MINNESOTA
JUDICIAL BRANCH

Court Integration Services

Integration Services

Setup Procedures

Consumer Documentation

TABLE OF CONTENTS

1. DOCUMENT REVISION HISTORY 3

2. PREFACE 3

3. ADMINISTRATIVE SETUP 3

3.1. Submit Information Access Request Form 3

3.2. Submit Communication Channel Request Form.. Error! Bookmark not defined.

4. TECHNICAL SETUP 3

4.1. MQ-Series Setup 3

4.1.1. Networking 4
4.1.2. SSL 4
4.1.3. Channel Names 4
4.1.4. Testing 4

4.2. MQ-Series Client Setup 4

4.2.1. Networking 5
4.2.2. SSL 5
4.2.3. Channel Configuration 6
4.2.4. Testing 6

4.3. HTTP/Web Services 6

4.3.1. Networking 6
4.3.2. Message Receiver Web Service 7

5. INTEGRATION CONFIGURATION 8

1. Document Revision History

Date	Author	Revision Highlights
1/21/05	P. McNair	Created, Version 1.0
4/8/06	T. Buchholz	Update with SSL and Client
9/16/2006	T. Buchholz	Update for MQ V6 clients
2/21/2007	T. Buchholz	Fixed some directions for MQ Client setup.
4/24/2007	T. Buchholz	Add information on HTTP
6/9/2009	T. Buchholz	Add in IP addresses.

2. Preface

This document explains steps required to setup communication channels with the Court's integration services. Integration services can be accessed via one of the following messaging technologies:

- IBM MQ-Series
- IBM MQ-Series Client
- HTTP/Web Services

See the document Integration Services Technical Overview for additional information on how to use Integration Services.

3. Administrative Setup

3.1. Submit Information Access Request Form

Submit an Integration Services Agreement and Request Form. See the "Request Access" area of the Integration Services Website http://www.mncourts.gov/is. You will be provided a user id and password to be used on all Integration Service Request

4. Technical Setup

4.1. MQ-Series Setup

This section describes the technical setup steps required if you choose to use IBM MQ-Series. If you are using this communications mechanism it is assumed that you have IBM MQ-Series server installed, and have personnel that are trained in the administration and programming of MQ-Series.

Setting up communication when using MQ-Series involves setting up sender and receiver channels between our 2 queue managers. If multiple environments/queue managers are involved the sender and receiver channels will need to be setup between each. Your staff that supports MQ Series will need to be in contact with technical staff at the court to complete the following:

4.1.1. Networking

The courts network staff will need to write an access list rule to allow your queue manager to communicate with the courts queue manager (IP address and Port). Your network staff will likely have to do the same. The IP addresses and ports are provided on the channel request form. The ports will depend on which port your queue manager is listening on.

4.1.2. SSL

The courts require SSL to be used on all channels with external business partners. You will be provided with a certificate for the courts queue manager, and will be required to provide a certificate for your queue manager to the courts. Generally we have used TRIPPLE_DES_SHA_US as the SSL Cipher Spec.

4.1.3. Channel Names

The Courts standard for naming channels is the following:

For Sender channels the name is the local queue manager name, followed by a period, followed by the remote queue manager name.

Ex:

MSCJB01D.*REQUESTERQM* (courts sender channel)
REQUESTERQM.MSCJB01D (requester's sender channel)

For Receiver channels the name is the remote queue manager name, followed by a period, followed by the local queue manager name.

Ex:

REQUESTERQM.MSCJB01D (courts receiver channel)
MSCJB01D.*REQUESTERQM* (requester's receiver channel)

We need to agree on the channel names.

The courts standard for naming transmission queues is to name them the same as the name of the remote queue manager.

4.1.4. Testing

Once the channels and transmission queues are set up they can be tested by doing a MQ Series Ping. This is done by right clicking on the sender channel and selecting "Ping". A message will be displayed that gives the results of this test. The normal command line "Ping" will most likely not work.

4.2. MQ-Series Client Setup

This section describes the technical setup steps required if you choose to use IBM MQ-Series Client. If you are using this communications mechanism it is assumed that you have IBM MQ-Series Client software installed, and have personnel that are trained in the administration of and programming using MQ-Series clients. The MQ client can be obtained using the following link:

http://www-01.ibm.com/software/integration/wmq/clients/

Note: If you are going to be doing your development using .Net you need to make sure that the program gactuil.exe can be accessed from a command prompt using the path system environment variable. This enables the client install to register certain libraries into the global assembly cache which makes them available to .Net programs that access queues.

4.2.1. Networking

The courts network staff will need to write an access list rule to allow your client machine to communicate with the courts queue manager (IP address and Port).

4.2.2. SSL

The courts require SSL to be used on all communications with external business partners. You will be provided with a certificate for the courts queue manager, and will be required to configure a SSL key store on your client machine. The following steps can be used to do this:

1. Download the Client SSL Configuration (zip) file from the Integration Services technical overview and setup procedures page. Export its contents into a secure folder on your server. You should have been provided with the password for the zip file. If not contact the courts Integration Services technical contact.
2. Configure the location of the certificate store. To do this set up a system environment variable named MQSSLKEYR with a value that is a path to a key repository file. An example value is “C:\Program Files\IBM\WebSphere MQ\ssl\key” where “key” is the name of the key repository file.
3. Use the IBM Key Management utility to create a key repository (unless you already have one) at the location indicated by the MQSSLKEYR environment variable. In the example above it would be named KEY.kdb and would be located in the folder: C:\Program Files\IBM\WebSphere MQ\ssl. You will be asked for a password. This is your password for your key repository. Make one up and keep track of it for later maintenance of your key repository. Make sure that you “stash” the password so that it is available to the client software.
4. Use the IBM Key Management utility to import the certificate(s) provided by the court into this key repository. To do this do the following:
    a. Open the key repository
    b. Select Personal Certificates
    c. Click Import
    d. Select Import
    e. Change key file type to PKCS12
    f. Browse to the location of the certificate files and select one

g. Click OK
h. Enter the password (provided by the court Integration Services technical contact)

4.2.3. Channel Configuration

You will be provided with a client channel table file that contains information on the client channels for the courts queue managers. This file needs to be saved somewhere on the clients file system. Two system environment variables then need to be set up to tell MQ-Series where this file is located.

Environment Variable Name	Value
MQCHLTAB	AMQCLCHL.TAB
MQCHLLIB	Path to where the AMQCLCHL.TAB file is located. Ex: C:\Program Files\IBM\WebSphere MQ

4.2.4. Testing

Once the client has been configured you can test connectivity from the client by performing a get from a test queue. The queue ADMIN_TESTCONNECTION was set up for the purpose. Execute the following from the command line:

```
amqsgetc ADMIN_TESTCONNECTION MSCJB01D
```

Substitute the appropriate queue manager name for the 2nd parameter. You should see a message stating “Sample AMQSGET0 start”. It should pause for around 10 seconds and then respond with “No more messages” and “Sample AMQSGET0 End”. As long as you don’t see any error messages the configuration is ok.

4.3. HTTP/Web Services

4.3.1. Networking

If you are going to be sending messages to the court, such as with queries and submission messages, the courts network staff will need to write an access list rule to allow the server you’re your application is running on to communicate with the courts web server. If your network blocks outbound then you will need your network to allow outbound requests to the following IP addresses:

For Production:	156.98.54.40
For Development and/or QA:	156.98.54.30

If the court will be sending you messages, such as with notification messages and e-file responses, your network staff may need to add an access list rule to allow the courts integration broker to communicate with your web server. The IP addresses that should be allowed in are:

For Production:	156.98.54.34 and 156.98.54.35
For Development and/or QA:	156.98.170.185

4.3.2. Message Receiver Web Service

If the court will be sending you messages, such as with notification messages and e-file responses, you will need to host a Web Service that the courts integration broker can send messages to. Note: this is not necessary if you are only going to be using the query services and will be calling them synchronously. This service needs to have the following characteristics:

- Conform to the Web Service standards as described in the document Integration Services Overview. This includes standards such as soap, ws-addressing and ws-security.
- Be hosted on a web server that keeps its system clock accurate. Messages can be rejected if the clocks on the submitting and receiving system are too greatly different.
- Provide for SSL encryption through the use of a digital certificate. If your certificate is not from a trusted certificate site you will have to provide it to the courts network staff so they can import it into the trusted sites on the integration broker.
- Optionally authenticate the submitter of the messages using the provided UsernameToken. If configured to do so messages will contain a wsse:UsernameToken element containing an account and password which can be used to authenticate that messages are coming from a trusted source.
- Provide a “document” style function for each type of message that is going to be received. These functions need to have the following characteristics:
    - Be assigned a Soap Action value that is associated with the message being received. See documentation for each service for what these values should be.
    - It should store the message somewhere for later processing.
    - Return a Boolean value of “true” if the function completes successfully. Note: Successfully only means that the message was received and persisted somewhere in the receivers’ environment.
    - Return a soap fault if it is unable to store the message. Within the fault you should provide a reason for why the message could not be stored.

If an error occurs while sending a message (sending program receives a soap fault back from your Web Service) it will attempt to resend it a few times after waiting a number of minutes. The amount of time it waits depends on how many times it has failed. The following shows what happens after each failed attempt:

Failure	Action
1	Wait one minute and try sending again
2	Wait 5 minutes and try sending again

3	Wait 10 minutes and try sending again
4	Wait 20 minutes and try sending again
5	Log this message as having failed and go on to the next message. This message can be recycled at a later time.

5. Integration Configuration

Please review the technical documentation for the various integration services to identify any additional requirements for setting up access to that service.