Exchange for WebSphere MQ – Support tools Version 1.5

23rd February, 2003

Arjan van Vught
e-business Hosting Services
Watsonweg 2
1423 ND Uithoorn
The Netherlands

avv@nl.ibm.com

Take Note!

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

Sixth Edition, February 2003

This edition applies to Version 1.5 of *Exchange for WebSphere MQ – Support tools* and to all subsequent releases and modifications unless otherwise indicated in new editions.

Table of Contents

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 be 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 MQ
- Exchange for WebSphere MQ
- AIX
- AS/400, OS/400

The following terms are trademarks of other companies:

- Windows NT, Visual Studio Microsoft Corporation
- Sun Solaris   Sun Corporation
- HP-UX   Hewlett-Packard Company
- Linux   Developed under the GNU General Public License

Acknowledgments

Summary of Amendments

Date	Changes
3 October 2000	Initial release
12 July 2001	AS/400 support added
23 July 2001	Sun Solaris (SunOS 5.7) support added for putedi and dspmailbox
1 September 2001	Linux (on Intel) support added for putedi and dspmailbox
3 September 2002	MQ JMS putedi GUI (with JNDI support) added
	Updated Exchange for WebSphere MQ URL
	New brand names: WebSphere MQ, Exchange for WebSphere MQ
23 February 2003	Updated MQ JMS putedi GUI (with JNDI support)
	AIX support added for xWMQexit

Preface

Description

This SupportPac describes a set of support tools that may be helpful when using Exchange for WebSphere MQ. The support tools are:

- Send and receive files using Expedite profiles
- Display Information Exchange alias Q entries with, if included, the MQSIE header contents
- Messages exit for setting the MsgDesc.ReplyToQ with Information Exchange alias Q for all messages for WebSphere MQ / Information Exchange Bridge queuemanager
- Put sample for Information Exchange messages (including MQ JMS GUI)

The source code is provided, and so these programs may be used by an installation to provide a basis for their own programs.

Possible uses

This SupportPac is designed for people who:

- Want to migrate from EDI file transfer using Expedite to native WebSphere MQ
- Will be writing B2B applications for Exchange for WebSphere MQ
- Are interested in the Design of Message Queuing Applications

Prerequisites

A platform supported by Exchange for WebSphere MQ.

Bibliography

- *IBM MQSeries Services Administration and Application Development,* IBM Corporation. GC34-2344
- IBM MQSeries Services Administration and Application Development Guide, IBM Corporation. GC34-2344
- MQSeries Application Programming Reference, IBM Corporation. SC33-1673

Exchange for WebSphere MQ – Support tools

Put sample for IE messages

Description

Using this put sample you can easily send plain text Information Exchange messages from WebSphere MQ, using Exchange for WebSphere MQ, to any Information Exchange trading-partner address.

Required parameters

```
NT, AIX, Sun Solaris, Linux
putedi -m LocalQMgrName -tq TargetQName -tm TargetQMgrName -q ReplyToQName –qm
ReplyToQMgrName [-debug]

AS/400
                          MO91: PUTEDI (QPUTEDI)

 Type choices, press Enter.

 Local QMgr Name  . . . . . . . .   LocalQMgrName

 Target IE address  . . . . . . .   TargetQName

 IE QMgr Name MQSeries Services     MQ31

 ReplyToQ Name  . . . . . . . . .   ReplyToQName

 ReplyToQMgr Name . . . . . . . .   ReplyToQMgrName
```
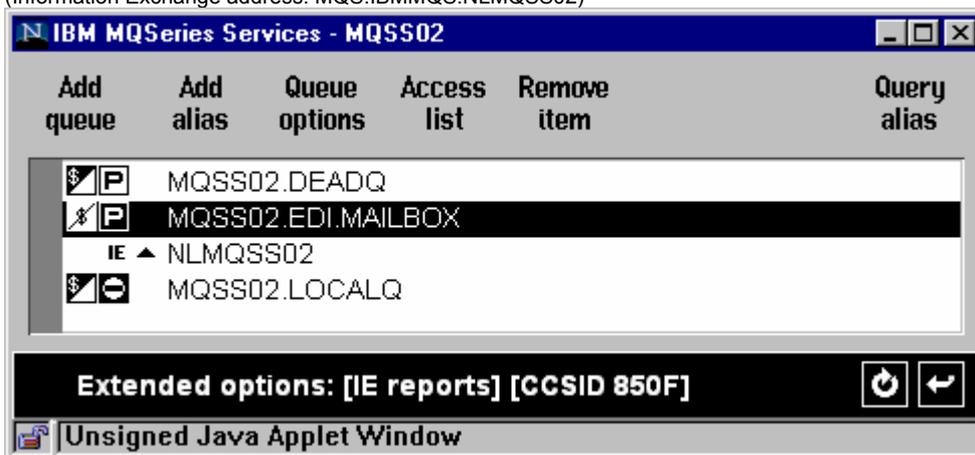
The reply-to queue must have an Information Exchange alias.

Example
```
putedi -m MQSS02 -tq EUR.NLEBCC.NLEBC001 -tm MQ31 -q MQSS02.EDI.MAILBOX
```

Local queue MQSS02.EDI.MAILBOX has an IE Alias: NLMQSS02
(Information Exchange address: MQS.IBMMQS.NLMQSS02)

Queuemanager alias definition for MQ31 (MQSeries / Information Exchange Bridge):

```
DEFINE QREMOTE(MQ31) +
       RQMNAME(MQ31) +
       XMITQ(MQX1)   +
       REPLACE
```

For more information about WebSphere MQ <-> Information Exchange addressing, please refer to PDF file: xWMQIE.pdf

Java version

Description

Using this MQ JMS put sample you can easily send plain text Information Exchange messages from WebSphere MQ, using Exchange for WebSphere MQ, to any Information Exchange trading-partner address. This sample supports both run-time WebSphere MQ and JNDI configuration settings.

Required parameters
```
jPutEDI [LocalQMgrName]
```

Example
```
jPutEDI MQSS02
```

Message Exit

Description

Setting MsgDesc.ReplyToQ with Information Exchange alias queue name for all messages for destination MQ31 (xWMQ / Information Exchange Bridge) queuemanager

With most standard products (e.g. WebSphere MQ product samples like amqsput), it is not possible to set the MsgDesc.ReplyToQ field in the MQMD of the WebSphere MQ message. The MsgDesc.ReplyToQ must be filled in when sending messages to Information Exchange. This message exit checks the destination queuemanager, and if this equals MQ31 and the MsgDesc.ReplyToQ is empty, the MsgDesc.ReplyToQ will set by the MSGDATA entry from the channel definition.

Example channel configuration with message exit

```
DEFINE CHANNEL('MQSS02/MQSS/00') CHLTYPE(SDR)  +
       TRPTYPE(TCP)                            +
       CONNAME('32.82.12.36')                  +
       XMITQ(MQX1)                             +
       MSGEXIT('xWMQexit(xWMQexit)')           +
       MSGDATA(MQSS02.EDI.MAILBOX)
```

Sample compile options for Windows NT with MS C/C++ Compiler:
```
cl /DNT /LD xWMQexit.c
```

Display messages on IE alias queue

Description

Display Information Exchange alias WebSphere MQ entries with, if included, the MQSIE header contents

Sample profile

```
qmgr=MQSS02
Replyq=MQSS02.EDI.MAILBOX
dismqsie=Y
logmqsie=N
loglevel=1
logdir=c:\temp
# Sample profile for displaying messages on the IE alias queue
```

Sample output

```
Msg #1 from EUR.NLEBCD.NLEBC123, 212 bytes [MQSIE   ], 25:08:2000, 11:40:14
        FROM        EUR.NLEBCC.NLEBC001
        CLASS       FFMSG002
        MSGNAME
        MSGSEQNO
        DATATYPE    T
        UNIQUEID    13708725
        CONTROLNUM
        SEND
        SENDQUAL
        RECEIVER
        RECVQUAL
        DATE        000825
        TIME        113708
        RECLEN      00079
        RECFM       F
        RECDLM      N

Queue depth for MQSS01.EDI.MAILBOX : 1 message
```

Profile

The default profile is 'profile'.

The profile must adhere to the following layout:

```
qmgr=<queuemanager name (blank=default)>
replyq=<local queue name with IE alias>
dismqsie=<Y / N>
logmqsie=<Y / N>
logdir=<full path to log directory>
loglevel=<1 / 2 / 3>
# Sample profile for displaying messages on the IE alias queue
```
Lines starting with '#' are ignored. Entries not meaningful to the program are ignored

loglevel choices: 1 / 2 / 3

1: MQSeries call failed
2 : MQSeries call failed / Warning (partial completion)
3 : MQSeries call failed / Warning (partial completion) / Successful completion

First step migrating from Expedite to Exchange for WebSphere MQ

Note:
- Exchange for WebSphere MQ does not currently support direct connections from WebSphere MQ Clients.
- When migration from Expedite on DOS or Windows 3x/9x to Exchange for WebSphere MQ, you have to link the programs with the client library.

Send files using Expedite profiles

Description

This program reads the BASEIN[1] file SEND entries and sends the files using Exchange for WebSphere MQ to the EDI trading partners. Send completion messages are written to BASEOUT[2]. The source code included can be used as a framework for a full functional migration program.

Sample profile:

```
qmgr=MQSS01
replyq=MQSS01.EDI.MAILBOX
expedite=c:\expedite
loglevel=1
logdir=c:\temp
# sample profile for migrating from Expedite to Exchange for WebSphere MQ
```

Receive files using Expedite profiles

Description

This program read the BASEIN[3] file RECEIVE entries and receives the files using Exchange for WebSphere MQ from the EDI trading partners. Receive completion messages are written to BASEOUT[4]. The source code included can be used as a framework for a full functional migration program.

Sample profile:

```
qmgr=MQSS01
replyq=MQSS01.EDI.MAILBOX
expedite=c:\expedite
loglevel=1
logdir=c:\temp
# sample profile for migrating from Expedite to Exchange for WebSphere MQ
```

---

[1] On NT, AIX basein.msg; On AS/400 INMSG

[2] On NT, AIX baseout.msg; On AS/400 OUTMSG

[3] On NT, AIX basein.msg; On AS/400 INMSG

[4] On NT, AIX baseout.msg; On AS/400 OUTMSG

Log files

The support tool uses the directory specified in the profile to store logging information.

The format of the log files is defined by the following:

<record> ::= <statement> - <statement> - <statement> - <statement>
<statement> ::= {any printable character}
<SEVERITY> ::= {INFO}{WARNING}{ERROR}
<MQI API> ::=
{MQCONN}{MQDISC}{MQOPEN}{MQCLOSE}{MQGET}{MQPUT}{MQPUT1}{MQCMIT}{MQBACK}

A record in the **MQILOG** file always contains the following:

yyyy/mm/dd - hh:mm:ss – [{<SEVERITY>}] - {<statement>}

When available, non-zero WebSphere MQ return codes are logged as:

yyyy/mm/dd - hh:mm:ss - [{<SEVERITY>}] - [{<MQI API>}] ended with reason code <value> :
<statement>

The following events are logged:

I.   startup with received parameters
II.   shutdown with reason
III.   warnings and errors received by the application from MQM
IV.   warnings and errors received by the application from the operating system.

The occurrence of errors can be detected by monitoring the log files. Also the WebSphere MQ logs should be monitored for communication errors e.g. WebSphere MQ channel down.

Package contents

```
\MQSeriesServices\mo91

<DIR>          putedi
<DIR>          msgExit
<DIR>          src
<DIR>          expedite
<DIR>          doc
<DIR>          dspmailbox
<DIR>          NT
<DIR>          AIX
<DIR>          AS400
<DIR>          SUN
<DIR>          LINUX86


\MQSeriesServices\mo91\putedi

putedi.c
export.jar
jPutEdi.bat

\MQSeriesServices\mo91\msgExit

xWMQexit.c

\MQSeriesServices\mo91\src

mqsie.c
mqsie.h
getmailboxindex.c
getmailboxindex.h
keyword.c
keyword.h

\MQSeriesServices\mo91\expedite

expedite.c
expedite.h
mqrecv.c
mqrecv.h
mqsend.c
mqsend.h
profile.c
profile.h

\MQSeriesServices\mo91\doc

xWMQIE.pdf
mo91.pdf

\MQSeriesServices\mo91\dspmailbox

dspmailbox.c
dspmailbox.h
profile.c
profile.h
```

```
\MQSeriesServices\mo91\NT

xWMQexit.dll
dspmailbox.exe
mqsend.exe
mqsend.exe
putedi.exe

\MQSeriesServices\mo91\AIX

xWMQexit
dspmailbox
mqsend
mqsend
putedi

\MQSeriesServices\mo91\AS400

MO91

\MQSeriesServices\mo91\SUN

dspmailbox
putedi

\MQSeriesServices\mo91\LINUX86

dspmailbox
putedi
```

Directory structure common functions

The functions in the files described below are not specific to this SupportPac. Therefore these functions can be used as a common library for writing WebSphere MQ programs.

```
\src

defs.h
file_util.h
file_util.c
mq_util.c
mq_util.h
str_util.c
str_util.h
```

Exchange for WebSphere MQ utils

Getmailboxindex.c

Functions

```
PMQMSGINFO *GetMailBoxIndex( char *LogDir,
                             MQHCONN Hconn,
                             MQCHAR48 QName,
                             MQLONG *Qdepth );


void DisplayMailBoxIndex( char *LogDir,
                          PMQMSGINFO *RecvMsgInfo,
                          char *DisMqsie,
                          char *LogMqsie,
                          MQLONG Qdepth );


void FreeMQMSGINFO( PMQMSGINFO *RecvMsgInfo,
                    MQLONG Qdepth );
```

Headers used

```
/* MQSeries product include file                         */
#include <cmqc.h>
/* Utils include files                                   */
#include "../../../src/file_util.h"   /* file functions   */
#include "../../../src/mq_util.h"     /* MQSeries functions */
#include "../../../src/str_util.h"    /* string functions */
/* MQSeries Services include files                       */
#include "../src/mqsie.h"
```

Keyword.c

Functions

```
char *GetKeyword( const char *Keyword,
                  char *s,
                  int len,
                  const char *b,
                  long blength );


char *FindKeyWord( const char *p,
                   const long plength,
                   const char *KeyWord);
```

Header used

```
/* Utils include files                                   */
#include "../../../src/str_util.h"    /* string functions */
```

Mqsie.c

Functions

```
PMQSIE GetMQSIE( PMQSIE pMqsie,
                 PMQBYTE Buffer,
                 MQLONG BufferLength );


PMQSIE SetMQSIE( PMQSIE pMqsie,
                 char *Keyword,
```

```
                    char *KeywordValue );


char *CreateMQSIEheader(PMQSIE pMqsie);


void DisplayMQSIE(PMQSIE pMqsie);


char LogMQSIE(char *LogDir, PMQSIE pMqsie);
```

Headers used

```
/* MQSeries product include file                        */
#include <cmqc.h>
/* Utils include files                                  */
#include "../../../src/str_util.h"    /* string functions    */
#include "../../../src/file_util.h"   /* file functions      */
/* MQSeries Services include files                      */
#include "../src/keyword.h"
```

Expedite utils

Expedite.c

Functions

```
PIESENDINFO GetSendInfo( PIESENDINFO SendInfo,
                         char *buffer,
                         long bufferlength );


PIERECVINFO GetRecvInfo( PIERECVINFO RecvInfo,
                         char *buffer,
                         long bufferlength );


int GetExpediteBaseFiles( PINITFILE InitFile,
                          char *BaseInFileName,
                          FILE **pBaseIn,
                          char *BaseOutFileName,
                          FILE **pBaseOut );


int ReceiveFile( char *LogDir,
                 MQHCONN Hconn,
                 MQCHAR48 QName,
                 MQLONG BufferLength,
                 char *FileName,
                 MQBYTE24 MsgId,
                 MQBYTE24 CorrelId );


int SendFile( char *LogDir,
              MQHCONN Hconn,
              MQCHAR48 QName,
              MQCHAR48 QMgrName,
              MQCHAR48 ReplyToQ,
              MQCHAR8 Format,
              char *FileName,
              PMQBYTE Header,
              MQLONG HeaderLength,
              PMQBYTE Trailer,
              MQLONG TrailerLength );
```

Header used

```
/* MQSeries product include file                          */
#include <cmqc.h>
/* Utils include files                                    */
#include "../../../src/file_util.h"   /* file functions    */
#include "../../../src/mq_util.h"     /* MQSeries functions */
#include "../../../src/str_util.h"    /* string functions   */
/* MQSeries Services include files                        */
#include "../src/keyword.h"
```

AS/400 installation instructions

Complete the following steps:

1. Create a destination save file on your AS/400. If you do not, the restore process will not recognize the files as being in the SAVF format.

    1. Enter **CRTSAVF** and then press F4 (prompt)

    2. Enter the save file name, the library name, and press Enter.

        Note: The LIBRARY name must be an existing library in your system. If it is not, use the CRTLIB command to create it.
        For example: **CRTLIB MO91**

3. Download the file from the Web to a PC and unzip the file. For example, if using pkzunip enter:
    pkunzip MO91

4. Use FTP to send the file from the PC to your AS/400, as follows:

    1. Open an FTP session from the PC to your AS/400 and use the FTP binary subcommand to set the mode to BINARY.

    2. Make sure the current library is the library where the SAVF file was created. If it is not, use thr FTP cd subcommand (cd MO91) to change the current library.

        Note: If you use the **dir** command to list a library, the mode is changed automatically to ASCII, so if you set binary mode and then entered **dir**, the mode is reset; therefore, issue the binary subcommand once more.

    3. Use the FTP put subcommand to transfer the file

        put MO91

5. Restore the file by RSTOBJ OBJ(*ALL) SAVLIB(MO91) DEV(*SAVF) SAVF(library/MO91)

Notes:
a) The fully qualified name of the save file you created in step 1 is library/MO91.
b) The SAVLIB parameter is the name of the library where the files were originally saved, and these parameters are required.
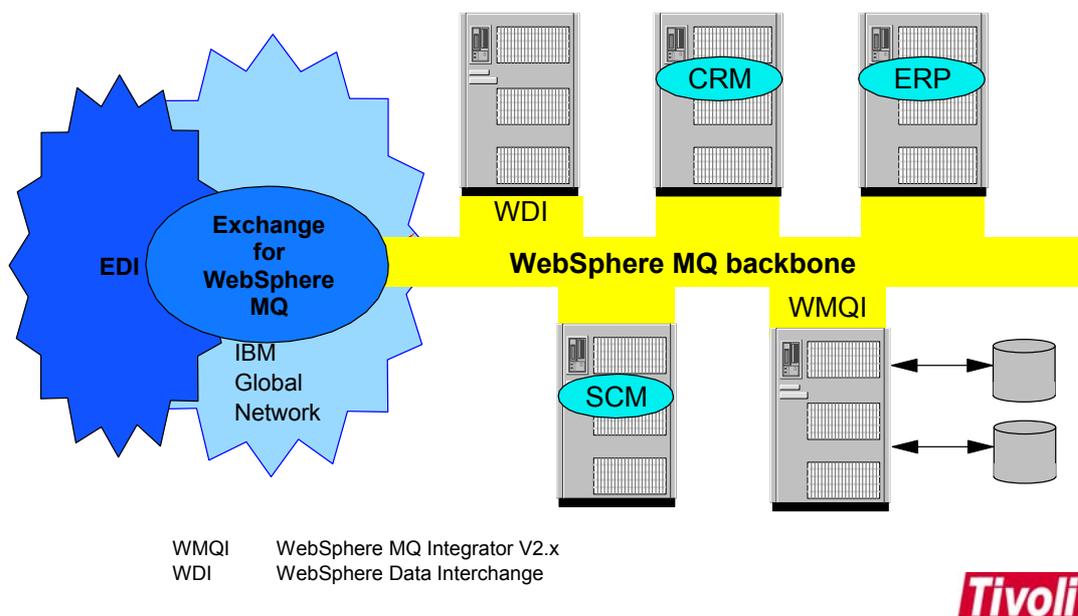
Appendix A: Exchange for WebSphere MQ

IBM Exchange for WebSphere MQ provides the network, protocol conversion, message routing, and storage to support secured transaction processing among WebSphere MQ and EDI trading partners. A single wide area networking or remote access connection to the service eliminates the need to connect individually with trading partners, while providing a security buffer from them. Transactions can be routed immediately or saved for later retrieval. A Java-based administration tool is provided for enhanced management of queues and trading partner definitions.

For further information: http://edi.services.ibm.com/mqseries/spec_sheets.shtml

Traditional EDI with a flat file gateway

WMQI	WebSphere MQ Integrator V2.x
WDI	WebSphere Data Interchange
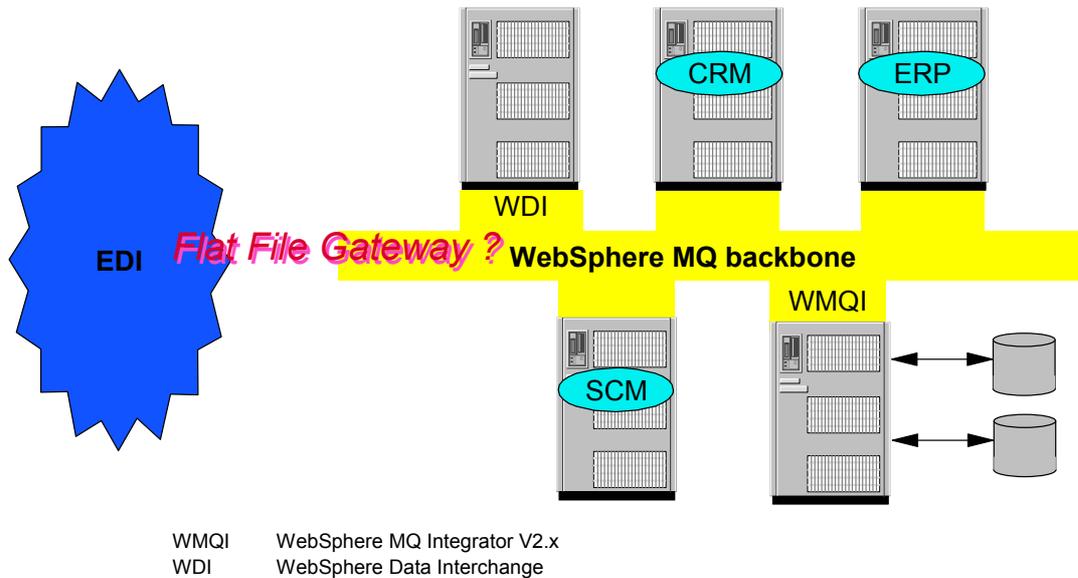
Connecting to EDI using Exchange for WebSphere MQ and a WebSphere MQ backbone

WMQI	WebSphere MQ Integrator V2.x
WDI	WebSphere Data Interchange

-- **End of Document** --