

SupportPac WA61 – WebSphere MQ Workflow – Enabling WebSphere Business Monitor Version 1.0

December, 2007

Flow Technology Team Böblingen

Property of IBM

Take Note!

Before using this documentation be sure to read the general information under "Notices".

First Edition, December 2007

This edition applies to Version 1.0 of *SupportPac WA61 - WebSphere MQ Workflow – Enabling WebSphere Business Monitor* and to all subsequent releases and modifications unless otherwise indicated in new editions.

© **Copyright International Business Machines Corporation 2007**. All rights reserved. Note to US Government Users -- Documentation related to restricted rights -- Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule contract with IBM Corp.

Table of Contents

Table of Contents.....	iii
Notices	5
Trademarks and service marks	5
Summary of Amendments.....	6
Preface.....	7
Bibliography	8
Chapter 1. Introduction.....	1
Chapter 2. Installation and Customization	2
Prerequisites.....	2
Installation	2
Customization.....	3
WebSphere MQ Workflow	3
1. FDL settings	3
2. Profile variables	4
3. Fmcibie	5
4. WebSphere MQ.....	6
WebSphere Application Server.....	9
1. WebSphere MQ.....	11
2. Common Event Infrastructure	17
3. Installing the application	19
Configured objects.....	19
Queue connection factory	19
JMS queue.....	20
Listener Port.....	20
Chapter 3. Event Description	22
Start and Stop events.....	22
Converted events	24
Variable substitution:.....	30

Sample CBE:.....	31
Chapter 4. Uninstalling.....	34
Chapter 5. Problem Determination	36
Traces.....	36
Version String	36
Flight Recorder	36
Chapter 6. Performance Considerations.....	38
Filtering of WebSphere MQ Workflow audit events	38
Number of connections for the MDB.....	40
Number of sessions for the defined Listener port of the MDB.....	41
Size of the thread pool for the defined Listener port of the MDB	41
Number of connections for the CEI Event Database Store	41
CEI Event Datastore.....	41
Chapter 7. Appendix	Error! Bookmark not defined.

Notices

The following paragraph does not apply in any country where such provisions are inconsistent with local law.

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any reference to an IBM licensed program or other IBM product in this publication is not intended to state or imply that only IBM's program or other product may be used. Any functionally equivalent program that does not infringe any of the intellectual property rights may be used instead of the IBM product.

Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, New York 10594, USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS-IS. The use of the information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item has been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Trademarks and service marks

The following terms, used in this publication, are trademarks of the IBM Corporation in the United States or other countries or both:

- IBM
- WebSphere
- SupportPac
- zOS

The following terms are trademarks of other companies:

- Java
- Windows

Summary of Amendments

Date	Changes
December 2007	First Edition

Preface

This SupportPac converts WebSphere MQ Workflow audit events into Common Base Events and sends them to the Common Event Infrastructure. It provides a J2EE application that receives WebSphere MQ Workflow audit events coming from the WebSphere MQ Workflow audit queue and converts the them into Common Base Events. The Common Base Events are then transmitted to the Common Event Infrastructure (CEI) that is part of WebSphere Application Server 6.1. These Common Base Events are in a format suitable for WebSphere Business Monitor 6.1.

Bibliography

WebSphere MQ Workflow V3.6 Concepts & Architecture (GH12-6285-07)

- Chapter 4. Running your business processes - Monitoring, locating, and analyzing processes

WebSphere MQ Workflow V3.6 Administration Guide (SH12-6289-08)

- Chapter 8. How to use the audit trail

WebSphere MQ Workflow V3.6 Programming Guide (SH12-6291-10)

- Chapter 10. Global containers
- Chapter 31. The MQ Workflow XML message format
- Chapter 70. Customizing the Web client - Sample CommandHandlers - AuditTrailHandler

WebSphere MQ Workflow V3.6 Getting Started with Buildtime (SH12-6286-10)

- Chapter 5. Advanced modeling techniques
- Chapter 7. Defining workflow information in an FDL file
- Chapter 8. FDL definitions

WebSphere MQ Workflow V3.6 Installation Guide (SH12-6288-11)

WebSphere MQ Workflow for z/OS V3R6 Customization and Administration (SC33-7030-10)

- Chapter 2. Planning your configuration
- Appendix M. Audit Trail

WebSphere MQ Workflow for z/OS V3R6 Programming Guide (SC33-7031-09)

IBM WebSphere MQ Workflow Buildtime online help

IBM WebSphere MQ Workflow FmcXML-Client.xsd

See `<xsd:element name="AuditTrailRecord">`

The XSDs are located in the MQ Workflow subdirectory "bin" on the Windows platform

WebSphere MQ Workflow TecDocs

- Enhanced audit trail capabilities to log container data

WebSphere MQ Workflow Technotes

- How can I enable the WebSphere® MQ Workflow audit trail in a WebSphere MQ Workflow runtime environment, when my Buildtime is not aware of current runtime domain settings?
- How can I use the audit trail for process analysis
- Is it possible for WebSphere® MQ Workflow to send an e-mail when a specific event has occurred?
- Global Data Container Monitoring

- AUDIT_TO_MQ and audit trail feature

WebSphere MQ Using Java Version 6.0 (SC34-6591-00)

- Chapter 2. Installation and configuration

Chapter 1. Introduction

The WebSphere MQ Workflow Event Converter shipped in WebSphere MQ Workflow SupportPac WA61, is a standard J2EE application that is deployed on a WebSphere Application Server V6.1

The Event Converter application transforms the data of entries from a WebSphere MQ Workflow audit trail destination MQ into Common Base Events (CBEs).

One Event Converter application can process only audit trail events from one WebSphere MQ Workflow system group.

The Event Converter works as a standard Message Driven Bean (MDB) if WebSphere MQ Workflow sends the audit information across WebSphere MQ. The application listens on the queue the WebSphere MQ Workflow system group sends its audit trail data to, picks up the incoming messages and transforms the information in those messages into CBEs. The CBEs are sent out on the CEI bus to applications like WebSphere Business Monitor. Each message is processed in one transaction.

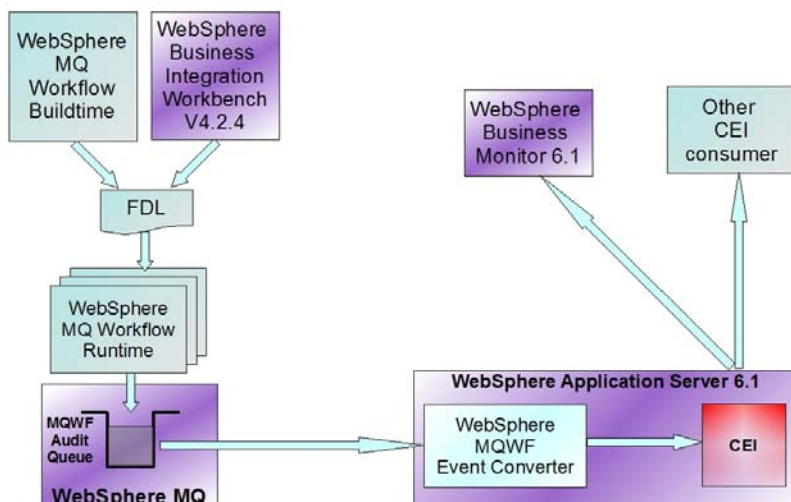
Once the Event Converter processed a message successfully, it will be gone from the WebSphere MQ Workflow audit trail queue. If the audit messages are supposed to be read by applications other than the Event Converter, the audit queue must be set up in a publish-subscribe mode. Such a setup is described in the WebSphere MQ Workflow documentation.

The data might be processed in parallel (e.g. by multiple instances of the MDB) to keep up with the workload.

The sequence of audit events can still be determined correctly by monitoring applications like WebSphere Business Monitor based on the timestamps written by the workflow engines. This is ensured by using the global clock feature of WebSphere MQ Workflow.

The WebSphere MQ Workflow Event Converter itself emits a start event to the CEI when the application is started and a stop event when the application is stopped.

The following figure shows an overview of the system design, which illustrates the main components and the interactions between these components.



Chapter 2. Installation and Customization

Prerequisites

- WebSphere MQ 6.0 with Java Messaging (JMS), or higher. You must perform all the JMS installation and post installation steps described in the WebSphere MQ manual "Using Java" in Chapter 2. "Installation and configuration"
- WebSphere MQ Workflow 3.6 with ServicePack 6.
- WebSphere Application Server 6.1

Installation

1. Download the SupportPac WA61 to a directory.
2. Extract the downloaded file to a directory with read/write access for the userID of the WebSphere Application Server and ensure you specify the option to re-create stored directories. The following folder structure will be created:

```

WA61.zip
|
+-- doc
|   |
|   +-- SupportPac WA61 - Readme.txt
|   +-- SupportPac_WA61_User_Guide.pdf
|   +-- WMQWF_Events.xsd
|
+-- install
|   |
|   +-- converter.py
|   +-- fmca2cbe.ear
|   +-- messageCatalog.p
|   +-- mqwflib.py
|
+-- licences
|   |
|   +-- Lic_cs.txt
|   +-- Lic_de.txt
|   +-- Lic_en.txt
|   +-- Lic_es.txt
|   +-- Lic_fr.txt
|   +-- Lic_it.txt
|   +-- Lic_ja.txt
|   +-- Lic_ko.txt
|   +-- Lic_pl.txt
|   +-- Lic_pt.txt
|   +-- Lic_tr.txt
|   +-- Lic_zh.txt
|   +-- Lic_zh_TW.txt
|   +-- Lic_zh_TW_Big5.txt

```

3. You can delete the zipped file.
4. Complete the customization of the SupportPac as described in the following chapters.

Customization

WebSphere MQ Workflow

1. FDL settings

Add the following settings to the FDL of your WebSphere MQ Workflow installation:

- Enable audit to MQ
- Define the queue name and the queue manager name for your WebSphere MQ Workflow system group as the target for the WebSphere MQ Workflow audit messages.

For further and more detailed information, refer to *IBM WebSphere MQ Workflow Getting Started with Buildtime V. 3.6*, Chapter 7: “Defining workflow information in an FDL file.”

Samples:

Setting FULL_AUDIT_TO_MQ on the workflow domain level

```
CODEPAGE 437
FM_RELEASE V3R6
// -----
// Domain
// -----
UPDATE DOMAIN 'DOMAIN'
  PROCESS
    FULL_AUDIT_TO_MQ
END
```

For performance reasons, filter the audit data.

How to enable filtering on the workflow domain level and how to set the audit filter to “process started” (21000) and “process ended normally” (21004) is shown in the following figure:

```

CODEPAGE 437
FM_RELEASE V3R6
// -----
// Domain
// -----
UPDATE DOMAIN 'DOMAIN'
  PROCESS
    FILTER AUDIT_TO_MQ
    AUDIT_FILTER_MQ 21000, 21004
END

```

Define a queue name and a queue-manager name for the workflow system group FMCGRP.

You must define these MQ resources with WebSphere MQ applications, such as the MQ Explorer or the MQ Command Interpreter because they are not defined as part of the WebSphere MQ Workflow customization or the customization of this SupportPac. This SupportPac needs access to the resources at runtime. See subchapter 4, “*WebSphere MQ*” for more information.

```

UPDATE GROUP 'FMCGRP'
  DESCRIPTION "Default System Group"
  //-----
  AUDIT_QUEUE_NAME 'FMC.FMCGRP.AUDITQ'
  AUDIT_QUEUE_MANAGER_NAME 'FMCQM'
  AUDIT_TOPIC_STRUCTURE '%Event%'
  //-----
END 'FMCGRP'

```

2. Profile variables

You need to set certain profile variables for various reasons:

- To write input or output container data in events.

Note: The GlobalDataContainer is emitted by event 21082 only, which is not affected by this setting.

```

Configuration/<WMQWF
configurationID>/EnableContainerDataInAuditTrail=TRUE

```

Set this variable by calling the WebSphere MQ Workflow configuration checker utility:

```

fmczchk -c inst:m,EnableContainerDataInAuditTrail,TRUE -y<WMQWF
configurationID>

```

- To guarantee the uniqueness of timestamps in workflow events:

```
Configuration/<WMQWF configurationID>/RTGlobalClock=TRUE
```

Set this variable by calling the WebSphere MQ Workflow configuration checker utility:

```
fmczchk -c inst:m,RTGlobalClock,TRUE -y<WMQWF configurationID>
```

- To correlate subprocesses with their calling activities:

```
Configuration/<WMQWF
configurationID>/EnableProcessActivityCorrelation=TRUE
```

Set this variable by calling the WebSphere MQ Workflow configuration checker utility

```
fmczchk -c inst:m,EnableProcessActivityCorrelation,TRUE -y<WMQWF
configurationID>
```

After setting these profile variables, restart the WebSphere MQ Workflow execution servers.

3. Fmcibie

To import your FDL process definitions into your workflow runtime, use the `-x` option of the `fmcibie` command to ensure that all names are XML-compliant. Non-compliant names lead to exceptions when parsing the generated CBEs.

Sample:

```
fmcibie -uMyUserID
        -pMyPassword
        -i "c:\MyProcessDefinitions.fdl"
        -t
        -x
        -yMyConfiguration
```

4. WebSphere MQ

The queue manager and the queue referred to in the previous section have to be accessible from the WebSphere Application Server on which this SupportPac application will be installed. This chapter describes the steps that you have to perform in order to ensure this.

If not already existing you can easily create the queue manager and the queue along with the connectivity to these MQ resources with WebSphere MQ applications, such as the WebSphere MQ Explorer or the WebSphere MQ command-line utilities.

Sample command to create a WebSphere MQ queue manager:

```
C:\>crtmqm FMCQM
```

Sample command to start the newly created WebSphere MQ queue manager:

```
C:\>strmqm FMCQM
```

Sample command to invoke the WebSphere MQ command interpreter for this newly created queue manager and to define the queue:

```
C:\>runmqsc FMCQM
5724-H72 (C) Copyright IBM Corp. 1994, 2004
Starting MQSC for queue manager FMCQM.

define qlocal(FMC.FMCGRP.AUDITQ)
  1 : define qlocal(FMC.FMCGRP.AUDITQ)
AMQ8006: WebSphere MQ queue created.
```

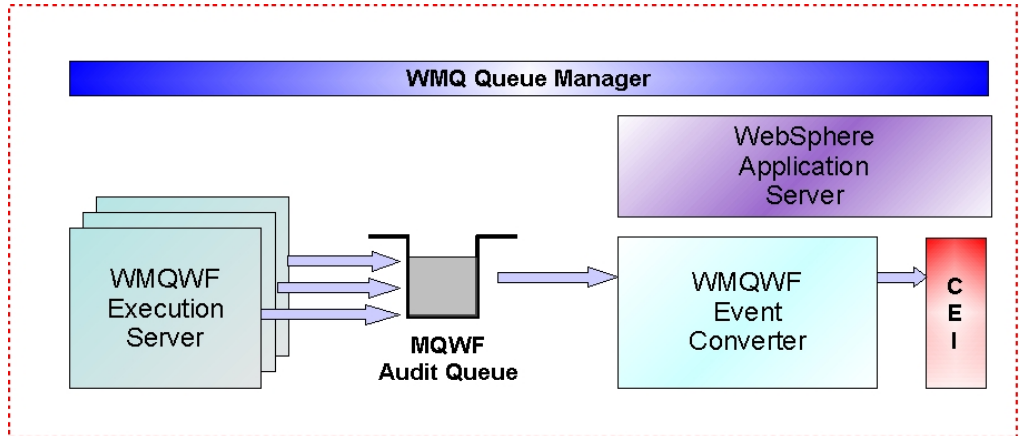
To get access to the queue manager and the queue, the Event Converter application within the WebSphere Application Server requires access to the defined queue manager and the queue. To grant access, you can use the WebSphere MQ Object Authority Manager (OAM).

Sample commands to give the userID `admin` the required authority:

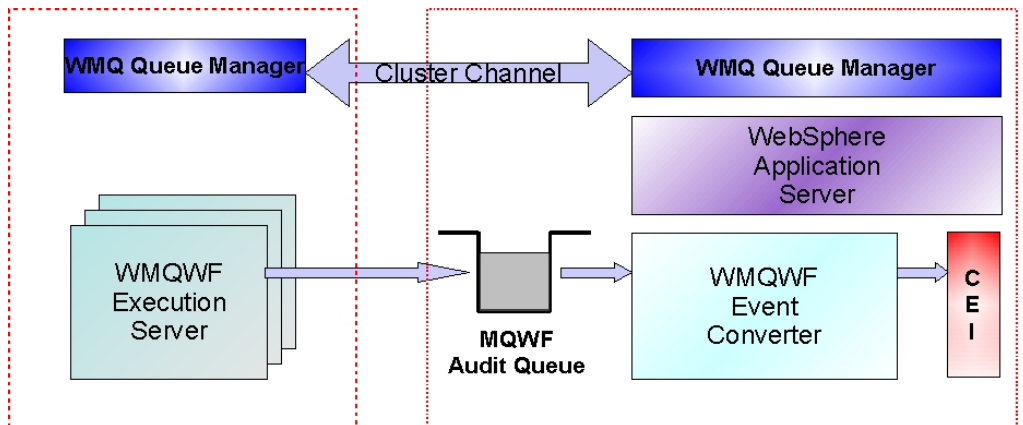
```
setmqaut -m FMCQM -t qmgr -p admin +allmqi
setmqaut -m FMCQM -t queue -n FMC.FMCGRP.AUDITQ -p admin +allmqi
```

WebSphere MQ Workflow, the Event Converter and the WebSphere Application Server can be located on different computers. The following figures give a brief overview of the various options.

- **Single-box setup**
WebSphere MQ Workflow, WebSphere Application Server and WebSphere MQ Workflow EventConverter are installed on the same computer and use the same queue manager.

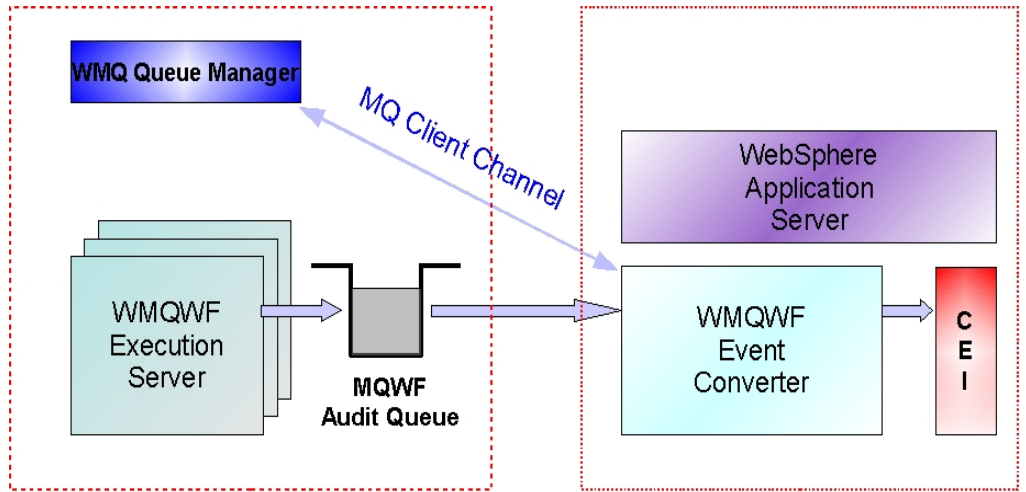


- **Multiple-box setup with different queue managers:**
WebSphere MQ Workflow resides on a computer other than the one on which WebSphere Application Server and WebSphere MQ Workflow EventConverter are located. Separate queue managers are used. The WebSphere MQ Workflow audit trail queue is located on the same computer as WebSphere Application Server. For this option, you need to connect WebSphere MQ Workflow to the WebSphere MQ Workflow audit trail queue. This can be done, for example, by use of MQ clustering. The WebSphere MQ Workflow Event Converter accesses the WebSphere MQ Workflow audit trail queue by means of local binding.



- **Multiple-box setup with a shared queue manager:**
WebSphere MQ Workflow resides on a computer other than the one on which WebSphere Application Server and WebSphere MQ Workflow EventConverter are located, but all of these components connect to the same queue manager. The queue manager with the WebSphere MQ Workflow audit trail queue is located on the same computer as WebSphere MQ Workflow. WebSphere MQ Workflow and the WebSphere MQ Workflow audit

trail queue are connected by local binding. The WebSphere MQ Workflow Event Converter accesses the WebSphere MQ Workflow audit trail queue through the WMQ Client channel.

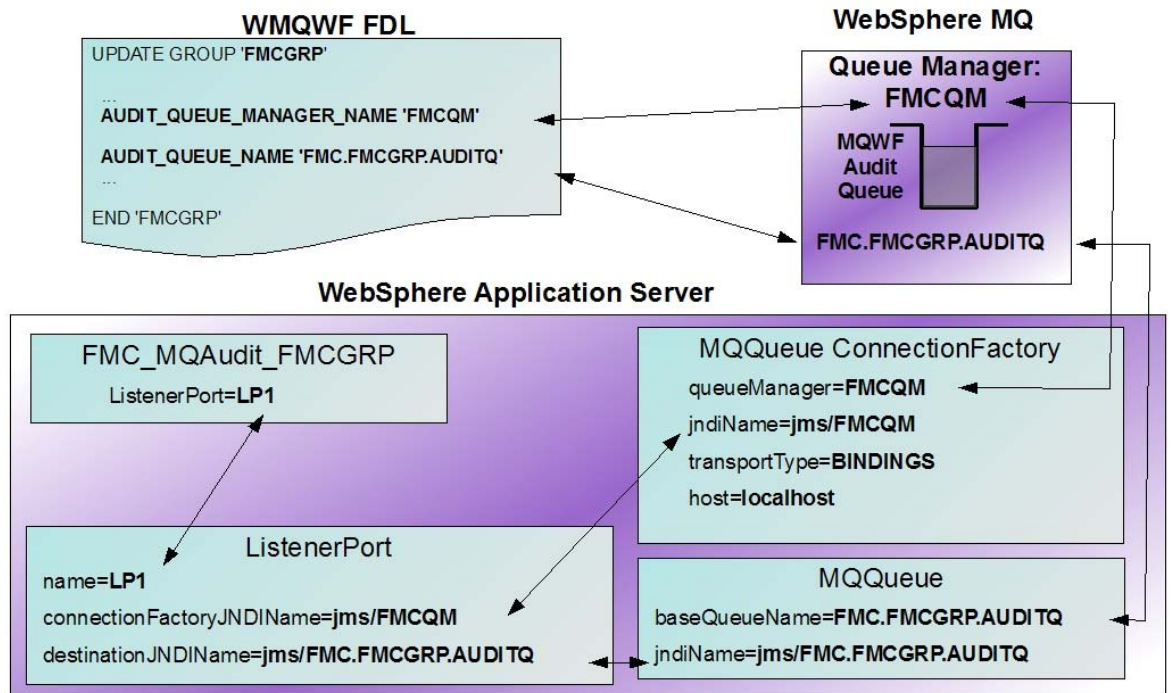


WebSphere Application Server

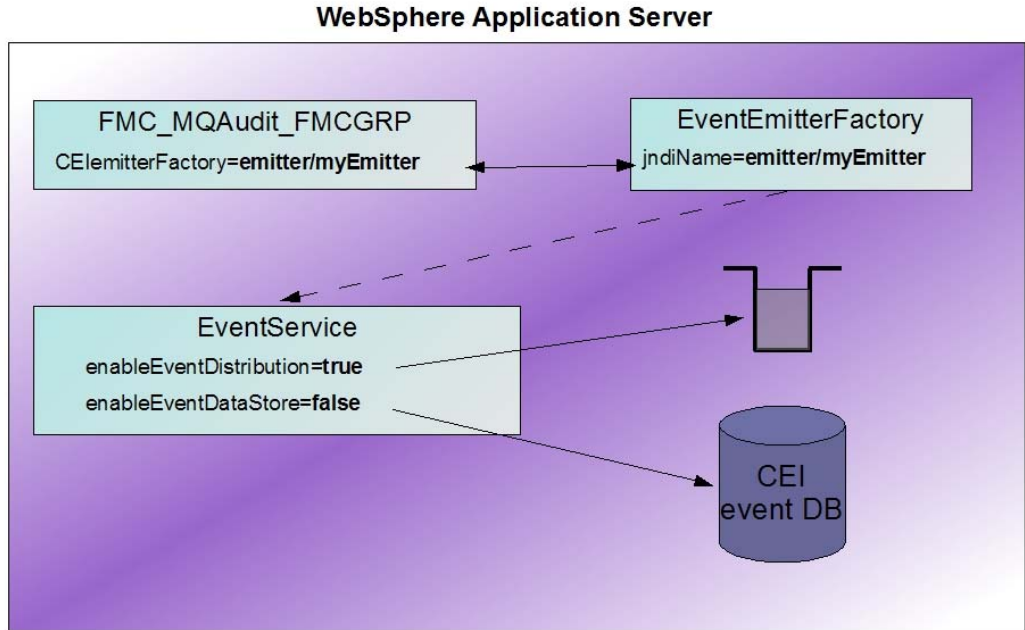
The WebSphereMQ Workflow Event Converter application must be customized and deployed on the WebSphere Application Server.

The following diagram illustrates how the WebSphere Application Server is customized. It also illustrates the relationship between the existing WebSphere MQ Workflow FDL definitions and the WebSphere MQ queue manager. In addition, it shows the queue definitions that WebSphere MQ Workflow uses for auditing.

Hint: FMC_MQAudit_FMCGRP in the following diagrams is the name of the event converter sample application.



The relationship between the Event Converter application and the Common Event Infrastructure is shown in the following diagram.



The customization and deployment tasks are performed via wsadmin script converter.py.

Note: A description of all objects configured by this script can be found in the next subchapter "Configured objects". You can note down your configured values in the tables provided there.

To start the script do the following:

1. Open a command prompt and navigate to the directory "install" below the directory where you have unzipped the SupportPac.
2. Issue the following command from the command line:

```
%WAS_HOME%/bin/wsadmin -f converter.py
```

If security is enabled, you are prompted for the userID and the password.

You are then prompted to enter the necessary information.

The script starts with a welcome message and prompts for the name of the Workflow system group that the audit messages are processed for. The name of the WebSphere MQ Workflow system group is appended to the application name of the deployed Event Converter.

Default values are enclosed in brackets. Your selection is echoed in the following line, preceded by the characters ==>.

```
-----
-----
Welcome to setup of SupportPac WA61 - WebSphere MQ Workflow
Enabling WebSphere Business Monitor
-----
-----
```

```
Enter MQ Workflow system group name: [FMCGRP]
==> FMCGRP
```

If you are installing the application in an network deployment (ND) environment, you are prompted to select the installation target.

```
Select installation target for 'FMC-MQAuditQ-FMCGRP'

  1) Server: nlserver1
(cells/fmtc7050Cell102/nodes/fmtc7050Node01/servers/nlserver1)

Your selection :[1]

==> Server: nlserver1
(cells/fmtc7050Cell102/nodes/fmtc7050Node01/servers/nlserver1)
```

After the completion of these steps, the connections from WebSphere Application Server to the involved middleware products are configured.

1. WebSphere MQ

To configure the connection to WebSphere MQ, first of all the queue manager name and the queue name of the WebSphere MQ Workflow audit queue must be specified.

```
Configure connection to MQ Workflow audit queue
=====
Enter MQ WORKFLOW queue manager name: [FMCQM]
==> FMCQM
Enter MQ WORKFLOW audit queue name: [FMC.FMCGRP.AUDITQ]
==> FMC.FMCGRP.AUDITQ
```

The script provides a selection of queue connection factories that match the previously entered parameters. If an appropriate connection factory is not available, a new one can be configured.

```
Select appropriate queue connection factory

  1) Create a new queue connection factory

Your selection :[1] 1

==> Create a new queue connection factory
```

When configuring a new connection factory, do the following:

1. Select a JMS provider.
2. Enter a display name for the queue connection factory.
3. Enter a unique JNDI name for the connection factory.

```
Select appropriate JMS provider

  1) WebSphere MQ JMS Provider
(cells/FMTC5088Node01Cell/nodes/N1/servers/server1)
```

```
2) WebSphere MQ JMS Provider
(cells/FMTC5088Node01Cell/nodes/N1)

3) WebSphere MQ JMS Provider (cells/FMTC5088Node01Cell)
Your selection :[1] 1

==> WebSphere MQ JMS Provider
(cells/FMTC5088Node01Cell/nodes/N1/servers/server1)

Enter display name: [FMCQM]
==> FMCQM

Enter JNDI name: [jms/FMCGRP/FMCQM]
==> jms/FMCGRP/FMCQM
```

4. Select a transport type for the connection to the queue manager.

```
Select appropriate transport type

1) BINDINGS
2) CLIENT

Your selection :[1] 1

==> BINDINGS
```

If you select transport type bindings, you are prompted to check the WAS environment variable MQ_INSTALL_ROOT.

```
MQ_INSTALL_ROOT is set to '${WAS_INSTALL_ROOT}/lib/WMQ'
Do you want to change it?

y) yes
n) no

Your selection :[n] y

==> yes

Enter value for MQ_INSTALL_ROOT:
[${WAS_INSTALL_ROOT}/lib/WMQ]
C:\IBM\WMQ

==> C:\IBM\WMQ
```

After entering the new value, you are prompted to save your changes.

```
Do you want to
  1) save your changes
  2) repeat this configuration step
  3) abort the script
Your selection :[1]
==> save your changes
The configuration has been saved.
```

If you select the transport type `client`, you are prompted to enter the host name, the port number, and the channel name of the queue manager that your WebSphere MQ Workflow belongs to.

```

Enter hostname: myhost

==> myhost

Enter port number: [14000]

==> 14000

Enter channel name: [FMCQM.CL.TCP]

==> FMCQM.CL.TCP

```

The parameters of the newly created queue connection factory are displayed and you are prompted to save your changes.

```

MQQueueConnectionFactory:

  JMSProvider: "WebSphere JMS

Provider(cells/FMTC5088Node01Cell/nodes/N1/servers/server
1|resources.xml#builtin_jmsprovider)"

  name: FMCQM

  jndiName: jms/FMCGRP/FMCQM

  description: Connection factory for MQ WORKFLOW Event
Converter SystemGroup: FMCGRP

  queueManager: FMCQM

  transportType: BINDINGS

  host: localhost

Do you want to

  1) save your changes

  2) repeat this configuration step

  3) abort the script

Your selection :[1]

==> save your changes

The configuration has been saved.

```

The script provides a selection of queues that match the previously entered parameters. If an appropriate queue is not available, you can configure a new one.

```
Select appropriate queue definition
  1) Create a new queue definition

Your selection :[1]
==> Create a new queue definition
```

When configuring a new JMS queue, do the following:

1. Enter a display name for the queue.
2. Enter a unique JNDI name for the queue.

```
Enter name:  [FMC.FMCGRP.AUDITQ]
==> FMC.FMCGRP.AUDITQ
Enter JNDI name:  [jms/FMCGRP/FMC.FMCGRP.AUDITQ]
==> jms/FMCGRP/FMC.FMCGRP.AUDITQ
```

The parameters of the newly created queue are displayed and you are prompted to save your changes.

```
MQQueue:  JMSProvider: "WebSphere JMS
Provider(cells/FMTC5088Node01Cell/nodes/N1/servers/server1|resources.x
ml#builtn_jmsprovider)"  name: FMC.FMCGRP.AUDITQ  jndiName:
jms/FMCGRP/FMC.FMCGRP.AUDITQ
  description: Queue definition for MQ WORKFLOW Event Converter
  SystemGroup: FMCGRP  baseQueueName: FMC.FMCGRP.AUDITQ
  ..baseQueueManagerName: FMCQM
Do you want to
  1) save your changes
  2) repeat this configuration step
  3) abort the script
Your selection :[1]
==> save your changes
The configuration has been saved.
```

To receive messages from MQ, a listener port defined in the WAS configuration must be assigned to the MDB.

If a listener port that matches the previously entered parameters is already defined, it is displayed and you are prompted to select it. Otherwise, a new listener port is created.


```

Select appropriate listener port
  1) WMQWFAuditListener-FMCGRP
      connection factory JNDI: jms/FMCGRP/FMCQM
      destination JNDI: jms/FMCGRP/FMC.FMCGRP.AUDITQ
Your selection :[1]
==> WMQWFAuditListener-FMCGRP      connection factory JNDI:
jms/FMCGRP/FMCQM      destination JNDI: jms/FMCGRP/FMC.FMCGRP.AUDITQ

```

When configuring a new listener port, do the following:

1. Enter a display name for the queue.
3. Enter a unique JNDI name for the queue.

```

Create a new listener port
Select appropriate message listener service
  1) (cells/FMTC5088Node01Cell/nodes/N1/servers/server1)
Your selection :[1]
==> (cells/FMTC5088Node01Cell/nodes/N1/servers/server1)
Enter name: [WMQWFAuditListener-FMCGRP]
==> WMQWFAuditListener-FMCGRP

```

The parameters of the newly created queue connection factory are displayed and you are prompted to save your changes.

```

ListenerPort:
  MsgListenerService:
(cells/FMTC5088Node01Cell/nodes/N1/servers/server1|server.xml#MessageL
istenerService_1186493587343)
  name: WMQWFAuditListener-FMCGRP
  description: Listener port for MQ WORKFLOW Event Converter
SystemGroup: FMCGRP
  connectionFactoryJNDIName: jms/FMCGRP/FMCQM
  destinationJNDIName: jms/FMCGRP/FMC.FMCGRP.AUDITQ

Do you want to
  1) save your changes
  2) repeat this configuration step
  3) abort the script
Your selection :[1]
==> save your changes
The configuration has been saved.

```

2. Common Event Infrastructure

To configure the connection to the CEI, specify the user name and the password that is needed to authenticate the user for the CEI bus. The password remains invisible.

```
CEI emitter factory profile selection
=====

Specify JMS user id to authenticate with the Common Event
Infrastructure Bus:
==> myuser

Specify the password for userid 'myuser' :
```

The script provides a selection of all event services that are configured.

```
Select appropriate CEI emitter factory profile:
  1) Default Common Event Infrastructure emitter
      JNDI: com/ibm/events/configuration/emitter/Default
  2) other (enter JNDI name)
Your selection :[1]
==> Default Common Event Infrastructure emitter
      JNDI: com/ibm/events/configuration/emitter/Default
```

If no event service is available, you are prompted to deploy a default CEI event service.

Note: For the JMS authentication alias of the event service, the previously entered user ID and password are used.

```
No CEI emitter factory profile found in configuration repository.
Do you want to deploy a default CEI event service?
  y) yes
  n) no
  c) cancel

Your selection :[y] y
==> yes

Default Common Event Infrastructure emitter
      JNDI: com/ibm/events/configuration/emitter/Default
```

The newly deployed event service is displayed and you are prompted to save your changes.

Note: To activate this event service you must restart the application server before you start your application.

```
Do you want to
  1) save your changes
  2) repeat this configuration step
  3) abort the script

Your selection :[1]
==> save your changes
The configuration has been saved.
Note: To activate the changes, you must restart the application server
server1.
```

It is also possible to only specify the JNDI name of the CEI emitter factory and configure the CEI service afterwards.

```
Select appropriate CEI emitter factory profile:
  1) Default Common Event Infrastructure emitter
      JNDI: com/ibm/events/configuration/emitter/Default
  2) other (enter JNDI name)

Your selection :[1] 2
==> other (enter JNDI name)

Enter JNDI name: [com/ibm/events/configuration/emitter/Default]
==> com/ibm/events/configuration/emitter/Default
```

If the storing of events in the CEI data store is enabled for the selected event service, you are prompted to disable it. Storing events in the CEI data store is not recommended in productive environments for performance reasons. For more information, see “CEI Event Datastore” in Chapter 6, “Performance Considerations”

```
The data store of the event service is enabled.
Do you want to disable it?
  y) yes
  n) no

Your selection :[y] y
==> yes
```

For the event converter application to emit start and stop events, the StartupBeansService needs to be enabled. If this service is disabled for the installation target, you are prompted to enable it.

```

The StartupBeanService is disabled for the installation target.
Do you want to enable it?
  y) yes
  n) no

Your selection :[y]
==> yes
The StartupBeanService has been enabled.

```

3. *Installing the application*

Finally the application is installed and you are prompted to save your changes.

Note: The application must be started from the WAS administration console.

```

The event converter ear file has been installed
Do you want to save the changes?
  y) yes
  n) no

Your selection :[y] y
==> yes

The configuration has been saved.
Start the application 'FMC_MQAudit_FMCGRP' from WAS admin console

```

Configured objects

This section describes the objects that are configured in WebSphere Application Server during the installation of WA61.

Queue connection factory

A queue connection factory is used to connect WAS to the queue manager hosting the WebSphere MQ Workflow audit queue.

You can view or change the attribute values of this queue connection factory by using the administration console. On the console, go to:

Resources --> JMS --> Queue connection factories

The default configuration of the queue connection factory is shown in the following table (you can note down your customized values in the last column):

Attribute	Default value	Your configured value
Name	FMCQM	
JNDI name	jms/FMCGRP/FMCQM	

Queue manager	FMCQM	
Transport type	BINDINGS	

If the transport type is CLIENT, the following additional attributes are defined by default:

Attribute	Default value	Your configured value
Host	<no default>	
Port	14000	
Channel	FMCQM.CL.TCP	

JMS queue

The WebSphere MQ Workflow audit queue is accessed by means of a JMS queue.

You can view or change the attribute values of the audit queue by using the administration console. On the console, go to:

Resources --> JMS --> Queues

The default configuration of the audit queue is shown in the following table (you can note down the value you configured in the last column):

Attribute	Default value	Your configured value
Name	FMC.FMCGRP.AUDITQ	
JNDI name	jms/ FMC.FMCGRP.AUDITQ	
Base queue name	FMC.FMCGRP.AUDITQ	
Base queue manager name	FMCQM	

Listener Port

The listener port defines the connection between MDB and the WebSphere MQ Workflow audit queue.

You can view or change the attribute values of the listener port by using the administration console. On the console, go to:

servers --> Application servers --> <server name> --> Messaging --> Message Listener Service --> Listener Ports

The default configuration of the listener port is shown in the following table (you can note down your customized values in the last column):

Attribute	Default value	Your configured value
Name	WMQWFAuditListener-FMCGRP	

Connection factory JNDI name	jms/FMCGRP/FMCQM	
Destination JNDI name	jms/ FMC.FMCGRP.AUDITQ	

Chapter 3. Event Description

The Event Converter emits two different kinds of events:

1. Events that are emitted when the event converter is started or stopped. The event converter application is the originator of these events. They give notification that the WA61 event converter starts or stops processing.
2. Events that are retrieved from WebSphere MQ Workflow audit trail messages are converted to the CBE format.

Both are described in the following.

Start and stop events

These events have the category name "StartSituation" or "StopSituation" in the situation field.

Sample start event (CBE)

```

<CommonBaseEvent creationTime="2007-11-14T11:29:46.513Z"
  extensionName="MQWF_EVENT"
  globalInstanceId="CED660C52FF9FFA939A1DC92A4E72E3810"
  sequenceNumber="1"
  severity="10"
  version="1.0.1">
  <contextDataElements name="WMQWF_EventVersion" type="string">
    <contextValue>1.0</contextValue>
  </contextDataElements>
  <sourceComponentId component="WebSphere MQ Workflow:SupportPac
WA61#1.0"
    componentIdType="ProductName"
    instanceId="FMC_MQAudit_FMCGRP"
    location="FMTC5088.boeblingen.de.ibm.com"
    locationType="FQHostname"
    subComponent="MQAudit"
    threadId="229"

    componentType="http://www.ibm.com/namespaces/autonomic/WS_RT/De
ployableObject"/>
  <situation categoryName="StartSituation">
    <situationType
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="StartSituation"
      reasoningScope="EXTERNAL"
      successDisposition="SUCCESSFUL"
      situationQualifier="START COMPLETED"/>
  </situation>
</CommonBaseEvent>

```


Sample stop event (CBE):

```

<CommonBaseEvent creationTime="2007-11-14T11:10:21.966Z"
  extensionName="MQWF_EVENT"
  globalInstanceId="CED660C52FF9FFA939A1DC92A23110E3E0"
  sequenceNumber="1" severity="10"
  version="1.0.1">
  <contextDataElements name="WMQWF_EventVersion"
    type="string">
    <contextValue>1.0</contextValue>
  </contextDataElements>
  <sourceComponentId component="WebSphere MQ Workflow:SupportPac
WA61#1.0"
    componentIdType="ProductName"
    instanceId="FMC_MQAudit_FMCGRP"
    location="FMTC5088.boeblingen.de.ibm.com"
    locationType="FQHostname"
    subComponent="MQAudit"
    threadId="226"
    componentType="http://www.ibm.com/namespaces/autonomic/WS_RT/De
ployableObject"/>
    <situation categoryName="StopSituation">
      <situationType
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="StopSituation"
        reasoningScope="EXTERNAL"
        successDisposition="SUCCESSFUL"
        situationQualifier="STOP COMPLETED"/>
    </situation>
  </CommonBaseEvent>

```

Converted events

These events have the category name "ReportSituation" in the situation field. They contain the converted WebSphere MQ Workflow audit trail information.

The WMQWF_Event.xsd, which can be found in the "doc" subdirectory of the SupportPac, defines the structure of the emitted CBEs:

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
Licensed Materials - Property of IBM
5697FM
(C) Copyright IBM Corporation 2007. All Rights Reserved.
US Government Users Restricted Rights- Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
-->
<xsd:schema
targetNamespace="http://www.ibm.com/xmlns/prod/websphere/monitoring/wmqwf/6.1"
    xmlns="http://www.w3.org/2001/XMLSchema"

xmlns:wmqwf="http://www.ibm.com/xmlns/prod/websphere/monitoring/wmqwf/
6.1"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="qualified"
    attributeFormDefault="qualified">
<xsd:element name="event" type="wmqwf:Event"/>
<xsd:complexType name="Event">
    <xsd:sequence>
        <!-- eventPointData has the one of the following types:
            WMQWF.Process,
            WMQWF.ProcessStart,
            WMQWF.Activity,
            WMQWF.Navigation,
            WMQWF.ActivityNotification,
            WMQWF.ProcessNotification,
            WMQWF.Workitem
        -->
        <xsd:element name="eventPointData" type="xsd:anyType"/>
        <xsd:element name="applicationData"
type="wmqwf:WMQWF.ContainerData" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="WMQWF.ContainerData">
    <xsd:sequence>
        <xsd:element name="value" type="xsd:anyType"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="WMQWF.Base">
    <xsd:sequence>
        <xsd:element name="EventCreation" type="xsd:dateTime"/>

```

```

    <xsd:element name="EventCode" type="xsd:string"/>
    <xsd:element name="ProcessInstanceName" type="xsd:string"/>
    <xsd:element name="ProcessInstanceID" type="xsd:string"/>
    <xsd:element name="TopLevelProcessInstanceName"
type="xsd:string"/>
    <xsd:element name="TopLevelProcessInstanceID"
type="xsd:string"/>
    <xsd:element name="ParentProcessInstName" type="xsd:string"
minOccurs="0"/>
    <xsd:element name="ParentProcessInstID" type="xsd:string"
minOccurs="0"/>
    <xsd:element name="ProcessTemplateName" type="xsd:string"/>
    <xsd:element name="ProcessTemplateValidFrom" type="xsd:string"/>
    <xsd:element name="UserID" type="xsd:string" minOccurs="0"/>
    <xsd:element name="AssociatedObjectID" type="xsd:string"
minOccurs="0"/>
    <xsd:element name="ObjectDescription" type="xsd:string"
minOccurs="0"/>
    <xsd:element name="ProcessInstanceContext" type="xsd:string"
minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="WMQWF.Process">
  <!-- 21001,21002,21004,21005,21017,21018,21019,21020,21025,
21026,21037,21038,21039,21040,21052,21082 -->
  <xsd:complexContent>
    <xsd:extension base="wmqwf:WMQWF.Base"/>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="WMQWF.ProcessStart">
  <!-- 21000-->
  <xsd:complexContent>
    <xsd:extension base="wmqwf:WMQWF.Process">
      <xsd:sequence>
        <!-- In case the WMQWF profile variable
EnableProcessActivityCorrelation is set
           to TRUE, "AssociatedObjectID" and the following two
attributes are set to
           the values of the calling process activity -->
        <xsd:element name="BlockNames" type="xsd:string"
minOccurs="0"/>

```

```

        <xsd:element name="ActivityName" type="xsd:string"
minOccurs="0"/>
    </xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="WMQWF.Activity">
    <!-- 21006,21007,21011,21012,21013,21014,21015,21016,21021,21022,
21023,21027,21031,21032,21041,21044,21053,21056,21080,21081-->
    <xsd:complexContent>
        <xsd:extension base="wmqwf:WMQWF.Base">
            <xsd:sequence>
                <xsd:element name="BlockNames" type="xsd:string"/>
                <xsd:element name="ActivityName" type="xsd:string"/>
                <xsd:element name="ActivityType" type="xsd:string"/>
                <xsd:element name="ActivityState" type="xsd:string"/>
                <xsd:element name="ActivityRC" type="xsd:string"
minOccurs="0"/>
                <xsd:element name="ProgramName" type="xsd:string"
minOccurs="0"/>
                <xsd:element name="CommandParameters" type="xsd:string"
minOccurs="0"/>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="WMQWF.Navigation">
    <!-- 21034-->
    <xsd:complexContent>
        <xsd:extension base="wmqwf:WMQWF.Base">
            <xsd:sequence>
                <xsd:element name="ActivityName" type="xsd:string"/>
                <xsd:element name="SecondActivityName" type="xsd:string"/>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="WMQWF.ActivityNotification">
    <!-- 21008,21024-->
    <xsd:complexContent>
        <xsd:extension base="wmqwf:WMQWF.Activity">

```

```

    <xsd:sequence>
      <xsd:element name="SecondUserID" type="xsd:string"/>
    </xsd:sequence>
  </xsd:extension>
</xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="WMQWF.ProcessNotification">
  <!-- 21003-->
  <xsd:complexContent>
    <xsd:extension base="wmqwf:WMQWF.Process">
      <xsd:sequence>
        <xsd:element name="SecondUserID" type="xsd:string"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="WMQWF.Workitem">
  <!-- 21009,21010,21030-->
  <xsd:complexContent>
    <xsd:extension base="wmqwf:WMQWF.Base">
      <xsd:sequence>
        <xsd:element name="BlockNames" type="xsd:string"/>
        <xsd:element name="ActivityName" type="xsd:string"/>
        <xsd:element name="SecondUserID" type="xsd:string"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
</xsd:schema>

```

The following describes the structure of the emitted CBEs. The names in bold represent variables that are substituted according to the following description.

```

<CommonBaseEvent      creationTime=CreationTime
extensionName="WMQWF_EVENT"
                    version="1.0.1" severity=10 >

<contextDataElements name="WBIRuntimeVersion" type="string">
    <contextValue>6.1</contextValue>
</contextDataElements>
<reporterComponentId component="WMQWFEventConverter"
componentIdType="ProductName"
                    location=Hostname locationType="FQHostname"
                    subComponent="WorkFlowEventEmitter"

                    componentType="{http://www.ibm.com/namespaces/autonomic/?Conver
ter?}" />

<sourceComponentId component="WMQWF" componentIdType="ProductName"
                    location="NA" locationType="Unknown"
                    subComponent="WorkFlowEngine"
application=SystemGroup

                    componentType="{http://www.ibm.com/namespaces/autonomic/Workflo
w_Engine}" />
<situation categoryName="ReportSituation">
    <situationType      xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
                    xsi:type="ReportSituation"
reasoningScope="EXTERNAL"
                    reportCategory="STATUS" />
</situation>
<wbi:event      xsi:type="wbi:Event"
                    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

                    xmlns:wbi="http://www.ibm.com/xmlns/prod/websphere/monitoring/w
mqwf/6.1">
    <wbi:eventPointData>
        <EventCreation>CreationTime</ EventCreation >
        <EventCode>EventCode</EventCode>
        <ProcessInstanceName>ProcessInstanceName</
ProcessInstanceName >
        <ProcessInstanceID>ProcessInstanceID</ProcessInstanceID>

        <TopLevelProcessInstanceName>TopLvlProcessInstName</TopLevelPro
cessInstanceName>
        <TopLevelProcessInstanceID>TopLevelProcessInstID</
TopLevelProcessInstanceID>

```

```

        <ParentProcessInstanceName>ParentProcessInstName</ParentProcess
InstanceName>

        <ParentProcessInstanceID>ParentProcessInstID</ParentProcessInsta
nceID>

        <ProcessTemplateName>ProcessTemplateName</ProcessTemplateName>

        <ProcessTemplateValidFrom>ProcessTemplateValidFrom</ProcessTemp
lateValidFrom>
            <BlockNames>BlockNames</BlockNames>
            <UserID>UserID</UserID>
            <SecondUserID>SecondUserID</SecondUserID>
            <ActivityName>ActivityName</ActivityName>

            <SecondActivityName>SecondActivityName</SecondActivityName>
            <ActivityType>ActivityType</ActivityType>
            <ActivityState>ActivityState</ActivityState>
            <ActivityRC>ActivityRC</ActivityRC>
            <CommandParameters>CommandParameters</CommandParameters>

            <AssociatedObjectID>AssociatedObjectID</AssociatedObjectID>
            <ObjectDescription>ObjectDescription</ObjectDescription>
            <ProgramName>ProgramName</ProgramName>

            <ProcessInstanceContext>ProcessInstanceContext</ProcessInstance
Context>
            </wbi:eventPointData>
            <wbi:applicationData>
                ContainerContent
            </wbi:applicationData>
        </wbi:Event>
    </CommonBaseEvent>

```

Variable substitution:

The Hostname variable is set to the host name of the computer that the WebSphere MQ Workflow Event Converter is running on.

The SystemGroup variable holds the system group name of the workflow that is configured during the deployment.

The other fields are filled with information retrieved from audit records as the following table shows:

CBE field	MQ Audit
-----------	----------

CBE field	MQ Audit
CreationTime	mqwf:Timestamp
EventCode	mqwf:AuditEvent
ProcessInstanceName	mqwf:ProcInstName
ProcessInstanceID	mqwf:ProcInstID
TopLvlProcessInstName	mqwf:ProcInstTopLevelName
TopLevelProcessInstID	mqwf:ProcInstTopLevelID
ParentProcessInstName	mqwf:ProcInstParentName
ParentProcessInstID	mqwf:ProcInstParentID
ProcessTemplateName	mqwf:ProcTemplName
ProcessTemplateValidFrom	mqwf:ProcTemplValidFromDate
BlockNames	mqwf:BlockNames
UserID	mqwf:UserID
SecondUserID	mqwf:SecondUserID
ActivityName	mqwf:ActivityName
SecondActivityName	mqwf:SecondActivityName
ActivityType	mqwf:ActivityType
ActivityState	mqwf:ActivityState
ActivityRC	mqwf:ProgramRC
CommandParameters	mqwf:CommandParameters
AssociatedObjectID	mqwf:AssociatedObject
ObjectDescription	mqwf:ObjectDescription
ProgramName	mqwf:ProgramName
ProcessInstanceContext	mqwf:ExternalProcessContext
ContainerContent	mqwf:GlobalContainerContent or mqwf:InputContainerContent or mqwf:OutputContainerContent

Sample CBE:

```
<CommonBaseEvent creationTime="2007-11-14T11:48:17.307376Z"
  extensionName="WMQWF.Process"
  globalInstanceId="CED1EB9435E3F69248A1DC92A7342F96C0"
```



```

sequenceNumber="1"
severity="10" version="1.0.1">
<contextDataElements name="WMQWF_EventVersion" type="string">
  <contextValue>1.0</contextValue>
</contextDataElements>
<contextDataElements name="ECSCurrentID" type="ECSID">
<contextValue>UAAAAAEADIAYAAAAAAAAAAAAAAAAABAAHAAAAAAAAAAAAUA==</contextValue>
</contextDataElements>
<reporterComponentId component="WebSphere MQ Workflow:SupportPac
WA61#1.0"
  componentIdType="ProductName"
  instanceId="FMC_MQAudit_FMCGRP"
  location="FMTC5088.boeblingen.de.ibm.com"
  locationType="FQHostname"
  subComponent="MQAudit"
  threadId="218"
  componentType="http://www.ibm.com/namespaces/autonomic/WS_RT/DeployableObject"/>
  <sourceComponentId component="CreditRequest"
    componentIdType="Process"
    executionEnvironment="WebSphere MQ Workflow"
    instanceId="UAAAAAEADIAYAAAAAAAAAAAAAAAAABAAHAAAAAAAAAAAAUA=="
    location="FMCGRP"
    locationType="Unknown"
    subComponent=" "
    componentType="http://www.ibm.com/namespaces/autonomic/Workflow_Engine"/>
  <situation categoryName="ReportSituation">
    <situationType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="ReportSituation"
      reasoningScope="EXTERNAL"
      reportCategory="STATUS"/>
  </situation>
  <wmqwf:event xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wmqwf="http://www.ibm.com/xmlns/prod/websphere/monitoring/wmqwf/6.1">
    <wmqwf:eventPointData xsi:type="wmqwf:WMQWF.Process">
      <wmqwf:EventCreation>2007-11-14T11:48:17.307376Z</wmqwf:EventCreation>
      <wmqwf:EventCode>21018</wmqwf:EventCode>

```

```

<wmqwf:ProcessInstanceName>CreditRequest</wmqwf:ProcessInstanceName>

<wmqwf:ProcessInstanceID>UAAAAAEADIAYAAAAAAAAAAAAAAAABAAHAAAAAAAAAAAAU
A==</wmqwf:ProcessInstanceID>

<wmqwf:TopLevelProcessInstanceName>CreditRequest</wmqwf:TopLevelProces
sInstanceName>

<wmqwf:TopLevelProcessInstanceID>UAAAAAEADIAYAAAAAAAAAAAAAAAABAAHAAAAAA
AAAAAAUA==</wmqwf:TopLevelProcessInstanceID>

<wmqwf:ProcessTemplateName>CreditRequest</wmqwf:ProcessTemplateName>
    <wmqwf:ProcessTemplateValidFrom>2007-05-
29T13:58:17.046000Z</wmqwf:ProcessTemplateValidFrom>
    <wmqwf:UserID>ADMIN</wmqwf:UserID>

<wmqwf:AssociatedObjectID>UAAAAAEADIAYAAAAAAAAAAAAAAAABAAHAAAAAAAAAAAA
UA==</wmqwf:AssociatedObjectID>
    <wmqwf:ObjectDescription>Credit
request</wmqwf:ObjectDescription>

<wmqwf:ProcessInstanceContext>MyProcessContext</wmqwf:ProcessInstanceC
ontext>
    </wmqwf:eventPointData>
    <wmqwf:applicationData>
        <wmqwf:value xsi:type="wmqwf:PersonInfo">
            <wmqwf:FirstName>John</wmqwf:FirstName>
            <wmqwf:LastName>Doe</wmqwf:LastName>
        </wmqwf:value>
    </wmqwf:applicationData>
</wmqwf:event>
</CommonBaseEvent>

```

Chapter 4. Uninstalling

To uninstall the WA61 Event Converter, the application and several configuration objects must be removed from the WAS configuration. To remove these objects, you can use the WAS administration console.

The objects that are configured during the installation of WA61 are described in “Installation and Customization, subchapter “Configured objects”.

Note: The various configuration entities may be used by other applications as well. Only delete the objects from the WAS configuration if you are sure that they are not needed any more.

Uninstalling the application

1. On the administration console, navigate to **Applications > Enterprise Applications**.
2. Select the converter application from the list and click **Uninstall**.
3. Click **OK** to confirm the deinstallation.
4. Click **Save** to save your configuration.

Deleting the listener port

1. On the administration console, navigate to **Servers > Application servers**.
2. Select your server from the list.
3. On the right, select **Messaging > Message Listener Service**.
4. Select **Listener Ports**
5. Select the check box of the appropriate listener port and click **Delete**.
6. Click **Save** to save your configuration.

Deleting JMS queue

1. On the administration console, navigate to **Resources > JMS > Queues**.
2. Select the check box of the appropriate queue and click **Delete**.
3. Click **Save** to save your configuration.

Deleting the queue connection factory

1. On the administration console, navigate to **Resources > JMS > Queue connection factories**.
2. Select the check box of the appropriate queue connection factory and click **Delete**.

3. Click **Save** to save your configuration.

Chapter 5. Problem Determination

Traces

In case that traces are needed, set the following trace settings on the WAS administration console:

```
com.ibm.workflow.monitor.*=all
```

Version String

To identify the version of the SupportPac you have installed, the WA61 event converter application writes a version string to the *system out* log when it is started.

The version string has the form:

```
WebSphere MQ Workflow:SupportPac WA61#V1.0#20071203
```

Flight Recorder

For performance analyses, the WA61 Event Converter implements a “flight recorder”, which records the processing times of five subcomponents:

Subcomponent	Description
MQ	Time needed to receive the next message from MQ (includes wait times when no messages are available)
PARSER	Time for building up an internal data structure from an audit message
CBE	Time for building up the CBE event from the internal data structure
CEI	Time for sending the CBE to the CEI server
OVERALL	Overall processing time (without wait times on MQ)

You can enable the flight recorder by the following trace setting:

```
com.ibm.workflow.monitor.performance.*
```

The frequency of log entries depends on the value of this setting:

Trace value	Frequency
<i>fine</i>	Every 100000 messages
<i>finer</i>	Every 10000 messages
<i>finest</i>	Every 1000 messages

When the Event Converter stops, information about this event is logged as well.

The following information is written to the trace log of each subcomponent:

```
OVERALL FLIGHT RECORDER: OVERALL  
OVERALL min: 15 ms  
OVERALL max: 657 ms  
OVERALL total: 45,550 ms  
OVERALL count: 1000
```

The first line shows the flight recorder subcomponent.

The second line displays the minimum time that this subcomponent needed for the processing of one message.

The third line displays the maximum time this subcomponent needed for processing one message.

The fourth line displays the total time that the subcomponent needed.

The fifth line displays the number of messages that have been processed by the subcomponent during the measured interval.

Chapter 6. Performance Considerations

Filtering of WebSphere MQ Workflow audit events

Writing audit events to WebSphere MQ decreases the performance of your WebSphere MQ Workflow runtime servers. Additionally, the network workload increases when many events are sent. Therefore audit filtering is strongly recommended. Audit filtering restricts the writing of events to those events that are relevant to your purpose.

With WebSphere MQ Workflow 3.6 Service Pack 6, audit trail records can contain input and output containers. As the size of the events depends on the size of the container, the container size affects the performance and the network load as well.

If the `EnableContainerDataInAuditTrail` profile variable is set to `TRUE` (see “*Profile variables*” in chapter 2) some events will hold a container. A list of these events can be found in the TecDoc “*Enhanced audit trail capabilities to log container data*”.

This chapter describes a number of sample monitoring scenarios that are based on the `CreditSample` process. This process is delivered with WebSphere MQ Workflow and shows the events that must be enabled.

For detailed information about the customizing of WebSphere MQ Workflow, see chapter 2, “Installation and Customization”, “WebSphere MQ Workflow”.

This documentation covers only those steps that have to be performed for WebSphere MQ Workflow. Afterwards, the FDL must be converted into a monitor model with the “FDL to monitor model utility”. You can find detailed information in the documentation of this utility..

Scenario 1: Process events

This scenario describes how to use events of the monitoring process.

By activating events of the types *process created* and *process deleted*, the monitor can show how many `CreditRequest` processes have been created and how many `CreditRequest` processes have been deleted. In calculating the difference between these two, the monitor determines how many `CreditRequest` processes remain within WebSphere MQ Workflow at a given point of time.

Events needed: 21017 - Process created
 21018 - Process created and started
 21020 - Process deleted
 21025 - Process ended normally and deleted
 21026 - Process terminated and deleted

To activate these events, you must update the `CreditRequest.fdl` in the following way:

```
PROCESS 'CreditRequest' ( 'PersonInfo', 'Default Data Structure' )
  DESCRIPTION 'Credit request'
  AUDIT_FILTER_MQ 21017, 21018, 21020, 21025, 21026
  PROMPT_AT_PROCESS_START
  ...
```

Scenario 2: Activity events of all activities

This scenario describes how to use events of all the activities of the monitoring process.

Activating events of the types *activity started* and *checkout activity*, the monitor can show how many activities have been navigated within a certain timeframe. Depending on the client, activities will change into state *running* issuing *activity started* event or into state *checked out* issuing *checkout activity* event.

Events needed: 21007 - Activity started
21022 - Checkout activity

To activate these events, update the CreditRequest.fdl in the following way:

```
PROCESS 'CreditRequest' ( 'PersonInfo', 'Default Data Structure' )
  DESCRIPTION 'Credit request'
  AUDIT_FILTER_MQ 21007, 21022
  PROMPT_AT_PROCESS_START
...
```

Scenario 3: Activity events of a specific activity

This scenario describes how to use events of a specific activity for monitoring.

Activating the *started events* activity of the RequestApproval activity, the monitor can show the number of credit requests that needed a manual approval step within a certain timeframe.

Events needed: 21007 - Activity started of activity RequestApproval
21022 - Checkout activity of activity RequestApproval

To activate this activity, you must update the CreditRequest.fdl in the following way:

```
PROGRAM_ACTIVITY 'RequestApproval' ( 'CreditInfo', 'CreditInfo' )
  DESCRIPTION 'Credit request for %CreditRequestor.FirstName%
    %CreditRequestor.LastName%'
  AUDIT_FILTER_MQ 21007, 21022
  START MANUAL WHEN AT_LEAST_ONE CONNECTOR TRUE
...
```

Scenario 4: Accessing container data

This scenario describes how to access container data of monitoring events.

By accessing the CreditAmount, the monitor can show the average or total amount of all CreditRequests within a certain timeframe. Additional dimensions can be added to the monitor model to show the amount by risk or approval state. The CreditSample process does not map the output containers of the activities to a sink, so an activity event is needed that holds the output container. The *checkin activity* event of the AcceptCredit and RejectCredit activities is used, since this is the event that holds all the data of the CreditRequest process that is interesting for monitoring.. If the process would map the output containers of the activities to a sink, the *process ended* event could be used as well.

Events needed: 21023 - Checkin activity of activity AcceptCredit
21023 - Checkin activity of activity RejectCredit

To activate these events, update the CreditRequest.fdl in the following way:

```

PROGRAM_ACTIVITY 'AcceptCredit' ( 'CreditInfo', 'Default Data Structure' )
  DESCRIPTION 'Credit request for %CreditRequestor.FirstName%
    %CreditRequestor.LastName%'
  AUDIT_FILTER_MQ 21023
  START MANUAL WHEN AT_LEAST_ONE CONNECTOR TRUE
...
PROGRAM_ACTIVITY 'RejectCredit' ( 'CreditInfo', 'Default Data Structure' )
  DESCRIPTION 'Credit request for %CreditRequestor.FirstName%
    %CreditRequestor.LastName%'
  AUDIT_FILTER_MQ 21023
  START MANUAL WHEN AT_LEAST_ONE CONNECTOR TRUE
...

```

Scalability of Event Converter application

To guarantee that all events that are sent from the different WebSphere MQ Workflow Workflow servers can be processed without building a backlog in the WebSphere MQ Workflow audit queue, it can be necessary to fine-tune the message-driven bean (MDB) of the WebSphere MQ Workflow Event Converter. The necessity for an adjustment of the parameter settings for the MDB depends on the number of WebSphere MQ Workflow execution server instances (which send out the WebSphere MQ audit trail messages) and the number of events to be processed.

Avoid full queues. As audit events must reflect the auditing transactions exactly, a full audit queue leads to a severe error, which results in a shutdown of the execution servers.

The MDB parameters that you can tweak are:

- Number of connections for the MDB
- Number of sessions for the listener port of the MDB
- Size of the thread pool for the listener port of the MDB
- Number of connections that are used to store CBEs in the CEI event data store (if needed)

Number of connections for the MDB

To adjust the number of connections, navigate to the connection pools page of your queue connection factory on the WebSphere Application Server administration console and update the minimum and maximum values for the connections to your needs. Specify the number of Workflow Execution server instances as the minimum value and four or five times this value as the maximum.

Path to the connection pools page:

Resources --> JMS --> Queue connection factories --> <yourConnectionFactoryName> --> Additional properties : Connection pools

Number of sessions for the defined Listener port of the MDB

Navigate to the Listener page of the MDB and update the `maximum number of sessions` parameter for your listener. Specify the maximum number of MDB connections as the value.

Path to the MDB listener page:

**Application servers --> <yourApplicationServerName> --> Messaging: Message Listener
Service --> Listener Ports --> <yourMDBListenerName>**

Size of the thread pool for the listener port of the MDB

Navigate to the thread pool page of the MDB and update the `maximum size` parameter of the listener. Specify the same value as for the maximum number of MDB connections.

Path to the MDB listener “Thread Pool” page:

**Application servers --> <yourApplicationServerName> --> Messaging: Message Listener
Service --> Thread Pool**

Number of connections for the CEI Event Database Store

Navigate to the “Connection pools” page of the CEI Event Database Store and update the `minimum number of connections` and the `maximum number of connections` parameters. Specify the same values as for the minimum and the maximum number of MDB connections.

Path to the “Connection pools” page:

JDBC providers -->Event_DB2_JDBC_Provider --> Data sources --> event --> Connection pools

CEI Event Datastore

Using the CEI Datastore for WebSphere MQ Workflow audit events has a big impact on the performance of the CEI and thus on the performance of the WebSphere MQ Workflow MQAudit enterprise application (MDB). If possible, do not use the CEI Event Datastore at all and have the events directly distributed to the target application (for example, WebSphere Monitor 6.1).

To update the event distribution or the event data store parameters, navigate to the CEI “Event Services” page on the WebSphere Application Server Administrative console and change the following flags as needed:

- **Enable event distribution**
- **Enable event data store**

Path to the CEI “Event Services” page:

**Service Integration --> Common Event Infrastructure --> Event Service --> Event Services -->
<yourEventService>**

----- End of Document -----