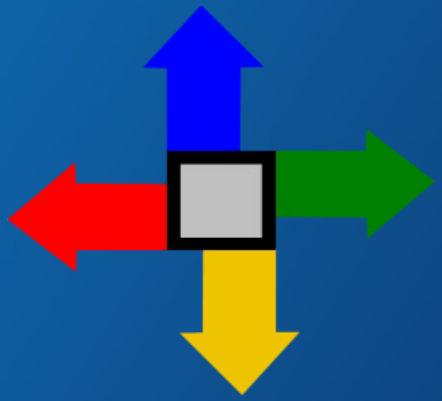
Managed File Transfer with Universal File Mover



Roger Lacroix
roger.lacroix@capitalware.biz
http://www.capitalware.biz

Universal File Mover Overview

- Universal File Mover (UFM) allows the user to combine business processes into a workflow.
- The user creates a UFM Workflow XML file which contains a series of Action commands.
- These Action commands define:
  - Which actions are to be taken,
  - The order of the actions,
  - And how errors are to be handled.

Universal File Mover Overview

- Prerequisite:
  - WebSphere MQ v5.3 or higher
  - Java Runtime Environment (JRE) 1.5 or higher

Universal File Mover Overview

- Universal File Mover (UFM and Universal File Mover Status Monitor (UFM-SM) are licensed under Apache License 2.
- They are free to use but support is not included. A support license may be purchased.
- http://www.capitalware.biz/ufm_overview.html

Universal File Mover Overview

UFM currently contains 41 Action commands. These action commands fall into 5 categories:

- WebSphere MQ Actions
- Network Actions
- File Actions
- Control Actions
- Other Actions

WebSphere MQ Action Commands

- **MQSend** - Sends 1 or more files as individual messages via MQ. Each file can be encrypted and/or compressed before it is sent. Each file can be sent to a single queue or multiple queues.
- **MQReceive** - Receives an incoming messages and writes them to a file (can be run as a daemon or as a Windows service). Receive Action will decrypt and/or decompress and encrypted and/or compressed messages.
- **MQPutQuit** - Puts a 'Quit' message on a queue (to stop the MQReceive Action running as a daemon)

Network Action Commands

- **Ftp** – Get and/or put files using FTP network protocol
- **HttpGetFile** - Get a file using HTTP application protocol from a web server
- **HttpPutFile** - Put a file to a web server using HTTP application protocol
- **Scp** – Securely copy files to and/or from a remote host
- **SendEmail** – Sends an email to 1 or more recipients.
- **SFtp** - Get and/or put files using Secure FTP network protocol

File Action Commands

- **Append** - Appends a file to another file
- **Convert** - Converts file format (i.e. CRLF <=> LF <=> CR and/or ASCII to EBCDIC)
- **Copy** - Copies 1 or more files from a directory to another directory
- **Delete** - Deletes 1 or more files in a directory
- **DecryptFile** - Decrypt a file using AES 128, 192 or 256-bit decryption
- **EncryptFile** - Encrypt a file using AES 128, 192 or 256-bit encryption

File Action Commands - continued

- **MakeDir** - Create a directory
- **Merge** - Merge 2 or more files to another file
- **MergeSort** - Merge 2 or more files and sort the data to another file
- **Move** - Moves 1 or more files from a directory to another directory
- **ReEncode** – Change the file's character encoding to another encoding
- **RemoveDir** - Remove a directory

File Action Commands - continued

- **Rename** - Renames a file
- **ReplaceText** - Performs a search and replace of text in a file.
- **Sort** - Sorts the data of a file into another file
- **SortUnique** - Sorts the data of a file into another file then removes duplicate lines from the file.
- **Split** - Split a file into several files
- **Tar** - Combine a file(s) or a directory of files into a tar archive

File Action Commands - continued

- **Touch** - Update a file's the modification time or create the file if it does not exist
- **Unique** - Removes duplicate lines from a file.
- **UnTar** - Extract tar archive to a directory
- **UnZip** - Uncompress a zip archive to a directory
- **Watch** - Monitor for a particular file or monitor a directory for files to appear and then executes a series of Actions.
- **WriteText** – Writes the text to a file.
- **Zip** - Compresses a file(s) or a directory of files into a Zip archive

Control Action Commands

- **If/Else** - Performs a conditional test against an action's variable
- **Loop** - Iterates over a group of Actions.
- **Sleep** – Pause the UFM Workflow for a period of time.

Other Action Commands

- **Echo** – Echo a text string to log file.
- **Execute** - Runs an external program / application
- **Launch** - Invokes an UFM Workflow XML file.
- **Schedule** - Invokes an UFM Workflow XML file at a specific date and/or time.

UFM Workflow XML – Sample 1

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_Workflow SYSTEM "UFM_Workflow.dtd">
<UFM_Workflow>
  <Actions>
    <Append file="data\atest.txt">
      <File dir="data">test.txt</File>
    </Append>
    <MQSend delete="Y" format="S">
      <File>data\test.txt</File>
      <MQ>
        <QMgrName>MQWT1</QMgrName>
        <QueueName>TEST.Q1</QueueName>
      </MQ>
      <Remote>
        <Directory>C:\temp</Directory>
      </Remote>
    </MQSend>
  </Actions>
</UFM_Workflow>
```

MQSend & MQReceive Info Exchange

- UFM MQSend Action places Remote Directory (path), Remote Filename and Remote JobName the MQMD header.
- Check for field length:

```
if ((remoteFilename.length() > MQC.MQ_APPL_IDENTITY_DATA_LENGTH)    ||
    (remoteDirectory.length() > MQC.MQ_ACCOUNTING_TOKEN_LENGTH)     ||
    ((remoteJobFile != null) &&
     ((remoteJobFile.length() - 4) > MQC.MQ_CORREL_ID_LENGTH) ) )
```

Note: If any field is too long to fit the UFM will create its own UFM embedded XML message.

UFM Workflow XML – Sample 2

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_Workflow SYSTEM "UFM_Workflow.dtd">
<UFM_Workflow>
  <Global>
    <MQ>
      <QMgrName>MQWT1</QMgrName>
      <QueueName>TEST.Q1</QueueName>
    </MQ>
  </Global>
  <Actions>
    <MQSend delete="Y" format="S">
      <File>data\test01.txt</File>
      <Remote>
        <Directory>C:\temp</Directory>
      </Remote>
    </MQSend>
    <MQSend delete="N" format="S">
      <File>otherdata\test02.txt</File>
      <Remote>
        <Directory>C:\temp</Directory>
      </Remote>
    </MQSend>
  </Actions>
</UFM_Workflow>
```

UFM Workflow XML – Sample 3

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_Workflow SYSTEM "UFM_Workflow.dtd">
<UFM_Workflow>
  <Global>
    <MQ>
      <MQFile>mq_mqwt1.xml</MQFile>
    </MQ>
  </Global>
  <Actions>
    <MQSend delete="Y" format="S">
      <File>data\test01.txt</File>
      <Remote>
        <Directory>C:\temp</Directory>
      </Remote>
    </MQSend>
    <MQSend delete="N" format="S">
      <File>otherdata\test02.txt</File>
      <Remote>
        <Directory>C:\temp</Directory>
      </Remote>
    </MQSend>
  </Actions>
</UFM_Workflow>
```

UFM_MQ XML Sample – bindings mode

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_MQ SYSTEM "UFM_MQ.dtd">
<UFM_MQ>
    <QMgrName>MQWT1</QMgrName>
    <QueueName>TEST.Q1</QueueName>
</UFM_MQ>
```

The UFM_MQ XML files must be stored in the **<UFM_Install_PATH>\mq** directory.

i.e. C:\Capitalware\UFM\mq\

UFM_MQ XML Sample – client mode

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_MQ SYSTEM "UFM_MQ.dtd">
<UFM_MQ>
    <QMgrName>MQWT1</QMgrName>
    <QueueName>TEST.Q1</QueueName>
    <Hostname>192.168.10.115</Hostname>
    <ChannelName>TEST.CHL</ChannelName>
    <Port>1415</Port>
    <UserID>tester</UserID>
</UFM_MQ>
```

The UFM_MQ XML files must be stored in the **<UFM_Install_PATH>\mq** directory.

i.e. C:\Capitalware\UFM\mq\

UFM Workflow XML Global Element

- The Global element has 2 attributes:
  - **onerrorfail** specifies whether or not the Workflow should abnormally terminate if an Action fails. The default value is 'Y'. This is a global value that can be overridden for each Action.
  - **usefilelocking** specifies whether or not the file locking will be used with the actions in this Workflow. The default value is 'Y'. This is a global value.

UFM Workflow XML Global Element

- The Global element has 6 child elements:
  - **Property** element specifies a used-defined token (keyword) and its value.
  - **Log4JFile** element specifies the path and filename of the log4j property file
  - **LogFile** element specifies the path and filename of the log4j logfile.
  - **OnError** element contains 2 element: Execute and SendEmail. OnError can contain either Execute or SendEmail Actions or both.
  - **MQ** element contains elements that describe how the action connects to the queue manager.
  - **Status** element contains an MQ element.

UFM Workflow XML – Sample 4

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_Workflow SYSTEM "UFM_Workflow.dtd">
<UFM_Workflow>
  <Global>
    <Property name="app_dir" value="D:\appdata" />
    <Property name="myfile" value="test.txt" />
    <MQ>
      <MQFile>mq_mqwt1.xml</MQFile>
    </MQ>
    <Status timetolive="30">
      <MQ>
        <MQFile>mq4status.xml</MQFile>
      </MQ>
    </Status>
  </Global>
  <Actions>
    <Move todir="${app_dir}">
      <File dir="data">${myfile}</File>
    </Move>
    <MQSend format="S">
      <File dir="${app_dir}">${myfile}</File>
      <Remote>
        <Directory>C:\temp</Directory>
      </Remote>
    </MQSend>
    <Execute xmlfile="appjob.xml" />
    <Delete>
      <File dir="${app_dir}">${myfile}</File>
    </Delete>
  </Actions>
</UFM_Workflow>
```

UFM Workflow XML – Sample 5

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_Workflow SYSTEM "UFM_Workflow.dtd">
<UFM_Workflow>
  <Global>
    <Status timetolive="30">
      <MQ>
        <MQFile>mq4status.xml</MQFile>
      </MQ>
    </Status>
  </Global>
  <Actions>
    <Watch pollinterval="5">
      <WatchItem type="D">
        <Directory>C:\temp\other</Directory>
        <Extension>txt</Extension>
        <Actions>
          <MQSend delete="Y" format="S">
            <File>${WATCH_FOUND_FILE}</File>
            <MQ>
              <MQFile>mq.xml</MQFile>
            </MQ>
            <Remote>
              <Directory>X:\something</Directory>
            </Remote>
          </MQSend>
        </Actions>
      </WatchItem>
    </Watch>
  </Actions>
</UFM_Workflow>
```

UFM Workflow XML – Sample 6

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_Workflow SYSTEM "UFM_Workflow.dtd">
<UFM_Workflow>
  <Global>
    <MQ>
      <MQFile>mq_mqwt1.xml</MQFile>
    </MQ>
    <Status timetolive="30">
      <MQ>
        <MQFile>mq4status.xml</MQFile>
      </MQ>
    </Status>
  </Global>
  <Actions>
    <MQReceive getwithconvert="N" run="E">
      <Default>
         <Directory override="Y">C:\temp\UFM</Directory>
      </Default>
    </MQReceive>
    <If actionname="MQReceive" fieldname="msgCount" operator="GT" value="0">
      <Actions>
         <Touch>
           <File dir="C:\temp">flag.txt</File>
         </Touch>
      </Actions>
    </If>
  </Actions>
</UFM_Workflow>
```

If Action Samples

```
<If actionname="Receive" fieldname="msgCount" operator="GT" value="0">
  <Actions>
    <Touch>
      <File dir="C:\temp">test.xml</File>
    </Touch>
  </Actions>
<Else>
  <Actions>
    <Copy todir="C:\temp\ufm" tofile="${FILE}_${DATE}">
      <File dir="data">test.xml</File>
    </Copy>
  </Actions>
</Else>
</If>

<!-- ---------------------------------------- -->

<If actionname="Receive" fieldname="msgCount" operator="GT" value="0">
  <Actions>
    <Touch>
      <File dir="C:\temp">test.xml</File>
    </Touch>
  <Actions>
<Else exitrc=8 />
</If>
```

UFM Workflow XML – Sample 7

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_Workflow SYSTEM "UFM_Workflow.dtd">
<UFM_Workflow>
  <Global>
    <MQ>
      <MQFile>mq_mqwt1.xml</MQFile>
    </MQ>
    <Status timetolive="30">
      <MQ>
        <MQFile>mq4status.xml</MQFile>
      </MQ>
    </Status>
  </Global>
  <Actions>
   <!-- Encryption -->
    <MQSend keysize="128" passphrase="this is the secret.">
      <File>data\test.txt</File>
      <Remote>
        <Directory>C:\temp</Directory>
      </Remote>
    </MQSend>
  </Actions>
</UFM_Workflow>
```

UFM Workflow XML – Sample 8

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_Workflow SYSTEM "UFM_Workflow.dtd">
<UFM_Workflow>
  <Global>
    <MQ>
      <MQFile>mq_mqwt1.xml</MQFile>
    </MQ>
    <Status timetolive="30">
      <MQ>
        <MQFile>mq4status.xml</MQFile>
      </MQ>
    </Status>
  </Global>
  <Actions>
   <!-- Compress and Encryption -->
    <MQSend format="Z" keysize="128" passphrase="this is the secret.">
      <File>data\test.txt</File>
      <Remote>
        <Directory>C:\temp</Directory>
      </Remote>
    </MQSend>
  </Actions>
</UFM_Workflow>
```

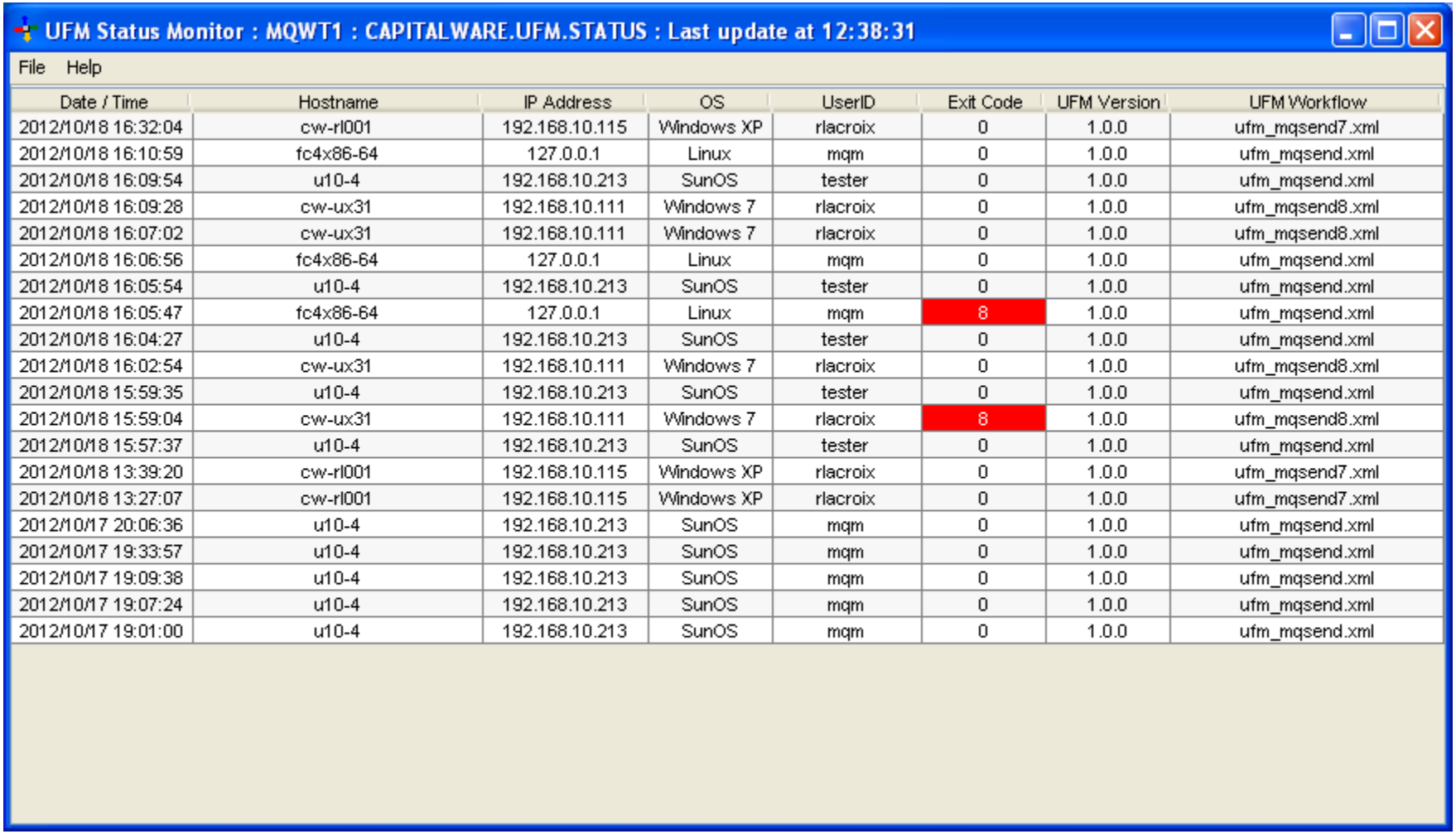
UFM Workflow XML – Sample 9

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE UFM_Workflow SYSTEM "UFM_Workflow.dtd">
<UFM_Workflow>
  <Global>
    <MQ>
      <MQFile>mq_mqwt1.xml</MQFile>
    </MQ>
    <Status timetolive="30">
      <MQ>
        <MQFile>mq4status.xml</MQFile>
      </MQ>
    </Status>
  </Global>
  <Actions>
   <!-- Decrpyt (automatically handle decompression) -->
    <MQReceive getwithconvert="N" run="E" keysize="128" passphrase="this is
the secret.">
      <Default>
         <Directory override="Y">C:\temp\UFM</Directory>
      </Default>
    </MQReceive>
  </Actions>
</UFM_Workflow>
```

Universal File Mover Status Monitor

- Designed to display the UFM Workflow status messages from the UFM status queue.
- Universal File Mover Status Monitor (UFM-SM) allows the user to view the status messages of an UFM Workflow from a centralized queue.
- Any UFM Workflow that has a non-zero “Exit Code” will be highlighted in red, so that the user can quickly identify problem Workflows.
- By default, status messages have a default expiry of 7 days.

Universal File Mover Status Monitor



Date / Time	Hostname	IP Address	OS	UserID	Exit Code	UFM Version	UFM Workflow
2012/10/18 16:32:04	cw-rl001	192.168.10.115	Windows XP	rlacroix	0	1.0.0	ufm_mqsend7.xml
2012/10/18 16:10:59	fc4x86-64	127.0.0.1	Linux	mqm	0	1.0.0	ufm_mqsend.xml
2012/10/18 16:09:54	u10-4	192.168.10.213	SunOS	tester	0	1.0.0	ufm_mqsend.xml
2012/10/18 16:09:28	cw-ux31	192.168.10.111	Windows 7	rlacroix	0	1.0.0	ufm_mqsend8.xml
2012/10/18 16:07:02	cw-ux31	192.168.10.111	Windows 7	rlacroix	0	1.0.0	ufm_mqsend8.xml
2012/10/18 16:06:56	fc4x86-64	127.0.0.1	Linux	mqm	0	1.0.0	ufm_mqsend.xml
2012/10/18 16:05:54	u10-4	192.168.10.213	SunOS	tester	0	1.0.0	ufm_mqsend.xml
2012/10/18 16:05:47	fc4x86-64	127.0.0.1	Linux	mqm	8	1.0.0	ufm_mqsend.xml
2012/10/18 16:04:27	u10-4	192.168.10.213	SunOS	tester	0	1.0.0	ufm_mqsend.xml
2012/10/18 16:02:54	cw-ux31	192.168.10.111	Windows 7	rlacroix	0	1.0.0	ufm_mqsend8.xml
2012/10/18 15:59:35	u10-4	192.168.10.213	SunOS	tester	0	1.0.0	ufm_mqsend.xml
2012/10/18 15:59:04	cw-ux31	192.168.10.111	Windows 7	rlacroix	8	1.0.0	ufm_mqsend8.xml
2012/10/18 15:57:37	u10-4	192.168.10.213	SunOS	tester	0	1.0.0	ufm_mqsend.xml
2012/10/18 13:39:20	cw-rl001	192.168.10.115	Windows XP	rlacroix	0	1.0.0	ufm_mqsend7.xml
2012/10/18 13:27:07	cw-rl001	192.168.10.115	Windows XP	rlacroix	0	1.0.0	ufm_mqsend7.xml
2012/10/17 20:06:36	u10-4	192.168.10.213	SunOS	mqm	0	1.0.0	ufm_mqsend.xml
2012/10/17 19:33:57	u10-4	192.168.10.213	SunOS	mqm	0	1.0.0	ufm_mqsend.xml
2012/10/17 19:09:38	u10-4	192.168.10.213	SunOS	mqm	0	1.0.0	ufm_mqsend.xml
2012/10/17 19:07:24	u10-4	192.168.10.213	SunOS	mqm	0	1.0.0	ufm_mqsend.xml
2012/10/17 19:01:00	u10-4	192.168.10.213	SunOS	mqm	0	1.0.0	ufm_mqsend.xml

Universal File Mover Status Monitor

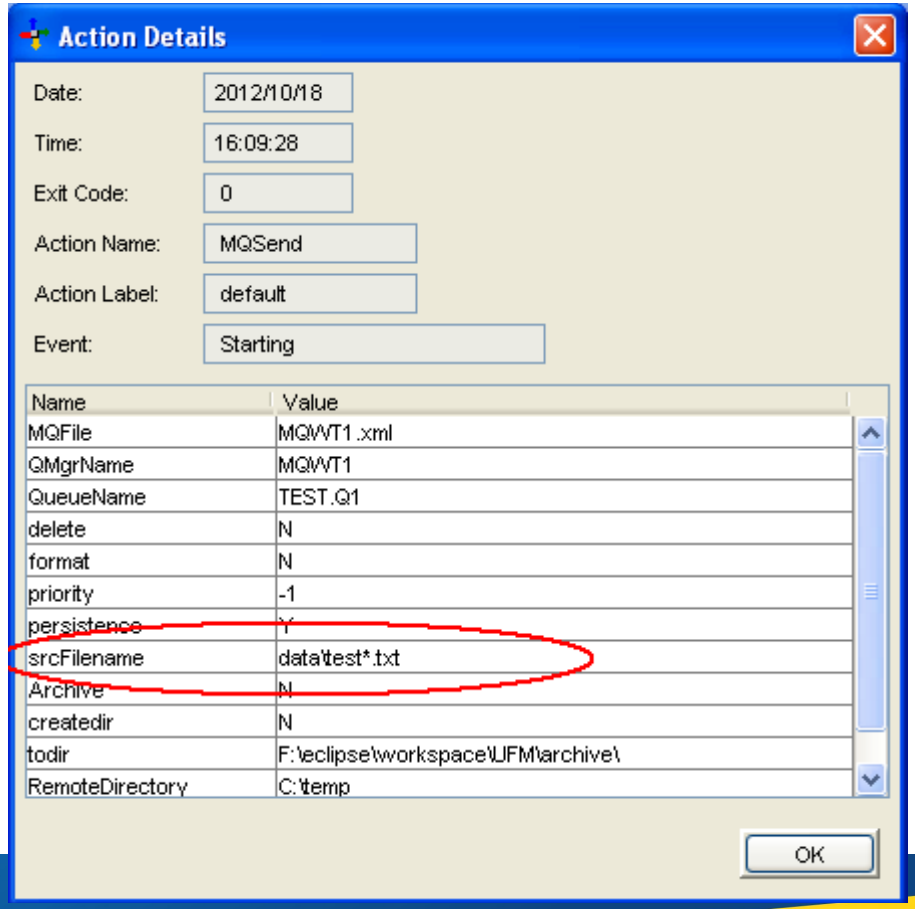
2012/10/18 16:09:28 : cw-ux31 : ufm_mqsend8.xml

File Help

Date / Time	Exit Code	Action	Label	Event	Parameters
2012/10/18 16:09:28	0	MQSend	default	Starting	MQFile=MQWT1.xml, QMgrName=MQWT1, QueueName=TEST.Q1, delete=N, format=N...
2012/10/18 16:09:31	0	MQSend	default	MQPut	MQFile=MQWT1.xml, QMgrName=MQWT1, QueueName=TEST.Q1, delete=N, format=N...
2012/10/18 16:09:31	0	MQSend	default	MQPut	MQFile=MQWT1.xml, QMgrName=MQWT1, QueueName=TEST.Q1, delete=N, format=N...
2012/10/18 16:09:31	0	MQSend	default	MQPut	MQFile=MQWT1.xml, QMgrName=MQWT1, QueueName=TEST.Q1, delete=N, format=N...
2012/10/18 16:09:31	0	MQSend	default	MQPut	MQFile=MQWT1.xml, QMgrName=MQWT1, QueueName=TEST.Q1, delete=N, format=N...
2012/10/18 16:09:31	0	MQSend	default	MQPut	MQFile=MQWT1.xml, QMgrName=MQWT1, QueueName=TEST.Q1, delete=N, format=N...
2012/10/18 16:09:31	0	MQSend	default	MQPut	MQFile=MQWT1.xml, QMgrName=MQWT1, QueueName=TEST.Q1, delete=N, format=N...
2012/10/18 16:09:31	0			Exiting	

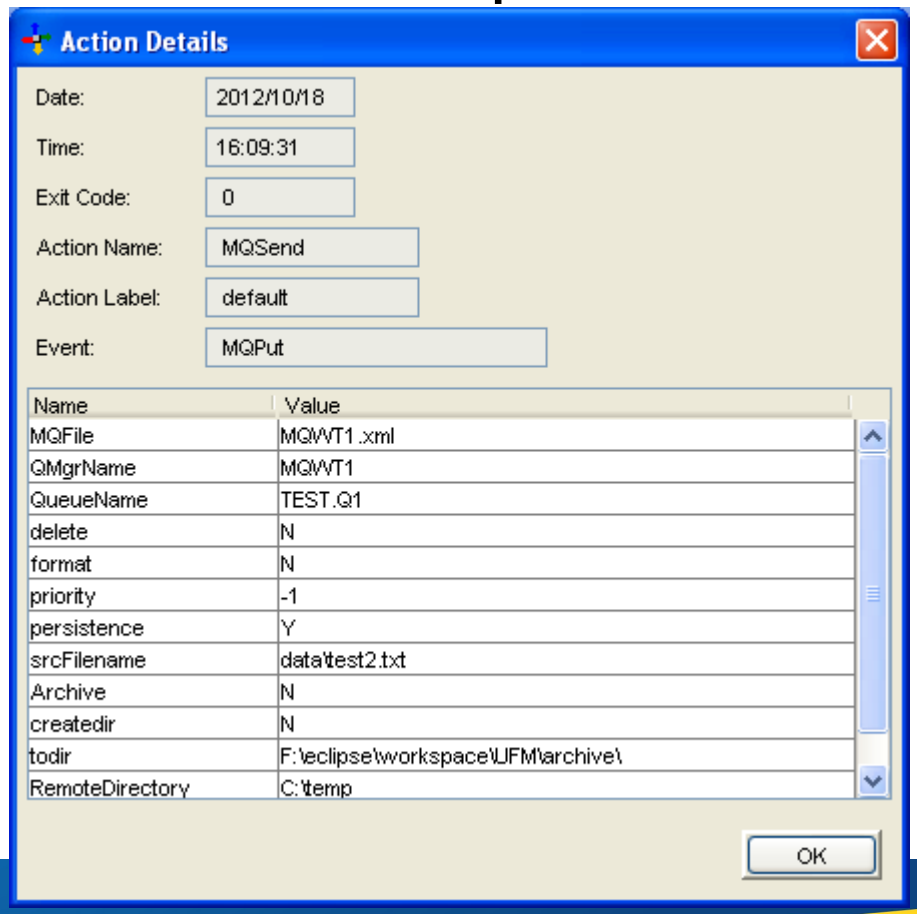
Universal File Mover Status Monitor

- The first row with the Event of “Starting” shows the attributes of the MQSend Action **before** they are processed.



Universal File Mover Status Monitor

- Selecting a row with an Event of “MQPut” will show the exact details as UFM performed the action.



MQSeries.net FTE Question

“i'm looking for a way to create a mq fte client agent on MSCS(Microsoft Cluster server) in active passive architecture, without installing MQ server on the same machines (MSCS machines)”

UFM can run in bindings or client mode and can be configured to be used with any hardware clustering software – it is just another application.

MQSeries.net FTE Question

“Sending file "filename.txt" from Windows to Linux. Can't use the text transfer option because contents may not be converted. But file must be converted with dos2unix to handle CR/LF.”

UFM can invoke any external program via the Execute Action.

MQSeries.net FTE Question

“I want to " Transfer a binary file from A to B with Encryption and Decryption " can you help me how can we encrypt and decrypt the messages in FTE.”

The MQSend and MQReceive have built encryption and decryption.

Questions & Answers