

IBM WebSphere Business Connection



Using Business Connection APIs

Version 1.1.0

Note!

Before using this information and the product it supports, be sure to read the general information under “Notices” on page 13.

Second Edition (October 2002)

This edition applies to Version 1, Release 1, Modification 0, of *IBM® WebSphere® Business Connection* (5724-D26) and to all subsequent releases and modifications until otherwise indicated in new editions.

IBM welcomes your comments. You can send them to the following address:

IBM Canada Ltd. Laboratory
Information Development
8200 Warden Avenue
Markham, Ontario, Canada L6G 1C7

Include the title and order number of this book, and the page number or topic related to your comment.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 2002. All rights reserved.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Using the Application Interfaces	1	Logging requests for a Web service	8
The Logging Client	3	The WebSphere Member Services API	
Types of logs	3	Wrapper.	9
Configuring the System Admin console to display logs	4	Trading partner profile	9
Configuring a queue exit	5	Methods	10
Creating and configuring a target queue	6	Sample code	10
Updating the configuration table	6	Notices	13
Creating the target queue	6	Programming interface information	14
Adding resources to the EAR file	7	Trademarks and service marks	15
Regenerating the EAR file	8		

Using the Application Interfaces

The IBM^(R) WebSphere^(R) Business Connection and WebSphere Business Connection Enterprise Edition provide a Logging Client (as part of the Solution Manager component) that you can use to set up logging for your own Web service. Similarly, they provide a WebSphere Member Services API Wrapper that lets you gain access to trading-partner profile information. This document describes how to use these programming interfaces. It is intended to be used in conjunction with the Logging Client Javadoc and the WebSphere Member Services API Wrapper Javadoc.

The Logging Client

If you are writing services for a Business Connection system, you will want to add logging to your services so that you can view and track your business processes. The Document Exchange and Registration and Provisioning services already include logging, and you can follow a similar pattern to log events in your application.

This section:

- Describes the types of logs you can generate
- Describes how to configure the System Admin console so your logs can be displayed
- Points you to the Javadoc for the Logging Client API
- Provides an example of an exception exit routine
- Shows you how to enable audit logging for a Web service

Types of logs

The Solution Manager uses three types of logs:

- Audit
- Business process
- Exception

An *audit log* is generally used for data that you will want to keep for an extended period of time. For example, you might want to save the time a purchase order was received and the contents of the purchase order in case there is any contention about the order.

A *business process log* is generally used for data that can be deleted after a certain period of time. For example, while you might want to keep a record (in the audit log) of when the purchase order was received, you might not want to know when it entered each process state. The movement of the purchase order through the various stages of processing would be kept in the business process log.

An *exception log* is used for error or unexpected conditions. You might want to log the data (when those conditions occur) as well as provide an exception-handling or compensation routine.

Your means of generating and sending a message from your application is via the Logging Client. The Logging Client is an API that you use to:

- Create a specific type of message content
- Create a specific type of message header
- Add the message payload
- Send the message

The API provides interfaces to the three types of logs described earlier in this section (audit, business process, and exception). Each of the logs has its own queues, and your messages are transported to the logs via these queues.

The message schema is composed of a header and a payload. The header contains the information about what is being logged, when to log, and how to log. The body contains data specific to the log.

The classes, methods, fields, and other details needed to write the messages for the logs are provided in the Javadoc for the Logging Client. An example showing the use of the API to generate a queue exit is shown in “Configuring a queue exit” on page 5.

Configuring the System Admin console to display logs

To view the logs for the Registration and Provisioning or Document Exchange services, you display the WebSphere Business Connection System Admin Console and click **Log Viewer > Solution Log Viewer**. You specify the service whose logs you want to view by clicking it from the **Select Application** list on the Solution Log Viewer page.

Each entry on the page represents an instance of a process and is displayed as shown in the following example:

Unique ID	Event	User	Event Time Stamp	Application ID	Correlation ID
34b9c7..	STARTED	null	2002-09-0509:31:56:968000	DocumentExchange	docsend001

If you want to be able to display logging for your service (so that the service is added to the **Select Application** list and that the logs can be viewed on the Solution Log Viewer page), do the following:

1. Make sure that you use a homogeneous correlation ID (of the same type and structure) across instances of a process.
2. Define the application ID in the BCT_SRAConsole.properties file:
 - a. Change the directory to `<BCT_HOME>/properties`.
 - b. Open the BCT_SRAConsole.properties file for edit.
 - c. Locate the following line (which will be one continuous line in the file):
`SELECTAPPLICATIONFILTER=DocumentExchange,RegistrationAndProvisioning`
 - d. Add your application name to the list of services by typing a comma immediately following “Provisioning” and then entering the name of the service, with no space between the comma and the name.

For example, if your service is named “Scheduling,” the line looks like this:

```
SELECTAPPLICATIONFILTER=DocumentExchange,
RegistrationAndProvisioning,Scheduling
```

3. Make sure that, in the log, you use the application ID defined in the previous step.
4. Make sure each instance is started with `EventType=“STARTED”`.

By following these steps, you will be able to view the logs for your service on the Solution Log Viewer page. You will be able to filter the list of processes by the Application ID and then select a specific instance and get process information on that specific instance.

Configuring a queue exit

This section describes how you can configure a queue exit for exception messages.

First, look at the exception.dtd file, which describes the XML payload for Business Connection exception messages:

```
<!ELEMENT EXCEPTION (ExceptionState?)>
<!attlist EXCEPTION
ExceptionTypeID CDATA #REQUIRED
Description CDATA #IMPLIED>
<!element ExceptionState(ExceptionState Variable*)>
<!element ExceptionStateVariable EMPTY>
<!attlist ExceptionStateVariable
VariableID ID #REQUIRED VariableVALUE CDATA #IMPLIED>
```

The ExceptionTypeID attribute is used to determine if the received message has been configured to issue a message that starts an exception handler or compensation routine. The target queue, along with other configuration information, is defined in a DB2 table named BCMSM.BCMSMCONFIGURATION.

The configuration table has an EXCEPTION TYPE column, which contains the matching ExceptionTypeID, and a TARGETQUEUE column, which has the name of the target queue defined in the default queue manager for the system.

The following is an example of an XML payload. Notice that there is a reference to the exception.dtd file shown above.

```
<?xml version="1.0"?><!DOCTYPE EXCEPTION SYSTEM "exception.dtd"
<EXCEPTION>
  Description="Exception exit test"
  ExceptionTypeID="ExcRecovery">
    <exceptionstate>
      <exceptionstatevariable VariableID="ExcVariable VariableValue="
        EXCP1001"/>
      <exceptionstatevariable VariableID="RecoveryVariable"
        VariableValue="EXCP3002" />
    </exceptionstate>
  </EXCEPTION>
```

This XML payload was generated by the Solution Manager Logging Client API as follows:

```
BCMExceptionContent exceptionBody=new BCMExceptionContent();
exceptionBody.setExceptionDescription("Exception exit test");
exceptionBody.setExceptionType("ExcRecovery");
exceptionBody.setExceptionState("ExcVariable", "EXCP1001");
exceptionBody.setExceptionState("RecoveryVariable", "EXCP3002");
BCMLog bcmLogTest = new BCMLog("User_Registration","WAS");
String body = exceptionBody.generateString();
//There are methods to create a message and send it; however, this
//sample shows the set of parameters that can be added by the user.
//The rest are defaulted.
String bodyCategory="BCM";
String bodyType="ExceptionLog";
boolean transactionRequired=false,
_bcmMsg = new BCMMessage();
_bcmMsg.setBodyData(body);
_bcmMsg.setBodyCategory(bodyCategory);
_bcmMsg.setBodyType(bodyType);
bcmLogTest.writeException(_bcmMsg);
bcmLogTest.close();
```

Next, you follow a series of processes to create a target queue and add it to a service.

Creating and configuring a target queue

First, you will update the configuration table, filling in the EXCEPTIONTYPE column. Then, you'll create a target queue, add the queue resource to the WebSphere Application Server, regenerate the EAR file, and finally import the EAR file and bind the EAR resources to the WebSphere Application Server JMS resources.

Updating the configuration table

To update the configuration table:

1. Use JDBC to open the BCMSM.BCMSMCONFIGURATION database table.
2. Enter the user ID and password.
3. Insert the value of the target queue in the EXCEPTIONTYPE column of the table.
4. Save and close the table.

Creating the target queue

To create the target queue:

1. Create a configuration file XYZ.scp and add the target queue name (*<target queue name>*) as follows:

```
chg ctx(BCM)
#
chg ctx(JMS)
#
# Display the current context
#
dis ctx
#
# Define the Queues, naming the Queues to be used
#
Def q(<target queue name>) Qu(<target queue name>)
#
# Display the new current context
#
dis ctx
#
end
```

2. Save the file to the *<BCT_HOME>\bin* directory.
3. Run the configuration program you created in step 1:
 - a. Start the WebSphere Advanced Administrative Console. Go to **Start > Programs > IBM WebSphere > Application Server V 4.0 AE > Start Admin Server**.
 - b. Open a command window in *<BCT_HOME>\bin* and enter:
`runjmsadminWAS.BAT jmsadminWAS.config < XYZ.scp`
 - c. After running the program, verify that the output contains no errors. Look in the *<BCT_HOME>\Logs\jmswas.log* file.
4. Make sure the local queue manager (*<MQMGR>*) is started.
5. Open the MQSeries Explorer.
6. Under *<MQMGR>*, add the target queue.
7. Complete the following steps to enable Extended Messaging support:

- a. Start the administrative server (if it is not already active) by going to **Start > Programs > IBM WebSphere > Application Server V 4.0 AE > Start Admin Server**.

If the administrative server is already active, stop it and then start it again by completing the following steps. Then open the Administrative Console.

- 1) Navigate the tree to *your_domain* > **Nodes** > *yournode* > **Application Servers**.
 - 2) For each server listed under **Application Servers**, right-click it to bring up the menu.
 - 3) Click **Stop**.
 - 4) When all the servers in the tree are stopped, right-click *yournode*.
 - 5) Click **Stop**. This will stop the administration server.
- b. In the administrative console, select **Resources > JMS providers**.
 - 1) Select **JMS Destinations** and click **New**.
 - 2) Specify the following values:
Name: *<target queue name>*
External JNDI path: **BCT/JMS/<target queue name>**
 - 3) Click **OK**.
 - c. Click **OK** after you see the Successful Completion window.

Adding resources to the EAR file

To add the resource to the EAR file:

1. Open the Application Assembly Tool by clicking **Start > Programs > IBM WebSphere > Application Server V 4.0 AE > Application Assembly Tool**. (Another way to do this is to click **Tool** in the WebSphere Advanced Administrative Console.)
2. Click **Cancel** in the Welcome window.
3. Click **File > Open** and from the Open window, select *<BCT_HOME>\lib\SM_EXCEPTION_EJB.ear*.
If the file does not exist, do the following:
 - a. Select *<BCT_HOME>\lib\SM_EJB.ear*.
 - b. On the General Tab, change the Display name from **SM_EJB.ear** to **SM_EXCEPTION_EJB.ear**.
 - c. Click **Apply**.
4. Add the target queue by following these steps:
 - a. Expand **SM_EXCEPTION_EJB.ear** and **EJB Modules** under it.
 - b. Expand **SM_EJB**, **Sessions Beans**, and **BCMSMRepository.EJB**.
 - c. Verify in the **Binding** tab that the JNDI name is **ejb/com/ibm/bct/sm/BCTSMRepositoryEJBHome**.
 - d. Highlight **Resource References**, right-click, and select **New**.
 - e. Add the target queue name in the Name field.
 - f. From the Type list, select **javax.jms.Queue**.
 - g. Click **Apply** from this window and then **Cancel** from the New Resource Reference window.
 - h. Click **File > Save as**, and enter:
<BCT_HOME>\lib\SM_EXCEPTION_EJB.ear.

- i. Exit from the Application Assembly Tool.

Regenerating the EAR file

To regenerate the EAR file:

1. Open the WebSphere Application Server Administrative Console and remove the enterprise application named **exception** by clicking **exception**, right-clicking, and then clicking **Remove**.
2. Click **Console > Wizards > Install Enterprise Application**.
3. Add the SM_EXCEPTION_EJB.ear file that you just created by entering **exception** in the Enterprise Application name field.
4. Select **jms/BCMPPQCF** and click **Select Resource**.
5. From the list, select **BCTSMConnectionFactory** and click **OK**.
6. Select **jms/BCMSMOutputQueue**, and click **Select Resource**.
7. From the list, select **BCTSMExceptionLogError** and click **OK**.
8. Select **jdbc/BCSMXADDataSource** and click **OK**.
9. From the list, select **BCTSMXADDataSource** and click **OK**.
10. Select **jdbc/<target queue name>** and click **Select Resource**.
11. From the list, **select <target queue name>** and click **OK**.
12. Continue to assign any existing queues.
13. Click **Next** until you see the Selecting the Application Server panel.
14. Click **Select Server**.
15. Select the server you created (**BCTExceptionLog**) and click **OK**.
16. Click **Next** and then **Finish**.
17. Answer **Yes** to the pop-up.
18. Click **OK** to deploy the application, which is then added to the Enterprise Applications list.

Logging requests for a Web service

This section describes how you can log incoming requests and responses for a Web service.

To configure the service to use logging:

1. From the IBM Web Services Gateway page, select **List**.
2. Click the name of the Web service you are configuring.
3. In the Gateway Service Properties section of the page that is displayed, check **Log requests for this service**.
4. Click **OK**.

The Solution Manager can now save a copy of the incoming requests and responses for the service. The requests and responses are sent from the Web Services Gateway machine to the Solution Manager via a message queue and are then stored in the Solution Manager Audit Log.

Note that the two other logs (the Business Log and the Exception Log) maintained by the Solution Manager component are not used by the Web Services Gateway component.

A queue manager and message queue, as well as JMS definitions for the queue manager and queues, are configured automatically during the installation process.

The WebSphere Member Services API Wrapper

This section gives you information about using the WebSphere Member Services API Wrapper.

The API Wrapper is designed to provide an interface to the repository of trading-partner profiles. It allows full access to basic as well as extended attribute data and provides for repository searches.

The WebSphere Member Services API Wrapper is defined and described in its Javadoc. This book gives you information to supplement the Javadoc.

Trading partner profile

The administrator who is registering with an exchange enters information on a Web page. This information includes such things as:

- Company Info
 - Company name
 - Interchange type
 - Interchange ID
 - Street Address
 - City
 - State/Province
 - Zip code/postal code
 - Country
- Primary contact (the user who will have administrative privileges for the exchange)
 - User ID
 - Password
 - First Name
 - Last Name
 - Phone Number
 - Email
 - Fax
- Secondary contact
 - First Name
 - Last Name
 - Phone Number
 - Email
 - Fax
- Exchange credentials
 - Exchange password (the password that will be given to other exchanges during registration processes, such as the Profile Upload service)
- Optional services:
 - Document Exchange
 - TPI/Cyclone

- XML MIME type
- EDI MIME type
- Binary MIME type

The repository of user information holds these values, and you can use the WebSphere Member Services API Wrapper to gain access to the information.

Methods

Three types of methods exist—one for user information, one for member information, and one for organization information.

- A *member* is either a user, an organizational entity, or a member group.
- An *organizational entity* can mean either an organization or an organizational unit.

Methods exist to create, delete, and update users, members, and organization entities. You can also search the repository, convert objects to and from XML, and determine parent and ancestor organizations. (Users are child objects of organizations, and organizations themselves can be child objects of other organizations). These methods are described in the Javadoc for the WebSphere Member Services API Wrapper.

You can use the WebSphere Member Services API Wrapper methods to do other tasks, such as:

- Determine the organization name associated with a particular user (an example of this is shown in “Sample code”)
- Get all members who belong to a particular member group
- Search for a set of users with a specific attribute value
- Assign a member to a group

Sample code

Here is some sample code, illustrating the use of the UserXML method to find the company name associated with a user:

```
import="java.util.*;
import="java.io.*;
import="com.ibm.bct.rp.util.*;
import="com.ibm.bct.rp.cms.*;
import="com.ibm.bct.sc.*;

public class CMSTest {

    public static void main(String[] args) {

        // this code finds the company name for a user

        String userId = "user4";
        String organizationName = null;

        // find company name in CMS

        try
        {

            List userDN = (new UserXML()).findUsers("shortName",userId);
            UserXML aUser = new UserXML(null, (String)userDN.get(0));
            organizationName = aUser.getAttributeValue("companyName");
```

```
        System.out.println("Company name= "+organizationName);
    }
    catch (Exception e)
    {
        System.out.println("Exception on finding company name
        "+e.toString());
    }
    return;
}
}
```

Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described 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:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other 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 NON-INFRINGEMENT, 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.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

WebSphere Business Connection Lab Director
IBM RTP Laboratory
3039 Cornwallis Road
P.O. BOX 12195

Raleigh, NC 27709-2195
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples may include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Programming interface information

Programming interface information is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM
alphaWorks
AIX
CrossWorlds
DB2
DB2 OLAP Server
DB2 Universal Database
DeveloperWorks
MQSeries
SecureWay
WebSphere

Lotus is a trademark of International Business Machines Corporation and Lotus Development Corporation in the United States, other countries, or both.

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

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

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