



IBM WebSphere Business Components Studio

Installation Guide for AIX

Version 1.1

... A member of the WebSphere Business Components family

Before using this information and the product it supports, be sure to read the general information under “Notices” on page 10.

First Edition (December 2000)

This edition applies to version 1.1 of IBM WebSphere Business Components Studio (product number 5639-M22), and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Corrections and suggestions for future revisions of this document are appreciated. Mail your comments to:

IBM Canada Ltd. Laboratory
Information Development
2G/KB7/1150/TOR
1150 Eglinton Avenue East
Toronto, Ontario, M3C 1H7
Canada

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

© Copyright International Business Machines Corporation 2000. All rights reserved.

Note to U.S. Government Users Restricted Rights — Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

IBM WebSphere Business Components Studio, Version 1.1 AIX installation . 1

General installation instructions.....	1
Hardware requirements	1
Software requirements	1

AIX installation 2

Configure the NT installed files.....	2
Configure the AIX script files.....	2

Tools..... 3

Advanced Component Services..... 4

Add .jar files to the server's dependent classpath	4
Add .jar files to the command line argument of the server's classpath.....	4
Deploy AC Services EJBs in WebSphere Application Server.....	4
Configure the JNDI database	5
Configure the AC Services	6
Configure the System Management Console	6

Advanced Components 7

Create the database and its tables.....	7
Configure the datasources in the application server.....	8
Configure the AC by making changes to its configuration file.....	8
Run the AC Deployment Tool	8
Deploy and register the AC in WebSphere	9

Notices..... 10

Trademarks and service marks.....	11
-----------------------------------	----

IBM WebSphere Business Components Studio, Version 1.1 AIX installation

General installation instructions

IBM WebSphere Business Components Studio, Version 1.1 provides a set of standard EJB-based components and associated development tools that help you build, manage, and deploy e-business applications.

To successfully install this product on your AIX system, make sure you have the proper hardware and software.

<i>General installation requirements</i>	
Usage description	Software requirements
To install the complete product	AIX (4.3.3 and ptf 9) and WebSphere® Application Server Advanced Edition (3.5 and ptf 2)

Hardware requirements

The minimum hardware requirements include the following:

- 450 MHz system
- 512 Mb of RAM
- 6 GB of available hard drive space
- CD ROM

Software requirements

- AIX (4.3.3 and ptf 9)
- WebSphere Application Server Advanced Edition(3.5 and ptf 2)
- DB2® UDB v6.1 ptf 4

AIX installation

To install WSBC Studio 1.1 on AIX, first install WSBC Studio 1.1 on Microsoft® Windows NT using the instructions in the *IBM WebSphere Business Components Studio Installation Guide* on the CD. Once installation is complete, follow the steps below to install on AIX:

1. Zip up the directory, using a tool that provides directory structure support.
2. Create a `wsvc` directory on AIX (for example: `mkdir /usr/wsvc`). This directory will be the `$WSBC_HOME` directory.
3. FTP the NT installation zip file to that directory.
4. Extract the NT installation zip file using directory structure support.
5. Copy the `AIXScripts.zip` file to the same `$WSBC_HOME` directory, and extract the zip file using directory structure support. This zip file contains Korn shell scripts for setting up the database and environment, as well as scripts for running the WSBC AC tools.

Configure the NT installed files

The `$WSBC_HOME/WSBCconfig.ini` file must be changed for AIX-specific information. Change the `WSInstallBase` line to point to where the WebSphere Application Server is installed, and change the `InstallBase` line to point to the `$WSBC_HOME` directory. The result might look like the following::

```
. . .
WSInstallBase=/usr/WebSphere/AppServer
. . .
InstallBase=/usr/wsvc
. . .
```

Configure the AIX script files

Edit the `$WSBC_HOME/bin/setwsbcenv.sh` file to specify the installation locations of WebSphere, DB2, and WebSphere Business Components.

```
. . .
WAS_HOME=/usr/WebSphere/AppServer
. . .
WSBC_HOME=/usr/WebSphere/AppServer/wsvc
. . .
DB2_HOME=/home/db2inst1/sqllib
. . .
```

This script is used by other scripts to set the WebSphere and the WSBC home directory. The `$WSBC_HOME/bin/setwsbcenv.sh` script can also be used to set up the client classpath.

Tools

This section describes what you have to do to complete the configuration of the AC Deployment Tool.

To configure the AC Deployment Tool, complete the following steps:

1. Edit the `actools.ini` file found in `$WSBC_HOME/tools` so that the `basepath` is set to the directory where the `settings.xml` file is found. The `basepath` should end in a slash, as in the following example:
`basepath=/usr/wsbc/tools/`
2. To run the AC Deployment Tool, use the `runDeploymentTool.sh` file from the `$WSBC_HOME/tools/bin` directory.

Once you start the AC Deployment Tool, you must specify the root directory for WebSphere, in order to deploy ACs to WebSphere. To specify the root directory for WebSphere, complete the following steps:

1. On the **Actions** menu, click **Set Preferences**.
2. In the **Options** dialog that appears, select **WebSphere** as your EJB Server.
3. In the **Root Directory** field, type the path to the directory where WebSphere is installed on your system (for example: `/usr/WebSphere/AppServer`).

Advanced Component Services

This section describes how to deploy the AC Services in WebSphere Application Server on AIX.

Add .jar files to the server's dependent classpath

1. Start the WebSphere Administrator's Console (`$WAS_HOME/bin/adminclient.sh`)
2. In the tree view, select the node that is named after your server.
3. In the right-side pane, append the following string to the **Dependent classpath** field (without line breaks or any intervening white space):

```
$WSBC_HOME/Services/BaseServices/ACServicesServer.jar  
:$WSBC_HOME/Services/BaseServices/jmxri.jar
```

4. Select **Apply**.
-

Add .jar files to the command line argument of the server's classpath

1. Start the WebSphere Administrator's Console (`$WAS_HOME/bin/adminclient.sh`)
2. On the menu bar, select **View > Topology**. Expand the node named after your server. Select the application server where you want to deploy an AC (for example, **Default Server**).
3. In the right-side pane, enter the following string to the **Command line arguments** field (without the line breaks or any intervening white space except the one after "classpath"):

```
-classpath  
$WSBC_HOME/Services/BaseServices/ACServicesEJBDeployed.jar:  
$WSBC_HOME/Services/BaseServices/ACServicesServer.jar:  
$WSBC_HOME/Services/BaseServices/ACEI18NEJBDeployed.jar:  
$WSBC_HOME/Services/BaseServices/jmxri.jar
```

4. Select **Apply**.
-

Deploy AC Services EJBs in WebSphere Application Server

For each application server where you want to deploy an AC, deploy the following Beans from their respective .jar files as listed in the table:

<i>AC Services EJBs</i>	
EJB	.jar file
LocalizableTextResourceAccessor	ACEI18NEJBDeployed.jar
ACCommandTarget, MBeanServerConnector	ACServicesEJBDeployed.jar

Note that there are two Beans stored in the `ACServicesEJBDeployed.jar` file. The following steps show you how to deploy these AC Services for a given application server:

1. On the menu bar, click **View > Topology**. Expand the node named after your server. Expand the application server where you want to deploy an AC (for example, **Default Server**). Right-click the container where you want to deploy an AC (for example, **Default Container**), and select **Create > EnterpriseBean** from the pop-up menu.

Note: If you are going to redeploy a Bean that already exists in a container, you first must stop, then remove that Bean. Afterwards, you may recreate that Bean.
2. In the **Create EnterpriseBean** window, select the **General** tab. In the **Jar file** field, specify the .jar file by following these steps:
 - a. Click on the **Browse** button (be prepared to wait 1-2 minutes).
 - b. Double-click the file you want to deploy. For example, to deploy the LocalizableTextResourceAccessor, you would double-click the ACEI18NEJBDeployed.jar file.
 - c. Double-click the .ser file for the Bean you want to deploy. For example, if you double-clicked the ACEI18NEJBDeployed.jar file, you would double-click the com.ibm.websphere.i18n.localizabletext.LocalizableTextResourceAccessor/LocalizableTextResourceAccessor.ser Bean.
 - d. You will be asked, "This jar is not enabled for Work Load Management. Would you like to enable it now?" Click **No**.

WebSphere Application Server should have automatically filled in the **Name** and **Deployment descriptor** fields for you.

Note: If you use more than one server to host multiple ACs, you must ensure that the JNDI home names of the ACCommandTarget and the MBeanServerConnector EJBs are different for each server. Therefore, you may have to change the **Deployment descriptor** fields.

The MBeanServerConnector Bean home name must be the ACCommandTarget Bean home name combined with the fixed name "MBeanServerConnector" like this (without any line breaks or intervening spaces):

```
commandTargetHomeNameMBeanServerConnector
```

The variable *commandTargetHomeName* is the home name used for the ACCommandTarget Bean in that server.

The LocalizableTextResourceAccessor Bean must also have a unique home name for each EJB server. If a new home name is assigned to LocalizableTextResourceAccessor, the corresponding entry in ACServices.ini must be changed, as described in "Configure the AC Services."

3. Click **OK**.
4. Right-click the Bean that you have just created in the tree and select **Start** from the pop-up menu.

Repeat steps 1-4 for the rest of the Beans.

Configure the JNDI database

A common problem you may encounter is that the AC Command Service uses JNDI to register the command properties, using as the name *<AC instance name + function name + properties>*. This often exceeds the default length for JNDI names, which is 64 characters. To use JNDI names that are longer than 64 characters, you have to alter the naming server's tables in the database to a length that can handle the JNDI name. In most circumstances, a length of 128 is sufficient. To do this for DB2, run the `$WSBC_HOME/bin/updateJNDI.sh <databaseName>`, replacing *databaseName* with the name of the database you are using (WAS for WebSphere).

The following example makes the maximum length 128.

```
CONNECT TO databaseName

ALTER TABLE EJSADMIN.BINDINGBEANTBL ALTER COLUMN NAME SET DATA TYPE
VARCHAR (128)

ALTER TABLE EJSADMIN.PROPERTYBEANTBL ALTER COLUMN NAME SET DATA TYPE
VARCHAR (128)

ALTER TABLE EJSADMIN.PROPERTYBEANTBL ALTER COLUMN PROPERTYNAME SET DATA
TYPE VARCHAR (128)

DISCONNECT databaseName
```

Configure the AC Services

You may extract and customize the file `ACServices.ini` contained in the file `ACServicesServer.jar`.

If a new JNDI home name (such as `newHomeName`) was assigned to the `LocalizableTextResourceAccessor` Bean, find the following entry in the `ACServices.ini` file:

```
formatterHomeName=com/ibm/websphere/i18n/localizabletext/LocalizableTextResourceAccess
orHome
```

Change the above entry to the following:

```
formatterHomeName=newHomeName
```

Whatever JNDI home name you use must be unique to the server.

To apply the changes you have made in this file, you may do either one of the following:

- Recreate the `ACServicesServer.jar` with your version of `ACServices.ini`
- Add your version of `ACServices.ini` *before* the file `ACServicesServer.jar` in the classpath, so that your version of the file takes precedence. You must specify the full path name of `ACServices.ini`.

Configure the System Management Console

You may extract and customize the file `SMConsole.ini` contained in the file `ACServicesSMConsole.jar`. To apply the changes you have made in this file, you may do either one of the following:

- Recreate the `ACServicesSMConsole.jar` with your version of `SMConsole.ini`
- Add your version of `SMConsole.ini` *before* the file `ACServicesSMConsole.jar` in the classpath, so that your version of the file takes precedence. You must specify the full path name of `SMConsole.ini`.

Advanced Components

This section provides high-level descriptions of the steps that must be performed to install and configure one of the Advanced Components of Version 1.1 Studio in WebSphere Application Server on AIX. While each step is only a high-level description, they also identify where to find more information should you require it. For example, information related to an Advanced Component can be found in the online information included on the NT Studio CD, in either the component's documentation or in the general section for Advanced Components.

The process of installing and deploying an AC involves the following major steps:

1. Creating the database and its data tables.
2. Configuring the datasources in the application server.
3. Configuring the AC by making changes to its configuration file.
4. Running the AC Deployment Tool.
5. Deploying and registering the AC in WebSphere.

Each step is described in more detail below.

Create the database and its tables

For each AC, create the database that it uses. The following table shows the default name of the database that you should create for each AC. If you do not use the default database names, you must make appropriate changes in the `ACImplementation.xml` file. If you do not use the default high-level qualifier of `USERID`, make the appropriate change to the `ACImplementation.xml` file as well.

<i>AIX script files</i>		
AC	Database name	Script files to create the columns (DB2 only)
Customer Profile	cpBoth	<code>\$WSBC_HOME/ACFeatures/Components/CustomerProfile/aix/db2/customerProfileEnterpriseDB.sh</code> <code>\$WSBC_HOME/ACFeatures/Components/CustomerProfile/aix/db2/customerProfilePersonDB.sh</code>
Product Catalog	proCat	<code>\$WSBC_HOME/ACFeatures/Components/ProductCatalog/aix/db2/productCatalogDB.sh</code>

Version 1.1 of the AC requires a specific database structure. The Persistence section of the online AC documentation (see **<component name> Tell me about >Version**) contains a diagram that shows the tables and their relationships within the structure. You can also consult the `ACImplementation.xml` file to obtain the default names of the tables and their columns. If you are using DB2 for AIX, you can use the provided script files to create the tables for the database.

To create the database in DB2, run the script file using the following format and arguments:

- For CustomerProfile:
`<scriptfile.sh> <databaseName> [<userID> <password>] [<qualifier>]`
- For ProductCatalog:
`<scriptfile.sh> <databaseName> [<userID> <password>] [<qualifier>]`

where *databaseName* is the name of the database you are creating, and the optional arguments of *userID*, *password*, and *qualifier* are the user ID, password, and qualifier that you use to access the database.

Note: Although you cannot change the structure of the database, you can customize the names of the tables and columns, add columns, and set the maximum size of the columns.

Configure the datasources in the application server

This step makes the application server aware of the databases you created earlier. To do this:

1. Start the WebSphere Administrator's Console (`$WAS_HOME/bin/adminclient.sh`)
2. For each datasource, configure it by selecting **Types View** and then right-clicking **DataSource** and selecting **Create**. For the datasource name, use its name (for example, proCat for the Product Catalog AC database) and not its URL. For the JDBC Driver, use the default Admin DB Driver.

Configure the AC by making changes to its configuration file

This step enables you to customize the `ACImplementation.xml` file, which is used to create the AC instance. If, for example, you have customized the name of a column, you must change a setting within the `ACImplementation.xml` file for the AC.

If you are not using the default values for the database qualifier and datasource name specified in the `ACImplementation.xml` file, you must change the appropriate settings in that file. For example, for the Customer Profile AC, you would change the `personDataSource`, `enterpriseDataSource`, `personDBQualifier`, and `enterpriseDBQualifier`. For other customizations that you can make, see the [Customize page](#) and the [Configure page](#) for the AC.

The `ACImplementation.xml` file is in the `.jar` file along with the AC's other files. At deployment time, the AC Deployment Tool stores the system management portion of the `ACImplementation.xml` file into its persistent storage.

Run the AC Deployment Tool

This step determines dependencies, registers the name of the AC instance, and creates the `.jar` files for the System Console. It involves the following steps:

1. Create a copy of the original `.jar` file and store the copy in the deployment directory. Use the original version of the file for runtime.
2. Run the AC Deployment Tool to create the output copy of the `.jar` file.
3. To update the AC with any changes made using the AC Deployment Tool, copy the `ACDD.xml` file from the output copy of the `.jar` file to the runtime copy.

You can use the default settings provided by the AC. If you change the name of the AC instance or the JNDI name of the AC's Session Bean (the Function Group Registered Name as it is called by the tool), note these changes as you will need them when deploying the AC's EJBs.

For more information on the AC Deployment Tool, see the online information. The .jar files used by the tool to create the output .jar files are located in the following directories:

<i>Location of the AC .jar files</i>	
AC	.jar file directory
Customer Profile	\$WSBC_HOME/ACFeatures/Components/ CustomerProfile
Product Catalog	\$WSBC_HOME/ACFeatures/Components/ ProductCatalog

Deploy and register the AC in WebSphere

The following are the high-level steps to deploy and register the AC. For more information on each step including, for example, which .jar files to add to the classpath and which Beans to deploy on the EJB server, see "Deploying an AC" (expand **Advanced Component** and then expand **General Information**) in the online information.

1. Open the WebSphere Administrator's Console.
2. Add the .jar files to the command line argument of the server's classpath.
3. Add the .jar files to the server's dependent classpath.
4. Deploy the EJBs in the EJB server.

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:

IBM Canada Ltd.,
Department 071,
1150 Eglinton Avenue East
Toronto, Ontario, M3C 1H7
Canada

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.

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.

Trademarks and service marks

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

AIX
CICS
DB2
DB2 Universal Database
e-business
IBM
LANDP
MQSeries
OS/2 Warp
OS/390
RS/6000
SanFrancisco
VisualAge
Visual Banker
WebSphere

Lotus, Domino, Lotus Notes, and Notes Mail are trademarks of the Lotus Development Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered 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 or registered trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark in the United States, other countries, or both and is licensed exclusively through X/Open Company Limited.

Rational Rose is a registered trademark of Rational Software Corporation.

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

End of document