqcode) parm('reason_code')
```

Alternatively, submit the command as a job and look for the output in the spool file:

```
SBMJOB CMD(CALL PGM(NASTEL/NSQCODE) PARM('reason_code')
[JOB(NSQCODE)] ) CPYENVVAR(*YES) ALWMLTTHD(*YES)
```

The nsqcode utility will display a description of IBM MQ and Nastel Navigator reason codes.

6.13 Getting a Spool File from the AS/400

To convert an APWMQ program's log/trace from a spool file to a format for transmitting to Nastel, follow this procedure:

1. On the command line type the Work with Spool Files command, WRKSPLF and find the spool file of interest.
2. Enter option **9** (Work with Printing Status) to display the job information and record the values for **Job**, **User**, and **Number**.

Example:

Work with Printing Status	
Job : NSQMSG	File : QPRINT
User : QMQM	Number : 1
Number : 261850	User data :
Job system name . . : S106EF2B	Creation date . . . : 10182011
	Creation time . . . : 214615

3. Create a physical file in a library for copying the spool file contents. Command line example:

```
CRTPF FILE(NASTEMP/SPLFILES) RCDLEN(120) MBR(*NONE) TEXT('Spool files for xfer to Nastel')
```

Physical file SPLFILES is created in library NASTEMP and allows for members with up to 120 characters per record.

4. Using the copy spool file command (CPYSPLF), copy the spool file to a member of the physical file using the QPRINT job information from step 2.

Command line example:

```
CPYSPLF FILE(QPRINT) TOFILE(NASTEMP/SPLFILES) JOB(261850/QMQM/NSQMSG)
```

System response:

Member MSGSRVTRC added to file SPLFILES in NASTEMP.

Or you can use **F4=Prompt** to display the following screen, enter the QPRINT job information, and press **Enter**.

```

Copy Spooled File (CPYSPLF)

Type choices, press Enter.

Spooled file ..... QPRINT Name
To data base file ..... SPLFILES Name
Library ..... NASTEMP Name, *LIBL, *CURLIB
Job name ..... NSQMSG Name, *
User ..... QMQM Name
Number ..... 261850 000000-999999
Spooled file number ..... 1 1-999999, *ONLY, *LAST, *ANY
Job system name ..... *ONLY Name, *ONLY, *CURRENT, *ANY
Spooled file created:
Creation date ..... *ONLY Date, *ONLY, *LAST
Creation time ..... Time, *ONLY, *LAST
To member ..... MSGSRVTRC Name, *FIRST
Replace or add records .... *REPLACE *REPLACE, *ADD
    
```

5. Verify the copied spool file by viewing it with command WRKMBRPDM.

Command line example:

```
WRKMBRPDM nastemp/splfiles
```

Enter option 5 (Display) next to the member you copied.

6. FTP in ASCII the spool file member of the physical file to your PC. Then e-mail it to support@nastel.com or attach it to your problem report.

6.14 Uninstalling Nastel Navigator on OS/400

Uninstall Nastel Navigator on OS/400:

1. Remove Nastel Navigator queues and restore the SYSTEM event local queues by typing the command:

```
RMVMQCQ QMNAME(qmgr) IFSOBJ(*NO)
```

IFSOBJ: YES or NO option to remove all IFS objects created by the Nastel Navigator installation. It includes Unix-like directories and files.

You can enter RMVMQCQ followed by pressing F4 to open screen form for the command. Press F1 on screen title or any fields for help.

2. Remove NASTEL library.

Chapter 7: HP OpenVMS Installation

This chapter explains how to install Nastel Navigator on HP Open VMS.



Nastel Navigator for OpenVMS can be installed only on Alpha machines.

7.1 Nastel Navigator Components for OpenVMS

The Nastel Navigator installation for OpenVMS installs the components listed in table 7-1.

Table 7-1. Nastel Navigator Components for OpenVMS	
Programs	Description
Main Programs	
MQ Agent (the “agent”)	The Nastel Navigator interface to the queue manager. Submits commands to and receives responses from the IBM MQ command server.
WMQ Event Publisher	Distributes IBM MQ events to programs that subscribe for them (e.g., the IBM MQ agent).
MMF (Message Management Facility) Server	Services requests from the MMF Explorer for reading, writing, moving, or deleting messages on queues.
Utility Programs	
nsqcode	Translates reason code to text and description.
nsqcl	Client program to test the IBM MQ agent.
nsqmqsc	MQSC console: Equivalent of runmqsc to execute MQSC commands; works across the IBM MQ network using the workgroup server.
nsqping	Measures ping response time of a queue manager; works across the IBM MQ network.
nsqmmfcl	MMF command line program; has much of the functionality of the MMF Explorer.
nsqmsgf	MMF command line utility to do put (get) messages to (from) queues using message files.

7.2 System Requirements

An APWMQ v6.x or later installation for VMS requires an HP OpenVMS Alpha machine with the operating system configurations listed in table 7-2.

Table 7-2. Operating System Configuration for VMS Version of Nastel Navigator	
Version of VMS	Run-time Library
Open VMS Alpha 8.2 and later	C++ library LIBCXXXSTD.OLB 7.1

A Nastel Navigator installation for OpenVMS also requires:

- 4 MB of memory or more
- 5 MB of disk space or more
- DEC UCX TCP/IP services OpenVMS 3.3 or later
- IBM MQ for HP OpenVMS 6.0.1.2 or later.

7.3 Before You Install

- Before you install Nastel Navigator, you need to copy the Nastel Navigator installation files to your VMS machine.
- When installing and configuring Nastel Navigator for VMS, you should be logged on as the user `SYSTEM`.

7.4 Upgrading Nastel Navigator on VMS

Nastel Navigator for VMS is installed from a save set file on a CD or downloaded from the Nastel FTP site or Resource Center. You need to load the installation files to a Windows machine first before sending them via FTP to your VMS machine.

To upgrade an existing Nastel Navigator installation



This section does not apply to first-time installation of Nastel Navigator.

When migrating from an earlier version of Nastel Navigator to a more recent one, you will want to preserve the existing configuration information included in the Nastel Navigator install directory, typically `SYS$COMMON:[NASTEL]`. This way you can reapply your original configuration to your new Nastel Navigator installation or, if needed, you can revert to the previous version of Nastel Navigator.

To upgrade Nastel Navigator

1. Create the directory `SYS$COMMON:[NASTEL_BACKUP]`:

```
create/dir SYS$COMMON:[NASTEL_BACKUP]
```

2. Copy the contents of your existing Nastel Navigator installation directory to the backup directory:

```
copy SYS$COMMON:[NASTEL...] *.* SYS$COMMON:[NASTEL_BACKUP...] *.*
```

3. Install the new Nastel Navigator version, following the instructions in the [section 7.5](#).
4. Copy files `permits.ini` and `group.ini` from the backup directory :
`SYS$COMMON:[NASTEL_BACKUP.APWMQ.CONFIG]` to the install directory.

7.5 Installing Nastel Navigator

Nastel Navigator for VMS is installed from a save file on a CD or downloaded from the FTP site or Resource Center. You will need to load the installation files to a Windows machine first before sending them via FTP to your VMS machine.

To load Nastel Navigator installation files onto your VMS machine:

1. On your VMS machine, make sure you are logged on as the user `SYSTEM`.
2. Create a temporary directory called `NASTEL_TEMP`:

```
create/dir SYS$COMMON:[NASTEL_TEMP]
```

Note that `SYS$COMMON` is the default installation disk, but you may choose another disk.

3. On a Windows machine, load the VMS installation file from the Nastel Navigator installation source to a temporary folder, such as `VMS`. In the `VMS` folder you will have a zip file named

```
M6WMQ_ver_OpenVMS_8.2_Alpha.zip
```

where `ver` is the Nastel Navigator version number, such as 6.1.5.

4. Unzip the file. You should have the following two files:

```
M6WMQ_ver.BCK (Example: M6WMQ_6_1_5.BCK)  
INSTALL.COM
```

The `.BCK` file is a VMS save set.

5. FTP the installation files from the Windows machine to your VMS machine and load them into the temporary directory `NASTEL_TEMP` that you created in step 2. FTP the files in this format:

```
M6WMQ_ver.BCK in binary format
```

```
INSTALL.COM in ASCII format
```

6. In your temporary directory `NASTEL_TEMP` on the VMS machine, enter the following command:

```
set def SYS$COMMON:[NASTEL_TEMP]
```

7. To move the product files from the temporary directory to the installation directory, enter the command:

```
@INSTALL.com p1 [p2 [p3]]
```

where:

p1 is the version number, with underscores, of the product being installed (for example, 6_1_5). The script looks for a backup save set file named M6WMQ_version.BCK in the current directory.

p2 is the disk name where the product will be installed.

Default: SYS\$COMMON

p3 is the name of the directory (without the outer square brackets) where the product will be installed.

Example: PRODUCTS.MONITORING.NASTEL

Default: NASTEL

p1 is a mandatory parameter; *p2* and *p3* are optional.

Example: To install version 6.1.5 in the default install directory:

```
@INSTALL 6_1_5
```

The files will be loaded into directory SYS\$COMMON: [NASTEL.APWMQ]

7.6 Verifying Files and Directories

After you load the files, you need to verify that all files are loaded correctly and that all directories have been created.

To verify files and directories

1. Check your working directory by entering the command:

```
show def
```

Your current working directory should be:

```
SYS$COMMON: [NASTEL.APWMQ.BIN]
```

2. Check the contents of the APWMQ home directory by entering the command:

```
dir [-]
```

The contents of the directory [NASTEL.APWMQ} should be:

```
BIN.DIR
CONFIG.DIR
LOGS.DIR
MQSC.DIR
README.HTM
README.TXT
SCRIPTS.DIR
SYSVER.TXT
```

```
TOOLS.DIR  
major_minor_build.TXT
```

3. Check the contents of the BIN directory by entering the command:

```
dir [NASTEL.APWMQ.BIN]
```

The contents of the directory should be:

```
ENDMCS.EXE  
MQMCRT.COM  
MQMDLT.COM  
MQMSTART.COM  
MQMSTOP.COM  
NSQCL.EXE  
NSQCODE.EXE  
NSQDLQLSR.EXE  
NSQEVQLSR.EXE  
NSQMMFCL.EXE  
NSQMQ.COM  
NSQMQ.EXE  
NSQMQSC.EXE  
NSQMSG.COM  
NSQMSG.EXE  
NSQMSGF.EXE  
NSQPING.EXE  
NSQPUB.COM  
NSQPUB.EXE  
NSQQW.EXE  
NSQSTOP.EXE  
NSQTRIG.EXE  
PCFPARM.EXE  
PCFVAL.EXE  
PUBEND.EXE  
PUBSTOP.EXE  
RUN_NSQMQ.COM  
RUN_NSQMSG.COM  
RUN_NSQPUB.COM  
SET_APWMQ.COM  
UCX_CREATE_SERVICE.COM
```

4. Check the contents of the LOGS directory by entering the command:

```
dir [-.LOGS]
```

The directory should be empty.

5. Check the contents of the MQSC directory by entering the command:

```
dir [-.MQSC]
```

The contents should be:

```
DEFAULT.TST
NASTEL.TST
MMF.TST
MQMGM.TST
PUBSUB.TST
```

6. Check the contents of the SCRIPTS directory. The directory should be empty at installation time. Enter the command:

```
dir [-.SCRIPTS]
```

This directory is used to hold automation scripts named
<event_nbr>_<reason_code>

Example: 45_2225.COM

where:

45 is the value of the IBM MQ defined symbol MQCMD_PERFM_EVENT and 2225 is the value of the queue depth low reason code symbol MQRC_Q_DEPTH_LOW. See SYS\$COMMON:[NASTEL.APWMQ.TOOLS.SAMPLES] for the README.txt with details of using automation scripts. Custom scripts can be created using the three samples provided for queue depth high event and queue depth low event used to dynamically adjust the queue depth warning levels, and channel stop event used to restart a channel that stopped due to some error.

7. Check the contents of the TOOLS directory by entering the command:

```
dir [-.TOOLS ...]
```

This directory contains header, library, C program source files to allow you to build custom

Nastel Navigator tools, and three automation scripts.

7.7 Setting Nastel Navigator Logicals and Foreign Commands

As part of the post-installation tasks, you need to set Nastel Navigator logical and foreign commands.

To set Nastel Navigator logicals and foreign commands

1. Configure MQ agent options: The agent options can be changed by editing the logical definition of `APWMQ_LM_OPTIONS` in the file `SET_APWMQ.COM` or from the command prompt.

There are many available options, such as `+u/-u`, `-t`, `-x`. For a display and explanation of the agent startup options, execute `nsqmq -?`. Also see [Appendix B](#).

2. Enter the command:

```
@ [NASTEL.APWMQ.BIN] SET_APWMQ.COM
```

This command defines the logicals and aliases (foreign commands) for the programs used by Nastel Navigator for OpenVMS. If this is not an initial install of Nastel Navigator on your system, DCL information messages will appear, indicating that these logicals have been superseded.

7.8 Configuring the Agent

After you have downloaded the Nastel Navigator installation files into your VMS machine, you can proceed to configure the agent.

The following conditions must be met:

- The queue managers must be active
- The IBM MQ command server must be active
- The Nastel Navigator workgroup server must be active.

Configure the UCX nsqmq service

1. Configure UCX options. Edit the file:

```
APWMQ_BIN:UCX_CREATE_SERVICE.COM
```

Change the qualifier `/limit = 5` to `n`, the expected maximum number of parallel connections needed, where:

```
n = (# of workgroup servers and other clients that will connect to the agent on this
computer) x
      (# queue managers +1).
```

The `+1` is for workgroup server auto-discovery connections. Use `n=5` as a minimum.

2. If the UCX service `NSQMQ` is already defined, delete the UCX service for the agent, `nsqmq`. Enter the command:

```
UCX set noservice nsqmq
```

3. Define the NSQMQL service:

```
@APWMQ_BIN:UCX_CREATE_SERVICE.COM
```



You may get a warning indicating that the device is already active. This warning can be safely ignored.

4. You can edit the file `APWMQ_BIN:NSQMQL.COM` to keep the last 100 agent log files. Add the following two lines at the top of the file:

```
PURGE/KEEP=100 APWMQ_LOGS:nsqmq.log
PURGE/KEEP=100 SYS$SYSROOT:[SYSMGR]nsqmq.log
```

Create Nastel Navigator agent queues

1. If your company has queue naming conventions, edit the `[APWMQ_HOME]/mqsc/nastel.tst` MQSC script to customize the NASTEL queue names with the high and middle level queue name qualifiers that are (or will be) specified in the qualifier file `[APWMQ_HOME]/config/mqqual.ini`. Refer to [Appendix C](#) for a discussion of qualifiers and the file syntax.
2. To create the agent event-handling queues, enter the IBM MQ command:

```
RUNMQSC qmgr_name < APWMQ_MQSC:NASTEL.TST
```

where *qmgr_name* is the name of your queue manager. The command creates the following queues:

```
NASTEL.EVENT.QUEUE (local queue)
NAstel.MODEL.QUEUE (local queue)
SYSTEM.ADMIN.CHANNEL.EVENT (alias queue)
SYSTEM.ADMIN.CONFIG.EVENT (alias queue)
SYSTEM.ADMIN.PERFM.EVENT (alias queue)
SYSTEM.ADMIN.QMGR.EVENT (alias queue)
SYSTEM.ADMIN.ACCOUNTING.QUEUE (alias queue)
SYSTEM.ADMIN.STATISTICS.QUEUE (alias queue)
```

Note that `NASTEL.EVENT.QUEUE` is the target queue. The last six queues are alias queues pointing to the target.

Create Nastel Navigator queue manager name file

To determine the queue managers to be managed, the agent does not access the normal `MQS.INI` file in `MQS_ROOT:[mqm.mqs.ini]`. Instead, you must create an alternate `mqs.ini` that contains the names of the queue managers to be managed, which could be all or a subset of your queue managers. Create a file named `mqs.ini` in the directory defined by the logical `APWMQ_MQS_HOME`, for example `SYS$COMMON:[NAstel.APWMQ.CONFIG.GROUPS]`.

The entry format is similar to the actual `mqs.ini` file:

QueueManager:

Name=QM1

QueueManager:

Name=QM2

...

For more information, see [Managing a Subset of Queue Managers](#).

7.9 Verifying the Agent UCX Configuration

After you install the agent, you should verify that all files have been installed correctly.

To verify the agent UCX configuration

1. Enter the command

```
dir APWMQ_BIN
```

The following files should be listed:

```
MQMCRT.COM
```

```
MQMDLT.COM
```

```
MQMSTART.COM
```

```
MQMSTOP.COM
```

```
NSQMQ.COM
```

```
NSQMQ.EXE
```

2. Start the UCX utilities by entering the command:

```
UCX
```

The UCX> (or TCPIP>) prompt appears.

3. At the UCX prompt, enter the command:

```
show service NSQMQ/fu
```

The following response should appear:

Service	Port	Proto	Process	Address	State	User_Name
NSQMQ	5010	TCP	NSQMQ	0.0.0.0	Enabled	MQM

If you do not see USER_NAME=MQM, do the following:

```
UCX > disable service nsqmq  
set service nsqmq/user_name=MQM  
enable service nsqmq
```

4. To get out of the UCX utility, enter EXIT.

7.10 Registering the OpenVMS Node with a Workgroup Server

After you verify that the agent has been installed correctly, you are ready to register your OpenVMS node with a workgroup server.

To register an OpenVMS node

1. Manually create and register the MQ node by selecting **Create > Node...** from the workgroup server's action menu within the *Workgroup servers* viewlet.
2. In the *Node Properties* dialog box, type a node name of your choice, IP address of the OpenVMS machine, listener port number (by default 5010), and the platform type (VMS).
3. Click **OK**.
4. Right-click the newly created node and choose **Manage**. The icon should shortly appear green, and the OpenVMS node is registered with the workgroup server. If not, check the VMS system:
 - Does the `NSQMQ` process appear in the `show system` display?
 - Check the `APWMQ_LOGS:NSQMQ.LOG` for possible cause of problem.
 - Check permissions of scripts `NSQMQ.COM` and executable `NSQMQ.EXE` in the `BIN` subdirectory.
 - Did you create the `APWMQ_HOME:MQS.INI` file? Is the format correct?

7.11 Configuring the MMF Server

The MMF Server is the back-end component of the Nastel Navigator Message Format Facility. Together with the Message Explorer (the front-end component which runs on a Windows machine) they manage IBM MQ messages.

To configure the MMF Server

1. If your company has queue naming conventions, edit the `[APWMQ_HOME]/mqsc/mmf.tst` MQSC script to customize the NASTEL queue names with the high and middle level queue name qualifiers that are (or will be) specified in the qualifier file `[APWMQ_HOME]/config/mqqual.ini`. Refer to [Appendix C](#) for a discussion of qualifiers and the file syntax.

2. Enter the IBM MQ command:

```
RUNMQSC qmgr_name < APWMQ_MQSC:MMF.TST
```

where `qmgr_name` is the name of your IBM MQ queue manager.

The following message management queues are created:

```
NASTEL.ADMIN.MSG.COMMAND.QUEUE
```

```
NASTEL.ADMIN.AUDIT.QUEUE
```

- To run the MMF Server as an interactive program, skip to [section 7.12](#), Starting the MMF Server. To run the MMF Server as a detached (background) process, open and edit the file `APWMQ_BIN:RUN_NSQMSG.COM` and substitute the queue manager name where specified.
- Define (or redefine) the logical `APWMQ_MS_OPTIONS` to set the options for the MMF Server.

```
define/system APWMQ_MS_OPTIONS "-mqmgr_name [-rretry_count]
[-gretry_time] [-a] [-b] [-c] [-t]"
```

where:

qmgr_name is the name of your IBM MQ queue manager. Note that the surrounding double quotation marks ("..") are mandatory.

-rretry_count is the number of times the MMF Server tries to connect to a stopped or quiescent queue manager before exiting. This parameter is optional. The default value is 100.

-gretry_time is the time in seconds between each connection attempt. This parameter is optional. The default value is 60.

-a Enable command auditing. Default: disabled.

-b Enable binary representation of message body. Default: disabled.

-c Enable conversion on MQGET. Default: disabled.

-t Enable trace for debugging purposes. The trace captures details about each request to the MMF Server, including the finite state machine (FSM) state name, the request received, the action taken, and the rejection reason code (if any). This parameter is optional. The default value is trace off.



When it is running as a detached process, the MMF Server logs two messages for each connection attempt to the file `APWMQ_LOGS:NSQMSG.OUT`.

When it is running as an interactive program, it logs messages to the output device `SY$OUTPUT` (`STDOUT`), typically your terminal.

- For each additional queue manager, create a file `APWMQ_BIN:RUN_NSQMSGi.COM`, where *i*=1, 2, 3 ... Edit them to use the new queue manager name, `/OUT=NSQMSGi.OUT`, `/PROCESS=NSQMSGi`, in place of the generic names. For the command, process, and log file names, you can try to include an abbreviation of the related queue manager name for easier identification.

7.12 Starting the MMF Server

You can start the MMF Server either as a detached process or as an interactive program.

To start the MMF Server

- To start the MMF Server as a detached process, provided you have set up the files to run as such (see previous section), enter the command:

```
@APWMQ_BIN:RUN_NSQMSG.COM
```

Repeat, if required, for `RUN_NSQMSGi.COM` where $i=1, 2, 3 \dots$

2. To start the MMF Server as an interactive program, enter the command:

```
NSQMSG -m"qmgr_name" [other options]
```

For an explanation of the options, see [step 4 in section 7.11](#), Configuring the MMF Server.

7.13 Verifying the MMF Server Configuration

After you configure the MMF Server, you should verify that it has been installed correctly and that it is functioning properly.

To verify the MMF Server configuration

1. Start the MMF Server as a detached process or as an interactive program, as described in [section 7.12](#), Starting the MMF Server.
2. If you are running the MMF Server as a detached process, enter the command:

```
show system
```

This command lists all running processes. Listed processes should include `NSQMSG`, representing an IBM MQ queue manager that is being accessed. If running the MMF Server as an interactive program, the list of processes shows automatically.

3. Start the workgroup server on your Windows or UNIX machine, if not already running.
4. Using the Message Explorer on a Windows machine that is on the same network, try to display, put, delete, copy, and move messages on the VMS machine.

7.14 Stopping the MMF Server

You can stop the MMF Server in one of two ways, depending on whether it is running as a detached process or as an interactive program attached to your window.

To stop the MMF Server

1. If you are running the MMF Server as a detached process, type:

```
ENDMSC qmgr_name
```

to stop it gracefully, or:

```
stop NSQMSG
```

to kill it.

2. If you are running the MMF Server as an interactive program, close the window by typing CTRL-C or CTRL-Y. This will stop the MMF Server.

7.15 Configuring the MQ Event Publisher

The MQ Event Publisher is a back-end component of Nastel Navigator that replicates IBM MQ system events and publishes them to applications and management products that subscribe to them. This is an optional component. Use it if you have the node managed in multiple workgroups and events must be received by all workgroup servers; or if you want

dead-letter queue events to be generated and appear in Nastel Navigator to make you aware when a message was placed on the DLQ. The DLQ event generation is controlled by four options:

- -q<DLQ Time Rate>
- -n<DLQ Count Rate>
- -k<DLQ Delete Rate>
- -v<DLQ Events Max>

Refer to [section B.3.2](#) for a full explanation of these options.

Set the MQ Event Publisher environment variable

An `APWMQ_PUBSUB_OPTIONS` environment variable (logical) may be optionally set up to define the run time options of the MQ Event Publisher. All parameters must be separated with a ";" (semicolon) or "," (comma) delimiter. For example, the logical can be defined as follows:

```
define/system APWMQ_PUBSUB_OPTIONS "-t;-w50000;-dDEAD.QUEUE"
```

When the MQ Event Publisher runs, it will create a temporary subscriber file in directory `APWMQ_MQS_HOME` (for example, `SYS$COMMON:[NASTEL.APWMQ.CONFIG.GROUPS]`) unless an alternate is given with the `-i` option.

If the logical is not defined, Event Publisher will use a set of default values. Enter "`nsqpub -?`" to display the options and their default values.

Configure the MQ Event Publisher queues

1. If your company has queue naming conventions, edit the `[APWMQ_HOME]/mqsc/pubsub.tst` MQSC script to customize the NASTEL queue names with the high and middle level queue name qualifiers that are (or will be) specified in the qualifier file `[APWMQ_HOME]/config/mqqual.ini`. Refer to [Appendix C](#), for a discussion of qualifiers and the file syntax.
2. Enter the IBM MQ command:

```
RUNMQSC qmgr_name < SYS$COMMON:[NASTEL.APWMQ.MQSC]PUBSUB.TST
```

The following message management queues are created:

```
NASTEL.PUBLISH.EVENT.QUEUE (local queue)
SYSTEM.ADMIN.QMGR.EVENT (alias queue)
SYSTEM.ADMIN.CHANNEL.EVENT (alias queue)
SYSTEM.ADMIN.PERFM.EVENT (alias queue)
```

The last three queues will be defined as alias queues, which point to the target queue `NASTEL.PUBLISH.EVENT.QUEUE`.

Create APWQM queue manager name file

The same queue manager name file, `MQS.INI`, created for the agent (see [section 7.8](#), Configuring the Agent), is also required for the Event Publisher.

7.16 Starting the MQ Event Publisher

To start the MQ Event Publisher as a detached process, enter the command:

```
@APWMQ_BIN:RUN_NSQPUB.COM
```

Two additional child processes, executables NSQDLQLSR.EXE and NSQEVQLSR.EXE will be started for each queue manager. To display their process names, enter:

```
show process nsqpub/all
```

7.17 Stopping the MQ Event Publisher

To stop the MQ Event Publisher gracefully, enter the command:

```
PUBEND
```

To kill it, enter:

```
stop NSQPUB
```

7.18 Setting Nastel Navigator Component for Automatic Startup

If you prefer, VMS Nastel Navigator components can automatically start each time the machine is booted.

To set Nastel Navigator Components for automatic startup

Enter the following lines in the system startup command file (for example, SYS\$STARTUP:SYSSTARTUP_VMS.COM):

```
$! Define APWMQ logicals and foreign commands
```

```
$_@apwmq_root:[NASTEL.APWMQ.BIN]SET_APWMQ.COM
```

```
$!
```

```
$! Enable the UCX service "nsqmq" for the APWMQ agent
```

```
$UCX ENABLE SERVICE NSQMQ
```

```
$! Start the APWMQ message servers, one per queue manager
```

```
$_@APWQ_BIN:RUN_NSQMSG.COM
```

```
$_@APWQ_BIN:RUN_NSQMSG2.COM
```

```
$! Start the APWMQ event publisher
```

```
$_@APWQ_BIN:RUN_NSQPUB.COM
```

7.19 Using the Utility Programs

7.19.1 Reason Code Lookup Utility (nsqcode)

Refer to the [nsqcode Utility](#) section.

7.19.2 Agent Client Utility (nsqcl)

The nsqcl utility is used to verify operation of the agent. It prompts you for the location of the Nastel Navigator agent (host name or IP address) and agent listener port number. Used in conversational mode only. Refer to the [Client Test Utility \(nsqcl\)](#) section for detailed example of the use of the command.

7.19.3 Message Client Utility (nsqmmfcl)

Refer to the [Message Management Utility](#) section.

7.19.4 Message File Utility (nsqmsgf)

Refer to the [Message Get/Put Utility](#) section.

7.19.5 Ping Utility (nsqping)

Refer to the [nsqping \(Ping\) Utility](#) section.

7.20 Periodic Maintenance Procedures

Purge files `APWMQ_LOGS:NSQMQ.OUT`, `NSQMQ.ERR`, `NSQPUB.OUT`, and `NSQMSGi.OUT`. Also purge `SYSSYSROOT:[SYSMGR]nsqmq.log`. A copy of the latter file is created each time the NSQMQ process starts and records the login.

7.21 Troubleshooting Procedures

Use the following commands for various troubleshooting scenarios.

monmq active

Use this command to show running IBM MQ processes, including their image name and their associated queue managers. The image name `AMQZLAA0.EXE` is the queue manager agent.

monmq delete

Use this command to clear the IBM MQ global sections if you have trouble starting a queue manager. Be sure all queue managers are stopped first. If you cannot stop a queue manager with `endmqm` (using `-i` or another option), kill the process `qmgr_EC` (execution controller) first. This will stop most of the queue manager processes. If any remain, use the `stop process_name` command. Then execute the `monmq` command.

monitor process/topcpu [/interval=refresh_secs]

Use this command to show the processes that use the most CPU. You can vary the statistics gathering interval. For more information, execute `HELP MONITOR MONITOR PROCESS`.

UCX show dev

Use this command to show any TCP connections to the NSQMQ (agent) service, including the remote host and port number. This command is the VMS counterpart of `netstat -a` (or `-n`) on UNIX or Windows platforms.

UCX show service nsqmq/fu

Use this command to show whether the NSQMQ service is enabled or disabled, and the number of active connections to the agent.

pipe show system | search sys\$input nsq

Use this command to show all running processes whose names start with NSQ. For the Event Publisher, to see its child processes, use the command `show process nsqpub/all`.

Queue manager gets error "global page file count exceeded"

7.22 Uninstalling Nastel Navigator on VMS

To uninstall Nastel Navigator

1. Remove the UCX nsqmq service:

```
ucx set nosevice nsqmq
```

Respond YES to the prompt:

```
Remove? [N]: YES
```

To exit from the UCX prompt, type `CTRL-Z`.

2. Cleanup files created by the Nastel Navigator agent:

```
del SYS$SYSROOT:[SYSMGR]NSQ*.*.*
```

3. Delete the Nastel Navigator product directory:

```
del SYS$COMMON:[NASTELE.APWMQ ...]*.*.*
```

4. Delete the NASTEL main directory if you have no other Nastel products installed under it:

```
del SYS$COMMON:[000000]NASTELE.DIR;*
```

5. Restore the event local queues and remove the Nastel queues. For every queue manager that was managed by Nastel Navigator, run the following MQSC commands:

```
runmqsc QmgrName
```

```
CLEAR QLOCAL('NASTELE.EVENT.QUEUE')
```

```
DELETE QLOCAL ( 'NASTEL.EVENT.QUEUE' )

CLEAR QLOCAL ( 'NASTEL.PUBLISH.EVENT.QUEUE' )
DELETE QLOCAL ( 'NASTEL.PUBLISH.EVENT.QUEUE' )

CLEAR QLOCAL ( 'NASTEL.MMF.ADMIN.COMMAND.QUEUE' )
DELETE QLOCAL ( 'NASTEL.MMF.ADMIN.COMMAND.QUEUE' )
CLEAR QLOCAL ( 'NASTEL.MMF.AUDIT.QUEUE' )
DELETE QLOCAL ( 'NASTEL.MMF.AUDIT.QUEUE' )

DELETE QALIAS (SYSTEM.ADMIN.QMGR.EVENT)
DELETE QALIAS (SYSTEM.ADMIN.CHANNEL.EVENT)
DELETE QALIAS (SYSTEM.ADMIN.PERFM.EVENT)
DELETE QALIAS (SYSTEM.ADMIN.CONFIG.EVENT)
DELETE QALIAS (SYSTEM.ADMIN.ACCOUNTING.QUEUE)
DELETE QALIAS (SYSTEM.ADMIN.STATISTICS.QUEUE)

DEFINE QLOCAL (SYSTEM.ADMIN.QMGR.EVENT)
DEFINE QLOCAL (SYSTEM.ADMIN.CHANNEL.EVENT)
DEFINE QLOCAL (SYSTEM.ADMIN.PERFM.EVENT)
DEFINE QLOCAL (SYSTEM.ADMIN.CONFIG.EVENT)
DELETE QALIAS (SYSTEM.ADMIN.ACCOUNTING.QUEUE)
DEFINE QLOCAL (SYSTEM.ADMIN.STATISTICS.QUEUE)
```

6. If you have set up your Nastel Navigator components to automatically start every time you reboot your system (see [section 7.18](#)), delete or comment out the related command lines in the `sys$startup:startup.com` file.
7. Reboot the computer.

Chapter 8: Log Adapter Installation

8.1 Log Adapter

The Nastel Navigator log adapter attaches to a single workgroup server and logs all enabled events to a text log file (event log) or sends them directly to OVO via `opcmsg` calls. The log file produced can be used as input to the HP OpenView log file encapsulator, Tivoli endpoint or any other log scraping agent. Based on event enable/disable information contained in the severity code file `nsqsev.ini`, the messages are either output to a log file or ignored.

8.2 Processing of Events

The workgroup server passes the events to any connected client program that uses the Nastel Navigator API and requests a copy of the events. The log adapter is such a client; it writes events to a log file or uses the `opcmsg` command to send an event message to the OVO management server.

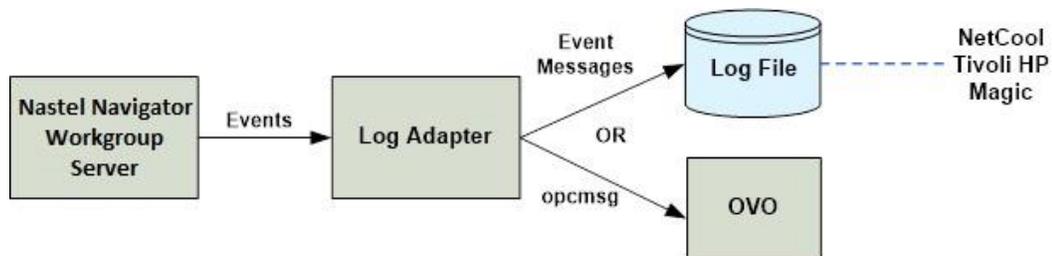


Figure 8-1. Processing of Events

8.3 Manual Installation of UNIX Nastel Navigator Log Adapter

1. Get the IBM MQ Agent package from the Nastel Resource Center or a CD-ROM. On the CD-ROM you will find the IBM MQ Agent package `m6wmq_version_platform.tar.Z`. Example: `M6WMQ_6.2.7_SUN_Solaris10_sparc.tar.Z`.
2. FTP the agent package in binary to the target node, directory `/tmp`, or another temporary directory. The target node can be either a managed node (where IBM MQ is installed) or the management server (if you were going to run the Nastel Navigator workgroup server and the log adapter from there).
3. Login as root.
4. Create group `mqm` and user `mqm` if they don't exist. Make `mqm` a member of the `mqm` group.
5. Change to the base install directory for the Nastel Navigator package. Example: `/opt`
6. Uncompress and un-tar the package into the base directory:

```
cd /opt
```

```
uncompress /tmp/M6WMQ_6.2.7_SUN_Solaris10_sparc.tar.Z
```

```
tar -xvf /tmp/M6WMQ_6.2.7_SUN_Solaris10_sparc.tar
```

The package will be installed to `/opt/nastel/apwmq`. This directory is called the APWMQ home directory. The files will be owned by user `mqm`, group `mqm`.

Verify the install; you should see a directory display similar to this:

```
$ ls -l /opt/nastel/apwmq
total 4872

-rw-r--r--  1 mqm      mqm      21449 Dec 14 08:42 LICENSE.htm
-rw-r--r--  1 mqm      mqm     105285 Dec 14 08:42 README.htm
-rw-r--r--  1 mqm      mqm     30684 Dec 14 08:42 README.txt
-rw-r--r--  1 mqm      mqm       28 Dec 14 09:21 SYSVER.txt
drwxrwxrwx  2 mqm      mqm     1024 May  4 15:38 bin
drwxr-xr-x  4 mqm      mqm     512 Dec 14 08:45 config
drwxr-xr-x  4 mqm      mqm     512 Feb 25 10:57 groups
drwxr-xr-x  2 mqm      mqm     512 Dec 14 09:21 ito
drwxr-xr-x  2 mqm      mqm    1536 Apr 19 12:50 lib
drwxr-xr-x  2 mqm      mqm     512 Dec 14 09:21 mib
drwxr-xr-x  2 mqm      mqm     512 Dec 14 08:44 mp
drwxr-xr-x  2 mqm      mqm     512 Dec 14 09:21 mqmdll
drwxr-xr-x  2 mqm      mqm     512 May 19 14:46 mqsc
drwxr-xr-x  2 mqm      mqm     512 Dec 14 08:44 mt
drwxr-xr-x  5 mqm      mqm     512 Dec 14 09:21 odbc
drwxr-xr-x  2 mqm      mqm     512 May 20 16:37 sbin
drwxr-xr-x  2 mqm      mqm     512 Dec 14 09:21 scripts
drwxr-xr-x  2 mqm      mqm     512 Feb 18 09:30 sql
drwxr-xr-x  2 mqm      mqm     512 Dec 14 09:21 tivoli
drwxr-xr-x  4 mqm      mqm     512 Dec 14 08:46 tools
```

7. Install the Log Adapter message severity file:

```
cp /opt/nastel/apwmq/ito/nsqsev.ini.ito
/opt/nastel/apwmq/config/groups/nsqsev.ini
```

8. Define environment variables in the profile of user `mqm` (Example: `/home/mqm/.profile`). These are needed to run all the IBM MQ Agent programs and many of the SPI scripts.

```
APWMQ_HOME= /opt/nastel/apwmq
library_path= $APWMQ_HOME/lib:$library_path
PATH= $APWMQ_HOME/bin:$PATH
export APWMQ_HOME library_path PATH
```

The *library_path* varies with platform:

SUN, LINUX: LD_LIBRARY_PATH

AIX: LIBPATH

9. On AIX or LINUX managed node: Define a soft link to the Bourne shell interpreter to satisfy the reference on line 1 of many SPI scripts, which is needed for Solaris compatibility:

```
ln -s /usr/bin/sh /usr/xpg4/bin/sh
```

10. Make the required directories with ownership mqm:mqm and the shown permissions:

```
/var/nsq      777 (rwx rwx rwx)
/var/nsq/log  777 (rwx rwx rwx)
/var/nsq/conf 775 (rwx rwx r-x)
```

8.4 Start the Log Adapter Using the SPI

Application name: Windows: Start LA Svc, UNIX: Start LA

Purpose: Starts the Nastel Navigator Log Adapter service on the selected Windows or UNIX log adapter node. The workgroup name option (-m), the opcmsg option (-i) or the event log file option (-l) and log file name, and the reconnect interval (-t) are defined by the system environment variable `APWMQ_LA_OPTIONS`.

Customized startup parameters: None

8.5 Stop the Log Adapter on a Windows Node Using the SPI

Application name: Windows: Stop LA Svc. UNIX: Stop LA.

Purpose: Stops the Nastel Navigator log adapter service on the selected Windows or UNIX log adapter node.

Customized startup parameters: None

8.6 Start the Log Adapter on a UNIX Node

Purpose: Start the Nastel Navigator log adapter (LA) process `nsqadp` on a selected UNIX node. The Log Adapter, an OVO message source, connects to the workgroup server and generates event messages for IBM MQ or Nastel Navigator.

The messages are written to a log file or generated directly to OVO via `opcmsg` calls based on event subscription information contained in the file `nsqsev.ini`. (See “Initialization Files” below.)

On a UNIX command line, the log adapter can be started with the command:

```
nsqadp [options]
```

using the options in the table below.

Table 8-1. Customized Startup Parameters to Start Log Adapter on UNIX

Parameter	Description	Default
<code>-mWorkgroup_name</code>	Workgroup server to connect to.	None. Required operation
<code>[-i]</code>	Run in ITO/OVO environment.	None. ITO environment

Table 8-1. Customized Startup Parameters to Start Log Adapter on UNIX

Parameter	Description	Default
-t [Time_out]	Workgroup reconnect interval: Time, in milliseconds, between connections retries to a workgroup server.	10000 ms
-l [log_name]	Write message to event log file. For OVO: writes events only using opcmmsg. If -l with no log_name, default log names are used.	Windows: OVO: event log is not used Tivoli: \temp\group_name.log UNIX: OVO: /var/nsq/log/nsqadp_GM.log Tivoli: /tmp/group_name.log
-s	UNIX option: Write to syslog. You can choose -s without -l or -i	No writing to syslog. Option -i or -l determines default.
-Fn	UNIX option: Syslog facility index 0-7 for facility LOG_LOCALn.	0. Use only with -s option.
-r {all nn none}	Append reason code description text: all, first nn bytes, or none.	Default and minimum: Length of message ID example: MQRC_NOT_AUTHORIZED:
-f [cc]	Use field delimiter 'cc' in the messages, max 2 chars. Default: '::' Use -fsp for space delimiter. If -f is selected, a space will appear vs ':' after QMGR, ObjName and Objtype	:: (double colon)
-h [Interval]	Heartbeat interval (secs): Send a heartbeat message EXRC_LA_HEARTBEAT if no event msg was sent in the last 'Interval' seconds.	No heartbeat
[-d]	Debug/trace option: shows the expanded opcmmsg message.	No trace

Initialization Files

The nsqadp program uses two initialization files that must be present for proper results. The files are installed as part of the IBM MQ Agent installation procedure:

```
$APWMQ_HOME/config/groups/nsqrc.ini
$APWMQ_HOME/config/groups/nsqsev.ini
```

nsqrc.ini File

The reason code file, `nsqrc.ini`, may be viewed with a text editor to look up reason codes and event ID codes to be included in the `nsqsev.ini` file.

nsqsev.ini File

The format of the `nsqsev.ini` file is:

```
event_ID_code=severity_code
```

For Example:

```
20002=1
20100=2
20101=0
...
```

The contents of the `nsqsev.ini` file determine which events will be subscribed to and acted upon. If an Nastel Navigator event ID code is not in the `nsqsev.ini` file, it will not be acted upon. The severity code determines the severity of the message appearing in the log file.

The following table explains the relationship between the severity code and severity type:

Table 8-2. nsqsev.ini Severity Codes	
Severity Code	Severity Type
1	Normal
2	Warning
3	Minor
4	Major
5	Critical
0 or negative value	Not acted on, ignore

Using the severity file example listed above, the events 20002 (EXRC_OBJECT_CHANGED) and 20100 (EXRC_MQNODE_COMMAND_FAILED) are subscribed events, which have severity types of normal and warning, respectively. The event 20101 (EXRC_MQNODE_CONNECTED) will not be acted on since the severity value is 0 (zero).

Log Adapter Message Formats

The log adapter message format is the same for messages sent to log files or to OVO as an `opcmsg` message:

```
Severity::ManagerName::MQNodeName::TimeStamp::EventIdValue::QMGR:
QmgrName<space> ObjType:ObjectType<space>
ObjName:ObjectName<space>RC (ReasonCodeValue) MessageText
```

where the message fields are described in the table below:

Table 8-3. Log Adapter Message Fields	
Field	Explanation
Severity	Severity value, a single digit.
ManagerName	Workgroup server name.
MQNodeName	Name or IP address of the node where the event occurred.
TimeStamp	Date-time when the event was converted to a message: mmm dd hh:mm:ss yyyy
EventIDValue	Nastel Navigator event ID EXRC_xxx numeric value as listed in the nsqrc.ini file.
QmgrName	Name of the IBM MQ queue manager that generated the event or is associated with the event.
ObjectType	Object type for the event. Examples: LQ, AQ
ObjectName	Name of the object, which is the subject of the event. For example, a queue or channel name.
ReasonCodeValue	Numeric IBM MQ reason code, if any, defining why the event occurred. Zero means successful completion or Not Applicable in cases where the event was generated by the workgroup server.
MessageText	Variable number of characters from the reason code description as it appears in the nsqrc.ini file, always starting with MQRC_xxx or EXRC_xxx, the reason code mnemonic. See the -r option in the <i>Customized Startup Parameters</i> in Table 8-1 above.

Example:

```
1::MQM::127.0.0.1::Aug 7 15:13:32 2000::20001::QMGR:DBTEST ObjType:LQ
ObjName: abc RC(0)EXRC_OBJECT_NEW: An Nastel Navigator workgroup server has
discovered a new object. The type of object can be local queue, queue manager, alias queue, remote
queue, process, model queue, or node. Corrective Action: none.
```

8.7 Stop the Log Adapter on a UNIX Node

Purpose: To stop the log adapter on a selected UNIX node.

Usage: Use the application \$APWMQ_HOME/sbin/nsqadp_stop.sh.

8.8 Start the Log Adapter from the Windows Command Line

This procedure can be used when running Log Adapter in test mode. Normally, in production mode, it would be started as a Windows service.

1. Run nsqadp -console on Windows as follows:

```
C:\>nsqadp -console -?
```

The following is displayed:

```
CONSOLE MODE: "M6-WMQ Log Adapter".
(C)Copyright 1995-2008, Nastel Technologies Inc.
Reproduction, adaptation, or distribution without prior
written permission is prohibited, except as allowed under
the copyright laws.
Sysname("Windows_XP")
NodeName("D-HC00-28")
Release("")
VERSION: 6.0.0007
```

```
Usage: nsqadp
```

```
  -m[groupname]      -- Workgroup Server to connect to
  [-i]              -- Enable ITO environment. Def: Tivoli environment.
  [-tTime_out]      -- Workgroup reconnect interval, msecs. Def: 10,000.
  [-l[log_name]]    -- Write message to event log file:
                    For ITO: if no -l, write events using opcmmsg
                    For ITO and Tivoli: if -l with no log_name, default log
is: Windows:
                    ITO:    \usr\OV\nsq\log\nsqadp_GM.log
                    Tivoli: \temp\group_name.log
                    UNIX:
                    ITO:    /var/nsq/log/nsqadp_GM.log
                    Tivoli: /tmp/group_name.log
  [-U]              -- Username for authenticating with Workgroup Server.
Default: current
                    user
  [-P]              -- Password for authenticating with Workgroup Server.
Default: current
                    user
  [-R]              -- Security realm for authenticating with Workgroup Server.
Default:
                    security realm for current user
  [-a]              -- Don't authenticate with Workgroup Server. Connect to
Workgroup Server
                    as a GUEST user.
  [-s] (UNIX only) -- Write to UNIX syslog. Not compatible with -i and l
  [-F[n]] (UNIX only) -- syslog facility index, 0-7, for facility LOG_LOCAL[n].
Default: 0
  [-b] (UNIX only) -- Runs the Log Adapter in the background as a daemon
```

2. Copy [APWMQ_HOME]/ito/nsqsev.ini.ito (or /tivol/nsqsev.ini.tivoli) to [APWMQ_HOME]/config/groups/nsqsev.ini

8.9 Troubleshooting the Log Adapter

The usual question is **“Why aren’t the MQ events appearing in my OVO message browser?”** Review this checklist if you have such a problem. If you cannot resolve the problem, call Nastel support.

1. Check if Log Adapter is running with the -i option.

If the Log Adapter runs on UNIX: Perform a status check by using the UNIX command

```
ps -aef | grep nsqadp
```

2. Check if Log Adapter severity file (nsqsev.ini) is present.

Check if the file \$APWMQ/config/groups/nsqsev.ini is present. If not, copy the file \$APWMQ_HOME/ito/nsqsev.ini.ito or \$APWMQ_HOME/Tivoli/nsqsev.ini.Tivoli into the directory \$APWMQ_HOME/config/groups/nsqsev.ini (for Windows systems, the path is the same %APWMQ_HOME%\...).

The Log Adapter looks up an event's severity value to decide if the event should be written to a log file, forwarded to OVO, or discarded. File entries contain event name and severity code 0-5, where 0 means ignore this event and 1-5 correspond to OVO severity codes. If the severity file is missing, it is equivalent to severity 0 for all events and therefore event messages will not be logged or sent as opcmmsg messages.

3. Are the environment variables APWMQ_xxx defined properly?
4. Are the event categories enabled in the queue manager attributes?
5. Are the nodes in question in the Node Bank and one of the UNIX SPI Node Groups, such as NSQ mq_ux?
6. Are the UNIX SPI node group and message group (WMQ and Nastel Navigator) intersections checked in the User Bank responsibility matrix of the user trying to view the message browser?
7. Are the SPI templates nsq_nsqmsg and nsq_SpiMsgs deployed/distributed to the Log Adapter node and enabled? Use command opctemplate to verify.
8. Is the IBM MQ Agent, process nsqmq, running on the subject nodes?
9. Is the IBM MQ Event Publish/Subscribe (PS), process nsqpub, running on the subject nodes (if the queue managers were configured to run PS)?
10. Is the IBM MQ workgroup server, process nsqmgr, running on its node?

8.10 Reason Code and Reason Qualifier Locations

The "RC(nn)" that appears in the message at point B (2162) is NOT the reason code; it's really a reason qualifier. The reason code numeric appears earlier in the msg at point A (2006).

```

                A
2::MQM::IBM-9CZCQBR1CXD::Mar 03 09:35:52 2006::2223::QMGR:MILVUS ObjType: ObjName:
                B
RC(2162)MQRC_Q_MGR_NOT_ACTIVE:

```

This Page Intentionally Left Blank

Chapter 9: Connection Manager for IBM MQ

The Connection Manager for IBM MQ allows you to connect to queue managers running on remote IBM MQ servers without installing agents on the server. It connects to queue managers using Server-Conn channels, and issues administration commands and receives replies and events. While the Server-Conn channel is available, you will be able to perform most IBM MQ administrative functions. If you have multiple workgroup servers, you will need one Connection Manager per workgroup.

For information on the version 10.x Java Agent/Connection Manager for IBM MQ, refer to the following chapter.

9.1 Platforms

The Connection Manager for IBM MQ is available on the following platforms:

Windows: refer to the Windows [Package Installation](#)

Unix (Linux, Solaris, AIX, HP_UX): refer to the Unix [Pack Installation](#)

9.2 Prerequisites

IBM MQ Client Library

The Connection Manager is an IBM MQ client application and needs the IBM MQ client library installed to run on a non-IBM MQ server machine.

The MCAUSER attribute of the SVRCONN channel

When using the Connection Manager, the connection to a remote queue manager is done via a SVRCONN (server-connection) channel on the remote server. By default, when the SYSTEM.DEF.SVRCONN channel is created, the MCAUSER attribute is set to blanks.

The user ID for connecting to a queue manager

On OS/400, UNIX systems, and Windows systems, the user ID whose authority is checked when an MCA connects to a queue manager is the one under which the MCA is running. This is known as the default user ID of the MCA. The responder (svrconn) MCA will be started by the IBM MQ listener and will run under the same user ID as that of the listener, such as mqm on UNIX systems. The default user ID of the MCA must be a member of the Windows/UNIX mqm group or the Windows Administrators group or the QMQMADM group on OS/400. This requirement is typically satisfied at IBM MQ install time.

The user ID for subsequent authority checks

- The user ID whose authority is checked when the MCA accesses queue manager resources after connecting to a queue manager might be different from the one that was checked when the MCA connected to the queue manager. User IDs that might be used are the value of the MCAUSER parameter in the channel definition.
- For a SVRCONN channel, the user ID that is received from a client system when an IBM MQ client application issues an MQCONN call.

MCAUSER Values

If the MCAUSER field is blank at connection time, the SRVRCONN MCA uses the user ID received from a client (such as Connection Manager) when attempting to access queue manager resources (for example, objects) after connecting to the queue manager. If the client user ID is not a valid IBM MQ user on the target queue manager, then the resource access will fail with a 'not authorized' error.

NOTE: If the user ID received from the client is blank, the user ID that started the server-connection channel is used. This case will not occur with the CM since the user ID is always non-blank.

Use of Model Queue

The Connection Manager uses a model queue for creation of temporary dynamic queues, used for command processing and message retrieval. The default is SYSTEM.DEFAULT.MODEL.QUEUE. Rather than alter the system queue if some parameter customization is needed, you can create a custom model queue to be used instead, say NSQCM.MODEL.QUEUE, with, for example, a very large MAXMSGL parameter value for retrieving very large messages. Enter the new queue name in the NASTEL.MODEL.QUEUE line in file [APWMQ_HOME]\config\groups\mqqual.ini. When the Nastel Navigator agent starts, it will check the qualifiers file and use that model queue instead of the default.

Using +a and Connection Authentication with Alternate User

Use the following authorization parameter to change how connection manager (CM) connects to the queue manager for IBM MQ.

[+a]: Enable alternate user connection authentication.

[-a]: Disable alternate user connection authentication(default).

Enabling this option will cause CM to connect to the remote queue manager using the user ID and password specified in the remote queue manager entry in the Remote Qmgrs configuration table (see [Remote Queue Manager Configuration in Nastel Navigator](#)), and not as the user under which CM is running.

To have the queue manager authenticate the user and password you must configure the queue manager CONNAUTH attribute to use a given AUTHINFO record, and in that AUTHINFO record, set attribute CHCKCLNT (REQUIRED).

When using +a, the Sharing Conversations parameter SHARECNV needs to be set to 1 in the definition of the server-conn (server connection) channels that CM will use on the remote queue manager, such as SYSTEM.DEF.SVRCONN or a customized NSQCM.SVRCONN, because the channel conversation should not be shared:

```
SHARECNV(1)
```

9.3 Configuration Steps

1. Start your remote queue manager TCP listeners with unusual port numbers that unauthorized system users would not know and would be hard to guess.
2. Use those port numbers when configuring the CM RemoteQmgrs.

3. For improved security, Connection Manager should use a svrconn ChannelName other than the default SYSTEM.DEF.SVRCONN. The MQ admin should set the MCAUSER attribute in channel SYSTEM.DEF.SVRCONN to an invalid user ID, such as JUNK or DUMMY. This will block all usage of the default svrconn channel.
4. For initial Connection Manager testing, avoid user ID and security problems:

On the remote queue managers, set the Connection Manager-selected SRVCONN channel attribute MCAUSER to a user ID that has full access rights to the queue manager resources. Thus, use the user ID used to run the queue manager, such as mqm on UNIX; MUSR_MQADMIN or SYSTEM on Windows; QMQM on OS/400; or MQM or SYSTEM on OpenVMS. This user ID has full access rights to the queue manager resources.

5. After initial Connection Manager testing, use a higher level of security:

On the remote queue managers, set the selected svrconn channel attribute MCAUSER to blank and define the Connection Manager user ID on the target queue manager as a valid MQ user using the AP-WMQ Explorer (right-click **Qmgr** > **Commands** > **Security** > **Authority Records**) or the setmqaut command.

9.4 Remote Queue Manager Configuration in Nastel Navigator

From the workgroup server, select **Create** > **Remote Queue Managers**.

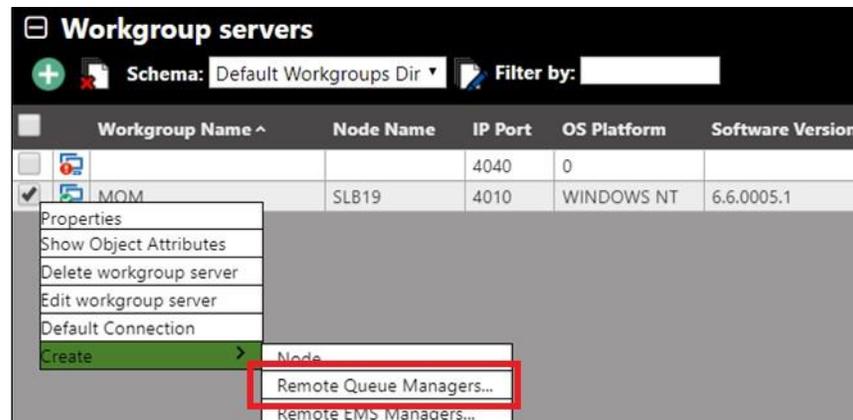


Figure 9-1. Creating Remote Queue Managers

The *Remote Queue Manager Connections* window opens. Click the **Add** button to add a new remote queue manager connection. You can also update or delete existing remote queue managers.

Remote Queue Manager Connections
? x

#		Instance Name	Queue Manager Name
1	<input type="checkbox"/>	REMOTE_QMGRS4	Special
2	<input type="checkbox"/>	REMOTE_QMGRS	Main
3	<input checked="" type="checkbox"/>	REMOTE_QMGRS1	T1
4	<input type="checkbox"/>	REMOTE_QMGRS2	T2
5	<input type="checkbox"/>	REMOTE_QMGRS3	General

Add
Modify
Delete

Attribute Name	Attribute Value
Instance Name	REMOTE_QMGRS1
Queue Manager Name	T1
Connection Name	IP ADDRESS(IP PORT)
Channel Name	SYSTEM.DEF.SVRCONN
Command Queue	SYSTEM.ADMIN.COMMAND.QUEUE
Conversion	DEFAULT
SSL Key Repository	
SSL Crypto Hardware	
SSL Cipher Specification	

Ok
Cancel

Figure 9-2. Remote Queue Manager Connection

When **Add** is clicked, the *Add Queue Manager Connection* window opens. Populate the fields on the **General**, **Communication** and **SSL** tabs.

The screenshot shows a dialog box titled "Add Queue Manager Connection" with a help icon in the top right. On the left, there are three tabs: "General" (selected), "Communication", and "SSL". The main area contains the following fields and text:

- Connection Manager Instance name:** A text box containing "REMOTE_QMGRS".
- Queue Manager name:** A text box containing "QMGR_NAME".
- Project name:** A text box containing "DEFAULT".
- Specify a user name and password to connect to the queue manager using security parameters (available in WMQ v.8.0 and later) or leave it empty if authentication is not required:**
- User ID:** An empty text box.
- Password:** An empty text box.

Figure 9-3. Add Queue Manager Connection

9.5 Start Connection Manager

9.5.1 Start Connection Manager on Windows

You can start Connection Manager instance from Windows services or the command line.

1. To start from Windows Services, first make sure the workgroup server is already running. Then navigate to the Services screen and start M6-WMQ Connection Manager (workgroup_name) service, where workgroup_name is the name of the workgroup server, such as MQM, that you chose during M6-WMQ for Windows installation.
2. To start from the command line:

- a. Define an environment variable used by CM:

```
Set APWMQ_CM_OPTIONS=-mworkgroup_name; -nREMOTE_QMGRS;
-sPort [; other_CM_options]
```

where REMOTE_QMGRS is the name of the Connection Manager instance that works with the given workgroup_name.

- b. To start the CM for a workgroup, enter the command:

```
nsqcm -console
```

To see other CM options that can be incorporated into the environment variable:

```
nsqcm -console -?
```

The options are the same as for the Nastel Navigator Agent.

Note: When connecting to a zOS queue manager, the `-N` option should be specified to prevent PCF buffer overflow.

3. If you have multiple workgroups, repeat steps 1-2 in a new command window for each workgroup. If you don't have a CM service defined for a particular workgroup, create it from the Windows command line:

```
nsqcm -install workgroup_name
```

9.5.2 Connection Manager on Linux

1. Edit script `[APWMQ_HOME]/sbin/nsqcm_start.sh`
 - a. Change the `APWMQ_HOME` environment variable setting `/opt/nastel/apwmq` if your installation directory is different.
 - b. Edit startup options `-n<alias_node>` and `-s<CM_listener_port>`, default 5011.
 - c. Add any other startup options as shown in the header of the script or in section 9.7, Command-Line Arguments.
2. Run the script:
`[APWMQ_HOME]/sbin/nsqcm_start.sh`

9.5.3 Stopping the Connection Manager on Linux

1. Edit script `[APWMQ_HOME]/sbin/nsqcm_stop.sh`
 - a. Change the `APWMQ_HOME` environment variable setting `/opt/nastel/apwmq` if your installation directory is different.
 - b. Edit options `-n<alias_node>` and `-s<CM_listener_port>`, default 5011.
 - c. Use the same values used in the start script.
2. Run the script:
`[APWMQ_HOME]/sbin/nsqcm_stop.sh`

9.6 Add Nastel Navigator IBM MQ Objects

Create a new viewlet to add an IBM MQ object. On the *Create new IBM MQ viewlet* screen, select **IBM MQ** from the **Product** drop-down and select the object type on the left side of the screen:

The screenshot shows a dialog box titled "Create new IBM MQ Queue viewlet". On the left is a vertical sidebar with categories: Product, Node, Manager, Queue (highlighted), Channel, Process, Topic, Listener, Namelist, Service, Auth info, Cluster QMgr, Subscription, and Channel auth rec. The "Product" dropdown is highlighted with a red box and shows "IBM MQ". The main area contains several input fields: "Viewlet name" (Temp_Queue1_Viewlet), "Workgroup server" (WGS107 - 0), "Temporary" (checked), "Node" (*), "Manager" (*), "Object name" (*), "Queue Type" (Local Queue), "Attribute filter" (empty), "Find messages" (checkbox), "Search criteria" (empty), "Data limit offset" (500), and "Search depth" (999999). At the bottom right are "Save changes" and "Cancel" buttons.

Figure 9-4. Create New IBM MQ Viewlet

9.7 Command-Line Arguments

The table below lists all command-line arguments applicable to the connection manager for IBM MQ.



The +a parameter cannot be used when running in conjunction with Nastel AutoPilot M6. Do not change the default -a setting.

Table 9-1. Connection Manager Command-Line Arguments

Arguments	Description	Default
Optional Arguments:		
-Q<mask>	Manage only queue managers whose names start with <mask> string.	
-D<name.com>	DNS name to create fully qualified host name.	
-m<WorkgroupName>	Specifies the workgroup server to which the Connection Manager must register.	
-m* (asterisk)	Registers the Connection Manager under all available workgroup servers. You can also use the asterisk to represent partial names.	
-t +t	Turns on/off PCF protocol trace.	Trace off
-x +x	Turns on/off tracing for TCP/IP packets.	Off
-e	Reuse TCP/IP address when binding. If port is in use or still lingering, workgroup can forcibly reclaim it.	Off
-r<retries>	Number of attempts IBM MQ agent makes to bind to an available TCP Port. It starts with specified port followed by successive port numbers until a port is found that is not in use.	1
-n<AliasNodeName>	Assigns a new name to be used when registering with workgroup servers. This name will override name of the local host.	
+u -u	Enables/disables authorization checks for letting users submit commands to the IBM MQ command server. +u gives all users effective authority over the IBM MQ Agent.	Authorization checked off
+q -q	Enables/disables queue manager verification. This option is needed to test if queue manager is properly configured for Nastel Navigator agent. Default: disabled.	Disabled
+z -z	Enables/disables use of DNS (Domain Name Server) for host name resolution. Disabling of DNS should be	Enabled

Table 9-1. Connection Manager Command-Line Arguments

Arguments	Description	Default
	used on systems where an Object Detached event states in its Description field that a host name cannot be resolved by the IBM MQ Agent.	
-d <discovery>	IBM MQ object auto discovery in minutes.	10 minutes
+i -i	Enable/disable reporting of node IP address to workgroup server.	On
-p <time>	Time interval to send registration to workgroup server periodically (in minutes).	Sent only at startup time.
-b <Buf. Size>	Set TCP send/receive buffer size in KB.	64KB
+g -g	Enable/disable the Nagle algorithm for send coalescing on TCP/IP connection. This option improves TCP performance.	Disabled
-s <port service_name>	Specifies the port number or service name to which the Connection Manager will bind.	5000
-v	Verify queue manager and communication only.	
-C	Disables user id conversion to uppercase. By default, the user id is converted to uppercase and passed to IBM MQ queue manager as an alternate user id.	
[-l <Que. Limit>]	TCP buffering queue limit. This option sets soft limit to buffer responses from IBM MQ to be sent to workgroup. The smaller buffer size, the more responsive the GUI will be.	10
-nosysevt	Bypasses collecting system event queue information	
-w	Maximum wait time (msec) for socket read/write event.	1000
-tN	Enable debug trace of name list discovery.	Off
N	Enable special name list discovery. Use on z/OS when thousands of queues, channels, processes, namelists, or auth info objects are defined and IBM 64KB PCF message size limit exists.	Disabled
Windows only arguments: (Used only when running this program from the command prompt)		
-install	Installs the Connection Manager as a service (if this has not already been done during installation.)	
-remove	Removes the Connection Manager as a service.	
-console	Runs the Connection Manager as a console application.	

9.8 Setup for Connection Manager for IBM MQ with SSL

1. The location of the key repository files will need to be specified on the *Remote Queue Manager Connections* screen. Perform the following:
 - a. From the workgroup server, select **Create > Remote Queue Managers**.

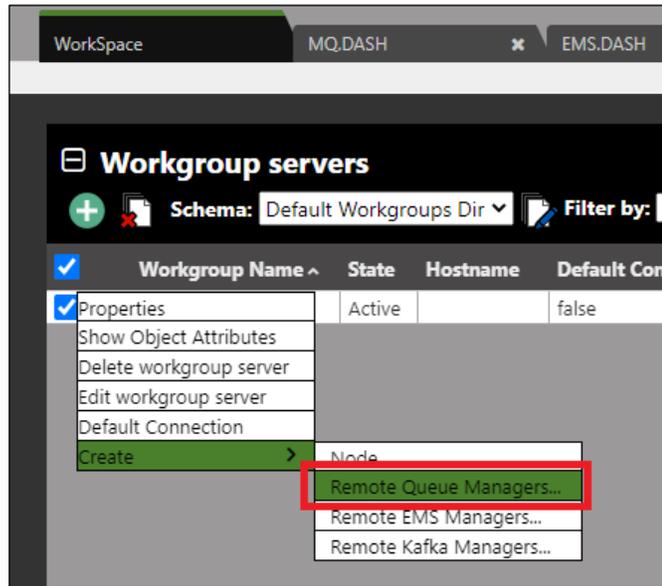


Figure 9-5. Create > Remote Queue Managers

- b. Select the connection manager from the left side of the screen and click **Modify**.

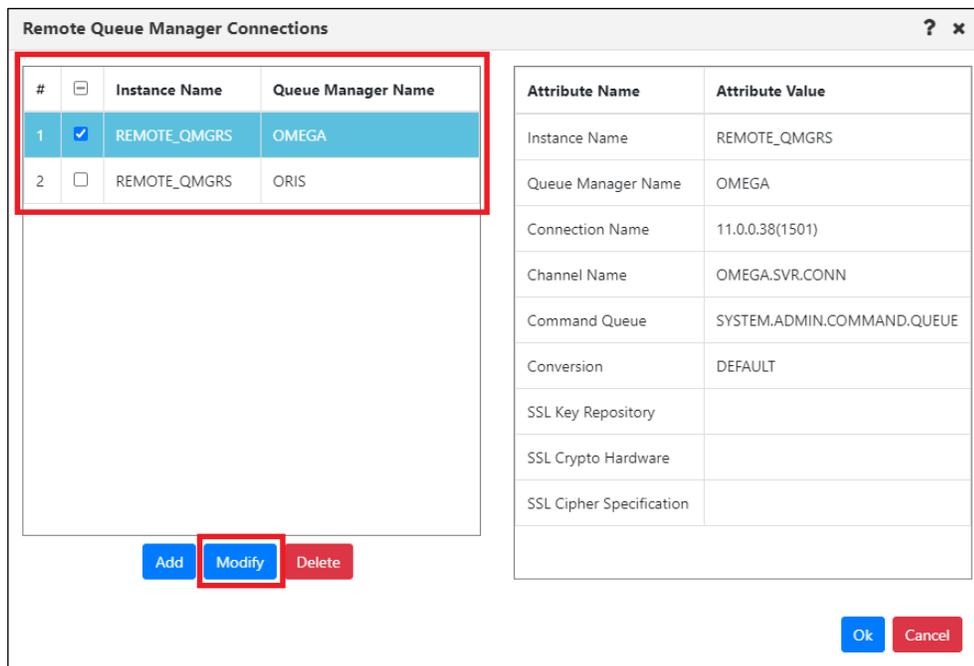


Figure 9-6. Select Connection Manager

- c. On the **SSL** tab within the **Key repository** field enter the path of the key files. Being that there are several **key.*** files (**key.kdb**, **key.rdb**, **key.sth**), do not include the file extension. Only specify the name that the key files start with. In the example below, the files all start with the name *key*.

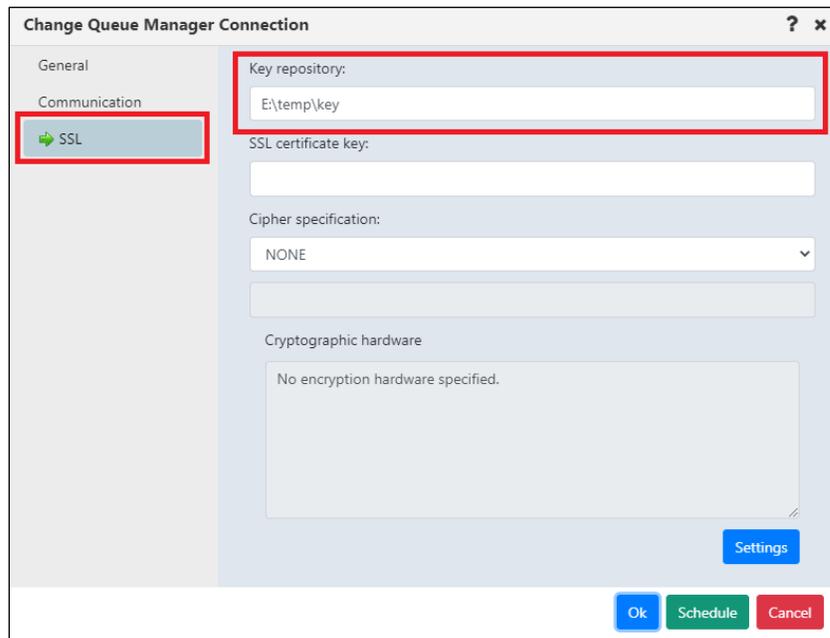


Figure 9-7. Key File Location

- d. Click **Ok** to save your changes.
2. When creating certificates for the key repository files for the server and client, the certificates will need to be the same, but the labels will differ.
 - a. **For the Server:** The queue manager key repository certificate will need to be named, **ibmwebspheremq{\$qmgrname}**. The name is case sensitive and must be all lowercase. In the example below, the key file's certificate label would be **ibmwebspheremqmqm_rhel5** for queue manager QM_RHEL5.

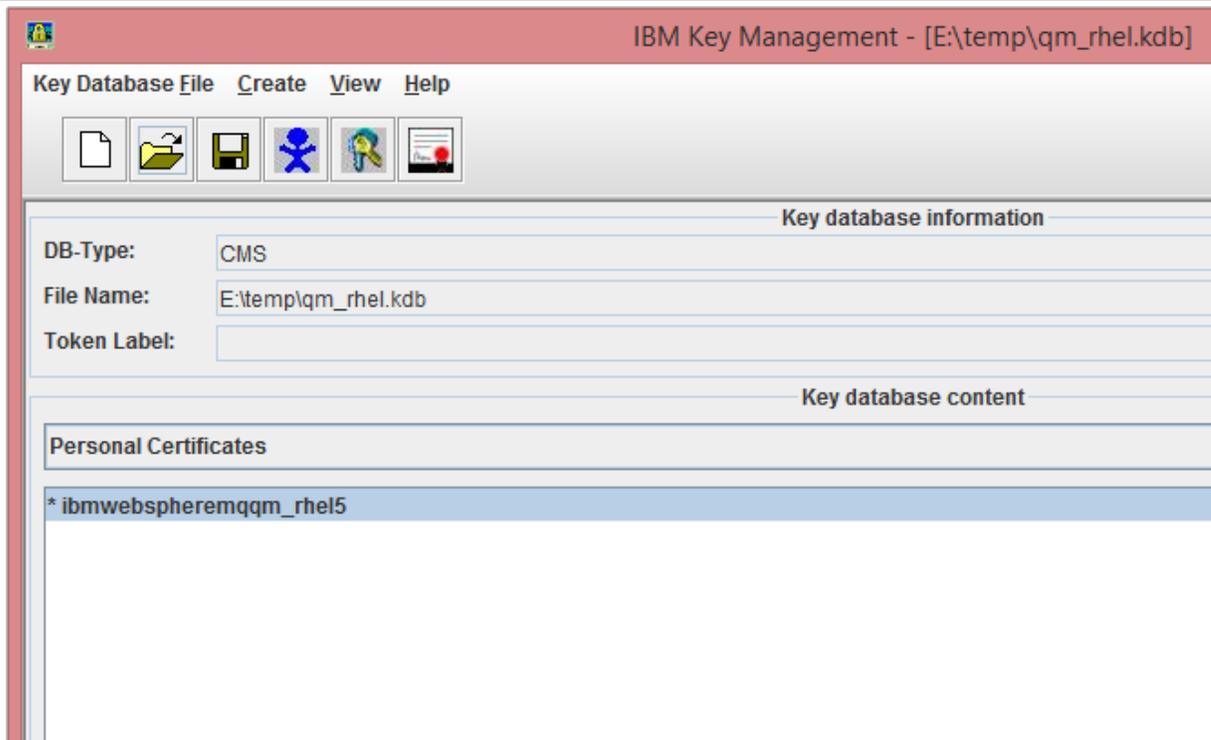


Figure 9-8. Server Key File Certificate

- b. **Client:** On the client side used by nsqcm, the same certificates will need to have the following label in all lowercase: **ibmwebspheremq{\$username}**. The username specified is the username that the program is running under. In the example below, nsqcm runs under the user rconl_000, so the client label would be named **ibmwebspheremqrconl_000**.

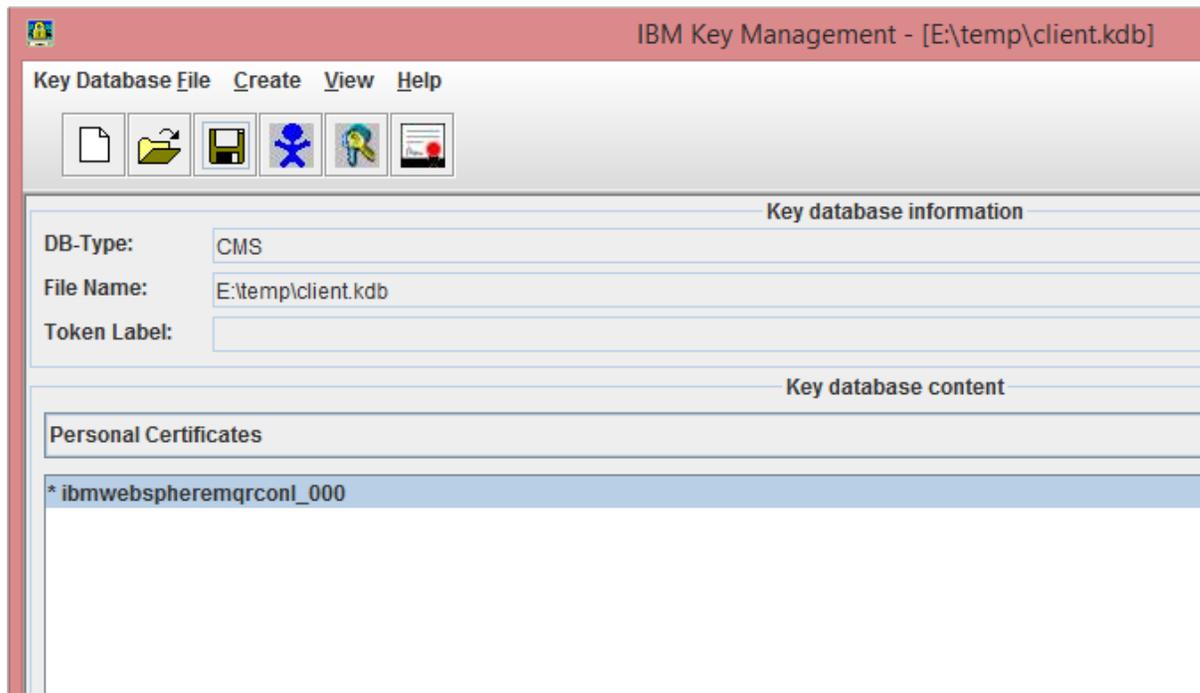


Figure 9-9. Client Key File Certificate

Chapter 10: Java Agent/Connection Manager for IBM MQ

In version 10.x, nsqcmmq.jar combines the M6-WMQ Connection Manager (nsqcm) and M6-WMQ Agent (nsqmq) from versions 6.7.x and earlier. Refer to the previous chapter for information on the version 6 Connection Manager for IBM MQ.

Version 10.x nsqcmmq.jar can be run in CM or Agent mode. In CM mode, commands can be issued on remote machines. (SSH is required.)

The Java Agent and Connection Manager do not support console commands. Some commands were moved to configuration (properties) file. For example, the +U/-U option is now defined in the properties file. A sample is provided below.

```
##### +U/-U Option #####
# GET and PUT with AlteruserId
# +U = runs with AlteruserId and -U runs without AlteruserId
```

```
command.passuser = false
```

```
#####
```

10.1 Requirements

The Java Agent/Connection Manager requires workgroup server version 10.5.

10.2 Starting the Java Agent on Linux

Starting the Agent:

```
/opt/mqm/java/jre64/jre/bin/java -
Djava.library.path=/opt/mqm/java/lib64 -Xmx2g -Xms2g -Xss256k -
Dlogback.configurationFile=logback.xml -jar nsqcmmq.jar
```

10.3 nsqcmmq.properties File Structure

The following subsections provide details about the nsqcmmq.properties configuration file, first as set up for Agent mode, then for Connection Manager (CM) mode.

10.3.1 Agent Mode

The nsqcmmq.properties file contains a number of configuration blocks, all listed below. The first three blocks are required: Application Configuration, Global mode, and Native command mapping to PCF Commands for execution. Other optional blocks may be added, such as IBM MQ Connection Pool, IBM MQ Authorization Verification, IBM MQ read message timeout setup, Java Agent port and name configuration, Workgroup server

configuration, ServerSocket application configuration, and IBM MQ Queue Manager list, and IBM MQ Event reading and delivery.



Please note that SSH settings are intended to be used for CM mode only, not for Agent mode. Any SSH settings in the Agent nsqcmmq.properties config file (such as user and password) overwrite the SSH settings on the GUI SSH tab.

- Application configuration:
 - trace - True or false. Used to trace PCF messages and print the Java Stack in the Agent log file based on logback configuration settings. There are three modes of agent logging: Support, Troubleshooting, and Package Trace. The logback.xml code for each one can be found below.

Support Mode

```
<property scope="context" file="./nsqcmmq.properties" />
```

```
<if condition='property("trace").equals("true")'>
<then>
  <root level="INFO">
    <!-- <appender-ref ref="STDOUT" /> -->
    <appender-ref ref="FILE" />
  </root>
</then>
<else>
  <root level="WARN">
    <!-- <appender-ref ref="STDOUT" /> -->
    <appender-ref ref="FILE" />
  </root>
</else>
</if>
```

Troubleshooting Mode

```
<property scope="context" file="./nsqcmmq.properties" />
```

```
<if condition='property("trace").equals("true")'>
<then>
  <root level="TRACE">
    <!-- <appender-ref ref="STDOUT" /> -->
    <appender-ref ref="FILE" />
  </root>
</then>
</if>
```

```

        </root>
    </then>

    <else>
        <root level="WARN">
            <!-- <appender-ref ref="STDOUT" /> -->
            <appender-ref ref="FILE" />
        </root>
    </else>
</if>

```

Package Trace Mode

```
<property scope="context" file="./nsqcmmq.properties" />
```

```

<if condition='property("trace").equals("true")'>
    <then>
        <logger name="com.mypackage.*" level="TRACE">
            <appender-ref ref="FILE" />
        </logger>
    </then>
    <else>
        <root level="WARN">
            <!-- <appender-ref ref="STDOUT" /> -->
            <appender-ref ref="FILE" />
        </root>
    </else>
</if>

```

See also <https://www.slf4j.org/extensions.html>.

- encryption - True or false. Encrypts data exchange between the workgroup server and the Agent.
- trusted.ip - allows requests from trusted IP addresses. Regex values can be used to define multiple addresses (for example: .*, 127\.\0\.\0\.\1, 192\.\168\.\1\.* , 192\.\168\.\1\.\2|192\.\168\.\1\.\3). Refer to the following Oracle documentation for more information on masking using regex expressions: <https://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html>.

- broadcast - True or false. This option allows automatic Agent node registration for all workgroup Servers that listen to broadcast messages by UDP protocol. This option can be helpful if you have many workgroup servers on different IP addresses and saves time for configurations with multiple nodes.

Default configuration settings example

```
##### Common Agent application settings #####  
trace=false  
  
# Encrypt PCF messages  
encryption=false  
  
#Broadcast flag  
broadcast=false  
  
# Regexp value for trusted IPs, e.g.:  
## .* - allow all  
trusted.ip=.*
```

- Define the global mode for the application: CM or Agent. This setting determines the mode for all queue managers.
 - cmMode=false indicates that you are running the application in Agent mode.
- Map commands from the application to commands in your script file. Some shell commands are used to execute actions. Examples are the commands to stop, start, delete, and create Queue Managers, and the RDQMSTATUS command, which collects data on the status of HA and RDQM Clusters. Those exe files could be wrapped in cmd or shell scripts.

There is a flexible way to map commands. @PARAMS @QMNAME maps commands from the application to commands in your script file for your operation system script processing. You have the opportunity to trace these parameters in a log file.

See the examples below, in which commands are mapped to exe files.

Example of command cmd or Unix shell mapping

```
EXCMD_START_Q_MGR = strmqm @PARAMS @QMNAME  
Q_MGR_START_ALL_CMD = mqmstart @PARAMS @QMNAME  
EXCMD_STOP_Q_MGR = endmqm @PARAMS @QMNAME  
EXCMD_DELETE_Q_MGR = dltmqm @PARAMS @QMNAME  
EXCMD_CREATE_Q_MGR = crtmqm @PARAMS @QMNAME  
EXCMD_RDQMSTATUS = rdqmstatus -m @QMNAME
```

- IBM MQ Connection Pool. The connection pool for the MQ channels connection allows connections to be pooled for use and reuse. There are two pool connection modes: MODE_AUTO (default) and MODE_ACTIVE. The active, or manual mode, allows you to fine-tune parameters of the pool, such as mqTimeout, mqMaxConnections, and mqMaxUnusedConnections.

```
mqPoolMode = MODE_ACTIVE
```

```
mqTimeout = 36000
```

```
mqMaxConnections = 150
```

```
mqMaxUnusedConnections = 50
```

The automatic mode does not use the above parameters:

```
mqPoolMode = MODE_AUTO
```

For more information about connection pools in IBM MQ, review the IBM documentation using the link below:

[Supplying a different connection pool in WebSphere MQ classes for Java - IBM Documentation](#)

- IBM MQ Authorization Verification. The value in command.passuser (true or false) controls whether or not an authorization verification check is performed.

Command.passuser = true

When Command.passuser = true, Java CM takes an AlteruserId from PCF Message header (which originates from a UserId variable in the Nastel MSHDR header) and puts it into the IBM MQ PCF Command API for authorization purposes. IBM MQ then performs the user verification. Java CM then returns the response code to the WGS.

Command.passuser = false (default)

When command.passuser = false, the authorization verification check by the IBM MQ server is disabled. So, PCF commands are sent on behalf of the OS process owner.

- IBM MQ read message timeout setup. The IBM MQ timeout interval in config for reading a message from the queue is waitintervalListener. This option is used for the Java listener class. If the timeout threshold is exceeded, then the message in the queue is considered expired and will not be read. The default value is waitintervalListener = 200.

A second option, waitinterval, is used only for PCF message reply compatibility. The default value is waitinterval = 0.

- Java Agent port and name configuration:

Java Agent is a ServerSocket-based application that can receive and send PCF messages between the workgroup server and an IBM MQ Server.

The node name and port should match the same settings in the Nastel Navigator user interface.

- node.name - Java Agent (node) name. By default, this is the host name of the local machine (node.name=HOSTNAME).
- node.port - Java connection manager listening port. By default, node.port=5010.
- Workgroup server configuration: This is a client-based mechanism. On startup, the connection manager sends a registration request to the workgroup server which adds it to the managed nodes if it is new, or triggers discovery for existing nodes. The ID running connection manager must have the correct registration. Up to 5 workgroup servers can be connected.
 - wgs.host - WGS host (By default, this is the localhost)
 - wgs.service - WGS port. By default, wgs.service=4010.
 - wgs.name - WGS name

Here is an example of multiple workgroup servers:

```
# WGS host
#wgs.host=localhost
wgs.host=127.0.0.1
#wgs.host=11.0.0.12
# WGS port
wgs.service=4011
# WGS name
wgs.name=MQM
wgs2.host=127.0.0.1
# WGS port
wgs2.service=4020
# WGS name
wgs2.name=TEST_WGS
```

- ServerSocket application configuration:

Since Agent is a ServerSocket application, it inherits ServerSocket settings from the Netty Framework. The Netty Framework uses Oracle Java API. See Java Oracle documentation at <https://docs.oracle.com/javase/7/docs/api/java/net/ServerSocket.html>

- SO_BACKLOG=20.
- CONNECT_TIMEOUT_MILLIS=30000
- SO_SNDBUF=8388608
- SO_RCVBUF=8388608.
- SO_KEEPALIVE=true

- IBM MQ Queue Manager list for Agent and CM. The Agent reads the config path from the Windows registry, but UNIX has a hardcoded path for the QM list. That path is defined via the `mqs_ini` option. It can be modified.

```
#mqs_ini = C:/ProgramData/IBM/MQ/mqs.ini
```

- IBM MQ Event reading and delivery. The trace PCF Message command in IBM MQ Server traces commands through an event mechanism so Navigator can monitor IBM MQ commands via event messages from the IBM MQ Server. There are many IBM System queues that deliver many types of events, but Nastel redirects all events to one PUBSUB queue. Both Java Agent and CM have an option called `{ eventQueue }` to define which queue should be used to aggregate all events that are needed from IBM. This queue becomes the source from which events are read.

The queue is defined using `eventQueue= NASTEL.EVENT.QUEUE`

- Password encryption. The banking Industry is subject to security requirements that protect private data. All passwords in the config file must be encrypted. Therefore, encryption option is supported.
 - Encrypt passwords:
`java -jar nsqcmq.jar encrypt Nasteil@12345678`

Sample Config File

Below is an example of the `nsqcmq.properties` file for Agent mode. Lines with a preceding pound or hash sign (#) are comments or “template” example lines that may be removed if not needed.

```
#####  
#####  
  
# Copyright (c) 2022 Nastel Technologies, Inc. All Rights Reserved.  
  
#  
# This software is the confidential and proprietary information of Nastel  
# Technologies, Inc. ("Confidential Information"). You shall not disclose  
# such Confidential Information and shall use it only in accordance with  
# the terms of the license agreement you entered into with Nastel  
# Technologies.  
  
#  
# NASTEL MAKES NO REPRESENTATIONS OR WARRANTIES ABOUT THE SUITABILITY OF  
# THE SOFTWARE, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO
```

```
# THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
# PURPOSE, OR NON-INFRINGEMENT. NASTEL SHALL NOT BE LIABLE FOR ANY DAMAGES
# SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING
# THIS SOFTWARE OR ITS DERIVATIVES.
```

```
#
```

```
# CopyrightVersion 1.0
```

```
#####
#####
```

```
##### Application configuration #####
```

```
# Trace PCF communication messages
```

```
trace=false
```

```
#trace=true
```

```
# Encrypt PCF messages
```

```
##encryption=false
```

```
encryption=true
```

```
#Broadcast flag
```

```
broadcast=false
```

```
# Regexp value for trusted ips, e.g.:
```

```
## .* - allow all
```

```
## 127\.\0\.\0\.\1 - allow exactly one 127.0.0.1
```

```
## 192\.\168\.\1\.* - allow ip by regexp 192.168.1.*
```

```
## 192\.\168\.\1\.\2|192\.\168\.\1\.\3 - allow several ip 192.168.1.2 or 19..168.1.3
```

```
trusted.ip=.*
```

```
##### Global mode for All QM #####
```

```
#
```

```
cmMode = false
```

```
#
```

```
#####
```

```
#####  
##  
## Native commands mapping to PCF Commands for execution  
##  
#####  
  
EXCMD_START_Q_MGR = strmqm @PARAMS @QMNAME  
Q_MGR_START_ALL_CMD = mqmstart @PARAMS @QMNAME  
EXCMD_STOP_Q_MGR = endmqm @PARAMS @QMNAME  
EXCMD_DELETE_Q_MGR = dltmqm @PARAMS @QMNAME  
EXCMD_CREATE_Q_MGR = crtmqm @PARAMS @QMNAME  
  
#####  
##  
## Event queue setup  
##  
#####  
  
##### +U/-U Option #####  
# GET and PUT with Alteruserld  
# +U = runs with Alteruserld and -U runs without Alteruserld  
  
command.passuser = false  
#####
```

10.3.2 Connection Manager (CM) Mode

In CM mode, management is done using IBM MQ channels.

The `nsqcmmq.properties` file contains a number of configuration blocks, all listed below. The first three blocks are required: Application Configuration, Global mode, and Native command mapping to PCF Commands for execution. SSH information is required if you will be using CM mode to issue commands on remote machines. (Examples of these

commands would be starting or stopping queue managers and viewing log files.) Other optional blocks may be added, such as IBM MQ Event reading and delivery, IBM MQ Connection Pool, IBM MQ Authorization Verification, IBM MQ read message timeout setup, Java CM node configuration, Workgroup server configuration, ServerSocket application configuration, and Additional SSL connection parameters.

- Application configuration:
 - trace - True or false. Used to trace CM PCF messages and print the Java Stack in the CM log file based on logback configuration settings. This trace provides a very detailed log for each method that is running. See also <https://www.slf4j.org/extensions.html>.
 - encryption - True or false. Encrypts data exchange between the workgroup server and the CM.
 - trusted.ip - allows requests from trusted IP addresses. Regex values can be used to define multiple addresses (for example: .*, 127\.\0\.\0\.\1, 192\.\168\.\1\.*, 192\.\168\.\1\.\2|192\.\168\.\1\.\3). Refer to the following Oracle documentation for more information on masking using regex expressions: <https://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html>.
 - broadcast - True or false. This option allows automatic CM node registration for all workgroup Servers that listen to broadcast messages by UDP protocol. This option can be helpful if you have many workgroup servers on different IP addresses and saves time for configurations with multiple nodes.
- Define the global mode for the application: CM or Agent. This setting determines the mode for all queue managers.
 - cmMode=true indicates that you are running the application in CM mode.
- Map commands from the application to commands in your script file. Some shell commands are used to execute actions. Examples are the commands to stop, start, delete, and create Queue Managers, and the RDQMSTATUS command, which collects data on the status of HA and RDQM Clusters. Those exe files could be wrapped in cmd or shell scripts.

There is a flexible way to map commands. @PARAMS @QMNAME maps commands from the application to commands in your script file for your operation system script processing. You have the opportunity to trace these parameters in a log file.

See the examples below, in which commands are mapped to exe files.

Example of command cmd or Unix shell mapping

```
EXCMD_START_Q_MGR = strmqm @PARAMS @QMNAME
Q_MGR_START_ALL_CMD = mqmstart @PARAMS @QMNAME
EXCMD_STOP_Q_MGR = endmqm @PARAMS @QMNAME
EXCMD_DELETE_Q_MGR = dlrmqm @PARAMS @QMNAME
EXCMD_CREATE_Q_MGR = crtmqm @PARAMS @QMNAME
```

```
EXCMD_RDQMSTATUS = rdqmstatus -m @QMNAME
```

- IBM MQ Event reading and delivery. The trace PCF Message command in IBM MQ Server traces commands through an event mechanism so Navigator can monitor IBM MQ commands via event messages from the IBM MQ Server. There are many IBM System queues that deliver many types of events, but Nastel redirects all events to one PUBSUB queue. Both Java Agent and CM have an option called { eventQueue } to define which queue should be used to aggregate all events that are needed from IBM. This queue becomes the source from which events are read.

The queue is defined using eventQueue= NASTEL.EVENT.QUEUE

- IBM MQ Connection Pool. The connection pool for the MQ channels connection allows connections to be pooled for use and reuse. There are two pool connection modes: MODE_AUTO (default) and MODE_ACTIVE. The active, or manual mode, allows you to fine-tune parameters of the pool, such as mqTimeout, mqMaxConnections, and mqMaxUnusedConnections.

```
mqPoolMode = MODE_ACTIVE
```

```
mqTimeout = 36000
```

```
mqMaxConnections = 150
```

```
mqMaxUnusedConnections = 50
```

The automatic mode does not use the above parameters:

```
mqPoolMode = MODE_AUTO
```

For more information about connection pools in IBM MQ, review the IBM documentation using the link below:

[Supplying a different connection pool in WebSphere MQ classes for Java - IBM Documentation](#)

- IBM MQ Authorization Verification. The value in command.passuser (true or false) controls whether or not an authorization verification check is performed.

Command.passuser = true

When Command.passuser = true, Java CM takes an AlteruserId from PCF Message header (which originates from a UserId variable in the Nastel MSHDR header) and puts it into the IBM MQ PCF Command API for authorization purposes. IBM MQ then performs the user verification. Java CM then returns the response code to the WGS.

Command.passuser = false (default)

When command.passuser = false, the authorization verification check by the IBM MQ server is disabled. So, PCF commands are sent on behalf of the OS process owner.

- IBM MQ read message timeout setup. The IBM MQ timeout interval in config for reading a message from the queue is waitIntervalListener. This option is used for the

Java listener class. If the timeout threshold is exceeded, then the message in the queue is considered expired and will not be read. The default value is `waitintervalListener = 200`.

A second option, `waitinterval`, is used only for PCF message reply compatibility. The default value is `waitinterval = 0`.

- Java CM node configuration:
 - `node.name` - Java connection manager (node) name. By default, this is the host name of the local machine (`node.name=HOSTNAME`).
 - `node.port` - Java connection manager listening port. By default, `node.port=5010`.
- Workgroup server configuration: On startup, the connection manager sends a registration request to the workgroup server which adds it to the managed nodes if it is new, or triggers discovery for existing nodes. The ID running connection manager must have the correct registration. Up to 5 workgroup servers can be connected.
 - `wgs.host` - WGS host (By default, ,this is the localhost)
 - `wgs.service` - WGS port. By default, `wgs.service=4010`.
 - `wgs.name` - WGS name

Here is an example of multiple workgroup servers:

```
wgs.host=127.0.0.1
```

```
wgs.service=4010
```

```
wgs.name=MQM
```

```
wgs1.host=192.168.1.1
```

```
wgs1.service=4052
```

```
wgs1.name=MQM1
```

- ServerSocket application configuration:

Since CM is a ServerSocket application, it inherits ServerSocket settings from the Netty Framework. The Netty Framework uses Oracle Java API. See Java Oracle documentation at <https://docs.oracle.com/javase/7/docs/api/java/net/ServerSocket.html>

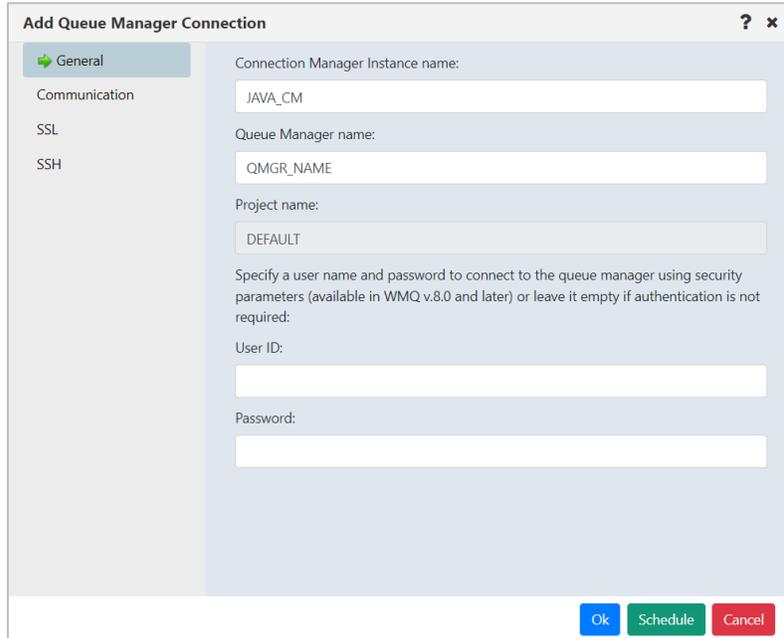
 - `SO_BACKLOG=20`.
 - `CONNECT_TIMEOUT_MILLIS=30000`
 - `SO_SNDBUF=8388608`
 - `SO_RCVBUF=8388608`.
 - `SO_KEEPALIVE=true`
- IBM MQ Queue Manager connection settings

You can define the connection settings in the GUI or a config file.

GUI Setup

To set up the connection in the user interface, from the Workspace dashboard, select the check box of the workgroup server connection in the list and choose **Create > Remote Queue Managers...** from the action menu. Then on the Remote Queue Manager Connections window, click **Add**.

- Fill in the connection information on the General and Communication tabs.



- In the version 10.x Java connection manager, the SSL parameters for IBM MQ SSL remote connections are different than they are for the C-based connection manager. The SSL Key Store Password on the SSL tab is required to continue to use the functionality from versions 6.7.x and earlier. For information about SSL parameters in the nsqcmmq.properties file, refer to the Additional SSL connection parameters list item below.
- To use the version 10.x remote command functionality, fill in the SSH tab. See also the IBM MQ Remote commands execution (define connections for the SSH Server) list item below.

Config File

Below is an example of the nsqcmmq.properties file for CM mode. Lines with a preceding pound or hash sign (#) are comments or “template” example lines that may be removed if not needed.

```
#####
#####
# Copyright (c) 2022 Nastel Technologies, Inc. All Rights Reserved.
#
# This software is the confidential and proprietary information of Nastel
# Technologies, Inc. ("Confidential Information"). You shall not disclose
```

```
# such Confidential Information and shall use it only in accordance with
# the terms of the license agreement you entered into with Nastel
# Technologies.
#
# NASTEL MAKES NO REPRESENTATIONS OR WARRANTIES ABOUT THE
# SUITABILITY OF
# THE SOFTWARE, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED
# TO
# THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
# PURPOSE, OR NON-INFRINGEMENT. NASTEL SHALL NOT BE LIABLE FOR ANY
# DAMAGES
# SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING
# THIS SOFTWARE OR ITS DERIVATIVES.
#
# Copyright Version 1.0
#####
#####
#####

##### Application configuration
#####
# Trace PCF communication messages
trace=false
#trace=true
# Encrypt PCF messages
##encryption=false
encryption=true
#Broadcast flag
broadcast=false
# Regexp value for trusted ips, e.g.:
## .* - allow all
## 127\0\0\0\1 - allow exactly one 127.0.0.1
## 192\168\1\.* - allow ip by regexp 192.168.1.*
## 192\168\1\2|192\168\1\3 - allow several ip 192.168.1.2 or 19..168.1.3
```

trusted.ip=.*

Global mode for All QM

#

cmMode = true

#

#####

#####

##

Native commands mapping to PCF Commands for execution

##

#####

EXCMD_START_Q_MGR = strmqm @PARAMS @QMNAME

Q_MGR_START_ALL_CMD = mqmstart @PARAMS @QMNAME

EXCMD_STOP_Q_MGR = endmqm @PARAMS @QMNAME

EXCMD_DELETE_Q_MGR = dltnmqm @PARAMS @QMNAME

EXCMD_CREATE_Q_MGR = crtmqm @PARAMS @QMNAME

#####

##

Event queue setup

##

#####

```
##### +U/-U Option #####  
# GET and PUT with AlteruserId  
# +U = runs with AlteruserId and -U runs without AlteruserId  
  
command.passuser = false  
#####
```

- Additional SSL connection parameters:

- sslMode
- sslCipher
- qmgrName
- connectionName
- channelName
- commandQName
- connStatus
- securityExit
- securityData
- connectUser
- connectPassword
- keyStore
- keyStorePassword
- trustStore
- trustStorePassword

Sample Configuration QM for CM without SSL

```
mq.sslMode = false  
mq.sslCipher =  
mq.qmgrName = TEST_QM  
mq.connectionName = 11.0.0.12(2414)  
mq.channelName = SYSTEM.DEF.SVRCONN  
mq.commandQName =  
mq.connStatus =  
mq.securityExit =  
mq.securityData =
```

Sample Configuration QM for CM with SSL

```
#mq2.sslMode = true  
#mq2.sslCipher =  
TLS_RSA_WITH_AES_128_CBC_SHA256  
#mq2.qmgrName = SSL_QM  
#mq2.connectionName = 11.0.0.12(2417)  
#mq2.channelName =  
SYSTEM.DEF.SVRCONN
```

```
mq.connectUser = mqUser           #mq2.commandQName =
mq.connectPassword = mqPassword   #mq2.connStatus =
mq.keyStore = client.jks          #mq2.securityExit =
mq.keyStorePassword = password    #mq2.securityData =
mq.trustStore = client.jks        #mq2.connectUser = user
mq.trustStorePassword = password  #mq2.connectPassword = zzzxxxxxx!
                                   #mq2.keyStore = C:\\Keys\\client.jks
                                   #mq2.keyStorePassword = password
                                   #mq2.trustStore = C:\\Keys\\client.jks
                                   #mq2.trustStorePassword = password
```

- IBM MQ Remote commands execution (define connections for the SSH Server). Since Java CM can work as a Java Agent, we have to define connections for the SSH Server for the UNIX Shell in order to execute UNIX Shell commands. When these connections are defined properly, and you click the button to create a QM in Navigator, then the CM is going to open a connection using the SSH protocol and execute a command (assuming the IP Address exists).

The config file includes a separate section for the SSH Connection list, as shown below.

```
#####
#
# SSH Connection list
#
# if remote QM IP match with SSH IP then cm will pick it up
#
#####
#ssh.user = nastel
#ssh.password = nastel
#ssh.passwordCrypto = R7yOAMgRaPWfp8JnBNaCrw==
#ssh.port = 22
#ssh.host = node3
#ssh.ip = 11.0.0.93

#ssh1.user = nastel
```

```
#ssh1.password = Nastel@12345678  
#ssh.passwordCrypto = S9zOSPwNuKCdj7KmAVbDYx==  
#ssh1.port = 22  
#ssh1.host = node2  
#ssh1.ip = 11.0.0.92
```

```
#ssh2.user = nastel  
#ssh2.password = Nastel@12345679  
#ssh.passwordCrypto = B4xPALdToRWIo3SmSTbJwr==  
#ssh2.port = 22  
#ssh2.host = node3  
#ssh2.ip = 11.0.0.91
```

- Password encryption. The banking Industry is subject to security requirements that protect private data. All passwords in the config file must be encrypted. Therefore, encryption option is supported.
 - Encrypt passwords:
java -jar nsqcmq.jar encrypt Nastel@12345678

Chapter 11: Connection Manager for TIBCO EMS

Connection Manager for TIBCO EMS, also referred to as "CM for EMS," is a jar file called `nsqcmems.jar`. All EMS connection information is passed from the workgroup server and is entered/modified either:

- Locally in the `nsqcmems.properties` EMS connection manager file.
- OR-
- From Nastel Navigator, using the **Create > Remote EMS Mangers** option from the workgroup server pop-up action menu. This is the most common method.

11.1 Installing TIBCO EMS for Unix and Windows

- Windows Installation: refer to the Windows [Package Installation](#)
- Unix Installation – refer to the Unix [Pack Installation](#)

11.2 Requirements

The requirements to add a TIBCO EMS server to Nastel Navigator are as follows:

- You must be running a TIBCO EMS server
- You must be running an EMS connection manager
- Your Nastel Navigator version must be 6.7.0 or higher



In Nastel Navigator Security Manager, there are permissions to add, modify, and delete remote EMS server information.

11.3 Starting the EMS Connection Manager

The EMS connection manager uses the following files which are included in the extracted package:

- `nsqcmems.jar`: the main file to start the EMS connection manager.
- `nsqcmems.properties`: the EMS connection manager configuration file. Configuration instructions are in the section immediately below.

The EMS connection manager can be started as follows:

- As a Java jar. Run `nsqcmems.jar` as follows:

```
java -jar ../nsqcmems.jar
```

It is recommended to copy the `nsqcmems.properties` file to a new subfolder for each unique configuration (for example, `EMS_test`, `EMS_prod`). Then run the above command from these locations. This way, if an updated sample is sent, the property file you are using will not be overwritten.

-OR-

- As a service on Windows machines by running the installer, `m6cmems.exe`.

The EMS connection manager stop command is as follows:

- on the EMS connection manager start console, press the **Ctrl + C** keys

-OR-

- kill the EMS connection manager process

11.4 nsqcmems.properties File Structure

The `nsqcmems.properties` file should be in the same location as `nsqcmems.jar`. But if it is in a different location, make sure that the environment variable `APMW_EMS_HOME` is set to the directory where the properties file is located. (When setting the `APMW_EMS_HOME` variable, include only the directory; do not include the file name.)

The `nsqcmems.properties` file contains five configuration blocks, all listed below. The first three blocks (EMS node configuration, workgroup server configuration, application configuration) are required whether you want to enter the EMS connection information locally within the `nsqcmems.properties` file or directly into Nastel Navigator. The last two blocks (Tibco EMS server configuration and additional SSL connection parameters) are only required if you are entering EMS connection information in the `nsqcmems.properties` file (and not Nastel Navigator).

- EMS node configuration:
 - `node.name` - EMS connection manager (node) name
 - `node.port` - EMS connection manager listening port
 - `node.ip` - Starts listener on provided IP. By default, `node.ip` is 0.0.0.0.
 - `report.ip` - Sends IP address to WGS. By default, `report.ip` is true.
- Workgroup server configuration: On startup, the connection manager sends a registration request to the workgroup server which adds it to the managed nodes if it is new, or triggers discovery for existing nodes. The ID running connection manager must have the correct registration. Up to 5 workgroup servers can be connected.
 - `wgs.host` - WGS host (IP or host name)
 - `wgs.service` - WGS port
 - `wgs.name` - WGS name
 - `wgs.user` - the user name (optional)

Here is an example of multiple workgroup servers:

```
wgs.host=127.0.0.1
wgs.service=4051
wgs.name=MQM
```

```
wgs1.host=192.168.1.1
```

```
wgs1.service=4052
```

```
wgs1.name=MQM1
```

- Application configuration:
 - trace - trace CM PCF messages
 - encryption - encrypt PCF messages
 - trusted.ip - regexp value for trusted IP's (e.g.: .*, 127\.\0\.\0\.\1, 192\.\168\.\1\.*, 192\.\168\.\1\.\2|192\.\168\.\1\.\3)
 - broadcast - send broadcast message on startup
 - tib.ems.admin - TIBCO EMS admin command line utility
- Tibco EMS server configuration:
 - server.url - TIBCO EMS server URL
 - user - username to authenticate to TIBCO EMS user
 - password - password to authenticate to TIBCO EMS user
 - encrypted.password - encrypted password to authenticate to TIBCO EMS user
- Additional SSL connection parameters:
 - ssl.trusted_certificates - one or more trusted certificates to the set of trusted certificates
 - ssl.identity - client-side identity
 - ssl.issuer - additional issuer certificates for the client certificate
 - ssl.password - password for the private key or the identity store
 - ssl.encrypted.password - encrypted password for the private key or the identity store
 - ssl.key - private key for the client certificate
 - ssl.hostname - common name in the server certificate
 - ssl.ciphers - set of cipher suites to be used by the SSL connection
 - ssl.vendor - SSL vendor to be used. Valid values are strings "j2se" or "entrust6."
 - ssl.verifyhostname - client should verify the common name in the server certificate
 - ssl.verifyhost - client should verify server certificate
 - ssl.trace - client trace is required
 - ssl.debug.trace - client debug trace is required

11.5 Remote EMS Managers Configuration in Nastel Navigator

Remote TIBCO EMS servers can be added to workgroup servers without changing the `nsqcmems.properties` configuration file (as explained in the section immediately above). Perform the following within Nastel Navigator:

1. Select the workgroup server. From the action pop-up menu, select **Create** > **Remote EMS Managers**.

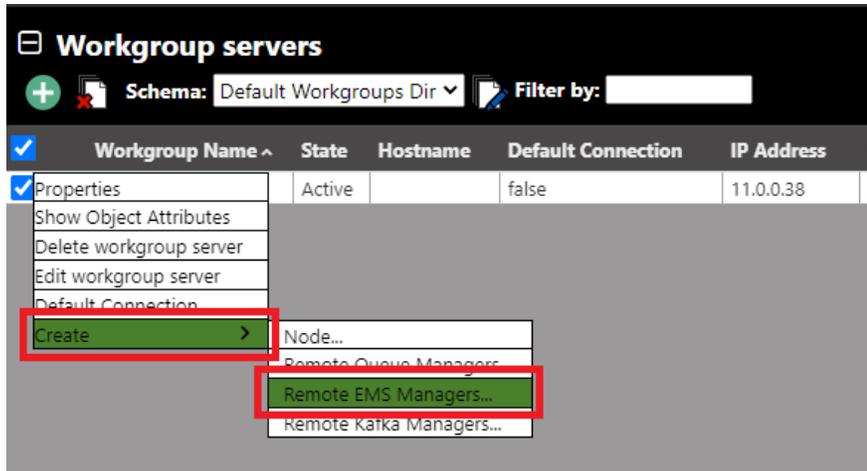


Figure 10-1. Adding Remote EMS Managers

2. The *Remote EMS Connections* window opens. Add the new TIBCO EMS server connection configuration information by clicking the **Add** button. You can also modify or delete existing connections.

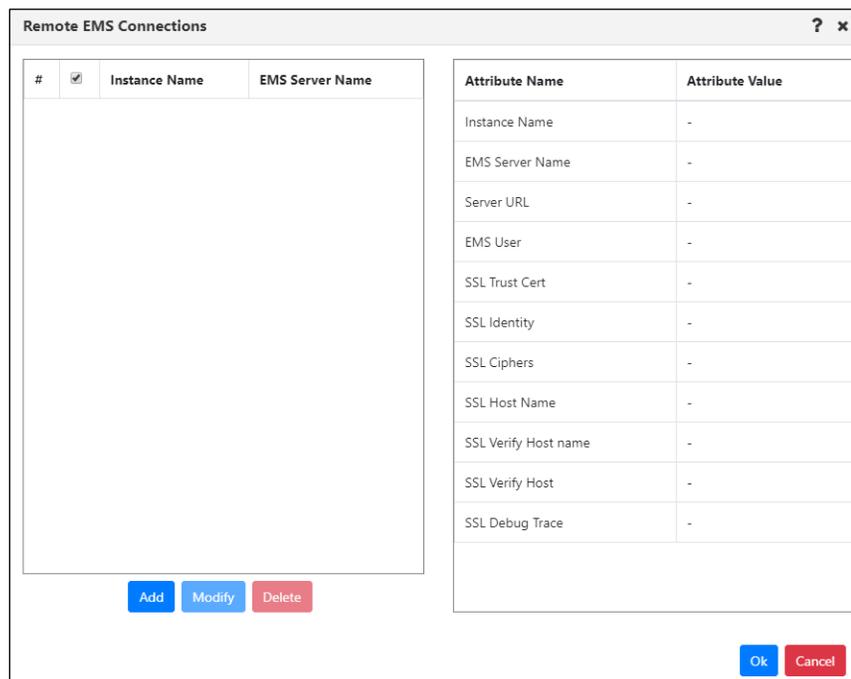


Figure 10-2. Add/Modify/Delete Remote EMS Connection Configuration

- When **Add** is clicked, the *Change EMS Manager Connection* window opens. Enter the EMS server connection configuration on the **General**, **SSL** and **SSL Options** tabs. The configuration parameters are the same whether you are configuring via the GUI or the `nsqcmems.properties` file.

When entering passwords (EMS and SSL), enter them as plain text. Passwords are masked when displayed onscreen and encrypted when saved to the database.

Click **Ok** once all parameters have been entered. Please note that once **Ok** is clicked, all fields can be updated except for **EMS Agent Instance Name** and **EMS Server Name**.

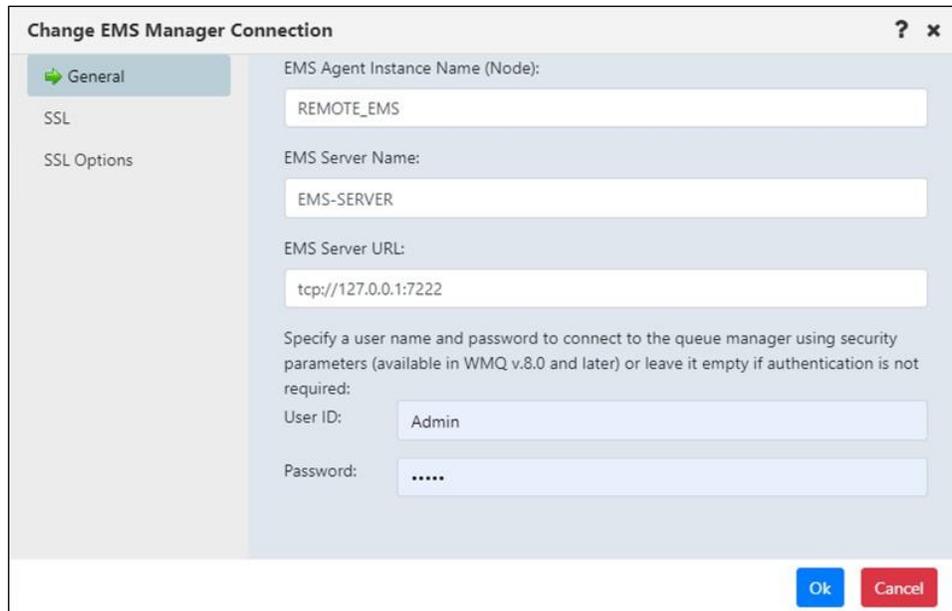


Figure 10-3. Set EMS Manager Connection Properties

- If running CM for EMS as a Windows service, it is recommended to run `nsqcmems.jar` in a console at first, so you can see any error messages it might display. Once everything is running correctly, you can run it as a service.

For more information on Remote EMS Managers, including descriptions of the fields on the dialogs displayed above, see the “Create Remote EMS Manager” section of the *Nastel Navigator User’s Guide*.

11.6 Adding Nastel Navigator EMS Connection Manager Nodes

If you need to connect to the EMS connection manager manually (by not translating data to the workgroup server via the `nsqcmems.properties` file), you can add a new node object. Perform the following in Nastel Navigator:

1. Select the workgroup server. On the pop-up action menu select **Create > Node**.

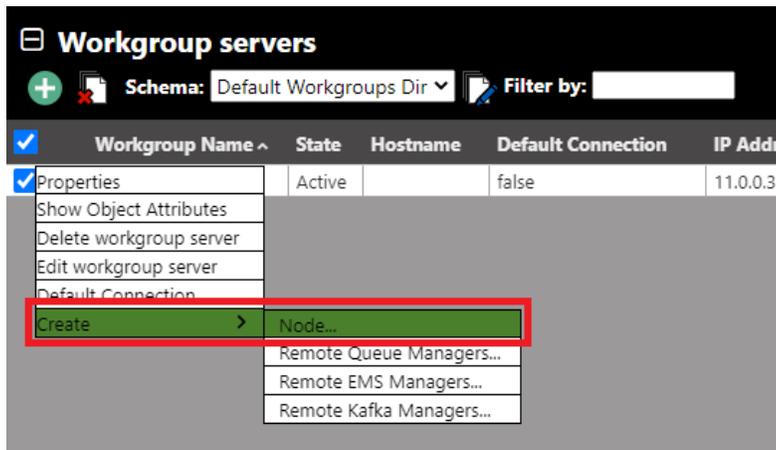


Figure 10-4. Add Nodes

2. The *Node Properties* window opens. Enter the node configurations and click **Ok**. For more information about Nodes in Nastel Navigator, see “Nodes” in the Viewlet section of the *Nastel Navigator User’s Guide*.

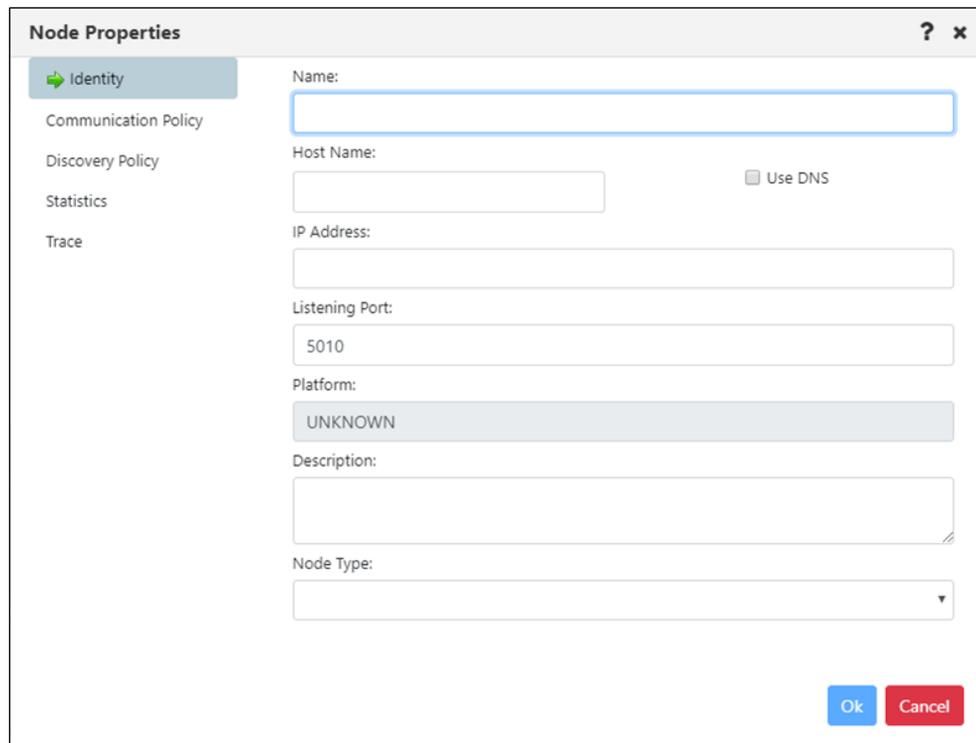


Figure 10-5. Node Properties

11.7 Add Nastel Navigator EMS Objects

Create a new viewlet to add an EMS object. On the *Create new EMS viewlet* screen, select **EMS** from the **Product** drop-down and select the object type on the left side of the screen.

Figure 10-6. Adding an EMS Manager Viewlet

A new viewlet of EMS managers is added:

Manager Name	Workgroup Name	Node Name	Estimated Response Time	State
EMS-SERVER	MQM	EMS-SERVER_QMA		Active
http://192.168.1.1:7332	MQM	EMS-SERVER_QMA		Active

Figure 10-7. EMS Manager Viewlet

For more information about creating dashboards and viewlets, see the “Dashboards” and “Viewlets” sections in the Using Nastel Navigator chapter of the *Nastel Navigator User’s Guide*.

11.8 Running nsqcmems with Simultaneous Connections to More than 1 WGS

The connection manager for TIBCO EMS has been designed to run on one workgroup server; it can only register with one workgroup server, specified in the properties file.

If you want to connect EMS simultaneously to multiple workgroup servers:

1. Change the `nsqcmems.jar` properties and re-run `nsqcmems`
2. Create it on the other workgroup server (right click **Workgroup Server**, select **Create Node**, and select **Node Type: EMS Agent-managed Node**).

Each workgroup server will need to have the remote EMS server settings added.

Chapter 12: Connection Manager for Kafka

This chapter discusses how to setup the connection manager for Kafka. All Kafka cluster connection information is passed from the workgroup server and is entered/modified either:

- Locally in the KafkaNode `nsqcmkafka.properties` file
- OR-
- From Nastel Navigator by using the **Create > Remote Kafka Managers** option from the workgroup server pop-up action menu. This is the most common method.

12.1 Installing Kafka for Unix and Windows

- Windows Installation: refer to the Windows [Package Installation](#)
- Unix Installation – refer to the Unix [Pack Installation](#)

12.2 Starting the Kafka Connection Manager

The Kafka connection manager uses the following files which are included in the extracted package:

- `nsqcmkafka.jar`: the main file to start the Kafka connection manager
- `nsqcmkafka.properties`: the Kafka connection manager configuration file. Configuration instructions are in the section immediately below.

The Kafka connection manager can be started as follows:

- As a Java jar. Run `nsqcmkafka.jar` as follows:

```
java -jar ../nsqcmkafka.jar
```

It is recommended to copy the `nsqcmkafka.properties` file to a new subfolder for each unique configuration (for example, `kafka_test`, `kafka_prod`). Then run the above command from these locations. This way, if an updated sample is sent, the property file you are using will not be overwritten.

-OR-

- As a service on Windows machines by running the installer, `m6cmkafka.exe`.

The Kafka connection manager stop command is as follows:

- on the Kafka connection manager start console, press the **Ctrl + C** keys
 - OR-
 - kill the Kafka connection manager process
-

12.3 Kafka Node Properties File Structure

There are five configuration blocks of the `nsqcmkafka.properties` file, all listed below. The first three configuration blocks (Kafka node configuration, workgroup server configuration, application configuration) are required whether you want to enter the connection information locally within the `nsqcmkafka.properties` file or directly into Nastel Navigator. The last two blocks (main Kafka cluster configuration and other Kafka cluster configuration) are only required if you are entering connection information in the `nsqcmkafka.properties` file (and not Nastel Navigator).

1. Kafka node configuration:
 - `node.name` - Kafka connection manager (node) name
 - `node.port` - Kafka connection manager listening port
 - `node.ip` - Starts listener on provided IP. By default, `node.ip` is 0.0.0.0.
 - `report.ip` - Sends IP address to WGS. By default, `report.ip` is true.
2. Workgroup server configuration: On startup, the connection manager sends a registration request to the workgroup server which adds it to the managed nodes if it is new, or triggers discovery for existing nodes. The ID running connection manager must have the correct registration. Up to 5 workgroup servers can be connected.
 - `wgs.host` - WGS host (IP or host name)
 - `wgs.service` - WGS port
 - `wgs.name` - WGS name

Here is an example of multiple workgroup servers:

```
8 ##### WGS configuration if need send data #####
9
0 # WGS host
1 wgs.host=127.0.0.1
2 wgs1.host=127.0.0.1
3 # WGS port
4 wgs.service=4040
5 wgs1.service=4060
6 # WGS name
7 wgs.name=WGS10
8 wgs1.name=WGS-ORACLE
9
```

Figure 12-1. Multiple WGS Configurations

3. Application configuration:
 - `trace` - trace connection manager PCF messages
 - `encryption` - encrypt PCF messages
 - `trusted.ip` - regexp value for trusted IPs (for example: `.*`, `127\.\0\.\0\.\1`, `192\.\168\.\1\.*`, `192\.\168\.\1\.\2|192\.\168\.\1\.\3`)

4. Main Kafka cluster configuration. All configuration values start with **kafka** and end with a possible configuration value. The following are some examples:
 - kafka.cluster.name - Kafka cluster name. Not used by Kafka (if not set, first bootstrap server will be used)
 - kafka.pool.timeout.ms - Kafka message pool timeout in ms. Not used by Kafka (default is 500 ms, increment when there are persistent message reading issues, or decrement on performance issues).
 - kafka.bootstrap.servers - Kafka cluster bootstrap servers (main value to enable Kafka cluster).
 - kafka.client.id - Kafka client ID (sets generated value if not defined)
 - kafka.request.timeout.ms - request timeout ID (sets to 30000 if not defined)
5. Additional Kafka cluster configurations can be configured. Values start with **kafka_<id>**. The following are examples:
 - kafka_1.cluster.name
 - kafka_1.bootstrap.servers

12.4 Remote Kafka Managers Configuration in Nastel Navigator

Kafka servers can be added to workgroup servers without changing the `nsqcmkafka.properties` configuration file (as explained in the section immediately above). Perform the following within Nastel Navigator:

1. Select the workgroup server. From the action pop-up menu, select **Create** > **Remote Kafka Managers**.

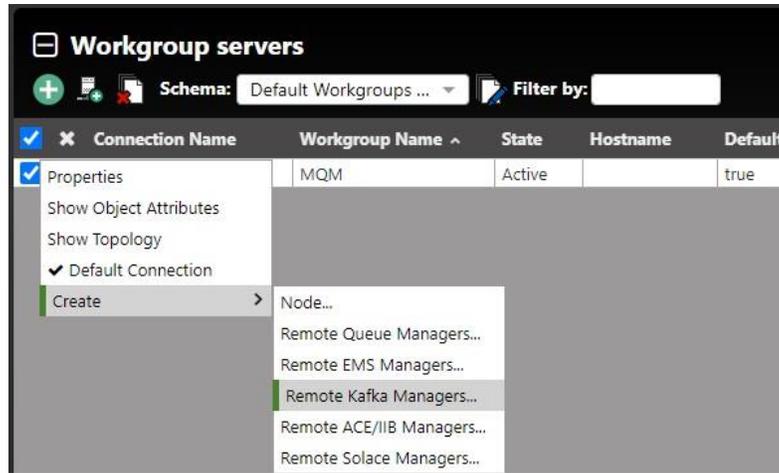


Figure 12-2. Add a Remote Kafka Manager

2. The *Remote Kafka Manager Connections* window opens. Add the new Kafka server connection configuration information by clicking the **Add** button. You can also modify or delete existing connections.

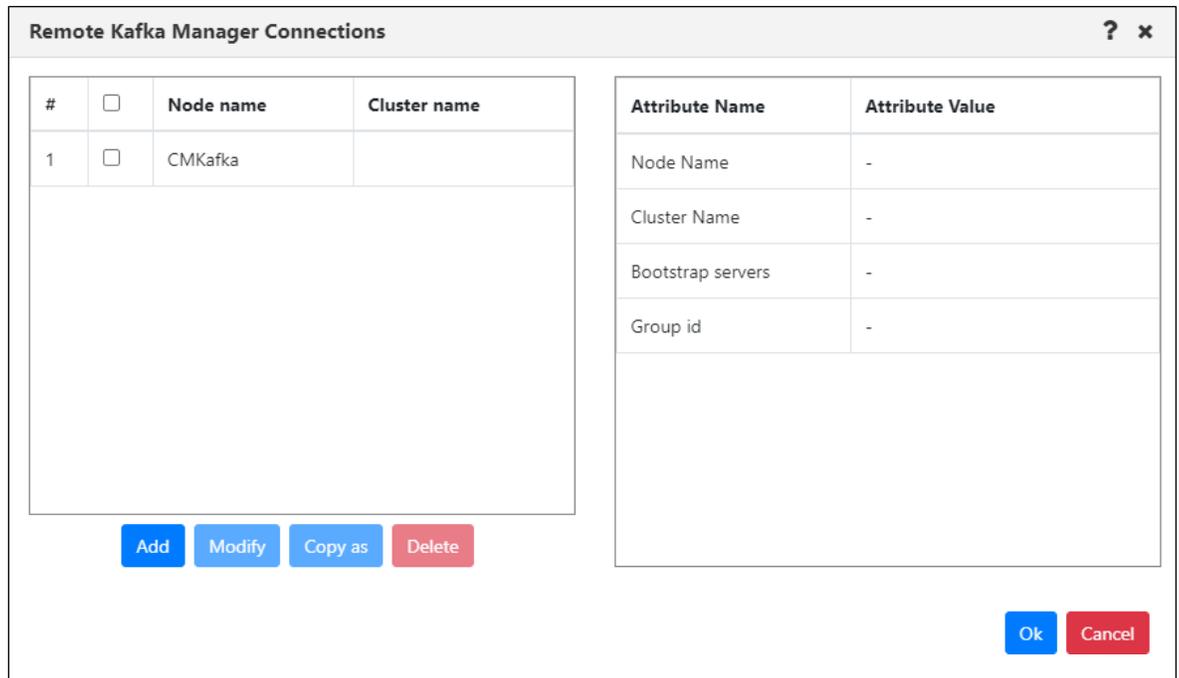


Figure 12-3. Remote Kafka Manager Connections

3. Once **Add** is clicked, the *Add Kafka Manager Connection* screen opens. Enter the Kafka server connection configuration on the **General** and **Config** tabs. Click **Ok** once all parameters have been entered.

For more information on Remote Kafka Managers, including SSL connection configuration, see the “Create Remote Kafka Manager” section of the *Nastel Navigator User’s Guide*.

12.5 Add Nastel Navigator Kafka Objects

Create a new viewlet to add Kafka objects. On the *Create new Kafka Viewlet* screen, select Kafka from the **Product** drop-down menu, and select the object type from the tabs on left side of the screen.

The screenshot shows the "Create New Kafka Topic Viewlet" form. The "Product" dropdown menu is highlighted with a red box and set to "Kafka". The form includes the following fields and options:

- Product:** Kafka (highlighted)
- Viewlet name:** TopicViewlet
- Workgroup server:** Primary Connection -...
- Temporary:**
- Node:** *
- Cluster:** *
- Object name:** *
- Schema:** Custom Viewlet Color (black) Flat Color:
- Schema Subject:** Project All
- Active attribute filtering:**
- Attribute filter:** + x
- Result limit:** 0

Buttons for "Save changes" and "Cancel" are located at the bottom right of the form.

Figure 12-4. Create New Kafka Viewlet

For more information about creating dashboards and viewlets, see the “Dashboards” and “Viewlets” sections in the Using Nastel Navigator chapter of the *Nastel Navigator User’s Guide*.

12.6 Kafka Configuration Values

- For admin client: <https://kafka.apache.org/documentation/#adminclientconfigs>
- For consumer: <https://kafka.apache.org/documentation/#consumerconfigs>

The following values are predefined by the KafkaNode and cannot be changed:

- key.deserializer=org.apache.kafka.common.serialization.StringDeserializer
- value.deserializer=org.apache.kafka.common.serialization.StringDeserializer
- max.poll.records=100
- enable.auto.commit=false
- auto.offset.reset=earliest
- For producer: <https://kafka.apache.org/documentation/#producerconfigs>

The following values are predefined by the KafkaNode and cannot be changed:

- key.serializer=org.apache.kafka.common.serialization.StringSerializer
- value.serializer=org.apache.kafka.common.serialization.StringSerializer

12.7 Kafka Connection Configuration Examples

The following are value sets for configuring Kafka cluster connections.

12.7.1 No Authentication Connection

When the Kafka server (broker) configuration contains listeners:

```
listeners=PLAINTEXT://:9092
#or
advertised.listeners=PLAINTEXT://192.168.1.132:9092
```

KafkaNode *nsqcmkafka.properties* must contain values:

```
#No configuration values is needed
```

Nastel Navigator Remote Kafka Managers must contain values:

```
#No configuration values is needed
```

12.7.2 Connection using authentication SASL/PLAIN

When Kafka server (broker) configuration contains listeners:

```
listeners=SASL_PLAINTEXT://:9092
#or
advertised.listeners=SASL_PLAINTEXT://192.168.1.132:9092

#additional configuration
security.protocol=SASL_PLAINTEXT
```

```
security.inter.broker.protocol=SASL_PLAINTEXT
sasl.mechanism.inter.broker.protocol=PLAIN
sasl.enabled.mechanisms=PLAIN
```

KafkaNode *nsqcmkafka.properties* must contain values:

```
kafka.security.protocol=SASL_PLAINTEXT
kafka.sasl.mechanism=PLAIN
kafka.sasl.jaas.config=org.apache.kafka.common.security.plain.PlainLogin
Module required username="xxx"
password="yyy";
```

Navigator Remote Kafka Managers must contain values:

```
security.protocol=SASL_PLAINTEXT
sasl.mechanism=PLAIN
sasl.jaas.config=org.apache.kafka.common.security.plain.PlainLoginModule
required username="xxx" password="yyy";
```

12.7.3 Connection using authentication SASL/SSL (connection to confluent.cloud)

When the Kafka server (broker) configuration contains listeners:

```
listeners=SASL_SSL://:9092
#or
advertised.listeners=SASL_SSL://192.168.1.132:9092

#additional configuration
security.protocol=SASL_SSL
security.inter.broker.protocol=SASL_SSL
sasl.mechanism.inter.broker.protocol=PLAIN
sasl.enabled.mechanisms=PLAIN
```

KafkaNode *nsqcmkafka.properties* must contain values:

```
kafka.ssl.endpoint.identification.algorithm=https
kafka.security.protocol=SASL_SSL
kafka.sasl.mechanism=PLAIN
kafka.sasl.jaas.config=org.apache.kafka.common.security.plain.PlainLogin
Module required username="xxx"
password="yyy";
```

Navigator Remote Kafka Managers must contain values:

```
ssl.endpoint.identification.algorithm=https
```

```
security.protocol=SASL_SSL
```

```
sasl.mechanism=PLAIN
```

```
sasl.jaas.config=org.apache.kafka.common.security.plain.PlainLoginModule  
required username="xxx" password="yyy";
```

Chapter 13: Connection Manager for IBM IIB/ACE

This chapter discusses how to set up the connection manager for IBM IIB/ACE.

13.1 Installing IIB/ACE for Unix and Windows

- Windows Installation: refer to the Windows [Package Installation](#)
- Unix Installation – refer to the Unix [Pack Installation](#)

13.2 Starting the IIB/ACE Connection Manager

The IIB/ACE connection manager uses the following files which are included in the extracted package:

- `nsqcmace.jar`: the main file to start the IIB/ACE connection manager
- `nsqcmace.properties`: the IIB/ACE connection manager configuration file. Configuration instructions are in the section immediately below.

The IIB/ACE connection manager can be started as follows:

- As a Java jar. Run `nsqcmace.jar` as follows:

```
java -jar ../nsqcmace.jar
```

It is recommended to copy the `nsqcmace.properties` file to a new subfolder for each unique configuration (for example, `IIBACE_test`, `IIBACE_prod`). Then run the above command from these locations. This way, if an updated sample is sent, the property file you are using will not be overwritten.

-OR-

- As a service on Windows machines by running the installer, `m6cmace.exe`.

The IIB/ACE connection manager stop command is as follows:

- on the IIB/ACE connection manager start console, press the **Ctrl + C** keys

-OR-

- kill the IIB/ACE connection manager process

13.3 IIB/ACE Properties File Structure

There are five configuration blocks of the `nsqcmace.properties` file, all listed below. Configure the properties file as follows:

1. ACE (IIB) node configuration:
 - `node.name` – IIB/ACE connection manager (node) name
 - `node.port` – IIB/ACE connection manager listening port
 - `node.ip` - Starts listener on provided IP. By default, `node.ip` is `0.0.0.0`.
-

- report.ip - Sends IP address to WGS. By default, report.ip is true.
2. Workgroup server configuration: On startup, the connection manager sends a registration request to the workgroup server which adds it to the managed nodes if it is new, or triggers discovery for existing nodes. The ID running connection manager must have the correct registration. Up to 5 workgroup servers can be connected.
 - wgs.host - WGS host (IP or host name)
 - wgs.service - WGS port
 - wgs.name - WGS name

Here is an example of multiple workgroup servers:

```

8 ##### WGS configuration if need send data #####
9
0 # WGS host
1 wgs.host=127.0.0.1
2 wgs1.host=127.0.0.1
3 # WGS port
4 wgs.service=4040
5 wgs1.service=4060
6 # WGS name
7 wgs.name=WGS10
8 wgs1.name=WGS-ORACLE
9

```

Figure 13-1. Multiple Workgroup Servers Configuration

3. Application configuration:
 - trace - trace connection manager PCF messages
 - encryption - encrypt PCF messages
 - trusted.ip - regexp value for trusted IPs (for example: .*, 127\.\0\.\0\.\1, 192\.\168\.\1\.* , 192\.\168\.\1\.\2|192\.\168\.\1\.\3)
4. Default ACE(IIB) integration configuration. This information can be entered here, or from Nastel Navigator by using the **Create > Remote ACE/IIB Managers** option from the workgroup server pop-up action menu (see next section for usage). Using the Nastel Navigator GUI is the most common method.

All configuration values start with **ace** and end with a configuration value. The following are examples:

- ace.name - ACE(IIB) integration name
 - ace.type - ACE(IIB) type: enter 1 for IIB or 2 for ACE
 - ace.url - ACE(IIB) integration (Web User Interface) URL
 - ace.user - ACE(IIB) integration username
 - ace.password - ACE(IIB) integration not encrypted user password
5. Additional ACE(IIB) integrations can be configured. Values start with **ace_<id>**. The following are examples:
 - ace_1.name

- ace_1.type
- ace_1.url

13.4 Remote IIB/ACE Managers Configuration in Nastel Navigator

IIB/ACE servers can be added to workgroup servers without changing the `nsqcmace.properties` configuration file (as explained in the section immediately above). Perform the following within Nastel Navigator:

1. Select the workgroup server. From the action pop-up menu, select **Create > Remote ACE/IIB Managers**.

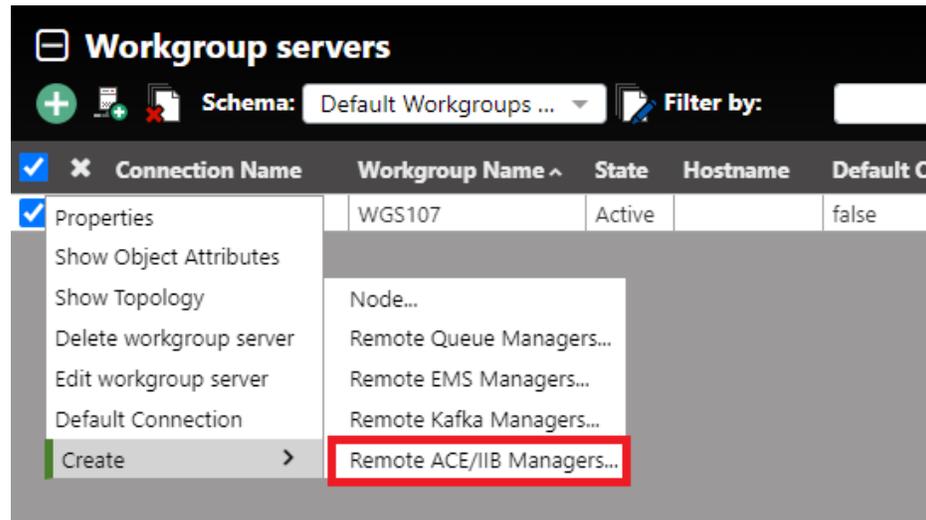


Figure 13-2. Add a Remote ACE/IIB Manager

- The *Remote IIB/ACE Manager Connections* window opens. Add the new IIB/ACE server connection configuration information by clicking the **Add** button. You can also modify or delete existing connections.

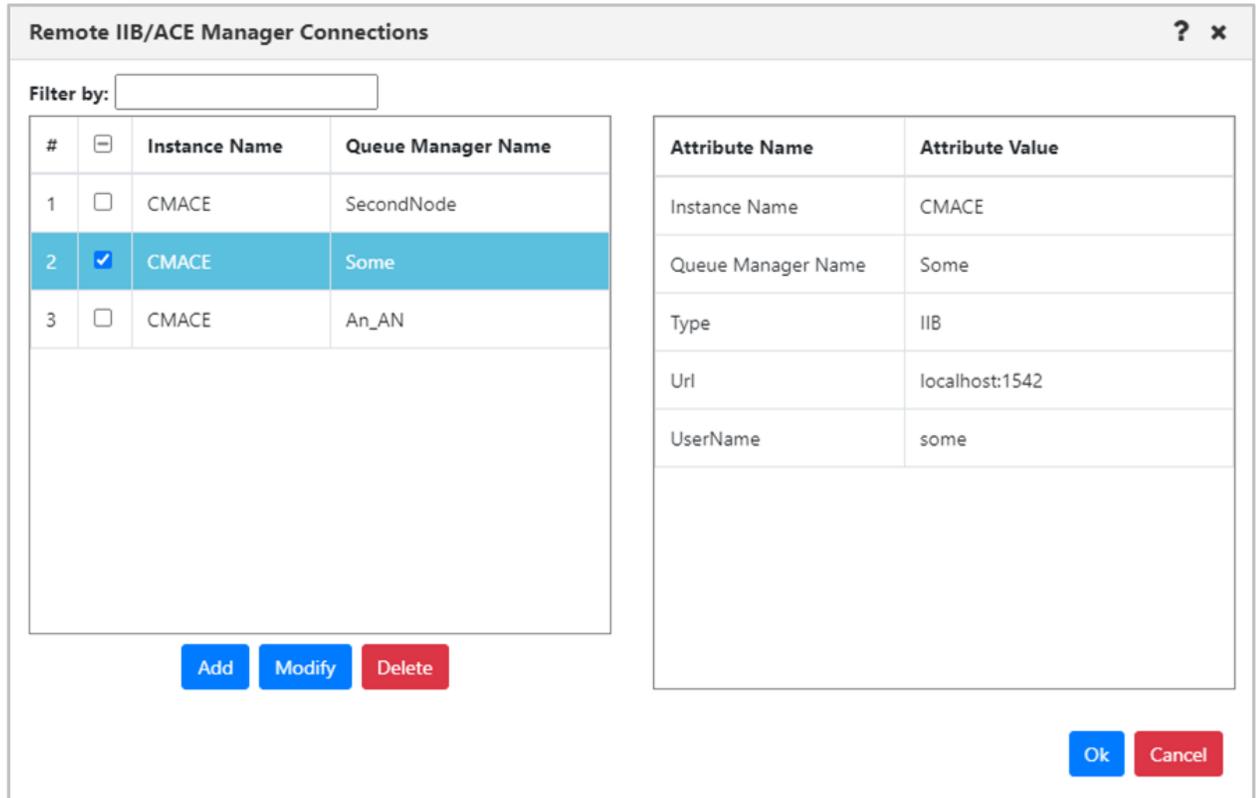
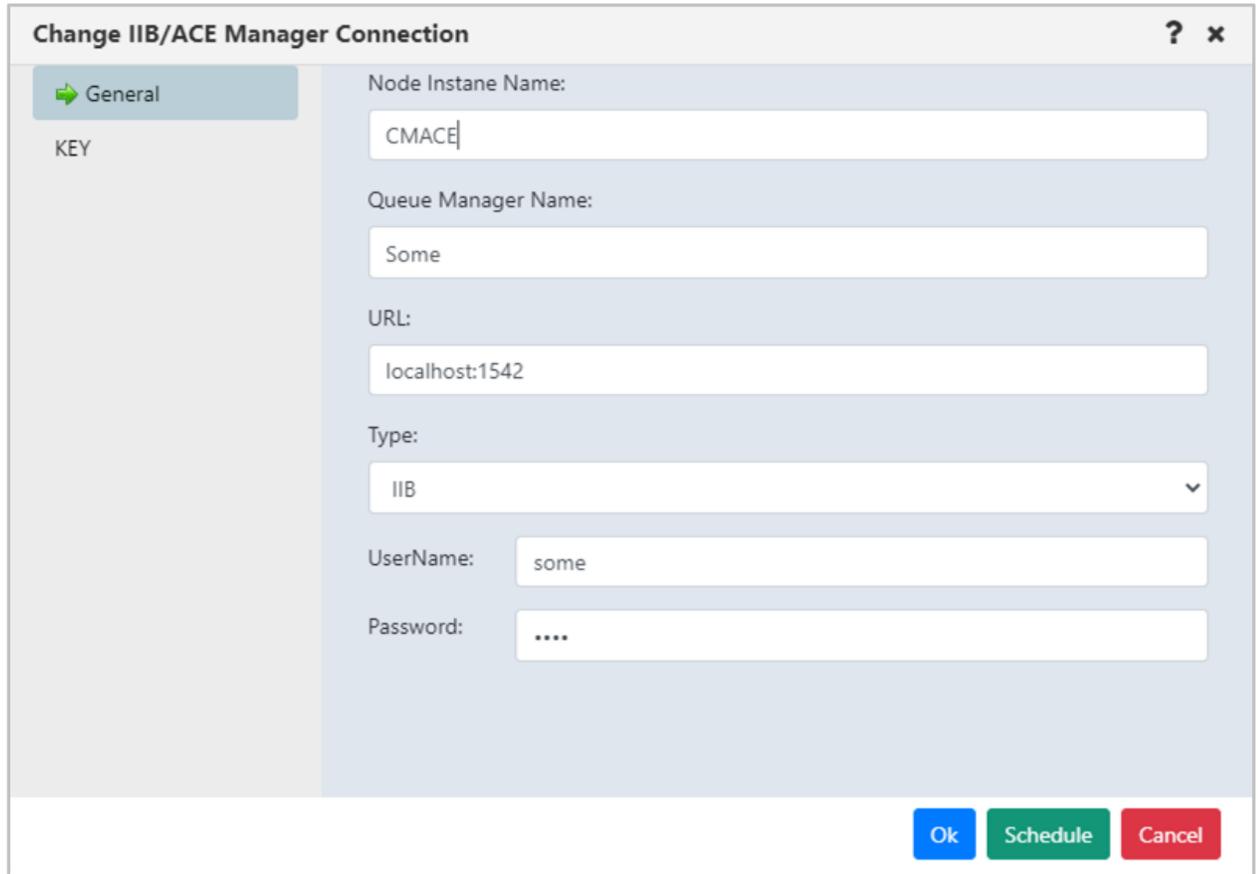


Figure 13-3. Remote IIB/ACE Manager Connections

3. Click the **Add** button. The *Change IIB/ACE Manager Connection* window opens. Enter the configurations for the new IIB or ACE connection manager on both the **General** and **Key** tabs. Click **Ok** when finished.



Change IIB/ACE Manager Connection

General
KEY

Node Instance Name:
CMACE

Queue Manager Name:
Some

URL:
localhost:1542

Type:
IIB

UserName: some

Password:

Ok Schedule Cancel

Figure 13-4. Change IIB/ACE Manager Connections

13.5 Adding Nastel Navigator Connection Manager Nodes

If you need to connect to the connection manager manually (by not translating data to the workgroup server via the `nsqcmace.properties` file), you can add a new node object. Perform the following in Nastel Navigator:

1. Select the workgroup server. On the pop-up action menu select **Create > Node**.

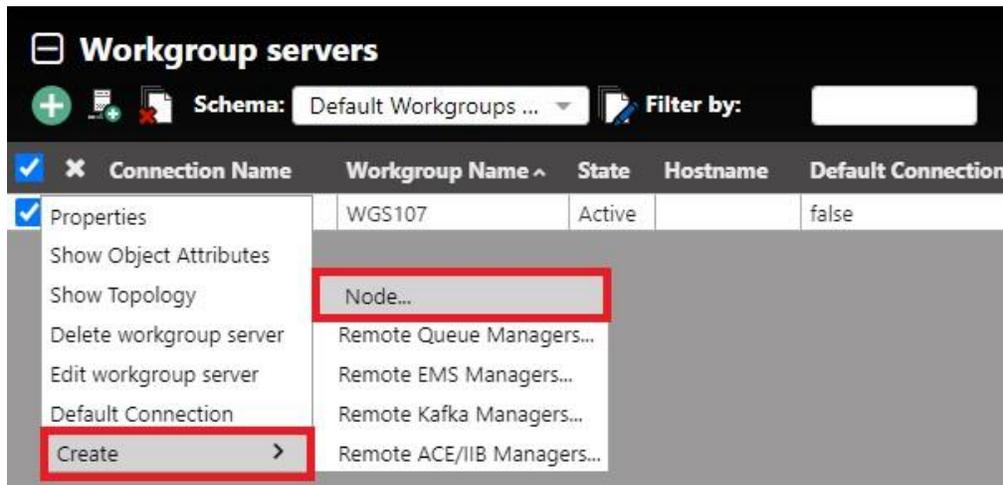


Figure 13-5. Create > Node

3. The *Node Properties* window opens. Select **ACE/IIB Agent-managed Node** from the **Node Type** drop-down. Enter all other node configurations and click **Ok**.

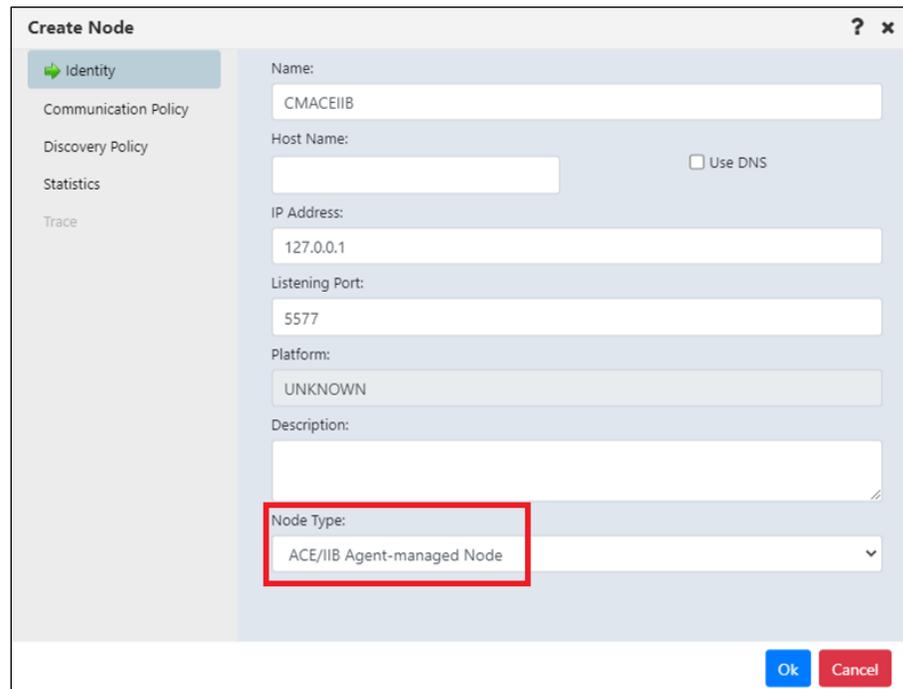


Figure 13-6. Create Node Window

For more information about Nodes in Nastel Navigator, see “Nodes” in the Viewlet section of the *Nastel Navigator User’s Guide*.

13.6 Add Nastel Navigator IIB/ACE Objects

Create a new viewlet to add IIB/ACE objects. On the *Create new Viewlet* screen, select either IIB or ACE from the **Product** drop-down menu. Select the object type from the left side of the screen and enter all configurations.

The screenshot shows a dialog box titled "Create new IIB Server viewlet". On the left is a vertical menu with categories: Product, Broker, Server, Application, Service, REST API, Library, Shared Library, Message Flow, Sub Flow, and Resource. The "Product" category is selected, and a dropdown menu shows "IIB" as the chosen option. The main area contains several input fields: "Viewlet name" (with a red error message "Viewlet name cannot be empty"), "Workgroup server" (set to "Default Connection - ..."), "Temporary" (checkbox), "Node" (with an asterisk), "Manager" (with an asterisk), "Object name" (with an asterisk), "Custom Viewlet Color" (checkbox and color picker), "Project" (set to "All"), "Active attribute filtering" (checked), "Attribute filter" (with "+" and "x" buttons), and "Result limit" (set to "100"). At the bottom right are "Save changes" and "Cancel" buttons.

Figure 13-7. Create New IIB or ACE Viewlet

For more information on ACE/IIB, including ACE and IIB viewlet samples and object actions, see “IIB Viewlets” and “ACE Viewlets” in the Viewlet section of the *Nastel Navigator User’s Guide*.

Chapter 14: Connection Manager for Solace

This chapter discusses how to set up the connection manager for Solace.

14.1 Installing Solace for Unix and Windows

- Windows Installation: refer to the Windows [Package Installation](#)
- Unix Installation – refer to the Unix [Pack Installation](#)

14.2 Starting the Solace Connection Manager

The Solace connection manager uses the following files, which are included in the extracted package:

- `nsqcmsolace.jar`: the main file to start the Solace connection manager
- `nsqcmsolace.properties`: the Solace connection manager configuration file. Configuration instructions are in the section immediately below.

The Solace connection manager can be started as follows:

- As a Java jar. Run `nsqcmsolace.jar` as follows:

```
java -jar ../nsqcmsolace.jar
```

It is recommended to copy the `nsqcmsolace.properties` file to a new subfolder for each unique configuration (for example, `Solace_test`, `Solace_prod`). Then run the above command from these locations. This way, if an updated sample is sent, the property file you are using will not be overwritten.

-OR-

- As of Nasted Navigator version 10.5, the option to run Solace as a service on Windows machines is not yet available.

The Solace connection manager stop command is as follows:

- on the Solace connection manager start console, press the **Ctrl + C** keys

-OR-

- kill the Solace connection manager process

14.3 Solace Properties File Structure

There are five configuration blocks of the `nsqcmSolace.properties` file, all listed below. Configure the properties file as follows:

1. Solace node configuration:
 - `node.name` – Solace connection manager (node) name
 - `node.port` – Solace connection manager listening port
 - `node.ip` - Starts listener on provided IP. By default, `node.ip` is `0.0.0.0`.
-

- report.ip - Sends IP address to WGS. By default, report.ip is true.
2. Workgroup server configuration: On startup, the connection manager sends a registration request to the workgroup server which adds it to the managed nodes if it is new, or triggers discovery for existing nodes. The ID running connection manager must have the correct registration. Up to 5 workgroup servers can be connected.
 - wgs.host - WGS host (IP or host name)
 - wgs.service - WGS port
 - wgs.name - WGS name

Here is an example of multiple workgroup servers:

```

8 ##### WGS configuration if need send data #####
9
0 # WGS host
1 wgs.host=127.0.0.1
2 wgs1.host=127.0.0.1
3 # WGS port
4 wgs.service=4040
5 wgs1.service=4060
6 # WGS name
7 wgs.name=MQM
8 wgs1.name=MQM1
9

```

Figure 14-1. Multiple Workgroup Servers Configuration

3. Application configuration:
 - trace - trace Solace communication messages
 - encryption - encrypt Solace PCF messages
 - trusted.ip - regexp value for trusted IPs (for example: .*, 127\.\0\.\0\.\1, 192\.\168\.\1\.* , 192\.\168\.\1\.\2|192\.\168\.\1\.\3)
4. Default Solace platform configuration. This information can be entered here, or from Nastel Navigator by using the **Create > Remote Solace Managers** option from the workgroup server pop-up action menu (see next section for usage). Using the Nastel Navigator GUI is the most common method.

All configuration values start with **solace** and end with a configuration value. The following are examples:

- solace.name - Solace platform name
 - solace.url - Solace platform (Web User Interface) URL
 - solace.user - Solace platform username
 - solace.password - Solace platform not encrypted user password
5. Additional Solace platforms can be configured. Values start with **solace_<id>**. The following are examples:

- solace_1.name
- solace_1.url

14.4 Remote Solace Managers Configuration in Nastel Navigator

Solace servers can be added to workgroup servers without changing the `nsqcmsolace.properties` configuration file (as explained in the section immediately above). Perform the following within Nastel Navigator:

1. Select the workgroup server. From the action pop-up menu, select **Create > Remote Solace Managers**.

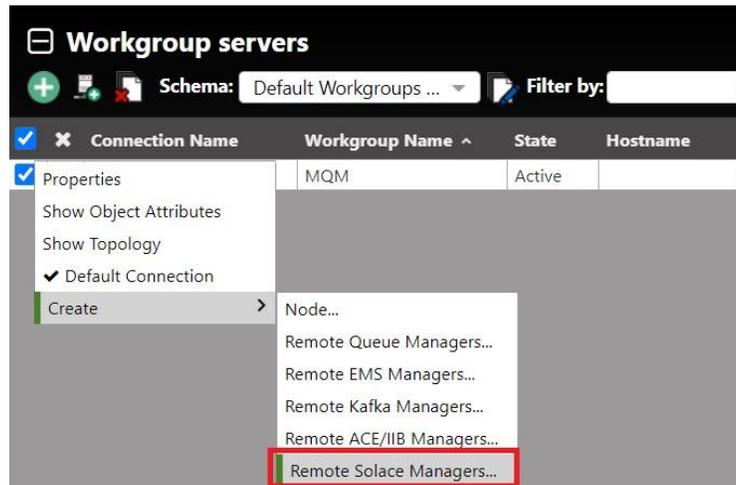


Figure 14-2. Add a Remote Solace Manager

2. The *Remote Solace Manager Connections* window opens. Add the new Solace server connection configuration information by clicking the **Add** button. You can also modify or delete existing connections.

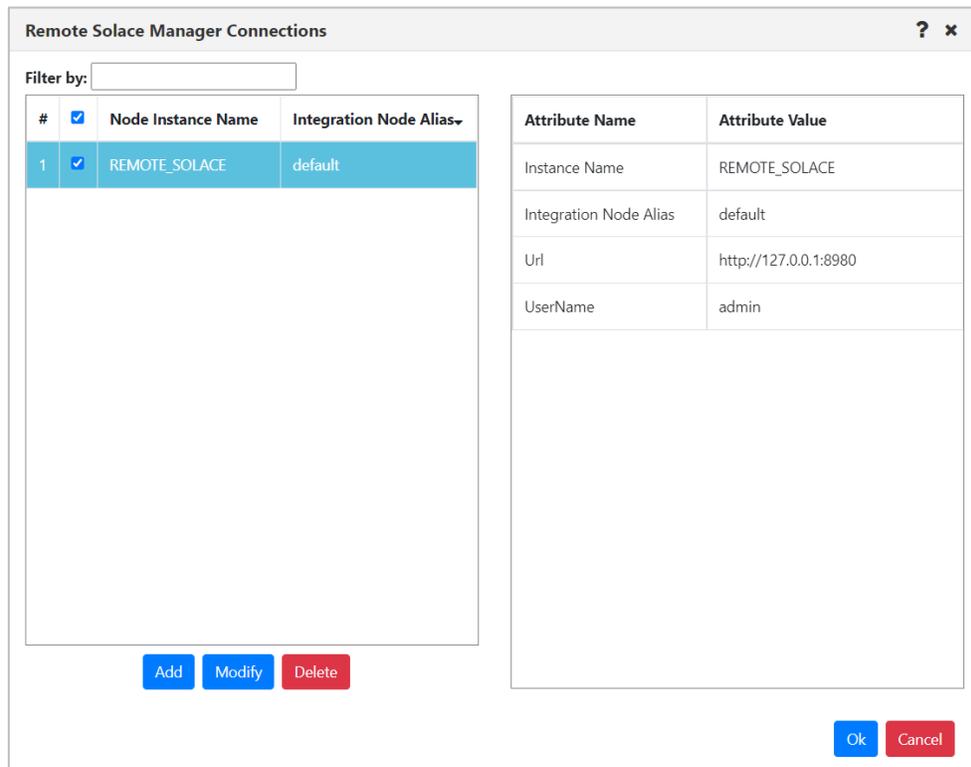


Figure 14-3. Remote Solace Manager Connections

3. Click the **Add** button. The *Add Solace Manager Connection* window opens. Enter the configurations for the new Solace connection manager on both the **General** and **Key** tabs. Click **Ok** when finished.

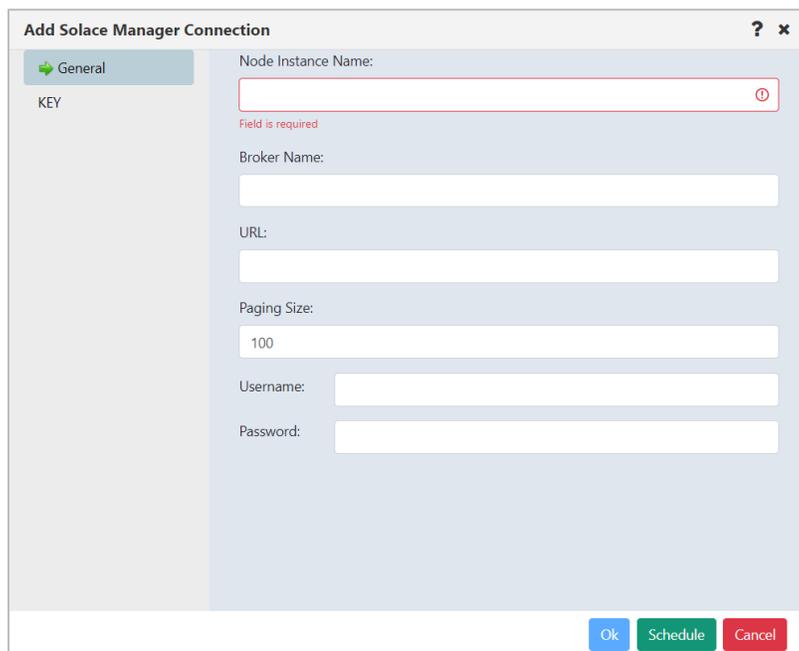


Figure 14-4. Add Solace Manager Connection

14.5 Adding Nastel Navigator Connection Manager Nodes

If you need to connect to the connection manager manually (by not translating data to the workgroup server via the `nsqcmsolace.properties` file), you can add a new node object. Perform the following in Nastel Navigator:

1. Select the workgroup server. On the pop-up action menu select **Create > Node**.

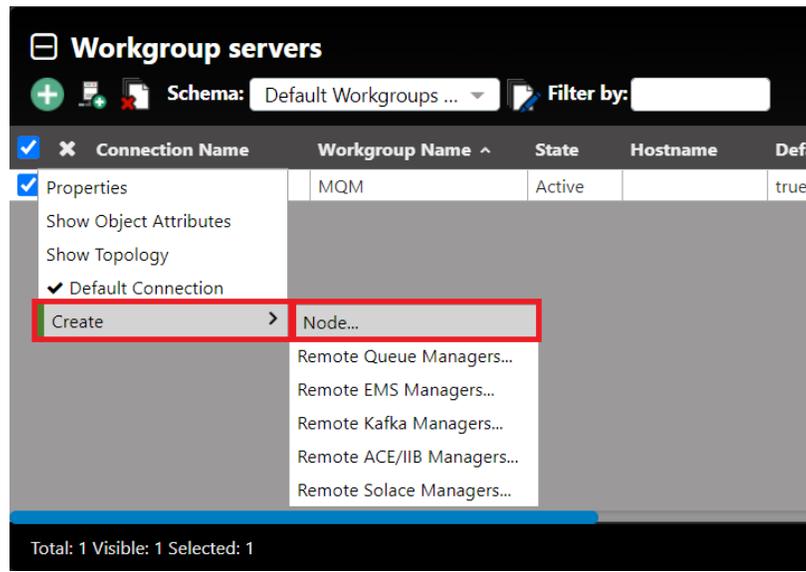


Figure 14-5. Create > Node

2. The *Node Properties* window opens. Select **Solace Agent-managed Node** from the **Node Type** drop-down. Enter all other node configurations and click **Ok**.

The screenshot shows the 'Node Create Window' interface. The sidebar on the left includes 'Identity', 'Communication Policy', 'Discovery Policy', 'Statistics', and 'Trace'. The main configuration area includes fields for Name (CMSolace), Host Name, IP Address (127.0.0.1), Listening Port (5010), Platform (UNKNOWN), and Description. A red box highlights the 'Node Type' dropdown menu, which is set to 'Solace Agent-managed Node'. The 'Use DNS' checkbox is unchecked. 'Ok' and 'Cancel' buttons are at the bottom right.

Figure 14-6. Create Node Window

For more information about Nodes in Nastel Navigator, see “Nodes” in the Viewlet section of the *Nastel Navigator User’s Guide*.

14.6 Add Nastel Navigator Solace Objects

Create a new viewlet to add Solace objects. On the *Create new Viewlet* screen, select Solace from the **Product** drop-down menu. Select the object type from the left side of the screen and enter all configurations.

The screenshot shows a dialog box titled "Create New Solace Message VPN Viewlet". On the left is a vertical menu with categories: Product, Node, Broker, Message VPN, Queue, Queue Template, Topic Endpoint, Topic Endpoint Template, Bridge, Client Profile, ACL Profile, Client UserName, JNDI Connection Factory, JNDI Queue, JNDI Topic, and Client Certificate. The "Product" dropdown is highlighted with a red box and set to "Solace". The "Viewlet name" field contains "Message VPN_Viewlet". The "Workgroup server" dropdown is set to "GEN_4010_127.0.0.1_...". The "Temporary" checkbox is checked. The "Node" dropdown is set to "*" and the "Broker" dropdown is also set to "*". The "Object name" field contains "*". There is a "Custom Viewlet Color" section with a black color swatch and a "Flat Color" checkbox. The "Project" dropdown is set to "All". The "Active attribute filtering" checkbox is unchecked. The "Attribute filter" field is empty with "+" and "x" buttons. The "Result limit" field contains "100". At the bottom right are "Save changes" and "Cancel" buttons.

Figure 14-7. Create New Solace Viewlet

For more information on Solace, including Solace viewlet samples, see “Solace Viewlets” in the Viewlet section of the *Nastel Navigator User’s Guide*.

Chapter 15: Nastel Navigator Component Installation in Multi-Instance Queue Manager Setup

15.1 Nastel Navigator Queues Unix

Refer to [section 4.4.1.1](#), Nastel Navigator Queues.



NOTE

Executing MQSC Scripts requires only one machine with an active instance of the queue manager. Refer to [Figure 15-1. Nastel Components Installation in Multi-Instance Queue Manager Setup](#).

15.2 Installing Nastel Navigator Components for Unix

Refer to [section 4.4.1](#), Configuring Agent on the MQServer.



NOTE

Repeat installation steps on both machines. Refer to [Figure 15-1. Nastel Components Installation in Multi-Instance Queue Manager Setup](#).

15.2.1 Configure mqgroup.ini File

Refer to [section 4.4.1.2](#), Configuring mqgroup.ini File.



NOTE

Repeat installation steps on both machines. Refer to [Figure 15-1. Nastel Components Installation in Multi-Instance Queue Manager Setup](#).

15.2.2 Starting and Stopping Nastel Navigator Components

Refer to section [4.5.1](#) and [4.5.2](#) for UNIX.

15.3 Installing Nastel Navigator Components for Windows

Refer to [section 3.3](#), Windows Installation.



NOTE

Repeat installation steps on both machines. Refer to [Figure 15-1. Nastel Components Installation in Multi-Instance Queue Manager Setup](#).

The mqsc scripts nastel.tst and mmf.tst require manual execution.

Select Custom Installation in section 3.3, [step 6](#) and continue with [step 8](#). In the Custom Setup list of program features select only Agents program features.

Continue with the rest of the in installation.

Refer to [section 3.4](#), Nastel Navigator Environment Variables to set start up options for the Local Agent and the Messages Server.

Refer to [section B.1.2](#) and [section B.2](#) for the full list of command line arguments applicable to the IBM MQ Agent and Message server.

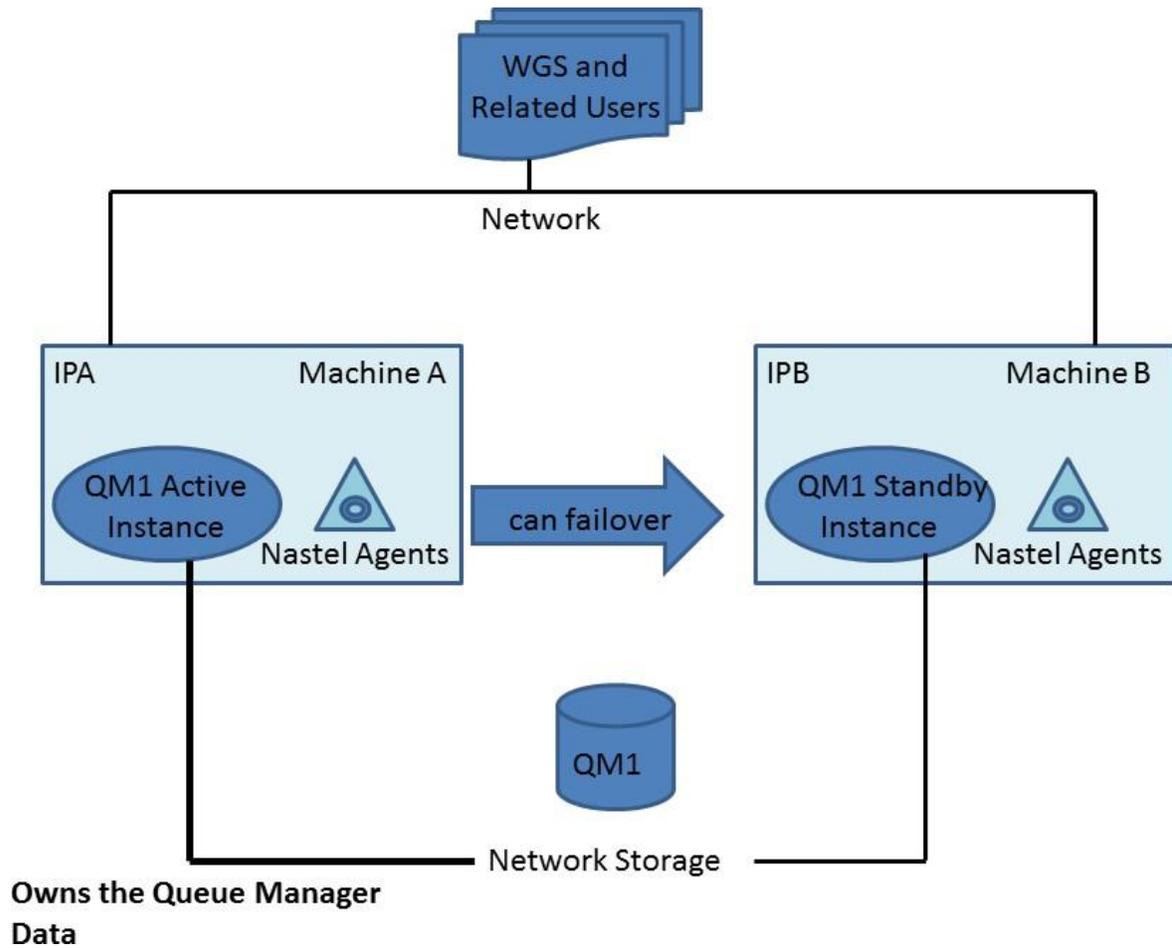


Figure 15-1. Nastel Components Installation in Multi-Instance Queue Manager Setup

Chapter 16: Z/OS Accounting and Statistics

(ZAS) Collector

Architecture Description

On z/OS, SMF (System Management Facility) is used to record information about WebSphere MQ when WMQ tracing is activated. (see Figure 16-1.) The SMF data set contains one or more files configured at the customer's site. Tracing starts with the first file in the data set. When it fills, the state of the file changes from ACTIVE to DUMP REQUIRED, an alternate SMF data set will become active, and an SMF exit occurs which activates the JCL NSQSMF in data set USER.PROCLIB.NSQSMF receives as input the name of the data set file that filled. NSQSMF then causes the dump to a temporary file.

After the dump, NSQSMF activates program NSQZAS, which reads SMF dump records from the temporary file and looks for records with type 116. These record types contain WMQ-related data as a sequence of subrecords. The type 116 records are written after an application disconnects from a queue manager or terminates. (In certain cases -- long running transactions -- records are written before the application ends.)

A subrecord type identifies the data as one of the following:

- WMQ SMF header
- Thread ID
- Thread-level accounting
- Queue-level accounting.

The type 116 records usually contain the above subrecords in that order. At times, when a record cannot hold the data for all the queues involved in an application connection to the queue manager, additional records may be written that contain the following:

- WMQ SMF header
- Thread ID
- Queue-level accounting

NSQZAS writes the ZAS PCF messages to a set of alias queues, NASTEL.ADMIN.XX.QUEUE (XX = ACCOUNTING OR STATISTICS) from which the messages are picked up by the M6-WMQ agent. The agent (possibly receiving PCF messages from the publisher NSQPUB, depending upon the configuration) sends the messages to the workgroup server, which writes them to a database. Nastel Navigator can then be used to retrieve and display the data.

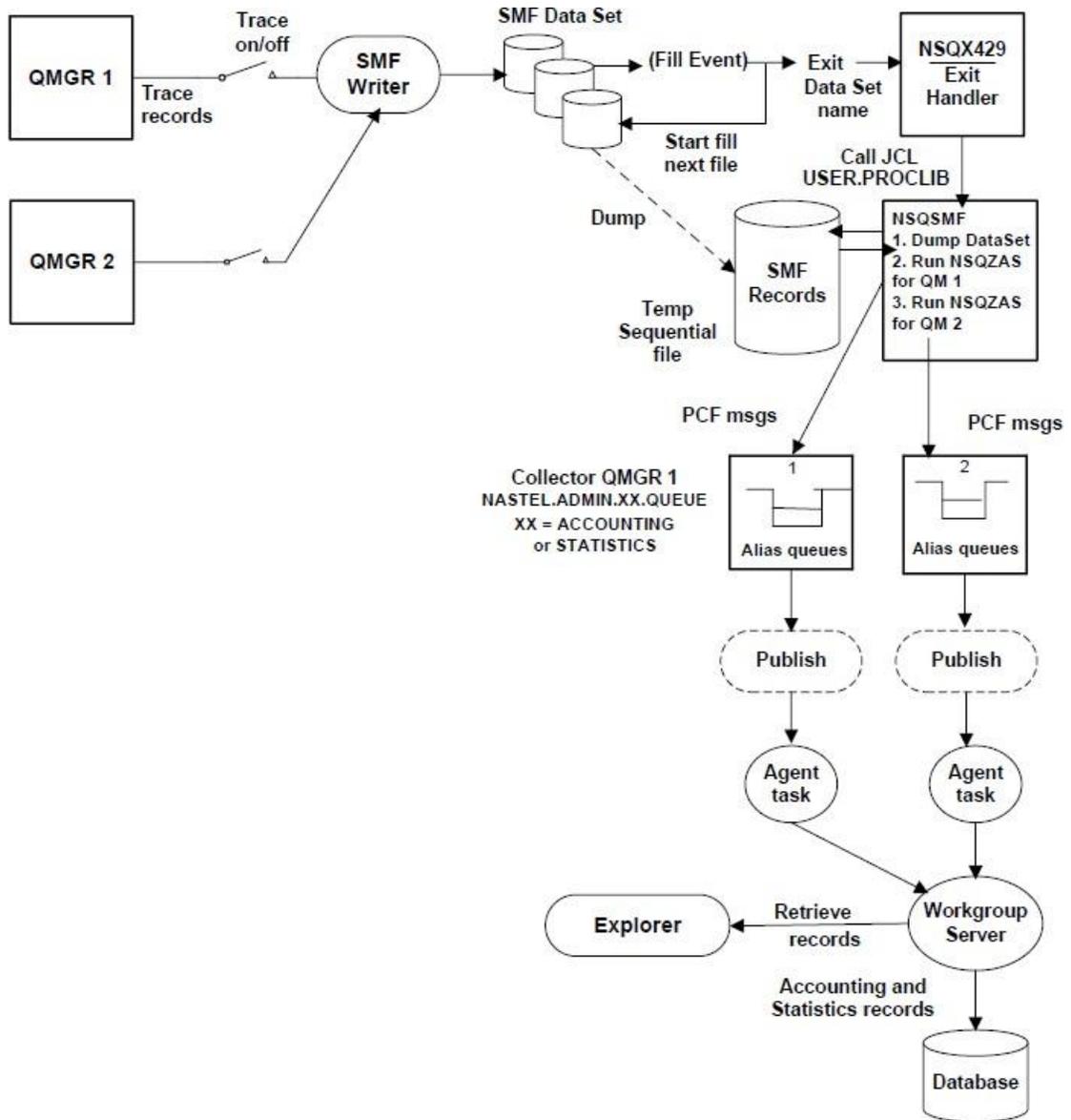


Figure 16-1. ZAS Flow Diagram

16.1 Configuring NSQZAS



Steps 1 and 2 are both required.

1. To control the SMF Exit through the Dynamic Exits Facility:

- a. Edit the system PARMLIB(PROGxx) file used for IPL and add the lines below (typically at the end of the file) that define the SMF IEFU29 exit module name and where the exit module resides.
- b. Replace ++hlq++ with the high-level qualifier for the AutoPilot M6-WMQ installation library. The 'FIRST' qualifier means that the AutoPilot exit must be first in a chain of vendor IEFU29 exits.

```

/*-----
/* DYNAMIC EXITS
/*-----

EXIT ADD EXITNAME (SYS.IEFU29) MODNAME (NSQXU29)
DSNAME(++hlq++.LOAD) FIRST

EXIT ADD EXITNAME (SYSSTC.IEFU29) MODNAME (NSQXU29)
DSNAME(++hlq++.LOAD) FIRST

```

2. To define the SMF Exit in SMFPRMxx”:

- a. To allow the system to invoke SMF exit IEFU29, define the exit in the SMF parmlib member PARMLIB (SMFPRMxx), where xx = 00 or other ID



Member PARMLIB (IEASYyy) should have a line SMF=nn, where nn is the suffix of the SMFPRMxx member in 1b above, and a line PROG=zz, where zz is the suffix of the PROGxx member in 1a above.

- b. Insert the range of SMF record types that are not to be written in the NOTYPE option of the SYS parameter. Ensure not to exclude type 116.
- c. Insert IEFU29 in the EXITS option of the SUBSYS parameter for STC.
- d. If your installation chooses not to define a SUBSYS parameter for STC, specify IEFU29 in the EXITS option of the SYS parameter. The data set names used by SMF may also be customized. The ACTIVE option should be defined versus NOACTIVE. The member should look similar to the Figure below.

```

ACTIVE                /* ACTIVE SMF RECORDING                */
DSNAME (SYS1.MAN1,
        SYS1.MAN2,
        SYS1.MAN3)
PROMPT                /* PROMPT OPERATOR                */
REC (PERM)            /* TYPE 17 PERM RECORDS ONLY      */
MAXDORM (3000)        /* WRITE IDLE BUFFER AFTER 30 MIN  */
STATUS (010000)      /* WRITE SMF STATS AFTER 1 HOUR    */
JWT (0400)           /* 522 AFTER 30 MINUTES           */

```

```

SID (&SYSNAME)
LISTDSN                               /* LIST DATA SET STATUS AT IPL          */
SYS (NOTYPE(14:19,62:69),EXITS (IEFU83,IEFU84,IEFACTRT,
                                IEFUSI,IEFUJI,IEFU29),NOINTERVAL,NODETAIL)
SUBSYS (STC,EXITS (IEFU29,IEFU83,IEFU84,IEFUJP,IEFUSO,IEFACTRT))

```

Figure 16-2. SMFPRMxx member

3. Insert Exit JCL into Procedure Library as follows:

JCL NSQSMF is called by the exit assembly language program NSQXU29. To find the JCL at run time, the JCL must be found in a searchable procedure library list.

- a. Edit JCL member ++hlq++.JCL (NSQSMF).
- b. Change the M6WMQ high level qualifier (HLQ) ++hlq++ and the WMQ HLQ ++mqser++.
- c. Add NSQZAS job sections, one per queue manager being monitored and edit the PARM options. These options are:
 - m List of qmgrs for which to collect acct/stats msgs
 - mc Qmgr to use by the collector for writing the acct/stats messages to forward to M6WMQ workgroup server

Note: -m and -mc values should be the same. To process multiple qmgrs, one NSQZAS job per qmgr is required. That is, replicate lines NSQZAS through SMFIN and edit the copied lines as required. All the NSQZAS jobs will process from the same temp SMF dump file.

```

-qa      Collect Queue Accounting on|off
-ma      Collect MQI Accounting on|off
-ms      Collect MQI Statistics on|off
-qs      Collect Queue Statistics on|off
-cs      Collect Channel Statistics on|off
-st n_secs  Statistics time: aggregate over n_secs. Default: 1800.
-t        Trace functions (stdout is file SYSPRINT)
-d        Dump SMF records in ascii/hex
-ds      Dump output to stdout. Default: file SMFDUMP.

```

- d. Copy it into a procedure library used on your system, such as USER.PROCLIB.

4. Configure queues used by Accounting/Statistics Collector as follows:

- a. Program NSQZAS writes PCF accounting and statistics messages to alias queues NASTEL.ADMIN.ACCOUNTING.QUEUE and NASTEL.ADMIN.STATISTICS.QUEUE.
- b. Configure the queues using MQSC member ++hlq++.JCL (MQMZAS) and command CSQUTIL. Refer to [z/OS Installation](#) for details of the command. Edit the MQSC command file first if you are using the M6-MQ publisher.



If you configure the M6-WMQ publisher and later decide not to use it, you must delete queue NASTEL.PUBSUB.EVENT.QUEUE, change the alias queue to point to NASTEL.EVENT.QUEUE and restart the agent.

5. Customize JCL procedures as follows:

There are several NSQZAS-related JCL procedures that need to be customized before they will work properly. Edit these JCLs in the ++hlq++.JCL data set:

- **NSQZAS** – select an SMF data set to dump and process with program NSQZAS
- **RUNZAS** – runs stand-alone test of program NSQZAS; variant of NSQZAS
- **AMQSMON** – runs modified IBM utility to do formatted browse/dump of messages on local queues NASTEL.ADMIN.ACCOUNTING (or STATISTICS).QUEUE
- **RUNCDMP** – runs IBM utility to do a formatted dump of type 115 and 116 records in an SMF dump file.

6. Create special datasets:

- a. Create data set NASTEL.MQSMF.CNTL and copy into it JCL member ++hlq++.JCL (MQSMF). That file contains control information for the SMF dump processing:

```
INDD (DUMPIN, OPTIONS (DUMP) )
OUTDD (DUMPOUT, TYPE (116) )
```

- b. Create GDG Data Set

Create data set NASTEL.MQSMF.DATA used to hold GDG (Generation Data Group) files that will be created by JCL NSQZAS:

```
SUBMIT ++hlq++.JCL(SMFGDG)
```

7. Configure system parameter as follows:

You can specify the interval at which WMQ collects statistics and accounting data in one of two ways (Refer to the *Using System Management Facility in InfoCenter for MQ 8.0* link in [Appendix A: Nastel and IBM Documentation](#)):

- a. Specify a value of STATIME in the system parameters using CSQ654SP.
- b. Specify a value for STATIME and use the SMF global accounting interval. Refer to *Using CSQ6SYSP* in [Appendix A: Nastel and IBM Documentation](#).

8. Control SMF Accounting Recording using one of the following methods:

- In Nastel Navigator, select Properties from the queue menu, go to the **Monitoring** tab, and use the **Collect Accounting Data** list to set accounting information recording to be **Enabled**, **Disabled**, or configured according to the **Queue Manager**.
- Turn on or off recording of accounting information at the queue or queue manager level using the ACCTQ parameter of the commands DEFINE QLOCAL/QMODEL, ALTER QLOCAL/QMODEL, or DEFINE/ALTER QMGR.
- The AutoPilot M6-WMQ Explorer can also be used to alter the settings. Right-click

on a queue or channel and select **Properties>Monitoring** to control an individual queue or channel. For a queue manager, you can affect accounting for all queues. See the *AutoPilot M6 for WMQ Administrator's Guide* for details.

16.2 Starting NSQZAS

If the configuration steps above were performed, then NSQZAS processing should automatically start each time an SMF data set is full of data collected from WMQ applications.

16.3 Stopping/Restarting NSQZAS

To stop collection of accounting and statistics, use the Stop Trace command.

```
%qmgr stop trace(A) class(3)
```

To restart data collection, use the **Start Trace** command to start WMQ tracing using SMF.

```
%qmgr start trace(A) class(3)
```

A response will be displayed similar to the following:

```
RESPONSE=ADCD CSQW130I %CSQ7 'A' TRACE STARTED, ASSIGNED TRACE NUMBER 02
RESPONSE=ADCD CSQ9022I %CSQ7 CSQWVCM1 ' START TRACE' NORMAL COMPLETION
```

These commands must be repeated for each queue manager.

To trigger ZAS message generation manually before an SMF data set is full, use the **SWITCH SMF** command. (Refer to the SMF Commands *section*.)



To collect channel statistics, you must first stop the channels.

16.4 Mapping Between Accounting and Statistics PCF Messages on Z/OS and Distributed Platforms

The following tables show the mapping from accounting and statistics message parameters on distributed platforms to the equivalent parameter in SMF records on z/OS.

Refer to *Monitoring performance and resource usage* in Appendix A for more information.

In the following table:

- **X** indicates *no equivalent mapping*.
- **Req** indicates *required*
- **WA** indicates *when parameter is available*

Table 16-1. MQI Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
QueueManager	Req	Y	Available in the program loop

Table 16-1. MQI Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
IntervalStartDate Date of the start of the monitoring period.	Req	WTASINTS	Interval start - for post processing. Includes date and time. [5] Might be same as or calculated from IntervalEndDate
IntervalStartTime Time of start of the monitoring period.	Req	WTASINTS	[5]
IntervalEndDate	Req	WTASINTE or SM116DTE	Interval end - for post processing. Includes date and time [5] Date when SMF moved record.
IntervalEndTime	Req	WTASINTE or SM116TME	Time when SMF moved record.
CommandLevel Queue manager command level.	Req	Y	Available from qmgr attributes.
ConnectionId	Req	QWHCCN or WTIDCHLC	Connection name. CONNAME of the Channel. For example: For a sender WINMVS2B(2162) For a receiver, where channel came from: 9.20.101.14 or NYCHOST
SeqNumber Sequence number. This value is incremented for each subsequent record for long running connections.	Req	X	Use constant 1 if it cannot be detected on z/OS.
ApplicationName	Req	??	Transaction name
ApplicationPid	Req	X	Use constant 1
ApplicationTid	Req	QWHCTNO - for CICS only, otherwise X	CICS thread number.
Userld	Req	QWHCAID or QWHCOPID	User ID associated with the OS/390 job; or User ID associated with the transaction.
ConnDate	WA	X	
ConnTime	WA	X	

Table 16-1. MQI Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
ChannelName	WA	WTIDCHL Channel name	
DiscDate	WA	X	
DiscTime	WA	X	
DiscType	WA	X	
OpenCount The number of objects opened. This parameter is an integer list indexed by object type.	WA	X	QMSTOPEN is number of MQOPEN requests, independent of object type.
OpenFail	WA	X	
CloseCount The number of objects closed. This parameter is an integer list indexed by object type.	WA	X	QMSTCLOS is number of MQCLOSE requests, independent of object type.
CloseFailCount	WA	X	
PutCount The number of persistent and nonpersistent messages successfully put to a queue, with the exception of messages put using the MQPUT1 call. This parameter is an integer list indexed by persistence value.	WA	X	QMSTPUT is number of MQPUT requests, independent of persistence value. QMACPUTA + QMACPUTB + QMACPUTC + QMACPUTD Number of MQPUT requests for messages of length 0 through 99 bytes + lengths 100-999 + lengths 1000-9999 + lengths > 10,000, All independent of persistence value. QMAC* data available from message manager accounting record QMAC
PutFailCount	WA	X	
Put1Count The number of persistent and nonpersistent messages	WA	X	QMSTPUT1 is number of MQPUT1 requests, independent of persistence value.

Table 16-1. MQI Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
successfully put to the queue using MQPUT1 calls. This parameter is an integer list indexed by persistence value.			
Put1FailCount	WA	X	
GetCount The number of successful destructive gets for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	WA	X	QMSTGET Number of MQGET requests, independent of persistence value. Also: QMACGETA, QMACGETB, QMACGETC, QMACGETD Number of MQGET requests for messages of length 0 through 99 bytes + Lengths 100-999 + Lengths 1000-9999 + Lengths > 10,000, all independent of persistence value.
GetFailCount The number of unsuccessful destructive gets.	WA	X	
GetBytes Total number of bytes retrieved for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	WA	X	
BrowseCount	WA	X	
BrowseFailCount	WA	X	
BrowseBytes	WA	X	
CommitCount	WA	X	
CommitFailCount	WA	X	
BackCount	WA	X	
InqCount The number of successful objects inquired upon. This	WA	X	QMSTINQ is number of MQINQ requests, independent of object type.

Table 16-1. MQI Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
parameter is an integer list indexed by object type.			
InqFailCount	WA	X	
SetCount The number of successful MQSET calls. This parameter is an integer list indexed by object type.	WA	X	QMSTSET is number of MQSET requests, independent of object type.
SetFailCount	WA	X	

Table 16-2. Queue Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
QueueManager	Req	Y	Available in the program loop
IntervalStartDate The date of the start of the monitoring period.	Req	WTASINTS	Interval start - for post processing. Includes date and time. [5] Might be same as or calculated from IntervalEndDate
IntervalStartTime The time of the start of the monitoring period.	Req	WTASINTS	Might be calculated from IntervalEndTime
IntervalEndDate	Req	WTASINTE or SM116DTE	Interval end - for post processing Date when SMF moved record.
IntervalEndTime	Req	WTASINTE or SM116TME	Time when SMF moved record.
CommandLevel The queue manager command level.	Req	Y	Available from qmgr attributes
ConnectionId	Req	QWHCCN or WTIDCHLC	Connection name. CONNAME of the Channel For example: WINMVS2B(2162)
SeqNumber [3] The sequence number. This value is incremented for each subsequent record for long running connections.	Req	X	Use constant 0
ApplicationName	Req	Y	Transaction name, CCBNAME, Connection name
ApplicationPid	Req	X	Use constant 1
ApplicationTid	Req	QWHCTNO - for CICS only, otherwise X	CICS thread number.
UserId	Req	QWHCAID QWHCOPID	User ID associated with the OS/390 job; or User ID associated with the transaction.
ObjectCount The number of queues accessed in the interval for which accounting data has	Req	Y	WQ structure can have data for more than 1 queue, so count must be available some place. WTASWQCT

Table 16-2. Queue Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
been recorded. This value is set to the number of <i>QAccountingData</i> PCF groups contained in message (1 to 100) [3]			From version 6 – Count of WQ blocks for this thread.
QAccountingData Grouped parameters specifying accounting details for a queue.	Req	Y	
NOTE: One or more of the following data appear in the QAccountingData group			NOTE: The data below is in a structure called the WQ in SMF 116 subtype 1 records.
QName	WA	OBJNAME	
CreateDate	WA	Y	Available from q attributes.
CreateTime	WA	Y	Avail from q attributes
QType	WA	QTYPE	Avail from q attributes
QDefinitionType	WA	Y	Avail from q attributes
OpenCount	WA	OPENN	
OpenDate	WA	OPENTIME	IBM STCK format
OpenTime	WA	OPENTIME	
CloseDate	WA	CLOSTIME	IBM STCK format
CloseTime	WA	CLOSTIME	
PutCount The number of persistent and nonpersistent msgs successfully put to the queue, with the exception of MQPUT1 calls. This parameter is an integer list indexed by persistence value.	WA	Y	PUTN=Number of MQPUT requests PUTPMSG=Number of persistent messages retrieved using MQPUT. PUTN – PUTPMSG = Number of non-persistent messages.
PutFailCount	WA	X	
Put1Count The number of persistent and nonpersistent msgs successfully put to the queue using MQPUT1 calls. This parameter is an integer list	WA	Y	Number of MQPUT1 requests PUT1PMSG= Number of persistent messages created using MQPUT1. PUT1N - PUT1PMSG = Number of non-persistent messages.

Table 16-2. Queue Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
indexed by persistence value.			
Put1FailCount The number of unsuccessful attempts to put a message using MQPUT1 calls.	WA	X	
PutBytes The total number of bytes put for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	WA	PUTBYTES	Total number of bytes put successfully. Assume all persistent; non-persistent = 0.
PutMinBytes Smallest persistent and nonpersistent message size placed on the queue. This parameter is an integer list indexed by persistence value.	WA	PUTMINMS	Put minimum message size Assume all persistent; non-persistent = 0.
PutMaxBytes Largest persistent and nonpersistent message size placed on the queue. This parameter is an integer list indexed by persistence value.	WA	PUTMAXMS	Put maximum message size. Assume all persistent; non-persistent = 0.
GetCount The number of successful destructive MQGET calls for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	WA	Y	GETDVAL (from v6) = Number of successful destructive MQGET Calls. - OR - Number persistent messages retrieved using MQGET = GETPMSG. Non-persistence msgs = VALIDGET-GETPMSG where VALIDGET = Number of MQGET s with data NOTE: GETN = GETA + GETS + GETBRWA + GETBRWS
GetFailCount The number of unsuccessful destructive gets.	WA	X	
GetBytes	WA	GETBYTES	Total number of bytes gotten

Table 16-2. Queue Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
The number of successful destructive MQGET calls for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.			successfully. Assume all persistent; non-persistent=0.
GetMinBytes This parameter is an integer list indexed by persistence value.	WA	GETMINMS	Get minimum message size
GetMaxBytes This parameter is an integer list indexed by persistence value.	WA	GETMAXMS	Get maximum message size
BrowseCount	WA	Y	MQGET Browse Any + Browse Specific = GETBRWA + GETBRWS
BrowseFailCount	WA	X	
BrowseBytes The number of bytes read in non-destructive gets that returned persistent messages.	WA MQCFIL64	X	
BrowseMinBytes This parameter is an integer list indexed by persistence value.	WA	X	
BrowseMaxBytes This parameter is an integer list indexed by persistence value.	WA	X	
TimeOnQMin The shortest time a persistent and nonpersistent message remained on the queue before being retrieved, in microseconds. This parameter is an integer list indexed by persistence value.	WA INTEGER	MINLATNT	Minimum latency of message IBM STCK format, long long. Must convert to INTEGER Microsecs. [5] Assume all persistent; non-persistent=0.
TimeOnQAvg	WA INTEGER	X	

Table 16-2. Queue Accounting Message Parameters

Distributed	REQ or WA	z/OS	z/OS Description
TimeOnQMax	WA INTEGER	MAXLATNT	Maximum latency of message IBM STCK format, long long. Must convert to INTEGER Microsecs. [5] Assume all persistent; non-persistent=0.

[1] Intentionally deleted.

[2] Intentionally deleted.

[3] "Queue accounting message data consists of a number of PCF parameters, and between one and one hundred *QAccountingData* PCF groups. There is one *QAccountingData* PCF group for every queue that had accounting data collected. If an application accesses more than 100 queues, multiple accounting messages are generated. Each message has the *SeqNumber* in the MQCFH (PCF header) updated accordingly, and the last message in the sequence has the *Control* parameter in the MQCFH specified as MQCFC_LAST."

[4] "The SMF statistics broadcast occurs, and it is a long running application, such as a channel. This means it was running prior to the last SMF statistics broadcast. Before Version 5.2 the records were produced only when the job or application ended. So, for a channel that was active for a year, you would get one record. In Version 5.2 you can request that records are produced at the SMF broadcast, typically every 30 minutes, by setting STATIME=0 in the CSQ6SYSP system parameter macro."

[5] "The time in the SMF record (SM116TME) is the time the record was produced (the number of hundredths of a second since midnight). It is usually close (within a second) to the time the transaction or channel ended. If you need more accurate times, you should use the time interval end time (WTASINTE), which is in STCK (Store Clock) format."

"S/390 Store Clock format (STCK) is a double word where bit 53 is a microsecond. To convert a STCK value to microseconds ignore the bottom 12 bits. In C this can be done by treating the values as long long and dividing by 4096 to get to microseconds."

[6] Other z/OS statistics available are Data Manager, Buffer Manager, Lock Manager, DB2 Manager, CF Manager.

Table 16-3. Queue Statistics Message

Distributed	REQ or WA	z/OS	z/OS Description
QueueManager	Req	Y	
IntervalStartDate The date of the start of the monitoring period.	Req	WTASINTS	Interval start - for post processing. Includes date and time. [5]
IntervalStartTime The time of the start of the monitoring period.	Req	WTASINTS	Might be calculated from IntervalEndTime
IntervalEndDate	Req	WTASINTE OR SM116DTE	Interval end - for post processing Date when SMF moved record.
IntervalEndTime	Req	WTASINTE OR SM116TME	Time when SMF moved record.
CommandLevel The queue manager command level.	Req	Y	Available from qmgr attributes
ObjectCount The number of queues accessed in the interval for which accounting data has been recorded. This value is set to the number of <i>QAccountingData</i> PCF groups contained in the message (1 to 100) [3]	Req	Y	WTASWQCT From version 6 – Count of WQ blocks for this thread.
QStatisticsData Grouped parameters specifying statistics for a queue.	Req	Y	
NOTE: One or more of the following data appear in the QStatisticsData group			
QName	WA	OBJNAME	
CreateDate	WA	Y	Avail from q attributes

Table 16-3. Queue Statistics Message			
Distributed	REQ or WA	z/OS	z/OS Description
CreateTime	WA	Y	Avail from q attributes
QType	WA	QTYPE	Avail from q attributes
QDefinitionType	WA	Y	Avail from q attributes
QMinDepth	WA	X	
QMaxDepth	WA	Y	In WQ block
TimeOnQAvg	WA INTEGER	X	
PutCount	WA The number of persistent and nonpersistent msgs successfully put to the queue, with the exception of MQPUT1 calls. This parameter is an integer list indexed by persistence value.	Y	PUTN=Number of MQPUT requests PUTPMSG=Number of persistent msgs retrieved using MQPUT. PUTN – PUTPMSG = nbr of non-persistent msgs.
PutFailCount	WA	X	
Put1Count	WA The number of persistent and nonpersistent msgs successfully put to the queue using MQPUT1 calls. This parameter is an integer list indexed by persistence value.	Y	Number of MQPUT1 requests PUT1PMSG= Number of persistent msgs created using MQPUT1. PUT1N - PUT1PMSG = non-persistent msgs.
Put1FailCount	WA The number of unsuccessful attempts to put a message using MQPUT1 calls.	X	
PutBytes	WA The total number of bytes put for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	PUTBYTES	Total number of bytes put successfully. Assume all persistent; non- persistent = 0.

Table 16-3. Queue Statistics Message

Distributed	REQ or WA	z/OS	z/OS Description
GetCount	WA The number of successful destructive MQGET calls for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	Y	GETDVAL (from v6) = Number of successful destructive MQGET Calls. - OR - Nbr persistent messages retrieved using MQGET = GETPMSG. Non-persistence msgs = VALIDGET-GETPMSG where VALIDGET = Number of MQGET call with data NOTE: GETN = GETA + GETS + GETBRWA + GETBRWS
GetFailCount	WA The number of unsuccessful destructive gets.	X	
GetBytes	WA The number of successful destructive MQGET calls for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	GETBYTES	Total number of bytes got successfully. Assume all persistent; non-persistent=0.
BrowseCount	WA	Y	MQGET Browse Any + Browse Specific = GETBRWA + GETBRWS
BrowseFailCount	WA	X	
BrowseBytes	WA The number of bytes read in non-destructive gets that returned persistent messages.	Y	

Table 16-3. Queue Statistics Message

Distributed	REQ or WA	z/OS	z/OS Description
	MQCFIL64.		
PurgeCount	WA	X	
ExpiredMsgCount	WA	X	
NonQueuedMsgCount	WA	X	

Table 16-4. MQI Statistics Parameters

Distributed	REQ or WA	z/OS	z/OS Description
QueueManager	Req	See previous table	
IntervalStartDate The date of the start of the monitoring period.	Req	See previous table	
IntervalStartTime The time of the start of the monitoring period.	Req	See previous table	
IntervalEndDate	Req	See previous table	
IntervalEndTime	Req	See previous table	
CommandLevel The queue manager command level.	Req	See previous table	
			Following are message manager statistics in QMST record [6]. Message manager processes the MQI verbs.
ConnCount	WA	X	
ConnFailCount	WA	X	
ConnsMax	WA	X	
DiscCount	WA	X	
OpenCount The number of objects successfully opened. This parameter is an integer list indexed by object type.	WA	QMSTOPEN	Number of MQOPEN requests. Not provided by object type. May have to assume all for QLOCAL.
OpenFailCount	WA	X	
CloseCount The number of objects successfully closed. This parameter is an integer list indexed by object type.	WA	QMSTCLOS	Number of MQCLOSE requests. Not provided by object type. May have to assume all for QLOCAL.

Table 16-4. MQI Statistics Parameters			
Distributed	REQ or WA	z/OS	z/OS Description
CloseFailCount	WA	X	
InqCount	WA	X	
InqFailCount	WA	X	
SetCount	WA	QMSTSET	Number of MQSET requests
SetFailCount	WA	X	
PutCount The number of persistent and nonpersistent messages successfully put to a queue using MQPUT requests. This parameter is an integer list indexed by persistence value.	WA	QMSTPUT	Number of MQPUT requests. Assume all persistent; non-persistent=0.
PutFailCount	WA	X	
Put1Count	WA	QMSTPUT1	Number of MQPUT1 requests. Assume all persistent; non-persistent=0.
Put1FailCount	WA	X	
PutBytes	WA	X	
GetCount	WA	QMSTGET	Number of MQGET requests
GetFailCount	WA	X	
GetBytes The number of successful destructive MQGET calls for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value	WA	X	
BrowseCount	WA	X	
BrowseFailCount	WA	X	
BrowseBytes	WA	X	
CommitCount	WA	X	
CommitFailCount	WA	X	
BackCount	WA	X	
ExpiredMsgCount	WA	X	

Table 16-5. Queue Statistics Message			
Distributed	REQ or WA	z/OS	z/OS Description
QueueManager	Req	Y	
IntervalStartDate The date of the start of the monitoring period.	Req	WTASINTS	Interval start - for post processing. Includes date and time. [5]
IntervalStartTime The time of the start of the monitoring period.	Req	WTASINTS	Might be calculated from IntervalEndTime
IntervalEndDate	Req	WTASINTE OR SM116DTE	Interval end - for post processing Date when SMF moved record.
IntervalEndTime	Req	WTASINTE OR SM116TME	Time when SMF moved record.
CommandLevel The queue manager command level.	Req	Y	Available from qmgr attributes
ObjectCount The number of queues accessed in the interval for which accounting data has been recorded. This value is set to the number of <i>QAccountingData</i> PCF groups contained in the message (1 to 100) [3]	Req	Y	WTASWQCT From version 6 – Count of WQ blocks for this thread.
QStatisticsData Grouped parameters specifying statistics for a queue.	Req	Y	
NOTE: One or more of the following data appear in the QStatisticsData group			
QName	WA	OBJNAME	
CreateDate	WA	Y	Avail from q attributes

Table 16-5. Queue Statistics Message

Distributed	REQ or WA	z/OS	z/OS Description
CreateTime	WA	Y	Avail from q attributes
QType	WA	QTYPE	Avail from q attributes
QDefinitionType	WA	Y	Avail from q attributes
QMinDepth	WA	X	
QMaxDepth	WA	Y	In WQ block
TimeOnQAvg	WA INTEGER	X	
PutCount	WA The number of persistent and nonpersistent msgs successfully put to the queue, with the exception of MQPUT1 calls. This parameter is an integer list indexed by persistence value	Y	PUTN=Number of MQPUT requests PUTPMSG=Number of persistent msgs retrieved using MQPUT. PUTN – PUTPMSG = nbr of non-persistent msgs.
PutFailCount	WA	X	
Put1Count	WA The number of persistent and nonpersistent msgs successfully put to the queue using MQPUT1 calls. This parameter is a list indexed by persistence value.	Y	Number of MQPUT1 requests PUT1PMSG= Number of persistent msgs created using MQPUT1. PUT1N - PUT1PMSG = non-persistent msgs
Put1FailCount	WA The number of unsuccessful attempts to put a message using MQPUT1 calls.	X	
PutBytes	WA The total number of bytes put for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	PUTBYTES	Total number of bytes put successfully. Assume all persistent; non- persistent = 0.

Table 16-5. Queue Statistics Message

Distributed	REQ or WA	z/OS	z/OS Description
GetCount	WA The number of successful destructive MQGET calls for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	Y	GETDVAL (from v6) = Number of successful destructive MQGET Calls. - OR - Nbr persistent messages retrieved using MQGET = GETPMSG. Non-persistence msgs = VALIDGET-GETPMSG where VALIDGET = Number of MQGET call with data NOTE: GETN = GETA + GETS + GETBRWA + GETBRWS
GetFailCount	WA The number of unsuccessful destructive gets.	X	
GetBytes	WA The number of successful destructive MQGET calls for persistent and nonpersistent messages. This parameter is an integer list indexed by persistence value.	GETBYTES	Total number of bytes got successfully. Assume all persistent; non-persistent=0.
BrowseCount	WA	Y	MQGET Browse Any + Browse Specific = GETBRWA + GETBRWS
BrowseFailCount	WA	X	
BrowseBytes	WA The number of bytes read in non-destructive gets that returned persistent messages.	Y	

Table 16-5. Queue Statistics Message

Distributed	REQ or WA	z/OS	z/OS Description
	MQCFIL64.		
PurgeCount	WA	X	
ExpiredMsgCount	WA	X	
NonQueuedMsgCount	WA	X	

Table 16-6. Channel Statistics Message			
Distributed	REQ or WA	z/OS	z/OS Description
QueueManager	Req	See previous table	
IntervalStartDate The date of the start of the monitoring period.	Req	See previous table	
IntervalStartTime The time of the start of the monitoring period.	Req	See previous table	
IntervalEndDate	Req	See previous table	
IntervalEndTime	Req	See previous table	
CommandLevel The queue manager command level.	Req	See previous table	
			Following data is in Task Id structure WTID
ChannelName	Req	WTIDCHL	Channel name Sender: The channel name For example: V521.V52A Rcvr: The job name For example: V52ACHIN
ChannelType	Req	Y	Avail from channel attributes
ChlStatisticsData Grouped parameters specifying statistics details for a channel.	Req		
RemoteQmgr	WA	X	
ConnectionName	WA	WTIDCHLC	Channel connection name Sender: This is defined in the CONNAME attribute of the Channel For example: WINMVS2B(2162) Rcvr: Where the channel

Table 16-6. Channel Statistics Message

Distributed	REQ or WA	z/OS	z/OS Description
			came from For example: 9.20.101.14
MsgCount The number of [persistent + nonpersistent] messages sent or received.	WA	Y	Derived from WQ data
TotalBytes The number of [persistent + nonpersistent] bytes sent or received.	WA	Y	Derived from WQ data
NetTimeMin	WA	X	
NetTimeAvg	WA	X	
NetTimeMax	WA	X	
ExitTimeMin	WA	X	
ExitTimeAvg	WA	X	
ExitTimeMax	WA	X	
FullBatchCount	WA	X	
IcplBatchCount	WA	X	
AverageBatchSize	WA	X	
PutRetryCount	WA	X	

16.5 Best Practices for ZAS Collection

For efficient collection of data, note the following configuration points:

1. Minimize the number of types of SMF records collected.

The only SMF records needed are accounting, record type 116. In SMFPRMxx member of SYS1.PARMLIB, specify:

```
SYS(TYPE(116, nnn,...))
```

where nnn are other record types that you want collected for other vendor products. The more record types that are written by SMF, the greater the system overhead.

2. Provide minimally 3 SMF datasets for collection.

In SMFPRxx, define the names of the SMF data sets to be used, for example:

```
DSNAME(SYS1.MAN1,
        SYS1.MAN2,
        SYS1.MAN3)
```

Three or more data sets allow time for the SMF writer to cycle through the data sets without overwriting any one data set too soon, in case you want to view or re-process it.

3. Size the SMF datasets to minimize frequent SMF exits.

The SMF IEFU29 exit is triggered and invokes NSQZAS each time an SMF data set fills. Making the data set sizes too small will cause them to fill more frequently, especially if there are multiple vendor products that do SMF data collection. This will cause more frequent exits and processing program startups. Making the data sets too large will reduce the fill rate and the frequency at which the SMF data is processed and converted to accounting and statistics data. In the latter case, if the fill rate is taking too long, a JCL to *SWITCH SMF* can be scheduled to execute more frequent SMF dumps and NSQZAS processing.

An SMF data set sizing estimation formula follows.

Standard SMF 116 header record:	SMFHdr_size	28 bytes
WMQ SMF header:	QWHS_size	128
TaskID record:	WTID_size	208
Thread/task level accounting record:	WTAS_size	700
Queue accounting record:	WQ_size	592

nq = avg. number of queues accessed during an application connection

nc = avg. number of application connections to be handled in an SMF data set

For each application connection to a queue manager, the SMF data size is:

$$\text{SMFHdr_size} + \text{QWHS_size} + \text{WTID_size} + \text{WTAS_size} + \text{nq} * \text{WQ_size} = 1064 + \text{nq} * 592$$

Total estimated SMF data set storage required for WMQ =

$$nc * (1064 + nq * 592) =$$

$$nc * 1064 + nc * nq * 592 \text{ bytes}$$

4. Maximize NSQZAS statistics aggregation interval to reduce network traffic.

Set the NSQZAS startup option `-t` value (the aggregation interval for collecting SMF WMQ message data into PCF statistics messages) to the maximum possible for your data analysis purposes. This will minimize the number of PCF statistics generated and reduce network traffic overhead from the z/OS machine to the AutoPilot workgroup server.

For example:

If an SMF data set, after filling, contains 30 minutes of uniformly distributed WMQ messages, a `-t` value of 1800 secs (30 mins) would cause the generation of 1 MQI statistics message covering that entire 30-minute interval; `m` queue statistics messages (`m` = number of queues accessed by all WMQ applications in the 30-minute interval) and `n` channel statistics message (`n` = number of active channels in the interval), or a total of $(1+m+n)$ messages. A `-t` value of 300 (5 mins) would cause approximately 6 times as many messages.

16.6 Initializing SMF Data Sets

To start off a test from a known state, the existing SMF data sets should be cleared as follows:

1. Use SDSF command `/D SMF` to display the SMF data sets and their state.
2. Open `++hlq++JCL(SMFCLEAN)`.
3. Edit the DUMPIN line and change the MANnn number to that of a data set in the DUMP REQUIRED state.
4. Repeat for each such data set.
5. If a data set is in the active state, use the **switch smf** command to switch the active SMF data set to the next available member.

switch SMF

```
RESPONSE=ADCD      IEE360I SMF NOW RECORDING ON SYS1.MAN2 ON
Z7SYS1 RESPONSE=TIME=15:59:28
```

There must be at least one data set in the ALTERNATE state. For example, the D SMF command might show this before the switch command:

```
RESPONSE=ADCD
IEE949I 15.53.15 SMF DATA SETS 209
      NAME                VOLSER    SIZE (BLKS)  %FULL
      STATUS P-SYS1.MAN1  Z7SYS1
      7200             5             ACTIVE
      S-SYS1.MAN2        Z7SYS1    1800         0         ALTERNATE
      S-SYS1.MAN3        Z7SYS1    1800         0         ALTERNATE
```

After the switch command, the display SMF response shows the active data set is now the first alternate, SYS1.MAN2:

```
RESPONSE=ADCD
IEE949I 16.02.08 SMF DATA SETS 252
      NAME                VOLSER    SIZE(BLKS)  %FULL  STATUS
      P-SYS1.MAN1        Z7SYS1    7200         7      DUMP REQUIRED
      S-SYS1.MAN2        Z7SYS1    1800         2      ACTIVE
      S-SYS1.MAN3        Z7SYS1    1800         0      ALTERNATE
```

The data set will now be in DUMP REQUIRED state and SMFCLEAN can be run for that data set.

16.7 SMF Commands

16.7.1 Running SMF Command from SDSF

1. Go to the SDSF panel by logging in and entering command "SDSF" or "=m.5" on an Option or Command Input line.
2. On the command line, enter `/command` using one of the commands below. The command may be entered in upper or lower case.
3. On the SDSF panel you may:
 - a. Enter a slash alone (/) on the COMMAND INPUT line to see a history of previous commands.
 - b. Page up and down through the command history.
 - c. Move the cursor to one of the commands and press **Enter** to recall that command to the command line. Then press **Enter** again to execute it.

16.7.2 Displaying SMF Status

To display SMF status, type

```
D SMF
```

Possible responses are:

- a. RESPONSE=ADCD IEE351I SMF SYS1.MAN RECORDING NOT BEING USED

This response indicates SMF has not been activated. Use the Start SMF command.

- b. RESPONSE=ADCD
IEE949I 14.46.02 SMF DATA SETS 029

NAME	VOLSER	SIZE(BLKS)	%FULL	STATUS
P-SYS1.MAN1	Z7SYS1	7200	28	ACTIVE
S-SYS1.MAN2	Z7SYS1	1800	49	DUMP REQUIRED
S-SYS1.MAN3	Z7SYS1	1800	12	DUMP REQUIRED

This response shows which data set is active and which require dumping.

16.7.3 Activating SMF for Recording Record Types 116

```
SETSMF SYS(TYPE 116)
```

```
RESPONSE=ADCD IEE712I SETSMF PROCESSING COMPLETE
```

16.7.4 Starting SMF

Use this command to start WMQ tracing using SMF.

```
set SMF=PQ
```

PQ in this example is the suffix of the PARMLIB(PROC.xx) file. (See Configuring NSQZAS.)

You will see a message that requires you to respond as follows:

1. On the command line, enter LOG. At the bottom of the LOG screen, you will see a message similar to

```
0000000 ADCD 14.41.53 *03 IEE357A REPLY WITH SMF VALUES OR U
```

2. At the top of the screen, on the COMMAND INPUT line, respond with /03U

Use the exact sequence number appearing at the beginning of the REPLY prompt. In this example, it is 03.

3. Repeat the display SMF status command 'D SMF'. You should now see a reply similar to this:

```
RESPONSE=ADCD
```

```
IEE949I 14.46.02 SMF DATA SETS 029
```

NAME	VOLSER	SIZE(BLKS)	%FULL	STATUS
P-SYS1.MAN1	7SYS1	7200	28	ACTIVE
S-SYS1.MAN2	Z7SYS1	1800	49	DUMP REQUIRED
S-SYS1.MAN3	Z7SYS1	1800	12	DUMP REQUIRED

16.7.5 Displaying Trace Status

Use this command to see the status of WMQ tracing.

```
%qmgr display trace
```

```
RESPONSE=ADCD
```

```
CSQW127I %CSQ7 CURRENT TRACE ACTIVITY
```

```
IS - TNO TYPE CLASS DEST USERID
```

```
RMID
```

```
1 GLOBAL 01 RES * *
2 ACCTG 03 SMF * *
00 CHINIT * RES * * END OF TRACE REPORT
```

```
CSQ9022I %CSQ7 CSQWVCM1 ' DISPLAY TRACE ' NORMAL COMPLETION
```

The second line in the trace shows that SMF trace is active for accounting, class 3. This is what is needed for NSQZAS to be able to collect WMQ accounting and statistics. If this line is not seen, use the `Start Trace` command.

16.7.6 Starting Trace

Use this command to start WMQ tracing using SMF.

```
%qmgr start trace(A) class(3)
```

```
RESPONSE=ADCD CSQW130I %CSQ7 'A' TRACE STARTED, ASSIGNED TRACE NUMBER 02
```

```
RESPONSE=ADCD CSQ9022I %CSQ7 CSQWVCM1 ' START TRACE ' NORMAL COMPLETION
```

16.7.7 Stopping Trace

Use this command to stop WMQ tracing using SMF.

```
%qmgr stop trace(A) class(3)
RESPONSE=ADCD CSQW131I %CSQ7 STOP TRACE SUCCESSFUL FOR TRACE NUMBER(S)
RESPONSE=02
RESPONSE=ADCD CSQ9022I %CSQ7 CSQWVCM1 ' STOP TRACE' NORMAL COMPLETION
```

16.7.8 Switching the Active SMF Data Set

Use this command to switch the active SMF data set to the next available member.

```
switch SMF
RESPONSE=ADCD IEE360I SMF NOW RECORDING ON SYS1.MAN2 ON Z7SYS1
RESPONSE=TIME=15:59:28
```

There must be at least one data set in the ALTERNATE state. For example, the D SMF command might show this before the switch command:

```
RESPONSE=ADCD
IEE949I 15.53.15 SMF DATA SETS 209

      NAME          VOLSER  SIZE(BLKS)  %FULL  STATUS
      P-SYS1.MAN1  Z7SYS1   7200        5      ACTIVE
      S-SYS1.MAN2  Z7SYS1   1800        0      ALTERNATE
      S-SYS1.MAN3  Z7SYS1   1800        0      ALTERNATE
```

After the switch command, the display SMF response shows the active data set is now the first alternate, SYS1.MAN2:

```
RESPONSE=ADCD
IEE949I 16.02.08 SMF DATA SETS 252

      NAME          VOLSER  SIZE(BLKS)  %FULL  STATUS
      P-SYS1.MAN1  Z7SYS1   7200        7      DUMP REQUIRED
      S-SYS1.MAN2  Z7SYS1   1800        2      ACTIVE
      S-SYS1.MAN3  Z7SYS1   1800        0      ALTERNATE
```

16.7.9 Running the SMF Exit Manually

Use this command to run the same JCL that is executed when the SMF exit occurs, when an SMF data set is full.

```
START NSQSMF,MAN=smf_dataset
```

where `smf_dataset` is the name of a full data set, such as `SYS1.MAN1`

The NSQSMF JCL will automatically be found in data set USER.PROCLIB. The screen will show this top line, which can be ignored:

```
HQX7720 ----- SDSF PRIMARY OPTION MENU -- NO RESPONSE RECEIVED
```

Enter LOG, find the last NSQSMF job and display it to check that the job ran properly.

Alternately, use the **SWITCH SMF** command.

16.8 Useful WMQ Commands

These commands are run from the SDSF panel

To start channel initiator:

1. Go to SDSF panel: =m.5
2. Enter `/%qmgr start chinit`

To display channel initiator:

1. Go to SDSF panel: =m.5
2. Enter `/%qmgr display chinit`

To start queue manager:

1. Go to SDSF panel: =m.5
2. Enter `/%qmgr start qmgr`

To display queue manager:

1. Go to SDSF panel: =m.5
2. Enter `/%qmgr display qmgr`

To start listener:

1. Go to SDSF panel: =m.5
2. Enter `/%qmgr start listener port_nbr` (for example: 1418)

16.9 Utilities

16.9.1 How to Process an SMF Dump File

This procedure takes an existing SMF dump file and processes it to form PCF accounting and statistics messages.

1. Go to *naslib.JCL*
2. Find RUNZAS and enter option **SUB** to submit the job. Leave the `-t` trace option enabled.
3. Go to SDSF panel, enter ST (status of jobs), and find and select job NSQZAS.
4. Examine SYSOUT file for error messages
5. Examine SYSPRINT trace file. First few lines should look like this:

```
NSQZAS, z/OS Accounting and Statistics Collector, version 1.1.1 Nov
8 2007 starting...
```

```
z/OS Accounting and Statistics Collector start-up parameters:
STARTUP OPTIONS:
```

```
  QMgr List                V22F CSQ6 CSQ7
  Collector QMgr           CSQ7
  MQI Accounting           on
  Queue Accounting         on
  MQI Statistics           on
  Queue Statistics         on
  Channel Statistics       on
  Statistics Interval      300
  secs Trace functions     on

  Dump SMF records         on
  Dump output to:         stdout

Tue Dec 4 17:01:28 2007    main: after initialization
Tue Dec 4 17:01:28 2007    zasConnQMgr: entry: CSQ7
Tue Dec 4 17:01:28 2007    zasConnQMgr: exit, hConn 1dc97ce0
```

Last 3 lines should like this:

```
Tue Dec 4 17:01:29 2007    zasFreeStatsBlocks: Free QStats
                             block malloc'd memory

Tue Dec 4 17:01:29 2007    main: close SMF input file and dump
                             file

Tue Dec 4 17:01:29 2007    main: exit
```

16.9.2 How to Use AMQSMON to Print Formatted Accounting and Statistics PCF Messages

When an SMF data set is dumped and processed during production, generated PCF messages are enqueued to alias queues NASTEL.ADMIN.STATISTICS.QUEUE and NASTEL.ADMIN.ACCOUNTING.QUEUE and sent to a workgroup server by the publisher or agent. In order to examine PCF messages before they leave the z/OS system, the queues have to be changed to local queues. Then utility `amqsmon` (the customized version that reads from those queues) can be run to analyze and print the formatted PCF messages.

1. Go to ISPF Primary Option Menu.
2. Go to IBM WebSphere MQ for z/OS – Main Menu. Enter on command either: M followed by 12 on the IBM Products Panel; or =m.12
3. Next to queue names NASTEL.ADMIN.STATISTICS.QUEUE and NASTEL.ADMIN.ACCOUNTING.QUEUE, enter Action 1, Object QALIAS, Name NAS*.
4. On the next screen, next to the queue names, enter option 4 followed by 1 in order to delete the definitions.
5. On the Main Menu, use option 2, Define Like, with Object QLOCAL to create the two local queues with the names defined in step 3 above.
6. Run the JCLs to dump and process an SMF data set. To run `amqsmon`:
 - a. Open for edit data set `naslib.JCL.AMQSMON`. Edit the queue manager name `-m CSQ7` to suit the test. Optionally append `'-b'` to the PARAM line to browse the queue without deleting the PCF message.
 - b. Edit the PARAM line to format the MQI Accounting messages and submit:
`PARM='-m CSQ7 -t accounting -a'`
 - c. Edit the PARAM line to format the Queue Accounting messages and submit:
`PARM='-m CSQ7 -t accounting -q'`
 - d. Edit the PARAM line to format the MQI Statistics messages and submit:
`PARM='-m CSQ7 -t statistics -a'`
 - e. Edit the PARAM line to format the Queue Statistics messages and submit:
`PARM='-m CSQ7 -t statistics -q'`
 - f. Edit the PARAM line to format the Channel Statistics messages and submit:
`PARM='-m CSQ7 -t statistics -c'`
7. To view the output, go to the SDSF panel: =m.5
8. Enter ST (status of jobs) and page down to the AMQSMON jobs; there should be 5 of them. Select each job in turn with the '?' option and then select the SYSPRINT log with the s' option. They should look like the following samples:

```
MonitoringType: MQIAccounting QueueManager: 'V22F' IntervalStartDate:
'2000-11-24'
IntervalStartTime: '10.03.35'
IntervalEndDate: '2000-11-24'
IntervalEndTime: '10.03.50'
CommandLevel: 0
ConnectionId:    x'c4e4d4d4e840404040404083f285f98484f4f6f2f0f0f7f5f5f083'
SeqNumber: 0
ApplicationName: 'PAICEB' ApplicationPid: 0
ApplicationTid: 0 UserId: 'PAICE'
OpenCount: [0, 2000, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
CloseCount: [0, 2000, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
InqCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
PutCount: [0, 1000]
Put1Count: [0, 0]
PutBytes: [0, 1000000]
GetCount: [0, 2000]
GetBytes: [0, 1000000]
CommitCount: 2000
BackCount: 0
```

```
MonitoringType: QueueAccounting QueueManager: 'V22F' IntervalStartDate:
'2000-11-24'
IntervalStartTime: '10.03.35'
IntervalEndDate: '2000-11-24'
IntervalEndTime: '10.03.50'
CommandLevel: 0
ConnectionId:    x'c4e4d4d4e840404040404083f285f98484f4f6f2f0f0f7f5f5f083'
SeqNumber: 0
ApplicationName: 'PAICEB' ApplicationPid: 0
ApplicationTid: 0 UserId: 'PAICE'
ObjectCount: 2
QueueAccounting: 0 QueueName: 'CP0000'
. . .
```

MonitoringType: **MQIStatistics** QueueManager: 'V22F' IntervalStartDate: '2000-11-24'

IntervalStartTime: '14.54.13'

IntervalEndDate: '2000-11-24'

IntervalEndTime: '14.54.31'

CommandLevel: 0

OpenCount: [0, 2001, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

CloseCount: [0, 2000, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

InqCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

SetCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

PutCount: [0, 1000]

Put1Count: [0, 1000]

PutBytes: [0, 2000000]

GetCount: [0, 4001]

GetBytes: [0, 2000000]

BrowseCount: [0, 0]

CommitCount: 3000

BackCount: 0

MonitoringType: **QueueStatistics** QueueManager: 'V22F' IntervalStartDate: '2000-11-24'

IntervalStartTime: '14.54.13'

IntervalEndDate: '2000-11-24'

IntervalEndTime: '14.54.31'

CommandLevel: 0

ObjectCount: 2

QueueStatistics: 0 QueueName: 'CP0000' QueueType: Local QMaxDepth: 0

PutCount: [0, 0]

Put1Count: [0, 1000]

PutBytes: [0, 1000000]

GetCount: [0, 2000]

GetBytes: [0, 1000000]

BrowseCount: [0, 0]

QueueStatistics: 1 QueueName: 'SERVER' QueueType: Local QMaxDepth: 0

PutCount: [0, 1000]

Put1Count: [0, 0]

PutBytes: [0, 1000000]

GetCount: [0, 2001]

GetBytes: [0, 1000000]

BrowseCount: [0, 0]

```
MonitoringType: ChannelStatistics
  DATA A:
  MonitoringType: ChannelStatistics
  QueueManager: 'CSQ7'
  IntervalStartDate: '2008-03-11'
  IntervalStartTime: '17.18.29'
  IntervalEndDate: '2008-03-11'
  IntervalEndTime: '17.21.38'
  CommandLevel: 600
  ObjectCount: 2
ChannelStatistics: 0
  ChannelName: 'CSQ7_TO_CSQ6'
  ChannelType: Sender
  ConnName: '11.0.0.55(1416)'
  MsgCount: 71
  TotalBytes: 30459
ChannelStatistics: 1
  ChannelName: 'CSQ6_TO_CSQ7'
  ChannelType: Receiver
  ConnName: '11.0.0.55'
  MsgCount: 2
TotalBytes: 404
```

16.9.3 How to Manually Dump an SMF Data Set

1. Use SDSF command `/D SMF` to display the SMF data sets.
2. Pick one that is in state READY TO BE DUMPED, for example: MAN1
3. Open JCL `naslib.JCL(NSQSMFDP)` for edit.
4. Edit the 3 lines where MANnn and MQRECSnn are located, changing nn to the chosen SMF data set number 1, 2, etc.
5. Submit the job.

The output in the log should

look like this: FA010I SMF DUMP

PARAMETERS FA010I END(2400)

-- DEFAULT FA010I

START(0000) -- DEFAULT

FA010I DATE(1900000,2099366) --

DEFAULT FA010I OUTDD(DUMPOUT,TYPE

116) -- SYSIN FA010I

INDD(DUMPIN,OPTIONS(ALL)) -- SYSIN

FA020I DUMPOUT -- HBC.MP1B.MQRECS3

FA020I DUMPIN -- SYS1.MAN3

FA018I SMF DATA SET DUMPIN HAS BEEN CLEARED.

SUMMARY ACTIVITY REPORT

RECORD TYPE	RECORDS READ	PERCENT OF TOTAL	AVG. RECORD LENGTH	MIN. RECORD LENGTH	END DATE-TIME
2	0				
3	0				
4	3	.03%	255.00	247	
5	1	.01%	145.00	145	
99	10,133	96.62%	633.63	182	
116	1	.01%	6,500.00	6,500	
TOTAL	10,488	100%	621.42	91	
NUMBER OF RECORDS IN ERROR		0			

16.9.4 How to Manually Process an Existing SMF Dump File

1. On SDSF panel, use command `/D SMF` to display the SMF data sets.
2. Select `naslib.JCL.RUNZAS`, used to manually start program NSQZAS processing of an SMF dump. (*naslib* is a system high level qualifier, such as M6WMQ)
3. Edit the *nn* value on the SMFIN line then SUBMIT the job.
4. Check the output files `naslib.SYSPRINT` and `naslib.SMFDUMP`.

The former is a trace of the NSQZAS program; the latter contains a hex-ascii dump of the SMF records, assuming that the option `-ds` (dumpfile to stdout) was not used when starting the job in step 2. With `-ds`, the hex-ascii dump appears together with the trace in the SYSPRINT file.

In the trace file, all PCF message traces are preceded by a line containing the words **"dump and print"**.

Other key words to look for when searching for a trace or dump of a PCF message:

- MQCMD_ACCOUNTING_MQI
- MQCMD_ACCOUNTING_Q
- MQCMD_STATISTICS_MQI
- MQCMD_STATISTICS_Q
- MQCMD_STATISTICS_CHANNEL

To check the SMFDUMP file:

1. Enter the `s` option next to the SMFDUMP data set name.
2. If the display shows "Used cylinders: 0", then an empty file was created because the `-ds` option was selected in the RUNZAS job.
3. If the Used Cylinder count is non-0, then use the `'e'` option to open the file.
4. The SMF records are printed in pure hex-ascii and then followed by a formatted print, displaying the field names, description (sometimes), memory address, hex value, and as a string if applicable. The dump will have lines similar to these:

```
000033 --SMFHEADER---H-E-X---P-R-I-N-T-----
000034 Address      = 1DC63868
000035 00000000 : 5E740061 725E0107 317FE2E8 E2F1C3E2 ;../.;..."SYS1CS>
000036 00000010 : D8F70001 F6F0F000 000018E4 00800001 Q7..600....U....>
000037 --SMFHEADER---F-O-R-M-A-T-T-E-D-----
000038 SMFRECF LG : (addr) '1DC63868' (hex) '5E' : (disp) ';'
000039 SMFRECR TY : (addr) '1DC63869' (hex) '74' : (disp) '.'
..
000081 --QWHS-HEADER---H-E-X---P-R-I-N-T-----
```

```

000082 Address = 1DC65148
000083 00000000 : 002401F7 00190410 1BFF8C98 C3E2D8F7 <...7. .... qCSQ7>
000084 00000010 : C17E7796 E9F7E200 0001111B 0001111B <A=.oz7S. ....>
000085 00000020 : 00000002 <....>
000086 --QWHS-HEADER---F-O-R-M-A-T-T-E-D----
000087 qwhsnsda (nbr self-defining sections): 4 (hex), 4 (dec)
000088 qwhsssid (subsystem name = QMgr) : (addr) '1DC65154' (hex) 'C3E2 000089
qwhcaid (user ID for the OS/390 job) : (addr) '1DC65170' (hex) 'E2E8 000090 qwhccv
(thread cross ref) : (addr) '1DC65178' (hex) 'F2F4 000091 qwhccn (connection name):
(addr) '1DC65184' (hex) 'C3E2

000107 ---W-T-I-D----H-E-X----P-R-I-N-T-----
000108 Address = 1DC638A0
000109 00000000 : F70000D0 E6E3C9C4 00000002 D5E2D8E2 <7..}WTID. .. NSQS>
000110 00000010 : D4C64040 E2E3C1D9 E3F24040 40404040 <MF START2 >
000111 00000020 : 40404040 00000000 00000000 40404040 < ..... >
000112 00000030 : 40404040 40404040 40404040 40404040 < >
000113 00000040 : 40404040 40404040 C17E7795 478D0001 < A=.n. ... >
...
000122 ---W-T-I-D----F-O-R-M-A-T-T-E-D-----
000123 wtidshex = fffff700
000124 wtidlen = 208
000125 wtideyec = WTID
000126 wtidatyp = 0002
000127 wtidccn: (addr) '1DC638AC' (hex) 'D5E2D8E2D4C64040' : (disp) 'NSQSMF
000128 wtidopid : (addr) '1DC638B4' (hex) 'E2E3C1D9E3F24040' : (disp) 'START2
000129 wtidnid: (addr) '1DC638BC' (hex) '40404040404040000000000000000000' :
000130 wtidcori : (addr) '1DC638CC' (hex) '40404040404040404040404040404040' : (disp) '
...
000148 ---W-T-A-S----H-E-X----P-R-I-N-T-----
000149 Address = 1DC63970
000150 00000000 : F70102D8 E6E3C1E2 C17E7795 47D53700 <7..QWTASA=.n.N...>
000151 00000010 : 54DCAB28 54DB5038 1BFF8CF8 00000000 <.....&.....8....>
000152 00000020 : 000003F4 00000000 30800000 00000008 <...4. ....>
000153 00000030 : 00000000 00000000 00000000 00000000 <. ....>
...
000196 ---W-T-A-S----F-O-R-M-A-T-T-E-D-----
000197 wtasshex = fffff701
000198 wtaslen= 0728
000199 wtaseyec = WTAS
000200 wtasstrt : (addr) '1DC63978' (hex) 'C17E779547D53700' : (disp) 'A=.n.N..
000201 zasSTCKToStr: IN: llStck <-4504031094284929280>, OUT: Date <2007-11-13>,
...

```

```
000289 ---W-Q-----H-E-X----P-R-I-N-T-----
000290 Address = 1DC63C48
000291 00000000 : F70202A0 E6D8E2E3 00000004 54D4F578 <7...WQST. ... M5.>
000292 00000010 : C17E7795 47D53700 54DCAB28 54DB5038 <A=.n.N. .... &.>
000293 00000020 : E2E8E2E3 C5D44BC3 D6D4D4C1 D5C44BD9 <SYSTEM.COMMAND.R>
000294 00000030 : C5D7D3E8 4BD4D6C4 C5D34040 40404040 <REPLY.MODEL      >
000295 00000040 : 40404040 40404040 40404040 40404040 <                          >
...
000333 ---W-Q-----F-O-R-M-A-T-T-E-D-----
000334 wqid = fffff702
000335 wql1 = 0672
000336 wqeye = WQST
000337 wqver = 00000004
000338 correl : (addr) '1DC63C58' (hex) 'C17E779547D5370054DCAB2854DB5038' :
000339 objname = SYSTEM.COMMAND.REPLY.MODEL
000340 basename = CSQ.C17E77964D480100
000341 opentime : (addr) '1DC63CC8' (hex) 'C17E77964E878500' : (disp) 'A=.o+ge.
```

16.9.5 How to Print Formatted SMF Records

1. On SDSF panel, use command **/D SMF** to display the SMF data sets.
2. Select *naslib.JCL(NSQSMFDP)* to dump a specific SMF data set. Edit the data set related lines and submit the job.
3. Select *naslib.JCL(RUNCDUMP)* to print the contents of type 116 records in a dump-like format, in both EBCDIC and hex. (See format in previous section). Individual files are created for each of the following record subtypes:
 - **SMFHDR** – SMF record headers
 - **QWHS** – Common WebSphere MQ header
 - **WTID** – Class 3 accounting – Thread identification data
 - **WTAS** – Class 3 accounting – General thread accounting data
 - **WQ** – Class 3 accounting – Queue specific accounting data
4. Edit the nn value and/or data set name on the SMFIN line and SUBMIT the job.
5. Check the SYSPRINT log files for job CDUMP.

16.10 SMF Exit JCL Listing

This JCL is in USER.PROCLIB(NSQSMF).

```

//NSQSMF      PROC MAN='SYS1.MAN1',CNTL=MQSMF
//*-----
//* CALLED BY: NSQXU29 assembler exit program
//* LOCATION:  member of a procedure library, e.g., USER.PROCLIB
//* INPUT:     MAN = Name of data set that just filled
//*           Default: SYS1.MAN1
//*           CNTL = Name of file containing SMF dump options
//*           Default: MQSMF
//* STEPS:
//*
//* 1. NSQSMFDP - Dump all SMF records to temporary file
//* 2. 1-2.n NSQZAS1,NSQZAS2,...
//*    - Analyze and report on SMF 116 from temporary file
//*    Need one NSQZAS job step per qmgr. Copy NSQZAS job
//*    lines, rename jobs NSQZAS1, NSQZAS2,....
//*    Change the -m and -mc values to a qmgr.
//* 3. NSQGDG - Copy temporary file to GDG for later reuse
//*-----
//* CUSTOMIZATION:
//*   Change ++hlq++ to M6WMQ install data set
//*   Change ++mqser++ to IBM WMQ data set
//*   Add NSQZAS job sections, one per qmgr being monitored
//*   Set PARM options for each NSQZAS job
//*-----
//* Dump a filled SMF data set:
//*
//* The NASTEL.MQSMF.CNTL(MQSMF) data set contains:
//*   INDD(DUMPIN,OPTIONS(DUMP|ALL))
//*   OUTDD(DUMPOUT,TYPE(116))
//* You can change the OPTIONS to ALL, to clear all SMF records.
//* Only the last SMF exit in the chain should do this.
//*
//* &&NSQSMF = temp file created automatically for the job duration
//*-----
//NSQSMFDP EXEC PGM=IFASMFDP
//SYSPRINT DD SYSOUT=*
//DUMPIN   DD DSN=&MAN,DISP=SHR
//DUMPOUT  DD DSN=&&NSQSMF,
//          DISP=(,PASS),
//          SPACE=(TRK,(15,15),RLSE),UNIT=SYSDA,
//          DCB=(RECFM=VBS,BLKSIZE=32760)
//SYSIN    DD DSN=NASTEL.MQSMF.CNTL(&CNTL),DISP=SHR
//*-----
//* Execute if prev step Return Code <= 4
//* OM = give all resources (memory) needed
//*-----
//* -m List of qmgrs for which to collect acct/stats msgs
//* -mc Qmgr to use by the collector for writing the acct/
//*     stats messages to forward to M6WMQ workgroup server
//*
//* Note: -m and -mc values should be the same. To process multiple
//*       qmgrs, one NSQZAS job per qmgr is required. That is,
//*       replicate lines NSQZAS through SMFIN and edit the copied
//*       lines as required. All the NSQZAS jobs will process
//*       from the same temp SMF dump file.

```

```

/**
/** -qa Collect Queue Accounting on|off
/** -ma Collect MQI Accounting on|off
/** -ms Collect MQI Statistics on|off
/** -qs Collect Queue Statistics on|off
/** -cs Collect Channel Statistics on|off
/** -st n_secs Statistics time: aggregate over n_secs. Default: 1800.
/** -t Trace functions (stdout is file SYSPRINT)
/** -d Dump SMF records in ascii/hex
/** -ds Dump output to stdout. Default: file SMFDUMP.
/**
/**-----
//NSQZAS EXEC PGM=NSQZAS,COND=(4,LT),REGION=0M,
// PARM='-m CSQ6 -mc CSQ6 -qa on -ma on
// -ms on -qs on -cs on -st 600 -t -d -ds'
//STEPLIB DD DISP=SHR,DSN=++hlq++.LOAD
// DD DISP=SHR,DSN=++mqser++.SCSQAUTH
// DD DISP=SHR,DSN=++mqser++.SCSQANLE
// DD DISP=SHR,DSN=++mqser++.SCSQLOAD
//SYSPRINT DD SYSOUT=*,DCB=(LRECL=132,RECFM=F)
//SMFDUMP DD SYSOUT=*,DCB=(LRECL=233,RECFM=F,BLKSIZE=233)
//SMFIN DD DSN=&&NSQSMF,DISP=(OLD,PASS)
/**
/**-----
/** Copy SMF dump to file within GDG to preserve data for later reuse
/** GDG = Generation Data Group
/** DATA(0) = is replaced with latest generation version, e.g., G0002.V00
/**-----
//NSQGDG EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=&&NSQSMF,DISP=(OLD,DELETE)
//SYSUT2 DD DSN=NASTEL.MQSMF.DATA(+1),
// DISP=(NEW,CATLG),
// SPACE=(TRK,(15,15),RLSE),UNIT=SYSDA,
// DCB=(RECFM=VBS,BLKSIZE=32760)
//SYSIN DD DUMMY

```

Chapter 17: nsqprid Utility

17.1 Description

Utility program nsqprid creates a specific user/password for a workgroup server and writes the credentials to the **mqgroup.ini** file.

17.2 Command

Use the following command to run the utility:

```
C:\nastel\apwmq\bin>nsqprid
```

17.3 Sample Output

```
C:\nastel\apwmq\bin>nsqprid
```

```
(C)Copyright 1995 - 2015, Nastel Technologies Inc.
```

```
Reproduction, adaptation, or distribution without prior  
written permission is prohibited, except as allowed under  
the copyright laws.
```

```
Date: Thu Oct 08 14:40:48 2015
```

```
NSQNSQ Processing ID Editor V6.5.0005, Build Jul 3 2015
```

```
Enter Workgroup Name [MQM]: MQMPRD
```

```
Enter Processing ID User Name: JohnM
```

```
Enter Processing ID Password: *****
```

```
Reenter Processing ID Password: *****
```

```
Wrote Processing ID credentials to: C:\nastel\apwmq\config\groups\mqgroup.ini  
(0028140)
```

```
Jane Doe (manager)
```

```
2015-10-08 14:48
```

```
The credentials lines written to the end of file mqgroup.ini:
```

```
MQMPRD::PROCIDUser=JohnM
```

```
MQMPRD::PROCIDPwd=jg1jx7TePOnNVE0Rggy8C2==
```

This Page Intentionally Left Blank

Chapter 18: nsqmigr Utility

The nsqmigr utility is used to export/import workgroup server and node information in bulk. nsqmigr connects to the workgroup server and either inquires (exports) or updates (imports) database data to or from a `.info` file. Optional parameters can be set to export/import remote queue manager or remote EMS server information. See the sections immediately below for instructions on how to restore or export data using nsqmigr.

18.1 Parameters

For a full listing of nsqmigr's command line options, enter the following command.

```
nsqmigr -?
```

The command prompt will display the following:

```
E:\nastel\apwmq\bin>nsqmigr.exe ?
Usage: nsqmigr.exe

    -m<Workgroup name>
    [-O<Console output file>] -- Log of program output. Default: console
    [-U<User ID>] -- (Optional) User identifier to authenticate
    [-P<Password>] -- (Optional) User password
    [-R<Realm>] -- (Optional) Security realm
    -g | -p [-e ] | -de <ctp> -- Select one of these options [default: -g]:
        -g Get info (export from WGS database) or
        -p Put info (import to WGS database)
        -e Used with -p to encrypt clear text passwords
            that appear in the -f<Info file>
        -de Display encrypted form of the given clear text password
            Example: nsqmigr -de passwd
                    IbWPCfTv6KjM03qgODtClA==
    [-n<Node name>] -- Name of agent, CM or EMS node to put/get. Default: all)
    [-f<Info file>] -- Info file for get, put. Default:
                    [APWMQ_HOME]\config\groups\workgroup_name.in
                    If only file name is given, default directory is
                    [APWMQ_HOME]\config\groups.
    [-C] -- (Optional) Connection Manager remote queue managers info to put/get
    [-E] -- (Optional) EMS nodes info to put/get
    [-K] -- (Optional) Kafka nodes info to put/get
    [+v|-v] -- Enable | Disable verbose mode. Default: Enable
    [?|-?] -- Display This Help Message
```

NOTE: The info file will be left in place after the put (import).



Command line options are case sensitive.

18.2 Get / Put File

When exporting with nsqmigr, a `.info` file is created. If importing, you will need to create a `.info` either manually, or by using nsqmigr to first export a `.info` file and updating it accordingly.

The default export/import directory is as follows:

```
[APWMQ_HOME]/config/groups
```

The default output/input file name is `workgroup_name.info`. Both the default directory and file name can be overridden by using the `-f` parameter.

Please see the sections immediately below for further information on exporting and importing with nsqmigr.

18.3 Importing/Export Workgroup Server Databases

To recreate a WGS database, you will need to:

- 1) **Export (get) / create a backup:** Use the nsqmigr utility to create the `*.info` backup file. Run a command similar to the following to create this file:

```
nsqmigr -g -mworkgroup_name -ffilename.info
```

The default export directory is `[APWMQ_HOME]/config/groups`

- 2) **Import (put) / restore:** Place the updated `info` file in `[APWMQ_HOME]/config/groups` if it is not there already. When the WGS is restarted, the file will automatically load, and the database will be created. Please note that the `*.info` file is automatically deleted after it is successfully imported. It is first backed up in `[APWMQ_HOME]/config/groups/workgroup/history`.

18.3.1 Imported / Exported Fields

When using `-g` or `-p` without other parameters, nsqmigr will read/write the following fields if supplied:

<i>Workgroup Server Level</i>	<i>Node Level</i>	<i>Node Level</i>	<i>Node Level</i>
DiscoveryTime	NodeName	AuthInfoMask	QMgrLogSec
BroadcastIPNetwork	UseDNS	LsrMask	QMgrLogFileSz
BroadcastIPPortNo	HostName	SvcMask	QMgrLogBuffSz
TraceMode	IPAddress	TopicMask	AgentTrace
Description	IPPort	SubMask	Authorization
MaxNodes	Platform	ClntMask	NodeOSInfo
MaxClients	Discovery	Description	NodeCPUCount
MaxQmgrs	CmdLimit	NodeType	NodeMIPCount
AuthRecMode	Timeout	MaxWtTime	
MaxBrowse	HBInterval	TcpQLim	
PortNo	Update	RetryCount	
EvtDepth	Manage	Uid2Upper	
AgentTrace	QmgrMask	BufferSize	
Authorization	LocQueMask	TcpNoDelay	
AcctDepth	AIQueMask	ReuseAddr	
StatDepth	RmtQueMask	ObjNmIsDisc	
Refresh	MdlQueMask	NodePing	
ClntCmdLimit	ChIMask	DnsName	
Uid2Upper	ProcMask	QMgrLogPath	
AdtinfDepth	NmlMask	QMgrLogMeth	
Alias	ClusQmgrMask	od	
		QMgrLogPri	

18.3.2 Example Get/Put WGS nodes.info File

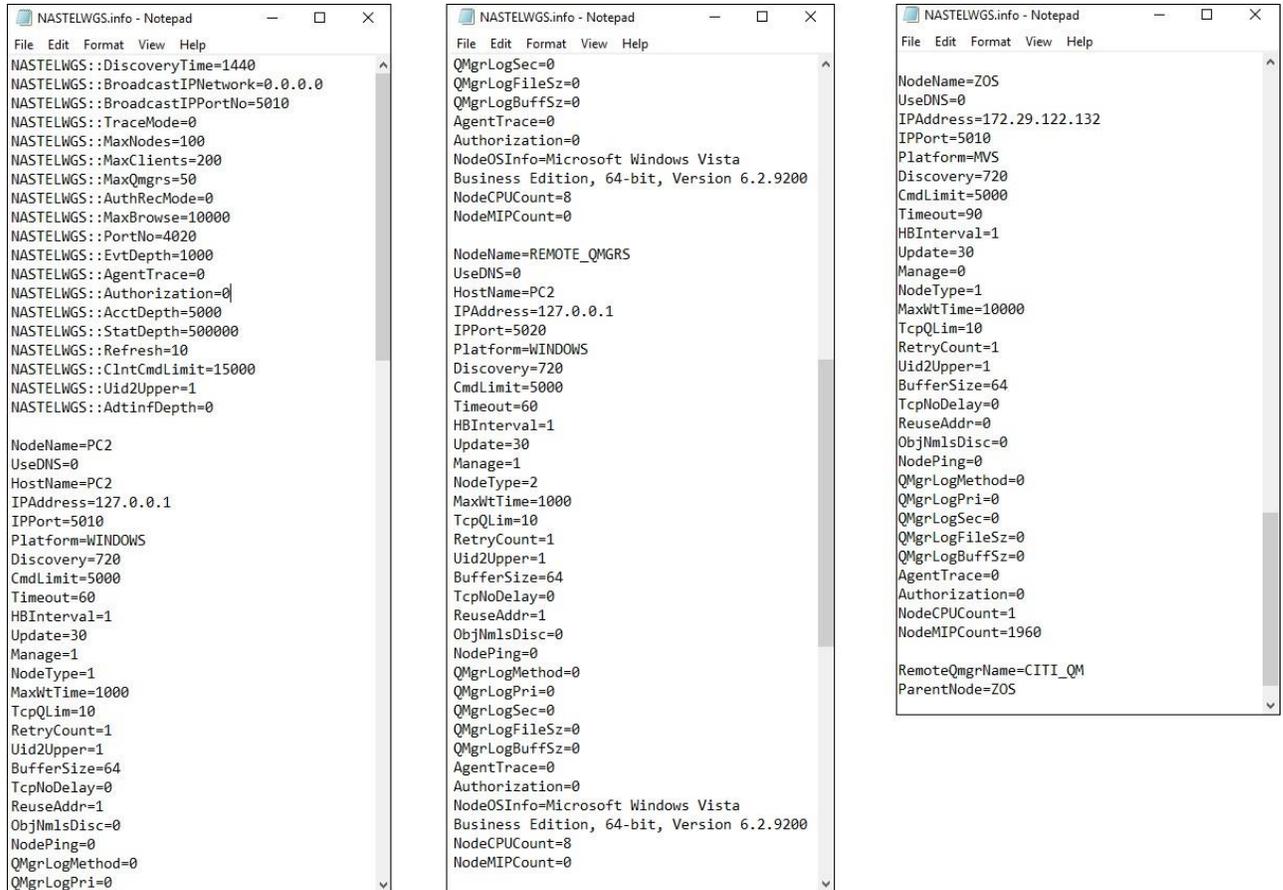


Figure 18.1 - Example Get / Put WGS File

18.4 Import/Export Remote Queue Managers

Use `-p` or `-g` with the `-C` parameter to import/export remote queue managers, which are managed by a connection manager (CM, nsqcm). nsqmigr will only include the remote queue managers. Note that the CM nodes are included in the info file created when all nodes are exported, by using `-g` without `-n,-C,-E`.

nsqmigr will read/write the following fields if supplied:

RemoteQmgrName
ParentNode
ConnectionName
ChannelName
SSLKeyRepos
SSLCrypHrdw
SSLCipherSpec
ConvertMsg
SecExit
SecExitData
Username
PW

18.4.1 Bulk Importing Remote Queue Managers (-p parameter)

Perform the following to bulk import remote queue managers:

- 1) Either
 - a. manually create the `filename.info` file, such as `CM_RemQmgrs.info`, to define each remote queue manager being imported. Add the fields listed immediately above for each remote queue manager; or
 - b. run command `nsqmigr -mworkgroup -g -C -ffilename.info` to export the remote queue managers from one or more workgroup servers, and then merge the files into one.
- 2) Place the file in `[APWMQ_HOME]/config/groups`. Please note that the `*.info` file is automatically backed up in `[APWMQ_HOME]/config/groups/workgroup/history` after importing and is not deleted.
- 3) Run a command similar to the following to import the file, perhaps into another workgroup. The `-C` parameter must be uppercase.
`nsqmigr -mworkgroup -p -C -ffilename.info`
- 4) The remote queue managers are now imported in to the WGS database. After they are imported, the import file is backed up in the `history` directory.

18.4.2 Bulk Exporting Remote Queue Managers (-g parameter)

Perform the following to bulk export remote queue managers:

- 1) Use a command similar to the following to create a file containing a list of all remote queue managers. The `-C` parameter must be uppercase.

```
nsqmigr -g -C -filename.info
```

- 2) The `filename.info` file is now created. For each remote queue manager restored, the following two messages will appear on the console:

```
Creating Remote Queue Manager qmgr_name...  
Remote Queue Manager created successfully!
```

The info file, such as `MQM.info`, will list all remote queue managers and their fields, as in Figure 18.2 below.

The default export directory is `[APWMQ_HOME]/config/groups` and the default output info file is `workgroup_name.info`. Both can be overridden by using the `-f` parameter.

18.4.3 Example Get/Put Remote Queue Managers Commands and Info File

The following is an example of `-get` and `-put` operations and a sample output remote queue managers file, `remote_qmgrs.info`.

- **Using the `-p` command:** You need to create this file manually or have one from a prior export before importing it. The following command would put, or import, the data into the WGS database:

```
C:\nastel\apwmq\bin>nsqmigr -mWGS38 -p -C
-fC:\nastel\apwmq\config\groups\remote_qmgrs.info
```

- **Using the `-g` command:** This command would get, or export, the remote queue managers from the WGS database to file `remote_qmgrs.info`:

```
C:\nastel\apwmq\bin>nsqmigr -mWGS38 -g -C
-fC:\nastel\apwmq\config\groups\remote_qmgrs.info
```

```
remote_qmgrs.info - Notepad
File Edit Format View Help
ParentNode=REMOTE_QMGRS
ConnectionName=1xx.0.0.xxx(1500)
ChannelName=IBM.SVR.CONN
SSLKeyRepos=
SSLCrypHrdw=
SSLCipherSpec=
ConvertMsg=
SecExit=
SecExitData=
ConvertMsg=0
Username=user
PW=gGTZtBxqvHDIByWu0xsTkA==

RemoteQmgrName=ORIS
ParentNode=REMOTE_QMGRS
ConnectionName=11x.0.0.xxx(1502)
ChannelName=ORIS.SVR.CONN
SSLKeyRepos=
SSLCrypHrdw=
SSLCipherSpec=
ConvertMsg=
SecExit=
SecExitData=
ConvertMsg=0
Username=user
PW=gGTZtBxqvHDIByWu0xsTkA==
```

Figure 18.2 - Example Remote Queue Managers Info File

18.5 Import/Export Remote EMS Servers

Use `-p` or `-g` with the `-E` parameter to import/export Remote EMS Servers (`nsqcmems.jar`). `nsqmigr` will only include nodes that are Remote EMS Nodes. `nsqmigr` will read/write the following fields if supplied:

EMS::ManagerName	SSLCiphers
ParentNode	SSLVendor
Server	SSLHostname
SrvUrl	SSLVerifyHostName
Username	SSLVerifyHost
Password	SSLTrace
SSLTrustCertificate	SSLDebugTrace
SSLIdentity	
SSLIssuer	
SSLPwd	
SSLKey	

18.5.1 Bulk Importing Remote EMS Servers (-p parameter)

Perform the following to bulk import remote EMS servers:

- 1) Create the `*.info` file, such as `remote_ems.info`, to define each remote EMS server being imported. Add the fields listed above for each remote EMS server.
- 2) Place the file in `[APWMQ_HOME]/config/groups.`
- 3) Run a command similar to the following to import the file. Please note that the `-E` parameter must be uppercase.


```
nsqmigr -p -E -f<filename>.info
```
- 4) The remote EMS servers are now imported in to the WGS database. The info file is automatically backed up in `[APWMQ_HOME]/config/groups/workgroup/history` and is not deleted.

18.5.2 Bulk Exporting Remote EMS Servers (-g parameter)

Perform the following to bulk export remote EMS servers:

- 1) Use a command similar to the following to create a `*.info` containing a list of all remote EMS servers. Please note that the `-E` parameter must be uppercase.


```
nsqmigr -g -E -f<filename>.info
```
- 2) The `*.info` file is now created. It will list all remote EMS servers and the fields listed in section 17.5 above. The default export directory is `[APWMQ_HOME]/config/groups`, but the default can be overwritten by using the `-o` parameter.

18.5.3 Example Get/Put Remote EMS Server Commands and Info File

Below is an example of a Remote EMS Server file.

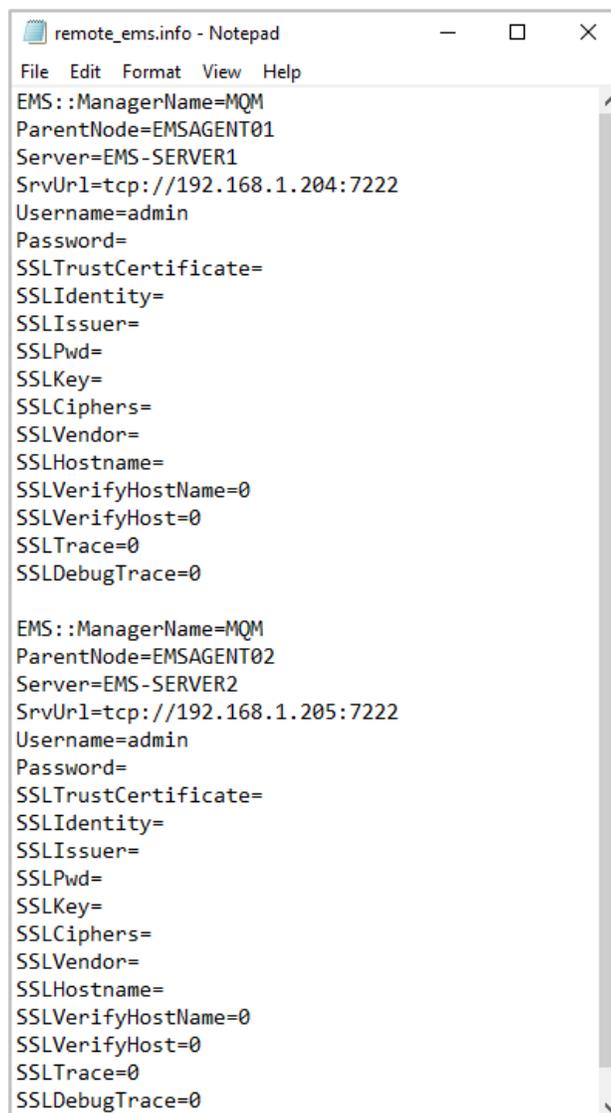
- **Using the -p command:** You need to create this file manually or have one from a prior export before importing it. The following command would put, or import, the data into the WGS database:

```
C:\nastel\apwmq\bin>nsqmigr -p -E  
-fC:\nastel\apwmq\config\groups\remote_ems.info
```

- **Using the -g command:** this file would be exported by nsqmigr. The following command would be run to generate it:

This command would get, or export, the remote EMS servers from the WGS database to the file `remote_ems.info`:

```
C:\nastel\apwmq\bin>nsqmigr -g -E  
-fC:\nastel\apwmq\config\groups\remote_ems.info
```



```
remote_ems.info - Notepad  
File Edit Format View Help  
EMS: :ManagerName=MQM  
ParentNode=EMSAGENT01  
Server=EMS-SERVER1  
SrvUrl=tcp://192.168.1.204:7222  
Username=admin  
Password=  
SSLTrustCertificate=  
SSLIdentity=  
SSLIssuer=  
SSLPwd=  
SSLKey=  
SSLCiphers=  
SSLVendor=  
SSLHostname=  
SSLVerifyHostName=0  
SSLVerifyHost=0  
SSLTrace=0  
SSLDebugTrace=0  
  
EMS: :ManagerName=MQM  
ParentNode=EMSAGENT02  
Server=EMS-SERVER2  
SrvUrl=tcp://192.168.1.205:7222  
Username=admin  
Password=  
SSLTrustCertificate=  
SSLIdentity=  
SSLIssuer=  
SSLPwd=  
SSLKey=  
SSLCiphers=  
SSLVendor=  
SSLHostname=  
SSLVerifyHostName=0  
SSLVerifyHost=0  
SSLTrace=0  
SSLDebugTrace=0
```

Figure 18.3 - Example Remote EMS Servers Info File

This Page Intentionally Left Blank

Appendix A: Nastel and IBM Documentation

This appendix contains a list of reference material and documents relevant to Nastel Navigator.

A.1 Nastel Documentation

Table A-1. Nastel Documentation	
Document Number (or higher)	Title
NAV/WGS 101.017	Nastel Navigator Workgroup Server Expert V10 - Installation Guide
NAV-SM 667.002	Nastel Navigator Security Manager V6 - User's Guide
NN10.019	Nastel Navigator User's Guide
M6/MQ 664.001	AutoPilot M6 Plug-in for IBM MQ
M6-INS 630.002	Nastel AutoPilot M6 - Installation Guide
M6/USR 630.003	Nastel AutoPilot M6 - Administrator's Guide

A.2 IBM Documentation

Using System Management Facility

<https://www.ibm.com/docs/en/ibm-mq/8.0?topic=monitoring-using-system-management-facility>

Using CSQ6SYSP

<https://www.ibm.com/docs/en/ibm-mq/8.0?topic=module-using-csq6sysp>

Monitoring performance and resource usage

<https://www.ibm.com/docs/en/ibm-mq/9.0?topic=network-monitoring-performance-resource-usage>

This Page Intentionally Left Blank

Appendix B: Nastel Navigator Command Lines by Component

Nastel Navigator provides the same basic features on most supported platforms. On some platforms, however, there are minor differences. These differences are summarized in this appendix. This section contains the following:

- Nastel Navigator components
- The names of their executables
- The platforms they run on
- The command-line arguments they accept.

B.1 IBM MQ Agent

The IBM MQ agent is an application named nsqmq (.exe).

B.1.1 Platforms

The IBM MQ agent is available for all platforms currently supported by Nastel Navigator, as listed below:

- Windows
- OS/400 (iSeries)
- Unix (Linux, Solaris, AIX, HP_UX)
- z/OS

B.1.2 Command-Line Arguments

Table B-1 lists all command-line arguments applicable to the IBM MQ Agent.



The +a parameter *cannot* be used when running in conjunction with Nastel AutoPilot M6. Do not change the default -a setting.

Table B-1. IBM MQ Agent Command-Line Arguments	
Arguments	Description
Optional Arguments	
-Q<mask>	Manage only queue manager whose names start with <mask> string.
-D<name.com>	DNS name to create fully qualified host name.
-m<WorkgroupName>	Specifies the workgroup server to which the MQ Agent must register.
-m* (asterisk)	Registers the MQ Agent under all available workgroup servers.

Table B-1. IBM MQ Agent Command-Line Arguments

Arguments	Description
	You can also use the asterisk to represent partial names.
-t +t	Turns on off PCF protocol trace. Default: trace off
-tu +tu	Turns on off authorization trace. Default: trace off
-x +x	Turns on off tracing for TCP/IP packets. Default: off
-e	Reuse TCP/IP address when binding. If port is in use or still lingering, workgroup can forcibly reclaim it. Default: off
+E	Enable encryption when communicating with the Workgroup Server.
-f	Force full discovery for initial discovery. Default: off
-r<retries>	Number of attempts MQ agent makes to bind to an available TCP Port. It starts with specified port followed by successive port numbers until a port is found that is not in use. Default: 1
-n<AliasNodeName>	Assigns a new name to be used when registering with workgroup servers. This name will override the name of the local host.
+u -u	Enables/disables authorization checks for letting users submit commands to the WebSphere MQ command server. +u gives all users effective authority over the MQ Agent. Default is Authorization checked off.
+q -q	Enables/disables queue manager verification. This option is needed to test if queue manager is properly configured for M6 MQ agent. Default: disabled.
-c<subtasks> (z/OS only)	Specifies the maximum number of z/OS MTF subtasks.
-P +P (z/OS only)	Run PCF Command Server as MTF subtask (-P) or in independent address space (+P). Default: MTF subtask.
-R (z/OS only)	Queue manager connect retry time (secs) when PCF Command Server running as MTF subtask. Default: 30 sec.
+z -z	Enables/disables use of DNS (Domain Name Server) for host name resolution. Default: enabled. Disabling of DNS should be used on systems where an Object Detached event states in its Description field that a host name cannot be resolved by the MQ Agent.
+i -i	Enable/disable reporting node IP address to workgroup server. Default: on
-p<time>	Time interval to send registration to workgroup server periodically (in minutes). Sent only at startup time.
-b<Buf.Size>	Set TCP send/receive buffer size in KB. Default: 64KB

Table B-1. IBM MQ Agent Command-Line Arguments

Arguments	Description
+g -g	Enables/disables TCP Nagle algorithm for send coalescing on TCP/IP connection. Improves TCP performance. Default: disabled.
-s<port service_name>	Specifies the port or service name to which the MQ Agent will bind. Default: 5000
-v	Verify queue manager and communication only.
-C	Disables user id conversion to uppercase. By default, the user id is converted to uppercase and passed to WMQ queue manager as an alternate user id.
[-l<Que. Limit>]	TCP buffering queue limit. This option sets soft limit to buffer responses from WMQ to be sent to the workgroup. The smaller the buffer size, the more responsive the GUI will be. Default: 10
-w	Maximum wait time (msec) for socket read/write event. Default: 1000
-tN	Enable debug trace of name list discovery. Default: off
-N	Enable special name list discovery. Use on z/OS when thousands of queues, channels, processes, namelists, or auth info objects are defined and IBM 64KB PCF message size limit exists. Default: Disabled
-f	Force full discovery for initial discovery. Default: off
-?	Display usage/help messages.
Windows only arguments (Used only when running this program from the command prompt)	
-install	Installs the MQ Agent as a service (if not done during installation.)
-remove	Removes the MQ Agent as a service.
-console	Runs the MQ Agent as a console application.

B.2 MMF Server

The MMF (Message Management Facilities) Server is an application named nsqmsg (.exe). A second application, endmcs (.exe), is used to stop the MMF Server.

B.2.1 Platforms

The MMF Server is available for all platforms currently supported by Nastel Navigator:

- Windows
- OS/400 (iSeries)
- Unix (Linux, Solaris, AIX, HP_UX)
- z/OS

B.2.2 Command-Line Arguments

Table B-2 lists all command-line arguments applicable to the MMF Server.

Table B-2. Start MMF Server Command-Line Arguments	
Arguments	Description
Optional Arguments:	
-m<Qmgr_Name>	Name of the IBM MQ queue manager where MMF is required.
-r<N>	Set reconnect retry number
-g<Grace>	Set grace period in seconds between reconnects
-a	Enable command auditing
-b	Enable binary representation of message body
-c	Enable conversion on MQGET
-t	Enable EPCF protocol trace
-?	Display usage information
Windows only arguments (Used only when running this program from the command prompt)	
-install	Installs MMF Server as a service (if this has not already been done during installation.)
-remove	Removes MMF Server as a service.
-console	Runs MMF Server as a console application.



NOTE

“-m*” will attach MMF Server to all QMGRs on Windows platforms.

Table B-3. Stop MMF Server Command-Line Arguments

Arguments	Description
Required Arguments	
QueueManager	Queue Manager whose MMF Server is to be stopped
Optional Arguments	
CommandQueue	Command Queue to send stop command to. Default: NASTEL.MMF.ADMIN.COMMAND.QUEUE

B.3 MQ Event Publisher

The MQ Event Publisher is an application named nsqpub(.exe). A second application, pubend.exe is used to stop the MQ Event Publisher.

B.3.1 Platforms

The MQ Event Publisher is available on all platforms currently supported by Nastel Navigator, as listed below:

- Windows
- z/OS
- OS/400 (iSeries)
- Unix (Linux, Solaris, AIX, HP_UX)

B.3.2 Command-Line Arguments

Table B-4 lists all command-line arguments applicable when starting the MQ Event Publisher.

Table B-4. Starting MQ Event Publisher Command-Line Arguments		
Arguments	Description	Default
<code>-f<perm_subscriber_file></code>	Path to permanent subscription file. PubSub distributes according to subscriptions. Nastel Navigator agent subscription request is stored in temp subscription file. This file is stored in the same directory and has name in the form QMGR.ini. Other applications may subscribe to events only thru permanent subscription file.	\\subscrib (UNIX) dd:subscrib (z/OS)
<code>-i<temporary_subscriber_file_HLQ></code>	Defines high level qualifier (HLQ) of temporary subscription files. z/OS: If specified, structure of the name is HLQ.QMGRName.INI. If not specified, structure of the name is UserID.QMGRName.INI. Other platforms: If specified, structure of the name is HLQ.MGRName.ini. If not specified, structure of the name is QMGRName.ini.	An empty string
<code>-e<SystemEventQ></code>	EventQ. Defines the name of the event queues. These queues are inbound for MQ Event Publisher. All queue managers must have same event queue name.	NASTEL.PUBSUB. EVENT.QUEUE
<code>-d<DeadLetterQ></code>	Defines the default name of the dead letter queues (DLQs) for all queue managers.	SYSTEM.DEAD. LETTER.QUEUE
<code>-w<WaitInterval></code>	Defines the wait interval, in milliseconds, on MQGET calls.	30000

Table B-4. Starting MQ Event Publisher Command-Line Arguments

Arguments	Description	Default
-l<DisableListenDLQ>	If specified, will not listen on DLQ.	Listens on DLQ.
-q<DLQTimeRate>	Defines time interval (milliseconds) that must elapse before MQ Event Publisher will report on a subsequent identical DLQ message.	300000
-n<DLQCountRate>	Defines number of messages that must elapse before MQ Event Publisher will report on a subsequent identical DLQ message.	1000
-k<DLQDeleteRate>	Defines time interval (in milliseconds). During this time interval the information about a reported DLQ message will be kept for future frequency testing.	3000000
-v<DLQEventsMax>	Defines maximum number of unique DLQ events.	100
-o<ShortReconnMax>	Defines number of short MQ reconnections.	5
-h<ShortReconnInt>	Defines time interval (milliseconds) for short MQ reconnect retries.	5000
-g<LongReconnInt>	Defines time interval (milliseconds) for long MQ reconnect retries.	60000
-c<SubscribersMax>	Defines the maximum number of subscribers.	100
-b<QMGRsMax>	Defines maximum number of queue managers. (Maximum number of queue managers is 50.) z/OS only: one queue manager requires three threads per connection which includes one thread for Publisher parent process and two DLQ threads. This translates to 1M ~ 4M of storage utilization by PubSub process. Since default maximum number of queue managers is 20, due to z/OS constraints (REGION=), PubSub process should be run with -b = actual number of queue managers.	20
-r<UDPConnRetry>	Defines the number of retries on a UDP connection.	5
-u	If specified, will reuse the UDP address. Publisher uses UDP port 5500 (default).	Off

Table B-4. Starting MQ Event Publisher Command-Line Arguments

Arguments	Description	Default
-s<ServiceName>	Defines the UDP port to send a SHUTDOWN_MGR command event to the MQ Event Publisher. The Publisher listens on this port number. It must be the same value as -sServiceName, specified when the Publisher was started.	5500
-p	Enable duplicate event suppression.	Disabled
-a	Duplicate event counter or suppression factor. (Maximum number of duplicates.)	100
-x	Event expiry within EventList in seconds.	60
-t	If specified, turns tracing on.	Off
-P/+P	Enable multi- or single-threaded process	+P; Single-threaded
-S	Do not use shared memory to shutdown process.	Off
-W (z/OS only)	Enable WTO logging.	Off
-?	Display usage information.	

Table B-5 lists all command-line arguments applicable when stopping the MQ Event Publisher.

Table B-5. Stopping MQ Event Publisher Command-Line Arguments

Arguments	Description	Default
-r<UDPConnRetry>	Defines number of retries on a UDP connection.	5
-u	If specified, will reuse UDP address. Publisher uses UDP port 5500 (default).	Off
-l<LstnrSrvName>	Defines UDP listening port used by pubend program for receiving PCF messages.	6000
-s<ServiceName>	Defines UDP port to send a SHUTDOWN_MGR command event to MQ Event Publisher. Publisher listens on this port number. It must be same value as: -sServiceName, specified when the Publisher was started.	5500
-h<HostName>	Defines MQ Event Publisher host name.	Localhost
-i<HostIPAddress>	Defines MQ Event Publisher host IP address.	127.0.0.1 (localhost)
-t	If specified, turns tracing on.	Off

Table B-5. Stopping MQ Event Publisher Command-Line Arguments

Arguments	Description	Default
-?	Display usage information.	

B.4 MQSC Utility

The MQSC utility is an application named nsqmqsc (.exe).

B.4.1 Platforms

The MQSC utility is available on the following platforms:

Windows

Unix (Linux, Solaris, AIX, HP_UX)

B.4.2 Command-Line Arguments

Table B-6 lists all command-line arguments applicable to the MQSC utility.

Table B-6. MQSC Utility Command-Line Arguments

Arguments	Description
Required Arguments:	
-m<GroupName>	Workgroup server name.
Optional Arguments:	
-n<NodeName>	Node name.
-q<QmgrName>	Name of queue manager.
-d [-r] [>FileName]	Dumps the database, in MQSC format, to the console or redirect to the output file. Cannot be used with -f.
-r	Adds REPLACE to every DEFINE statement, to force changes when the script is applied to a queue manager. Use with -d.
-f<FileName>	Restores the database for the given node and queue manager from the given MQSC file. Cannot be used with -d.
-v	Enables verbose mode.
-t	Enables debug trace.
-U	Username for authenticating with workgroup server. Default: current user.
-P	Password for authenticating with workgroup server. Default: current user.
-R	Security realm for authenticating with workgroup server. Default: security realm for current user.

-a	Don't authenticate with workgroup server. Connect to workgroup server as GUEST user.
-w<WaitInterval>	Number of seconds to wait for reply messages. Default: infinite.

B.5 nsqping (Ping) Utility

The Ping Utility allows for testing of IBM MQ response time and IBM MQ links. The Ping Utility can be used only with local queue managers, but it can test links to remote queue managers. To use the utility, the NSQPING symbol must be installed.

B.5.1 Platforms

The nsqping utility is available on the following platforms:

Windows

Unix (Linux, Solaris, AIX, HP_UX)

z/OS

B.5.2 Command-Line Arguments

Table B-7 lists all command-line arguments applicable to the MQSC utility.

Table B-7. Ping Utility Command-Line Arguments		
Arguments	Description	Default
Required Arguments		
-m<QmgrName>	Specifies the name of the local queue manager.	N/A
-q<QueueName>	Queue name of any type.	N/A
Optional Arguments		
-b<BatchSize>	Number of messages in a batch.	10
-s<Bytes>	Size, in bytes, of generated messages.	512
-d<Interval>	Interval, in seconds, that ping messages are generated. An interval of 0 (zero) indicates to ping only once.	0
-w<Wait>	Wait interval, in milliseconds, before timeout.	60000
-e<Expiry>	Message expires after Expiry time in 0.1 seconds.	Unlimited
-u<AltUser>	Issues ping on behalf of another user.	N/A
-cco	Generates confirm-on-arrival report.	False
-ccod	Generates confirm-on-delivery report.	False
-cexp	Generates confirm-on-expiry report.	False
-cexc	Generates confirm-on-exception report.	False

Table B-7. Ping Utility Command-Line Arguments

Arguments	Description	Default
-cdlq	Generates message that will go onto DLQ if undelivered.	False
-h<HostName>	Host name of the target Nastel AutoPilot M6 Process Wrapper	localhost
-p<Port>	Port number of the target Nastel AutoPilot M6 Process Wrapper	6000
-v<0 1 2>	Sets verbose mode to 0 – summary; 1- reports; 2- all	0

B.6 nsqcode Utility

The nsqcode utility is used to identify and define IBM MQ and Nastel Navigator system error codes. If the code you received is not listed, contact either IBM or Nastel support for further assistance.

B.6.1 Platforms

The MQSC utility is available on the following platforms:

Windows
 OS/400 (iSeries)
 Unix (Linux, Solaris, AIX, HP_UX)
 z/OS

B.6.2 Additional Information: Using nsqcode

To identify error definitions, perform the following steps.

1. At the command prompt go to your operating system directory (example: C:\[winnt]).
2. Type: `nsqcode <code>`. Where code= error code received.
(example: `nsqcode IME0001`)
3. The error code and its definition as well as possible variations will be displayed as shown below.

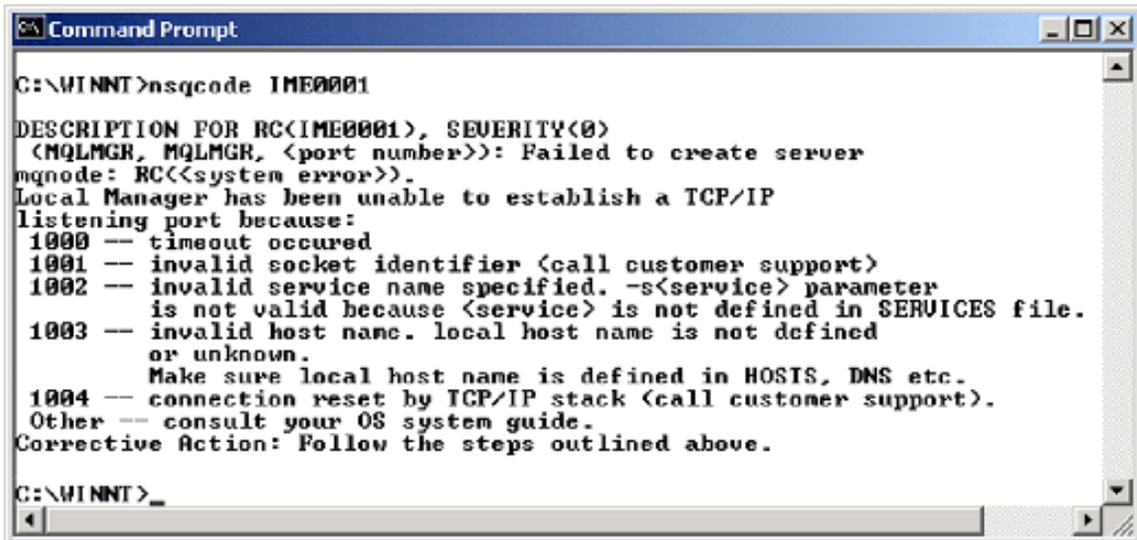


Figure B-1. nsqcode Definitions

B.7 Log Adapter

The log adapter is an application named `nsqadp (.exe)`.

B.7.1 Platforms

The log adapter is available on the following platforms:

- Windows
- Unix (Linux, Solaris, AIX, HP_UX)

B.7.2 Command-Line Arguments

Table B-8 lists all command-line arguments applicable to the log adapter.

Table B-8. Log Adapter Command Line Arguments	
Arguments	Description
Required Arguments	
<code>-m<GroupName></code>	Specifies the name of the workgroup to connect to.
Optional Arguments	
<code>-?</code>	Displays usage/help message.
<code>-U</code>	Username for authenticating with workgroup server. Default: current user
<code>-P</code>	Password for authenticating with workgroup server. Default: current user
<code>-R</code>	Security realm for authenticating with workgroup server. Default: security realm for current user.

Table B-8. Log Adapter Command Line Arguments

Arguments	Description
-a	Don't authenticate with workgroup server. Connect to workgroup server as a GUEST user.
-i	Enables ITO environment. Default: Enable Tivoli environment.
-t<TimeOut>	Interval, in msec, between attempts to connect to workgroup server. Default: 10000 (10 seconds)
-l [LogName]	Writes messages to log file. Defaults: For ITO: if no -l, write events using opcmsg For ITO and Tivoli: if -l with no LogName, default log is: Windows: ITO: \\usr\OV\nsq\log\nsqadp_GM.log Tivoli: \\temp\group_name.log UNIX: ITO: /var/nsq/log/nsqadp_GM.log Tivoli: /tmp/group_name.log
-s (UNIX only)	Writes to UNIX syslog. Not compatible with -i and -l.
-F<n> (UNIX only)	syslog facility index, 0-7, for facility LOG_LOCAL<n>. Default: 0
-b (UNIX only)	Runs the Log Adapter in the background as a daemon
-r<all nn none>	Appends reason code description text: all: Entire text of descriptions <nn>: first <nn> bytes of descriptions none: Don't append reason code descriptions Default: <nn> = length of message ID
-f<cc>	Uses field delimiter <cc> in the messages, maximum 2 chars. Use -fsp to use space character as delimiter. Default: ':'
-h	Interval, in seconds, to send heartbeat message if no event message was sent in the last <Interval> seconds. Default: 0 (don't send heartbeat message)
-d	Enables debug trace
-?	Displays usage information
Windows only arguments (Used only when running this program from the command prompt)	

Table B-8. Log Adapter Command Line Arguments

Arguments	Description
-install	Installs Log Adapter as a service.
-remove	Removes Log Adapter as a service.
-console	Runs Log Adapter as a console application.

B.8 Message Management Utility

The message management utility is an application named `nsqmmfcl (.exe)`.

B.8.1 Platforms

The message management utility is available on the following platforms:

- Windows
- Unix (Linux, Solaris, AIX, HP_UX)
- OS/400 (iSeries)
- z/OS

B.8.2 Command-Line Arguments

Table B-9 lists all command-line arguments applicable to the message management utility.

Table B-9. Message Management Utility Command-Line Arguments

Arguments	Description
Required Arguments	
-m<GroupName>	Specifies the name of the workgroup to connect to.
Optional Arguments	
-n<NodeName>	Accesses queues from named IBM MQ Node.
-q<QmgrName>	Accesses queues from named Queue Manager.
-U<User>	Username for authenticating with workgroup server. Default: current user
-P<Password>	Password for authenticating with workgroup server. Default: current user
-R<Realm>	Security realm for authenticating with workgroup server. Default: security realm for current user
-a	Don't authenticate with Workgroup Server. Connect to workgroup server as a GUEST user.
+v -v	Enables/Disables verbose mode. Default: Enabled

B.9 Queue Monitor Utility

The queue monitor utility is an application named `nsqmon (.exe)`.

B.9.1 Platforms

The queue monitor utility is available on the following platforms:

Windows

Unix (Linux, Solaris, AIX, HP_UX)

B.9.2 Command-Line Arguments

Table B-10 lists all command-line arguments applicable to the queue monitor utility.

Table B-10. Queue Monitor Utility Command-Line Arguments	
Arguments	Description
Required Arguments	
<code>-m<GroupName></code>	Specifies the name of the workgroup to connect to
<code>-n<NodeName></code>	Monitors queues from named IBM MQ Node
<code>-q<QmgrName></code>	Monitors queues from named Queue Manager
Optional Arguments	
<code>-o<QueueName></code>	Monitors specific queue. Default: all queues
<code>-U<User></code>	Username for authenticating with workgroup server Default: current user
<code>-P<Password></code>	Password for authenticating with workgroup server Default: current user
<code>-R<Realm></code>	Security realm for authenticating with workgroup server Default: security realm for current user
<code>-a</code>	Don't authenticate with workgroup server. Connect to workgroup server as a GUEST user.

B.10 Message Get/Put Utility

The message get/put utility is an application named `nsqmsgf (.exe)`.

B.10.1 Platforms

The message get/put utility is available on the following platforms:

Windows

Unix (Linux, Solaris, AIX, HP_UX)

OS/400 (iSeries)

z/OS

B.10.2 Command-Line Arguments

Table B-11 lists all command-line arguments applicable to the message get/put utility.

Table B-11. Message Get/Put Utility Command-Line Arguments	
Arguments	Description
Required Arguments	
<code>-m<QmgrName></code>	Queue Manager to connect to.
<code>-l<QueueName></code>	Queue to get/put messages to/from.
Optional Arguments	
<code>-g<0 1></code>	Gets messages from queue 0 = browse queue, leaving messages on queue 1 = read and remove messages from queue This is the default if <code>-p</code> is not specified
<code>-p</code>	Puts messages onto queue.
<code>-f<FileName></code>	Name of file to read messages from (for put) or to write messages to (for get). Default: <code><QueueName>.msg</code>
<code>-d<1,2></code>	Indicates character(s) used to separate messages 1 = Messages separated by newline (“\n”) 2 = Messages separated by carriage-return newline sequence (“\r\n”) Default: 1
<code>-c</code>	Enables conversion of application data to conform to CCSID and Encoding. Default: Disable conversion
Optional Arguments for getting messages	
<code>-w<WaitInterval></code>	Interval, in seconds, to wait for message. Default: 1
<code>-s<1,2></code>	Strips IBM MQ headers from messages 1 = Strip Message Descriptor (MD) header 2 = Strip all headers Default: Don't strip headers
<code>-b<MessageCursor></code>	Cursor of message to start reading from. Range: 1 – 999999999 Default: 1
<code>-n<MessageCount></code>	Number of messages to get. Range: 1 – 999999999 Default: all messages
<code>-r</code>	Generate replies to messages. Default: don't generate replies

Table B-11. Message Get/Put Utility Command-Line Arguments

Arguments	Description
Optional Arguments for putting messages	
-i<1,2,3>	Indicates how to interpret contents of <FileName> 1 = Treat entire file as a single message 2 = Treat file as containing many messages (delimited as defined by -d option) 3 = Treat file as an XML file where each XML structure defines a separate message
-h	Generates default Message Descriptor (MD) for messages
-o	Generates message with zero-length application data

B.10.3 Options to Get Messages from Queue to File

```
-m<Queue Manager name>
-q<Queue name>
[-g<0,1> (0 - BROWSE, 1 - GET)
[-w<Wait interval in sec.>]
[-f<File name>]
[-s<1,2>] (Strip WMQ headers, 1 - MD, 2 - ALL)
[-d<1,2>] (Message Separator, 1 - NL, 2 - @@)
[-b<Message Cursor> (1 - 999999999)
[-n<Message Count> (1 - 999999999)
[-c] (Enable conversion)
[-r] (Generate reply)
```

Example 1:

1. Transfer messages from queue to a file

a. First put two sample messages on a queue:

```
>amqspout AAA.LQ QMA71
Sample AMQSPUT0 start
target queue is AAA.LQ
message 1
message 2
Sample AMQSPUT0 end
```

b. Now get the messages from the queue to a file in browse mode (the messages remain on the queue):

```
nsqmsgf -mQMA71 -qAAA.LQ -g0 -fAAA.LQ.msgs -d2
```

Command Options:

Queue Manager : QMA71

Queue Name : AAA.LQ

```
File Name : AAA.LQ.msgs
Wait Interval : 1 seconds
MD Conversion : Disabled
Message Separator : @@
Strip MQ Headers : None
Message Cursor : 1
Message Count : -1
Delete Messages : NO
Queue AAA.LQ:
    Depth : (2\100000)
    GET : ALLOWED
Getting message 1, size of 9 bytes, format 'MQSTR' ...
Getting message 2, size of 9 bytes, format 'MQSTR' ...
No more messages available.
Current queue depth is 2
```

- c. Show the message file:

```
>dir AAA*
Volume in drive C is HP
Volume Serial Number is 0C14-E063
Directory of C:\temp
05/17/2016 11:01 AM 750 AAA.LQ.msgs
1 File(s) 750 bytes
0 Dir(s) 32,397,086,720 bytes free
```

- d. Verify the messages are still on queue:

```
>echo dis qlocal(AAA.LQ) curdepth |runmqsc QMA71
5724-H72 (C) Copyright IBM Corp. 1994, 2011. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QMA71.
1 : dis qlocal(AAA.LQ) curdepth
AMQ8409: Display Queue details.
QUEUE(AAA.LQ) TYPE(QLOCAL)
CURDEPTH(2)
One MQSC command read.
```

No commands have a syntax error.

All valid MQSC commands were processed.

B.10.4 Options to Put Messages from File to Queue

```
-m<Queue Manager name>
-q<Queue name>
[-p<PUT>]
[-f<File name>]
[-i<1,2,3>]
i1 - Import file as 1 message;
i2 - Import text file as many messages;
i3 - Import XML file as many messages.
[-h<Generate default Message Descriptor>]
[-d<1,2>] (Message Separator, 1 - NL, 2 - @@)
[-c] (Enable conversion)
[-o] (Generate message with zero-length application data)
[-U<UserName>] - User ID to be used in queue manager authentication
[-P<Password>] - Password to be used in queue manager authentication
```

Example:

1. Restore the messages from example 1 to another queue BBB.LQ:

First check queue BBB.LQ:

```
>amqsbcg BBB.LQ QMA71
AMQSBCG0 - starts here
*****
MQOPEN - 'BBB.LQ'
No more messages
MQCLOSE
MQDISC
```

2. Restore the messages:

```
> nsqmsgf -mQMA71 -qBBB.LQ -p -fAAA.LQ.msgs -d2
```

Putting messages to the queue:

Command Options:

```
Queue Manager : QMA71
Queue Name : BBB.LQ
File Name : AAA.LQ.msgs
MD Conversion : Disabled
Message Separator : @@
Generate MQ Headers : No
```

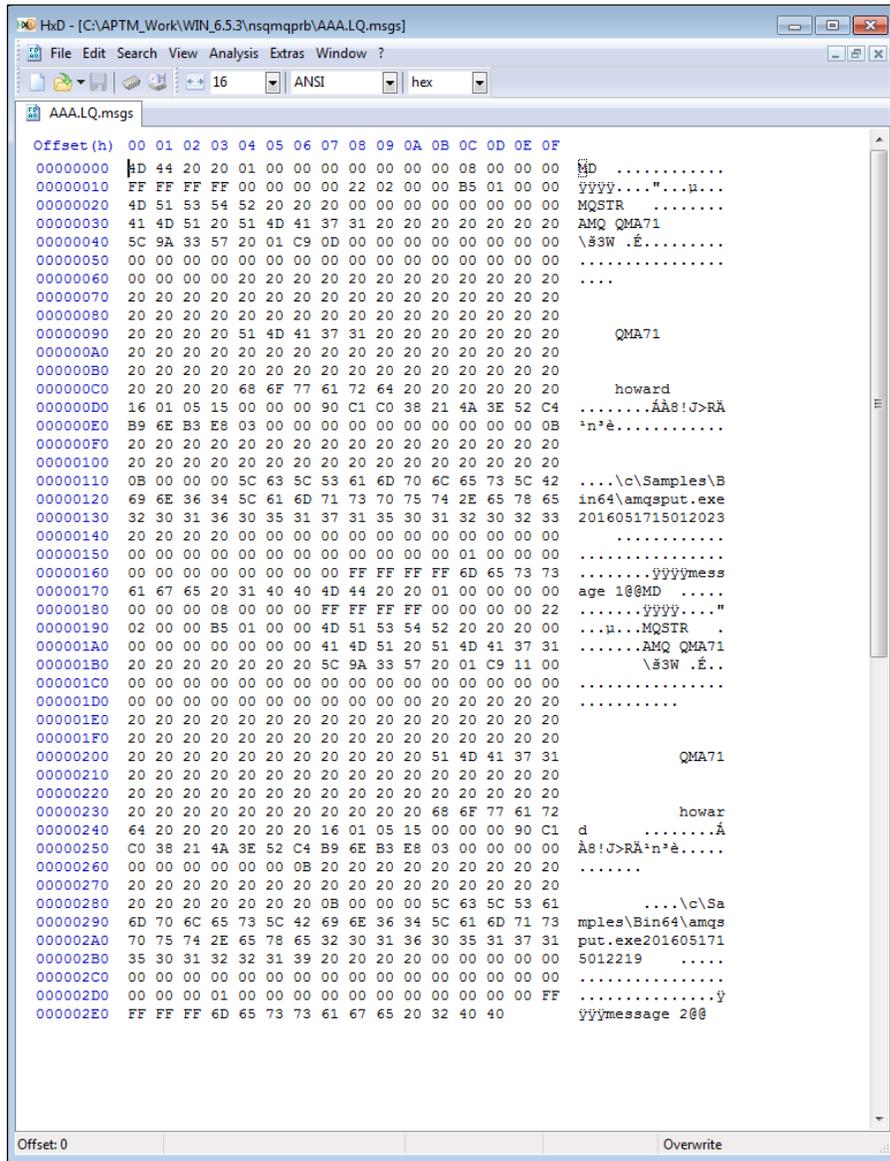



Figure B-2. Sample Dump of Exported Message File

B.11 Remote Shutdown Utility

The remote shutdown utility is an application named `nsqstop (.exe)`.

B.11.1 Platforms

The remote shutdown utility is available on the following platforms:

- Windows
- Unix (Linux, Solaris, AIX, HP_UX)
- OS/400 (iSeries)
- z/OS

B.11.2 Command-Line Arguments

Table B-12 lists all command-line arguments applicable to the remote shutdown utility for the remote workgroup server.

Table B-12. Remote Workgroup Server Shutdown Command-Line Arguments	
Arguments	Description
Required Arguments (if not silent installation)	
-m<GroupName>	Name of workgroup server to shut down.
Optional Arguments	
-n<NodeName>	Node where IBM MQ Agent should be shut down. Default: only workgroup server is shut down.
-g<GracePeriod>	Interval, in seconds, to wait for the workgroup server to shut down.
-U<User>	Username for authenticating with workgroup server. Default: current user
-P<Password>	Password for authenticating with workgroup server. Default: current user
-R<Realm>	Security realm for authenticating with workgroup server. Default: security realm for current user
-a	Don't authenticate with workgroup server. Connect to workgroup server as a GUEST user.

Table B-13 lists all command-line arguments applicable to the remote shutdown utility for the remote MQ Agent.

Table B-13. Remote IBM MQ Agent Shutdown Command-Line Arguments	
Arguments	Description
Required Arguments (if not silent installation)	
-n<NodeName>	Node where IBM MQ Agent should be shut down.
Optional Arguments	
-i<IPAddress>	IP Address of host IBM MQ Agent is running on. Default: 127.0.0.1
-s<IPPort>	TCP/IP Port that the IBM MQ Agent is listening on. Default: 5010

B.12 Trigger Message Utility

The trigger message utility is an application named `nsqtrig (.exe)`.

B.12.1 Platforms

The trigger message utility is available on the following platforms:

Windows

Unix (Linux, Solaris, AIX, HP_UX)

B.12.2 Command-Line Arguments

Table B-14 lists all command-line arguments applicable to the trigger message utility.

Table B-14. Trigger Message Utility Command-Line Arguments	
Arguments	Description
Required Arguments (if not silent installation)	
<code>-m<GroupName></code>	Specifies the name of the workgroup to connect to.
<code>-n<NodeName></code>	Name of IBM MQ Node to use.
<code>-q<QmgrName></code>	Name of Queue Manager to use.
<code>-l<LocalQueueName></code>	Name of Local Queue on <code><QmgrName></code> to put trigger message on.
Optional Arguments	
<code>-U<User></code>	Username for authenticating with workgroup server. Default: current user.
<code>-P<Password></code>	Password for authenticating with workgroup server. Default: current user.
<code>-R<Realm></code>	Security realm for authenticating with workgroup server. Default: security realm for current user
<code>-a</code>	Don't authenticate with workgroup server. Connect to workgroup server as a GUEST user.

B.13 IBM MQ Event Adapter

The IBM MQ event adapter is an application named mqevent (.exe).

B.13.1 Platforms

The IBM MQ event adapter is available on the following platforms:

Windows

B.13.2 Command-line Arguments

Table B-15 lists all command-line arguments applicable to the IBM MQ event adapter.

Table B-15. IBM MQ Event Adapter Command-Line Arguments	
Arguments	Description
Optional Arguments	
<code>-m<GroupName></code>	Specifies the name of the workgroup to connect to.
<code>-a</code>	Enables auditing of IBM MQ object change events. Default: disabled
<code>-r<Reconnect></code>	Interval, in sec, between attempts to connect to workgroup server. Default: 60
<code>-U<User></code>	Username for authenticating with workgroup server. Default: current user
<code>-P<Password></code>	Password for authenticating with workgroup server. Default: current user
<code>-R<Realm></code>	Security realm for authenticating with workgroup server. Default: security realm for current user
<code>-A</code>	Don't authenticate with workgroup server. Connect to workgroup server as a GUEST user.
<code>-install</code>	Installs Event Adapter as a service.
<code>-remove</code>	Removes Event Adapter as a service.
<code>-console</code>	Runs Event Adapter as a console application.

Appendix C: Queue Naming Conventions and Qualifiers

Qualifiers are used to qualify the model queue name used to create the Nastel Navigator queues. Qualifiers are dictated by a customer's queue naming conventions. Normally, for example, the MMF-related queues are created with names:

```
NASTEL.MMF.ADMIN.COMMAND.QUEUE
NASTEL.MMF.AUDIT.QUEUE
```

But, say at the ABC Carpet Company, they have IBM MQ installed with queue naming conventions: all queues for the nylon carpet division start with `ABC.NYLON` and for wool carpet division, `ABC.WOOL`. Headquarters' queues simply start with `ABC`. So, at headquarters they need a `config/mqqual.ini` file that will cause Nastel MMF queues to be created with the `ABC` prefix or qualifier:

```
NASTEL.MODEL.QUEUE=NASTEL.MODEL.QUEUE
NASTEL.HLQ=ABC
```

The MMF queues must be created as:

```
ABC.NASTEL.MMF.ADMIN.COMMAND.QUEUE
ABC.NASTEL.MMF.AUDIT.QUEUE
```

The `NASTEL.EVENT.QUEUE` would be created as:

```
ABC.NASTEL.EVENT.QUEUE
```

A temporary dynamic queue might look like this:

```
ABC.NASTEL.EVENT.NSQMGR4B02205D07390020
```

and similarly, for the other `NASTEL.*` queues and dynamic queues.

At the nylon division factory, the `config/mqqual.ini` file would look like this:

```
NASTEL.MODEL.QUEUE=NASTEL.MODEL.QUEUE
NASTEL.HLQ=ABC
NASTEL.MLQ=NYLON
```

and the MMF-related queues must be named:

```
ABC.NYLON.NASTEL.MMF.ADMIN.COMMAND.QUEUE
ABC.NYLON.NASTEL.MMF.AUDIT.QUEUE
```

The creation of the predefined, permanent `NASTEL` queues with the proper prefix is accomplished by editing the MQSC scripts `nastel.tst`, `pubsub.tst` and `mmf.tst` before the queues are created with the command.

```
runmqsc qmgr_name < <script_name>
```

Nastel Navigator Components Installation Guide Appendix C: Queue Naming Conventions and Qualifiers

The temporary, dynamic NASTEL queues will be created automatically from the data in the qualifier file. All `AP_WMQ nsq*` applications that access the NASTEL.xxx queues (nsqmq, nsqpub, nsqmsg, and utilities) must check if qualifiers for the queue names are in effect to create or access them.

Appendix D: Nastel Navigator IBM MQ Objects

The tables below list Nastel Navigator and IBM MQ objects that are created by these Nastel Navigator components: Workgroup servers, IBM MQ Agents, IBM MQ Event Publisher, MMF Server, z/OS PCF Command Server.

For each Nastel Navigator object, Table D-1 lists the component by which it is created (and when), the object's type, the platforms that utilize it, and a comment about which Nastel Navigator components access it. Table D-2 lists IBM MQ objects used by Nastel Navigator components.

Table D-1. Nastel Navigator Objects and Description				
Nastel Navigator Object	Platform	Created By/When	Attribute	Comments
NASTEL.MMF.AUDIT.QUEUE	All	MMF installation (mmf.tst MQSC script)	Permanent, local, persistent	For future use by MMF
NASTEL.MMF.ADMIN.COMMAND.QUEUE	All	MMF installation (mmf.tst MQSC script)	Permanent, local, non-persistent	MMF Server uses this queue to receive Message Explorer commands. Messages are received and put there by IBM MQ Agent
NASTEL.EVENT.DEFAULT_date_time_hash** For non- z/OS platforms, the date_time yyyyymmddhhmmsshh is attached as suffix for a dynamic queue. For z/OS, a hash number is attached.	All	Created by IBM MQ Agent before subscribing to Publisher. Occurs when workgroup server sends EXCMD_OPEN_EVENT command to IBM MQ Agent.	Temporarily dynamic, local, non-persistent	Publisher puts IBM MQ instrumentation events requested during subscription and DLQ events on the specified temporary
NASTEL.PUBSUB.EVENT.QUEUE	All	Created during Publisher installation (pubsub.tst MQSC script)	Permanent, local, non-persistent	Publisher uses this queue to: Get QMGR instrumentation event messages. Get Nastel Navigator command message (open/close subscription, shutdown publisher), Put DLQ event message
NASTEL.REPLY.ClientName_Date_Time_hash where ClientName is name of connecting client application. Example: NSQMGR for	All	Created by IBM MQ Agent when a workgroup server connects and sends command EXCMD_MQ_OPEN	Temporarily dynamic, local, non-persistent	MQ Agent thread uses this queue as a ReplyTo queue for responses from the IBM MQ PCF or MQSC command server

Table D-1. Nastel Navigator Objects and Description				
Nastel Navigator Object	Platform	Created By/When	Attribute	Comments
WGS or NSQCL for the nsqcl test program		qmgr_name to connect to the queue manager		
NASTEL.REPLY. NASTE.PCF.MQhash	z/OS	Created by MMF Server when it starts up	Temporarily dynamic, local, non-persistent	MMF Server uses this as ReplyTo queue to obtain responses for commands it submits to IBM MQ command server
NASTEL.REPLY.NASTEL. MSG.Redate_Time_hash	All	Created by MMF Server when it starts up	Temporarily dynamic, local, non-persistent	MMF Server uses this as ReplyTo queue to obtain responses for commands it submits to IBM MQ command server
NASTEL.ADMIN.EVENT. QUEUE	All	Created during Nastel Navigator installation (by mqmgm.tst MQSC script)	Permanent, local, non-persistent	Nastel Navigator gets IBM MQ instrumentation event messages from this queue when Publisher is not configured.

Table D-2. IBM MQ Objects Used by Nastel Navigator Components				
Nastel Navigator Object	Platform	Created By/When	Attribute	Comments
SYSTEM.ADMIN.COMMAND. QUEUE	Non-z/OS	Created during QMGR creation	Permanent, local, non-persistent	MQ Agent puts PCF commands on this queue for the IBM MQ PCF Server
	z/OS	Created during QMGR creation	Permanent, local, non-persistent	z/OS PCF Command Server gets commands from this queue, converts them to MQSC commands and submits them to MQSC command server.
SYSTEM.COMMAND.INPUT	z/OS	Created during QMGR creation	Permanent, local, non-persistent	z/OS PCF Command Server puts converted PCF commands onto this queue. IBM MQ Command Server gets commands from this queue.
SYSTEM.ADMIN.CHANNEL.	All	Created during QMGR	Permanent, local,	Nastel Navigator installation script mqmgm.tst converts

<p>EVENT SYSTEM.ADMIN.PERFM. EVENT SYSTEM.ADMIN.QMGR. EVENT SYSTEM.ADMIN.COMMAND. EVENT</p>		<p>creation as local queues. Modified during Nastel Navigator and publisher installation (MQSC scripts mqmgm.tst, pubsub.tst)</p>	<p>non-persistent</p>	<p>from the local queue to the alias queue with the target queue set to NASTEL.EVENT.QUEUE. Publisher installation script pubsub.tst redefines the alias queue with the target queue set to NASTEL.PUBSUB.EVENT.QUEUE</p>
<p>SYSTEM.DEAD.LETTER.QUEUE</p>	<p>All</p>	<p>Created during QMGR creation</p>	<p>Temporary dynamic, local, non-persistent</p>	<p>Publisher monitors this queue and creates an event when appropriate.</p>

Glossary

This appendix contains a list of reference material and documents relevant to Nastel Navigator and other related Nastel products.

AutoPilot M6: Nastel Technologies' Enterprise Application Management Platform. AutoPilot M6 monitors and automates the management of eBusiness integration components such as middleware application, application servers and user applications.

AP-WMQ: Nastel Navigator; Nastel Technologies' IBM MQ management solution. Abbreviated as AP-WMQ.

AutoPilot: AutoPilot Server enables AutoPilot M6 to monitor and manage eBusiness applications for continuous operations in addition to its standard features.

AutoPilot/WebSphere Message Queue Integrator (AP-WMQI): Formerly AP/MQSI.

BSV: *see* Business Views.

Business View (BSV): A collection of rules that define a desired state of an eBusiness environment. Business Views can be tailored to presents information in the form most suited to a given user, as defined by the user.

Client: Any programming component that uses the AutoPilot M6 infrastructure; for example, the AutoPilot M6 Console.

Common Object Request Broker Architecture (CORBA): A Common Object Request Broker Architecture (CORBA) object can be invoked from a Web browser using CGI scripts or applets.

Console: The console acts as the graphical interface for AutoPilot M6.

Contacts: A subordinate to a given Manager or Expert.

CORBA: *see* Common Object Request Broker Architecture.

Data Source Name: A Data Source Name (DSN) is the logical name that is used by Open Database Connectivity (ODBC) to refer to the drive and other information that is required to access data. The name is use by Internet Information Services (IIS) for a connection to an ODBC data source, (Example: Microsoft SQL Server database). The ODBC tool in Control Panel is used to set the DSN. When ODBC DSN entries are used to store the connection string values externally, you simplify the information that is needed in the connection string. This makes changes to the data source completely transparent to the code itself.

Dependent IBM MQ Node: IBM MQ nodes that are not directly managed by Nastel Navigator. Because dependent nodes do not run an IBM MQ Agent, they must be managed by proxy.

Deploy: To put to use, to position for use or action.

Domain Server: A specialized managed node that maintains the directory of managed nodes, experts etc. The domain server is also capable of hosting experts, managers etc.

DSN: *see* Data Source Name

EVT: Event Log file extension (e.g.: *sample.evt*).

Event: An *Event* is something that happens to an object. Events are logged by AutoPilot M6 and are available for use by AutoPilot M6 Policies or the user.

Expert: Services that monitor specific applications such as an applications server, webserver, or specific components within the applications. (Example, channels in IBM MQ.) Experts generate facts.

Fact: Facts are single pieces of data that has a unique name and value. One or more facts are used to determine the health of the object, application, or server.

Graphical User Interface (GUI): A type of environment that represents programs, files, and options by means of icons, menus, and dialog boxes on the screen. The user can select and activate these options by pointing and clicking with a mouse or, often, with the keyboard. Because the graphical user interface provides standard software routines to handle these elements and report the user's actions (such as a mouse click on a particular icon or at a particular location in text, or a key press); applications call these routines with specific parameters rather than attempting to reproduce them from scratch.

GUI: see Graphical User Interface.

IBM MQ: IBM's message queuing product. Formerly known as MQSeries.

IBM MQ Commands: A command-line language used to configure IBM MQ.

Independent IBM MQ Node: An IBM MQ node that runs an IBM MQ Agent and which is managed directly by a workgroup server. Independent nodes can be used as proxy nodes for managing dependent nodes.

IIS: See Internet Information Services.

Internet Information Services: Microsoft's brand of Web server software, utilizing HTTP to deliver World Wide Web documents. It incorporates various functions for security, allows CGI programs and provides for Gopher and FTP services.

JCL: See Job Control Language

Java: A platform-independent, object-oriented programming language developed and made available by Sun Microsystems.

Java Developer's Kit (JDK): A set of software tools developed by Sun Microsystems, Inc., for writing Java applets or applications. The kit, which is distributed free, includes a Java compiler, interpreter, debugger, viewer for applets, and documentation.

JDBC: See Java Database Connectivity.

Java Database Connectivity (JDBC): The JDBC API provides universal data access from the Java programming language. Using the JDBC 2.0 API, you can access virtually any data source, from relational databases to spreadsheets and flat files. JDBC technology also provides a common base on which tools and alternate interfaces can be built. The *JDBC Test Tool* that was developed by Merant and Sun Microsystems may be used to test drivers, to demonstrate executing queries and getting results, and to teach programmers about the JDBC API.

Java Server Pages (JSP): JSP technology enables rapid development of web-based applications that are platform independent. Java Server Pages technology separates the user interface from content generation enabling designers to change the overall page layout without altering the underlying dynamic content. Java Server Pages technology is an extension of the Java™ Servlet technology.

Java Virtual Machine (JVM): The “virtual” operating system that JAVA-written programs run. The JVM is a hardware- and operating system-independent abstract computing machine and execution environment. Java programs execute in the JVM where they are protected from malicious programs and have a small compiled footprint.

JDBC: See Java Database Connectivity

JDK: See Java Developer's Kit.

Job Control Language (JCL): Most commonly used in larger computer systems, JCL is any control language that controls the execution of applications. The syntax is usually strict; not permitting the addition or deletion or spaces or characters where they are not expected.

JRE: JAVA Run-time Environment. The minimum core JAVA required to run JAVA Programs.

JSP: See Java Server Pages

JVM: see JAVA Virtual Machine.

M6 Web: A browser-based interface that provides monitoring and operational control over managed resources and applications.

Managed Node: A container that can host any number of AutoPilot M6 services such as experts, managers, policies etc. Unlike managed nodes, it is a physical process.

Management Information Base (MIB): A specification that describes the properties and behavior of a network device. Network managers use MIBs to interact with SNMP-compatible devices. Each MIB is part of a directory structure that specifies where objects are found on the network.

Manager: Managers are the home or container for policies. All business views must reside on managers, and manager must be deployed prior to deploying a business view or policy.

Message Management Facility (MMF): Nastel's message management service.

Message Queue Interface: The Message Queue Interface (MQI) is part of IBM's Networking Blueprint. It is a method of program-to-program communication suitable for connecting independent and potentially non-concurrent distributed applications.

MIB: see Management Information Base

MMF: see Message Management Facility

MOM: see Message-Oriented Middleware.

MQI: see Message Queue Interface

MQSC: See IBM MQ Commands

MQSeries: IBM's message queuing product. Renamed by IBM as IBM MQ.

Naming Service: A common server records “names” of objects and associates them with references, locations, and properties.

Nastel Navigator: Nastel Technologies’ middleware management solution. Re-designated as M6 for WMQ with release 6.0, prior releases retain the AP-WMQ or MQControl trademark. Renamed to Nastel Navigator.

ORB: Object Request Broker.

Orbix: CORBA product distributed by IONA Technologies.

Package Manager: The command line utility that allows users to list, install, uninstall, verify, and update AutoPilot M6 installation on any Managed Node.

PCF: See Programmable Command Format.

PKGMAN: See Package Manager.

Policy/Business Views: Business views are a collection of one or more sensors. Business views are used to visually present the health and status of the different systems as well as automatically issue remedial actions.

Programmable Command Format (PCF): A set of programmable commands that Nastel Navigator uses to manage IBM MQ. PCF includes data definitions for items such as integers, strings, and lists. The commands can be submitted directly to a queue manager. PCF is comparable to MQSC, except for the fact that MQSC cannot be programmed.

Proxy Management: The indirect management of MQ objects by an intermediate entity. For example, a proxy queue manager might be used to handle another queue manager.

Sensor: A rule that is used to determine the health of an object or application based on one or more facts. Actions can then be issued, based on the health. Sensors are definable in business views by use of the sensor wizard.

Simple Mail Transfer Protocol (SMTP): A TCP/IP protocol for sending messages from one computer to another on a network. This protocol is used on the Internet to route e-mail. *See also* communications protocol, TCP/IP. *Compare* CCITT X series, Post Office Protocol.

Simple Network Management Protocol (SNMP): A de facto standard for managing hardware and software devices on a network. Each device is associated with a Management Information Base (MIB) that describes its properties and behavior.

SMTP: See Simple Mail Transfer Protocol.

SNMP: See Simple Network Management Protocol.

SNMP Master Agent: An implementation of the SNMP protocol. It includes a definition of the standard MIB. The master agent routes SNMP requests from subagent to subagent.

SNMP Subagent: The implementation of an MIB for a particular device. The MIB describes the device’s desired behavior; the SNMP subagent carries it out.

TCP/IP: See Transmission Control Protocol/Internet Protocol.

Transmission Control Protocol/Internet Protocol (TCP/IP): A protocol developed by the Department of Defense for communications between computers. It is built into the UNIX

system and has become the de facto standard for data transmission over networks, including the Internet.

Virtual Machine: Software that mimics the performance of a hardware device, such as a program that allows applications written for an Intel processor to be run on a Motorola chip. *Also See* Java Virtual Machine.

Websphere_MQ_Manager: A specialized manager capable of hosting one or more IBM MQ specific policies, apart from the regular policies.

Wireless Application Protocol (WAP): An open global specification that is used by most mobile telephone manufacturers. WAP determines how wireless devices utilize Internet content and other services. WAP enables devices to link diverse systems contents and controls.

z/OS: See Z Series Operating System.

Z Series Operating System: IBM architecture for mainframe computers and peripherals. The zSeries family of servers uses the z/Architecture. It is the successor to the S/390 and 9672 family of servers.

Index

<p>B</p> <p>blocksizes57</p> <p>books22</p> <p>C</p> <p>clients 16, 22</p> <p>Command-line Arguments</p> <p> database creation utility 274</p> <p> Log Adapter 262</p> <p> Message Get/Put Utility..... 266</p> <p> Messagement Management Utility 264</p> <p> MMF Server..... 254</p> <p> MQEvent Publisher 256</p> <p> MQSC Utility..... 259</p> <p> Ping Utility..... 260</p> <p> Queue Monitor Utility..... 265</p> <p> Remote Shutdown utility..... 274</p> <p> Trigger Message Utility 275</p> <p> WMQ Event Adapter 276</p> <p>communication hardware..... 17</p> <p>componenets on z/OS..... 53</p> <p>Connection Manager</p> <p> Platforms 129</p> <p>connection manager port 24</p> <p>conventions</p> <p> PARM Field, EXEC statement 68, 72, 74, 77</p> <p>D</p> <p>database creation utility Command-line Arguments</p> <p> 274</p> <p>Dataset Allocations 56</p> <p>documents 11</p> <p>domain..... 22</p> <p>E</p> <p>EMS 159</p> <p>environment variables</p> <p> on OS/400..... 86</p> <p>F</p> <p>features, M6 for WMQ..... 22</p> <p>feedback..... 11</p> <p>H</p> <p>hardware requirements..... 17</p>	<p>I</p> <p>IBM MQ</p> <p> components..... 251</p> <p> installation first time..... 82</p> <p> installation upgrade..... 86</p> <p> Objects 279</p> <p> remove from z/OS..... 79</p> <p>IBM MQ agent</p> <p> installation consideration..... 41</p> <p> MQSINI file..... 68</p> <p> nsqmq.exe..... 251</p> <p> platform availability 251</p> <p> port 24</p> <p> starting on UNIX..... 44</p> <p> starting on z/OS 67</p> <p> stopping multiple components 77</p> <p> stopping on UNIX..... 46</p> <p> stopping on z/OS 69</p> <p>IBM MQ command server 71, 73, 88</p> <p>IIB/ACE Connection Manager 175, 182</p> <p>Install agents as Windows services 24</p> <p>installation materials..... 277</p> <p>intended audience..... 11</p> <p>ITO 22, 23</p> <p>ITO agent..... 120</p> <p>L</p> <p>limitations</p> <p> MMF Server 54</p> <p> z/OS..... 53</p> <p>LMSTOP 70</p> <p> run as Started Task 70</p> <p>load data sets,z/OS 54</p> <p>Log Adapter</p> <p> about 120</p> <p> Command-line Arguments..... 262</p> <p> nsqadp.exe..... 262</p> <p> platforms..... 262</p> <p> starting 122</p> <p> stopping..... 125</p> <p> troubleshooting 127</p> <p>log file encapsulator..... 120</p> <p>logon..... 29, 33</p> <p>M</p> <p>M6 for WMQ explorer</p> <p> starting in Windows..... 28</p>
---	--

M6WMQ.LOAD54
 material, installation277
 message explorer
 starting on Windows 32
 message file utility (nsqmsgf) 94
 Message Get/Put Utility
 Command-line Arguments 266
 nsqmsgf.exe265
 platforms265
 message groups 120
 Message Management
 platforms264
 Message Management Facilities254
 Message Management Utility
 Command-line Arguments 264
 nsqmmfcl.exe264
 message server
 starting on UNIX 48
 messages
 format of 124
 migration on z/OS 54
 Minimum Dataset Allocations 56
 minimum operating standards 17
 MMF Server 64
 endmcs.exe 254
 platforms 254
 restarting 72
 stopping 72
 MMF Server Command-line Arguments 254
 MQ Event Publisher
 MQSINI configuration file 75
 on z/OS65
 starting on z/OS73
 SUBSCRIB file on z/OS75
 MQCSTOP
 run as started task 78
 MQEvent Publisher
 nsqpub.exe 256
 platforms256
 MQEvent Publisher Command-line Arguments ... 256
 MQGROUP 69
 mqgroup.ini 69
 MQSC Utility
 nsqmqsc.exe 259
 platforms259
 MQSC Utility Command-line Arguments 259
 MQSINI file (MVS) 75
 MSSTOP
 run as started task 73

N

Nastel Navigator
 agent 22
 components on z/OS 53

features 22
 limitations on z/OS53
 load modules z/OS 54
 objects 279
 z/OS configuration59
 z/OS installation 53
 nsqadp 262
 nsqcl 93
 nsqcmkafka.properties 167, 168
 nsqcode utility 261
 nsqcode Utility
 platforms261
 nsqmigr 239
 nsqmmfcl 264
 nsqmon 265
 nsqmq 251
 NSQMQ
 run as started task68
 nsqmsfg 94, 101
 NSQMSG 71
 run as started task71
 nsqmsgf 265
 nsqping 93
 nsqprid 237
 NSQPUB
 run as started task74
 nsqpub.exe 256
 nsqrc.ini file 124
 nsqsev.ini file 120, 124, 127
 nsqstop.exe 273
 nsqtrig.exe 275

O

OS/400
 environment variable for86
 system requirements for81

P

PARM field68, 72, 74
 Conventions 70
 PARM File75
 PBSTOP 77
 PCF Command Server 64, 73
 PERMITS file
 on z/OS69
 Ping Utility 260
 platforms260
 ping utility (nsqping) 93
 Ping Utility Command-line Arguments260
 Platforms
 Connection Manager 129
 Log Adapter 262
 Message Get/Put Utility 265

Message Management 264
 MMF Server..... 254
 MQ Agent..... 251
 MQEvent Publisher 256
 MQSC Utility..... 259
 nsqcode Utility..... 261
 Ping Utility..... 260
 Queue Monitor Utilities 265
 Remote Shutdown Utility..... 273
 Trigger Message Utility 275
 WMQ Event Adapter 276
 publish subscriber
 starting on UNIX 50
 stopping on UNIX 51

Q

Queue Monitor
 nsqmon.exe 265
 Queue Monitor Utility
 Command-line Arguments 265
 platforms 265
 Queue Naming Conventions 278
 Queue Qualifiers 278
 queues,AP/MQ Agent..... 42, 85
 queues,AP/WMQ Message Server 43, 86
queues,AP/WMQ Publish-Subscriber 43, 86

R

REGION 70, 71, 72, 77
 Remote Shutdown utility
 platforms 273
 Remote Shutdown Utility
 Command-line Arguments 274
 Remote Shutdown utility, nsqstop.exe 273
 requirements, minimum 17
 restore data set, z/OS 56
 Restore Data Sets..... 59
 Running multiple services
 connection manager 28
 event adapter 27
 local agent 27
 log adapter 27

S

SDK..... 22
 Server
 MMF 254
 MMF on z/OS 64
 MMF, restarting 72
 MMF, starting on z/OS 71
 severity file
 on z/OS 76

SNMP 22
Solaris
 system requirements for 41
 spool file 101
 starting M6 for WMQ Workgroup Statistics 37
 starting, stopping compoments with JCL..... 66
 SUBSCRIB file (z/OS)..... 75
 SYS1.PROCLIB..... 66, 73
 system error..... 261
 system requirements
 for OS/400 81
 for VMS 104

T

TCP/IP 17
 templates
 log file 120
 message 120
 Tivoli 22, 23
 Transfer Data Sets 57
 Trigger Message Utility
 Command-line Arguments..... 275
 nsqtrig.exe 275
 platforms..... 275
 troubleshooting MQControl
 on VMS..... 117

U

uninstall M6 for WMQ from z/OS 79
 upgrade M6 for WMQ on z/OS..... 54
user authentication 29, 32
 utilities 22, 23
 nsqcl 93
 nsqmsgf 94, 101
 nsqping 93
 utility
 Queue Monitor 265
 Utility
 Message Get/Put Utility 265
 Message Management..... 264
 MQSC 259
 Remote Shutdown..... 273
 Trigger Message 275

V

VMS
 system requirements for..... 104
 troubleshooting MQControl on..... 117
 upgrading Nastel Navigator on..... 104

W			
WMQ Event Adapter		M6 for WMQ..... 64	
Command-line Arguments	276	MQ Event Publisher..... 65	
platforms	276	MQGROUP file..... 69	
workgroup statistics		MQSINI configuration file..... 75	
starting on Windows	37	P (STOP), M6 for MQ agent..... 69	
		P (STOP), MMF Server..... 72	
		requirements	53
		SEVERITY file.....	76
		starting AP/MQ agents.....	66
		starting MQ Event Publisher.....	73
		starting the MMF Server.....	71
		SUBSCRIB file.....	75
		uninstalling M6 for WMQ.....	79
Z			
z/OS			
configuration requirements.....	59		
installations.....	54		