IBM Component Broker Products
Release 2.0
Late Breaking News

This page last updated Wednesday, January 13, 1999 at 5:50 PM.

Contents

Migration Information for Release 2.0 Users

For information on migrating models from Release 1.3 and release-to-release compatibility and interoperability, please refer to the Planning, Performance and Installation Guide for Component Broker 2.0.

Release Contents

Changes from Prior Release

The following has been added or completed testing since Release 1.3. For more details on what's new, refer to Chapter 1 of the Application Development Tools Guide and the "What's New" section of the System Administration Guide.

- Improved Stability
- Improved Performance
- Platform Support:
    - AIX 4.2.1 (NT equivalence, except as noted)
    - Windows NT 4.0 Server
- Visual Basic 5.0 Client
- Application Adaptors (NT, AIX, except as noted)
    - CICS (3270, ECI, LU 6.2 Sync Level 2) (3270 is only supported on NT)
    - IMS (3270, LU 6.2 Sync Level 2) (3270 is only supported on NT)
    - Oracle Application Adaptor (Windows NT only)
- Enterprise Java Beans sample
- SSL - Java client to server (Windows NT only)
- Notifications - Transient
- Object Builder usability improvements
- Object Builder team support
- Workload Management (transient application objects)

Note: Workload management is now available on both NT and AIX. The use of workload management is recommended only with transient application objects as exemplified in the Tutorial\WLM sample. The use of workload management with other application architectures is still considered early-tested function.

Early-Tested Functions

In order to promote rapid feedback on new functions, the Component Broker team includes certain functions in a release that have not been completely tested. We encourage you to work with these functions and report any problems observed, in the same manner as with the remainder of the release. In this way, we can deliver function more quickly and improve its quality by exposing it to your environment earlier. Please be assured that problems in these functions are treated with the same gravity as problems elsewhere in the release.

In this release, the following functions are classified as "early-tested" and available on both Windows NT and AIX, except as noted:

- Business Rules framework sample (NT only)
- Flowmark 2.3 client (NT only)
- Object Builder Model Consistency Checker
- Rose Bridge - round tripping
- Workload Management - (Other than transient application objects)

Known Component Broker Restrictions

This release of the Component Broker products has the following restrictions. These restrictions arise from

known problems that are being resolved for subsequent refreshes. As more information becomes available about restrictions with this release, that information along with additional hints and tips will be posted on the Component Broker pages of the IBM Technical Support page at: http://service.boulder.ibm.com.

Installation and Configuration

Note: Most installation and configuration information is discussed in the *IBM Component Broker Planning, Performance and Installation Guide*.

Using the Prerequisite Checker	The "CICS and IMS Application Adaptor SDK" is erroneously listed under the Prerequisite Checker "Products" list. Do not start the Prerequisite Checker with this option selected.	All Platforms
	To check if the "CICS and IMS Application Adaptor SDK" will install, use the product option "Component Broker Toolkit". Under this option use the package "CICS and IMS Application Adaptor SDK".	
Installing UDB	If the **CLASSPATH** or **PATH** environment variables have very long values, installation of UDB Version 5.0 may not succeed and may leave these variables with incomplete settings.	Windows NT

Either of the following symptoms indicate that this problem has occurred:

- Either of these variables is left with only the data added to it by the UDB install.
- The installation may fail, showing a window titled "DlgcacWinName:_INS0432._MP - Application Error" and containing the statement "The instruction at 0x00706c65" referenced memory at 0x00706c65. The memory could not be read."

In either case, the environment variable settings must be corrected.

To recover from this problem, save the environment settings prior to starting the UDB installation. There are two methods to save the settings:

- Save the settings into a text file, then manually update the settings after the UDB installation completes.
- Create new environment variables to hold the original settings, then manually update the environment variables with the original settings after the UDB installation completes.

In either case, reducing the **CLASSPATH** and the **PATH** variables' settings to lengths less than 300 characters allows the UDB installation to complete successfully. After the installation completes, these variables may be modified to include all of the

original settings. When reducing these variables' settings, do not remove any operating system paths such as C:\WINNT\system32 or C:\WINNT.

This problem has been corrected in UDB V5.2.

Installing the DB2 SDK	The instructions in the Quick Beginning Guide and the Planning, Performance and Installation Guide, "Installing the Prerequisite Software on AIX/Installing the DB2 Software Developer's Kit" chapter/section should be revised as follows:	AIX

Step 1. Installing the Bundle Definitions...

Substep 4. is not necessary (cd'ing to /cdrom/db2/aix/db2)

Insert new steps between 5 and 6 like this:

x. On this screen:

1. In the INPUT device / directory for software field, type
2. */cdrom/db2/aix/db2*
3. Press the Enter key to continue.

Step 2. Installing the Software...

Replace substep 4. with the above x. substep.

Correcting path lengths greater than 512 bytes	Windows NT(R) 4.0 has problems forming the correct PATH environment variable when the combined length of the System PATH and User PATH environment variables are greater than 512 bytes. Component Broker installation will warn users of this potential problem if changes to the PATH environment variable will increase the number of bytes over 512 bytes by displaying the following message:	Windows NT

Component Broker installation has detected a potential problem with the length of the path environment variable.

If this informational message is seen, the following steps should be done before continuing with the install.

1. See the Problem Determination chapter of the Planning, Performance, and Installation Guide for additional information on possible solutions on reducing the PATH environment variable size.
2. Create or maintain the Emergency Repair Disk prior to installing Component Broker using the RDISK.EXE utility provided with the operating system. The Emergency Repair Disk records vital information about your Windows NT configuration. Performing registry backups on a regular basis is also recommended.

Selecting the directory to install a Java client on OS/2	On OS/2, the browse function of InstallShield Java Edition does not work correctly when selecting the directory into which the Java client is to be installed.	OS/2
	If the default directory is not acceptable, type the desired directory into the entry box rather than using the browse function to select the directory.	
Installing Client SDKs	While installing the CBToolkit, attempting to install both the C++ Client SDK and the ActiveX Client SDK on the same system will cause the following message:	Windows NT
	You have selected both C++ Client SDK and ActiveX Client SDK to be installed on the system. Installing both SDK on the same system can cause VisualAge C++ compilation problem. Please follow special procedures in Appendix D of Quick Beginnings booklet to resolve the problem.	
	The message incorrectly refers to the Quick Beginnings Guide. It should be the Planning, Performance, and Installation Guide.	
Supported Levels of the JDK	Component Broker is supported on and has been tested with the Java Development Kit (JDK) 1.1.6. While Component Broker will install with JDK 1.1.7 and above, these levels of the JDK are not currently supported. If you run into problems while using JDK 1.1.7 or above, you must reproduce the problems using JDK 1.1.6 before reporting the problems.	All Platforms
Installing DCE	Certain background processes may cause DCE 2.0 installation to fail, including Microsoft IIS.	Windows NT
	Stop Microsoft IIS before installing DCE 2.0. If DCE 2.0 installation still fails, try stopping other non-essential processes before installing DCE.	
libipfx.a missing on AIX	The install code for Component Broker checks for the prereq, ipfx.rte 2.2.0.0, or higher by querying the system ODM. In some systems, an error message has been reported:	AIX
	can not load library libipfx.a	
	when trying to start the EUI. This should be a symbolic link:	
	/usr/lib/libipfx.a -> /usr/lpp/ipfx/lib/libipfx2.a	
	and the target file, /usr/lpp/ipfx/lib/libipfx2.a, should exist. If this link is broken or if the target file does not exist, then the EUI will report an error similar to the above.	
	To fix the problem, reinstall ipfx.rte from the AIX 4.2.1 compact disc.	

Component Broker Toolkit

Tools

Note: Most tool information is discussed in the online documentation and the *IBM Component Broker Application Development Tools Guide*, including many restrictions. Additional restrictions are described below.

Planning for Component Broker 2.1	It is anticipated that Component Broker 2.1 will require AIX 4.3 and CSet 3.6. This will require that any programs developed with CSet 3.1.4 for AIX 4.2.1 on CB 2.0 be recompiled for deployment on CBConnector 2.1.	AIX
Starting Object Builder	An invalid Object Builder configuration file may cause a failure when starting Object Builder, and produce exception messages like the following: *java.lang.NullPointerException at com.ibm.ivb.ob.ui.FilterContribution.updateMenuItems(Compiled Code)* To recover from this problem, locate the configuration files, delete them and restart Object Builder. The configuration files are named panes.ini, workbook.ini, and default.cfg. The files are stored in a "config" directory, located by the "UserProfile" environment variable (NT) or the "Home" environment variable (AIX).	All Platforms

Managing Object Builder's memory usage	Object Builder consumes memory as it is used. Performance may degrade and the likelihood of deadlocks within the tool may increase with prolonged use. If Object Builder deadlocks, all changes since the last save are lost.
	
	There are a few techniques to help avoid excessive memory consumption and avoid or reduce the impact of deadlocks in Object Builder.
	
	1. Keep the number of programs running on your computer to a minimum while running Object Builder to help maximize the available memory.
	2. Save your model frequently, for example, every 15 minutes
	3. Save, shutdown and restart Object Builder periodically. For example, do this...
	
	    a. every hour, or
	    b. after deleting many objects, or
	    c. if there is a clear degradation in performance or an increase in paging activity.
	
	4. When you wish to open a different project from Object Builder, shutdown and restart Object Builder instead of using the "File->Open New Project..." menu option.
		All Platforms

If Object Builder deadlocks, go to the Object Builder command window and issue a "CTRL-BREAK" (not a "CTRL-C"). This will dump a description of the internal state of the Java Virtual Machine to the command window. Please report this information to your IBM sponsor to assist in solving the deadlocks permanently. Note: Your default command window screen buffer size should be set to at least 100 lines to accomodate most JVM dump reports.

Populating an IR	When two IR-populating programs are executed simultaneously the following error message may be displayed:	All Platforms

An internal error has occurred in the Interface Repository database.

An IR-populating program is generated by compiling and linking the source code emitted by the idlc compiler using the -sir option. They are normally named X_IR.exe or X_IR, where X.idl is the idl filename.

To recover, run the IR-populating program that encountered the error again to ensure proper population of the IR. To avoid the error, don't run more than one IR-populating program at a time.

Support for CORBA Event and Notification services added	In R2.0, the Object Builder has added direct support for the CORBA Event and Notification services. If you have previously added any of these interfaces into your model, you should delete them from the model and use the system provided interfaces. The files containing the interfaces are:	All Platforms

CosEventComm
CosEventChannelAdmin
IExtendedQuery
CosNotifyComm
CosNotifyChannelAdmin

Building code from the UI or for "team support"	If the mode of makefile generation is "standalone", the makefile will expect all code for all dependent projects to exist in the working directory of the current project. The batch "obgen" command will correctly generate all the source for all dependent projects if the "-linked" option is NOT specified. However, if the generation is run from the UI, only the source for the current project is generated.	All Platforms

To work around this problem, the source for all dependent projects should be generated using obgen prior to starting Object Builder.

If the mode of makefile generation is "team support" (or "-linked" from obgen), then both the UI and batch tool will generate source for only the current project.

To work around this problem, the source for the dependent

projects should be generated from each of those projects.

See the Object Builder team support sections of the Application Development Tools Guide for more details.

Using "sqlprep" command when run on DB2 Client installation	The "sqlprep" command that is run by an OB makefile to preprocess an Embedded SQL Persistent Object or ".sqx" file may fail when run on a DB2 Client installation or without an authorized connection to the database.	All Platforms

If this occurs, for an AIX client, edit the emitted makefile(s) and add a "user" clause to the sqlprep recipes according to this example.

If the sqlprep command in the recipe reads:

sqlprep MyPO.sqx MyDBName -p=MyPOPkg -b

Then replace the command with:

sqlprep MyPO.sqx "MyDBName user myUserid" -p=MyPOPkg -b

where myUserid is the userId under whose authority you wish the sqlprep operation to run. Start the make process from a regular command shell and be prepared to enter the password for the userid when prompted.

A similar update will be required for the obdatapr command for an NT client.

Avoiding unwanted attribute initializers in Data Objects	Object Builder 2.0 requires a default initializer for attributes of most variable-length constructed types such as object references, sequences and variable-length structs and unions. If you clear the "Initializer" field on the Attributes page for such attributes, you will continue to get initializing calls to the setters in the internalize...() methods of the DO Implementation and in the constructor if the DO is delegating. These initializing calls may be unwanted if you have provided your own implementation for a setter method. Unlike for simple attributes, you cannot suppress the initializing call by clearing the "Initializer" field.	All Platforms

Use one of the workarounds listed below:

- Write a setter implementation that can tolerate the initializer call. For example, provide an initialization value that your implementation can detect and handle as a special case. or,
- Customize the setter in the Business Object instead of in the Data Object.

Importing Release 1.3 XML and errors	When importing XML that was generated by Release 1.3 Object Builder, you may see errors reported of the form:	All Platforms

reported by OB

Method insertToDataStore is not a method of BankPO

Method updateToDataStore is not a method of BankPO

Method retrieveFromDataStore is not a method of BankPO

Method deleteFromDataStore is not a method of BankPO

These errors refer to a changed framework method on CB/390, and they can be ignored. The imported model is correct for all platforms.

Overriding framework methods or attributes in the Rose model	When overriding framework methods or attributes in the Rose model to be exported to Object Builder, the overridden methods and/or attributes in your user model will appear in Object Builder as user-defined instead of overridden attributes/methods.	All Platforms

To successfully override methods or attributes, you must load the appropriate Component Broker Framework units into your Rose model, and specify them as uncontrolled (managed.cat,services.cat or boim.cat). To specify that a unit is uncontrolled, click on it in the diagram where it was loaded and click:

'File->Units->Uncontrol <package name>'.

Handling the Out of Memory exception message	When using Object Builder, the following exception may appear occasionally on the console:	Windows NT

java.lang.OutOfMemoryError
at sun.awt.image.Image.<init>(Compiled Code)
<additional lines>

This exception causes a failure in repainting the display, but does not affect the model.

The exception only occurs if the JIT is enabled.

This message may be ignored.

Creating Persistent Objects from Views	When creating POs from views, ensure that the key is set in the view prior to creating the PO.	All Platforms
Understanding the behavior of PO attributes not mapped by DO attributes	When a DO is initialized from a Copy Helper and then inserted into the persistent store, any attributes of the underlying PO not mapped to the DO will be initialized to non-NULL "sentinel" values (such as 0 for numbers and "" for strings).	All Platforms

This behavior may be unwanted in some scenarios, such as DO Inheritance with Views into a Single Table (where the parent and child DO views test for NULL and non-NULL respectively on a child column in the shared table).

To avoid this behavior, do the following steps:

1. Use the View Editor of CBToolkit to create a new View over the Table. When defining the View, select only those columns that will be mapped to the DO.
2. Open the Add Persistent Object SmartGuide on the new View. Select "View is Updatable". Indicate the appropriate key attributes. Finish the SmartGuide to create a new PO.
3. Open the properties SmartGuide on the DO Impl and associate the new View PO with the DO.
   Reminder: Make sure that the first associated PO has all the key attributes and that remaining POs are in alphabetical order.
4. Map the DO attributes to the attributes of the new View PO. Do not remove the mappings to the existing POs.
   Note: In the DO Inheritance with Views scenario, make sure that the last mapping of each DO attribute is to the original View PO from which the data is being retrieved.
5. Map the DO insert, update, del and setConnection methods to the new PO instead of the Table PO. Map the retrieve method to the original View PO.

Creating transactional managed objects	There is a restriction on the type of container that may be configured for transactional managed objects. When Use RDB Transaction Services is checked on the Container - SmartGuide frame for Services, there are three types of behavior for methods called outside a transaction.	All Platforms

1. Start a new transaction and complete the call.
2. Throw an exception and abandon the call.
3. Ignore the condition and complete the call.

The third option is not supported in this release and should not be selected. While the Object Builder tool will successfully generate the model with this container, the application may experience intermittent failure while executing methods outside of a transaction. The exception from the server is

IBOIMException::IDataObjectFalied

and the DB2 database error is

SQL0998N, Reason Code = 5. Subcode = "". SQLSTATE=58005 routine invoked in an improper context.

The reasons for this restriction are complex. Using SQL outside of a transaction requires that the thread making the request have its own non-transactional connection established. Component Broker continues to support customization of data objects to do calls to databases, but not as part of a transactional being external commit coordinated by Component Broker.

	Do not select the "Ignore the condition and complete the call" behavior type.	
Generating application installation images using InstallShield 5.1	Due to base path limitations with NT, generating installable application images using Object Builder and InstallShield 5.1 may result in a "The parameter is incorrect" message when attempting to execute the resulting setup.exe. The setup.exe file is on the following path: *E:\myproject\Working\NT\myAppFam\IS5.1\myAppFam\Media\Default\Disk Images\disk1\setup.exe* where 'E:\myproject' is the user project directory and 'myAppFam' is the user Application Family. A workaround for this problem has been provided in the form of an Object Builder generated batch file located at: *E:\myproject\Working\NT\AppFam\IS5.1\setup.bat* Execute setup.bat instead of setup.exe. This batch program calls setup.exe.	Windows NT
Using Object Builder batch files remotely	When executing Object Builder batch files, such as obgen, obimport, obexport, and obcheck, remotely via a telnet connection to an AIX development machine, error messages similar to the following may be returned: *Xlib: connection to ":0.0" refused by server* *Xlib: Client is not authorized to connect to Server* To execute batch files remotely, the "$DISPLAY" environment variable must be set on the AIX server.	AIX
Syntax error when using DB2 keywords	When defining a database schema in object builder, if the table name or column names are OO-SQL keywords, then you must bracket the name with quotes (for example, "select"). If importing a relational schema, make sure the column and table names in the file being imported are enclosed in quotes. If you use OO-SQL keywords as object attribute names and you have any user defined OO-SQL queries, enclose the attribute names in quotes, for example: select a."select" from myHome a; The OO-SQL keywords are documented in the Component Broker Programming Reference.	All Platforms
Debugging Java BO code	Java Business Object debugging on AIX is not supported.	AIX
Using the Remote	When using the Remote Debugger, hangs may occur. In the	All

Debugger with the -nojit option	Application Development Tools Guide, for the Remote Debugger in the "Java Client Application" section, it states : "At a command prompt, enter this command:" *java_g -debug ..... PolicyApp* To avoid the hangs, run the debugger as: *java_g -debug -nojit ......PolicyApp*	Platforms
Debugging the Home business object	The debugger will cause the server to exit if it hits a breakpoint in the Home business object. To work around this problem, use 'Trace and Debug with prompt' option in Object Level Trace and skip all breakpoints within all Home business objects. That will prevent the debugger from shutting down the server.	All Platforms
Transferring emitted files to AIX or OS/390	When running the CBToolkit on NT, any files emitted for the AIX and 390 platforms should be transferred to those platforms using a file transfer protocol,such as FTP or NFS, that is capable of converting CR/LF sequences to Unix newlines. If the emitted files are transferred in binary, either individually or within an archive such as a "tar" file, then the files may not build successfully due to the presence of trailing control-M characters inside the files.	Windows NT
Creating debug compilations	Code generated from the Object Builder tool is normally compiled using the -Ti+ option to produce a debug version. The VisualAge C++ Version 3.5.5 compiler provided with this release will automatically switch to limited debug information if the type information exceeds 64KB. If this occurs, the debug version will only support line-number-only debugging.	Windows NT

Samples

Finding the Policy sample documentation	The documentation for the Policy sample is located in the following files: NT: x:\CBroker\samples\InstallVerification\ProgrammingModel\Docs\progmodel.html AIX: /usr/lpp/CBToolkit/samples/InstallVerification/ProgrammingModel/Docs/progmodel.html (The Quick Beginnings Guide and Planning, Performance and Installation Guide mistakenly refer to the "IVP" directory)	All Platforms

Programming Model

Using security - delegation	Several areas of the documentation discuss "delegation" support. This capability is not present in this release.	All Platforms
	Please ignore those areas of the documentation referring to "delegation" support.	
	In this release, the method SecurityLevel2::Credentials::refresh() is not functional.	All Platforms
Matching key strings to their underlying resource manager fields	The default mapping of a DECIMAL column in a DB2 table is to an attribute of type double. If a DECIMAL column must be mapped to an attribute of type string in a Business Object, contact your IBM sponsor for assistance, referencing Feature 50919.	
Using Factory Finding with Reference Collections	Reference collections are configured to be available on all application servers that run the DB2 Adaptor. In situations where there are multiple servers on a host configured with reference collections, the decision about which server to create reference collections on is based on the factory which is found via the factory finder. If the normal host scoped factory finder is used, the server returned is indeterminate.	All Platforms
	Using various scopes on the factory finder can alter this pattern. See the online help for factory finders for more information.	
	In release 1.3, the following strings were used in the factory finder interface for reference collections:	All Platforms

> *"IManagedCollections::IReferenceCollection.object interface/DB2ReferenceCollectionFactory.object home"*
>
> *"IManagedCollections::IKeyedReferenceCollection.object interface/DB2KeyedReferenceCollectionFactory.object home"*
>
> *"IManagedCollections::IReferenceCollection.object interface/TransReferenceCollectionFactory.object home"*
>
> *"IManagedCollections::IKeyedReferenceCollection.object interface/TransKeyedReferenceCollectionFactory.object home"*

In release 2.0, these strings have been changed to hide the underlying database technology and to be consistent across all Component Broker platforms.

The new strings are:

> *"IManagedCollections::IReferenceCollection.object interface/PersistentReferenceCollectionFactory.object home"*
>
> *"IManagedCollections::IKeyedReferenceCollection.object interface/PersistentKeyedReferenceCollectionFactory.object home"*
>
> *"IManagedCollections::IReferenceCollection.object interface/TransientReferenceCollectionFactory.object home"*

> *"IManagedCollections::IKeyedReferenceCollection.object*
> *interface/TransientKeyedReferenceCollectionFactory.object home"*

New applications should code to the new strings. An additional "DB2" home is provided in this release, however, which allows access of collections using the 1.3 version of the factory finding strings. The old strings may be deprecated in Release 2.1.

Using transactions over reference collections	You may encounter a transaction deadlock problem while running more than one instance of a program which, in one transaction:	All Platforms

1. Adds a number of records to reference collections (or home collections)
2. Then retrieves the records either directly by creating an iterator on a collection or indirectly through queries on a collection.

This error could be reported either as CORBA::INTERNAL or CORBA::PERSIST_STORE.

To prevent transaction deadlock problems, split the operation between two transactions. Add records to collections in one transaction and commit the transaction. Then create a second transaction to retrieve the records from the collections.

Accessing userData in a specialized home	The IBOIMExtLocalToServer::IMSpecializedHomeDataObject interface is no longer available in release 2.0. It was replaced with the IManagedAdvancedServer::ISpecializedHomeDataObject interface. This change affects only those who were using the getConfigInfo() method for accessing the userData field in the definition of the specialized home. So, if your code was previously narrowing to the old interface like this:	All Platforms

myDO = IBOIMExtLocalToServer::IMSpecializedHomeDataObject::_narrow(theDO);

you will have to change your code to do the following:

myDO = IManagedAdvancedServer::ISpecializedHomeDataObject::_narrow(theDO);

Developing applications that use the security service	To develop applications that call the Security Service external interfaces documented in the Programming Reference:	All Platforms

Visual Age C++:

For client applications which will run as pure C++ clients, the Security Service client library, somscc1i.lib, should be linked in your client application makefile.

For server applications which will run in a C++ server process, the

Security Service server library, somscs1i.lib, should be linked in your server application makefile.

Microsoft Visual C++:

Due to a packaging problem, the Security Service client library is not available for ActiveX clients. Although the Security Service APIs are not available, ActiveX clients can still be enabled with security.

Using the Query service

- A query statement that references a struct embedded within a struct such as "SELECT x.struct1..struct2 FROM myHomeCollection x:" does not work correctly.

- Queries that use an object reference along with a DISTINCT, UNION or as an argument of a GROUP BY operation, may result in an incorrect error message or incorrect result.

  For example, the following query statement containing dept which is a reference to object of type dept, will produce an incorrect error message:

  select e.dept, avg(e.salary) from employeeHome e group by e.dept;

  To work around the problem, include the primary key attributes in the operation.

  select e.dept..deptno, avg(e.salary) from employeeHome e group by e.dept..deptno;

- When an object contains one or more relationships implemented with the foreign key pattern, both the object and the related object(s) must be configured into homes on the same server. If the homes are configured into different servers, attempting to iterate over the home collection, or using the home evaluate() method, or using the query evaluator object to query the home will fail. For further help on resolving this problem contact IBM personnel.

Handling Nulls in ESQL and Caching Services

Part 1 - Retrieval of NULLs

All Platforms

NULLs are handled primarily by the PO getters. Whenever a getter is called against an ESQL or Caching Services PO, the getter determines if the corresponding host variable (ESQL) or DAO attribute (Caching Services) is NULL and, if it is, converts the NULL into a "sentinel" value.

The sentinel values ultimately returned from the DOImpl to its consuming BO are as follows:

Type	Sentinel value
Long, Short, ...	0
Double, Float, ...	0
String	""
Char	' '
Wstring	L""
Wchar	L' '
Object Reference	NULL
Everything else	0

_____

A note about Object References

A NULL Object Reference returned by a getter is not equivalent to a NULL in the Database. A NULL Object Reference is returned whenever the getter tries to find a BO in a Home from a set of foreign key attributes from the PO, but fails due to an INoObjectWKey exception being thrown from the Home.

It is actually possible for a set of foreign key attributes including one or more sentinels resulting from NULLs to "accidentally" coincide with a legitimate Key and result in the return of a legitimate but unexpected Object Reference.

Please pay close attention to the sentinel values returned by the getters. If the sentinel values are inappropriate, the PO getter method body should be customized to return a different sentinel.

_____

Part 2 - Insertion of NULLs

When createFromPrimaryKeyString() or createFromCopyString() is driven against a Home to create a BO, any DO attributes excluded from the helper that do NOT have an initializer will be inserted as NULL into the Database. Beyond that, any data pushed into the DO from the BO will generally end up in the Database as non-NULL.

A side effect that may occur with caching Business Objects is that a NULL attribute in the Database may be replaced by a sentinel value even without the client having called the setter on the BO.

A possible workaround is for the PO setters to detect the same

sentinel values returned by the getters and converts them back to NULLs.

For future changes to address the setter issue, contact your IBM sponsor for assistance, referencing Feature 41409.

Handling Floating Point Underflow exceptions	On Windows NT, "float underflow" exceptions may appear in javai.dll. This exception occurs due to loading a VAC++ DLL into a JVM where this exception has been unmasked by DLL initialization when the JVM did not expect it to be unmasked. The unmasking occurs in DLLs compiled with VisualAge C++ prior to the 3.5.4 level	Windows NT

The JDK sets the control word to 0x27F when the VM is started and the Component Broker runtime calls FIX_FLOAT (see below) to reset the control word before every method invocation in a Java BO.

To locate all DLLs on your system that might be calling _fpreset, use the Windows NT Start->Find->Files or Folders... tool and to search for files named "*.dll" that include the text "fpreset" on all drives and subdirectories.

Examine the setting of the FPU Control Word in the idebug tool by opening the Registers monitor and select Options->Display style. This will open another pane in the Registers window for the "FP Stack". The "FPCW" field is the control register.

For JDK to run correctly, the control register should be set to something like 0x27F, with the last 5 bits masked.

The incorrect setting that _fpreset sets on DLL load is 0x362.

To avoid these exceptions, choose from the following:

- For your own Java native method DLLs written in VisualAge C++, recompile them with the VisualAge C++ 3.5.5 compiler provided with this release of Component Broker.
- If recompiling your DLLs is not possible, apply the workaround described below.
- For other Java native method DLLs that refer to _fpreset, contact the supplier and request a recompiled DLLs that doesn't use _fpreset.

There are a few DLLs supplied with Component Broker 1.3 that call _fpreset. They will be fixed in a future release of CB. The CBConnector runtime has applied the workaround described below for all DLLs provided with the product.

Workaround

Re-mask the floating point exceptions after a VisualAge C++ DLL has called _fpreset. The following VAC++ code does this remasking:

```
// code in header
#if (defined(_WIN32) && defined(__IBMCPP__))
#include <float.h>
#define FIX_FLOAT() _control87(EM_UNDERFLOW, EM_UNDERFLOW);

#else
#define FIX_FLOAT()
#endif

... //code that loads a DLL
FIX_FLOAT();
```

When using the Java "System.loadLibrary" command to load a VAC++ DLL containing Java native methods, define an additional "fix_float" static Java native method in the DLL that contains the C++ code above. Call this method after the System.loadLibrary call to reset the control word.

Handling IDL parser data type exceptions	An integer constant variable will not be coerced back to the target type if the types don't match. The following statement: const double d = 120; will cause the IDL parser to generate the following exception: "The type of the constant expression is long, but it should be double." To avoid the exception change the statement to: const double d = 120.;	All Platforms
Overriding key methods	Key methods should not be overridden for the purpose of changing managed object identity. Component Broker does not call the "isEqualToKey" or "isEqualToKeyString" methods on keys to determine if a given managed object exists in memory. As in past releases, managed object identity is determined solely by binary comparison, using the byteString form of keys (i.e. the result of calling "toString" on the keys).	All Platforms
Implementing getPrimaryKeyString for UUID keyed BOs	Either directly or indirectly attempting to get the primary key string from a UUID keyed BO will result in a "CORBA::NO_IMPLEMENT" exception from the getPrimaryKeyString method. You may be attempting to call getPrimaryKeyString directly or you may be using reference collections with UUID keyed BOs. To avoid this exception, introduce a new user-defined method on your DO interface with no parameters and a return type of	All Platforms

IManagedClient ByteString. The name of this method is up to you. The implementation of this method should contain the following:

```
::ByteString* YourDOImpl_Impl::yourMethodName()
{
  return( externalizePrimaryKeyString() );
}
```

The method externalizePrimaryKeyString() is introduced in the IBOIMExtLocalToServer::IUUIDDataObject implementation. But, since this method is not part of the standard DO interface, it is a requirement to introduce your own user-defined method that will in turn call the implementation of the externalizePrimaryKeyString() method.

Override the default implementation of getPrimarKeyString() on your BO by opening the properties for this method and selecting the radio button for "Use the implementation defined in the editor pane". Replace the default implementation with the following:

```
::ByteString* YourBO_Impl::getPrimaryKeyString()
{
  return( iDataObject->yourMethodName() );
}
```

Save your model, re-gen your code, re-build and refresh your application(s).

The getPrimaryKeyString() method calls should now return the expected result.

Working with DO Implementations in an OS/390 model	Because member names of a PDS cannot be greater than eight characters, the name of the persistent object class of a DO Implementation should be restricted to eight characters or less, if working with an OS/390 model. The class name is used to create a member on a PDS at installation time.	OS/390
Attribute of unbounded or bounded > 4000 string	If you have an attribute of unbounded or bounded > 4000 string in the Key (that is, a string whose size is not specified, or a string that is greater than 4000), you may encounter a warning message when you create a PO Schema for the DO. The message indicates that there is no mapping helper for the attribute and prompts you to provide one. This is because the string attribute does not have appropriate size information.	All Platforms

The message box brings you to the Attributes Mapping page. Click **Yes** to provide a mapping helper, then click on the data object attribute that requires a mapping helper. Type:

DB2MappingHelper/stringToVarChar/varCharToString

as the mapping helper class name/DO to PO method/PO to DO method. Finally, click on the mapped PO attribute and enter a length in the SQL type of the schema column.

Using nested sequences

Invoking methods in Java on a proxy of an object residing in a C++ server may result in a sequence of zero length, if the methods return OUT or RETURN parameters containing nested sequences .

All Platforms

To avoid this problem, modify the IDL and remove nested sequences from use as OUT and RETURN parameters. This is a known restriction that will be addressed in a future release.

Using non-IBM ORBs

The following paragraphs are clarifications to the Advanced Programming Guide, "Non-IBM ORB Usage" chapter, "Additional Tips for Non-IBM ORB Usage section", "CORBA IIOP" subsection:"

All Platforms

Setting SOMDGETENV indicates that the server first looks at the environment variables for its configuration settings. SOMDGETENV must be set before the server can find GENERATE_IIOP10_OBJREFS. But be aware that the server may also find other configuration settings in the list of environment variables. If these variables are not what the server expects, the server may not be configured correctly.

For example, be sure that the HOSTNAME environment variable has a value that is meaningful to the server. Sometimes HOSTNAME="localhost". "localhost" is not meaningful to the server. Unless this is changed to the actual host name of the server machine, the server is configured such that no client will communicate with it.

Policy Has Moved

You previously were able to use the following call:

org.omg.CORBA.Object[] objs = factoryFinder.find_factories (name);

where name was:

{{"Policy", "object interface"}}.

Now use the following name:

{{"Policy", "object interface"}, {"policyBOIMSpecializedFactory", "object home"}}.

The method Bootstrap.factory must take a NameComponent array as a parameter instead of two strings; the code that uses those strings must be changed to use the NameComponent array:

```
                          public GenericFactory factory (NameComponent[] name)
                          {
                             try
                             {
                                // Resolve to the factory finder
                                NameComponent[] ffName = new NameComponent [4];
                                ffName[0] = new NameComponent ("host", "");
                                ffName[1] = new NameComponent ("resources", "");
                                ffName[2] = new NameComponent ("factory-finders","");
                                ffName[3] = new NameComponent ("host-scope", "");
                                org.omg.CORBA.Object obj = rootNC.resolve (ffName);
                                factoryFinder ff = FactoryFinderHelper.narrow (obj);
                                // Resolve to the factory
                                org.omg.CORBA.Object[] objs = ff.find_factories(name);
                                return GenericFactoryHelper.narrow (objs[0]);
                             }
                             catch (Exception e)
                             {
                                System.out.println ("Getting the factory finder FAILED.");
                                e.printStackTrace ();
                                return null;
                             }
                          } // factory
```

And since PolicyTest calls Bootstrap.factory, PolicyTest must change as well:

```
import java.net.URL;
import CosLifeCycle.GenericFactory;
import CosNaming.NameComponent;
public class PolicyTest
{
   public static void main (String[] args)
   {
    try
      {
         URL url = new
URL("file:///e:/vbroker.client/NameService.ior");
         Bootstrap bs = new Bootstrap (url, null);
         // Construct the full name of the Policy object.
         NameComponent[] name = new NameComponent [2];
         name[0] = new NameComponent("Policy", "object
interface");
         name[1] = new
NameComponent("policyBOIMSpecializedFactory", "object home");
         GenericFactory gf = bs.factory (name);
         PolicyHome ph = PolicyHomeHelper.narrow (gf);
         org.omg.CORBA.Object obj = ph.create ((float)100.00,
(float) 10000.00);
         Policy p = PolicyHelper.narrow (obj);
         System.out.println ("Policy number = " + p.policyNo ());
      }
    catch (Exception e)
```

```
                    {
                        e.printStackTrace ();
                    }
                }
            } // class PolicyTest
```

Component Broker Connector

General

Using DB2 and UDB clients on AIX	DB2 and UDB clients on AIX can not successfully connect to a UDB Server database via the XA interfaces used by Component Broker. The failed connection results in a handleSignal entry in the Component Broker activity.log with libdb2lai.a at the top of the call stack.	AIX

A workaround for this problem is to rename the libdb2lai.a file on your UDB client machine as follows:

1. Stop all application servers
2. db2 force applications all
3. db2stop
4. cd $HOME/sqllib/lib
5. mv libdb2lai.a libdb2lai.a.orig
6. db2start

Using UDB Version 5.0 clients with Version 5.2 servers	Attempting to use a UDB Version 5.0 client to a UDB Version 5.2 server will result in an error message like:	All Platforms

> *"/var/CBConnector/aixApps/ClaimAppFam/ClaimPO.bnd" is not a valid bind file.*

If you have installed UDB Version 5.2 on your server, then the bind files that are produced can only be handled by Version 5.2 clients. If you use Version 5.0 bind files, they should be used with version 5.0 clients.

Therefore, if you have installed UDB Version 5.2 on your DB2 Servers, you will need to use the Version 5.2 clients that came with your UDB Version 5.2 package instead of the Version 5.0 clients that were shipped on the Component Broker Supplemental CD.

Sharing Component Broker and MQ Series login IDs	Using a CBConnector user ID to start MQ queues will result in the inability to use some features of MQ. For example, the user will subsequently be unable to create a queue or start a queue manager.	AIX

To avoid this problem, login under a user ID (not a CBConnector ID) to start the MQ queues, and then switch to different user ID to run the CB application.

Managing the AIX semaphores	In this release for AIX, Component Broker Connector does not free semaphores correctly. Over time, the system-wide pool of 4096 semaphores will be exhausted, causing system failures. To manage the semaphores, do the following steps while the system is quiesed: 1. Use the **ipcs -s** command to capture a list of semaphores in use before starting any Component Broker processes. 2. After running Component Broker, use the shell script **ipcrmls** provided by Component Broker. This will create a list of commands to delete ALL the semaphores in the system. Do not run this list of commands. 3. Edit the list of commands created in step 2 to remove the commands deleting semaphores you want to keep. The list created in step 1 contains semaphores that should be kept. 4. Execute the remaining commands to remove the semaphores allocated by Component Broker.	AIX
Stopping DCE gracefully	DCE does not always clean up its state properly when Windows NT is shut down. Consequently upon restart, DCE does not start all of its daemons properly. Before shutting down, issue the following DCE commands from a DCE command line: *>>dcecp stop.dce* *>>dcecp clean.dce*	Windows NT
Managing IE exceptions	Microsoft's Internet Explorer may cause some AppletRAS related exception messages when running a CB Applet . AppletRAS is attempting to create a log file for the Applet. If IE's JVM has not been configured (relaxed) to allow local file system access, these messages may appear. These messages may be ignored.	Windows NT

Systems Management

Setting the LANG environment variable	Setting the LANG environment variable to "C" during initial configuration of AIX will result in multiple "message text could not be found" messages. Change the setting to en_US to properly link to the message library.	AIX
Setting the server locale	To change the code page under which the C++ ORB in the server runs, the LC_ALL environment variable must be set to the desired locale in the server's process before starting the server. On NT, this environment variable can be set using the Control Panel. On AIX, the variable can be set in the environment before starting the System Management End User Interface (EUI).	All Platforms

The locale should be specified as "<Language>_<Territory>.<Codeset>". For example, setting LC_ALL to "en_US.IBM-1252" says the Language is English, the Territory is the United States, and the Codeset is IBM-1252.

When the C++ ORB initializes, it will set the codepage to the value specified by LC_ALL.

Configuring an application to use the Cache Service when the Managed Object Class name is greater than 32 bytes	Following the instructions in the System Administration Guide pertaining to Configuring an Application to use the Cache Service will generate an error message indicating that the Managed Object Class Name could not be created when creating a Managed Object Class with a name greater than 32 bytes. The System Managed EUI has a limit of 32 characters for the name of the Managed Object Class. To enable the Cache Service when a Managed Object Class is greater than 32 bytes, the following can be done.	All Platforms

Instead of configuring the application in the Configuration model, configure the application for a particular host.

To configure the data cache options for an application, complete the following steps:

1. Change the SM EUI View to Super User.
2. Expand the server that contains your application and server. Expand Host Images-><host> ->Server Images -> <server name>.
3. Create a new Profile Image. It must have a name of iDefault. If the Profile image already exists, omit this step.
4. Edit the iDefault Profile Image to set the cache size attribute to the default number of 1K blocks of memory to be allocated for all data caches.
5. Create one or more Profile Class Images. Each Profile Class Image defines different characteristics for the data cache, for use by different managed object classes.

Relate the profile created in the profile classes that are to be used, by completing the following steps:
6. Select the Profile Class Images created in step 5.
7. Click Selected - Drag
8. Expand the iDefault Profile Image created in step 3. On the pop-up menu of Configured Profile Classes, click Create relationship. This creates a Configured Profile Classes relationship between the Profile Image and the Profile Class Images.
9. Edit each Profile Class Image to edit and set the access, defer updates, lock confidence and refresh interval attributes.

Relate each Profile Class Image to the Managed Object Class Images that are to use the cache configuration, by completing the following steps:

10. Select the Managed Object Class Images for one Profile Class Image.
11. Click Selected - Drag.
12. Expand the Profile Class Image (created in step 5). On the pop-up menu of Configured Managed Object Classes, click Create relationship. This creates the configured Managed Object Classes relationship between the Profile Class Image and the Managed Object Class Images.
13. Repeat steps 10 to 12 for each additional Profile Class Image.
14. Restart the server for the caching configuration to take effect.

Terminating and reactivating secured servers	With the server security enabled, if a server is terminated and reactivated after server configuration changes, the following entries may be logged in the activity.log of the reactivated server: *SECURITY ERROR: Session entry that was created previously is not found.* *CORBA::INTERNAL exception:* *SOM_SECURITY_INTERNAL_SECURITY_ERROR.* This message reports a security checking problem between the System Management agent and the reactivated server. However, it should not impact the messaging between the application clients and the application servers. After this error, the System Management agent can not monitor or interact with the reactivated server. For example, the System Management agent can not report the health of the reactivated server. To recover, restart System Management and activate the servers.	AIX
Refreshing basic and specific applications	Refresh of "basic" or "specific" applications works only if there is no change to the corresponding .ddl file. In other words, if you just modify one or more .dll or .jar images, then refreshing "basic" or "specific" applications should still work. Due to interdependent relationships between "basic" and "specific" applications, refreshing of "basic" and "specific" application families is not possible in Component Broker Release 2.0 when a .ddl file corresponding to that application is modified. If you modify the .ddl file, perform the following steps. To refresh a "basic" application family:	All Platforms

1. Uninstall the "basic" application family which also automatically uninstalls the "specific" application family.
2. Load the "basic" application family and "specific" application family
3. Configure both "basic" and "specific" applications on the corresponding servers.
4. Re-activate the configuration.

To refresh a "specific" application family:

1. Uninstall the "specific" application family only
2. Load the "specific" application family
3. Configure the "specific" application on the corresponding servers.
4. Re-activate the configuration.

Setting CLASSPATH in the SMEUI	In the SMEUI, the CLASSPATH that is displayed on the "Java Virtual Machine" page of the Object Editor only supports the deletion of text. To put additional files into the class path, modify the CLASSPATH environment variable, restart the CBConnector service and reactivate the configuration.	All Platforms
Handling server restart loops	System Management can get into an infinite loop attempting to restart a server, if system exceptions (SEGV, FP_ERROR, INVALID_OP etc) thrown by a server thread are not caught by a user defined exception handler, AND if Dr. Watson or an equivalent debugger is configured to catch these exceptions. If this happens, clicking on the 'OK' button of the Dr. Watson pop-up will allow a successful restart of the server. To avoid the potential looping, do any one of the following: • Code exception handlers for all system exceptions. • Configure servers as "runOnRequest" rather than "runImmediate." • Disable Dr. Watson on the server machine.	All Platforms
Enabling a server with SSL security	The procedure to enable a server with SSL security in Appendix E of the Planning, Performance and Installation Guide is incorrect. Please use the following procedure in place of Appendix E. "Enabling the Security Service, Activate SSL Security Enablement for Servers in a Server Group": 1.From the System Manager User Interface, select View > User Level > Expert 2.Configure the server keyring file for each server image. Expand Host Images > myhost > Server Images.	All Platforms

3.Open the pop-up menu for Server Images.

4.Select Edit. This opens the Object Editor.

5.For each configured server image:

- Select the Security tab to check the SSL keyring file field. (Note a default path and keyring file name have already been defined for each configured server image. You can only use the default path and name.)

  You can also select to use the default server keyring file "CBDevTest.kdb" or create your own server keyring file (Please note that "CBDevTest.kdb" keyring file is provided for test only. Using this file with any application that is in deployment or production, will lead to serious security exposure. Refer to the "Use the Test Certificate Provided by Component Broker" section of the Component Broker for Windows NT and AIX System Administration Guide for more details.)

- If you select to use the default server keyring file (CBDevTest.kdb), do the following:
- 1.Do not edit the SSL Keyring file field.
  2.Go to the directory defined in the SSL Keyring file field.
  3.Inside the directory, you should find CBDevTest.kdb.
  4.Copy CBDevTest.kdb to the default file name that is defined in the SSL Keyring file field.
- If you select to use your own server keyring file, do the following:
- 1.Do not edit the SSL Keyring file field.
  2.Go to the directory defined in the SSL Keyring file field.
  3.Copy your own server keyring file to the default file name that is defined in the SSL Keyring file field.

6.Expand Management Zones > Sample Cell and Work Group Zone > Configurations > Sample Configuration.

7.Open the pop-up menu of the Sample Configuration and select Activate.

Activating multi-host or multi-zone configurations	If while performing multi-host and/or multi-zone activations, either the name server and/or the application servers fail to start, it could be caused by an orb request timeout.	AIX
	If this happens, try increasing the orb request timeout to 300 from the default value of 30 for name servers and application servers in the model world and re-trying activation.	
Reconfiguring or uninstalling System	In a CB configuration, if an AIX System Management Agent is reconfigured or uninstalled, definitions of the reconfigured or	AIX

Management Agent   uninstalled Agent may still exist in the cache of the System Manager, whether the System Manager is on NT or AIX. These definitions have to be completely removed before adding this machine back into the CB network. An attempt to configure the host into the CB network will fail because the System Manager wrongly assumes that the host is already available in the CB network.

To work around this problem, the remote System Manager host's cache has to be flushed by bringing the process down and back up using the following steps:

RECONFIGURE

1. On the AIX System Management Agent, run the reconfiguration to delete the Host Object. This step will generate an error like:

Warning: HostImage cbtest1.austin.ibm.com already exists on SMappl system.

This ensures that the Host Object has been deleted. Proceed to step 2.

UNINSTALL

1. Uninstall the System Management Agent.

(Note: At this time, don't attempt to reinstall the Agent and do the configuration step. Doing so will generate an error like:

Warning: HostImage cbtest1.austin.ibm.com already exists on SMappl system)

Proceed to step 2.

(Steps 2 and 3 are different depending on whether the System Manager is running on NT or AIX.)

System Manager on NT:

2. Stop the CBConnector service on the remote SM Host machine.

3. NT: Restart the CBConnector service on the remote SM Host machine. Proceed to step 4.

System Manager on AIX:

2. Stop the System Management Application using smit on the remote SM Host machine.
3. Start the System Management Application using smit on the remote SM Host machine.

4. Start the reinstall and/or (re)configure process for the System

Management Agent.

Using the Server Group Gateway (SGGW) Server Affinity Management with Security	An enhancement in this release allows the workload of multiple clients to be distributed across all the servers in a server group, rather than routing all requests to just one target server. The server affinity management capability of the SGGW server has been enhanced to exploit the existence of security service context information accompanying each client request. In this environment the SGGW server uses the client principal to distinguish between clients and will maintain a separate client affinity mapping for each client principal. To use this capability, enable the Security Service and ensure that each end-user authenticates with their own principal (userid). Consult the System Administration Guide for information on the use of DCE- and SSL-based security. (Note: The 2.0 System Administration Guide does not reflect this affinity mapping capability.)	All Platforms

Server

Understanding server hangs on AIX	At server startup, a hang may occur due to an initializer not completing within its allocated time. If the Activity Log shows the server waiting on an initializer, the initializer is not completing its initialization. Contact your IBM sponsor for assistance if you observe this problem.	AIX

Clients

General

Using workload management for direct client access to persistent objects	If you directly access a workload managed persistent object from a client application the application may receive a CORBA::NO_IMPLEMENT exception. An activity log entry will state the following: *"The function OTSXAXId::OTSXAXId(OTSSRMGlobalId*):175 raised CORBA exception CORBA::INTERNAL, error code is 0x494202F2 SOMTRRAS::Minor_invalidState."* You should not see this problem if the length of the generated name for the Sgcp server does not exceed 29 characters. This problem is not known to affect access to workload managed persistent objects from another business object.	All Platforms

C++ Client

Using WLM with C++ Clients	In order for a C++ Client application to benefit from the WLM enhancements it must execute on a host where DCE is installed. If you run a C++ Client application on a host which does not have DCE installed, you may get the following error message: *libdce.dll can't be loaded* If this happens, update the default settings of the relevant Client Style to disable the ORB's workload management extension. See the related tasks information in the section "Workload Management (WLM)-Enhanced clients" in the System Administration Guide.	All Platforms

ActiveX Client

Accessing AIX servers	In this release, an ActiveX client does not communicate with an AIX server.	Windows NT

Java Client

Compiling IDL files	Customer-written IDL files that #include IBM-provided IDL files such as orb.idl, Cos<xxx>.idl, IExtended<xxx>.idl may cause the IDL-Java compiler to issue error messages to the effect of "IDL file not found" if the compiler is run on a machine on which only the Java client SDK installation option was chosen. This is because that installation option does not install the global set of IBM-provided IDL files on the target machine in this release. There are two workarounds available: • Copy the contents of the include directory (or at least the *.idl files in that directory) from a C++ client machine or a server machine, to a suitable directory on the Java client development machine. (This only needs to be done once per Java client development machine.) When running the IDL-Java compiler, instruct the compiler to search that directory for #include files by specifying that directory after the -I option, for example: *java com.ibm.idl.toJava.Compiler -I <dirname> <other options> <idl file>* or, • Copy the customer IDL file to a machine on which the C++ client is installed, or a machine on which a Component Broker server is installed, and emit the Java bindings using the above syntax using the -I option. Then copy the resulting .java files back to the Java client development machine.	Java

Using Netscape Navigator	When running a Component Broker Java client applet in more than one Netscape browser window simultaneously on the same client machine, the applet may experience Java exceptions or hang. To avoid this problem, run only one Java client applet at a time from a Netscape browser window.	Java

Java Virtual Machine Tuning	Currently, the Java Virtual Machine parameters for:
	
	   native stack size
	   java stack size
	   minimum heap size
	   maximum heap size
	
	are in units of megabytes. When tuning these parameters, especially the minimum heap size, it is sometimes necessary to set the parameter to less than 1 megabyte.
	
	The work-around is to set the parameter to a value greater than or equal to 100000 so that it is interpreted in units of bytes. For example, to set the minimum heap size to 128K, enter 131072. To reset it back to 1 megabyte, enter 1.
		Java

Using SSL-enabled servers	When receiving a large response buffer (approximately 17,260 or more bytes) from an SSL-enabled server, the Java Client may throw a COMM_FAILURE exception. The workaround for release 2.0 will be specific to the application, but the goal is to limit the amount of data sent back in responses from the server to less than 17,260 bytes. For instance, if your application requests a sequence of 1000 structures and you receive the COMM_FAILURE, try reducing the number of structures requested in each request and make more requests.	Java

CICS / IMS Application Adaptor

Installing and configuring CICS and IMS and the prerequisites	For assistance with CICS and IMS configuration and problem determination, and the prerequisites for attaching to CICS and IMS, see the Component Broker support page at: http://service.boulder.ibm.com/ Select the Component Broker link from the home page.	All Platforms
Using HOD with IMS	For servers configured with objects that attach to IMS using HOD, there is a restriction of one concurrent client per server	Windows NT
Using APPC with IMS	Attaching to IMS with APPC has some limitations. For specifics check the Component Broker support page at: http://service.boulder.ibm.com/	All Platforms

Select the Component Broker link from the home page.

| *Running the CICS HOD sample without modifying the IVP* | The PAA sample for CICS Host-On-Demand is based on the CICS Installation Verification Procedure (IVP). The full sample requires two new transactions to be added to the IVP for delete (DELE) and debit (DEBT). | All Platforms |

If you want to try out the full sample, you must install the IVP modifications as described in Appendix C of the CICS and IMS Application Adaptor Quick Beginnings. However, the IVP modifications can ONLY be installed on Transaction Server for Windows NT or Transaction Server for AIX. If your server is CICS/ESA or if you do not want to modify the CICS IVP, you can try out the partial CICS-HOD sample. The partial sample only uses the add, update, and retrieve methods, and it does not require any changes to be made to the server CICS IVP.

Chapter 6. "Developing a CICS-HOD Application" describes the steps you should follow to build the full sample (requiring the CICS IVP modifications). If you want to build a partial sample (using the unmodified IVP), you should follow Chapter 6 with the following changes.

- In the section "Creating the DFHDGA Record Type and Record Bean": instead of filename "dfhdga.bms", use filename "dfhdga2.bms"
- Omit the section "Creating the CmdMenuToMenuDelDebit Command".
- Omit the section "Adding Features to the CmdMenuToMenuDelDebit Command".
- Omit the section "Creating the NavigatorDelDebit Navigator".
- In the section "Editing the MenuCustomer::debit method", use this code:

```
public void debit(int amount) {
    throw new RuntimeException("No implementation.");
}
```

- In the section "Editing the MenuCustomer::del method", use this code:

```
public void del() throws com.ibm.ipaa.IDataKeyNotFoundException {
    throw new RuntimeException("No implementation.");
}
```

- In the section "Unit Testing the EAB Object" you should omit the block of code following the comment: // ATTEMPT DEBIT

Editing APPC connection objects	In the SMEUI Object Editor, while editing some of the fields of an APPC Connnection under	AIX

Management Zones -> Configurations -> APPC Connections

the user may get the following error ...

"The function could not be executed because the specified size is inadequate to contain the text"

... not allowing the user to edit the value of those fields.

To circumvent this problem use the "Full Panel View" mode of the Object Editor by choosing ...

Options -> View Type -> Full Panel View

Running CICS/APPC applications	Running PAA APPC on AIX requires the xlC_rte fileset to be at level 3.1.4.7 or above, but Component Broker documentation currently only prereqs level 3.1.4.4. Running CICS/APPC applications using level 3.1.4.4 of the xlC_rte fileset may result in a segment violation in OTSAPPCLibrary::instance	AIX

The activity log will contain:

PrimaryMessage: The function OTSAPPCLibrary::instance():203 received CORBA exception CORBA::UNKNOWN.
ExtendedMessage:

followed by a Handle Signal and the following activity log entry:

PrimaryMessage: The function APPCConnection_OTSAPPCConnection_Impl::connectOptimistic(const char,const char*,const char*,APPCConnection::OTSAPPCConnection::programType,const char*,const char*,const char*,const char*):536 reported an activity.*
ExtendedMessage: Connection 'APPC_ACashAcct_Server' contains unsupported values. Use the systems management EUI to correct them.

This is caused by a known problem in the C++ Runtime for AIX component xlC.rte which is fixed by level 3.1.4.7 or later.

For assistance in obtaining an appropriate C++ Runtime, contact your IBM sponsor.

Using endResource() methods in Business Objects	For Business objects backed by CICS or IMS and using session services, the user can select that the business object implementation provide an implementation of the ISession::Resource::endResource() by selecting the Object Builder "Provides end resource" check box on the "Name and Data Access Pattern" smart guide. Making this selection allows the BO	All Platforms

implementer to write code which will be called when sessions end.

This feature is not supported in this release. If "Provides end resource" is checked, the code generated for C++ implementations will not be called and the code generated for Java implementations will not build correctly.

Using "sessionReset" for PAA sessions over ECI	In this release there are known problems with "sessionReset" for PAA sessions using ECI connections. The problems occur when a session reset is expected to undo a previous operation that was applied to the 3rd tier. For example, a createFromPrimaryKeyString followed by a session reset is expected to leave the system looking like the managed object was never created. Similarly, a remove followed by a session reset is expected to leave the system looking like the managed object still exists. In R2.0 scenarios like this do not work correctly. You will get unexpected results. There is no known workaround for these problems. Please avoid using sessionReset until a fix can be supplied. For future changes to address this issue, contact your IBM sponsor for assistance, referencing Feature 54188.	All Platforms
Using Host On-Demand on AIX	HOD connections will not work on AIX.	AIX
Exporting .jar files from VisualAge for Java	Compiling Object Builder artifacts that reference the beans exported from VisualAge for Java can result in an "I/O error." To avoid this problem, when you export a .jar file from VisualAge for Java for consumption by Object Builder, use the default options - do not select the option to export .java source files	All Platforms
Handling embedded NULLs	CORBA IDL does not allow embedded NULLs in Strings or Wstrings. However, values retrieved from CICS via ECI may contain such NULLs. A value containing an embedded NULL will generate a DATA_CONVERSION message. To handle this case, a converter routine is needed to either truncate the String at the embedded NULL or to replace the NULL with some other character. The following sample demonstrates what such a converter routine must do:	Windows NT

```
public class MyConverter {
  public static String trimNull(String instr) {
    String retval = instr;
    int ind = instr.indexOf(0);
    if (ind != -1)
      retval = instr.substring(0,ind);
    return retval;
```

```
      };
    public static String replaceNull(String instr, char newchar) {
       String retval = instr.replace(java.lang.Character.forDigit(0,0),newchar);
       return retval;
     };
};
```

This routine must be placed into the **CLASSPATH** as MyConverter.java and compiled using the Java compiler.

To use this converter, do the following steps:

1. Add the following line to the yyyyyPAO.java file:
   import MyConverter;
2. In the PAO, under the retrieve method, and after the data has been successfully retrieved, modify the fields to be converted by adding the following code to replace the null with a space:
   setXxxxx(MyConverter.replaceNull(getXxxxx(),' '));
   or to trim the string at the null:
   setXxxxx(MyConverter.trimNull(getXxxxx()));
   where Xxxx is the name of the PAO attribute.

Managing connections	If a CRUD method is called outside of the scope of a Session, a connection is allocated during the execution of the method. At the conclusion of the method, the connection is marked for garbage collection by the Java virtual machine. Since the connection may not be garbage collected immediately, it will count towards the connection limit. Especially for IMS connections which allow only one connection at a time, this causes subsequent CRUD method calls to hang while waiting for a connection. To avoid this problem, always call a CRUD method within the scope of a Session.	Windows NT
Installing an application on multiple servers that uses APPC	An application that uses APPC communication must be configured with the Local Logical Unit (LU) name of the server. Since multiple servers cannot share the same LU name, additional configuration must be done after server activation. Once activation is complete, modify one of the servers' image to specify a different LU name for the APPC connection configured for that server.	Windows NT
Handling initial 3270 screen navigation	When using 3270 communications, it is possible for the navigation script's sends and receives to get out of sequence, i.e. a reply is sent before the request is received. The first interaction of a navigation script normally assumes the initial screen will be "clear" and that the script will initiate a CICS transaction by sending a script including a transaction identification.	Windows NT

If the initial screen is not "clear", the problem may appear, depending upon system response times.

To force a wait for the initial screen to be fully received, set the ScreenOnConnect attribute of the HODConnectionSpec to true. The technique for setting parameters of the HODConnectionSpec is described in the subsection "Define on a Procedural Adaptor Object" contained under the section "Connection Specifications in the Unit Test Environment" in the *CICS and IMS Application Adaptor Quick Beginnings* manual.

Using Optimistic inserts or deletes	In this release, optimistic locking over APPC for insert or delete operations is not supported.	Windows NT
Understanding non-zero IMS return codes	The tier3 IMS console will show non-zero return codes (such as 18 and 4) when the transaction in the IMS APPC sample completes against the IMS phonebook IVP. These return codes result from differences between the way Component Broker and IMS commit transactions and deallocate conversations.	Windows NT
	Similar non-zero return codes may occur with other IMS transactions.	
	These return codes may be ignored, since they have no effect on the processing.	

Oracle Application Adaptor

Using Oracle databases	When installed on NT, the Oracle Application Adaptor (OAA) can be used to access local and remote Oracle databases installed on NT. It can be used to access remote Oracle databases installed on AIX. The OAA will not work on AIX. For information on the status of this feature, please contact your IBM sponsor, referencing item 56051.	AIX

Known DBCS Restrictions

Installation and Configuration

Using English VisualAge C++ on the Japanese NT 4.0	When using the English version of VAC++ for Windows 3.5 with Fixpack 3 on a Japanese Windows NT 4.0 system, please add the following lines at the end of IPF.INI, which is in \IBMCPPW\BIN, in order to display DBCS characters in IPF .HLP files.	Windows NT

```
;IPF Display Fonts
[ipfdspfonts]
ProportionalFont="System"
MonospaceFont="Terminal"

;IPF Print Fonts
[ipfprtfonts]
ProportionalFont="System"
MonospaceFont="Terminal"
```

Installing JDK 1.1.6	JDK 1.1.6 or above is required for Java client runtime, but depending on the package, when installing the JDK software on a Windows 95 system, the installer may offer you to install Winsock 2. Please be careful when you get this message, as some Windows configurations may be seriously affected by this Winsock 2. Please read the readme document that comes with JDK 1.1.6 for more information. This readme also provides information on how you can remove Winsock 2 that had been installed by JDK 1.1.6 installer.	Windows 95
Entering a domain user during installation	On the installation panel titled "Select User ID and Enter Password for Component Broker Service," the prompt text says "To enter domain user, select <Enter_domain_user> from the list," However, the drop down list box does not contain this selection. Instead you should select <Enter_your_own_user>.	Windows NT

Component Broker Toolkit

Tools

Marking in the Local Debugger	Marking is done byte-by-byte, not character-by-character on the user interface.	Windows NT
Specifying an editor to invoke from Object Builder	Editing \CBroker\bin\sqledit.bat (Windows NT) or sqledit.sh (AIX) enables you to invoke an editor you prefer to use from "Open in Editor" menu in Object Builder. If you would like to use the key stroke recording function with DBCS characters, it is advised that you edit this file to invoke an editor that supports the key stroke recording with DBCS characters.	All Platforms
Mapping wstring/wchar attributes in a DO with DB2 CHARACTER	Object Builder maps wstring/wchar IDL types in the Data Object with LONG VARGRAPHIC column type in the Persistant Object by default. When you are creating an application that involves wstring data, and needs to store persistent object data in a CHARACTER column in a DB2 table, you need to write your own mapping helper (a sample is given	All Platforms

column type in a
PO

below), and follow the procedure described in "Map a Data Object to a
Persistent Object" section of the on-line "Tasks" book.

```
/*****************************************************
 Name: myMappingHelper.hpp
 Description: A mapping helper class for mapping a WString_var DO
data to
  char* PO data, and vice versa.
   *****************************************************/
#include <stdlib.h>  // mbstowcs(), wcstombs()
#include <string.h>  // strcat()
#define DEBUG

class myMappingHelper {
 public:
   // Conversion from WString_var to char*
   static void wstringToString(CORBA::WString_var& wszData, char*
szData);

   // Conversion from char* to Wstring_var
   static void stringToWstring(const char* szData,
CORBA::WString_var& wszData);
};

inline void
myMappingHelper::wstringToString(CORBA::WString_var& wszData,
char* szData)
{
    size_t iRc, iRc2;

    if (wszData == NULL){
     strcpy (szData, ""); // This behavior, when the passed pointer is a
                     // NULL pointer, is user dependent.
    }else{
      // Count the number of bytes to expect in the result char* string.
      iRc = wcstombs(NULL, wszData, 0);

      if (iRc < 0)          // This behavior, when the conversion is not
        strcpy(szData, ""); // possible, is user dependent.
      else{
        iRc2 = wcstombs(szData, wszData, iRc);
        szData[iRc2] = '\0';
      }
    } // end of if (wszData == NULL)
}; // end of wstringToString()

inline void myMappingHelper::stringToWstring(const char* szData,
CORBA::WString_var& wszData)
{
    size_t iRc, iRc2;
    int i;
```

```
                char* szPtr1;

              if (szData == NULL){
                wcscpy(wszData, L"\0");
              }else{
                /* Removing trailing blanks that DB2 inserted.  This procedure is
                   optional, but if you choose not to remove them, please keep the
                   string length from exceeding the buffer length.
                   Another option is to use IString::stripTrailingBlanks function.*/

                szPtr1 = (char*)szData;    // Point to the beginning of the buffer.
                for (; *szPtr1; ++szPtr1); // Go to the end of the buffer.
                szPtr1--;
                for (; szData <= szPtr1 && *szPtr1 == 0x20; --szPtr1); // Move
back until the pointer
                                              // points to the end of the data.
                memset(szPtr1+1, '\0', 1);   // Terminate the string.
                // End of removing trailing blanks

                // Count the number of wide characters to expect in the
                // result wide string.
                iRc = mbstowcs(NULL, szData, 0);

                if (iRc < 0)          // This behavior, when the conversion is not
                  wcscpy(wszData, L""); // possible, is user dependent.
                else{
              wszData = ::CORBA::wstring_alloc(iRc);  // Allocate a buffer for the
            wide string of size iRc.
              for (i=0; i < iRc; i++) // Initialize the buffer
                wszData[i] = L'\0';
                  iRc2=mbstowcs(wszData, szData, iRc); // Do the conversion
              wszData[(int)iRc2] = L'\0'; // NULL terminate
                }
              } // end of if (wszData == NULL)
            }; // end of stringToWstring
```

Displaying DBCS characters on the Object Builder tool and IRBrowser	When the locale is set to one of the DBCS languages' locales, characters in Object Builder and IRBrowser may appear in a very large font. In order to avoid large characters, set an environment variable, JAVA_FONT_OFFSET=0.	AIX
Use of the vendor selected characters in Object Builder	Due to a restriction in JDK, in Object Builder, users are restricted from using the NEC selected characters, IBM selected characters and NEC's IBM selected characters. This includes the characters in the files to be imported, and the characters entered as comments in a model.	All Platforms
Viewing the messages in the SmartGuides	Some SmartGuides in Object Builder may be too small by default to display all the translated messages, in which case you need to resize the SmartGuide to make it large enough to contain all the messages.	All Platforms

Inputting DBCS characters in Object Builder	In this release, due to a restriction in JDK, DBCS input in Object Builder is limited. Specifically, when you enter DBCS characters, the interim Kana-Kan conversion window will not appear on AIX 4.2.1. If you hit Enter key to confirm your DBCS input, though, the final input will still appear on the target object.	AIX

Programming Model

Transmitting DBCS characters between ORBs	When characters other than those in the ISO Latin-1 character set are used in the data during transmission, the proper IDL types to use are wchar and wstring, not char and string types. C++ mapping of those attributes are wchar_t and wchar_t*, respectively.	All Platforms
Naming identifiers	DBCS characters are not permitted in the following places: • directory names and file names • object names - identifier names - application family names - containers • literals in IDL	All Platforms
Entering information in the State fields of the Personal Life Insurance Application	Only capital letters are allowed in the "State" fields in the Graphical User Interface of the Personal Life Insurance Application. Lower case letters are not allowed.	Windows NT

Component Broker Connector

Server

Naming objects in the system manager	DBCS are not permitted in the names of objects.	All Platforms
Using the File Browser	File Browser contains a time stamp information of GMT -05:00. Action menu of the File Browser is inactive.	Windows NT
Searching objects in the Interface Repository Browser	Use single byte alphanumeric characters for searching objects using the search window in IR Browser.	All Platforms
Viewing Interface Repository Help	If you do not have your web browser executable ("netscape" or "iexplore") in your path, you must specify the full path name of the web browser	All Platforms

	executable in the Help Setup window. To bring up the Help Setup window, select "Options" from the IR Browser menu bar and then select the "Help Setup..." menu item.	
Using mixed code sets	For this release, both the server and client applications must use the same code set, either an ASCII code set or the Unicode code set. If this is not done, the user's applications must do the appropriate code set conversion.	All Platforms
Entering data into the Personal Life Insurance Application	A maximum of 15 characters can be entered in the entry fields of the Personal Life Insurance Application Graphical User Interface.	Windows NT
Displaying Help Text	If you are using the Microsoft Internet Explorer, then Version 4.0 or later must be used in order to display some of the help panels. If you do not use this version, you will get an error saying that the specified file could not be found.	Windows NT
Time stamp information in the ction Console	In this release, when you run the System Manager EUI in a Japanese locale, the time stamp information in the Action Console contains the year indicated incorrectly as ".0".	AIX
Showing the Error Log on the System Management EUI	In this release, when you try to display the file browser for the error log by choosing "Browse" from the menu of the "Latest" object in the Error Log folder, it may result in a "Client Timeout Error" message, which subsequently may cause the System Management EUI to freeze. If this happens, the Error Log can be viewed by entering the following command on the command line: `showlog -nt -d > errorlog.out`	NT
Making selections in the Component Trace window of the System Management EUI Object Editor	In this release, the vertical scrool bar and the right arror on the horizontal scroll bar are not displayed in the Component Trace window of the System Management EUI Object Editor. To scroll vertically, click on an item in the list and use the cursor keys to see the other items in the list. To scroll horizontally, use the mouse to slide the scroll bar. In addition, the down-arrow control for the drop down list box of the Attribute Value field is not displayed. For this release, the contents of this field cannot be changed.	AIX
Displaying the vertical scroll bar in the System Management EUI Object Editor	In this release, the vertical scroll bar is not displayed when the System Management EUI Object Editor is opened. In order to see the scroll	AIX

bar, you must resize the window to make it
wider.

CICS/IMS Application Adaptor

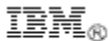|		
Setting CLASSPATH for Visual Age for Java classes	When you create a COMMAREA bean and run the application outside VisualAge Java with DBCS strings, if you specified the server code page to be one of the DBCS code pages, such as 932 or 939, you may receive a message that looks like this: "Exception from Cash Account Insert com.ibm.record.RecordConversionFailureException: IVJC0216E: The following Java string <DBCS string> cannot be converted to COBOL PIC X(1200) string." Alternatively, after you import such bean into Object Builder and create a client/server application, when you execute the CB client application, you may receive a CORBA::MARSHAL exception. In either case, modify your CLASSPATH setting so that x:\IBMVJava\eab\runtime20 appears first in the CLASSPATH, where 'x' is the drive letter where you installed VisualAge Java.	Windows NT
Having COMMAREA beans to convert DBCS data from Unicode to IBM code pages	When you create COMMAREA beans to connect to CICS servers running on IBM platforms such as OS/390 or AIX, 43 DBCS characters may get corrupted if you make COMMAREA beans to perform the code conversion from Unicode to such code pages. This is a restriction in JDK, and will be resolved as soon as the JDK converters are fixed.	All Platforms