
Error Messages and System Messages

Successful calls to the JADE Object Manager return a value of zero (**0**). If a value other than **0** is returned, it is your responsibility to determine the cause of the error and take the appropriate action.

Throughout this document, references to the JADE log file refer to the file defined by the [LogFile](#) parameter in the [\[JadeLog\]](#) section of the JADE initialization file. By default, this file is named **jommsg.log**.

Jade Software Corporation Limited cannot accept any financial or other responsibilities that may be the result of your use of this information or software material, including direct, indirect, special or consequential damages, or loss of profits. There are no warranties extended or granted by this document or software material.

You should be very careful to ensure that the use of this software material and/or information complies with the laws, rules, and regulations of the jurisdictions with respect to which it is used. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Jade Software Corporation Limited.

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

Copyright © 2021 Jade Software Corporation Limited.

All rights reserved.

JADE is a trademark of Jade Software Corporation Limited. All trade names referenced are the service mark, trademark, or registered trademark of the respective manufacturer.

For details about other licensing agreements for third-party products, you must read the JADE **Readme.txt** file.

-99 through -1251 Fatal Client Server Network Errors

This section describes client server network-related error messages that can occur when using JADE.

-99 Over ran a buffer, memory has been corrupted

Cause

This error occurs if data copied to an internal buffer copies past the end of the allocated space.

Action

Check your usage of non-JADE code to ensure that the user is not copying more data into a JADE-supplied buffer than the allocated size of the buffer; for example, an external function call may be given a **binaryBuffer[30]** and the external function expected a larger buffer and copied 40 bytes into the buffer, causing the overrun.

-100 A memory allocation request failed

Cause

This error occurs if a memory allocation request fails.

Action

Shut down one or more applications to regain memory, and then retry the request. If this problem persists, it may be necessary to add additional physical memory or reduce the cache size limits. For details, see the **TransientCacheSizeLimit** and **ObjectCacheSizeLimit** parameters in the [\[JadeClient\]](#) and [\[JadeServer\]](#) sections of the *JADE Initialization File Reference*.

-101 Invalid client id number

Cause

This error occurs if the internal client id number is out of range.

Action

This is an internal error.

Report the error to your network administrator. If you or your network administrator suspects this is a JADE problem and if your JADE licenses include support, contact your local JADE support center or JADE Support.

-102 Network connection not open

Cause

This error occurs if you try to access a network that has not yet been opened.

Action

This is an internal error.

Report the error to your network administrator. If you or your network administrator suspects this is a JADE problem and if your JADE licenses include support, contact your local JADE support center or JADE Support.

-103 Out of memory for network command

Cause

This error occurs if the network could not allocate resource memory.

Action

Perform one or more of the following actions.

- Shut down other applications that are not required
- Install more memory
- Increase virtual memory

-104 Failed to establish network connection

Cause

This error occurs if a network connection could not be opened.

Action

Check the following.

- The JADE server node is running
- TCP/IP transport is enabled
- Port numbers on the client and the server are the same

Ensure that the [ServerNodeSpecifications](#) parameter in the [\[JadeClient\]](#) section of the JADE initialization file is a valid host name (or alias) for the server workstation and that the host name to Internet Protocol (IP) address mapping is correctly defined in your network configuration (local host file, Domain Name Service (DNS), or Windows Internet Naming Services (WINS)). Contact your network administrator if you are unsure how to do this.

-105 Read/Write error on network

Cause

This error is raised by a network error or if the target workstation is no longer on the network.

Action

This is an unrecoverable error. Check that the target workstation is still running or that any gateway workstation is still running.

-106 Network not available

Cause

This error occurs if the network has not been installed.

Action

Install the network.

-107 Out of resource

Cause

This error occurs if the network is unable to allocate the required memory resources.

Action

Perform one or more of the following actions.

- Shut down other applications that are not required
- Install more memory
- Increase virtual memory

-109 Network maximum message length exceeded

Cause

This error occurs if the network message that is offered to the network layers is too large.

Action

This is an internal error. Report the error to your network administrator. If you or your network administrator suspects this is a JADE problem and if your JADE licenses include support, contact your local JADE support center or JADE Support.

-110 User signon currently disabled on server

Cause

This error occurs if the user is trying to log on to a server for which log-on is currently disabled.

Action

Contact your JADE system administrator or network administrator. There may be some reason why log-on is currently disabled. Try to start your application again later.

-111 Host not found

Cause

This error occurs if the host to which you are trying to connect cannot be found.

Action

Ensure that the [ServerNodeSpecifications](#) parameter in the [\[JadeClient\]](#) section of the JADE initialization file is a valid host name (or alias) for the server workstation and that the host name to Internet Protocol (IP) address mapping is correctly defined in your network configuration (local host file, Domain Name Service (DNS), or Windows Internet Naming Services (WINS)). Contact your network administrator if you are unsure how to do this.

-112 Wait for receiver failed

Cause

This error occurs when the JADE wait for a system event has failed. This is a fatal error. Synchronization between the client and server has been lost.

Action

Try to start up your application again.

-113 Server Network initialization failed

Cause

This error occurs when the client is initiating a connection to an additional secondary server and the initialization attempt fails.

Action

This is an internal error. Report the error to your network administrator. If you or your network administrator suspects this is a JADE problem and if your JADE licenses include support, contact your local JADE support center or JADE Support.

-118 Network connection lost

Cause

This error occurs if you lose connectivity to the remote server.

Action

Shut down the application client and then restart.

-119 Remote node closed connection

Cause

This error occurs if a remote node has closed the connection or if the JADE Monitor has forced you to sign off.

Action

Try to start up your application again. If you still cannot connect to the server, contact your network administrator.

-120 Unable to load jomsrvr DLL

Cause

This error occurs if the server is unable to dynamically load the **jomsrvr.dll** library.

Action

Check the JADE messages log file (**jommsg.log**) for additional information which may clarify the cause of the error. If the error occurs again and your JADE licenses include support, contact your local JADE support center or JADE Support.

2 through 1299 JADE Object Manager Errors

This section describes general errors and errors that can occur when using the JADE Object Manager.

3 Caught an unexpected C++ exception

Cause

This error occurs if an unexpected or unanticipated operating system error has occurred.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

4 Object not found

Cause

This error occurs when your application logic attempts to access an object and the referenced object does not exist. The most-common situation in which this error occurs is when application logic accesses a feature of an object (for example, a method or property) using a reference to an object that existed when the reference was first obtained (for example, from a collection) and has since been deleted by a concurrent transaction.

A reference to a deleted object is deemed to be invalid.

Action

Determine why an invalid object reference is being used in this situation and then correct the logic to avoid the error, or code and arm an exception handler to handle the error.

5 Caught an unexpected SEH exception

Cause

This error occurs if an unexpected or unanticipated operating system error has occurred.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

6 The designated object branch does not exist [obsolete]

Cause

This error occurs if the requested branch of a versioned object was not found.

Action

Write application code to detect this situation and to return an appropriate message to the user.

7 Caught an unexpected C++ Pure Virtual Call exception

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

8 Invalid address

Cause

This error occurs when the object identifier (OID) of an object has been corrupted.

The object identifier contains the instance identifier, which is used to access the record in the database for that object. If this instance identifier is corrupted, a subsequent attempt to read the record based on this instance identifier could return this error or an incorrect record.

This error can also be returned if your application code attempts to access a deleted object.

Action

Debug the application to determine the cause of the error and then correct your code accordingly.

12 Cannot access object in an offline file or offline file partition

Cause

This error occurs if your application code references an object (using the object identifier of the object) in a database file or file partition that is absent because it is offline.

Action

Your application code should detect this and handle the situation appropriately in an exception handler method.

30 Server timeout

Cause

This error occurs when a request times out within the server due to a resource not being available.

Action

Ensure that all resources are available and then retry the request.

64 Operating system or runtime error occurred, refer to jommsg.log

Cause

This error relates to the operating system or runtime.

Action

Check the JADE messages log file (**jommsg.log**) for more details.

123 Invalid library

Cause

This error occurs if an external method library is not available on the current path.

Action

Copy the library to the current path to the JADE binary (**bin**) directory.

124 Cannot access memory buffer created on a different node

Cause

This error occurs if a **JadeHTTPConnection** object is accessed from a node other than the one where it was created.

Action

Ensure that you only access a **JadeHTTPConnection** object on the node where it is created.

125 The specified or default Jade INI file does not exist

Cause

This error occurs if an attempt is made to start a **JADE** executable and the JADE initialization file is not at the specified or expected location. For more details, see "[Location of the JADE Initialization File](#)", in Chapter 1 of the *JADE Initialization File Reference*.

Action

Specify the fully qualified name of an existing JADE initialization file in the command line for a JADE executable.

126 Supplied buffer size is too small

Cause

This error occurs when copying data to a buffer that is not large enough.

Action

If you are making a JADE Application Programming Interface (API) call that provides a buffer and specifies the size, increase this value and try again.

If you cannot specify a larger buffer size and your JADE licenses include support, contact your local JADE support center or JADE Support.

131 This method can only be executed in single user

Cause

This error occurs if a method that is supported only in single user mode is called in multiuser mode.

Action

Run the application in single user mode.

1000 Invalid parameter type

Cause

This error occurs if a parameter passed to a method or function has the wrong type or format. For example, when using the JADE Application Programming Interface (API), this error occurs if the **format** variable in the signature of the **DskParam** structure has a value that is inconsistent with the primitive type of the expected parameter; for example, the **format** variable has a flag indicating that the data is an integer but the expected parameter is a string.

This error also occurs if the method passed to the **Window** class **setDragAndDropFiles** method:

- Is not on the current **Form** class or on the targeted **Control** class
- Signature is not one of the following formats
 - If the method is defined on the form of the targeted window, the required signature is:

```
method-name(win: Window; array: HugeStringArray);
```

The **win** parameter specifies the window on which the files were dropped and the **array** parameter specifies an array of file or directory names that were dropped.

- If the method is defined on the class of the targeted control targeted, the required signature is:

```
method-name(array: HugeStringArray);
```

The **array** parameter specifies an array of file or directory names that were dropped (**self** is the control instance on which the files were dropped).

- Has a return type defined

Action

Ensure that the correct parameter type or format is used.

With respect to the example earlier in this topic, ensure that when the **DskParam** structure is used, the **format** variable in the signature corresponds to the expected parameter type.

1001 Invalid directory

Cause

This error occurs if the database **path** parameter specified in the JADE Object Manager initialization call is invalid.

Action

Check your database **path** parameter. The problem can be caused by one of the following.

- The name is too long. It must not be greater than 260 characters.
- The client path does not match the server path.

1002 Invalid parameter value

Cause

This error occurs if the value of a parameter or a primitive type receiver is not correct.

Action

Change your application code to use the correct parameter value. Refer to the JADE messages log file (**jommsg.log**) for help in identifying the invalid value and the called method.

1003 Class definition not found in schema branch

Cause

This error occurs if a class definition was not found in the schema agent hierarchy on the current node.

Action

This is an internal error.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1004 Class is closed

Cause

This error occurs if your application program references a class that it has not opened.

Action

Open the class before you reference it. Additionally, check that the classes of all exclusive subobjects have been opened.

1006 Library not found

Cause

This error occurs when an external method or function references a Dynamic Link Library (DLL) that cannot be found in the executable directory or within the current path.

Some platforms support a reduced feature set and are shipped without some of the usual JADE DLLs. This error could occur if you attempt to use a feature that is not available on these platforms.

Action

Ensure that the external method specifies the correct library name. If the library name is correct, ensure that the specified DLL is present in the executable directory or within the current path (defined in the **System** application of your Windows Control Panel). Refer to the JADE messages log file (**jommsg.log**) for information that will help identify the library, entry point, and externalMethod involved.

1007 Cannot manually delete an exclusive subobject

Cause

This error occurs if there is an attempt to delete an exclusive subobject. Your application program cannot delete exclusive subobjects, as JADE automatically deletes them.

Action

Remove the portion of code that resulted in the error from your application program, or make the subobject non-exclusive.

1008 Entry point not in library

Cause

This error occurs when a reference is made to an entry point that is not exported from the Dynamic Link Library (DLL). This occurs when a class is being opened and one of its methods or database methods is not defined in the DLL associated with the method.

Some platforms support a reduced feature set and are shipped without some of the usual JADE DLLs. This error could occur if you attempt to use a feature that is not available on these platforms.

Action

Ensure that the external method or function specifies the correct entry point name. If the entry point name is correct, check that the DLL contains an entry point of the specified name and that it is exported.

Refer to the JADE messages log file (**jommsg.log**) for information that will help identify the library, entry point, and externalMethod involved.

1009 Cannot set subobject reference

Cause

This error occurs when there is an attempt to set a property of an exclusive subobject. Exclusive subobjects do not respond to **set** methods.

Action

Make the subobject non-exclusive, or remove the portion of code that resulted in the error from your application program.

1010 The requested method is not defined for the Type of the receiver

Cause

This error occurs if the message specified in the [jomSendMsg](#) call is not defined in the schema for this object class, or the message does not exist in the DLL specified for this message.

Action

Check that the specified message exists and that it has been defined correctly in the schema. (For details, see [Chapter 3](#) of the *JADE Object Manager Guide*.)

1011 The requested property is not defined for this object class

Cause

This error occurs if the property name specified in the [jomGetProperty](#) call or the [jomSetProperty](#) call is not defined in the schema for this object class.

Action

Check that the specified property has been correctly defined in the schema. (For details, see [Chapter 3](#) of the *JADE Object Manager Guide*.)

1012 Cannot modify system object

Cause

This error occurs if your application program attempts to modify a **RootSchema** schema object.

Action

This is an internal error.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1013 Shared reference to subobject

Cause

This error occurs if more than one object attempts to reference a subobject that has been defined as exclusive.

Action

Remove the shared reference, or make the subobject shared.

1014 Agent error

Cause

This is an internal JADE error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1015 Class agent mismatch

Cause

This is an internal JADE error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1016 Cannot manually create an exclusive subobject

Cause

This error occurs if your application program attempts to create an exclusive subobject. An application program cannot create exclusive subobjects because the JADE Object Manager automatically creates them.

Action

Remove the portion of code that resulted in the error from your application program, or make the subobject non-exclusive.

1017 Invalid server mode

Cause

This error occurs if the server name parameter specified in the JADE Object Manager initialization call is invalid.

Action

Check your server name parameter.

1018 No memory for buffers

Cause

This error often occurs if a cache setting is too small (for example, when you attempt to load a large image file).

When a process needs to place an object in the transient cache and the cache is full, objects must be swapped out of the cache to make room.

If the process currently has the most amount of cache allocated to it, it swaps out its own objects. If not, one of the other processes is requested to swap out its objects. Because there may be a delay before this process swaps out its objects, the cache is temporarily expanded to allow the new object to be placed into it.

The process that has been requested to swap out objects does so when it next attempts to allocate cache space or it has been detected as idle (that is, there have been no cache allocations for between 10 and 20 seconds). When temporarily expanded, the maximum by which the cache is allowed to exceed the specified size is 50 percent. If the excess is greater than 50 percent, this error occurs.

It can also occur if a cache setting exceeds physical memory, or if there is not enough physical memory for the system.

Action

Increase the size of your cache to allocate more memory for caching database objects. For details, see the **TransientCacheSizeLimit** and **ObjectCacheSizeLimit** parameters in the [\[JadeServer\]](#) and [\[JadeClient\]](#) sections of your JADE initialization file. For details, see the *JADE Initialization File Reference*.

Alternatively, correct the cache setting if it is too large, or close some other applications to free up memory.

1019 Cannot load schema

Cause

This error occurs when the schema information cannot be loaded into memory.

When the JADE **initialize** routine is called, the schema information is loaded into memory. This error is sent to your application when this schema cannot be loaded.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1020 Too many classes

Cause

This error occurs if you attempt to exceed the limit on the number of classes.

JADE currently allows user class numbers in the range 2,048 through 999,999, inclusive. Any attempt to use a user class number outside this range returns this error

Action

Rationalize your class hierarchy to reduce the number of classes.

1021 Invalid class for property

Cause

This error occurs if you attempt to assign an object reference to a property and the class of the reference is not compatible with the defined type of the property.

Action

Redefine the property definition to the correct class, or fix the portion of code that resulted in the error in your application program. For details, see "[Assignment Compatibility](#)" under "JADE Language Reference", in Chapter 1 of the *JADE Developer's Reference*.

1022 Type is incompatible with collection membership

Cause

This error occurs if your application code attempts to use a **foreach** instruction **as** clause and the evaluated **foreach** collection cannot contain values that are of the **as** clause type; for example, the evaluated collection may have a membership of type **String** but the **as** clause has a type of **Object**.

Action

Change the application code that resulted in this error. Check that all collections being provided to the **foreach** instruction have a compatible membership type.

1023 Type is incompatible with foreach target variable

Cause

This error occurs if your application code attempts to execute a **foreach** instruction with an **as** clause expression that has an evaluated type that is incompatible with the type of the target variable; for example, the **as** clause may have evaluated to the type **String** but the target variable of the **foreach** instruction may be type **Object**.

Action

Change the application code that resulted in this error. Check that all possible values of the **as** clause expression are compatible with the type of the target variable.

1024 No classmap defined for class

Cause

This error occurs if your application code attempts to use a class for which no class map has been defined. All open classes must have a class map.

Action

Define the class map for this class, by using the Class Maps Browser.

1025 Already in transaction state

Cause

This error occurs if your application code attempts to enter transaction state for a database that is already in transaction state.

Action

Change your application code so that it does not attempt to enter transaction state for a database that is already in transaction state. For more details, see the [Process](#) class [isInTransactionState](#) method.

1026 Not in transaction state

Cause

This error occurs if your application code attempts to end transaction state for a database that is not in transaction state.

Action

Correct your application code so that it does not attempt to end a transaction state for a database that is not in transaction state. For more details, see the [Process](#) class [isInTransactionState](#) method.

1027 Object locked

Cause

This error occurs if you attempt to place a lock on an object that is already directly locked. The exception is raised when the value of the [JadeServer] [ServerTimeout](#) parameter in the [JadeServer](#) section of the JADE Initialization file is reached with the attempted object still being directly locked.

Action

Lock exceptions are *continuable*; that is, the [continuable](#) property is set to **true**. A lock exception handler could therefore attempt to acquire the lock using the [tryLock](#) method and if successful return **Ex_Continue**.

Your application code should detect this and handle the situation appropriately; for example, your application might retry the lock after waiting a short time and if it is still not successful, provide an option to wait or to cancel the action requiring the lock.

1031 Cannot update a bootstrap schema class

Cause

This error occurs if a process attempts to modify a bootstrap schema class. This is not allowed.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1032 Cannot clear buffers in transaction state

Cause

This error occurs if an attempt is made to clear buffers specifying all instances of a class while the object is being modified in the current transaction.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1034 Duplicated object

Cause

This error occurs if an attempt is made to add an entry to a collection with a key that already exists in that dictionary.

Action

Change your application code to handle the addition and changing of entries appropriately.

1035 String too long

Cause

This error occurs if your application code attempts to set a string property with a value string longer than the declared length of the string variable.

Action

Perform one of the following actions, as appropriate.

- Change your application code to store only strings of the correct length.
- Update your schema to increase the length of the string that can be stored in this property.

1036 Failed to find superschema agent

Cause

This error occurs if JADE fails to find a required schema agent when opening a schema.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1037 Failed to find class descriptor in superschema

Cause

This error occurs if JADE fails to find a required class descriptor in a superschema agent when opening a class.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1038 Failed to find base class descriptor for subschema copy

Cause

This error occurs if JADE fails to find a descriptor for the original class in a superschema when opening a class descriptor for a subschema copy class.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1039 Failed to open JADE schema agent

Cause

This error occurs if an attempt to open the JADE schema fails.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support to report this error.

1040 Failed to open schema agent

Cause

This error occurs if an attempt to open a schema fails; for example, if an application attempts to sign on but the schema is not available because of a reorganization.

Action

If the error occurred because of a reorganization, wait for the reorganization to complete and try again; otherwise, if your JADE licenses include support, contact your local JADE support center or JADE Support.

1041 Failed to open root schema agent

Cause

This error occurs if JADE fails to open the **RootSchema**.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1042 Invalid handle

Cause

This error occurs if the contents of the handle pointer in the current call are invalid. The contents of a handle are loaded at context initialization time and are made invalid at context finalization time.

Action

Ensure that a context has been initialized properly and that it is still current.

1043 Cannot create instances of an abstract class

Cause

This error occurs if your application code attempts to create an instance of a class that is defined as abstract. This is not allowed.

Action

Perform one of the following actions, as appropriate.

- Change your application code so that instances of abstract classes are not created.
- Change the definition of the class.

1044 Application received a forced sign off request

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1045 Assoc buffers leader not locked

Cause

This error occurs if a collection block is referenced and the collection header object is not in the cache.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1046 Invalid class number

Cause

This error occurs if your application code makes a call for an object that has a class number that does not exist in the dictionary of schema classes.

Action

Change your application code to ensure that only known classes are used.

1047 Invalid lock type

Cause

This error occurs if the **lockType** parameter in a lock request issued by a **lock** method is not **Exclusive_Lock (3)**, **Update_Lock (4)**, **Reserve_Lock (2)**, or **Share_Lock (1)**.

Action

Change your application code to pass a valid **lockType** parameter.

1048 Update outside transaction

Cause

This error occurs if your application code requests an update operation for a persistent object outside transaction state. Persistent objects accept updates only in transaction state.

Action

Change your application code to ensure that updates are made within transaction state. For more details, see the [isInTransactionState](#) method of the [Process](#) class.

1049 Database is unavailable

Cause

This error occurs if you attempt to open a database that is already open in single user (exclusive) mode.

Action

Determine in which mode the database should be opened; that is, single user or multiuser mode.

1051 Invalid schema agent

Cause

This error occurs if a service is required that is not supported by the available schema agent.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1052 Too many users

Cause

This error occurs during process sign-on if there is insufficient room in an internal table to record an entry for this process.

Action

This error is an indication that the correct license has not been applied. In an SDS environment, it is important that the license is applied to both the primary and secondary databases.

If the correct license has been installed and this error occurs, contact JADE Support.

1055 Internal transaction trace information is inconsistent

Cause

This error occurs if an internal fault occurs while transaction tracing information is being gathered or processed.

Action

Submit a PAR for the fault.

1056 Incompatible schema file

Cause

This error occurs if you attempt to open a database in which the internal JADE Object Manager format has changed between releases.

Action

Ensure that the software is accessing the correct database. If the change is valid, unload and recreate the schema file.

1057 Incompatible jomscm library

Cause

This error occurs if your **jomscm.dll** file is incompatible with your schema file.

Action

Ensure that the software is accessing the correct database or that the correct DLL is being used.

1058 Cannot set a schema in transaction state

Cause

This error occurs if a process attempts to change the currently set schema while in transaction state.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1059 Internal use only

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1060 Class still has subclasses

Cause

This error occurs if you attempt to delete a class that still contains subclasses. This would occur only when using a schema maintenance tool.

Action

Delete all subclasses before deleting the class.

1061 Class is referenced by other classes

Cause

This error occurs if you attempt to delete a class that is still referenced by other classes. This would occur only when using a schema maintenance tool.

Action

Remove all references to this class before deleting it.

1062 Invalid context handle passed to kernel

Cause

This is an internal error. An invalid or out-of-date context handle was passed to a JOM API.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1063 File still has class maps assigned

Cause

This error occurs if you attempt to delete a file while it still has an assigned class map. This would occur only when using a schema maintenance tool.

Action

Remove the class map before deleting the file.

1064 Database still has a file assigned

Cause

This error occurs if you attempt to delete a database that still has an assigned file. This would occur only using a schema maintenance tool.

Action

Remove file assignments before deleting the database.

1065 Notification for this OID already set

Cause

This error is for internal use only and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1066 Notification not set for this OID

Cause

This error occurs if you attempt to end notification on an object identifier (OID) that has no matching begin notification or if the target is null.

Action

Change your application code to prevent this.

1067 Notification to system events can only be set on persistent objects

Cause

This error occurs if the **target** parameter of the **Object** class **beginNotification** method is transient and the **eventType** parameter is a system event (that is, a value less than **16**, which includes the **RootSchema** global constants **Any_System_Event** (0), **Object_Create_Event** (4), **Object_Delete_Event** (6), and **Object_Update_Event** (3)).

Action

Change your application code so that notification to system events is not set on transient objects.

1068 Feature not available in this release

Cause

This error occurs if a JADE request is made for a feature that is not implemented in this release.

Action

Change your application code to prevent this.

1069 The object is not the latest version

Cause

This error occurs if an internal mismatch exists within the JADE Browser.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1071 Cannot inherit method

Cause

This error occurs if your application code makes an inherit method call for an object and no corresponding method exists in the superclasses of that object, or the call is made from a destructor method.

Action

Ensure that the called method is valid.

1072 Object has been deleted in the current transaction

Cause

This error occurs if an object reference to a deleted object attempts to access that object.

Action

Change your application code to ensure that it does not attempt an object reference to a deleted object.

1073 Invalid put object

Cause

This error occurs if an attempt to modify a database results in a failure between the server and database modules.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1074 User must logoff

Cause

This error occurs when you have been remotely signed off with the server [forceOffUser](#) request.

Action

You must issue a log-off request, as this is the only valid request when you have been forced off.

1075 Cannot force off user

Cause

This error is returned if you attempt to force off a user that the JADE Object Manager does not allow to be forced off.

Action

Check that the user being forced off is not an internal process.

1078 Client request is invalid

Cause

This error occurs if the server considers a request made from the client to the server invalid.

Action

Close down the server and then retry the request. If this fails and if your JADE licenses include support, contact your local JADE support center or JADE Support.

1079 Access mode violation

Cause

This error occurs if you attempt to modify the database schema while the **ReadOnlySchema** parameter in the [\[JadeServer\]](#) or [\[JadeClient\]](#) section of your JADE initialization file is enabled.

Action

Set the **ReadOnlySchema** parameter in the [\[JadeServer\]](#) or [\[JadeClient\]](#) section of your JADE initialization file to **false** before updating the database schema. For details, see the *JADE Initialization File Reference*.

1080 Double call to JADE Object Manager

Cause

This error occurs if there is an attempt to perform a JADE request while a JADE request is still outstanding. This occurs only if the network communication is asynchronous.

Action

Ensure that your network is running in synchronous mode, or change your application code to ensure that double calls do not occur.

1081 Deadlock

Cause

This error occurs if an attempt to lock a JADE object results in a deadlock situation.

Action

Change your application code to ensure that the application can respond to this result by retrying the request or by notifying the user of a course of action; for example, the option to wait or to cancel the request.

1082 No memory available for notification

Cause

This error occurs if an attempt to register a begin notification request or a notification event occurs when there is no available memory to record this.

Action

Add additional memory to the server, or reduce the number of notification requests and events. Ensure that delayed user events are followed by an end transaction.

1084 Invalid database usage

Cause

This error occurs if the usage parameter passed in the open database API call is not valid.

Action

Change your application code to pass a legal usage value.

1085 Read only database

Cause

This error occurs if you attempt to update an object in a read-only schema file. It is also raised if a client connection to JADE through the ODBC interface attempts a transaction.

Action

Ensure that you do not run the development environment when the read-only schema parameter is set in the JADE initialization file. For details, see the **ReadOnlySchema** parameter in the [\[JadeServer\]](#) or [\[JadeClient\]](#) section of the *JADE Initialization File Reference*.

1086 Class of object is invalid in this context

Cause

This error occurs when an object reference in a guarded expression is not of the type asserted by the type cast. For example, in the JADE language expression **object.Automobile**, if **object** is not a member of the class of **Automobile** (that is, it is not of type **Automobile**), this error would occur when the expression is evaluated.

Action

Determine why an object reference of the wrong type is being used in this context, and correct the code that led to the error. The method and source position containing the expression at which the exception was raised are reported in the error description.

For more details about type cast expression, see Chapter 1 of the [JADE Developer's Reference](#).

1088 Cannot manually update an automatic reference

Cause

This error occurs if an attempt is made to manually update a reference that is defined as being automatic. Automatic references are maintained by JADE and cannot be updated manually.

Action

If it is not necessary to manually update the reference, change your application code so that the reference is not updated. If it is necessary to manually update the reference, change the update mode in the Define Reference dialog to **manual/automatic**.

1089 Context error

Cause

This error indicates an internal JADE Object Manager kernel problem.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1090 Attempted access via null object reference

Cause

This error occurs if your application code attempts to use a **null** object reference in a JADE language or JADE Object Manager operation. It may be that the referenced class has been defined but no instances of that class exist.

The most-common situation in which this occurs is when your code accesses a feature of an object (for example, a method or a property) by means of an object reference that has a **null** value. The reference can be thought of as a pointer or pathway for the operation. JADE cannot proceed past a **null** reference, even though JADE does allow properties and references that contain **null** values.

Action

Determine why a **null** object reference is being used in this situation and then correct the code that led to the error. It may be that the object must be created, or you may have to change your code to ensure that the object reference is set to a value other than **null**. The method and source position containing the expression at which the exception was raised are reported in the error description.

1091 Binary too long

Cause

This error occurs if your application code attempts to store a binary item that is longer than the specified length.

Action

Perform one of the following actions, as appropriate.

- Change your application code to store only binary items of the correct length.
- Update the schema to increase the size of the binary item that can be stored in this property, by using the **Define Attribute** dialog.

1092 Object not available

Cause

This error occurs during an exception handling operation when your application code attempts to access the object that caused the error.

This is usually caused when a lock exception handler attempts to access a target object other than to retry the lock.

Action

Change your application code so that objects that caused a lock exception are not referenced in the exception handler, except to retry the lock.

1093 Invalid lock duration

Cause

This error occurs if the **lockDuration** parameter in a lock request issued by a **lock** method is not **Transaction_Duration (0)** or **Session_Duration (1)**; that is, locks cannot be of persistent duration.

Action

Change your application code to pass a valid **lockDuration** parameter.

1097 Exclusive collections cannot have subobject properties

Cause

This error occurs if an attempt is made to access a property on an exclusive collection where the property is a subobject. This operation is not allowed.

This error is most-commonly caused if you add a subobject property (for example, slob, blob, or another collection) to one of your collection classes and then define an exclusive reference to this collection on another class. By making the reference exclusive, it becomes a subobject reference. Any attempt to use the subobjects on the collection from this reference will raise this error.

Action

Remove the subobject properties from the collection class, or make the reference to the collection a shared reference rather than an exclusive reference.

1099 Key path component does not have an inverse to its prior component: refer to jommsg.log

Cause

This error occurs if you have defined a key path that has a component that does not have an inverse to its prior component.

When a collection with key paths participates in a relationship (that is, a reference of the collection type is used as an inverse), the following applies.

- Except for the first component in a key path, each class on which every subsequent component is defined must have at least one property that is an inverse to the previous component in the key path.

If this is not the case, an error occurs when the offending class is opened.

Consider the following example.

class	Person
reference	myEvent of type Event
class	Event
attribute	timestamp of type Timestamp
member key dictionary	PersonByEventDict
membership	Person
key path	myEvent.timestamp

In this example, accessing an instance of **Person** or **Event** (thereby opening its class) would result in a *1099 - Key path component does not have an inverse to its prior component* exception because there is no property on **Event** that is an inverse of the **myEvent** property on **Person**. The **timestamp** component of the key path is therefore missing an inverse to its prior component, **myEvent**. In this case, the solution is to ensure that **myEvent** on **Person** has at least one inverse on the **Event** class.

Action

For each component of the key path (except for the first), ensure that the class on which it is defined has at least one inverse to the prior component in the key path. If this is not possible, you cannot use a key path in this case.

1100 The target object is already versioned

Cause

This internal error occurs only when objects are being versioned due to a change to the definition of a class.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing a memory dump taken at the time, if possible.

1101 Interface still has subinterfaces

Cause

This error occurs if you attempt to delete an interface that still has sub-interfaces. It would occur only when using a schema maintenance tool.

Action

Change the interface so that it does not extend any sub-interfaces before you attempt to delete it.

1102 Interface still has implementors

Cause

This error occurs if you attempt to delete an interface that still has implementors. It would occur only when using a schema maintenance tool.

Action

Remove all implementors of the interface before you attempt to delete it.

1103 Already signed on

Cause

This error occurs if a thread uses `jomSignOn` more than once without an intervening `jomSignOff` call.

Action

If `jomSignOn` is being used in locally written software, examine the code to determine why the `jomSignOff` call is being repeated without an intervening `jomSignOff` call; otherwise, report this error to your JADE system administrator.

1104 Could not terminate all user processes

Cause

This error occurs if all user applications are not terminated within a specified time. When a reorganization is being replayed on an SDS secondary server, all user applications running in the SDS secondary environment must be terminated before the reorganization can proceed.

Action

Identify the applications that have not terminated, and find out why they are still running.

1105 Could not close all user classes

Cause

This error definition is reserved for future use. It should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1106 Can not update a frozen object

Cause

This error occurs if you attempt to update an object whose volatility state is **Volatility_Frozen**.

Action

The volatility state of a frozen object must be changed before the object can be updated.

1107 The current operation cannot be performed on a transient instance

Cause

This error occurs if you attempt an operation on a transient instance that is allowed only on a persistent instance.

Action

Modify your code to check whether an instance is transient, and if it is, do not perform the operation.

1108 Too many interfaces

Cause

This error occurs if you attempt to exceed the limit on the number of interfaces.

The maximum number of interfaces, including system interfaces, is 16,383. The maximum number of user interfaces is 12,953.

Action

Rationalize your class hierarchy to reduce the number of interfaces.

1109 Too many schemas

Cause

This error occurs if you attempt to exceed the limit on the number of schemas.

The maximum number of schemas, including system schemas, is 65,535. The maximum number of user schemas is 65,435.

Action

Rationalize your class hierarchy to reduce the number of schemas.

1110 Too many translatable strings

Cause

This error occurs if you attempt to exceed the limit on the number of translatable strings.

The maximum number of translatable strings for each locale is 65,535.

Action

Rationalize your class hierarchy to reduce the number of translatable strings.

1111 Too many locale formats

Cause

This error occurs if you attempt to exceed the limit on the number of locale formats.

The maximum number of locale formats for each locale is 65,535.

Action

Rationalize your class hierarchy to reduce the number of locale formats.

1112 Feature has been deprecated

Cause

This error occurs if you attempt to use a feature that is no longer implemented in this release.

Action

Modify the code to avoid the use of this feature. For details about deprecated features, see the JADE Release Information.

1113 Receiver does not implement the specified interface

Cause

This error occurs if you attempt to register for a notification or arm a timer by using interface methods on an object that does not implement the specified interface.

Action

Modify the code to correct the problem, or check that the class of the object implements the interface before registering for the notification or arming the interface timer.

1114 The specified interface does not define this operation

Cause

This error occurs if you attempt to register for a notification or arm a timer using an interface that does not include the **sysNotification**, **userNotification**, or **timerEvent** method.

Action

Modify the code to correct the problem or include the correct method in the interface.

1115 The corresponding sysNotification or userNotification method has an incompatible signature

Cause

This error occurs if you attempt register for a notification using an interface that includes a **sysNotification** or **userNotification** method with an incorrect method signature.

Action

Modify the signature of the **sysNotification** or **userNotification** method in the interface for which the notification is being registered to match the correct signature.

1116 The corresponding timerEvent method has an incompatible signature

Cause

This error occurs if you attempt to arm an interface timer by using an interface that includes a timer method with an incorrect method signature.

Action

Modify the timer method in the interface for which the timer is being armed to match the **timerEvent(eventTag: Integer)** signature.

1119 Node is terminating

Cause

This error occurs when an operation cannot be completed because the node is in the process of terminating.

Action

Wait until the node has been restarted and then attempt the operation again.

1120 The maximum number of processes are already signed on

Cause

This error occurs on a standard fat client or an application server when the number of users signed on exceeds the value specified in the **MaxLocalProcesses** parameter in the JADE initialization file.

It occurs on a database server node when the number of server applications exceeds the value specified in the **MaxLocalProcesses** parameter in the JADE initialization file.

Action

Increase the value specified in the **MaxLocalProcesses** parameter in the JADE initialization file to allow more processes to sign on when the node is next initialized. Alternatively, set the value to zero (**0**), to remove the restriction on the number of processes that are signed on.

1121 Cannot delete versioned object due to lock conflicts

Cause

This error occurs when an attempt to un-version a schema fails, or while a reorganization is in progress, because another process holds locks on objects that need to be deleted.

Action

Wait until the processes have released their locks and then attempt the reorganization or the un-versioning of the schema again.

1122 Too many exported packages

Cause

This error occurs if you attempt to create more than approximately 27,000 packages in a single schema.

Action

Reduce the number of declared packages or distribute them between the schema and its superschema or subschema.

1123 ExportedPackage has not been numbered

Cause

This error occurs if you attempt to import a package that has not been assigned a unique number.

The assignment should have occurred during the upgrade to JADE version 7.1 or when the package was created in a JADE version 7.1 environment.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1124 StringUtf8 too long

Cause

This error occurs if your application code attempts to set a **stringUtf8** property with a **stringUtf8** value longer than the declared length of the **stringUtf8** variable.

Action

Perform one of the following actions, as appropriate.

- Change your application code to store only **stringUtf8** values of the correct length.
- Update your schema to increase the length of the **stringUtf8** value that can be stored in this property.

1125 Cannot make an object a version of itself

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1126 Cannot conditionally change volatility from frozen if class is currently open

Cause

This error occurs when you use the [changeObjectVolatility](#) method of the [Object](#) class to change the volatility of a frozen object. If the **condition** parameter is set to **true**, situations can be detected where it is not safe to allow the change.

Action

Ensure that no other applications are holding the class open when attempting to change the volatility of a frozen instance of that class.

1128 The target process is not valid

Cause

This error occurs when you attempt to extract local or remote request statistics by using the [extractRequestStatistics](#) method and you specify an invalid target process as the **proc** parameter.

Action

Ensure that the **proc** parameter to the [extractRequestStatistics](#) method represents a valid current process.

1130 Cannot invoke a method marked notImplemented

Cause

This error occurs if an attempt is made to invoke a method that has the [notImplemented](#) method option.

Action

Provide an implementation for the class of the method receiver or ensure that the method is not called for instances of the class.

1131 Another process is currently in control of lock contention statistics

Cause

This error indicates that a process has attempted to use the [beginLockContentionStats](#) method, the [clearLockContentionStats](#) method, or the [endLockContentionStats](#) method defined in the [System](#) class when it is not the process that started lock contention recording.

A lock contention occurs when a lock request on an individual persistent object is rejected or queued because the object is already locked.

Lock contention information is recorded on the database server node. The information recorded includes the number of contentions, the total time processes have spent waiting for a lock on each object, and the maximum time spent waiting for each lock.

Action

Retry the operation later when the process that was in control of lock contention recording has caused it to end. The [queryLockContentionStats](#) method defined in the [System](#) class can be used to find the process that currently has control of lock contention recording.

1132 WBEM interface error when connecting or disconnecting

Cause

This error indicates that an error occurred when trying to establish or end a Web-Based Enterprise Management (WBEM) interface connection. The WBEM interface is used by the WBEM-related [wbemListClasses](#), [wbemQueryQualifiers](#), [wbemListInstanceNames](#), and [wbemRetrieveData](#) methods defined on the [Node](#) class.

When this error occurs, more-specific information is written to the JADE messages log file ([jommsg.log](#)).

Action

Refer to the JADE messages log file ([jommsg.log](#)) to determine the cause of the error. You may need to adjust the security settings or other operating system configuration settings, to resolve this issue.

1133 WBEM interface error when querying data

Cause

This error usually indicates that a Web-Based Enterprise Management (WBEM) class name passed to the WBEM interface is not recognized. The WBEM interface is used by the WBEM-related [wbemListClasses](#), [wbemQueryQualifiers](#), [wbemListInstanceNames](#), and [wbemRetrieveData](#) methods defined on the [Node](#) class.

This error can arise from an internal error when trying to use the WBEM interface.

Action

Ensure that the WBEM class name used with the WBEM-related method is valid. (The [wbemListClasses](#) method can be used to list valid class names.) The JADE messages log file ([jommsg.log](#)) includes more-specific information about this error.

If the problem does not appear to be related to an invalid WBEM class name and your JADE licenses include support, contact your local JADE support center or JADE Support.

1134 Collection block accessed not via a collection header method

Cause

This error occurs if a collection block is being access directly by JADE code; for example, by the **sendMsg** method or a get property operation.

Collection blocks should not be accessed directly by applications, as they are maintained by the JADE collection code.

Action

Change the JADE code so that it does not directly access collection blocks.

1135 Versioning is not supported on this platform

Cause

This error is for internal use only and should not normally occur.

Action

This is normally an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1136 WBEM class name is not valid

Cause

This error occurs if the name supplied as a parameter value to the [wbemListInstanceNames](#), [wbemQueryQualifiers](#), or [wbemRetrieveData](#) method is not a recognized or valid Web-Based Enterprise Management (WBEM) class name.

Windows allows a number of WBEM class names that contain instances to measure various performance-related figures (for example, disk activity and CPU activity). JADE allows only a subset of these class names to be used.

Action

Change your code so that the unsupported class name is not used.

1137 An internal data packet inconsistency was detected

Cause

This error occurs as the result of incorrect encoded information being passed to various internal methods.

If it occurs as a result of using the [extractRequestStatistics](#) method, it could be that the **any** parameter is specified wrongly. The **any** parameter should be the **userInfo** parameter of the notification received in response to a [sendRequestStatistics](#) method request.

Action

Ensure that the **any** parameter to the **extractRequestStatistics** method represents correctly encoded process request statistics values.

1138 Object tracking is already active

Cause

This error occurs if you attempt to start an object tracking session using the **beginObjectTracking** method of the **System** class when object tracking is already active. One object tracking session only is allowed at a time.

Action

Do not call the **beginObjectTracking** when object tracking is already active.

1139 Object tracking is not active

Cause

This error occurs if the **endObjectTracking** method of the **System** class is called, which ends an object tracking session, when object tracking is not active.

Action

Do not call the **endObjectTracking** method when object tracking is not active.

1141 WBEM data exceeded maximum allowable size

Cause

This error occurs when the **wbemRetrieveData** method defined on the **Node** class could not return requested Web-Based Enterprise Management (WBEM) class instances because the total size exceeded the maximum allowed.

Action

Use the **instNameArray** parameter of the **wbemRetrieveData** method to limit the set of WBEM instances returned for each call.

1142 Lifetime of Class instance is invalid

Cause

This error occurs if an attempt is made to create a shared transient instance of a class with a class number for which shared transient instances cannot be represented. This is an internal error.

Action

This is an internal error.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1144 Class access frequency counters are not enabled

Cause

This error occurs if the `getClassAccessFrequencies` method of the `System` class is called and the counting of class accesses has not been enabled by calling the `enableClassAccessFrequencies` method of the `Process` class.

Action

Ensure that the counting of accesses to classes is enabled.

1145 Cannot delete the parent object of the receiver

Cause

This error occurs if your application code invokes an operation on the receiver that attempts to delete the parent object of the receiver; for example, if you attempt to purge an exclusive collection and the parent object is also a member of the collection.

Action

Change your application code to ensure that you do not attempt to delete the parent object of the receiver.

1146 The object was updated before the lock upgrade completed

Cause

This error occurs if an object is updated before the upgrade lock is completed. When the lock on an object is upgraded from **shared** to **update** (for example, when a collection is iterated then updated while in transaction state), the object is unlocked before the **update** lock is requested, which allows another process to update the object.

This exception is continuable.

Action

If it does not matter whether the object is updated before the **update** lock is acquired, instruct the exception handler to return **Ex_Continue**, which allows the commit to proceed.

1147 Invalid intra process request

Cause

This error indicates a problem with the internal intra-process message exchange.

Action

This is an internal error.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1148 Application signon from this node is restricted

Cause

This error arises when an attempt by a client node to run an application is rejected by the database server based on application restrictions in the JADE initialization file.

The restrictions can apply to clients on specific networks or to all clients.

Action

Make changes to the [[JadeClientAppRestrictions](#)] section of the JADE initialization file used by the database server, and restart the client node.

1150 Invalid Object Volatility Value

Cause

This error occurs if an invalid volatility parameter value is passed to a method that changes object or file volatility.

Action

Change your application logic to ensure that a valid value is passed.

1151 More than one class mapped to this DbFile

Cause

This error occurs if you attempt to partition a database file that has more than one class mapped to it.

Action

A requirement for a database file to be partitioned is that at most one class is mapped to it. You can move instances of the other classes to different map files using the **MoveInstances** command in the [commandFile](#) parameter of the batch Schema Load utility (**jadloadb**). For details, see "[Moving Class Instances](#)" in Chapter 14 of the *JADE Developer's Reference*.

1152 Collection class cannot be partitioned

Cause

This error occurs if you attempt to partition a database file that has a collection class mapped to it.

Action

Do not attempt this invalid operation.

1153 Operation aborted due to a lock timeout

Cause

This error occurs when an operation fails because another process holds locks on meta-schema objects that the operation needs to access. This can occur if a global exception handler passes back a lock exception to the kernel, forcing the current operation to be aborted; for example, opening a class.

Action

Wait until the processes have released their locks and try again, or check your application code to ensure that no environmental objects or dictionaries are locked.

1154 Method tracking is not allowed

Cause

This error occurs when the current setting of the [MethodTrackingEnabled](#) parameter in the [\[JadeSecurity\]](#) section of the JADE initialization file is **false** and the [startMethodTracking](#) method of the [Process](#) class is called to track a method.

It can also occur when method tracking is enabled if you attempt to call a method for which method tracking is not allowed.

The following methods cannot be tracked.

- [getAndValidateUser](#) in the [Global](#) class or a reimplementation in a subclass
- [isUserValid](#) in the [Global](#) class or a reimplementation in a subclass

Method tracking is not currently supported for [serverExecution](#) methods.

Action

Ensure that method tracking is enabled and that you are not calling methods that cannot be tracked.

1155 Transaction trace is already currently started for this process

Cause

This error occurs when the [startTransactionTrace](#) method of the [Process](#) class is called and transaction tracing is already active for the current process.

Action

Correct your logic.

1156 Transaction trace is not currently started for this process

Cause

This error occurs when the [stopTransactionTrace](#) method of the [Process](#) class is called and transaction tracing is not active for the current process.

Action

Correct your logic.

1157 The object is currently being used for transaction tracing

Cause

This error occurs when an application attempts to delete an instance of the [JadeTransactionTrace](#) object class that is currently being used to record transaction tracing information. This instance cannot be deleted while transaction tracing is enabled.

Action

Correct the application code.

1158 The method stack is non-empty

Cause

This error occurs if the JADE call stack is not empty when the **jomSignOff** call is made. The call stack must be empty before a process can sign off.

Action

Change your application code so that the call to **jomSignOff** is not done until the call stack is empty. Consider using the **jomTerminate** API call instead.

1159 Update not allowed

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1160 Background Process request was not completed

Cause

This error occurs when the JADE background process for a node is unable to complete a request; for example, to sign a process on or off.

Action

Retry the operation that caused the exception to be raised. Refer to the JADE messages log file (**jommsg.log**) for more details about the error.

1161 Background Process request encountered a lock error

Cause

This error occurs when the JADE background process for a node is unable to obtain a lock it requires whilst carrying out a request; for example, to sign a process on or off.

Action

Retry the operation that caused the exception to be raised. Refer to the JADE messages log file (**jommsg.log**) for more details about the error.

1162 The system is not Delta Database capable

Cause

This error occurs if the database server node is not specified to be delta database-capable.

Action

Set the **DeltaDatabaseCapable** parameter in the [**JadeServer**] section of the JADE initialization file on the database server node to **true** and restart the database server.

1163 Could not change delta database mode because not all processes are idle

Cause

This error occurs and the deactivation attempt is abandoned if it cannot be completed because not all applications have become idle within the specified time.

For an application to be idle it must not be currently executing any methods invoked from the JADE object manager; that is, its object manager call stack should be empty. The deactivation attempt is immediately abandoned if there are any idle processes that have persistent objects locked or that are in transaction state.

Action

Ensure that the conditions for deactivating the delta database are satisfied and retry the deactivation.

1164 Parent-subobject edition mismatch

Cause

This error occurs when a blob (or slob or slobUtf8) is updated and the edition is out of step with its owner.

Action

Certify the data using the JADE Logical Certifier utility and use the *fix-up* script that is generated, to repair the data.

1165 A delta database transition request is already in progress

Cause

This error occurs if an activation or deactivation request is currently in progress.

Action

Wait for the current delta database activation or deactivation request to complete.

1166 Node already initialised

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1167 Exceeded the maximum load state/lock state nesting level

Cause

This error occurs if a process attempts to enter load state or lock state after previously entering those states 255 times in nested fashion.

Action

Change your application code to prevent too many nested [beginLock](#) or [beginLoad](#) instructions.

1168 Exceeded the maximum number of nodes allowed

Cause

This error occurs when too many client nodes attempt to connect to the database server.

The current limit is 65,536 client nodes.

Action

Shut down some client nodes before attempting to start new nodes.

1169 Node is not initialised

Cause

This error occurs when the specified node handle used in a JOMAPI call is for a client node that has not been initialized.

Action

Modify the code to ensure that the node handle used in the JOMAPI call specifies a currently initialized client node.

1170 Node has finalised

Cause

This error occurs when the specified node handle used in a JOMAPI call is for a client node that has been finalized and is therefore no longer active.

Action

Modify the code to ensure that the node handle used in the JOMAPI call specifies a currently initialized client node.

1171 Configuration string is readonly

Cause

This error occurs when a call to the `jomSetNodeConfigurationString` JOMAPI was made by a non signed-on thread, or was attempting to change the JADE initialization file name or server type.

Action

Modify the code to ensure that calls to `jomSetNodeConfigurationString` JOMAPI are from a non signed-on thread, and only to change allowed values.

1172 Only one local server node is allowed

Cause

This error occurs when an attempt was made to initialize more than one database server node within the same program.

Action

This is an internal error. Contact JADE support if this error occurs.

1173 Class definition incomplete

Cause

This error occurs when an attempt was made to use a class with a new dynamic property definition in a transaction that has not been committed.

Action

Commit the transaction where the property is defined before using the class.

1174 Class is in use by another process

Cause

This error occurs when an attempt was made to delete a user class (**JadeUserClass**) that is in use by another process.

Action

Shut down the process or processes using the user class, before deleting the class.

1176 Operation not permitted: instances of this class or a subclass exist

Cause

This error occurs when a dynamic property cluster is deleted if instances of the class or subclass in which the cluster is defined exist.

Action

Delete instances of the class or subclass.

1177 The value of the Dynamic Property is not valid for the current property definition

Cause

This error occurs if the current value of the dynamic property cannot be converted to the current type of the property. For example, the value is an **Integer64** greater than **Max_Integer** and the type has been changed to **Integer**.

Action

To correct this, explicitly set the property to a value that is within the range for the type or change the property type to be compatible with the value.

1178 The type of the Dynamic Property value is not valid for type of the current property definition

Cause

This error occurs if the type of current value of the dynamic property cannot be converted to the current type of the property. For example, the value is an **Integer64** and the type has been changed to **TimeStamp**.

Action

To correct this, explicitly set the property to a value that is correct for the type or change the property type to be compatible with the value.

1179 A feature with this number is already defined

Cause

This error occurs if more than one feature on the class has the same number; for example, duplicate property or method numbers.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1180 An error occurred during Process sign off: refer to jommsg.log

Cause

This error occurs when a process is unable to sign off in the normal manner; for example, a lock exception could have been detected.

Action

No action is required, as the process has signed off.

Check the JADE messages log file (**jommsg.log**) for additional information that may clarify the cause of the error. If the error occurs again and your JADE licenses include support, contact your local JADE support center or JADE Support.

1182 Logical or Meta Certify errors encountered

Cause

This error is returned as the exit code of the JADE Logical Certifier when it is run as a non-GUI application and encounters errors in logical or meta certify operations.

Action

Examine the **_logcert.cls** file, which lists the errors. For a description of these errors and how to fix them, see Chapter 5 of the *JADE Object Manager Guide*.

1183 Uncompiled or in error methods remain

Cause

This error occurs if uncompiled methods or methods in error remain after recompiling.

Action

Correct or comment out the offending code before recompiling the methods.

1184 The requested method is not defined as a type method

Cause

This error occurs if you attempt to call a type method using the **Object** class **sendMsg** method.

Action

Call type methods using the **Object** class **sendTypeMsg** method.

1185 Invalid call to a type method

Cause

This error occurs if you attempt to call a class instance method using the **Object** class **sendTypeMsg** method.

Action

Call class instance methods using the **Object** class **sendMsg** method.

1186 Exceeded the maximum number of databases allowed

Cause

This error occurs when an executable program attempts to initialize more than the allowed number of databases; for example, in C++, too many calls to **jomInitialize** or having too many active **JoobContext** instances in .NET.

The current limit is 20 databases.

Action

Modify the code so that the number of concurrently initialized databases does not exceed the limit.

1187 Attempt to change transaction state within a constructor or destructor

Cause

This error occurs when a persistent transaction is committed while executing a constructor or destructor for a persistent instance, a shared transient transaction, or a shared transient instance or when an exception handler armed in a constructor or destructor aborts the transaction and attempts to return control within the constructor or destructor by returning **Ex_Resume_Next**.

Action

Change your application code to handle the committing of transactions appropriately.

1188 Constructor not called

Cause

This error occurs when an instance is created using the **create as** instruction and the constructor of the class of the instance being created has parameters.

Action

Replace the **create as** instruction with an extended **create** instruction and provide the parameters to the constructor of the instance being created.

1189 Constructor already called

Cause

This error occurs when a superclass constructor with parameters has already been called from an externally defined (that is, non-JADE) constructor or there were multiple calls to the **jomInheritCreate** Application Programming Interface (API).

Action

Change your application code to ensure that the **jomInheritCreate** API is called once only.

1190 A Batch Extract parameter file references a class or method that does not exist in the Schema

Cause

This error occurs if the parameter file specified when the **File** extract argument is used lists a class, method, interface method, property, or RPS mapping that does not exist in the schema specified in the argument list.

Action

Remove the class, method, interface method, property, or RPS mapping. Alternatively, add the entity to the schema before you attempt to run the application with these arguments.

1191 The requested method cannot be called

Cause

This error occurs when you attempt to call the **create** constructor or the **delete** destructor using an **Object** class send message method (for example, the **sendMsg** or **sendMsgWithParams** method).

Action

Change your application code to ensure that the constructor or destructor is not called directly.

1192 Feature is restricted to system processes

Cause

This error occurs when a user application makes a JADE request for a feature that is restricted to a system process; for example, when you attempt to execute **Object** class **sendMsg**, **sendMsgWithIOParams**, **sendTypeMsg**, **sendTypeMsgWithIOParams**, **sendTypeMsgWithParams**, **invokeIOMethod**, **invokeMethod**, and **setProperty** methods to access system-only features, as these are restricted to internal system processes only.

Action

Change your application code to prevent this.

1200 Class needs reorganization

Cause

This error occurs when an attempt is made to open a class that is marked for reorganization.

Action

Perform a reorganization of the classes before attempting to access their instances.

1201 Kernel stack overflow

Cause

This error occurs when your application code attempts to call a new method and it has already reached the maximum number of embedded method call levels (which is currently 256). One cause can be recursive method calls with no clear exit conditions.

In addition, this error occurs when:

- A method is invoked with a short timer event that performs a **doWindowEvents** method call. If the processing takes too long, the **doWindowEvents** event method processes the next timer event, eventually resulting in a stack overflow.
- Processing a Web service reply message that is too complex or poorly formatted.

Action

Change your application code to prevent too many embedded method call levels.

1203 Invalid exception class

Cause

This error occurs when an invalid exception class is specified while running an exception handler.

Action

Change your application code to specify a valid exception class.

1204 Invalid exception instance

Cause

This error occurs when an invalid exception instance is specified while raising an exception.

Action

Change your application code to specify a valid exception instance.

1205 Method aborted

Cause

This error occurs when an exception was detected and the exception handler aborted the action.

Action

Correct your application code, if required.

1206 Exception handlers stack overflow

Cause

This error occurs if too many exception handlers have been armed.

Currently, each process can have up to 128 global exception handlers armed at any one time. However, there is no JADE-imposed restriction on the number of non-global exception handlers that can be armed at any time by any one process.

Action

Change your application code to ensure that it does not arm more than the maximum number of exception handlers. (For details, see "[Handling Exceptions](#)", in Chapter 3 of the *JADE Developer's Reference*.)

1207 Invalid primitive number

Cause

This error occurs if an invalid primitive number is encountered.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1209 Application terminate request

Cause

This error is for internal use only and should not normally occur when executing JADE code.

The exception can be raised if the controlling application is external code. For example, if you access JADE using the .NET class library exposure the [terminate](#) instruction cuts back the execution stack and returns **APPLICATION_TERMINATED (1209)** to the controlling application.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1210 Duplicated relationship reference

Cause

This error occurs if JADE encounters a duplicate relationship reference. An example of this error follows. Consider the following sequence.

1. There are two classes: **Department** and **Employee**.
2. Each **Department** has a collection, **allEmployees**, of all of its employees.
3. Each **Employee** has a reference, **dept**, to their department.
4. An inverse reference is defined between **dept** and **allEmployees**, so that one department has many employees and a single employee has only one department. Both properties are defined to have an update mode of **manual/automatic**.
5. An instance of **Employee**, **e**, is created.
6. Two instances of **Department**, **d1** and **d2**, are created.
7. **e.dept** is set to **d1** to automatically add **e** to **d1.allEmployees**.
8. **e** is then manually added to **d2** with **d2.allEmployees.add(e)**, at which point JADE attempts to automatically update **e.dept**.

This causes a duplicated relationship reference error, as in attempting to update **e.dept** to **d2**, JADE will find that **e** is already referenced by **d1**. Because **e.dept** is a single-value property (that is, it is not a collection), **e** can be referenced only by one department at a time.

Action

Change your application code to avoid a duplicated relationship reference.

In relation to the above example, you should consider the following coding changes.

- If an employee can have multiple departments, the **dept** property of **Employee** should be changed to a collection, **allDepartments**, and the relationship between **Department** and **Employee** should be changed to **many-to-many**.

This then allows a single **Employee** object to be referenced by multiple **Department** objects.

- If an employee cannot have multiple departments, the code should be changed to avoid creating duplicate references.

1211 Cannot delete an object that has references

Cause

This error occurs if your application code attempts to delete an object that is referenced by one or more other objects. The object cannot be deleted until the references have been removed.

Action

Change your application code to remove references to the object before deleting it.

1212 Reference inverse missing

Cause

This error occurs as a result of an internal error in the JADE kernel (**jom.dll**).

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1213 Invalid schema

Cause

This error occurs if a JADE Object Manager sign-on is received with the name of a schema that does not exist.

Action

Correct the schema name that is supplied when signing on. If you are running a JADE application from the desktop, check the command line parameters to ensure that a valid schema name is present.

1214 Invalid application

Cause

This error occurs if a JADE Object Manager sign-on is received with the name of an application that does not exist in the schema that you have supplied.

Action

Correct the application name that is supplied when signing on. If you are running a JADE application from the desktop, check the command line parameters to ensure that a valid application name is present.

1215 Persistent objects cannot reference transient objects

Cause

This error occurs if your application code attempts to reference a transient object from a persistent object.

This can be caused directly by assigning a transient reference to the property of a persistent object or indirectly by assigning a persistent reference to the property of a transient object where an inverse exists back to the persistent object (which results in an attempt to assign a reference to the transient on the persistent object).

The error can also occur if you attempt to clone or copy a persistent instance of an object from a transient instance that has references to transient objects.

Action

Change your application code so that persistent objects do not reference transient objects.

You can use the [Process](#) class [allowTransientToPersistentInvs](#) method with a parameter of **true** to allow a reference to a persistent object from a transient object without its inverse being maintained. Alternatively, check the **Allow Transient to Persistent Reference** check box on the extended Define References dialog.

Calling the [allowTransientToPersistentInvs](#) method with a parameter of **true** is equivalent to checking this check box (until this method is called with a parameter value of **false**). For details, see "[Defining an Inverse Reference Property](#)" under "[Adding a Reference Property](#)", in Chapter 4 of the *JADE Development Environment User's Guide*.

The rules for assigning references are summarized as follows.

- Where no inverse references exist, a persistent object can reference other persistent objects but not transient objects.
- Where no inverse references exist, a transient object can reference both transient and persistent objects. However, shared transient objects cannot reference non-shared transient objects (for details, see error [1289 - Shared transient objects cannot reference non-shared transient objects](#)).
- Where inverse references exist (and the option to allow transient to persistent references is not enabled by the **Allow Transient to Persistent Reference** check box or the [allowTransientToPersistentInvs](#) method), the objects involved must both be transient or both persistent. A mixture of transient and persistent objects results in this error.

1216 Cannot invoke an abstract method

Cause

This error occurs if your application code attempts to invoke an abstract method. An abstract method has no code associated with it.

Action

You may need to change your application code so that you call a method in a subclass of the abstract class of this method.

1217 Application terminated

Cause

This error is for internal use only and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1219 Invalid calling thread

Cause

This error occurs when a call to the JADE Object Manager is made in a thread that is not currently signed-on.

Action

You must call the [jomSignOn](#) sign-on JADE API, before issuing calls from it.

1220 Local file mismatch

Cause

This error occurs if a client node has the read-only schema parameter set but one or more of the local schema files does not match the schema counterpart of the server. It can also occur if the **ReadOnlySchema** parameter is set to **true** in the [[JadeClient](#)] section of the JADE initialization file but not in the [[JadeServer](#)] section.

Action

Copy the local schema file from the server node to the client node and then start the transaction again.

1221 Object locked locally

Cause

This error is for internal use only and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1222 Cannot delete a currently active environmental object

Cause

This error occurs if your process attempts to delete one of its own environmental objects.

A process cannot delete an environmental object (system variable). Environmental objects (system variables) are:

- self
- app
- global
- system
- node
- process
- currentSchema
- rootSchema
- method
- exception
- currentSession
- appContext

Action

In JADE, you can reference the environmental objects by using a JADE system variable reserved word. Ensure that you do not attempt to delete any of these. In addition, ensure that your process does not add an environmental object to a collection (either by a manual add or through a relationship) and then try to delete the contents of the collection.

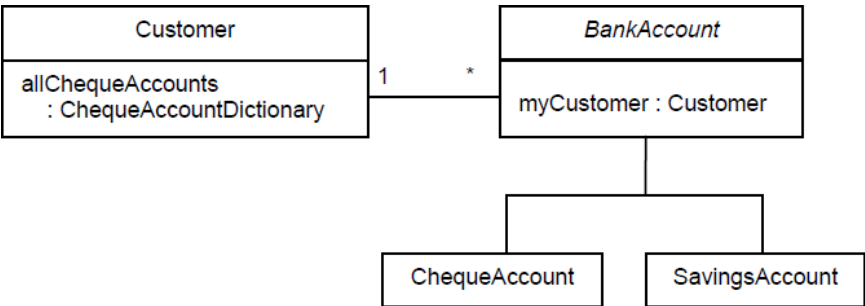
1223 Cannot establish inverse

Cause

This error occurs if your application code attempts to assign one end of a relationship and JADE is unable to assign the inverse (the other end of the relationship).

Note The error does not occur if you check the **Inverse Not Required** check box on the Define Reference dialog.

The following example assumes that the **Inverse Not Required** check box has not been checked.



Consider the following code.

```
vars
    chq : ChequeAccount;
    sav : SavingsAccount;
    cust : Customer;
begin
    chq := ChequeAccount.firstInstance;
    sav := SavingsAccount.firstInstance;
    cust := Customer.firstInstance;
    chq.myCustomer := cust; // This assignment succeeds
    sav.myCustomer := cust; // This assignment fails
end;
```

The assignment **chq.myCustomer := cust;** is valid. When **cust** is assigned to **chq.myCustomer**, JADE can add **chq** to the dictionary of cheque accounts on **cust** because **chq** is compatible with the membership of the dictionary.

The assignment **sav.myCustomer := cust;** results in a [1223 - Cannot establish inverse](#) error. When **cust** is assigned to **sav.myCustomer**, JADE cannot add **sav** to the dictionary of cheque accounts on **cust** because **sav** is not compatible with the membership of the dictionary. Consequently no inverse can be established.

When one end of a relationship is established, JADE must be able to assign at least one inverse to the originating object, or this error occurs. JADE cannot accept the assignment of **cust** to **sav.myCustomer** because it would result in a relationship with only one end of the relationship established.

Action

Change your application code in one of the following ways.

- Remove the assignment for which JADE is unable to create an inverse.
- Define an inverse so that the other side of the relationship can be established.
- Check the **Inverse Not Required** check box on the Define Reference dialog, which allows a reference to be set even if its inverse reference cannot be established.

Note that a reference that has no inverse (for example, **sav.myCustomer**) would have to be manually set to **null** if the **myCustomer** object were deleted.

1224 Automatic lock ignored

Cause

This error occurs if the following sequence has taken place.

1. The system attempted to place an automatic lock on an object.
2. A lock exception was raised.
3. The user exception handler returned the **continue** code, but the exception handler could not acquire the lock.

Action

Ensure that your application code does not continue lock exceptions until the actual lock has been acquired.

1225 Lock cannot be continued

Cause

This error occurs if the following sequence has taken place.

1. Your application code attempted to place an explicit lock on an object.
2. A lock exception was raised.
3. The user exception handler returned the **continue** code, but the exception handler could not acquire the lock.

Action

Ensure that your application code does not continue lock exceptions until the actual lock has been acquired.

1226 Uninitialized notification target

Cause

This error occurs if your application code attempts to perform a **beginNotification** with a null value of the first parameter (**theObj**) to the method call (that is, the target of the notification has not been initialized).

Action

Ensure that your application code provides an object reference with the **beginNotification** method.

1227 Notification subscriber not found

Cause

This error occurs if the notification delivery mechanism cannot find the notification subscriber.

Action

Remove a notification subscriber that does not exist, by applying the [endNotificationForSubscriber](#) method of the [Object](#) class. For details, see "Notifications", in Chapter 3 of the *JADE Developer's Reference*.

1228 Already in remote transaction state

Cause

This error occurs if a [beginTransaction](#) instruction is attempted on the server but a **beginTransaction** was already executed on the client without a [commitTransaction](#) instruction encountered on the client.

Action

Correct the portion of code in your application that resulted in the error.

1229 Not in local transaction state

Cause

This error occurs if a [commitTransaction](#) instruction is executed on the server but the [beginTransaction](#) instruction was executed on the client.

Action

Correct the portion of code in your application that resulted in the error.

1230 Update outside local transaction

Cause

This error occurs if the application is in a remote transaction state; that is, a [beginTransaction](#) instruction is executed on a client node and an update is attempted on the server node.

Action

Correct the portion of code in your application that resulted in the error.

1231 Invalid exception code

Cause

This error occurs when a user-defined exception is raised but the value of the [errorCode](#) property is zero (0).

Action

Correct the portion of code in your application that resulted in the error. For details, see "Exceptions", in Chapter 3 of the *JADE Developer's Reference*.

1232 Property has an undefined type: refer to jommsg.log

Cause

This error occurs if a property is encountered with an undefined (that is, **null**) type for a class in the current schema or one of its superschemas.

Action

This is an internal error indicating an inconsistency in the schema definition.

Refer to the JADE messages log file (**jommsg.log**) for more details and then contact your local JADE support center or JADE Support if your JADE licenses include support.

1233 Feature has an undefined type: refer to jommsg.log

Cause

This error occurs if a feature is encountered with an undefined (that is, **null**) type for the current schema or one of its superschemas.

Action

This is an internal error indicating an inconsistency in the schema definition.

Refer to the JADE messages log file (**jommsg.log**) for more details and then contact your local JADE support center or JADE Support if your JADE licenses include support.

1234 Slob or blob property has a zero sub id: refer to jommsg.log

Cause

This error occurs if a slob or blob property is encountered with a subobject identifier of zero (**0**). This is not allowed.

Action

This is an internal error.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1235 Required database upgrade has not been run

Cause

This error occurs if you attempt to sign on to JADE without having run the upgrade successfully.

Action

Refer to your release information, and check that you have completed all of the steps of the upgrade process.

1236 Cannot create this instance of a system class or subclass: refer to jommsg.log

Cause

This error occurs if your application code attempts to create an instance of a restricted **RootSchema** class or an instance of a subclass of a restricted **RootSchema** class.

Action

Refer to the JADE messages log file (**jommsg.log**) for the class number and object identifier (OID) of the offending object. Avoid creating instances of a restricted class. Alternatively, if your JADE licenses include support, contact your local JADE support center or JADE Support.

1237 Cannot delete this instance of a system class or subclass: refer to jommsg.log

Cause

This error occurs if your application code attempts to delete an instance of a restricted **RootSchema** class or an instance of a subclass of a restricted **RootSchema** class.

Action

Refer to the JADE messages log file (**jommsg.log**) for the class number and object identifier (OID) of the offending object.

Avoid deleting instances of a restricted class. Alternatively, if your JADE licenses include support, contact your local JADE support center or JADE Support.

1238 Exception handler invalid return code

Cause

This error occurs if an invalid code is returned from an exception handler; for example, when resuming a non-resumable exception or continuing a non-continuable exception.

Action

Check the JADE messages log file (**jommsg.log**) for the **resumable** and **continuable** attributes on the exception you are trying to resume or continue. Change the logic of your exception handler so that it does not attempt to resume a non-resumable exception or continue a non-continuable exception.

Notes If this exception is caught by an exception handler which then tries to continue the exception, exception *1239 - Nested exceptions limit exceeded* is raised eventually, due to the repeated *1238 - Exception handler invalid return code* exceptions.

It is recommended that exception handlers check if an exception is continuable or not before attempting to return **Ex_Continue**. It is also recommended that the exception handler includes checks to detect a nested exception situation. It may also be beneficial to specifically check for the *1238 - Exception handler invalid return code* exception.

1239 Nested exceptions limit exceeded

Cause

This error occurs if the maximum number of nested exceptions is exceeded (the limit is currently 20).

A common cause is when an exception handler itself raises further exceptions; for example, if the original exception handler displays a message box or dialog on the client workstation and the original exception caused a thin client disconnect, this then causes a further exception, and so on.

Action

Check your application messages log (the log file with the name of the application that raised the unhandled exception). A history of nested exceptions is shown in the Exception Stack History section of the application messages log file. The initial exception that triggered the sequence of nested exceptions is listed at the base of the exception stack. Knowing this may help you to determine the best way to deal with the problem.

1240 Exceeded the maximum number of sub-objects allowed for a class

Cause

This error occurs if your application code attempts to add more than 255 subobjects (exclusive collections) to a class. A class cannot have more than 255 subobjects.

Action

Create a subclass on which to add the additional subobjects. You should always try to avoid adding more than 255 subobjects.

1241 Exceeded the maximum number of slob and blobs allowed for a class

Cause

This error occurs if your application code attempts to add more than 255 slob or blobs to a class. A class cannot have more than 255 slob or blobs, or both slob and blobs.

Action

Create a subclass on which to add the additional slob and blobs. You should always try to avoid adding more than 255 slob or blobs, or both slob and blobs.

1242 A method executing in another node was aborted

Cause

This error occurs if an exception occurs in a server method and no exception handler is armed on the server node for the exception or the exception handler returns **Ex_Abort_Action**.

Action

Find out which server method has caused the exception, and arm an exception handler on the server node for the server method exception.

The default exception handler is invoked for the exception. If this is a JADE exception handler, the application stack is written to the application log, but the corresponding dialog is not displayed.

The [extendedErrorText](#) property of this exception contains the text that corresponds to the original server method exception.

Note You cannot return from a global exception handler. Returning **Ex_Resume_Next** from a global exception handler is handled in the same way as returning **Ex_Abort_Action**.

1244 Invalid application type

Cause

This error occurs if there is an attempt to start a GUI or a GUI, no-forms application on the server. Starting a GUI server application is not allowed.

Action

Ensure that GUI applications are not started on a JADE server.

1245 Incorrect application type

Cause

This error occurs if the application type does not match the start-up location or the application type is incorrectly selected.

Action

Set the correct application type on the **Application** sheet of the Define Application dialog.

Ensure that GUI applications are not started on a JADE server.

1246 Automatic update aborted

Cause

This error occurs if an automatic update operation (that is, automatic inverse maintenance or dictionary key maintenance) is aborted as a result of a prior error.

Action

Change your application code to resolve the prior error.

1247 Buffer error

Cause

This is an internal error indicating that a corruption has been detected in an object buffer.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1248 Cannot execute a user schema server method in this process

Cause

This error occurs if an attempt is made to invoke a server method while the JADE Painter is executing user schema methods. This is not permitted.

This occurs most often when the JADE Painter has called a user schema method to implement development time behavior for one of your (typically subclassed) controls; for example, [paint](#), [windowCreated](#), or [controlNamePrefix](#). If your method invokes any server methods (or it calls other methods that do), this error occurs.

Action

Avoid calling [serverExecution](#) methods from methods invoked by the JADE Painter. You can do this by removing the calls entirely or by guarding calls to server methods with the [isInPainter](#) method for controls.

1249 Attempt to access method in an undefined library: refer to jommsg.log [obsolete]

Cause

This error occurs if JADE attempts to access a method (typically an external method) and the method has an undefined (null) library.

The definition of methods without a library is not permitted.

Details of the error (including the method name and the type on which it is defined) are written to the JADE messages log file ([jommsg.log](#)).

Action

Recompile the method. If the problem persists, this error can indicate an internal JADE error, and you should contact your local JADE support center or JADE Support if your JADE licenses include support.

1250 Attempt to change key value in a mapping method get operation: refer to jommsg.log

Cause

This error occurs if a mapping method for a property that is used as a key in a dictionary returns a value in the [_value](#) parameter that is not equal to the value of the property in the object. This is not allowed.

Note The restriction does not apply to properties used as keys in dynamic dictionaries, unless defined in the schema as keys in any other dictionary.

An example of such a method is as follows.

```
// mapping method for "rate" property which is a dictionary key
rate(set: Boolean; _value: Real io) mapping;
begin
    if not set then
        _value := _value * app.getCurrentRateModifier;
    endif;
end;
```

If **app.getCurrentRateModifier** returns a value that is different from the value returned when the **rate** property was set, executing this method raises a *1250 - Attempt to change key value in a mapping method get operation* exception during automatic inverse maintenance or when adding to or removing the instance from the collection, because **_value** is changed to be different from the value of the **rate** property in the object. JADE does not allow this because such a mapping method would defeat dictionary key maintenance (the method may return one value for the key when an object is added to a dictionary and a different value for subsequent accesses, causing the object not to be found in the collection).

Notes By implication, virtual properties can never be used as keys because they are never stored in the object; their values are always derived from their mapping method. The **_value** parameter will therefore always be different from the object value (which is always **null** for virtual collections).

Action

Change the mapping method so that it always returns the object property value.

1251 A stack overflow exception handler exceeded 20 call levels

Cause

This error occurs if JADE detects a stack overflow condition.

JADE initially raises exception *1201 - Kernel stack overflow*, during which the corresponding exception handler or handlers are invoked. However, during the process of the original exception, the stack continued to grow to more than 20 call levels. This error causes the current JADE process to terminate.

Action

Change your application code to solve both stack overflow problems. (For details, see error *1201 - Kernel stack overflow*.)

1252 Transaction must be aborted

Cause

This error occurs if your application code attempts to commit a transaction that must be aborted.

The most-common way for a transaction to enter this state is when an error occurs during an automatic update (inverse maintenance or dictionary key maintenance).

If an error occurs during an automatic update, JADE must abort the transaction. If your exception handler does not abort the transaction, JADE does so automatically when the stack unwinds after handling the error.

Any attempt to commit the transaction while it is in this state raises this error.

Action

Ensure that your application code does not try to commit a transaction while JADE is in abort transaction state.

1253 Application did not complete signon within the specified startup waitTime

Cause

This error occurs if the application thread for an application does not complete sign-on within the specified wait time.

This error is only an indication that the application thread did not complete sign-on within the specified time. When this error occurs, JADE:

- Is unable to determine if the application being started is currently running (for example, if the operating system is overloaded or the started application has not yet had time to start up completely).
- Cannot provide a process when it returns.

Action

Check the load and available resources on the machine on which the application is starting. Check the JADE messages log file (**jommsg.log**) for error messages.

The **MaxWaitAppStart** parameter in the [[JadeServer](#)] or [[JadeClient](#)] section of the JADE initialization file can be adjusted, if required. The default value for this parameter is zero (**0**); that is, the application never times out.

Take the resource requirements of your system into account when you specify a value (for example, if your machine has a high system load, a value of 45,000 milliseconds, or 45 seconds, may not be sufficient to complete sign-on).

If this does not resolve the problem, contact your local JADE support center or JADE Support if your JADE licenses include support.

1254 Invalid transient class number

Cause

This error is reserved for future use. It should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1255 Invalid transient class

Cause

This error is reserved for future use. It should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1256 A transient class cannot be the superclass of another class

Cause

This error is reserved for future use. It should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1257 Transient class already registered with this number: refer to jommsg.log

Cause

This error is reserved for future use. It should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1258 The current operation cannot be performed on a transient class

Cause

This error is reserved for future use. It should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1259 Transient subclasses of Collection are not allowed

Cause

This error is reserved for future use. It should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1260 Exceeded the maximum number of transient classes allowed for a process

Cause

This error is reserved for future use. It should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1263 Cannot create persistent or shared transient instances of a transient class

Cause

This error occurs if an attempt is made to create a persistent or shared transient instance of a class that is defined so that transient instances only are allowed to be created.

Action

Change the lifetime options of the class or create the object as a transient.

1264 Invalid type for property

Cause

This error occurs if your application code attempts to assign a value to a property where the type of the value and the type of the property are not compatible.

Action

Change your application code to avoid the incompatible assignment, or convert the value to an appropriate type before the assignment.

1265 Environmental object operation is out of scope for process

Cause

This error occurs if a process attempts an operation on an environmental object that is out of scope (for example, a foreign node). Typically, the operation would be accessing an attribute or executing a method. Only a limited number of operations are available on foreign environmental objects. Environmental objects (JADE system variables) are:

- self
- app
- global
- system
- node
- process
- currentSchema
- rootSchema
- method

- exception
- currentSession
- appContext

This error occurs in a secondary database server when the node representing the primary server, the *replay* node, is used as the method receiver for methods that start node sampling or request statistics. Methods that should not be called for a *replay* node include [beginSample](#), [endSample](#), [getLocks](#), [getMutexCounts](#), [getObjectCaches](#), [getQueuedLocks](#), [getRequestStats](#), [getRpcServerStatistics](#), [wbemListClasses](#), [wbemListInstanceNames](#), [wbemQueryQualifiers](#), and [wbemRetrieveData](#).

A *replay* node can be identified by testing whether the [nodeRole](#) method returns the **Node** class **Role_Replay** constant.

Action

Change your application code so that it does not attempt to perform the offending operation on the foreign environmental object.

1266 A non-shared transient object is being accessed by another process

Cause

This error occurs if you attempt to pass a non-shared transient object reference from one process to another. This is sometimes referred to as a *dirty* sharing of transients.

Action

Correct your application code. You can instruct JADE to ignore some kinds of non-recommended behavior. For details, see "[JADE Object Handling](#)", in Chapter 1 of the *JADE Object Manager Guide*.

1267 Notification info object too big

Cause

This error occurs if the size limit for notification data has been exceeded.

Notifications containing binary and string (**Binary**, **String**, **StringUtf8**) data of up to 48K bytes can be sent across the network. For applications running within the server node, the limit for notifications containing binary or string data is 2G bytes. Note, however, that this applies only to single user and server applications. In multiuser applications, persistent notifications are sent via the database server, even if the receiving process is on the same node as the sender.

Refer to the JADE messages log file ([jommsg.log](#)), to get an indication of how much the size limit is being exceeded by the [causeEvent](#) notification.

Action

Ensure that the data being passed on the notification is less than the recommended size limit.

In addition, avoid sending large amounts of data using the notification. It is preferable to pass an object reference, rather than a large primitive type value.

1268 Already in transient transaction state

Cause

This error occurs if your application code attempts to enter transient transaction state for a database that is already in transient transaction state. This occurs if you attempt to enter transient transaction state in the **create** method of a [WebSession](#) subclass.

Action

Change your application code so that it does not attempt to enter transient transaction state for a database that is already in transient transaction state. For details, see the [Process](#) class [isInTransientTransactionState](#) method.

1269 Not in transient transaction state

Cause

This error occurs if your application code attempts to end transient transaction state for a database that is not in transient transaction state.

Action

Change your application code so that it does not attempt to end a transient transaction state for a database that is not in transient transaction state. For details, see the [Process](#) class [isInTransientTransactionState](#) method.

1270 Update outside transient transaction

Cause

This error occurs if your application code requests an update operation for a shared transient object outside transient transaction state. Transient objects accept updates only in transient transaction state.

Action

Ensure that updates are made within transient transaction state. For details, see the [Process](#) class [isInTransientTransactionState](#) method.

1271 An attempt was made to perform an operation eg. beginTransaction etc that is prohibited in this context.

Cause

This error occurs if an attempt is made to perform an action such as [beginTransaction](#), which is prohibited in the context; for example, attempting to begin a transaction when using the JADE Report Writer.

Action

Change your application code to avoid the prohibited action in the context.

1272 Attempt to add duplicate element to array with inverse

Cause

This error occurs if JADE encounters a duplicate entry in an array that participates in an inverse relationship. An example of this error is as follows.

1. There are two classes: **Department** and **Employee**.
2. Each **Department** has an array collection, **allEmployees**, of all of its employees.
3. Each **Employee** has a reference, **myDept**, to their department.
4. An inverse reference is defined between **myDept** and **allEmployees**, so that one department has many employees and a single employee has only one department. The **allEmployees** property is defined with an update mode of manual.
5. An instance of **Employee**, **e**, is created.
6. An instance of **Department**, **d**, is created.
7. **e** is added to **d.allEmployees** with **d.allEmployees.add(e)**, which automatically sets **e.myDept** to **d**.
8. **e** is again added to **d.allEmployees** with **d.allEmployees.add(e)**, at which point JADE attempts to automatically update **e.myDept**.
9. This causes a *1272 - Attempt to add duplicate element to array with inverse* error, as in attempting to update **e.myDept** to **d**, JADE finds that **d** is already referenced by **e.myDept**. This is disallowed, because if it were allowed and later one of the references to **e** was removed from the array **d.allEmployees**, the reference from **e.myDept** to **d** would be automatically removed even though there was still a copy of **e** in the array.

Note that having duplicates in an array is not an error if there is no inverse relationship.

Action

Change your application code to avoid a duplicate entry in an array that participates in an inverse relationship.

With respect to the above example, you should consider the following coding changes.

- If an employee must appear in a department collection more than once, the **myDept** property of **Employee** should be changed to an array collection (for example, **allDepartments**) and the relationship between **Department** and **Employee** should be changed to many-to-many.

This then allows a single **Employee** object to be referenced by the **Department** object many times.
- If an employee cannot appear more than once in a department, the code should be changed to avoid creating duplicate entries.

1273 The requested constraint is not defined for this object class

Cause

This error occurs if JADE cannot find the requested condition for the class of the object.

Action

Remove or correct the invalid condition.

1275 Feature not defined so do not set inverse

Cause

This error occurs if an internal error is encountered whilst maintaining an inverse reference. It should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support to report this error.

1276 Potential cache inconsistency

Cause

This error occurs under the following conditions.

- If a server method attempts to exclusively lock a persistent object while in transaction state, and the object is already locked by a method executing on the client node.
- If a client method attempts to exclusively lock a persistent object while in transaction state, and the object is already locked by a server method request.
- If a server method attempts to lock a persistent object when the object is exclusively locked by a method executing on the client node and transaction state is in effect on the client node.
- If a client method attempts to lock a persistent object when the object is exclusively locked by a server method request and transaction state is in effect on the server node.

Action

Change your application code to ensure that the requested object is not already locked when calling a remote method that updates it, or so that the method is not accessing an object that has been updated but not committed on the other node.

1277 Lock waiting on queue to be retried

Cause

This error occurs only in sampling output.

This error, recorded in the lock sample output, means that the lock was queued to wait for the lock and will now be retried in an attempt to get the requested lock.

This error code is not passed back to the user systems.

Action

There is no specific course of action.

1278 Change in a constraint removed all inverses

Cause

This error occurs if the last inverse on a reference has been removed as a result of changing a condition.

Where a condition expression can act as a constraint on an object reference, changing a condition can result in an inverse being removed, where the changed condition no longer applies to any of the constraints on the reference. For example, the following references are defined on class **Company** with inverse **myCompany**.

```
allCasual of type Employee, where isCasual
```

```
allPermanent of type Employee, where isPermanent
```

The **Employee** class has the property **hours**, with the following conditions.

```
isCasual(): Boolean condition;  
begin  
    return hours <= 20;  
end;  
  
isPermanent(): Boolean condition;  
begin  
    return hours >= 38;  
end;
```

This error occurs when the **hours** attribute is assigned a value greater than **20** but less than **38**. Since neither condition applies (they are both **false**), the inverse relationship is removed.

Action

Change your application code to make sure that at least one condition is set, or apply a catch-all condition to the inverse reference (for example, **isOnPayroll**).

Alternatively, if an inverse is not required for automatic maintenance of the relationship, you can check the **Inverse Not Required** check box on the Define Reference dialog used to prevent the exception occurring.

1279 Automatic delete aborted

Cause

This error occurs if an automatic delete operation (that is, automatic delete children maintenance) is aborted as a result of a prior error.

Action

Change your application code to resolve the prior error.

1280 Delete children operation left children undeleted

Cause

This error is raised by the deletion of members of a child class, if the number of members of a collection processed during a delete of the children does not agree with the initial number of children.

If the destructor of a child changes the keys of a collection so that children are missed (because children later in the collection are changed to have a key less than or equal to the child being deleted), those children can be left as orphans (with pointers to the deleted parent).

Action

Change your application code to ensure that the destructors do not change the order of members of the collection while they are being deleted.

1282 An exception handler cannot be disarmed while it is active

Cause

This error occurs if there is an attempt to disarm an exception handler while an exception handler is executing.

Action

Change your application code to prevent this.

1283 Failed to open package agent

Cause

This error occurs if an attempt to open a package fails.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support to report this error.

1284 User sign-on is currently disabled

Cause

This error occurs if an attempt to sign on is made while signing on is disabled; for example, when attempting to sign on to a secondary server node in a Synchronized Database Service (SDS) environment while the secondary server node is processing a database reorganization or when a secondary database is shut down with incomplete transactions and it cannot establish communications with its primary.

Action

Try to sign on again later, after signing on has been re-enabled (for example, if the primary was unavailable because of a network outage that has been resolved).

If this sign-on error is raised when you are attempting a hostile takeover, sign on to the SDS Administration application in single user mode and then perform a hostile takeover.

1285 Cannot create instance of a class: refer to jommsg.log

Cause

This error occurs if your application code attempts to create a persistent, transient, or shared transient instance of a class for which the relevant **`persistentAllowed`**, **`transientAllowed`**, or **`sharedTransientAllowed`** class option is not **`true`**.

Action

Refer to the JADE messages log file (**`jommsg.log`**) for details. Change the relevant class option to **`true`**, or change your application code to avoid the offending operation.

1286 Cannot delete instance of a class: refer to jommsg.log

Cause

This error occurs if your application code attempts to delete a persistent, transient, or shared transient instance of a class for which the relevant **`persistentAllowed`**, **`transientAllowed`**, or **`sharedTransientAllowed`** class option is not **`true`**.

Action

Refer to the JADE messages log file (**`jommsg.log`**) for details. Change the relevant class option to **`true`**, or change your application code to avoid the offending operation.

1287 Cannot subclass a final class: refer to jommsg.log

Cause

This error occurs when opening a class if a superclass is marked as final.

Action

Refer to the JADE messages log file (**`jommsg.log`**) for details. Remove any subclasses of the final class.

1288 No more instances can be created because the instance id is at the maximum value

Cause

This error occurs when the instance id for a class reaches the maximum value and a request is made to create a new instance.

Action

Change your application code so that it no longer attempts to create a persistent instance of the affected class. One possibility is to define and use a subclass of the affected class.

1289 Shared transient objects cannot reference non-shared transient objects

Cause

This error occurs if your application code attempts to reference a non-shared transient object from a shared transient object. This can be caused directly, by assigning a non-shared transient reference to the property of a shared transient object, or indirectly, by assigning a shared transient reference to the property of a non-shared transient object where an inverse exists back to the shared transient object (which results in an attempt to assign a reference to the non-shared transient on the shared transient object).

Action

Change your application code so that shared transient objects do not reference non-shared transient objects.

You can use the **Process** class **allowTransientToSharedTranInvs** method with a parameter value of **true** to allow a reference to a shared transient object from a non-shared transient object without its inverse being maintained. The rules for assigning references are summarized as follows.

- Where no inverse references exist, a shared transient object can reference persistent or other shared transient objects but not non-shared transient objects.
- Where no inverse references exist, a non-shared transient object can reference persistent and both shared and non-shared transient objects.
- Where inverse references exist (and the option to allow non-shared transient to shared transient inverse reference is not enable by the **allowTransientToSharedTranInvs** method, as above), the objects involved must both be persistent, both shared transient, or both be non-shared transient. A mixture of transient and persistent objects results in this error.

1291 GUI request not allowed in this context

Cause

This error occurs if a non-GUI application attempts to call the **Application** class **msgBox** method.

Action

Change your application code so that the GUI message box is not called.

1292 Maximum number of local exception handlers exceeded

Cause

This error occurs if the application attempts to arm more than 128 concurrent local exception handlers in a method.

Action

Change your application code so that the maximum limit of 128 locally armed exception handlers is not exceeded.

1293 Required database upgrade has not been completed

Cause

This error occurs in either of the following situations.

- When upgrading to a new JADE release, the database upgrade did not fully complete.
- When attempting to use a release of JADE with an older incompatible database; for example, system files.

Action

Perform one of the following actions.

- If the problem is a result of the database upgrade not completing, determine why it did not complete and then take the appropriate steps to remedy the situation.
- If the problem is a result of using an older incompatible database, determine why JADE was trying to use the older system files.

1294 Cannot read object modified by a different user

Cause

This error occurs if a process attempts to access a blob or slob object buffer that has been modified by a different process and the transaction that updated the buffer has not yet committed.

A JADE lock operation might be more appropriate to access the owner object of the blob or slob in question.

Action

The possible actions that you could take are as follows.

- Write an exception handler that retries the operation or aborts the action. The operation is successful as soon as the other process has committed the object and released its lock.
- Change your application code to lock the owner object of the blob or slob. If locks are used and the object is being updated by another process (and consequently has an exclusive lock), no attempt is made to read the uncommitted object and a **LockException** occurs instead.
- Leave the application as it is, since this is not a fatal error or a permanent situation. The operation can be safely resubmitted.

1295 Cannot update a non-environmental object on a Secondary database

Cause

This error occurs if there is an attempt to update a non-environmental object on a secondary database. The secondary database server maintains a secondary copy of the database synchronized with a primary updating copy and supports read-only access only. Application programs cannot update a secondary database.

The following environmental objects can be updated.

- **process**
- **node**
- **system**

An attempt to update any other object on the secondary database causes this exception to be raised.

Action

Change your application code to prevent this.

1296 SDS Transaction Overlap

Cause

This is an internal error and should not normally occur. It occurs if the Synchronized Database Service (SDS) tracking attempts to replay a transaction for a shadow process that is already in transaction state.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, and provide the following information:

- The JADE messages log file (**jommsg.log**) and SDS log files from the primary and secondary systems
- A process dump of SDS secondary (if one was produced)
- A copy of the journal being processed at the time of the failure

This error is not usually fatal, so it may be possible to resume tracking; otherwise, restart the SDS secondary database.

1298 Immediate sdsCauseEvent is not allowed in this SDS context

Cause

This error occurs if the **sdsCauseEvent** method has the **immediate** parameter set to **true** on a SDS primary system or a non-SDS system.

Action

If using **sdsCauseEvent** method on a SDS primary system or a non-SDS system, ensure that the **immediate** parameter is set to **false** and that it is used only when in a transaction state.

1300 through 1399 Collection Exception Errors

This section describes errors that can occur when using collections.

1300 Array index out of bounds

Cause

This error occurs when the index used in an array index operation is outside the bounds of the array; that is, the index is less than one (1) or greater than the highest allocated position in the array.

Action

Determine the cause of the invalid index and correct the code that led to the error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1301 Entry not found in collection

Cause

This error occurs from collection operations that are passed the value of an entry that is expected to be in the collection but was not found.

The entry was never in the collection or it has already been removed. An example of this sort of operation is the [remove](#) method of the [Array](#) class, which is passed the value of the entry to remove.

Action

Determine why the entry is not in the collection and correct the code that led to the error.

The method and source position containing the expression at which the exception was raised are reported in the error description.

1302 Collection description invalid

Cause

This error occurs if the collection description is incomplete or inconsistent.

When a [Collection](#) class is first opened, the collection, along with the JADE Object Manager, analyzes its collection description defined in the schema. If this description is incomplete or inconsistent, a 1302 error occurs.

Action

Check that the class definition of the collection that is referenced is complete; for example, if it is a dictionary, check that its keys are correctly defined.

If JADE allows you to define a collection with an incomplete or inconsistent description, this is a fault in the JADE software and should be reported. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1303 Iterator has no collection to iterate

Cause

This error occurs if your application code is using a collection iterator but it has not been given a collection to iterate. This occurs if the iterator is created outside of the collection iterator protocol, or if it is used with a collection that did not correctly implement the `createIterator` method.

Action

To create an iterator correctly, use the `createIterator` method of the collection and ensure that the collection implements this method correctly.

1304 Start from only valid for dictionaries

Cause

This error occurs if a collection that is not of type `Dictionary` is passed to an iterator relative key `start` method. As the iterator relative key `start` methods are not part of the public interface to iterators, this error is unlikely to occur in practice.

Action

Determine why the iterator relative key `start` method is receiving a non-dictionary collection and correct the problem.

1305 Relative key operation not valid for Start

Cause

This error occurs if an internal problem is detected in the dictionary iterator relative key access protocol.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1306 Iterator not associated with this dictionary

Cause

This error occurs if your application code passes an iterator to a `Dictionary` class relative key `start` method (`startKeyxxx`) when the dictionary did not create the iterator.

Action

Ensure that the iterator passed to the relative key `start` method was created by the dictionary using the `createIterator` method.

1307 Cannot pass an exclusive subobject reference to this method

Cause

This error occurs if your application code attempts to pass an exclusive (that is, non-shared) object reference to a collection method that expects a shared reference. An exclusive reference, by definition, can be owned only by its parent object.

Action

Determine why an exclusive reference is being used in this way and then correct the code that led to the error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1308 Cannot update an array during iteration

Cause

This error occurs when iterating over an array if entries are added to or removed from the array during the iteration. Adding and removing entries from an array causes entries above the insertion or removal point to be moved up or down, which renders the iteration invalid.

Note Array entries can be removed or added automatically by the JADE Object Manager if the array is an automatic inverse property.

Action

If it is necessary to update an array while accessing its elements sequentially, use array indexing instead of an iteration.

1309 Object already added to this set or dictionary

Cause

This error occurs if you attempt to add an object to a set and the object already exists in the set, or to a dictionary when an entry with the same key and object identifier (OID) already exists in the dictionary. Duplicate objects are not allowed in an automatically maintained member key dictionary, even if the dictionary allows duplicate keys. It is possible to add the same object more than once to a manually maintained dictionary if the object is added with different keys.

Action

Determine why an object is being added to the set or dictionary more than once and correct the code that led to the error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1310 Key already used in this dictionary

Cause

This error occurs if your application code attempts to add a duplicated key entry to a dictionary that specifies duplicate keys are not allowed.

Action

Determine why a duplicated key is being added to the dictionary and correct the code that led to the error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1311 Null object reference cannot be passed as a value argument to a collection method

Cause

This error occurs if a **null** object reference is passed to a **Collection** class method (such as **add** or **remove**).

Action

Determine why a **null** object reference is being passed and correct the code that led to the error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1312 Class of value passed to a collection method incompatible with membership

Cause

This error occurs if the class of an object reference passed to a **Collection** class method (for example, **add** or **remove**) is not assignment-compatible with the defined membership of the collection.

Action

Determine why the class is not assignment-compatible and correct the code that led to the error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1313 Cannot iterate a virtual collection backwards

Cause

This error occurs if your application code attempts to iterate a collection type that does not support reverse iteration.

Action

Remove any code that attempts to iterate this collection type in reverse order. Check everywhere that you use the **back** method of the **Iterator** class or the **reversed** option with the **foreach** instruction.

1314 The ordinal position of the requested key is out of range

Cause

This error occurs if your application code attempts to retrieve a key from a dictionary and the **ordinal** parameter is out of range. You can use the **getCurrentKey** method of the **Iterator** class to retrieve keys from the dictionary while iterating through the dictionary. However, the value applied as the ordinal position is out of range.

Action

Check your application code and find out why it is applying a value that is too large in the **ordinal** parameter of the **getCurrentKey** method.

1315 New collection block oid mapped differently to existing blocks

Cause

This error occurs when an operation that is adding a new entry to a collection causes a new collection block to be created and the class mapping in the object identifier (OID) for this new block does not match the class map in the oid of the collections root block. This error would normally occur only with root schema collections that have at some time been created by a system user and are now being modified by a non-system user.

Action

The JADE messages log file (**jommsg.log**) contains information that identifies the collection and its usage context. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1316 Collection access path no longer current

Cause

This error is raised by an internal exception raised during automatic key change handling by the dictionary methods of the kernel.

Action

If your JADE licenses include support, contact your local JADE support center and provide them with the JADE messages log file (**jommsg.log**) and any other relevant diagnostics.

1317 Object group operations not valid for collection containing primitive types

Cause

This error occurs when there is an illegal attempt to make a group read or lock operation using a collection of primitive types.

Action

Change your application code to prevent this.

1318 Dynamic dictionary definition incomplete

Cause

This error occurs if your application code has not completed a required key definition for a dynamic dictionary method.

Action

Use the [endKeys](#) method of the [DynaDictionary](#) class to ensure that the key definition is completed correctly.

1319 Dynamic dictionary membership not set

Cause

This error occurs if your application code fails to set the dictionary membership for a dynamic dictionary key definition method.

Action

Use the [setMembership](#) method of the [DynaDictionary](#) class to define the membership for the dictionary.

1320 Operation invalid - collection not empty

Cause

This error occurs if the collection is not empty when your application attempts to perform an operation (for example, changing properties such as the membership or key definitions of a dynamic dictionary) that requires the collection to be empty.

Action

Ensure that the collection is empty before attempting this operation.

1321 Dynamic dictionary keys not defined

Cause

This error is raised by the dynamic dictionary [endKeys](#) method if your application code has failed to add keys to the dictionary definition.

Action

Use the [addMemberKey](#) method or the [addExternalKey](#) method of the [DynaDictionary](#) class to define one or more keys before ending the key definition.

1322 Total key size exceeds 512 characters

Cause

This error is raised by the dynamic dictionary [addMemberKey](#) method or [addExternalKey](#) method if the total size of your keys when concatenated exceeds the dictionary limit of 512 character units.

Action

Ensure that the total concatenated key size is less than or equal to the documented limit of 512 character units.

Note The limit of 512 character units applies to the *total* concatenated length of the dictionary key. For example, in practice a single binary key is restricted to exactly 512 octets in an ANSI build and 1024 octets in a Unicode build, since in the Windows UTF16 encoding a character unit is two octets in length. If you have multiple keys of different length, the limit is still precisely 512 octets for ANSI and 1024 octets for Unicode. However, a single string key is restricted to 511 effective characters in that the null termination character subsumes one additional character. If you had two concatenated string keys, they would be limited to 510 characters between them, as each string key requires a **null** termination character.

The only reason the physical (octet) length doubles in a Unicode system is to enable a schema containing dictionary key definitions that are at the logical (character length) limit in an ANSI system to load in Unicode without exceeding the physical key size limit.

1323 Dynamic dictionary operation requires member keys

Cause

This error is raised by dynamic dictionary methods (for example, [addMemberKey](#), [add](#), and [remove](#)) if you have not defined member keys. These methods are valid only when the dictionary has member keys defined.

Action

Change your application code to use the correct methods for the type of dynamic dictionary being used, or define the required keys.

1324 Dynamic dictionary operation requires external keys

Cause

This error is raised by dynamic dictionary methods (for example, [addExternalKey](#) and [putAtKey](#)) if you have not defined external keys. These methods are valid only when the dictionary has external keys defined.

Action

Change your application code to use the correct methods for the type of dynamic dictionary being used, or define the required keys.

1325 String or binary key requires a non-zero length

Cause

This error is raised by the dynamic dictionary [addExternalKey](#) method when your application code attempts to define a [String](#) or [Binary](#) key with a zero length.

Action

Change your application code to ensure that you specify a key of non-zero length.

1326 Cannot change dictionary specification after endKeys called

Cause

This error is raised by the dynamic dictionary [addMemberKey](#) or [addExternalKey](#) methods if these methods are called after the [endKeys](#) method has been called to end the definition of the keys for the dynamic dictionary.

Action

Change your application code to ensure that all keys are specified before calling **endKeys**, or start a new dynamic dictionary key specification by using the [clearKeys](#) method to clear any existing key specification.

1327 Dynamic dictionary definition cannot span nodes

Cause

This error is raised by dynamic dictionary methods when the definition of the dictionary is attempted across nodes. Once the definition of the keys has begun, the dynamic dictionary is in a state that is unfit for processing on another node. Definition commences with the initial [setMembership](#) call but can recommence with a [clearKeys](#) call or another **setMembership** call (which performs an implicit **clearKeys**), and these calls could be performed on another node.

Definition on a node is completed with an [endKeys](#) call that causes the meta-information defining the dictionary to be stored in the [DynaDictionary](#) object.

Action

Change your application code so that the definition of the dynamic dictionary occurs entirely within the node that began the definition; that is, the node that last performed a **setMembership** or **clearKeys** operation on the dynamic dictionary.

1328 Iterators cannot be shared between heterogeneous platforms

Cause

This error occurs when an iterator is used on a node with a different byte order to the node that created the iterator. For example, if you create an iterator in a server method executing on an IBM RS/6000 server and you then attempt to use the iterator on a client or application server executing on an Intel platform, this error is raised.

Action

Change your application code to avoid this situation.

1329 Requested collection block size exceeds block size limit

Cause

This error is returned by the [Collection](#) class [setBlockSize](#) method if the requested block size value in *entries per block* multiplied by the current collection *entry size* in bytes exceeds the system-defined upper limit for collection block sizes, as defined by the [SystemLimits](#) category [MaximumCollectionBlockSize](#) global constant.

Action

Change your application code to ensure that the *entries per block* multiplied by the current collection *entry size* in bytes does not exceed the specified upper limit.

1332 JadeBytes access is out of bounds

Cause

This error is raised by methods of the [JadeBytes](#) class called with parameter values that are outside the bounds of the allocated length of the data content.

When you attempt to retrieve part of the contents of a **JadeBytes** object by using the [getData](#) method or update part of the contents using the [putData](#) method, the value of the:

- **offset** parameter must be one (1) through the allocated length
- **offset** parameter and the **length** parameter combined must not exceed the length of the current content

Action

Ensure that the values of parameters are within the bounds of the allocated length of the data content.

1333 JadeBytes not empty

Cause

This error is raised by methods of the [JadeBytes](#) class that require that the receiver has no data content; that is, the **JadeBytes** object is empty. For example, it is not valid to call the [allocate](#) method on a **JadeBytes** object that has been allocated or loaded with data.

Action

If you need to change the allocation size of a **JadeBytes** object, use the [grow](#) or the [truncate](#) method.

1334 Dictliterator keys too long to support resynch

Cause

When a dictionary iteration fails to load a key into the key block of the iterator because it does not fit, the iteration is allowed to proceed but operations that would normally require access to iterator keys fail when the keys are not loaded.

This error occurs when the [Iterator](#) class [next](#) method or [back](#) method attempts to access a key that is not loaded.

Action

This is an internal error and should not normally occur. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1335 Dictliterator has no current key

Cause

When a dictionary iteration fails to load a key into the key block of the iterator because it does not fit, the iteration is allowed to proceed but operations that would normally require access to iterator keys fail when the keys are not loaded.

This error occurs when the [Dictionary](#) class [getliteratorKeys](#) method or the [literator](#) class [getCurrentKey](#) method attempts to access a key that is not loaded.

Action

This is an internal error and should not normally occur. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1336 Cannot change singleFile or unaudited attribute after instantiation

Cause

This error occurs if you attempt to change the value of the [singleFile](#) attribute of a [JadeBytes](#) object after the object is instantiated by assigning content or allocating storage; that is, after it is created and committed in a transaction, even if it is empty (no content assigned).

Action

Change the value of the [singleFile](#) attribute before you assign content or allocate storage.

1337 Collection block size cannot be set to less than 4 entries per block

Cause

This error occurs if you attempt to set or change the block size value of a collection (for example, by calling the `Collection::setBlockSize` method) to a value less than 4 entries per block.

Action

Avoid this illegal operation.

1338 Collection block structure error

Cause

This error occurs when a collection method accesses a collection block with an invalid structure. The instance of [SystemException](#) raised contains information to identify the specific collection and collection block affected.

Action

Certify the data using the JADE Logical Certifier utility and use the *fix-up* script that is generated, to repair the data. If your JADE licenses include support, report the error to your local JADE support center or JADE Support, providing appropriate error logs.

1342 JadeBytes maximum content size exceeded

Cause

This error occurs if an attempt has been made to create or update a **JadeBytes** instance to a length greater than approximately 1,019G bytes.

In addition, the **JadeBytes** class **allocate** method raises this error if its **length** parameter is less than zero (0).

Action

Change your application to prevent this.

1347 Collection history stack overflow

Cause

This error is raised if a collection operation exceeded the size of the internal operation history stack.

Action

If your JADE licenses include support, report the error to your local JADE support center or JADE Support, providing appropriate error logs.

1348 Cannot update a ReadOnly JadeBytes instance

Cause

This error occurs when an attempt is made to update a dedicated **JadeBytes** instance which has its **readOnly** property set **true**. This is not allowed.

Action

Correct your logic.

1349 JadeBytes unaudited can only be set true when singleFile is true

Cause

This error occurs when an attempt was made to set the **unaudited** property of a **JadeBytes** instance that has the **singleFile** property set to **false**. This is not allowed.

Action

Correct your logic.

1350 JadeBytes readOnly can only be set true when singleFile is true

Cause

This error occurs when an attempt is made to set the `readOnly` property of a `JadeBytes` instance that has the `singleFile` property set to `false`. This is not allowed.

Action

Correct your logic.

1351 JadeBytes feature only supported on instances with singleFile true

Cause

This error occurs when an attempt is made to invoke the `loadFromFileDirect` method or the `extractToFileDirect` method of the `JadeBytes` class and the `singleFile` property is set to `false`.

Action

Change your logic to test the value of the `singleFile` property before invoking the method.

1352 Dynamic dictionary key property definition has been changed

Cause

This error occurs when the following sequence of actions occurs.

1. A property that is used as a member key of an instance of `DynaDictionary` or as a subclass is changed and the database is reorganized.
2. A persistent instance of the `DynaDictionary` class or subclass is accessed.

Note Unlike persistent instances of `MemberKeyDictionary` subclasses, persistent instances of `DynaDictionary` subclasses are *not* reorganized when key properties change.

Action

Call the `rebuild` method defined on the `Collection` class for all persistent instances of `DynaDictionary` subclasses, as shown in the following example.

```
vars
  subs: ClassColl;
  cls: Class;
  obj: Object;
begin
  create subs transient;
  DynaDictionary.withAllSubclasses(subs);
  beginTransaction;
```

```
        foreach cls in subs do
            foreach obj in cls.instances do
                obj.DynaDictionary.rebuild();
            endforeach;
        endforeach;
        commitTransaction;
    epilogs
        delete subs;
    end;
```

1400 through 1499 Miscellaneous Runtime Errors

This section describes the general errors that can be returned at run time.

1400 The receiver has already registered a timer with the specified tag

Cause

This error occurs when the object invoking the `beginTimer` method has already registered a timer with this tag.

Action

Use unique tags when registering multiple timers on the same object. The method and source position containing the `beginTimer` call at which the exception was raised are reported in the error description.

1401 The operating system failed to start the timer

Cause

This error occurs if the operating system cannot start the timer.

Action

Refer to the JADE messages log file (`jommsg.log`) for a description of the error reported by the operating system.

1402 Date is not valid

Cause

This error is raised by methods of the `Date` primitive type when the receiver is not a valid date.

Note The `Date` primitive type `isValid` method returns whether a date is a valid Gregorian date (that is, it is within the valid internal representation range of 24th November -4713 through 31st December 1465072 Gregorian).

Gregorian dates can be invalid in non-Gregorian calendars; for example, some Arabic calendars do not accept dates before 1318. This error can occur when testing a non-Gregorian date for validity in that calendar or when testing that the equivalent Gregorian date is valid.

Action

Determine the cause of the invalid date and correct the code that led to the error. The method and source position containing the `Date` method call at which the exception was raised are reported in the error description.

1403 Operation is not valid because Date is null

Cause

This error occurs if a primitive method was invoked on a `null` date value.

Action

Determine the cause of the invalid **null** date and correct the code that led to the error. The method and source position containing the **Date** type method call at which the exception was raised are reported in the error description.

1404 hours, minutes, and seconds parameters are not valid

Cause

This error occurs if the arguments supplied to the **setTime** method of the **Time** primitive type do not represent a valid time value.

Action

Determine the cause of the invalid time arguments and correct the code that led to the error. The method and source position containing the **setTime** call at which the exception was raised are reported in the error description.

1405 Failed to obtain the JADE installation directory

Cause

This error occurs if JADE fails to obtain a path to the directory in which it is installed. For some operations, JADE must obtain a path from the operating system to the directory in which it is installed.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1406 Result of expression overflows Integer precision

Cause

This error occurs when the result of an **Integer** primitive type expression is outside the allowable range. This error should not normally occur as the minimum and maximum values of the **Integer** primitive type in JADE are - 2,147,483,648 and 2,147,483,647, respectively.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1407 Invalid argument passed to method

Cause

This error occurs if the argument supplied to a primitive method is invalid. The following methods can return this error if they receive an invalid argument:

- `arcos`
- `arcsin`
- `arctan`
- `asUuid`
- `cos`
- `exp`
- `log`
- `log10`
- `sin`
- `sqrt`
- `tan`

For example, as the negative argument in `(-1.0).sqrt` is invalid, this error is raised.

Action

You can find an explanation of the error in the `errorItem` property of the exception. This explanation identifies the method in error and the reason for the exception. The reason can be one of the following.

- The argument was not in the domain of the function; for example, `log(-1)`.
- The argument would result in a singularity.
- The argument would produce a function result greater than **Max_Real**; for example, `exp(1000)`.
- The argument would produce a function result less than **Min_Real**; for example, `exp(-1000)`.
- The argument would produce a function result with total loss of significant digits; for example, `sin(10e70)`.
- The argument was not formatted as a valid UUID representation.

1408 Cannot create a persistent instance of this class

Cause

This error occurs if you attempt to create a persistent instance of a class that is not intended to be persistent. Classes such as `File` class and `Iterator` class cannot be persistent, so you cannot create persistent instances of them.

Action

Ensure that you do not attempt to create persistent instances of classes that are not persistent.

1409 Compress operation failed

Cause

This error occurs if the compress operation on a string or binary value fails.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1410 Uncompress operation failed

Cause

This error occurs if the uncompress operation on a string or binary value fails.

Action

Check that the value being uncompressed was compressed correctly.

1411 Attempt to add unknown system file

Cause

This error occurs if the system schema maintenance function attempts to add a new unknown system file.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1413 Index used in string operation is out of bounds

Cause

This error occurs when the starting position in a string operation is less than 1 or the starting position is greater than the string size when the string expression is being read (not being assigned to).

Action

Determine the cause for the invalid argument in the string expression and then correct the code that led to the error. The method and source position containing the string expression at which the exception was raised are reported in the error description.

1414 Method called with an incorrect number of parameters

Cause

This error occurs when a method is invoked (called) and the number of parameters passed by the calling method does not match the signature of the method defined in the schema.

Action

Locate the method call that is at fault in your method, and change it so that it passes the correct number of parameters defined in the method signature in your schema.

1415 An internal JADE software exception occurred

Cause

This error occurs if an internal JADE software error has occurred. It is usually a sign of a fatal internal error.

Action

This is an internal error. Consult the JADE messages log file (**jommsg.log**) and then contact your local JADE support center or JADE Support if your JADE licenses include support.

1416 Unicode to Ansi conversion failed

Cause

This error occurs during the conversion of data from Unicode to ANSI format. This error occurs when converting a Unicode string into an ANSI string.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1417 Ansi to Unicode conversion failed

Cause

This error occurs during the conversion of data from ANSI to Unicode format. This error occurs when converting an ANSI string into a Unicode string.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1418 Unicode UTF8 file contains illegal byte sequence

Cause

This error occurs whilst decoding a binary that encodes a UTF8 string that is found to include an illegal byte sequence.

Action

Check that the binary is actually an encoded UTF8 string.

1419 The application could not be started due to a lock error

Cause

This error occurs if an application could not be started due to a lock error; for example, an environmental object such as [node.processes](#) or a dictionary is locked, or there is an attempt to start too many applications simultaneously.

Action

Try again, or check your application code to make sure that no environmental objects or dictionaries are locked.

1420 Base64 Decoding failed

Cause

This error occurs if the **base64Decode** method of the **String** class fails because the string was not a valid Base64-encoded value.

Action

Check the contents of the string.

1422 Result of expression overflows Decimal precision

Cause

This error occurs if you have not specified enough decimal places to accept the result of an expression.

Action

Change your application code to specify more decimal places. For details, see the [Decimal](#) primitive type in Chapter 1 of the JADE *Encyclopaedia of Primitive Types*.

The method and source position containing the assignment at which the exception was raised are reported in the error description.

1423 Dynamic Object property index out of range

Cause

This error occurs when a property index value passed to a method of the [JadeDynamicObject](#) class exceeds the number of properties currently defined for the object.

Action

Change your application code to ensure that the index is within the range of one (1) through the value of the [propertyCount](#) attribute.

1424 Dynamic Object not correctly initialised

Cause

This error occurs when a property index value that is passed to a method of the [JadeDynamicObject](#) class exceeds the number of properties currently defined for the object.

Action

Change your application code to ensure that the index is within the range of one (1) through the value of the [propertyCount](#) attribute.

1425 Dynamic Object property name too long

Cause

This error occurs when attempting to add a property that has a name with a length greater than 100 characters to a [JadeDynamicObject](#) class.

Action

Reduce the length of the name of the property you are attempting to add.

1426 Dynamic Object property type mismatch

Cause

This error occurs when the type of the value that is passed to a [setProperty](#) operation on a dynamic object does not match the defined type for the specified property.

This error occurs if a dynamic object that is populated and returned by a specified method is passed to a different method that attempts to reuse the predefined properties on the object, but the property types are inconsistent.

Action

Avoid passing dynamic objects with predefined properties to methods that expect and create a dynamic object with different properties.

1427 Dynamic Object property name mismatch

Cause

This error can be returned when a dynamic object populated and returned by a specified method is passed to a different method that attempts to reuse the predefined properties on the object but the property names are inconsistent.

Action

Avoid passing dynamic objects with predefined properties to methods that expect and create a dynamic object with different properties.

1429 Cannot access a JadeAuditAccess object created on a different Node

Cause

This error occurs if a [JadeAuditAccess](#) class instance is used on a node other than the node on which the instance was created.

Action

Change your application code so that the **JadeAuditAccess** instance is created and used on a single node.

1431 Invalid class number map file format

Cause

This error occurs when the class number map file format is not correct. The [extendedErrorText](#) property of the error provides details of the nature of the error, including the file name and line number.

Action

Correct the format of the input file.

1433 Dynamic Object property name conflict

Cause

This error is raised by the [addProperty](#) method of the [JadeDynamicObject](#) class when the value of the name parameter conflicts with an existing fixed or dynamic property name.

Action

Change the value of the name parameter.

1434 Invalid code point when converting Unicode character

Cause

This error occurs when processing Unicode text and a character was discovered that is not valid; for example, half of a Unicode surrogate pair was found.

Action

Review any JADE strings for possible bad Unicode characters or sequences, and then alter data as appropriate to the application.

1435 Requested timer has expired/timed out

Cause

This error occurs when a C API-level call waiting on a timer has returned due to the time interval expiring.

Action

The user logic should proceed based on the knowledge that the timer has expired.

1436 Required log file is not currently open

Cause

This error occurs if JADE attempts to access a JADE log file that is not currently open. The most likely reason that the file cannot be opened is that a valid file name has not been specified.

Action

Check your application code to ensure that the JADE log file name specified in [fileName](#) property is correct.

1437 UNICODE codepoints greater than U+FFFF not supported

Cause

This error occurs if you attempt to convert a UTF8 string that contains a code point greater than U+FFFF to a native character. (The receiving field is not large enough to store that 2-character value.)

This error also occurs if such a string is attempted to be converted to ANSI. (There is no ANSI representation for Unicode surrogate characters.)

Action

Replace the characters causing the problem in the source UTF8 string value.

1438 UTF8 character to Native character conversion failed

Cause

This error occurs if you attempt to convert a UTF8 character to a native JADE [Character](#) type and the UTF8 character cannot be represented as a single native character.

Action

Replace the character causing the problem in the source UTF8 string value.

1439 Native string to UTF8 string conversion failed

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

1440 Result of expression overflows Integer64 precision

Cause

This error occurs when the result of an **Integer64** primitive type expression is outside the allowed range.

This error should not normally occur, as the minimum and maximum values of the **Integer64** primitive type in JADE are **-9,223,372,036,854,775,808** and **9,223,372,036,854,775,807**, respectively.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1441 Result of expression overflows Byte precision

Cause

This error occurs when the result of a **Byte** primitive type expression is outside the allowed range, which is zero (0) through 255.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1442 Obsolete - StringUtf8 values containing surrogate characters are not supported

Cause

This error occurs if a source **StringUtf8** value contains code points greater than U+FFFF. Primitive methods cannot handle a **StringUtf8** character, which cannot be represented in native characters.

Action

Replace the characters causing the problem in the source UTF8 string value.

1443 MemoryAddress is not valid for current Node

Cause

This error occurs when a **MemoryAddress** value is used on a node other than the node where the underlying address is valid. The underlying memory address has no meaning on nodes other than the node where it was assigned a non-null value.

Action

Change your code to avoid this error. Use only a **MemoryAddress** value on the node where it was assigned a non-null value.

1444 Invalid Unicode UTF-16 surrogate sequence

Cause

This error occurs when converting UTF16 data to a different encoding scheme (UTF8, ANSI + locale, UTF32, and so on) and the source data contained an invalid or incomplete UTF16 surrogate pair.

Action

Review the data and then fix or remove the bad surrogate pairs.

1445 Supplied codepage, or codepage derived from LCID is unsupported on this system

Cause

This error occurs when changing the encoding of user data that requires the use of code pages (for example, 936 Simplified Chinese) and the code page that was requested is not currently supported by the system.

Action

Install the necessary operating system code page support for the code page that caused the error, and then re-run the operation.

1446 Result of expression underflowed Integer precision

Cause

This error occurs when the result of an **Integer** primitive type expression is less than the minimum value allowed. This error should not normally occur, as the minimum and maximum values of the **Integer** primitive type in JADE are -2,147,483,648 and 2,147,483,647, respectively.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1447 Result of expression underflowed Integer64 precision

Cause

This error occurs when the result of an **Integer64** primitive type expression is less than the minimum value allowed. This error should not normally occur, as the minimum and maximum values of the **Integer64** primitive type in JADE are -9,223,372,036,854,775,808 and 9,223,372,036,854,775,807, respectively.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

1448 A memory allocation request failed

Cause

This error occurs when the current action could not be completed because a memory allocation failed.

Action

Make more memory available or change the application to work with smaller chunks of data.

1450 An incomplete package can not be opened

Cause

This error occurs if an application is started from a schema which contains a package that is not complete (that is, the package has a circular dependency with another package in the schema).

For details about circular packages, see "[Specifying Advanced Load Options](#)", in Chapter 10 of the *JADE Development Environment User's Guide*, and "[Specifying Additional Load Options](#)", in the *Jade Schema Load User's Guide*.

[Specifying Additional Load Options](#)", in Chapter 1 of the *JADE Schema Load Utility User's Guide*.

Action

Load the exporting schema to ensure that the exporting package is complete, and then reload the importing schema file.

1451 Delta identifier does not exist

Cause

This error occurs:

- During a load using **jadloadb** if a delta is specified using the **delta=delta-name** parameter.
- During a load using the JADE Schema Load utility (**jadload**) and delta specified in the **Delta Name** text box does not exist.
- If an application is started with **jadclient**, **jadloadb**, or **jade** using the **delta=delta-name** parameter when the specified delta does not exist.

For details, see "delta", in Chapter 1 of the *JADE Schema Load User's Guide*.

Action

If the error occurs during a schema load, set the **createDeltaIfMissing** parameter to **true** or create the delta before reloading the schema.

If the error occurs when an application is started, create the delta before restarting the application.

1452 String printf format text missing or invalid

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

1453 String printf text exceeds maxLength

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

1454 The SystemSequenceNumber name is invalid

Cause

This error occurs if a system sequence number name passed to the [createSystemSequenceNumber](#) method or the [getSystemSequenceNumberNext](#) method of the [System](#) class:

- Is an empty string
- Contains embedded null characters
- Is longer than 60 characters

Action

Correct your code.

1455 The SystemSequenceNumber initial value cannot be negative

Cause

This error occurs if the initial value passed to the [createSystemSequenceNumber](#) method of the [System](#) class is not zero (0) or greater.

Action

Correct your code.

1456 The SystemSequenceNumber has reached the maximum value (Max_Integer64)

Cause

This error occurs if the maximum value for the specified system sequence number has already been returned by the [getSystemSequenceNumberNext](#) method of the [System](#) class.

Action

None.

1457 Error populating Formatter from locale info. See jommsg.log

Cause

This error occurs when one or more expected locale attributes are not available. See **jommsg.log** for details. Locale attributes are obtained from the operating system runtime.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1458 Formatter has not been populated

Cause

This is an internal error. An attempt was made to use a DTNC parser without initializing it.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1459 LCID parameter value is not recognized

Cause

This error occurs if a primitive method that requires a locale identifier (LCID) parameter is passed a value that is not recognized.

Action

Correct your code.

1460 Invalid format string passed to Date/Time formatter. See jommsg.log

Cause

This error occurs if an invalid picture string was passed to a date or time formatter or parser. See **jommsg.log** for details.

Action

Correct your code.

1461 LCID parameter value is not an installed locale

Cause

This error occurs if a system is running with enhanced locale support and a presentation client has a Locale ID (LCID) or language that is not installed on the application server.

Action

The locale must be installed on both the application server and the presentation client when running with enhanced locale support.

1463 LCID parameter value is not an installed locale on Thin Client machine

Cause

This error occurs if the **Application** class **setJadeLocale** method is called with an LCID value which is not recognized by the thin client machine operating system.

Action

Use a different locale or install the appropriate language pack on the thin client machine.

1464 SetThreadLocale failed at Thin Client machine

Cause

This error occurs if the **Application** class **setJadeLocale** method is called with an LCID value which is not recognized by the thin client machine operating system.

Action

Use a different locale or install the appropriate language pack on the thin client machine.

1465 Incompatible Dynamic object format, eg. Ansi to Unicode

Cause

This error occurs if the client node and server node are using different Dynamic Object formats. See **jommsg.log** for details.

Action

Check that the client and server nodes are using compatible binaries. If your JADE licenses include support, contact your local JADE support center or JADE Support.

1466 Unable to run a unit test

Cause

This error occurs if the unit test framework is unable to run a unit test. If you encounter this error, it is an indication that an incorrect parameter was specified for the unit test framework (for example, an invalid schema name or test name).

Action

Check the parameters used for the unit test and correct any that are invalid.

1467 Unit tests failed

Cause

This error occurs if one or more unit tests fail when you are batch-running unit tests.

Action

Refer to the JADE unit test log file (**JadeUnitTest.log**) for details about the failed test or tests.

1469 Time zone not found

Cause

This error occurs when you attempt to create a time zone and the relevant time zone is not found in the Windows registry of the database server.

Action

For a single user or a database server call, check that the specified time zone is available in the Windows registry (see **JadeTimeZone** class **create...** type method constructors).

For a thin client call, check that the time zone name of the thin client (which you can find with **JadeTimeZone** class **getTimeZoneNameByLocation** method) is present in the Windows registry of the database server.

Applies to Version: 2020.0.01 and higher

1470 Retrieved value type is incompatible with as clause type

Cause

This error occurs when a type retrieved from a **foreach** iteration is not compatible with the **as** clause type specified in the **foreach** instruction. For example, if **Automobile** is specified as a **foreach as** clause type, if the object retrieved from the iteration is not of type **Automobile** (or a sub-type of **Automobile**), this error occurs.

Action

Determine why an object reference of the wrong type is being used in this context, and correct the code that led to the error. The method and source position containing the expression at which the exception was raised are reported in the error description.

For more details about **foreach** instruction **as** clause statements, see ["Using an as Clause in foreach Instructions"](#) in Chapter 1 of the *JADE Developer's Reference*.

Applies to Version: 2020.0.01 and higher

1471 Collection locking is incompatible with prior updates

Cause

This error occurs when either a deferred collection method has been called on a persistent collection that is already locked for update or an updating collection method attempts to lock a persistent collection that has pending deferred operations.

Action

Change your application code so that deferred collection operations and updating collection method are not called in the same transaction.

Example

```
collection.add(anObject);  
collection.tryAddDeferred(anotherObject); // exception 1471 will be raised
```

Applies to Version: 2020.0.01 and higher

1600 through 1699 Remote Procedure Call Request Errors

The errors of the range 1600 through 1699 are associated with call requests to remote procedures.

These represent internal errors. If you receive any errors in this range, contact your local JADE support center or JADE Support if your JADE licenses include support.

1700 through 1750 Security Errors

This section describes errors that can be returned by the JADE security system.

1700 Connection refused due to security violation

Cause

This error occurs if your client authentication checks have failed. As the authentication support feature is in place, your client connections must pass the authorization checks that are in place.

Action

Obtain the required authorization from your security administrator.

1701 User id or password is invalid

Cause

This error occurs if your user identification (user id) or password fails the security check. As the user validation support feature is in place, every user must have a valid user id and password in order to access the system.

Action

Check that you have correctly entered your user id and password. Alternatively, obtain a valid user id and password from your security administrator.

1702 Encryption enabled, but non encrypted RPC message received

Cause

This error occurs if your node expects encrypted messages but receives a message that is not encrypted. As the network message encryption feature is in place, incoming request or response Remote Procedure Call (RPC) messages must be encrypted. The RPC protocols always check the encrypted message flag in the header against the enabled or disabled state of the encryption feature.

Action

Report this error to your security administrator.

1703 Encryption not enabled, but encrypted RPC message received

Cause

This error occurs if you receive an encrypted message but your node is not set up to handle encrypted messages. As the network message encryption routine is not in place, incoming request or response Remote Procedure Call (RPC) messages that are encrypted are rejected.

Action

Report this error to your security administrator.

1704 Cyclic redundancy check on encrypted message failed

Cause

This error occurs if a message fails the Cyclic Redundancy Check (CRC).

After the site decryption routine is called, a CRC is generated against the data part of the message and compared to the CRC in the header. A CRC mismatch results in the message being rejected.

As part of the network message encryption feature, the CRC provides protection against loss of data if something goes wrong in the encryption or decryption routines; for example, different or incompatible algorithms being used at each end. If this check fails, it would normally mean that the original message has been corrupted following encryption and subsequent decryption.

Action

Report this error to your security administrator.

1705 Authentication enabled, but failed to initialize

Cause

This error occurs if you have the authentication feature in place but it fails to initialize. This could be due to the failure of the dynamic load of the authentication library or to the failure of any of the entry points.

If the dynamic load fails on a remote access capable node, an exception is logged in the JADE messages log file (**jommmsg.log**) and remote access provider support is disabled, preventing any clients from connecting.

Action

Report this error to your security administrator.

1706 Encryption enabled, but failed to initialize

Cause

This error occurs if you have the network message encryption feature in place but it fails to initialize. This could be due to the failure of the dynamic load of the library or to the failure of any of the entry points.

If the dynamic load fails on a remote access capable node, an exception is logged in the JADE messages log file (**jommsg.log**) and remote access provider support is disabled, preventing any clients from connecting.

Action

Report this to your security administrator.

1707 rpcEncryptMessage failed

Cause

This error occurs if your encryption function fails.

Action

Check and correct your encryption function accordingly.

1708 rpcDecryptMessage failed

Cause

This error occurs if your decryption function fails.

Action

Check and correct your decryption function accordingly.

1711 Development Security library not found

Cause

This error occurs if the development security library specified in the **DevelopmentSecurityLibrary** parameter in the [**JadeSecurity**] section of your JADE initialization file could not be located.

Action

Place the development security library in the correct place in the **DevelopmentSecurityLibrary** parameter in the [**JadeSecurity**] section of your JADE initialization file, or remove the initialization file setting.

1712 Development Security missing entry points

Cause

This error occurs if the development security library specified in the **DevelopmentSecurityLibrary** parameter in the [**JadeSecurity**] section of your JADE initialization file does not have the required entry points.

Action

Use a development security library that has the required entry points, or remove the JADE initialization file setting. (For details about JADE development environment security, see [Chapter 2](#) of the *JADE Object Manager Guide*.)

1713 Development Security failed

Cause

This error indicates a security problem.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1800 through 1899 Locale Management and Formatting Errors

This section describes errors that are related to the parsing and formatting of date, time, number, and currency information for a specified locale.

1800 The text is valid but incomplete

Cause

This error occurs when text passed to the date, time, number, and currency parsing methods of the primitive types matches the earlier part of the expected format but more text is required to complete the value; for example, if the date parser is passed text containing *dd/MM* and the date parser is expecting *dd/MM/yy*.

Action

Change your application to obtain the required information. For example, prompt the user to enter all the required information.

1801 The magnitude of the value exceeds the maximum value of the datatype

Cause

This error occurs if the numeric value of the text passed as the **source** parameter to the number and currency parsing methods of the primitive types is larger than the maximum value supported by the receiving primitive type. For example, if the **source** parameter passed to the [parseNumberWithFmtAndLcid](#) method is **"12345678901"**, which exceeds the maximum allowed value for an [Integer](#).

Action

Change your application to obtain the required information. For example, prompt the user to enter an allowed value.

1802 Source character does not match the format rules

Cause

This error occurs when a character in the text passed to the date, time, number, and currency parsing methods of the primitive types does not match the formatting rules; for example, passing **\$123** to the [parseNumberWithFmtAndLcid](#) method of the [Integer](#) primitive type or passing **23-12-2001** to the [parseShortWithCurrentLocale](#) method of the [Date](#) primitive type if the current locale is New Zealand, where the separator character is a slash (/).

Action

The source character causing the error is identified by the **errOffset** parameter of the parsing method. Change your logic to handle this invalid input.

1803 Source character is not a recognised digit

Cause

This error occurs when the date, time, number, and currency parsing methods of the primitive types find a character that is not a digit at a position where a digit is expected; for example, if the **source** parameter passed to the [parseShortWithPicAndLcid](#) method of the [Date](#) primitive type is **"25/Dec/2012"** and the **pic** parameter is **"dd/MMM/yyyy"** (that is, dates where the month is specified as a number).

Action

The position of the character causing the error is identified by the **errOffset** parameter of the parsing method. Change your logic to handle this invalid input.

1804 Too many significant digits before decimal point

Cause

This error occurs when the number and currency parsing methods of the primitive types find more significant digits before the decimal point than are allowed by the format.

Leading zeros are not considered significant; for example, if the **source** parameter passed to the [parseNumberWithFmtAndLcid](#) method is **"12.3"** and receiver is defined as [Decimal](#)[2,1].

Action

Change your application logic to handle this invalid input.

1805 Too many significant digits after decimal point

Cause

This error occurs when the number and currency parsing methods of the primitive types find more significant digits after the decimal point than are allowed by the format.

Trailing zeros are not considered significant; for example, if the **source** parameter passed to the [parseNumberWithFmtAndLcid](#) method is **"1.23"** and receiver is defined as [Decimal](#)[2,1].

Action

Change your application logic to handle this invalid input.

1806 Source text is null or blank

Cause

This error occurs when the text passed to the date, time, number, and currency parsing methods of the primitive types is empty or only contains spaces; for example, if the **source** parameter passed to the [parseShortWithCurrentLocale](#) method of the [Date](#) primitive type is null.

Action

Change your application logic to handle this invalid input.

1807 Max fractional digits > max significant digits

Cause

This error occurs when the number format specifies more digits after the decimal point than are allowed in the complete number (before and after the point). This error should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1808 Expected sign

Cause

This error occurs when number and currency parsing methods of the primitive types expect to find a sign character in the text of the **source** parameter.

Action

The position where the sign character was expected is specified in the **errOffset** output parameter of the parsing method. Change your logic to handle this invalid input.

1809 Group separator must be followed by a digit

Cause

This error occurs when number and currency parsing methods of the primitive types expect to find a digit following the group separator character; for example, if the **source** parameter passed to the [parseNumberWithCurrentLocale](#) method of the [Decimal](#) primitive type is "123,." and the current locale expects the thousands receiver to be followed by a digit rather than the decimal point.

Action

Change your application logic to handle this invalid input.

1810 HighPrecisionNumber overflowed

Cause

This error occurs when the number and currency parsing methods of the primitive types find more than 30 significant digits preceding the decimal point; for example, if the **source** parameter passed to the [parseNumberWithCurrentLocale](#) method of the [Real](#) primitive type is "12345678901234567890123456789012345".

Action

Change your application logic to handle this invalid input.

1811 HighPrecisionNumber received non-decimal value

Cause

This error occurs when the number and currency parsing methods of the primitive types find a digit whose binary value is greater than nine.

Action

Change your application logic to handle this invalid input.

1812 Decimal point not allowed

Cause

This error occurs when the number and currency parsing methods of the primitive types find a decimal point character when the receiver type is of type [Integer](#); for example, if the **source** parameter passed to the [parseNumberWithCurrentLocale](#) method is "1.2", which is invalid for an [Integer](#).

Action

Change your application logic to handle this invalid input.

1813 Sign not allowed

Cause

This error occurs when the number and currency parsing methods of the primitive types find a negative sign character when the receiver type is of type [Byte](#); for example, if the **source** parameter passed to the [parseNumberWithCurrentLocale](#) method is "-1", which is an invalid value for a [Byte](#).

Action

Change your application logic to handle this invalid input.

1814 Too many significant digits in total

Cause

This error occurs when the number and currency parsing methods of the primitive types find too many significant digits before the decimal point. Leading zeros are not considered significant.

Action

Change your application logic to handle this invalid input.

1815 Text must contain at least one digit character

Cause

This error occurs when the number and currency parsing methods of the primitive types finds no digits preceded the sign or currency character when the format specifies that the sign or currency is trailing.

Action

Change your application logic to handle this invalid input.

1816 Currency symbol positioned incorrectly

Cause

This error occurs when the currency parsing methods of the primitive types find a currency character (for example, "\$") but it is not in the position specified by the format; for example, if the **source** parameter passed to the [parseCurrencyWithFmtAndLcid](#) method of the [Decimal](#) primitive type is "\$1" and the **fmt** parameter, which is of type [CurrencyFormat](#), specifies a trailing currency character.

Action

Change your application logic to handle this invalid input.

1817 Day number is out of range

Cause

This error occurs when the date parsing methods of the primitive types find a day number less than one or greater than the last day in the month; for example, if the **source** parameter passed to the [parseLongWithCurrentLocale](#) method of the [Date](#) primitive type is "29 February 2011" (2011 is not a leap year).

Action

Change your application logic to handle this invalid input.

1818 Month number is out of range

Cause

This error occurs when the date parsing methods of the primitive types find a month number less than one or greater than the last month in the year, which is 12 (or 13 if the locale has 13 months); for example, if the **source** parameter passed to the [parseShortWithCurrentLocale](#) method of the [Date](#) primitive type is "25/99/2012" (no year has 99 months).

Action

Change your application logic to handle this invalid input.

1819 Group separator must be preceded by at least one digit

Cause

This error occurs when the number and currency parsing methods of the primitive types find a group separator character before the first digit; for example, if the **source** parameter passed to the [parseNumberWithCurrentLocale](#) method of the [Real](#) primitive type is ",123,456".

Action

Change your application logic to handle this invalid input.

1820 Source text has mixed ASCII and native digits

Cause

This error occurs when the date, time, number, and currency parsing methods of the primitive types find a number that has a mixture of ASCII and native digits.

Action

Change your application logic to handle this invalid input.

1822 Text limited to ASCII digits

Cause

This error occurs when the date, time, number, and currency parsing methods of the primitive types find a native digit when they have been disallowed (probably by using a control panel locale setting).

Action

Change your application logic to handle this invalid input.

1823 Text missing balancing closing parenthesis

Cause

This error occurs when the currency parsing methods of the primitive types find a leading opening parenthesis but fail to find a trailing parenthesis if the format specifies that negative currency values are to be displayed in parentheses; for example, if the **source** parameter passed to the [parseCurrencyWithFmtAndLcid](#) method of the [Decimal](#) primitive type is "(12.34".

Action

Change your application logic to handle this invalid input.

1824 Sign positioned incorrectly

Cause

This error occurs when the number and currency parsing methods of the primitive types find a negative sign character but not in the position specified by the format; for example, if the **source** parameter passed to the [parseNumberWithFmtAndLcid](#) method of the [Decimal](#) primitive type is "-1" and the **fmt** parameter, which is of type [NumberFormat](#), specifies a trailing sign.

Action

Change your application logic to handle this invalid input.

1825 Text has unbalanced closing parenthesis

Cause

This error occurs when the currency parsing methods of the primitive types find a trailing parenthesis but fail to find the corresponding opening parenthesis if the format specifies that negative currency values are to be displayed in parentheses; for example, if the **source** parameter passed to the [parseCurrencyWithFmtAndLcid](#) method of the [Decimal](#) primitive type is "12.34)".

Action

Change your application logic to handle this invalid input.

1827 Secondary day number does not match primary

Cause

This error occurs when the date parsing methods of the primitive types use a date format picture that contains more than one day number element with different values; for example, if the **pic** parameter passed to the [parseLongWithPicAndLcid](#) method of the [Date](#) primitive type is "dd dd MMMM yyyy" and the **source** parameter is "24 25 December 2012" (the day cannot simultaneously be the 25th and the 24th).

Action

Change your application logic to handle this invalid input.

1828 Secondary month does not match primary

Cause

This error occurs when the date parsing methods of the primitive types use a date format picture that contains more than one month element with different values; for example, if the **pic** parameter passed to the [parseLongWithPicAndLcid](#) method of the [Date](#) primitive type is "dd MMMM MMMM yyyy" and the **source** parameter is "25 December November 2012" (the month cannot simultaneously be December and November).

Action

Change your application logic to handle this invalid input.

1829 Year number is out of range

Cause

This error occurs when the date parsing methods of the primitive types find a year number greater than 30000.

Action

Change your application logic to handle this invalid input.

1830 Secondary year does not match primary

Cause

This error occurs when the date parsing methods of the primitive types use a date format picture that contains more than one year number element with different values; for example, if the **pic** parameter passed to the [parseLongWithPicAndLcid](#) method of the [Date](#) primitive type is "**dd MMMM yyyy yyyy**" and the **source** parameter is "**25 December 2011 2012**" (the year cannot simultaneously be 2011 and 2012).

Action

Change your application logic to handle this invalid input.

1832 The day, month, year combination is invalid

Cause

This error occurs when the date parsing methods of the primitive types encounter a Gregorian day number, month, and year combination that is invalid; for example, if the **source** parameter passed to the [parseLongWithCurrentLocale](#) method of the [Date](#) primitive type is "**31 February 2012**" (February can never have a day number greater than 29).

This error can also occur when parsing a full date that includes the day name, but the day name does not match the date.

Note The [Date](#) primitive type [isValid](#) method returns whether a date is a valid Gregorian date (that is, it is within the valid internal representation range of 24th November -4713 through 31st December 1465072 Gregorian).

Action

Change your application logic to handle this invalid input.

1833 The weekday name does not match the date

Cause

This error occurs when the date parsing methods of the primitive types are passed a day name element that does not match the rest of the date; for example, if the **source** parameter passed to the [parseLongWithCurrentLocale](#) method of the [Date](#) primitive type is "**Monday, 25 December 2012**" (Christmas in 2012 is on a Tuesday).

Action

Change your application logic to handle this invalid input.

1834 Secondary day name does not match primary

Cause

This error occurs when the date parsing methods of the primitive types use a date format picture that contains more than one day name element with different values.

Action

Change your application logic to handle this invalid input.

1835 Not a recognised day name

Cause

This error occurs when the date parsing methods of the primitive types are passed a day name element that is not recognized; for example, if the **source** parameter passed to the [parseLongWithCurrentLocale](#) method of the [Date](#) primitive type is "**Chooseday, 25 December 2012**" (**Chooseday** is not a day of the week).

Action

Change your application logic to handle this invalid input.

1836 Hour number is out of range

Cause

This error occurs when the time parsing methods of the primitive types are passed an hour element value that is greater than 12 (or greater than 24 for a 24-hour clock value); for example, if the **source** parameter passed to the [parseWithPicAndLcid](#) method of the [Time](#) primitive type is "**13:00:00**" and the **pic** parameter is "**hh:mm**".

Action

Change your application logic to handle this invalid input.

1837 Minute number is out of range

Cause

This error occurs when the time parsing methods of the primitive types are passed a minute element value that is greater than 59; for example, if the **source** parameter passed to the [parseWithCurrentLocale](#) method of the [Time](#) primitive type is "**12:60:00**".

Action

Change your application logic to handle this invalid input.

1838 Second number is out of range

Cause

This error occurs when the time parsing methods of the primitive types are passed a second element value that is greater than 59; for example, if the **source** parameter passed to the [parseWithCurrentLocale](#) method of the [Time](#) primitive type is "**12:00:60**".

Action

Change your application logic to handle this invalid input.

1839 Unrecognised AM/PM designator

Cause

This error occurs when the time parsing methods of the primitive types are passed an **AM/PM** designator; for example, if the **source** parameter passed to the [parseWithCurrentLocale](#) method of the [Time](#) primitive type is **"12:00:00 BC"**.

Action

Change your application logic to handle this invalid input.

1840 Fractional second is out of range

Cause

This error occurs when the time parsing methods of the primitive types are passed a milliseconds element value that is greater than 999; for example, if the **source** parameter passed to the [parseWithPicAndLcid](#) method of the [Time](#) primitive type is **"06:10:12.3456"** and the **pic** parameter is **"hh:mm:ss:fff"**.

Action

Change your application logic to handle this invalid input.

1841 Secondary hour does not match primary

Cause

This error occurs when the time parsing methods of the primitive types use a time format picture that contains more than one hour number element with different values; for example, if the **pic** and **source** parameters passed to the [parseWithPicAndLcid](#) method of the [Time](#) primitive type are **"hh:mm:ss hh"** and **"01:00:00 02"**, respectively (the hour value cannot simultaneously be **1** and **2**).

Action

Change your application logic to handle this invalid input.

1842 Secondary minute does not match primary

Cause

This error occurs when the time parsing methods of the primitive types use a time format picture that contains more than one minute number element with different values; for example, if the **pic** parameter passed to the [parseWithPicAndLcid](#) method of the [Time](#) primitive type is **"hh:mm:ss mm"** and the **source** parameter is **"00:01:00 02"** (the minutes value cannot simultaneously be **1** and **2**).

Action

Change your application logic to handle this invalid input.

1843 Secondary second does not match primary

Cause

This error occurs when the time parsing methods of the primitive types use a time format picture that contains more than one seconds number element with different values; for example, if the **pic** parameter passed to the [parseWithPicAndLcid](#) method of the [Time](#) primitive type is "**hh:mm:ss ss**" and the **source** parameter is "**00:00:01 02**" (the seconds value cannot simultaneously be **1** and **2**).

Action

Change your application logic to handle this invalid input.

1844 Secondary millisecond does not match primary

Cause

This error occurs when the time parsing methods of the primitive types use a time format picture that contains more than one milliseconds number element with different values.

Action

Change your application logic to handle this invalid input.

1845 AM/PM designator does not match hour number

Cause

This error occurs when the time parsing methods of the primitive types use a time format picture that finds a 24-hour clock hour element greater than 12 together with a PM designator or a 24-hour clock hour element greater than or equal to 12 together with a AM designator; for example, if the **pic** parameter passed to the [parseWithPicAndLcid](#) method of the [Time](#) primitive type is "**HH:mm tt**" and the **source** parameter is "**23:00 AM**".

Action

Change your application logic to handle this invalid input.

1846 Unrecognised era name

Cause

This error occurs when the date parsing methods of the primitive types find text for the era element that does not match the era name for the locale.

Action

Change your application logic to handle this invalid input.

1847 Secondary era name does not match primary

Cause

This error occurs when the date parsing methods of the primitive types use a date format picture that contains more than one era name element. All era name elements in the text must have the same value.

Action

Change your application logic to handle this invalid input.

1848 Day number text is missing

Cause

This error occurs when the date parsing methods of the primitive types are processing a day number element and find the next separator character; for example, if the **source** parameter passed to the [parseShortWithCurrentLocale](#) method of the [Date](#) primitive type is `"/12/2011"` (the day number is missing).

Action

Change your application logic to handle this invalid input.

1850 Month number text is missing

Cause

This error occurs when the date parsing methods of the primitive types are processing a month number element and find the next separator character; for example, if the **source** parameter passed to the [parseShortWithCurrentLocale](#) method of the [Date](#) primitive type is `"25//2011"` (the month number is missing).

Action

Change your application logic to handle this invalid input.

1851 Month name text is missing

Cause

This error occurs when the date parsing methods of the primitive types are processing a month number element and find the next separator character; for example, if the **source** parameter passed to the [parseLongWithCurrentLocale](#) method of the [Date](#) primitive type is `"25 2011"` (the month name is missing).

Action

Change your application logic to handle this invalid input.

1852 Year number text is missing

Cause

This error occurs when the date parsing methods of the primitive types are processing a year number and finds the next separator character; for example, if the **source** parameter passed to the [parseShortWithPicAndLcid](#) method of the [Date](#) primitive type is `"12/25"` and the **pic** parameter is `"yyyy/MM/dd"` (the year number is missing).

Action

Change your application logic to handle this invalid input.

1856 Unrecognised month name

Cause

This error occurs when the date parsing methods of the primitive types are expecting a month name but the text does not match any locale month name; for example, if the **source** parameter passed to the [parseShortWithPicAndLcid](#) method of the [Date](#) primitive type is `"12/25"` and the **pic** parameter is `"yyyy/MM/dd"` (the year number is missing).

Action

Change your application logic to handle this invalid input.

1857 Partially matched month name

Cause

This error occurs when the date parsing methods of the primitive types are processing a month name and the text begins with characters that match a locale month name, but it is not a complete match; for example, if the **source** parameter passed to the [parseLongWithCurrentLocale](#) method of the [Date](#) primitive type is `"25 De 2011"` (the month name **De** is an incomplete version of **December**).

Action

Change your application logic to handle this invalid input.

1858 Partially matched AM/PM designator

Cause

This error occurs when the time parsing methods of the primitive types are expecting an AM/PM designator and the text begins with characters that match a locale designator, but it is not a complete match.

Action

Change your application logic to handle this invalid input.

1859 Partially matched era name

Cause

This error occurs when the date parsing methods of the primitive types are expecting an era name and the text found begins with characters that match a locale era name but are not a complete match.

Action

Change your application logic to handle this invalid input.

1870 Error encountered converting Real to display text

Cause

This error occurs in the conversion of a **Real** variable value to a fixed point display format failed.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

1871 Error encountered converting Date to short display text

Cause

This error occurs when using the **userShortFormatAndLcid** and **userShortFormatPicAndLcid** of the **Date** primitive type.

Action

Check that the value of the receiver is not zero (**0**) or some other invalid value.

1872 Error encountered converting Date to long display text

Cause

This error occurs when using the **userLongFormatAndLcid** and **userLongFormatPicAndLcid** of the **Date** primitive type.

Action

Check that the value of the receiver is not zero (**0**) or some other invalid value.

3000 through 3199 Database Engine Errors

This section describes errors that can be returned by the database engine or the server node.

3001 Cannot open a file that is already open

Cause

This error is an indication of an internal fault in the database engine.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3002 Cannot use file - not open

Cause

This error occurs if an attempt is made by the JADE Object Manager to access a file that is not open. In normal circumstances, this error should not occur because database files are opened implicitly as they are required.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3003 Exceeded maximum allowed files for database

Cause

This error occurs when an attempt to add a new database file using the JADE development environment or a schema load exceeds the maximum number of files allowed by the current database session.

The maximum number of persistent database files that can be added without restarting the database server is determined by the **FileNumberHeadroom** parameter in the **[PersistentDb]** section.

The maximum number of processes (each of which requires a transient database file) that can be started in a node is determined by the **HighestFileNumber** parameter in the **[TransientDb]** section.

The maximum number of user database files that can be defined in a database is 65,484, with the default value being 2,048. The maximum number of processes that can be started in a node is 65,483, with the default value being 2,047.

Action

When the limit that results in this error occurring has been reached, the node must be restarted to allow further files to be defined. Increase the value of the **FileNumberHeadroom** parameter in the **[PersistentDb]** section or the **HighestFileNumber** parameter in the **[TransientDb]** section, if you envisage exceeding the default limits.

3004 Required DB file not included in RPS database

Cause

This is an internal error and should not normally occur. It arises when an attempt is made to open a database file that is not included in an RPS mapping.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3009 File encrypted and partition unspecified

Cause

This error occurs if the database cannot invoke the **Object** class **autoPartitionIndex** method because the buffer contains an encrypted object.

Action

The partition must be explicitly set in the created object using the **Object** class **setPartitionID** or **setPartitionIndex** method.

3010 Update version for illegal object type

Cause

This error occurs if your application code attempts to version a subobject. This operation is not supported.

Action

Change your application code to avoid this operation.

3011 Invalid database handle

Cause

This error occurs if the database handle passed from the JADE Object Manager is no longer current or it is invalid. This error is an indication of an internal fault in the protocol used to communicate with the database.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3012 Cannot update outside of transaction state

Cause

This error occurs if the JADE Object Manager attempts to create, update, or delete an object while the database is out of transaction state.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3013 An error was encountered by DB file free space management

Cause

This error occurs if the database encounters an error while maintaining the freespace index. Information about the error is recorded in the JADE messages log file (**jommsg.log**).

Action

Certify the affected database file to ensure that there are no serious errors. If there are no errors in the object records, a file compaction restores the integrity of the freespace index. For details about running the certify and compaction routines, see "[The JADE Database Utility](#)", in Chapter 1 of the *JADE Database Administration Guide*.

3014 Transaction journal timestamp mismatch

Cause

This error occurs if the timestamp in a transaction journal file does not match the expected timestamp found in the database control file. The JADE messages log file (**jommsg.log**) records the expected and found timestamps.

Action

Restore the correct journal file from backup.

3015 Database role change in progress

Cause

This error occurs when a Synchronized Database Service (SDS) administration method of the **JadeDatabaseAdmin** class is invoked and a database role change (or takeover operation) is in progress.

This error can also be returned if a user process running on a primary system attempts to initiate a persistent transaction while a role change is in progress.

Action

To avoid this error in an SDS administration application, arm an exception handler to deal with the possibility of this error being raised. Most query-related administration methods can be called while a database is changing roles, while most commands result in this error.

To avoid this error in user applications, shut down updating applications running the primary system before initiating the takeover operation or arm an exception handler to deal with the error. Note that updating applications running on a primary system need to revert to an inquiry-only mode when the role change is complete. Applications can subscribe to the **SDS_RoleChangeEvent** event notified on the system instance, in order to be informed of database role changes.

3016 An object with the same OID already exists in the database

Cause

This error occurs if an attempt is made to create an object with an object identifier (OID) that has already been assigned to an existing object in the database. This error is an indication of an internal fault in the JADE Object Manager protocol that maintains the unique object identifier mapping.

Action

Investigate the issue further by running the JADE Logical Certifier utility.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3017 Operation not allowed while database transactions are in progress

Cause

This error occurs if an operation (for example, a database reorganization) is requested while database transactions are still in progress.

Action

The application should detect this and handle the situation appropriately; for example, retry the request after waiting a short time, and if still not successful, give the user an option to wait or to cancel the request.

3018 A database memory allocation request failed

Cause

This error occurs if a memory allocation request fails.

Action

Shut down one or more applications to regain memory and then retry the request. If this problem persists, it may be necessary to add additional physical memory to the database server or to reduce the object cache size limits.

For details, see the **TransientCacheSizeLimit** parameter in the [\[JadeServer\]](#) or [\[JadeClient\]](#) section of the JADE initialization file, or the **ObjectCacheSizeLimit** parameter in the [\[PersistentDb\]](#) section. See also the [JADE Initialization File Reference](#).

3019 Database control file directory not found

Cause

This error occurs if the database directory supplied on a database open is invalid. This normally means that the command line **path** parameter used to initiate the client or server is invalid on the machine hosting the database.

Action

Correct the command line **path** parameter used to initiate the server or single user client. The name of the invalid directory is written to the JADE messages log file (**jommsg.log**).

3020 Access to a database file or directory denied

Cause

This error occurs if a database operation attempts to access a file or directory and the account under which the database server is running does not have access permission, or (in the case of a file) it is currently in use by a different process.

Action

Check the **jommsg.log** for further details about the underlying operating system error. The account under which the database is running is included in diagnostic log messages. Take appropriate corrective action or contact JADE support if you need help understanding and correcting the issue.

3021 Open of a database file failed

Cause

This error occurs if an attempt to open a database file by the JADE Object Manager failed.

Action

Use the information contained in the JADE messages log file (**jommsg.log**) to determine the cause of the failure, and then take corrective action. The logged information includes the name of the file and the reason for the open failure.

3022 Illegal attempt to open a file in an unsupported format

Cause

This error occurs if the format of a database file is not supported by the version of the JADE database software in use.

Action

Ensure that your database files are compatible with your release of the JADE software.

3023 Database file requires recovery

Cause

This error occurs when a database file open operation determines that the file was last opened in update mode and has yet to be recovered; that is, the file may require the application of transaction journal records to recover it to a consistent state.

This error indicates an operational misadventure; for example, copying a file from an online updating backup into the database directory and attempting to reset the timestamp.

Action

Take care not to inadvertently try to use a database file copied from an online updating backup or copied from the database directory while the database was active.

3024 Database duplicate begin transaction

Cause

This error occurs if the JADE Object Manager attempts to perform a begin transaction request when the database is already in transaction state.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3025 Database journal directory not valid on this machine

Cause

This error occurs if the fully specified path supplied for the journal directory has been rejected by the operating system as invalid. The JADE messages log file (**jommmsg.log**) indicates if it is the **JournalRootDirectory** or the **JournalArchiveDirectory** specification that is in error.

Action

Correct the directory specification in the JADE initialization file.

3026 Database mode conflict

Cause

This error occurs if you attempt to open the database in a mode that is incompatible with the current mode of the database. The most likely cause of this error is if you attempt to open the database while it is already open in exclusive mode for a different user.

Action

Wait until the database is available and then retry the operation.

3027 Database usage conflict

Cause

This error occurs if you attempt to open the database with a usage that is incompatible with the current database usage. The most likely cause of this error is if you attempt to open the database for NO AUDIT usage while one or more other users have it opened with AUDITED usage.

Action

Wait until the database is available and then retry the operation.

3028 Database file is in use by another process

Cause

This error occurs if you attempt to open a database file that is already open by another process.

Action

Refer to the JADE messages log file (**jommsg.log**) for information about the file. Generally, another program is accessing the file or the database as a whole.

3029 Database audit error

Cause

This error occurs if the JADE Object Manager attempts to perform a database **commitTransaction** instruction for a process that is not in transaction state.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3030 Database control file version not supported

Cause

This error occurs if the version of the database engine DLL **jomdb.dll** is not compatible with the database control file format.

Action

Install the correct versions of JADE libraries or upgrade your database files.

3031 Database control file is required but was not found

Cause

This error occurs if the database control file (**_control.dat**) cannot be located in your default database directory.

Action

Restore the missing file from backup.

3032 Cannot create an object version that already exists

Cause

This error occurs if the JADE Object Manager attempts to create a version of an object that already exists in the database.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3033 A database write encountered an out of disk condition

Cause

This error occurs if a physical write operation to disk was terminated prematurely due to an out-of-disk situation. The database transaction that encountered this error is automatically aborted, to prevent a partial transaction being applied to the database.

Action

Obtain sufficient disk space for the database to grow and resubmit the failed transaction.

3034 File compaction terminated due to error

Cause

This error occurs if a file compaction operation is terminated because of an error in the database.

Action

Refer to the **compact.log** file for a description of the error.

3035 Errors encountered certifying file

Cause

This error occurs if a certification operation encounters an error while verifying the integrity of the structures in a database file using the JADE Database utility, or checking the integrity of the database control file using the **dbcontrolcheck.exe** program.

Action

If the error occurred when running an online backup with the **certify** option set, save a copy of the damaged database files, the transaction journal files, the JADE messages log file (**jommsg.log**), and the **certify.log** file or the **backup.log** file

If the error occurred when using the **dbcontrolcheck.exe** program, run the following command to fix the control file entries and the **DbFile** instances, where possible.

```
dbControlCheck path=database-path ini=initialization-file-name  
[server=singleuser|multiuser] applyFixes=true
```

If your JADE licenses include support, contact your local JADE support center for assistance.

3036 The database file being opened is required but was not found

Cause

This error occurs if the database file could not be found in the database directory or the directory specified in the file map.

Action

Locate the file and copy it to the correct directory, or update the file map.

3037 No write access to database file

Cause

This error occurs if the database is opened for update and does not have write access to the database control file (**_control.dat**); for example, if the database files are read-only and if they reside on non-writable media such as a CD-ROM or they have been copied from a CD-ROM with their read-only attributes retained.

Action

If you require access to a read-only JADE database, the database must be opened in read-only mode by selecting a local read-only server type. Alternatively, if you intend to update the database, the read-only attribute on database files must be reset.

3038 Database object TYPE does not match requested type

Cause

This error occurs if the object buffer type passed in a JADE Object Manager request does not match the type of the object stored in the database.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3039 Schema to Database object size mismatch

Cause

This error occurs if the size of an object buffer that is passed in a JADE Object Manager request does not match the size of the object that is stored in the database. This error normally means that there is a mismatch between the schema definition of an object and the physical representation of that object on disk. For example, this error can occur if the definition of a class changes when properties are added or removed, but the required class versioning and reorganization does not result in the necessary changes to object instances of that class in the database.

This error can also occur in a deployment situation where schema or data files are selectively replaced and where, for any reason, the schema definition for a class no longer matches the instances in the database.

Action

If the error occurs in the persistent database, the only way to correct the problem is to undo the action that resulted in the error. This may involve restoring schema files or both schema and data files from a backup.

You can use the JADE messages log file (**jommsg.log**) to determine the object identifier (OID) of the instance that has a size mismatch. The log shows the size discrepancy and you can determine the class number from the oid. The class number and size mismatch should provide an indication of the source of the problem.

If your JADE licenses include support, you should also report this type of problem to your JADE support center to obtain assistance and to record the fault.

3040 Database is locked due to crash while not audited

Cause

This error occurs if the database was previously opened in NO-AUDIT mode, and terminated abnormally before all updates were committed. It is not safe to use the database, because updates will not have been audited, and therefore the database is most likely in an inconsistent state and is not recoverable.

Action

Restore the database from backup.

3041 Error accessing the Database control file

Cause

This error occurs if an error is encountered while accessing the database control file. This can be due to a security or accessibility problem (for example, if another process is accessing the file), the file becoming corrupted, or the device containing the file encountering read or write errors.

Action

Use the information contained in the JADE messages log file (**jommmsg.log**) to determine the cause of the failure and then take corrective action. The logged information includes the path name of the control file and the actual error that was encountered.

3042 Unable to delete file

Cause

This error occurs if a temporary file cannot be deleted. The database creates temporary files during the file reorganization, compaction, or rebuild process.

Action

Check that the current user has **delete** privileges for the file.

3043 Unable to rename file

Cause

This error occurs if you try to rename a temporary file. The database creates temporary files during file reorganization, compaction, or rebuild process.

Action

Check that a file with the new file name does not already exist in the relevant directory and that there is sufficient disk space for the renamed file.

3044 Unable to find backup file (.BAK)

Cause

This error occurs if the database cannot find the backup of a file that was made before commencing a file reorganization.

Action

The backup file should exist in the database directory. Locate the file, and restore it to this directory.

3045 Database directory not valid

Cause

This error occurs if a database directory that you have specified in a class map is invalid or the drive letter used in the path does not exist.

Action

Correct the database file definition in your schema.

3046 File reorganization terminated due to error

Cause

This error occurs if a file reorganization operation encounters an unexpected error in the database.

Action

Refer to the **jomreorg.log** file located in the directory identified by the **TEMP** environment variable or in the current directory if no **TEMP** environment variable is defined.

In a multiuser JADE environment, errors are output to one of two log files. Local errors are output to the **jomreorg.log** file located in the directory identified by the **TEMP** environment variable or the current directory if no **TEMP** environment variable is defined. Errors on the remote (server) node are output to the **jomreorg.log** file located in the directory identified by the **TEMP** environment variable defined on the server node.

3047 A prior file reorganization was prematurely interrupted

Cause

This error occurs if a database file reorganization fails to complete when the database is restarted and the file is reopened.

Action

Retry the operation. The database restores the file from backup before commencing the interrupted reorganization.

3048 Object edition invalid

Cause

This error occurs if the edition of a requested object is greater than the current edition of the object. This is generally an indication of a problem in the cache coherency protocols between client and server nodes.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3049 Object edition out of date

Cause

This error occurs if the edition of a requested object is less than the current edition of the object. This indicates that the object has been updated by another process.

This error is normally handled internally by the JADE Object Manager and is employed as a mechanism to synchronize editions of binary large objects (blobs).

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3050 File reconstruct failed

Cause

This error occurs if a file rebuild operation encounters an unexpected error in the database.

Action

Refer to the **rebuild.log** file in the database directory for a description of the error.

3051 Operation aborted by user request

Cause

This error occurs when you request that a certify, compact, evaluate free space, or rebuild operation (using **jdbutil.exe**) is to be aborted. (For details, see "[The JADE Database Utility](#)", in Chapter 1 of the *JADE Database Administration Guide*.)

Action

None.

3052 Database engine not initialized

Cause

This error occurs if the JADE Object Manager attempts a database operation without initializing the database.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3053 Database File name not valid for file system

Cause

This error occurs if a database file or directory name that is defined in the schema is not supported by the file system that hosts the file. For example, if the file is supported by a Novell file server and the path name contains non-ANSI Unicode characters, it may not be supported. If you are using long file names on a Novell file server, the OS/2 namespace option must be enabled.

Action

Change the file names to comply with file system constraints, or use a file system that supports the file names you want to use.

3054 Cannot access invalid file number

Cause

This error occurs if the JADE Object Manager attempts to access a file number outside of the valid range of currently defined database files.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3055 Database already initialized

Cause

This error occurs if the JADE Object Manager attempts to initialize the database module more than once.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3056 Error encountered closing database file

Cause

This error occurs if the operating system encounters an error when closing a database file.

Action

Refer to the JADE messages log file (**jommmsg.log**) for a description of the reported error and then take the appropriate action. If this does not provide a solution, contact your local JADE support center or JADE Support if your JADE licenses include support.

3057 Database file timestamp does not match control file

Cause

This error occurs if the database file timestamp does not match the control file timestamp.

Action

The JADE messages log file (**jommsg.log**) contains the timestamp found in the database and the expected timestamp found in the control file. If the wrong control file is present, you must restore the correct control file from backup.

If the correct control file cannot be located, you must restore the entire database from backup and then perform a roll-forward recovery to reapply committed transactions applied since the backup.

3058 Database engine failed to start

Cause

This error occurs if an internal fault is detected while initializing the JADE database engine.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3059 Database is locked for reorganization

Cause

This error occurs if the operation you attempt is not possible at this time because the database is locked while performing a reorganization.

Action

Wait until the reorganization has completed and then retry the operation.

3060 Database read past the current end of file

Cause

This error can occur during routine operation and is normally handled internally by the JADE database module. If the error is not handled and it is encountered by your application, it is an indication of an internal software fault.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3062 Unexpected I/O error encountered accessing database file

Cause

This error is raised by the operating system if an error is found on the last database read or write operation. This can occur for several reasons; for example, a hardware device error, file corruption, a security issue, or a software problem.

Action

Refer to the JADE messages log file (**jommmsg.log**) for a description of the error reported and then take the appropriate action. If this does not provide a solution, contact your local JADE support center or JADE Support if your JADE licenses include support.

3063 Database file control record was not found

Cause

This error occurs if a file operation (for example, opening a file during database recovery) fails to locate a valid entry in the control file for the file. If this error occurs, the control file is corrupted or it has not been properly maintained by the database.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3064 Object record not resident in cache

Cause

This error is used internally by the database during recovery to avoid disk reads when not required.

Although the error should never be seen, if it does occur, it is an indication of an internal software fault.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3065 Record sequence exception in transaction journal

Cause

This error occurs if a record in the transaction journal is out of sequence or corrupted. The header of each journal record contains an integral sequence number and a timestamp.

The sequence number in each journal record must be exactly one higher than the number in its predecessor, and the timestamp must be valid and greater than or equal to its predecessor.

Action

If this error results in a failure in database recovery, the database must be restored from backup.

If the problem occurred during a system crash recovery, it may still be possible to perform an archival recovery by using the last backup and recovering through the transaction journals up to the point where the journals become invalid.

If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

3067 Invalid database role

Cause

This error occurs if internal database logic encounters an invalid or undefined database role.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3068 Transaction journal format not supported

Cause

This error occurs when you attempt to open a database and the format of the latest transaction journal is incompatible with the version of JADE software being used.

Action

This error is likely to occur only if you have upgraded a JADE database to a new release and then restored earlier versions of your binaries and database files while retaining a transaction journal created in the new format. To avoid this problem, ensure that all database files, including transaction journals, are kept synchronized.

It is safe to remove an incorrect transaction journal when the database does not require recovery. In this case, the database switches to the next journal in sequence.

3069 The Database Reorg Manager is still active

Cause

This error indicates an internal software state error.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3071 User disallowed replacing an existing database file

Cause

This error occurs if a database file already exists when you have requested the creation of a new file.

Action

Use the existing database file, or create a new file with a different name.

3072 Database locked for exclusive access

Cause

This error occurs if another user has the database locked in exclusive (single user) mode. Only one user at a time can open the database in exclusive mode.

Action

Wait until the database becomes available and then retry the operation.

3073 Recovery across no audit usage is not possible

Cause

This error occurs if you attempt roll-forward recovery while the database is open for no-audit usage.

The error can also be returned by a secondary database attempting to replay a non-replayable reorganization. Database tracking on the secondary database halts when it encounters the audit discontinuity record.

Action

If your site makes use of roll-forward recovery for production databases, you must ensure that a backup is taken following periods of usage in no-audit mode.

If the error is encountered by a secondary database attempting to replay a non-replayable reorganization, the secondary database must be re-cloned.

3074 Database file directory was not found

Cause

This error occurs if a required database file directory was not found.

Action

Use the information contained in the JADE messages log file (**jommsg.log**) to determine the cause of the failure and then take corrective action. The information that is logged includes the path name of the file.

3076 Illegal update of uncommitted object buffer

Cause

This error occurs if a JADE Object Manager process attempts to update an object buffer that has been modified by a different process, and the transaction that updated the buffer has not yet committed.

The JADE Object Manager locking protocol should prevent this from occurring.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3077 Maximum time to wait for quiet point was exceeded

Cause

This error occurs when an operation cannot complete because a database quietpoint was not achieved within a specified timeout period.

A database quietpoint is a point in time when there are no database transactions in progress. When an operation requests a database quietpoint and one or more database transactions are still in progress, new transactions are blocked at **beginTransaction** and the operation waits for **MaxWaitForQuietPoint** seconds for existing transactions to complete.

If in-progress transactions do not complete within the timeout period, the operation is terminated with this error and new transactions are permitted to proceed.

The **MaxWaitForQuietPoint** parameter can be specified in the [\[PersistentDb\]](#) section of the JADE initialization file.

This error can occur when programmatically transitioning a file to read-only mode by using **changeAccessMode** method of the **DbFile** class. To ensure that the file is in a consistent state for closure and reopen in read-only mode, a database quietpoint is required.

This error can also occur when a Synchronized Database Service (SDS) takeover operation fails because a database quietpoint was not achieved.

Action

Consider increasing the **MaxWaitForQuietPoint** parameter value so that it is sufficiently long to allow the longest transactions in the application to complete.

You should also change your application code to handle this error when requesting a **changeAccessMode** of a **DbFile** to **DB_READ_ONLY** mode. When attempting a database takeover, ensure that all pending transactions are completed before retrying the operation.

Note that large values for the **MaxWaitForQuietPoint** parameter could result in transaction processing delays while pending transactions are blocked for the timeout period.

3078 Unexpected error accessing btree

Cause

This error occurs if an internal error is encountered accessing a database Btree block.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3079 Database is locked for archive backup

Cause

This error occurs if an update operation is received by the database while it is locked for archival backup.

A user has placed the database in archive or quiesced backup mode in order to temporarily suspend update operations while a database backup is taken.

Action

Wait until the database backup is complete and then retry the operation.

As the error is continuable, you could code an exception handler that would wait an appropriate time, enter transaction state, and attempt to continue.

```
handler_for_3079(ex : Exception): Integer;  
begin  
    if ex.errorCode = 3079 then  
        app.sleep(60000);      // Wait to allow backup to complete  
        beginTransaction;      // Enter transaction state again  
        return Ex_Continue;  
    else  
        return Ex_Abort_Action;  
    endif;  
end;
```

3080 Database was not initialized in read only schema mode

Cause

This error occurs if a client node has requested read-only schema access but the server was not initialized in read-only schema mode.

Action

Correct the **ReadOnlySchema** setting in the [[JadeServer](#)] or [[JadeClient](#)] section of the JADE initialization file to reflect the required mode of access.

3081 A re-entrant call was made for the same database user

Cause

This error occurs when a user process sends a request to the database while it is still processing a request for the same user.

Action

The operation can be safely retried.

To report this fault if your JADE licenses include support, contact your local JADE support center or JADE Support.

3082 Attempt to change the name or number of an existing database file

Cause

This error occurs if a schema maintenance operation, usually a schema load, attempts to rename or renumber an existing deployed database file. This operation is not allowed, as it could result in loss of access to data. The most-common cause of this problem is a mismatch in the map file definitions between the user schema and the database control file.

This problem is likely to occur only when deploying new user schema files or loading a schema that has incompatible map file definitions.

Action

Ensure that map file definitions that are being replaced or modified in the operation that triggered this error are compatible with the existing definitions in the database control file.

You can use the JADE Database utility to obtain a list of the file name to number maps defined in the database control file, by selecting any of the common file-based operations. File names and numbers are listed in the list box for file selection.

3083 Cannot update read only system file

Cause

This error occurs if you attempt to update a system object or to create a new system object in a system file that is read-only.

Action

Use the information recorded in the JADE messages log file (**jommsg.log**) to determine the class of object being updated. You may need to avoid performing the operation that results in this error by performing it in some other way.

In addition, contact your local JADE support center or JADE Support if your JADE licenses include support.

3084 Cannot perform this operation when not in backup state

Cause

This error occurs if the database receives an online backup request when it is not in backup state.

Action

Ensure that the database is placed in backup state before performing online backup operations. Online backup operations must be performed in a backup transaction.

3085 Failed to undo BTree operation - state is not done

Cause

This error can occur when undoing a top-level Btree operation during a transaction abort. The error indicates that it is not possible to undo the operation given the current state of the Btree.

This is a fatal error that results in termination of the database server to invoke restart recovery. If restart recovery encounters the same problem, a [3113 - Database recovery not possible - fatal error encountered](#) error occurs.

Action

See error [3113 - Database recovery not possible - fatal error encountered](#), for details about possible causes and the best course of action.

3086 Database system file directory not found

Cause

This error occurs if the directory specified by the **SystemFileDirectory** parameter found in the [PersistentDb] section of the JADE initialization file is invalid for the machine hosting the database.

Action

Correct the **SystemFileDirectory** parameter in the JADE initialization file. The name of the invalid directory is written to the JADE messages log file (**jommsg.log**).

3088 Errors encountered upgrading file

Cause

This error occurs if an error is encountered while attempting to upgrade a file from a previous JADE format to the latest JADE format.

Action

Refer to the **upgrade.log** file for details about the error, and take corrective action before retrying the upgrade.

3089 Database file doesn't require an upgrade

Cause

This error occurs if you attempt to upgrade a file that does not require upgrading.

Action

Do not attempt to upgrade the file.

3090 File upgrade from this version not supported

Cause

This error occurs if the upgrade from this database file version is not supported or the file is not in a recognized format.

Action

Do not attempt to upgrade from an unsupported version.

3091 Database is already in backup state

Cause

This error occurs if you attempt a **beginBackup** transaction when the database is already in backup state.

Action

Change your application code to prevent this.

3092 Attempt to use invalid backup mode for a backup transaction

Cause

This error occurs if you attempt to perform an online backup by executing the [JadeDatabaseAdmin](#) class [beginBackup](#) method and the database is not audited.

Action

Perform a quiesced (offline) backup.

3093 Cannot close transaction journal during an online backup

Cause

This error occurs if an illegal attempt is made to close a transaction journal while online backup is in progress.

Action

Change your application code to prevent this.

3094 Cannot perform this operation when database requires recovery

Cause

This error occurs if the database requires recovery.

Action

Recover the database first, by opening and closing the database using current version binary files.

3095 The backup directory parameter cannot be null for this operation

Cause

This error occurs if you attempt to call a method of the [JadeDatabaseAdmin](#) class with a **null** value in the backup directory parameter (**backupDir**). The parameter **backupDir** must be specified.

Action

Check the signature of the method, and change your application code to ensure that the method call passes a valid parameter for **backupDir**.

3096 The attempted database operation is not valid in its current mode

Cause

This error occurs if an invalid state is detected in the protocol used by the JADE Database Administration framework to communicate with the database engine.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3097 Transaction journal is not the correct edition

Cause

This error occurs during recovery from online or offline backup if the transaction journal does not contain certain control records required for the particular type of recovery being performed.

This occurs if a version of the transaction journal that was retained prior to the backup being performed is erroneously used as the latest version for use in recovery of an online backup or roll-forward recovery from an offline database backup.

Action

Restore the correct version of the transaction journal and re-initiate backup or roll-forward recovery.

3098 A database file checksum error was encountered

Cause

This error occurs when restoring a database file from backup or verifying a backed-up or restored database with the JADE Database utility [Verify Checksum](#) command if the data content of the file being processed is corrupted.

Action

You cannot restore a corrupted database backup, and you cannot run your application on a corrupt restored database. Determine the cause of the database corruption in order to determine the appropriate course of action.

If your JADE licenses include support, contact your local JADE support center for assistance. The files you have saved will need to be made available to the support center engineers for analysis. When the problem has been identified, an appropriate course of action can be determined to restore your database to an operational state.

3099 Database file header record is invalid

Cause

This error occurs if the file header record has incorrect leading or trailing tag and length values. This normally indicates a physical file corruption.

Action

Take the database offline as soon as possible and use the [Certify Files](#) command from the Operations menu in the JADE Database utility to analyze the database. Results of the analysis are recorded in the **certify.log** file. Save a copy of the damaged database files, the transaction journal files, the JADE messages log file (**jommmsg.log**), and the **certify.log** file.

3100 Illegal transient object reference sent to persistent DB

Cause

This error occurs when a transient object reference is sent in a **get** or **put** request to the persistent database engine. The error indicates that there is a problem in the internal protocol used to communicate between the object manager kernel and the physical database engine.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

The files you have saved will need to be made available to the support center engineers for analysis. When the problem has been identified, an appropriate course of action can be determined to restore your database to an operational state.

3101 Illegal persistent object reference sent to transient DB

Cause

This error occurs when a persistent object reference is sent in a **get** or **put** request to the transient database engine. The error indicates that there is a problem in the internal protocol used to communicate between the JADE Object Manager kernel and the physical database engine.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3102 Database state not valid for roll-forward recovery

Cause

This error occurs when the database determines a roll-forward recovery cannot be permitted because the database state is inconsistent with a restored offline backup.

Action

Refer to the JADE messages log file (**jommmsg.log**), where the reason the utility recover request was rejected is logged.

The following message is logged when a roll-forward recovery is attempted with an online backup.

```
Roll-Forward Recovery Error: cannot process an online backup
```

In this case, you must use the JADE Database utility [Restore](#) command to process an online backup.

The following message is logged when a roll-forward recovery is attempted with a database image that is *not* in a closed state.

```
Roll-Forward Recovery Error: database requires Restart Recovery
```

A backup containing a database image that is not in a closed state can occur if:

- You attempt to copy the files when the database is in a crashed and non-recovered state.

The JADE Database utility **touchDB** command is provided, to open and close the database and ensure it is not in a recovery state. Restart recovery is performed when the database is opened, if required. If the state of the database is unknown when the backup is to be performed, you should execute the **touchDB** command prior to copying the files.

- There are third-party backup tools copying files while the database is open.

The database opens files in exclusive mode to prohibit external access to the files. However, some third-party backup tools utilize privileged operating system functions to access and copy the file data. Database files copied in such a manner form an inconsistent database image and are invalid for any purpose.

3103 Database character set encoding not compatible with the software

Cause

This error occurs when an attempt is made to open a database that has an internal character set encoding that is incompatible with the software being used.

Action

Use compatible versions of code files to match the database format.

3104 Process opening database is running on an incompatible OS platform

Cause

This error occurs when you attempt to open a database created on a specified platform (operating system and hardware platform) with code files executed from an incompatible platform.

Action

Use compatible versions of code files to match the database operating system platform.

3105 Non local drive not supported

Cause

This error occurs if you attempt to use a network drive and network drives are not supported.

Action

Change your application code to use a local drive.

3106 Database file access at invalid address

Cause

This error indicates an internal software fault.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3107 Fetched record type doesn't match expected type

Cause

This error is reported by the database if the record type field in a database record does not match the expected type. This would normally indicate a physical file corruption.

This error is also reported by methods of the **JadeAuditAccess** class when the requested record access does not match the type of the audit record being processed.

Database Error Action

Take the database offline as soon as possible and use the [Certify Files](#) command from the Operations menu in the JADE Database utility to analyze the database. Results of the analysis are recorded in the **certify.log** file.

Save a copy of the damaged database files, the transaction journal files, the database control file (**_control.dat**), the JADE messages log file (**jommsg.log**), and the **certify.log** file.

If your JADE licenses include support, contact your local JADE support center for assistance. The files you have saved will need to be made available to the support center engineers for analysis. When the problem has been identified, an appropriate course of action can be determined to restore your database to an operational state.

JadeAuditAccess Method Error Action

The following table contains the **JadeAuditAccess** methods and causes that can generate this error.

Method	Cause
getAfterPropertyValue	Record has no after image
getBeforePropertyValue	Record has no before image
getBlobValue	Record is not a create, update, or delete blob
getChangeUserData	Record is not a user code change
getCollectionBlockKeys	Record is not a create, update, or delete of a MemberKeyDictionary class block
getCollectionBlockOids	Record is not a create, update, or delete of a collection block
getUserData	Record is not a sign-on or sign-off record

Take the appropriate action to fix your code.

3108 Fetched record length doesn't match expected length

Cause

This error occurs if the record length field in a database record does not match the expected length. This normally indicates a physical file corruption.

Action

Take the database offline as soon as possible, and use the [Certify Files](#) command from the Operations menu in the JADE Database utility to analyze the database. Results of the analysis are recorded in the **certify.log** file. Save a copy of the damaged database files, the transaction journal files, the database control file (**_control.dat**), the JADE messages log file (**jommsg.log**), and the **certify.log** file.

If your JADE licenses include support, contact your local JADE support center for assistance. The files you have saved will need to be made available to the support center engineers for analysis. When the problem has been identified, an appropriate course of action can be determined to restore your database to an operational state.

3109 Fetched record OID doesn't match requested OID

Cause

This error occurs if the record oid (internal object identifier) field in a database record does not match the expected oid. This would normally indicate a physical file corruption.

Action

Take the database offline as soon as possible and use the [Certify Files](#) command from the Operations menu in the JADE Database utility to analyze the database. Results of the analysis are recorded in the **certify.log** file. Save a copy of the damaged database files, the transaction journal files, the database control file (**_control.dat**), the JADE messages log file (**jommsg.log**), and the **certify.log** file.

If your JADE licenses include support, contact your local JADE support center for assistance. The files you have saved will need to be made available to the support center engineers for analysis. When the problem has been identified, an appropriate course of action can be determined to restore your database to an operational state.

3110 Roll-forward terminate date-time antedates audit trail

Cause

This error occurs when a roll-forward to date-time is attempted with termination criteria earlier than the first audit trail record.

The audit trail date-time and the termination date-time criteria are written to the **recovery.log** file.

Action

Re-issue the roll forward request with corrected termination date-time criteria.

3111 Insufficient system resources

Cause

This error occurs when the operating system refuses service to the database due to constrained resources.

Action

Refer to the JADE messages log file (**jommsg.log**) log file for the value of the operating system error code. It is possible that the system is under-configured for the applied workload.

If you consider the operating environment to be sound with adequate memory resources, refer the problem to your operating system vendor, as there may be a resolution available in a service pack. There have been instances where operating system bugs have caused such errors.

3112 Database transaction audit file has exceeded maximum size

Cause

This error is raised when an audit record is written to the current database journal at an offset larger than 3G bytes. If this happens, the server is shut down to ensure that the database remains in a recoverable state. However, a switch to a new journal file should occur long before the current journal reaches this size.

The journal switch mechanism could be blocked for a number of reasons. A common operationally induced cause is insufficient disk space, which can be verified by examining entries in the JADE messages log file (**jommsg.log**).

Action

If lack of disk space is the cause, ensure that sufficient space is always available for the creation of database journal files; otherwise, if your JADE licenses include support, contact your local JADE support center or JADE Support, providing the JADE messages log file together with the dump file that was created when the server shut down.

3113 Database recovery not possible - fatal error encountered

Cause

This error occurs when database restart recovery encounters a fatal error and cannot continue.

The most-likely cause of this error is that one or more database files have been corrupted. The JADE messages log file (**jommsg.log**) contains further details about the error, including the name of the file being recovered and the transaction recovery record that was being processed when the error was detected.

Action

Restore the database from backup and then perform a roll-forward recovery to reapply transactions completed since the backup. Ensure that you ascertain and eliminate the root cause of the problem, especially if it is hardware-related, to avoid a recurrence of the error.

If your JADE licenses include support and you need assistance with recovering your database, contact your local JADE support center or JADE Support.

3114 Checksum mismatch in fetched data

Cause

This error occurs when the checksum calculation, performed on a record fetched from disk, does not match the checksum that was computed and stored with the record when it was written to disk.

A checksum mismatch means that the record was only partially written to disk or the record has been corrupted for some other reason. (For details, see "[Data Corruption](#)", in Chapter 3 of the *JADE Database Administration Guide*.)

Action

Restore the database from backup and then perform a roll-forward recovery to reapply transactions completed since the backup. Ensure that you ascertain and eliminate the root cause of the problem, especially if it is hardware related, to avoid a recurrence.

3115 GetObject request buffer too small

Cause

This error occurs when there is a mismatch between the class definitions loaded into memory and the actual definitions in the database. It can also indicate an internal error due to character conversion issues between nodes.

Action

If the schema was changed around the time that the exception occurred, stop and restart the node on which it occurred. This refreshes class definitions loaded into memory. In these circumstances, the exception is similar to the [3039 - Schema Database object size mismatch](#) exception.

If your JADE licenses include support and the issue is not resolved by stopping and restarting the node, contact your local JADE support center or JADE Support.

3116 Database file is locked for reorganization

Cause

This error occurs if there is an attempt to update a file marked as physically reorganizing. It also occurs if an attempt is made to update an object in any user data file (whether that file is involved in the reorganization or not) during the offline phase of a reorganization. Only the reorganization process is allowed to update files during this phase.

Action

Wait for the reorganization to complete before continuing.

3120 The database file is not defined in the control file

Cause

This error occurs if a database request attempts to access a file by a number that is not defined in the database control file. It means that a database file is defined in the schema but not in the database control file, so a mismatch normally occurs only as a result of an operational error where the ability to detect this mismatch via timestamp checking has been subverted by using the JADE Database utility reset timestamp operation, or similar. (See "[Resetting Timestamps](#)", in Chapter 3 of the *JADE Database Administration Guide*.)

Action

Investigate the steps that lead to this mismatch and take corrective action, if possible. If corrective action is not possible, a restore to a database that is known to be in a self-consistent state will be necessary.

3121 The database file or file partition is not created

Cause

This error is raised by database file operations such as the **DbFile** class **backupFile** method when the database file has been deleted or it is defined but not yet created. An exception with this error code is of concern only if the file was previously created and is expected to exist.

Action

Database administration or backup applications should handle this error with an appropriate exception handler tailored to site requirements. The JADE sample backup application in **RootSchema** reports all files that are defined but not yet created at the end of the backup process.

3122 Parameter value is out of range

Cause

This error occurs when a parameter value passed to a database method falls outside of the valid range for the specific value.

Action

Provide a value that falls within the documented range.

3123 Access to objects in this secondary database is currently disabled

Cause

This error occurs when an inquiry application that is executing on a secondary database system attempts to access persistent non-system objects when read-access to the database is disabled.

Action

Inquiry applications running on a secondary database should arm an exception handler to handle this error. JADE system administrators may want to avoid occurrences of this error by ensuring that inquiry users and applications are not using the system when read-access is disabled.

3124 Backup is incomplete or backupinfo corrupted

Cause

This error occurs if an attempt is made to restore an incomplete database backup. It is also possible that the **backupinfo** file is corrupted. In either case, the database backup is unusable.

Action

Find and restore a complete and checked database backup.

3125 A required transaction journal was not found

Cause

This error occurs when a transaction journal required for a database recovery or secondary database replay operation was not found. The path name of the missing journal is recorded in the JADE messages (**jommmsg.log**) log file.

Action

Locate and restore the missing journal and repeat the failed operation.

3126 Roll-forward terminate journal value is invalid

Cause

This error occurs when a roll-forward is attempted with an end journal number less than the required start journal for the database backup as stored in the control file. The recovery is terminated with this error.

Action

Re-issue the roll-forward request with a corrected termination end journal number.

3127 Object State Manager: Entry not found

Cause

This error occurs when a required entry is not found in the object state manager index used by the transaction isolation mechanism in the database engine.

Action

If your JADE licenses include support, report this error to your local JADE support center or JADE Support.

3128 Licensed database size exceeded

Cause

This error occurs if a database operation attempts to open or extend a database file and the total size of user data files open at the time exceeds the maximum size allowed by the terms of your license agreement.

Action

Avoid exceeding the database size restriction or purchase a license without size restrictions. Refer to your license agreement for further details.

3129 Database control file does not require upgrade

Cause

This error occurs if an attempt is made to convert the control file (that is, **_control.dat**) using the batch JADE Database utility **jdbutilb** executable when the control file does not require conversion. (The database code version is updated when the database is closed after it has been used, rather than when the control file is converted.)

Action

None.

3130 Invalid record sequence data in transaction journal

Cause

This error occurs when recovery has encountered a corrupt journal record header synchronization field and cannot proceed. A copy of the journal file appended with the **.bad** extension is automatically made and the recovery action is abandoned.

Action

Restore the journal from backup and retry recovery. If no backup of the journal is available, the database must be restored from backup and a roll-forward recovery performed through journals specifying termination at a point-in-time that precedes the corrupt record.

The journal saved with the **.bad** extension should be submitted to JADE Support with an incident report for analysis.

3131 Class version of journal record does not match requested version

Cause

This error is raised by the [JadeAuditAccess](#) class [getBeforeImage](#) and [getAfterImage](#) methods when the mapped class version number is not zero and it does not match the class version number of the object in the audited update record.

Action

Reload the schema definition in the analyzing system so that it matches the definition of the system that generated the journals. Load a class number map file that was created on the generating system containing the correct version information.

3134 Inconsistent system file versions

Cause

This error is raised if the version of one or more system files does not match the version of the **_system.bin** file.

Action

Ensure that all system files are at the same version level.

3135 Null Audit header exception

Cause

This is an internal error used by the database journal read logic to signify that an audit record followed by a region of nulls was read. This is a benign condition when the record is contained in the last block of the current journal. If this error is reported during an operation such as roll-forward recovery, it means that a null region was detected in a journal block other than the final block and the operation is terminated at that point.

Action

If this error results in a failure in database recovery, the database must be restored from backup.

If the problem occurred during a system crash recovery, it may still be possible to perform an archival recovery by using the last backup and recovering through the transaction journals up to the point where the journals become invalid.

If your JADE licenses include support and you require assistance, contact your local JADE support center or JADE Support.

3137 Error encountered Redoing a BTree operation

Cause

This is an internal error returned when redoing a block operation during recovery or transaction abort.

If the **DisableAutoReindex** parameter in the **[PersistentDb]** section of your JADE initialization file is set to the default value of **false** and the error is encountered in restart recovery, these errors trigger post-recovery reindexing on the affected files. If the **DisableAutoReindex** parameter is set to **true**, the recovery operation subsequently fails with a 3113 error ([Database recovery not possible - fatal error encountered](#)).

Action

The errors can provide useful diagnostic information if the **DisableAutoReindex** parameter is set to **true**, which prevents the automatic reindexing.

3138 Error encountered Undoing a BTree operation

Cause

This is an internal error returned when undoing a block operation during recovery or transaction abort.

If the **DisableAutoReindex** parameter in the **[PersistentDb]** section of your JADE initialization file is set to the default value of **false** and the error is encountered in restart recovery, these errors trigger post recovery reindexing on the affected files. If the **DisableAutoReindex** parameter is set to **true**, the recovery operation subsequently fails with a 3113 error ([Database recovery not possible - fatal error encountered](#)).

Action

The errors can provide useful diagnostic information if the **DisableAutoReindex** parameter is set to **true**, which prevents the automatic reindexing.

3139 Error encountered Undoing a block operation

Cause

This is an internal error raised when undoing a block operation during recovery or transaction abort.

Action

If feasible, retain a copy of the corrupted database for analysis. Restore the database from a checked backup and then perform a roll-forward recovery to reapply committed transactions applied since the backup.

If your JADE licenses include support, contact your local JADE support center for assistance with root cause analysis.

3140 Error accessing a database partition control file

Cause

This error occurs if an error is encountered while accessing a partition control file. This can be due to a security or accessibility problem (for example, some other process has it open), the file becoming corrupted, or a hardware failure that results in file read or write errors.

Action

Use the information contained in the JADE messages log file (**jommsg.log**) to determine the cause of the failure and then take corrective action. The logged information includes the path name of the control file and the actual error that was encountered.

3141 Partition operation not valid: file is not partitioned

Cause

This error is reported when an operation that is applicable only to a partitioned file is attempted on a non-partitioned file.

Action

Do not attempt the invalid operation.

3142 Cannot change the partitioned state of an instantiated file

Cause

This error is reported when the **setPartitioned** method is called on an instantiated database file that contains objects.

Action

Empty the file or use the **jdbadmin** utility **MakePartitioned** command to convert an existing non-partitioned file to a partitioned structure.

3143 Database Partition not found

Cause

This error occurs if a partition file operation (for example, opening a partition during a database backup) fails to locate a valid entry in the partition control file. If this error occurs, the partition control file is corrupted or it has not been properly maintained by the database.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3144 Cannot access an offline file partition

Cause

This error occurs if your application code attempts to create, update, or delete an object in a database file or file partition that is absent because it is offline. It can also occur if a partition operation is attempted on an offline partition that requires the partition to be online; for example, calling the [freeze](#) method of the [JadeDbFilePartition](#) class.

Action

Your application code should detect this and handle the situation appropriately in an exception handler method.

3145 Database partition is locked for admin

Cause

This error occurs when a database operation such as accessing an object needs to open a partition that is locked for administrative purposes.

Action

Retry the failed operation when the partition is no longer locked.

3146 Partition ID or partition index is out of range - refer to jommsg.log

Cause

This error occurs when a database operation refers to a partition identifier that is outside of the assigned range or an object is created with a partition index that exceeds the value of partition modulus - 1.

Action

Refer to the JADE messages log file ([jommsg.log](#)) for help in identifying the failed operation, the value that is out of range, and the current valid range for the value. Correct the application logic or command script that is at fault.

3147 Operation not valid: partition required for object creation

Cause

This error occurs when an operation such as **Freeze** or **MarkOffline** is attempted on a partition required for object creation that is a partition in the creation window subset of partitions defined by partition modulus.

Action

Ensure that partition modulus is set to the correct value for the partitioning strategy in use.

3148 Reorganization collection FastBuild invalid parameter

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3149 Reorganization collection FastBuild I/O error

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3150 Reorganization collection FastBuild assert failed

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3151 Reorganization collection FastBuild construction error

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3153 Invalid record size for database file

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

To avoid this error, abort and retry the reorganization with collection fast build disabled.

3154 Invalid buffer size for database file

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs. To avoid this error, abort and retry the reorganization with collection fast build disabled.

3155 Read/write operation conflicts with file mode

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs. To avoid this error, abort and retry the reorganization with collection fast build disabled.

3156 Cannot perform this operation when in backup state

Cause

This error occurs when the database receives a file operation request that cannot be performed because the database is in backup state; that is, an online backup is in progress. Operations that can raise this exception include file or partition removal, moving a partition, marking a partition as absent, and changing the location of a partition.

Action

Defer the operation until the online backup is completed.

3157 Predecessor reorg job was unsuccessful

Cause

This error occurs if a reorganization job cannot run because it depends on the successful completion of another job that failed.

Action

Examine the **jommsg.log** and **reorg.log** files to determine the problem.

3158 Partition method execution failed

Cause

This error occurs when execution of a partition method encounters an unhandled exception.

This error can be returned when an application implementation of the **autoPartitionIndex** method is invoked during the processing of the commit and execution of the method encounters an exception. In this case execution of the **commitTransaction** instruction encounters the exception.

The error can also occur during execution of the **MakePartitioned** command, in which case, the operation is terminated.

Action

Use the information contained in the JADE messages log file (**jommsg.log**) and application exception log to determine the cause of the failure and then take corrective action.

3159 Partition not assigned

Cause

This error occurs when an object is referenced and its partition cannot be determined by consulting the partition index.

Action

This error can occur if an application process attempts to refer to an object (using its OID) and the object is not visible to the referencing process; for example, the transaction that created the object could have aborted or has not been committed.

Note The process that creates an object can always access the object even if it overflows node cache and has not been assigned a partition.

3160 Partition for parent of single file UDR has not been assigned

Cause

This error occurs when an exclusive **singleFile** instance of **JadeBytes** is sent to the database engine and the partition of its owner has not been assigned.

Action

This exception can occur when an exclusive instance of **JadeBytes** is populated in the same transaction that created the owner object and the partition index has not been explicitly set. There are two solutions:

- Create the parent and populate the **JadeBytes** property in separate transactions
- Call **setPartitionIndex** to assign a partition to the parent before populating the **JadeBytes** property

3161 Partition modulus cannot be set to a value outside the range of 1 to 1024

Cause

This error occurs when the **setPartitionModulus** operation is invoked to change the partition modulus of a partitioned file to a value outside of the range **1** through **1024**.

Action

Change the application logic or command script to ensure that the value is within range.

3162 Cannot access an offline database file

Cause

This error occurs if application logic attempts to create, update, or delete an object in a database file that has been marked *offline*. It can also occur if a file operation is attempted on an offline file that requires the file to be online; for example, a file reorganization, compact, or certify operation.

Action

If the file is required for an administrative task, the offline file should be restored and marked as online.

3163 Cannot perform this operation on a partitioned database file

Cause

This error occurs when the **jdbadmin** utility **MakePartitioned** command is executed and the specified file is already partitioned.

Action

No action is required.

3164 Reorganization has no map entries for class mutate

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3165 Operation not valid for an online partition

Cause

This error occurs when an operation such as [SetPartitionLocation](#) is attempted and the specified partition is not marked as offline. The [SetPartitionLocation](#) operation is intended to change the specified location of a partition that has been taken offline before it is brought online; for example, to mount an archival partition burnt to optical media on a specified device or mount point.

Action

In the example scenario, use the [MovePartition](#) operation to move an online partition to a different location.

3166 Creation window cannot include frozen partitions

Cause

This error occurs if the [setPartitionModulus](#) operation is called to change the partition modulus and the creation window defined by the new modulus includes frozen partitions.

Action

Ensure that the partition modulus conforms to the partitioning strategy in use and that all partitions required for object creation can be updated.

3167 Cannot perform this operation on a non-empty partition

Cause

This error occurs when an operation (for example, partition removal) is attempted on a partition that is not empty (that is, the partition contains objects).

Action

Ensure that the objects in the partition you are attempting to remove are no longer required. If the objects are no longer required, execute a purge to ensure that the objects in the partition are no longer referenced from the active model.

Note Once a file partition is removed, it cannot be re-assigned or brought back into the database.

3168 Reorganization collection FastBuild replay error

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3169 Rollback stack boundary error

Cause

This error is only possible when using a feature that is for internal use only.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3170 Database is locked for rollback

Cause

This error is only possible when using a feature that is for internal use only.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3171 Delta database ID mismatch

Cause

This error occurs when a database server node is started with the delta database active and the root database does not match the delta database that was created from it when the root database went into delta mode most recently.

A check is made to ensure that the delta database and root database match. If they do not match, the server node is shut down. The following diagnostic messages are recorded in the JADE messages (**jommsg.log**) log file.

```
*** ERROR: Delta database ID does not match the ID of the last delta ID created by  
this root ***
```

```
*** Root created delta database ID: { <uuid of the last delta database created by  
root database> } ***
```

```
*** Failed to activate delta mode with delta database ID: { <uuid of the delta  
database> } ***
```

Action

Locate and restore the required version of the delta database. If this is not possible, the server node can be started in non-delta mode by setting the **ResetDeltaModeOnRestart** parameter in the [**JadeServer**] section of the JADE initialization file to **true**.

The server node takes the root database out of delta mode and restarts in normal mode; that is, not using a delta database. The contents of the delta database are lost and it will be re-created when the delta database is next activated.

If all else fails, the [clearDeltaMode](#) command of the batch JADE Database utility can be used to clear the delta mode state persisted in the root database control file.

3172 Operation not permitted when database is in delta mode

Cause

This error occurs when a database operation that is considered invalid for a database that is in delta mode is attempted. Refer to the documentation on administrative restrictions on a delta database.

Action

Avoid performing the operation while the database is in delta mode.

3173 File not backed up

Cause

This error occurs if:

- The file or partition is not backed up in the specified location when using the [restoreFile](#) or [restorePartitionFile](#) command.
- An error condition is detected when copying the file or partition information from the base **backupinfo** file using the [makeBackupInfo](#) command of the batch JADE Database utility program.

When a file or partition is not backed up in a delta, the file cannot have been modified between the base backup and the delta backup. When modification is detected, the 3173 error occurs because the file or partition should have been backed up. Current and base **backupinfo** file update timestamps must match and the single Unstructured Data Resource (UDR) file lists must match.

Action

For more-specific information, see the JADE messages ([jommsg.log](#)) log file.

3174 Base backupinfo is newer than delta backupinfo

Cause

This error occurs when using the [makeBackupInfo](#) command of the batch JADE Database utility program to create a new backup information file (**backupinfo**) if the base (full) **backupinfo** file is more recent than the current delta (partial) **backupinfo** file.

Action

Check that you have correctly specified the location of the base and delta **backupinfo** files.

3175 Partitioned status of file differs in base and delta backupinfo files

Cause

This error occurs when using the [makeBackupInfo](#) command of the batch JADE Database utility program to create a new backup information file (**backupinfo**) if the partitioned status of a file is different in the base (full) **backupinfo** file is more recent than in the current delta (partial) **backupinfo** file and the file was not backed up in the delta backup.

Changing the partitioning status of a file constitutes a modification, which means the file must be included in the delta backup.

Action

Check that you have correctly specified the location of the base and delta **backupinfo** files.

3176 Base backupinfo is not full backup

Cause

This error occurs when using the [makeBackupInfo](#) command of the batch JADE Database utility program to create a new backup information file (**backupinfo**) if the file specified as the (full) **backupinfo** file is not from a full backup.

Action

Check that you have correctly specified the location of the base **backupinfo** file.

3177 Fetched block type does not match expected type

Cause

This error occurs if the **type** field in a disk block does not match the expected type. This would normally indicate a physical file corruption.

Action

If feasible, retain a copy of the corrupted database for analysis. Restore the database from a checked backup and then perform a roll-forward recovery to reapply committed transactions applied since the backup. If your JADE licenses include support, contact your local JADE support center for assistance with root cause analysis.

3178 File unusable - initialisation incomplete

This error occurs if an attempt to open a database file fails because the file was not created properly during a reorganization.

Action

Reload database files from backup.

3179 The file must be created as an encrypted file

Cause

This error occurs if an attempt is made to create a database file after it has been deleted, by using the JADE Database utility **jdbutil**, and the file was encrypted before it was deleted.

Action

Use **jdbcrypt** to mark the file for encryption; this will cause the file to be created.

3180 Operation not permitted on this file

Cause

This error occurs in two cases.

- When a transaction has created an object in a partitioned database file and this overlaps with an existing transaction that has added one or more new partitions to the file, the transaction creating the object will encounter an exception with this error.
- When a transaction creating a new partition overlaps with an existing transaction that has added one or more new partitions to the file, the second *create partition* action will encounter an exception with this error.

Action

Implement logic in your application or work flow to serialize these operations.

3181 Attempt to replay incomplete journal

Cause

This error occurs when the **jdbupgrade ReplayJournals** command encounters an incomplete journal. A journal is considered incomplete in the following scenarios.

- It was not properly closed, which normally means the database that created the journal was not closed and requires recovery.
- The journal is truncated; that is, the logical end-of-file stored in the journal header is greater than the physical file size.
- The last record in the journal is not a database close or audit switch.

Action

Ensure that all journals used for upgrade replay are complete and the database was closed before the final journal set was captured. The **jdbutil verify journals** operation can be used to check journal integrity.

3182 SingleFile JadeBytes file is ReadOnly

Cause

This error occurs when an attempt is made to update a dedicated **JadeBytes** instance whose associated disk file has its **ReadOnly** file attribute set to **true**. This is not allowed. A probable cause is that something other than the database set the attribute as this attempt to update a property marked **readOnly** should generate error 1348.

Action

Manually reset the **ReadOnly** attribute of the file, or set the **readOnly** property of the **JadeBytes** instance in code to the required state. The database will then change the file attribute to match the requested state.

3183 Database INI file is missing or invalid

Cause

This error occurs when the initialization file name passed to the database module during initialization is invalid, or the file is not present.

Action

Retry, specifying a valid, present initialization file.

3184 A dual update failed during an online reorganization. It must be aborted

Cause

This error occurs during the dual update phase of an online reorganization when a mutate failed and the reorganization has been marked as failed. This can be caused by some conversions (for example, a **String** to a **Decimal**) failing because of an out-of-range value error.

Action

Abort the reorganization. Identify the underlying cause from the **jomreorg.log** file and then repeat the reorganization.

3185 Updating the reorg restart state from the database failed

Cause

This error occurs when an unexpected problem occurs while trying to store restart information following a failed reorganization.

Action

Perform the following actions.

1. Abandon the reorganization.
2. If the original problem can be resolved as indicated in the **jomreorg.log** file, then do so.
Refer to additional details output to the reorganization (**jomreorg.log**) log file.
3. Restart the reorganization.

3186 Operation not valid: partition being purged or dropped

Cause

This error occurs when an object create or object update operation is attempted that requires the partition being purged or dropped.

Action

Coordinate partition administrative actions with application availability requirements.

3187 Object buffer partition may only be set during create

Cause

This error occurs when calling the **setPartitionID** or **setPartitionIndex** method on a committed object buffer.

Action

Change the application logic to ensure that the **setPartitionID** or **setPartitionIndex** method is called only on an uncommitted buffer; for example, within the transaction creating the object.

3189 Operation not permitted when database has files or partitions with **unaudited=true**

Cause

This error occurs when you attempt to create a delta database and there are unaudited files or partitions.

Action

Ensure all files and partitions are in an audited state before attempting to create a delta database.

3190 Operation not permitted on file or partition with **unaudited=true**

Cause

This error occurs when you attempt an operation on a file or partition that requires auditing to be enabled; for example, updating, compacting, re-indexing, or reorganization.

Action

Ensure the file or partition is in an audited state before attempting the operation on the file or partition.

3191 Database is locked for snapshot backup

Cause

This error occurs when a database mode change (for example, to archive or reorganize) is attempted while the database is in snapshot mode.

Action

Retry the operation after completing the snapshot backup and exiting from snapshot mode.

3192 Snapshot recovery terminated before end-snapshot condition established

Cause

This error occurs when a termination condition specified on the roll-forward operation was satisfied before the minimal required audit records to re-establish file integrity were processed.

An **END_SNAPSHOT** record indicates end of snapshot; **DATABASE_CLOSE** or **DATABASE_OPEN** records imply end of snapshot.

Action

Review early termination specifications; that is, **end-journal** and **date-time**.

3193 Roll-forward terminate serial number is earlier than audit trail

Cause

This error occurs when a roll-forward to serial number is attempted with termination criteria earlier than the first audit trail record.

The audit trail serial number and the termination serial number criteria are written to the **recovery.log** file.

Action

Re-issue the roll forward request with the corrected termination serial number criteria.

3194 Roll-forward terminate before backup recovery complete

Cause

This error occurs when a roll-forward recovery of an online backup is performed specifying a termination condition that terminates the roll forward before the end backup checkpoint record is processed; that is, before file integrity is restored.

The audit trail serial number and log serial number (LSN) and the **End-Backup** LSN are written to the **recovery.log** file.

Action

Re-issue the roll forward request with termination criteria beyond the **End-Backup** checkpoint record.

3195 Class not valid for file

Cause

This error is raised by the certify or verification process during database backup when the class of an encountered object or subobject is not mapped to the file being processed.

This error may indicate that there is file corruption, or that there is an error in the meta data.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3200 through 3249 Synchronized Database Service Errors

This section describes the errors that can be returned when using the Synchronized Database Service (SDS).

3200 SDS Attempt to replay incomplete journal

Cause

This error is detected and reported by the Synchronized Database Service (SDS) tracker module while replaying a journal, if it determines that the journal it is replaying is somehow incomplete.

When this error is detected, it is recorded in the JADE messages (**jommsg.log**) log file and SDS log file, and the secondary tracker requests a re-send from the primary database and replays the journal once it is received. If three attempts to receive and replay a specific journal fail in sequence, this is considered fatal and database tracking is halted.

Action

If this error is detected, logged, and a retry of the send and replay succeeds, no further action is required.

If multiple journal send and replay operations fail, check the disposition of the original journal on the primary and check and correct the integrity of the network link.

3201 SDS not initialized

Cause

This error occurs when a Synchronized Database Service (SDS) administration method of the **JadeDatabaseAdmin** class is invoked and the SDS is not initialized.

Action

Use the **sdsIsRunning** method to determine whether SDS is initialized, before attempting to use SDS administration methods.

3202 SDS MyName parameter not found in INI file

Cause

This error occurs when Synchronized Database Service (SDS) is activated and the JADE initialization file contains a **DatabaseRole** parameter in the [**SyncDbService**] section but no corresponding **MyName** parameter. A **MyName** parameter is required for an SDS-capable system.

Action

Ensure that the [**SyncDbService**] section of the JADE initialization file contains a **MyName** parameter.

3203 SDS PrimaryServerName parameter not found in INI file

Cause

This error occurs during the initialization of a Synchronized Database Service (SDS) secondary database if the **PrimaryServerName** parameter is missing. A **PrimaryServerName** parameter is required for an SDS secondary system.

Action

Ensure that the [SyncDbService] section of the JADE initialization file for a secondary database contains a **PrimaryServerName** parameter.

3204 SDS secondary not attached

Cause

This error occurs when a Synchronized Database Service (SDS) administration method of the **JadeDatabaseAdmin** class is invoked on a primary system and directed to execute on a specified secondary server but the specified secondary server has disconnected.

Action

Modify your SDS administration application to check the connection state of a secondary database before issuing commands to the secondary. In addition to or as an alternative to this, arm an exception handler to deal with the case in which the secondary disconnects after the check is performed.

3205 SDS invalid command

Cause

This error occurs when an invalid or unsupported command is received by the Synchronized Database Service (SDS) administrative interface.

Action

An occurrence of this is an indication of a mismatch between software modules. Ensure that you have compatible versions of software installed, and if necessary, check with your JADE support center for assistance.

3206 Operation not permitted on a secondary database

Cause

This error occurs when a database operation that is considered invalid on a secondary database is attempted.

Action

Avoid performing the operation on a secondary database.

3207 Operation not permitted on a primary database

Cause

This error occurs when a database operation that is considered invalid on a primary database is attempted.

Action

Avoid performing the operation on a primary database.

3208 SDS named secondary server not found

Cause

This error occurs when a Synchronized Database Service (SDS) administration method of the [JadeDatabaseAdmin](#) class is invoked on a primary system and directed to execute on a specified secondary server, but the specified secondary server is not a registered member of the synchronized database environment; that is, a secondary server with the specified name has never connected to the primary database.

Action

Modify your SDS administration application to ensure that it calls only secondary directed methods with a valid name. You can obtain the names of registered secondary systems by using the [JadeDatabaseAdmin](#) class [sdsGetSecondaryProxies](#) method.

3209 SDS takeover operation failed

Cause

This error occurs if a negotiated or hostile takeover fails for any reason. The reason for the failure to complete the takeover is also logged in the JADE messages ([jommsg.log](#)) log file, and in most cases, the server that failed to complete its role change terminates abruptly.

Action

Use the diagnostics recorded in the JADE messages ([jommsg.log](#)) log file and the Synchronized Database Service (SDS) log file to determine the reason for the failure, correct the condition if possible, and repeat the takeover. This may require performing a hostile takeover.

3210 SDS maximum secondary servers exceeded

Cause

This error occurs when a secondary database server attempts to connect to a primary database server and a further secondary connection would result in exceeding the allowed *max secondary server* limit. The allowed *max secondary server* limit is determined by the lower of [MaxSecondaries](#) configured in the JADE initialization file or **Maximum Secondaries** permitted by the registered primary license key.

When a connection attempt exceeds the allowed limit, the connection attempt is refused and the message *SDS maximum secondary servers exceeded* is recorded in the JADE messages ([jommsg.log](#)) log file and the primary and secondary Synchronized Database Service (SDS) log files.

Action

Avoid exceeding the allowed limit.

Alternatively, reconfigure the **MaxSecondaries** JADE initialization file parameter, or upgrade your primary license, or both, as required, to allow the required number of secondary servers to connect.

3211 SDS tracker thread is busy

Cause

This error occurs when the **JadeDatabaseAdmin** class **sdsStartTracking** or **sdsStartTrackingAt** method is called and the tracker process is busy starting or stopping.

Action

Wait until the tracker process has completely started or stopped and then retry the operation if it is still required.

3212 SDS a response was not received within a reasonable timeframe

Cause

This error occurs when a Synchronized Database Service (SDS) administration method of the **JadeDatabaseAdmin** class is invoked on a primary system and directed to execute on a specified secondary server, and the specified secondary server failed to respond within 60 seconds.

Action

Arm an exception handler to deal with this error. It can occur if the secondary system is busy or the network link to the secondary is congested.

3213 SDS backup terminated by replay of reorg or compact

Cause

This error occurs if the tracker process on a secondary database is required to replay a file reorganization or compact operation and a database backup is in progress. The backup is terminated with this error.

Action

Postpone and retry the backup at a time when reorganization or compact operations, or both, are not going to be replayed.

3214 Operation only permitted on a secondary database

Cause

This error occurs when an attempt is made to perform roll-forward recovery through a range of journals with the **no undo** option specified. The **no undo** option is valid, and is in fact required, only on a secondary database.

Action

Do not use the roll-forward **no undo** recovery option on a non-secondary database.

3215 SDS Transaction not found or already committed

Cause

This error is an internal error that can be recorded in the JADE messages (**jommsg.log**) log file. If this error is reported, there is a problem in the *two-phase* Synchronized Database Service (SDS) commit protocol.

Action

If your JADE licenses include support, report this error to your local JADE support center or JADE Support.

3216 SDS Transaction status not valid

Cause

This error can be detected and reported by the SDS database tracking process. When this error is detected, details are recorded in your JADE messages (**jommsg.log**) log file and database tracking is halted.

Action

Depending on the cause, it may be possible to shut down user applications querying the secondary database and to resume tracking. If resuming tracking immediately encounters the same error, you must restart the secondary database server to clear the condition.

If your JADE licenses include support, report this error to your local JADE support center or JADE Support.

3217 SDS Upgrade version mismatch

Cause

This is an internal error that is returned to the SDS secondary database to halt tracking at the version upgrade boundary when a version mismatch exists. It is never returned to a user application. The following messages are written to your JADE messages (**jommsg.log**) log file.

```
SDS: Secondary upgrade version mismatch: tracking will now halt
```

```
SDS: Upgrade to the same software release level as the primary and restart server
```

Action

Upgrade to the same software release level as the primary and then restart the server.

3218 SDS secondary server name conflict

Cause

This error occurs when:

- A secondary database attempts to connect with an SDS primary database using the same name as another secondary database that is already connected.
- Two or more secondary databases are configured with the same name specified in the **MyName** parameter in the [SyncDbService] section of the JADE initialization file.
- A secondary database disconnects from a primary database (for example, due to a temporary network error) and then attempts to reconnect before the primary database has completely processed the disconnect event.

Action

Ensure that each secondary database in an SDE has a unique name.

3219 Cannot run SDS secondary server in SingleUser

Cause

This error occurs when an attempt is made to start an SDS secondary server node in single user mode; that is, the value of the **DatabaseRole** parameter in the [SyncDbService] section of the JADE initialization file is **SecondaryRole** and the command line specifies **server=singleUser**.

Action

Do not attempt to start an SDS secondary server node in single user mode.

3220 Operation not permitted when primary is connected

Cause

This error occurs when a hostile takeover is initiated at a secondary, while it is connected to the primary.

Action

Shut down the connection between the secondary and the primary.

3221 SDS primary not connected

Cause

This error occurs when an operation is initiated that cannot proceed because the secondary is not connected to the primary.

Action

Establish the connection between the secondary and the primary.

3250 through 3299 Relational Population Service (RPS) Errors

This section describes the errors that can be returned when using the Relational Population Service (RPS).

3250 RPS not initialized

Cause

This error occurs when an administrative method attempts to perform an operation that is not supported on RPS nodes.

Action

Avoid performing this operation on an RPS node.

3251 RPS operation not supported

This error occurs when attempting to set the storage mode for an RPS database to an illegal or unsupported value. The exception is logged in your JADE messages (**jommmsg.log**) log file, along with the illegal value.

Action

If your JADE licenses include support and this error is encountered while using the [RPS Manager application](#), contact your local JADE support center or JADE Support, providing appropriate error logs.

3253 RPS Mapping mismatch with Target Relational DB

Cause

This error occurs when you attempt to start the RPS **Datapump** application and the tables and columns that are defined in the specified RPS mapping do not match the tables and columns in the RDBMS.

Action

Check your JADE messages (**jommmsg.log**) log file for details of the table and column that do not match. The RDBMS must be corrected to match the RPS mapping before the RPS **Datapump** application can be started.

Use the [RPS Manager application](#) on the RPS node to run a consistency check, output the RPS mapping table definitions, and reload the table data, if required.

3254 RPS Target Relational DB - SQL operation failed

Cause

This error occurs when the RPS **Datapump** application receives an error on an insert, update, or delete of the RDBMS.

Action

Check your JADE messages (**jommsg.log**) log file for details of the error returned from the RDBMS. You may need to consult your RDBMS error documentation to determine the cause of the error.

As the transaction in error is replayed on restart of the RPS **Datapump** application, the RDBMS must be corrected to avoid the error before the **Datapump** application can be restarted.

3255 RPS instance ID mismatch with Target Relational DB

Cause

This error occurs when you attempt to start the RPS **Datapump** application when connected to an RDBMS that does not match the RPS system last used with this database.

The relational database identifier is stored in the **JADE_CONTROL_INFO** table of the RDBMS.

Action

Check that the RPS node has been connected to the correct RDBMS system by checking the **RPSConnectionStr** parameter in the [**JadeRps**] section in the JADE initialization file for the RPS node. Check that the **JADE_CONTROL_INFO** table has not been modified externally to the JADE system.

You can reset the relational database identifier value by using the **RPS Manager application**, but ensure that the correct database is being used before resetting.

3256 RPS DB state is later than Target Relational DB

Cause

This error occurs when you attempt to start the RPS **Datapump** application when connected to an RDBMS that has an earlier database state than the RPS node database state. This means that transactions applied to the JADE database have not been applied to the RDBMS.

Action

The RPS JADE database must be restored to a state equal to or earlier than the state of the RDBMS database. Alternatively, the RDBMS database must be recreated and loaded with data to match the RPS JADE database.

3257 RPS Target Relational DB not opened by this application

Cause

This internal error occurs when an attempt is made to close a relational database that is either not open or was opened by another application.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3258 RPS Insert row failed due to a duplicated key

Cause

This error occurs when the RPS **Datapump** application receives a duplicated key error on an *insert* of the RDBMS.

Action

Check your JADE messages (**jommsg.log**) log file for details of the error returned from the RDBMS. You may need to consult your RDBMS error documentation to determine the cause of the error. As the transaction in error is replayed on restart of the RPS **Datapump** application, the RDBMS must be corrected to avoid the error before the RPS **Datapump** application can be restarted.

The RPS mapping option **Alternative Action on Create Exception Policy** can be set to attempt an update of the RDBMS row when this error is encountered.

3259 RPS SQL update failed - zero rows updated or deleted

Cause

This error occurs when the RPS **Datapump** application receives a zero-rows error on an update or delete of the RDBMS.

Action

Check your JADE messages (**jommsg.log**) log file for details of the error returned from the RDBMS. You may need to consult your RDBMS error documentation to determine the cause of the error. As the transaction in error is replayed on restart of the **Datapump** application, the RDBMS must be corrected to avoid the error before the application is restarted.

The RPS mapping option **Alternative Action on Update Exception Policy** can be set to attempt an insert of the RDBMS row when this error is encountered on an update. The RPS mapping option **Alternative Action on Delete Exception Policy** can be set to ignore this error when encountered on a delete action.

3260 RPS SQL update failed - multiple rows updated or deleted

Cause

This error occurs when the RPS **Datapump** application receives a multiple rows updated error on an update or delete of the RDBMS.

Action

Check your JADE messages (**jommsg.log**) log file for details of the error returned from the RDBMS. You may need to consult your RDBMS error documentation to determine the cause of the error.

As the transaction in error is replayed on restart of the **Datapump** application, the RDBMS must be corrected to avoid the error before the application is restarted.

3261 RPS Invalid storage mode

Cause

This error occurs when attempting to set the storage mode for an RPS database to an illegal or unsupported value. The exception is logged in your JADE messages (**jommmsg.log**) log file along with the illegal value.

Action

If this error is encountered while using the [RPS Manager application](#), report the fault.

3262 RPS Reorg admin halt required

Cause

This error occurs if a reorganization occurs while an RPS node is running. The RPS **Datapump** application is automatically shut down and restarted.

Action

If the reorganization does not change the RPS mapping being used or if the changes to the mapped tables can be applied automatically, no user action is required. If the change must be applied by the RDBMS administrator, the **Datapump** application is not automatically restarted.

The RDBMS administrator must apply the alter script to the RDBMS database, reload data if required, and then restart the **Datapump** application using the [RPS Manager application](#).

3263 RPS Failed to open target Relational DB

Cause

This error occurs when the connection to the relational database has not been successful.

Action

Check your JADE messages (**jommmsg.log**) log file for error messages. Check the connection string, user name, and password defined for the RPS node in the RPS Manager RPS Node Configuration dialog. Check the status of the relational database server.

3264 Operation only permitted on RPS node

Cause

This error occurs when the **rpsDataPumpInitialize** or **rpsDataPumpFinalize** methods of the **Application** class, or the **rpsStartDataPump** or **rpsStopDataPump** methods of the **JadeDatabaseAdmin** class are called on a non-RPS node.

You can call these methods only on an RPS node.

Action

Ensure that these methods are not called on non-RPS nodes.

3265 RPS DataPump Application already running

Cause

This error occurs when you attempt to initialize an application as an RPS **DataPump** application using the **rpsDataPumpInitialize** method defined in the **Application** class and an RPS **DataPump** application is already running.

You can run one RPS **DataPump** application only at a time.

Action

Start one RPS **DataPump** application only.

3266 Application not defined as DataPump Application in ini file

Cause

This error occurs when you attempt to initialize an application as an RPS **DataPump** application using the **rpsDataPumpInitialize** method defined in the **Application** class, but the application is not defined as the **DataPumpApplication** in the JADE initialization file of the RPS node.

Action

Check that the application calling the **rpsDataPumpInitialize** method defined in the **Application** class is defined as the **DataPumpApplication** parameter in the **[JadeRps]** section of the JADE initialization file.

3268 RPS DataPump application not running

Cause

This error occurs if the RPS **DataPump** application is no longer running.

Action

Check your JADE messages (**jommsg.log**) log file for details about why the RPS **DataPump** application has terminated.

If you cannot correct the problem with the **DataPump** application and your JADE licenses include support, contact your local JADE support center or JADE Support.

3269 RPS Extract Request Error

Cause

This error occurs if you call the **extractData** method of the **RelationalView** class with an invalid parameter.

Action

Check your JADE messages (**jommsg.log**) log file for details about the parameter in error. Fix the calling code to send the correct parameter.

3270 RPS Relational DB already opened by this application

Cause

This internal error occurs when an attempt is made to open a relational database that has already been opened by the current application, although this should not normally happen.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

3271 RPS Relational DB already opened by another application

Cause

This error occurs when an attempt is made to open a relational database that has already been opened by another application.

Action

Check other applications running which may have opened the relational database. When the current application has completed, run the required application again.

3272 Operation is only permitted on RPS Mapping

Cause

This error occurs when the [generateRpsTableCreationScript](#) method is called for a [RelationalView](#) object that is an ODBC relational view; that is, it is not an RPS mapping.

Action

Before calling the [generateRpsTableCreationScript](#) method, check that the relational view is an RPS mapping by calling the [isRpsMapping](#) method.

3273 RPS Table name not found in RPS Mapping

Cause

This error occurs when the [generateRpsTableCreationScript](#) method is called when the **tablesOption** parameter is set to **RpsScript_SelectedTables** and the **selectedTableNames** parameter includes an invalid table name.

Action

Before calling the [generateRpsTableCreationScript](#) method, check that the relational view is an RPS mapping by calling the [isRpsMapping](#) method.

3274 RPS Target Relational DB - SQL operation connection failure

Cause

This error occurs when the RPS **Datapump** application receives a connection or timeout error.

Action

If the [JadeRps] section **AutoRestartOnError** parameter is set to **true**, the RPS Datapump an attempt will be made to restart the data pump.

If this option is not set, or if the connection is not reestablished and the restart of the **Datapump** application fails, correct the connection problem and manually restart the **Datapump** application.

3275 RPS Oid Instancelid value is too large for Oid Mapping Option

Cause

This error occurs if the oid **Instancelid** is too large for the RPS **Oid Mapping Option** currently selected in the RPS mapping used in an RPS node. The Datapump application will be stopped.

Action

Change the RPS **Oid Mapping Option** to accommodate the current **Instancelid** values and then recreate the RPS node and SqlServer database.

3276 RPS Oid ClassNumber value is too large for Oid Mapping Option

Cause

This error occurs if the oid **ClassNumber** is too large for the RPS **Oid Mapping Option** currently selected in the RPS mapping used in an RPS node. The Datapump application will be stopped.

Action

Change the RPS **Oid Mapping Option** to accommodate the current **ClassNumber** values and then recreate the RPS node and SqlServer database.

3277 Cannot create an RPS node from a standard updating backup

Cause

This error occurs if the initialization of an RPS node was attempted from a standard updating backup.

Action

Make sure that the backup is either:

- Is commit-coherent
- Was created using the RPS manager

3278 Cannot change role to RPS when database requires recovery

Cause

This error occurs if a role change to RPS is attempted on a database that requires recovery.

Action

Create your RPS node through either:

- A commit coherent backup
- Using the RPS Manager

3301 through 3399 Database Encryption Errors Messages

This section describes the errors that can be returned relating to encryption of the JADE user database files.

3301 Database encryption is not supported on this platform

Cause

This error occurs if you are using JADE executables that do not support database encryption. This error should not happen with executables obtained through a standard release.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3302 Database encryption cannot be disabled until all files have been decrypted

Cause

This error occurs when you use the JADE Database Encryption utility (**jdbcrypt**) with the [DisableDatabaseEncryption](#) action and some of the files are still encrypted.

Action

Decrypt the remaining encrypted files before attempting the [ApplyPendingChanges](#) action.

3303 Database encryption has not been enabled

Cause

This error occurs when you use the JADE Database Encryption utility (**jdbcrypt**) to encrypt database files and encryption has not been enabled for the database.

Action

Enable database encryption for the database.

3304 Database system files cannot be encrypted

Cause

This error occurs if you attempt to encrypt a system database file; for example, the **_system.bin** file.

Action

Do not attempt to encrypt the system database files.

3305 Database has no files with pending encryption changes

Cause

This error occurs when you use the JADE Database Encryption utility (**jdbcrypt**) with the [ApplyPendingChanges](#) action and there are no changes to apply.

Action

No action is required.

3321 DbCrypt module uninitialised or terminating

Cause

This error occurs if an attempt is made to use the database encryption module after it has been terminated. This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3323 DbCrypt memory allocation failure

Cause

This error occurs if a memory allocation error occurs in the use of the database encryption module.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3324 DbCrypt DB callback table version mismatch

Cause

This error occurs if the database encryption module encounters mismatched DLLs.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3326 Windows Cryptography error. See jommsg.log

Cause

This error occurs when the Microsoft Windows cryptography module returns an error.

Action

Check the JADE messages log file (**jommsg.log**) for additional information that may clarify the cause of the error.

3327 Database cryptInfo error. See jommsg.log

Cause

This error occurs when the encryption information for a map file or for the database is missing, invalid, or corrupt.

Action

Check the JADE messages log file (**jommsg.log**) for additional information. Recover database files from backup.

3328 DbCrypt failed to decrypt object

Cause

This error occurs when the encryption module is called to decrypt an object but the returned buffer is still marked as encrypted. This could arise if JADE wrongly regards the file containing the object as encrypted.

Action

Stop and restart the node.

3331 DbCrypt decryption failure

Cause

This error occurs when the validation of the object buffer after decryption fails. Possible causes include disk corruption, memory errors, RPC transfer errors, and key corruption. A program dump is created.

Action

If your JADE licenses include support and if your security policy allows (the dump contains decrypted information), send the dump to your local JADE support center or JADE Support for analysis.

Recover database files from backup.

3332 Object is not encrypted

Cause

This error occurs when the database rejects an object create or update because the file is marked encrypted but an unencrypted object buffer has been passed by the kernel.

Action

Restart the client node.

3333 File is marked encrypted but database has encryption disabled

Cause

This error occurs when the database is not encrypted according to the database control file, but when a map file is opened it contains encryption information.

This could arise if the wrong database control file or the wrong map file is being used.

Action

Recover database files from backup.

3341 DbCrypt decrypted object size does not match expected size

Cause

This error occurs when the size of a decrypted object does not match its expected size.

Action

Restart the node. If the problem persists, recover database files from backup.

3344 DbCrypt Master Key not present in account key store

Cause

This error occurs if the database is running from the wrong account. You must run **jdbcrypt** from the same account as the database server so that the master key is stored in the correct account keystore.

Another way in which the error can occur is if the key has been removed. The full name of the key is reported in the JADE messages log file (**jommsg.log**).

Action

Run the database server from the correct account. If the error is related to a missing key, use the JADE Database Encryption utility (**jdbcrypt**) with the [ListStoredKeys](#) action to determine which keys are present, and import any missing key from an exported copy.

3345 DbCrypt Cannot create user objects in unencrypted files

Cause

This error arises if you attempt to create an object in an unencrypted file of an encrypted database and the **MandatoryFullEncryption** option is set to **true**.

Action

Encrypt the database file in which the object is to be written.

Alternatively, disable **MandatoryFullEncryption**, by running the JADE Database Encryption utility (**jdbcrypt**) with the **EncryptFile** action and the **MandatoryFullEncryption** parameter set to **false**.

3348 DbCrypt file crypt status is unknown

Cause

This error occurs when a map file is opened and its encryption state cannot be determined.

Action

Restart the server node.

3349 Delta mode cannot be activated when database encryption is enabled

Cause

This error occurs when database encryption has been enabled and an attempt is made to enter delta mode. Delta mode is not supported in this case.

Action

No action is required.

3350 DbCrypt cannot encrypt object edition zero

Cause

This error occurs if an attempt is made to encrypt an object buffer that has an edition number of zero (0). This is an internal error and should not normally occur.

Action

Check the JADE messages log file (**jommsg.log**) for the OID of the object involved. If your JADE licenses include support, contact your local JADE support center or JADE Support.

3352 Invalid context handle passed to DbCrypt module

Cause

This error occurs if an invalid or out-of-date context handle was passed to a **DbCrypt** module API.

Action

Internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

3353 Attempt to store encrypted object in unencrypted file

Cause

This error occurs if a client node sends an encrypted object buffer to the database for a file that is not currently encrypted.

Action

Stop and restart client nodes to refresh map file encryption information.

3400 through 3499 Database Reorganization Errors

This section describes errors that can be returned when reorganizing your database.

3400 Unable to reorganize class: refer to JOMREORG.LOG

Cause

This error occurs if JADE is unable to reorganize the requested class.

Action

Refer to the reorganization (**jomreorg.log**) log file in the directory identified by the **TEMP** environment variable or in the current directory. The log file contains the class in which the error occurred, to enable you to correct your data before beginning a successful reorganization.

A common data error that can make a reorganization fail is if duplicate keys are found when you have not specified **Duplicates Allowed** in your class definition. The following steps provide an example of the actions to be taken to correct this situation.

1. Check the reorganization log file to determine the class that contains duplicate keys.
2. Open the Define Class dialog for the required class (by selecting the **Change** command from the Classes menu) and then check the **Duplicates Allowed** check box on the **Keys** sheet to allow duplicate keys for that class.
3. Initiate the reorganization of the class or classes in which the data errors were detected.
4. When the reorganization has completed, write additional code to remove the duplicate keys from the collection (for example, by using a method in the **JadeScript** class).
5. Uncheck the **Duplicates Allowed** check box on the **Keys** sheet of the Define Class dialog to disallow duplicate keys, if required.
6. Initiate the reorganization of your schema again.

3404 Reorg operation cancelled by user request

Cause

This error occurs when you request that a reorganization operation be aborted.

Action

None.

3406 Constraint is not compiled

Cause

This error occurs during reorganization if the constraint on a property is not compiled or in error. As a result, the reorganization cannot proceed.

Action

Refer to the reorganization (**jomreorg.log**) log file to determine the name of the constraint and then correct or compile the constraint.

3407 An incomplete reorg must be either restarted or aborted

Cause

This error occurs if a replayable reorganization terminates; for example, because of a system crash.

Action

The reorganization must be restarted by initiating another replayable reorganization. If the restarted reorganization is not replayable, this error will result.

3408 Database reorganization failed

Cause

This error occurs when reorganization encounters an error.

Action

Refer to additional details written to the reorganization (**jomreorg.log**) log file.

If the cause of the error is not immediately obvious and your JADE licenses include support, contact your local JADE support center or JADE Support.

3411 Database reorganized cannot be initiated from a Secondary Server

Cause

This error occurs if there is an attempt to initiate a database reorganization from a secondary server. This is not permitted.

Action

Database reorganizations must be initiated from the primary server.

3412 Previous incomplete schema load must be restarted

Cause

When a large schema file is loaded, schema changes and recovery information are written at regular intervals to the JADE database. If the schema load fails, this error occurs.

Action

Restart the schema load. You can initiate the reorganization when the schema load has completed.

JADE automatically restarts the loading of the schema file from the last recovery point.

3413 Reorg is suspended waiting for the transition to be initiated

Cause

This error occurs when the online phase of the reorganization is complete and the reorganization is ready to initiate the transition.

Action

Use the **JadeReorgApp** application or the **Reorg** command from the Schema menu to initiate the transition and resume the reorganization.

3414 Reorg cannot be restarted, reorg must be aborted

Cause

This error occurs when a non-replayable reorganization failed to complete. This reorganization is unable to be restarted.

Action

Use the **JadeReorgApp** application or the **Reorg** command from the Schema menu to abort the reorganization.

3415 Schema cannot be unversioned, dependent schemas exist

Cause

Dependencies among schemas (based on a variety of conditions) determine whether un-versioning is allowed.

If a superschema is versioned and the latest version of the superschema was created before the schema was processed, un-versioning is not possible and this error occurs. For example, if a schema is added to a schema that has already been versioned, the versioned schema cannot be unversioned.

Action

As un-versioning is not possible, your only course of action is to carry out the reorganization or to restore your system from backup.

3416 Reorg cannot be initiated when a schema compile is in progress

Cause

This error occurs when a database reorganization cannot be initiated because a schema compile is in progress.

Action

Start the reorganization when the compile has completed.

3417 Transition must be initiated in singleuser if production mode is set

Cause

This error occurs when running in production mode, as JADE does not maintain details of classes being used by other nodes.

As the reorganization transition can be initiated only when no classes are in use, the transition must be initiated in single user mode if production mode is set.

Action

Close the database server and complete the database reorganization by running the **JadeReorgApp** application or from the JADE development environment in single user mode.

3418 Database reorganization in progress

Cause

This error occurs if the operation you attempt is not possible at this time because the database is locked while a reorganization takes place.

Action

Wait until the reorganization has completed and then retry the operation.

3419 Schema cannot be unversioned after onlyStructuralVersioning load

Cause

This error occurs if you attempt to un-version a schema after a load into the latest version of a schema with only structural changes has been performed. This is not permitted because any method that was recompiled has references to the new versions of classes and properties.

Action

Do not attempt to un-version a schema after a structural changes only schema load.

3420 RPS reorg extract on primary failed

Deprecated in Version: 2020 and higher

Cause

This error occurs if there is a failure in the automatic extract on a primary SDS node that is supporting an RPS database in [Working Set](#) database replication mode. This could happen, for example, if a column-mapping method is in error.

Action

Check the JADE messages ([jommsg.log](#)) log file for details about the exception raised during the extract. If possible, correct the problem causing the exception and restart the reorganization.

If the problem cannot be corrected, remove the [AutoExtractOnPrimary<n>](#) parameter in the [[JadeRps](#)] section of the JADE initialization file and restart the reorganization. When the reorganization is complete, manually extract and load the data for the RPS node.

3421 Reorg auxiliary server node has not been configured

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

To avoid this error, abort and retry the reorganization with collection fast build disabled.

3422 Reorg auxiliary server node has not been initialised

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

To avoid this error, abort and retry the reorganization with collection fast build disabled.

3423 Reorg auxiliary server node error

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

To avoid this error, abort and retry the reorganization with collection fast build disabled.

3424 Reorg error chasing keypath for fastbuild extract

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs. To avoid this error, abort and retry the reorganization with collection fast build disabled.

3425 Reorg collection FastBuild found duplicate key entry

Cause

This error occurs if collection fast build finds a duplicate key value while building a collection in which duplicates are not allowed.

Action

The duplicate keys are identified with collection and member object identifiers in the reorganization log file. Abort the reorganization, resolve the duplicate keys, and then retry the reorganization. To resolve the duplicate keys:

- Change the member object key values.
- Add a duplicate resolver key element to the collection keys.
- Change the collection to allow duplicates.

3426 Reorg restart requires reorgJournal - restore it or abort the reorg

Cause

This error occurs when the reorganization journal file for the reorganization being restarted has been removed from the current journal directory for the environment. The file name is **reorgjournal_***journal-number***_***file-offset*, in which **journal-number** and **file-offset** refer to the journal LSN of the **startReorg** entry.

Action

If the reorganization journal file can be restored, restore the file and restart the reorganization. If the reorganization journal file cannot be restored, abort the reorganization and then retry the reorganization.

3427 Reorg journal is corrupt - abort the reorg

Cause

This error occurs if the reorganization journal file is corrupt.

Action

Abort the reorganization and then retry it.

3428 An error occurred in a database reorg caretaker operation

Cause

This error occurs when an unexpected problem occurs when stopping and tidying up after a failed reorganization operation

Action

Perform the following actions.

1. If the original problem can be identified and resolved as indicated in the **jomreorg.log** file, then do so.
2. Attempt to restart the reorganization.

Refer to additional details output to the reorganization (**jomreorg.log**) log file.

3500 through 3599 Server Remote Interface Errors

This section describes the errors that can occur during the running of the server remote interface.

3500 Exceeded server maximum connections

Cause

This error occurs if a JADE **initialize** operation fails because the maximum number of client connections for a client workstation is exceeded.

Action

Shut down one or more client applications.

3501 Invalid server connection type

Cause

This error occurs if the JADE client attempts to open a connection to a remote server specifying an invalid or unsupported connection type.

Action

Ensure that the parameters in your JADE initialization file are correct. In particular, check that the *transport-type* value is correctly specified in the [ServerNodeSpecifications](#) parameter in the [\[JadeClient\]](#) section of the JADE initialization file and that it is spelled correctly. A common mistake is the misspelling of **Tcplp**.

If your JADE initialization file is correct and your JADE licenses include support, contact your local JADE support center or JADE Support.

3502 Incorrect handle returned in response

Cause

This error occurs if the connection handle returned in a response from a remote JADE server does not match the connection handle of the client requestor.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3503 Callback connection not supported for this network type

Cause

This error occurs if you attempt to use server methods that have references to transient objects on the client. This is currently supported using TCP/IP connections only.

Action

Use the TCP/IP protocol.

3504 Remote node disconnected

Cause

This error occurs if the network connection has been closed by the remote node.

Action

Terminate the application.

3505 Fatal network error

Cause

This error occurs if a fatal network error occurs.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3506 Remote Procedure Call version mismatch

Cause

This error occurs if a JADE Remote Procedure Call (RPC) version mismatch is detected between the JADE client and the JADE server after a connection has been established. The most-likely cause is that the JADE system libraries have been upgraded on either the client or the server only.

Action

Ensure that the JADE client and the JADE server are running the same version of JADE software.

3507 Client HostData registration failure

Cause

This is an internal error.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3508 Server HostData registration failure

Cause

This is an internal error.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

3509 RPC maximum callLevel exceeded

Cause

This error can be returned if your remote procedure calls between client and server execution get into a loop situation.

Action

If you determine that this is caused by a logic loop, change your application code accordingly.

If this appears not to be the problem and if your JADE licenses include support, contact your local JADE support center or JADE Support.

3510 Invalid network specification

Cause

This error occurs if you specify a network specification parameter that has invalid syntax.

Action

Correct the syntax of the network specification parameter.

3512 Automated upgrade version mismatch

Cause

This error occurs if the JADE system has detected a version mismatch between the database and the fat client (or application server).

Action

Update the JADE client node to the same level as the JADE database server node.

3513 Jade Data Interchange version mismatch

Cause

This error occurs when a JADE Data Interchange version mismatch was detected when a client connected to the server.

Action

Ensure that the JADE client and the JADE server are running the same version of JADE binary files.

3514 Distributed Management version mismatch

Cause

This error occurs if a JADE Distributed Management version mismatch was detected when a client connected to the server.

Action

Ensure that the JADE client and the JADE server are running the same version of JADE binary files.

3517 Reply to management request truncated

Cause

This error occurs if a management request directed to another client node attempts to return a result of more than 50,000 bytes.

Action

Change the request parameters so that it generates a smaller reply

3518 Remote management request failed

Cause

This error occurs if a management request directed to another node could not be successfully processed. The JADE messages (**jommsg.log**) log file for one of the three nodes (source, target, or database) may contain additional details. A common cause for this error is that the target node is terminating.

Action

None.

3519 Large request buffer allocation failed

Cause

This error occurs if an attempt was made to send a large request to the server (for example, executing a **serverExecution** method with a very large **String** or **Binary** parameter) and the server could not allocate a single contiguous piece of memory in which to receive the request.

Action

Make more memory available to the server or change the application to send several smaller requests.

3520 Remote intra process request failed

Cause

This error occurs if an intra-process request directed to a remote node could not be successfully processed. The JADE messages (**jommsg.log**) log file for one of the three nodes (source, target, or database) may contain additional details. A common cause for this error is that the target node is terminating.

Action

None.

3522 Invalid context handle passed to RPC

Cause

An invalid or out-of-date context handle was passed to a RPC API.

Action

Internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

3524 Invalid server URI

Cause

This error occurs if the server URI specified in the command line server parameter is invalid or incomplete.

Action

Retry with a corrected URI.

3525 Invalid environment/server identity

Cause

This error occurs if the environment UUID or the environment and server UUID pair in the server URI are not correctly formatted.

Action

Retry with a corrected URI.

3526 Environment/server identity mismatch

Cause

This error occurs if the environment UUID or the environment and server UUID pair in the server URI does not match that of the server. The correct text is written to the server **jommsg.log** file.

Action

Retry with a corrected URI.

3571 SSPIAuth memory allocation failure

Cause

This error occurs if there is a memory allocation error when a client attempts to authenticate a database server with a database that has been encrypted.

Action

Restart the node.

3572 SSPIAuth state storage not allocated

Cause

This is an internal error in the Windows Security Support Provider Interface (SSPI) implementation and should not normally occur. A possibly cause is the use of unpublished APIs.

Action

Restart the node.

3573 SSPIAuth SSPI error. See jommsg.log

Cause

This error occurs when the Windows Security Support Provider Interface (SSPI) module has returned an error. Information is logged to the JADE messages log file (**jommsg.log**).

Action

If you are unable to resolve the problem and your JADE licenses include support, contact your local JADE support center or JADE Support.

3574 INI entry [Client] SspiAuthServicePrincipalName is required

Cause

This error occurs when a standard client or application server connects to the database server and the database is encrypted but the **SspiAuthServicePrincipalName** parameter in the **[JadeClient]** section of the JADE initialization file is missing. This parameter specified the Service Principal Name (SPN), which is required for authentication.

Action

Add the required parameter to the entry to the JADE initialization file.

3575 SSPIAuth securityPackage does not support required capabilities

Cause

This error occurs if the installed Windows SSPI package does not support the following required capabilities.

- SECPKG_FLAG_INTEGRITY
- SECPKG_FLAG_PRIVACY
- SECPKG_FLAG_IMPERSONATION
- SECPKG_FLAG_MUTUAL_AUTH.

Action

Disable database encryption or use a version of Windows with the required SSPI support.

3578 SSPIAuth encryption error

Cause

This error occurs if an attempt to encrypt a message for RPC fails. Information is logged to the JADE messages log file (**jommsg.log**).

Action

Restart the node to re-establish the RPC encryption keys. If the problem persists and your JADE licenses include support, contact your local JADE support center or JADE Support.

3579 SSPIAuth decryption error

Cause

This error occurs if an attempt to decrypt a message for RPC fails. Information is logged to the JADE messages log file (**jommsg.log**).

Action

Restart the node to re-establish the RPC encryption keys. If the problem persists and your JADE licenses include support, contact your local JADE support center or JADE Support.

4000 through 4099 JADE Language Interpreter Errors

This section describes the errors that can be reported at run time by the JADE Interpreter when executing JADE methods.

4001 Executable code not generated for this method

Cause

This error occurs if the JADE method being invoked has not yet been compiled.

Action

Compile the method in the JADE development environment.

4002 Invalid type for node create

Cause

This error occurs if an invalid parameter is passed to the interpreter when an internal node is being created.

Action

This error is an indication of an internal fault in the JADE interpreter. If this error occurs, contact your local JADE support center or JADE Support if your JADE licenses include support.

4003 Invalid operation code for node create

Cause

This error occurs if an invalid parameter is passed to the interpreter when an internal node is created.

Action

This error is an indication of an internal fault in the JADE interpreter. If this error occurs, contact your local JADE support center or JADE Support if your JADE licenses include support.

4004 Attempt to evaluate a recursive translatable string

Cause

This error occurs if you attempt to define a translatable string in terms of itself, either directly, or indirectly via another translatable string.

Action

Change the translatable string definition to remove the circular reference.

4005 Error loading JADE method

Cause

This error occurs if an error is detected while loading the executable code for a JADE method from the database.

Action

This error is an indication of an internal fault in the JADE interpreter or a corruption in the JADE database. If this error occurs, contact your local JADE support center or JADE Support if your JADE licenses include support.

4006 Cannot evaluate translatable string because it requires compilation

Cause

This error occurs if the translation for the current locale of a translatable string was not saved and compiled after it was changed.

Action

Compile the translatable string by displaying and saving it. You can access the String Browser that enables you to translate strings, by selecting the **Strings** command from the Schema menu.

4007 Cannot load method because it requires recompilation

Cause

This error occurs when a method has been compiled prior to JADE 6.0.16 with Unicode binaries.

Action

Recompile all methods prior to loading the method.

4008 Cannot change or overlay an in-use method

Cause

This error occurs if you attempt to compile a method that is currently executing.

Action

Terminate the activity in your application that is executing the method you need to compile. You do not have to shut down the whole application.

4009 Method is compiled by a future compiler version

Cause

This error occurs if the version of the JADE interpreter being used is incompatible with the version of the compiled JADE code being executed.

Note Information about the method being compiled is written to the JADE messages log file.

Action

Replace the JADE interpreter executable (`jlintp`) with a version that is compatible with the compiled JADE code.

4010 An application is already being debugged

Cause

This error occurs if you attempt to debug an application when an application is already being debugged.

Action

Shut down the JADE debugger on the other application and the application.

4011 Result of expression overflows Byte precision

Cause

This error occurs when the result of a **Byte** primitive type expression is outside the allowed range, which is zero (0) through 255.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

4012 Result of expression overflows Character precision

Cause

This error occurs when the result of a **Character** primitive type expression is outside the allowed range.

This error should not normally occur, as the minimum and maximum values of the **Character** primitive type in JADE are 0 and 255 in ANSI, and 0 and 65535 in Unicode.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

4013 Result of expression overflows Integer precision

Cause

This error occurs when the result of an **Integer** primitive type expression is outside the allowed range. This error should not normally occur, as the minimum and maximum values of the **Integer** primitive type in JADE are -2147483648 and 2147483647.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

4014 Result of expression overflows Integer64 precision

Cause

This error occurs when the result of an **Integer64** primitive type expression is outside the allowed range. This error should not normally occur, as the minimum and maximum values of the **Integer64** primitive type in JADE are - **9223372036854775808** and **9223372036854775807**.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

4015 Too many references to this item

Cause

This error occurs if a variable or parameter is referenced more than 65535 times in a single method.

Action

Rewrite the method as two or more smaller methods.

4016 Result of expression is not a valid Date value

Cause

This error occurs when the result of a **Date** primitive type expression is outside the allowed range. The **Date** type is used to store Julian day numbers.

The valid range for Julian day numbers in JADE is zero (**0**) through **4294967294**.

Action

Change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

4017 Profiling has not been enabled for this Process

Cause

This error occurs if you call the **report** or **reset** method of the **JadeProfiler** class and profiling has not been enabled for this process by calling the **start** method.

Action

Correct your application code.

4019 Parameter type mismatch in external method

Cause

This error occurs when a JADE method is invoked from an external method and one of the following conditions applies.

- The number of parameters passed by the external method does not match the signature of the method that is defined in the schema.
- The types (formats) of parameters passed by the external method do not match the signature of the method that is defined in the schema.

Action

Locate the call to the JADE method that is at fault in your external method, and correct it so that it passes the correct number and types of parameters that are defined in the method signature in your schema.

4020 Return type mismatch in external method

Cause

This error occurs when a JADE method is invoked from an external method and one of the following conditions applies.

- The **return** value parameter is missing.
- The method primitive type is string or binary and the format of the **return** value parameter is not suitable to receive the contents of a string or a binary value.

Action

Locate the call to the JADE method that is at fault in your external method, and correct it so that it passes the correctly typed parameter for the return value defined in the method signature in your schema.

4021 Cannot execute JADE method because it requires compilation

Cause

This error occurs if the JADE method being invoked encountered an error in its last compile.

Action

Correct the error in your code, and recompile the method in the JADE development environment.

4022 Illegal operation attempted

Cause

This error occurs if the execution of a JADE instruction requires an operation that is not valid or not supported by the JADE interpreter. This would normally occur only in the following cases:

- The operation is syntactically valid but has invalid semantics in the context.
- The operation is a valid language operation but is not implemented by the interpreter.

In either case, the error is a result of a problem in the JADE compiler or the interpreter.

Action

If this error is reported for some unforeseen circumstance, contact your local JADE support center or JADE Support if your JADE licenses include support.

4023 Code Coverage has not been enabled for this Process

Cause

This error occurs if you call the [reportCodeCoverage](#) method of the [JadeProfiler](#) class and code coverage has not been enabled for this process by calling the [startCodeCoverage](#) method.

Action

Correct your application code.

4024 Out of resources for operation

Cause

This error usually occurs if your system is out of memory.

This error could be caused by an external function that has been called where the number or types of the parameters in the external function definition do not match the parameters expected by the function. In this situation, it is possible that the incorrect parameter definition has caused the external function to corrupt memory.

Action

Check that the definition of any external function in JADE exactly matches the function in the DLL. If this is not the case, you may need to shut down one or more applications to free up memory. You may also need to increase the size of the virtual memory page file or install more physical memory to prevent this error from occurring again.

4025 Power operator argument is invalid

Cause

This error occurs if the left operand of the exponentiation operator (^) is a [Real](#) primitive type and is less than 0 (zero) and the right operand is not a whole number, or if the left operand is zero (0) and the right operand is negative.

Action

Change your application code to ensure that the exponentiation operator argument is valid.

4026 Too many parameters

Cause

This error occurs if you attempt to define a method or external function with more than the maximum allowable number of parameters. This maximum is 129 parameters.

Action

Change the method or external function definition so that there are not more than 129 parameters.

4027 Method called with incorrect number of parameters

Cause

This error occurs when a JADE method is invoked (called) from an external method and the number of parameters passed by the external method does not match the signature of the method defined in the schema.

Action

Locate the method call that is at fault in your external method, and change it so that it passes the correct number of parameters defined in the method signature in your schema.

4028 The execution stack is broken

Cause

This error occurs when there is an internal fault in the JADE interpreter.

Action

If this error occurs, contact your local JADE support center or JADE Support if your JADE licenses include support.

4029 Illegal type conversion attempted

Cause

This error occurs if an illegal type conversion is encountered. For example, you cannot assign an **Integer** value to an **Any** variable and then cast the **Any** variable to a **TimeStamp**. Direct **Integer** to **TimeStamp** conversions are not supported in JADE and the compiler reports a syntax error.

Action

Change your application code to ensure that it does not attempt an illegal type conversion.

4030 Division by zero

Cause

This error occurs if a JADE instruction contains an arithmetic expression that results in a division by zero when evaluated.

Action

Determine the cause of a zero divisor and then correct the code that led to the error. The method and source position containing the arithmetic expression at which the exception was raised are reported in the error description.

4031 Index used in substring is out of bounds

Cause

This error occurs when the starting position in a substring operation is less than 1, or the starting position is greater than the string size when the substring expression is being read (not being assigned to).

Action

Determine the cause for the invalid argument in the substring expression and then correct the code that led to the error. The method and source position containing the substring expression at which the exception was raised are reported in the error description.

4032 Assignment of non-character to indexed string variable

Cause

This error occurs if an attempt is made to assign a non-character value to a subscripted string variable.

Action

This type of invalid assignment should be caught by the JADE compiler and is not expected to occur at run time. If this error occurs, contact your local JADE support center or JADE Support if your JADE licenses include support.

To avoid the problem, determine the cause of the invalid assignment and then correct the code that led to the error. The method and source position containing the assignment at which the exception was raised are reported in the error description.

4033 Result of expression overflows Decimal precision

Cause

This error occurs if the result of the expression is too large to fit in the specified precision.

Action

Change your application code to increase the precision of the decimal value. For details, see the [Decimal](#) primitive type. The method and source position containing the assignment at which the exception was raised are reported in the error description.

4034 Unimplemented operation attempted

Cause

This error occurs if the execution of a JADE instruction requires an operation that is not valid or not supported by the JADE interpreter. This would normally occur only in the following cases:

- The operation is syntactically valid but has invalid semantics in the context.
- The operation is a valid language operation but is not implemented by the interpreter.

In either case, the error is a result of a problem in the JADE compiler or the interpreter.

Action

If this error occurs for some unforeseen circumstance, contact your local JADE support center or JADE Support if your JADE licenses include support.

4035 User interrupted method execution

Cause

This error occurs when you interrupt the execution of an iterative operation in a JADE method (for example, a **while** or **foreach** instruction) by using the Jade User Interrupt utility.

Action

Use the error information to determine the location of the loop. The method and source position containing the instruction loop that was interrupted are reported in the error description.

4036 External library file not found

Cause

This error occurs if the external library file cannot be found.

Action

Ensure that the DLL declared as containing the external function is in the current directory or in a directory contained in the current path.

4037 External function not found in library

Cause

This error occurs if a required external function is not found in the external library.

Action

Check that the declaration of the external function corresponds to the name of the function exported from the DLL.

4038 External function caused an exception

Cause

This error occurs if an external function causes an exception.

Action

The external function being called has caused a fatal error; for example, a stack corruption or a General Protection Fault (GPF).

Check that the number and type of all parameters in the function declaration correspond with the external implementation. Check the external function for coding errors.

4039 Result of expression overflows Real precision

Cause

This error occurs when the result of a **Real** primitive type expression is outside of the allowable range. This error should not normally occur, as the maximum and minimum values of the **Real** primitive type in JADE fall between approximately $\pm 1.7 \times 10^{308}$.

Note that for complete accuracy, the **Real** primitive type can be relied on only to fifteen significant digits.

Action

Change your code to avoid this expression. The method and source position containing the expression at which the exception was raised are reported in the error description.

4040 Profiling has already been enabled

Cause

This error occurs if there is an attempt to enable profiling when profiling is already enabled. Multiple profilers are not allowed.

Action

Change your application code so that if profiling is already enabled it does not create a new profiler.

4041 Profiler dispatch error

Cause

This error occurs if there is an attempt to call a JADE profiler method in a server execution method. This is not allowed.

Action

Change your JADE profiler usage to make sure that all references are on the client.

4042 Decimal conversion error

Cause

This error occurs if there is an attempt to convert a non-**Decimal** value to a **Decimal**. For example, this error could be returned when converting a **Binary** value to a **Decimal** where the **Binary** does not represent a **Decimal** value.

Action

Correct the application code that is using the **Decimal** primitive type-cast operation.

4043 Result of expression overflows Decimal precision

Cause

This error occurs when a decimal overflow condition occurs if the **truncateOnDecimalOverflow** method of the **Process** class is called with a parameter of **true**.

This error differs from error [4033 - Result of expression overflows decimal precision](#) in that this error is continuable.

If an exception handler continues this exception, the decimal number is truncated and execution continues. For example, the value **123.456** is truncated to **23.46** when assigned to a **Decimal** [4, 2].

Action

For an example of using this continuable exception, consider the following code.

```
vars
  d : Decimal[4,2];
begin
  on Exception do e0(exception);
  process.truncateOnDecimalOverflow(true);
  d := 123.456;
  write d;
  d := -123.456;
  write d;
end;
```

In the above method, **e0** is defined as follows.

```
e0(e : Exception): Integer;
begin
  // exception 4043 is a 'continuable' decimal overflow
  if e.errorCode = 4043 then
    // continuing here will result in the value being truncated
    return Ex_Continue;
  endif;
  return Ex_Resume_Next;
end;
```

The output of the **write** instruction using these examples is as follows.

```
23.46
-23.46
```

Note It is the integral part of the decimal that is truncated. The fractional part is rounded.

Note also that the extended error text of the exception contains the value of the decimal *before* it is truncated. The following method is an example of an exception handler.

```
e0(e : Exception): Integer;
begin
    // exception 4043 is a 'continuable' decimal overflow
    if e.errorCode = 4043 then
        write e.extendedErrorText;
        // continuing here will result in the value being truncated
        return Ex_Continue;
    endif;
    return Ex_Resume_Next;
end;
```

The output from the above method is as follows.

```
123.456
23.46
-123.456
-23.46
```

4500 through 4599 JADE Query Engine Messages

Errors in the range 4500 through 4599 can be reported through the JADE Query Engine, and are for internal use only.

If you receive any other errors in the range 4500 through 4599, contact your local JADE support center or JADE Support if your JADE licenses include support.

4520 Data has changed edition while building the query

Cause

This error occurs when data being used in the report changes while the report is running, when the **Repeatable Read Optimistic** option is selected.

Action

Rerun the report when the data is not being updated.

5000 through 5099 File Handling Errors

This section describes the errors that can occur during file handling; that is, while opening, closing, or accessing files.

5001 Cannot open file

Cause

This error occurs if the file open operation was denied by the operating system for a reason other than an access violation.

Action

Determine the root cause of this error from your JADE messages (**jommsg.log**) log file, specifically the description reported by the operating system.

5002 Requested access to file not permitted

Cause

This error occurs if the file is not accessible with the selected mode and sharing attributes. This usually indicates that another process on the same workstation, or elsewhere in the network, already has the file open with an incompatible sharing mode.

Action

Provide an appropriate exception handler to shield users from this error.

5003 Requested file not found

Cause

This error occurs if an attempt to open a file with a mode of **File.Mode_Input** or **File.Mode_IO** fails to locate the file and the **allowCreate** attribute of the **FileNode** class is set to **false**.

Action

Provide an appropriate exception handler to shield users from this error.

5004 File read operation incomplete

Cause

This error occurs if the file read operation is incomplete. A file read operation failed to transfer the requested amount of data from the physical disk file.

This error will not occur when a sequential read reaches or attempts to read beyond the end of file, which sets the file status to end-of-file. The error is therefore an indication that the file has been corrupted or a hardware error occurred on the device containing the file.

Action

Use a disk verification utility to verify the file or the device containing the file (or both).

5005 File write operation incomplete

Cause

This error occurs when a file `writeBinary`, `writeLine`, or `writeString` operation fails to transfer all of the requested data to the physical disk file.

The most-common cause of this error is an "out of disk space" situation. However, physical hardware failures could also cause the same error condition.

Action

Correct the problem of insufficient disk space, or if this is not the problem, check for a hardware failure.

5006 Cannot set file pointer

Cause

This error is an internal error and may indicate that there is a corruption of the file information block used by JADE.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

5007 Required file is not currently open

Cause

This error occurs if JADE attempts to access a file that is not currently open.

Action

Change your application code to ensure that files are opened before any access calls.

5008 Cannot overwrite existing file

Cause

This error occurs if an attempt is made to overwrite an existing file when opening a file with a mode set to `File.Mode_Output` and the `allowReplace` attribute of the `File` class is set to `false`.

The `allowReplace` attribute is intended to prevent the accidental overwriting of existing files.

Action

Determine if the existing file is still required. If it is required, move, rename, or back up and remove the file, and then repeat the operation. If it is not required, remove the file and repeat the operation. Alternatively, provide an appropriate exception handler to shield users from this error.

5009 Cannot read past the end of file

Cause

This error occurs if there is an attempt to read beyond the end of a file. If the file was at the end before the read, no data will have been read. If the file was not at the end before the read, data up to the end of the file will have been read.

Action

Change your application code so that it does not attempt to read past the end of the file. Do this by checking for end-of-file before reading and by ensuring that you do not attempt to read records larger than the difference between the length of the file and the current file offset.

5010 Read/write operation conflicts with file mode

Cause

This error occurs when a file operation conflicts with the current mode of the file; for example, if an attempt is made to write to a file opened with the mode set to **File.Mode_Input** or to read from a file opened with the mode set to **File.Mode_Output**.

Action

Change your application code to avoid these conflicts. The method and source position containing the file operation at which the exception was raised are reported in the error description.

5011 Record truncated to maxRecordSize characters

Cause

This error occurs when a **readLine** operation is performed and the line read is found to be larger than the **maxRecordSize** property that was set when the file was opened. The line is truncated to the size specified by the **maxRecordSize** property.

This could be caused, for example, when reading in a Unicode text file when your system expects an ANSI file.

Action

Increase the value of the **maxRecordSize** property of the **File** object or if you are reading in a Unicode text file, add the following line after the **create file transient**; line in your method or JadeScript.

```
file.kind := File.Kind_Unknown_Text; // works for ANSI and Unicode text files
```

5012 Unicode to Ansi conversion failed

Cause

This error occurs during the conversion of data from Unicode to ANSI. This error can occur when reading a Unicode text file into an ANSI JADE application or when writing an ANSI file from a Unicode JADE application.

This error also occurs if you attempt to load Unicode schema files into an ANSI JADE environment; this type of schema load is not supported.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

5013 Ansi to Unicode conversion failed

Cause

This error occurs during the conversion of data from ANSI to Unicode. This error can occur when reading an ANSI text file into a Unicode JADE application or when writing a Unicode file from an ANSI JADE application.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

5014 Cannot rename a file to a different device

Cause

This error occurs if there is an attempt to rename a file to a different device.

Files are renamed only to the same device; for example, if you attempt to rename **c:\myfile.txt** to **a:\newfile.txt**. You cannot rename **myfile.txt** from the **c:** drive to the **a:** drive.

Action

Change your application code so that files are not renamed to different devices.

5015 Attempt to rename a file failed

Cause

This error occurs if an attempt to rename a file failed. The most-common reasons for a rename failure are:

- The path or file name was not found.
- Access to the file was denied.

An attempt to rename a file to another device could fail for the following reasons. (often related to an operating system restriction).

- File name on output device is invalid.
- Permission to create a file on the output device is denied.
- Space on the output device is insufficient.

Action

Refer to your JADE messages (**jommsg.log**) log file where the operating system error is written. This can be useful in determining exactly why the rename operation failed.

5016 Attempt to delete a file failed

Cause

This error occurs if an attempt to delete a file failed. The most-common reasons for a delete file failure are:

- The path or file name was not found
- Access to the file was denied

This error occurs if a delete file attempt failed for a reason other than these.

Action

Refer to your JADE messages (**jommsg.log**) log file where the operating system error is written. This can be useful in determining exactly why the delete operation failed.

5017 Cannot perform a binary read on a file opened as text

Cause

This error occurs if your application code attempts to perform a **readBinary** operation on a file opened as ANSI text or Unicode text.

Action

Change your application code so that binary reads are not done on text files.

5018 Cannot perform a binary write on a file opened as text

Cause

This error occurs if your application code attempts to perform a **writeBinary** operation on a file opened as ANSI text or Unicode text.

Action

Change your application code so that binary writes are not done on text files.

5019 Cannot perform a string read on a file opened as binary

Cause

This error occurs if your application code attempts to perform a **readString** or **readLine** operation on a file opened as binary.

Action

Change your application code so that text reads are not done on binary files.

5020 Cannot perform a string write on a file opened as binary

Cause

This error occurs if your application code attempts to perform a `writeString` or `writeLine` operation on a file opened as binary.

Action

Change your application code so that text writes are not done on binary files.

5021 Cannot open a file with an invalid kind property

Cause

This error occurs if your application code attempts to open a file with its `kind` property not set to one of the `Kind_` constants defined in the `File` class.

Action

Change your application code so that the `kind` property is set to one of the constants defined in the `File` class before the file is opened.

5022 File cannot be opened as Unicode as it contains an odd number of bytes

Cause

This error occurs if your application code attempts to open a file containing an odd number of bytes as Unicode.

As all Unicode characters occupy two bytes, a file containing an odd number of bytes cannot be a Unicode file.

Action

Change your application code to open the file as `File.Kind_Binary` or `File.Kind_ANSI`.

5023 Directory with the specified name could not be created

Cause

This error occurs if the create directory operation is denied by the operating system for a reason other than an access violation.

Action

Determine the root cause of this error from your JADE messages (`jommsg.log`) log file, specifically the description reported by the operating system.

5024 Requested file directory was not found

Cause

This error occurs if the operating system cannot locate the requested directory. The most-common reasons for a directory not being found are:

- The path or directory name is not found
- Access to the file is denied

This error occurs if the failure is for a reason other than these.

Action

Examine your JADE messages (**jommsg.log**) log file where the operating system error is written. This can be useful in determining exactly why the locate operation failed.

5025 Specified directory could not be removed

Cause

This error occurs if an attempt to remove a directory fails. The most-common reasons for a delete directory failure are:

- The path or directory name is not found
- Access to the file is denied

This error occurs if the directory operation fails for a reason other than these.

Action

Refer to your JADE messages (**jommsg.log**) log file, where the operating system error is written. This can be useful in determining exactly why the remove operation failed.

5026 File already open

Cause

This error occurs if your application code attempts to open a file that is already open.

Action

Change your application code to prevent this.

5027 Requested file directory name is not valid

Cause

This error occurs if a file directory that you have specified is invalid, or the drive letter used in the path does not exist.

Action

Refer to your JADE messages (**jommsg.log**) log file for information about the file and then correct the file definition.

5028 Requested file directory not found

Cause

This error occurs if a file directory that you have specified does not exist.

Action

Refer to your JADE messages (**jommsg.log**) log file for information about the file and then correct the file definition.

5029 File name not valid for file system

Cause

This error occurs if you have specified a file or directory name that is not supported by the file system hosting the file. These characters are reserved and have special meanings.

Action

Change the file names to comply with file system constraints, or use a file system that supports the file names you want to use.

5030 File is in use by another process

Cause

This error occurs if you attempt to open a file that is already open by another process.

Action

Refer to the JADE messages log file (**jommsg.log**) for information about the file. Generally, another program is accessing the file.

5031 A file write encountered an out of disk condition

Cause

This error occurs if a physical write operation to disk was terminated prematurely due to an out-of-disk situation.

Action

Obtain sufficient disk space for the file to grow and then resubmit the failed operation.

5032 Error encountered closing file

Cause

This error occurs if the operating system encounters an error when closing a file.

Action

Refer to your JADE messages (**jommsg.log**) log file for a description of the error reported and then take any appropriate action.

5033 An unexpected I/O error occurred accessing file

Cause

This error is raised by the operating system on finding an error on the last read or write operation of a file. This can occur for several reasons, including a hardware device error, file corruption, security issue, or software problem.

Action

Refer to your JADE messages (**jommsg.log**) log file for a description of the error reported and then take any appropriate action.

5034 Attempt to copy a file failed

Cause

This error occurs when an attempt to copy a file fails. This can occur for several reasons, including a hardware device error, file corruption, security issue, or software problem.

Action

Check that your program has sufficient security rights to read the input file and has sufficient security for writing the output file. In addition, check that there is sufficient space for the data required on the output file. If these actions do not resolve the problem, refer to your JADE messages (**jommsg.log**) log file for a description of the error reported and take any appropriate action.

5035 Cannot access a file opened by a different Node

Cause

This error occurs when an attempt is made to access a file on a different node to the one that opened the file; for example, when a file is opened on a client node and a server method attempts to read or write to the file, this error is raised.

Action

Change your code to ensure that when file objects are employed on different nodes, the file is opened on the same node that performs file I/O operations.

5036 Cannot open file, the kind property is different to actual file kind

Cause

This error occurs when an attempt is made to open a file where the **kind** property of the **File** class has been set to a value such as **Kind_Unicode_UTF16LE** and the actual Byte Order Mark (BOM) header of the file does not match the requested kind.

Action

Check the **kind** property against the actual kind of the file. Specify the actual file kind, or use a more generic kind such as **Kind_Unicode** or **Kind_Unknown_Text**, as appropriate.

5038 Unicode UTF8 file contains illegal byte sequence

Cause

This error occurs when the input Unicode file in UTF8 format does not contain characters that are legal UTF8 characters.

This is most-likely caused by specifying a file kind of **Kind_Unicode_UTF8**, when the actual file is not a true UTF8 format file.

Action

Check the validity of the file. This error may indicate a corrupted file. Use a non-JADE tool to find and correct the illegal UTF8 character sequence.

5039 Unicode UTF8 file kind not supported

Cause

This error occurs when an attempt is made to load a schema (**.scm**) or form and data definition (**.ddb** or **.ddx**) file in Unicode UTF8 format. JADE **.scm** and **.ddb** or **.ddx** files must be in either ANSI or Unicode UTF16 format.

Action

Convert the schema file into a standard format supported by JADE (that is, either ANSI or Unicode UTF16) and then reload the schema.

5040 Insufficient system resources

Cause

This error can be raised by JADE file classes if the operating system refuses to service a file system request due to constrained resources. In addition, see error [3111 - Insufficient system resources](#).

Action

Refer to your JADE messages (**jommsg.log**) log file for the value of the operating system error code. It is possible that the system is under-configured for the applied workload. If you consider the operating environment to be sound with adequate memory resources, refer the problem to your operating system vendor, as there may be a resolution available in a service pack. There have been instances where operating system bugs have caused such errors.

5045 Broken symbolic link

Cause

This error occurs when the **open** method of the **File** class is executed on a broken symbolic link (symlink) file; a broken or orphan symbolic link is one that refers to a file that no longer exists.

Note Symbolic link files, whether valid or broken, are contained in the array of files returned by the **files** method of the **FileFolder** class.

Action

Use exception handling to cater for the possibility that a symbolic link file could be broken.

5047 Invalid record size

Cause

This error occurs if a presentation client attempts to read or write more than 2G bytes of data in a single read or write statement and the value of the **FileNode** class **usePresentationFileSystem** property is set to **true**. It also occurs if the length of data is less than zero (**0**).

Action

Read or write data in smaller chunks.

5100 through 5299 Multimedia Handling Errors

This section describes errors that can be returned when using multimedia files.

5101 Error accessing multimedia file

Cause

This error occurs when reading a multimedia file that is not a valid wave sound (.wav) or video (.avi) file, or a multimedia file that has become corrupted.

Action

Attempt to play the file with a media player utility, to determine if the file is valid or corrupted.

5102 Error accessing multimedia device

Cause

This error is raised by the operating system if a problem is encountered when a multimedia device is being used to play back a sound or video file.

Action

The error code and description reported by the operating system are recorded in your JADE messages (jommsg.log) log file. This information can be useful in determining exactly why the multimedia playback failed.

5200 Out of resources for operation

Cause

This error occurs if the operating system has insufficient resource (for example, memory) to complete the operation.

Action

Allocate more resource.

5201 Thread allocation failed

Cause

This error usually indicates a memory problem.

Action

Restart your system.

If the fault persists, contact your local JADE support center or JADE Support if your JADE licenses include support.

5202 Thread start failed

Cause

This error usually indicates a memory problem.

Action

Restart your system.

If the fault persists, contact your local JADE support center or JADE Support if your JADE licenses include support.

5203 Thread create failed

Cause

This error usually indicates a memory problem.

Action

Restart your system.

If the fault persists, contact your local JADE support center or JADE Support if your JADE licenses include support.

5300 through 5399 Sort Handling Errors

This section describes the errors that can be returned during sorting.

5301 Sort target file not present

Cause

This error occurs if the sort target file cannot be opened.

Action

Arm an exception handler that is triggered in the event the target file is already open.

5302 Sort temporary file not created

Cause

This error occurs when a temporary file used to sort extract data fails to be created.

Action

Check that the environmental **TEMP** variable is correctly defined and that it points to a directory with create permission.

5303 Sort temporary file write failed

Cause

This error occurs when a **write** operation on the sorted file has failed.

Action

Check that there is sufficient scratch space in the directory defined by the **TEMP** environmental variable.

5304 Sort temporary file close failed

Cause

This error occurs when a **close** operation on the sorted file has failed.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

5305 Sort negative offsets not allowed

Cause

This error occurs if a negative offset is encountered in a **SortActor** member of the **SortActorArray** passed in the **extractSort** method of the **File** class.

Action

Change your application code so that it does not use negative offsets in sort actors.

5306 Sort negative offsets not allowed

Cause

This error occurs if a negative offset is encountered in a **SortActor** member of the **SortActorArray** passed in the **extractSort** method of the **File** class.

Action

Change your application code so that it does not use negative field numbers in sort actors.

5307 Only one record found in the sort file buffer

Cause

This error occurs if one record only exists within the source sort file.

Action

Check that the **endOfLine** delimiters are correct.

5308 Call to GetTempPath failed in JomSort

Cause

This error occurs when the sort could not find the directory defined by the **TEMP** environmental variable.

Action

Check that the **TEMP** environmental variable is correctly defined and that it points to a directory with **create** permission.

5309 Call to GetTempFileName failed in JomSort

Cause

This error occurs if the sort could not create a temporary file.

Action

Check that the **TEMP** environmental variable is correctly defined and that it points to a directory with **create** permission.

In addition, check that the [SortDirectory](#) parameter in the [\[JadeExtractSort\]](#) section of the JADE initialization file is correctly defined and that it points to a directory with **create** permission.

5310 Cannot set the IO Buffer

Cause

This error occurs if sort buffers could not be allocated.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

5311 Cannot access File IO

Cause

This error occurs if a call to get information about the sort buffers fails.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

5312 IO block failed to advance

Cause

This error occurs if a call to advance the sort buffers fails.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

5313 UNICODE is not currently supported with extractSort

Cause

This error occurs if you attempt to sort a Unicode file.

This is not currently supported under the [extractSort](#) method of the [File](#) class.

Action

Use ANSI files.

5314 Move sorted file to output failed

Cause

This error occurs if the call to move the sorted file failed.

Action

Check for the existence and access rights of the target file.

5315 Cannot use SortActorArray without any elements

Cause

This error occurs if you attempt to use a [SortActorArray](#) class without any elements.

Action

Ensure that you use the required elements in conjunction with the **SortActorArray** class.

5316 Overflow or underflow of a numeric sort field

Cause

This error occurs if you attempt to define a numeric field in an external sort file with a value outside the valid range **-2,147,483,647** through **2,147,483,647** or a JADE [Decimal](#) of more than 23 characters (including sign, decimal point, and digits).

Action

Review your sort parameters, making sure that numeric values do not fall outside the range **-2,147,483,647** through **2,147,483,647** and **Decimal** fields do not exceed 23 characters (including sign, decimal point, and digits).

5317 Sort input/output files must be on appServer

Cause

This error occurs if you are using JADE on a thin client and the input and output of your sort are not on the application server.

Action

Ensure that your input and output are on the application server when using the thin client.

5318 Sort input file must be multiple of recordsize

Cause

This error occurs if you have specified a record size and the input file size is not a multiple of the record size.

Action

Check that you have the correct input file or that it is not a corrupt file.

5319 Last record missing end-of-line delimiter

Cause

This error occurs if the last record is missing the end-of-line delimiter.

Action

Add the end-of-line delimiter.

5320 Unable to process all input in the file

Cause

This error occurs if the input file passed to `extractSort` method of the `File` class contains a null character (hexadecimal `00`).

Action

Remove the null characters from the file, and then try again.

5500 through 5599 License Errors

This section describes the licensing errors that can be returned when you are installing JADE.

5500 Registered license name is invalid

Cause

This error occurs you specify a license name that does not match your license registration.

Action

Enter your license name again. It must match the name for which your JADE software is licensed. If the error still occurs, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5501 License key is invalid

Cause

This error occurs if the license key that you specify does not match your license registration.

Action

Enter your license again. It must match the license key that was supplied with your JADE software. If the error still occurs, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5502 License key is incomplete

Cause

This error occurs if you specify a license key that does not match your license registration because you have not entered the complete license key. The error can also occur if an SDS or RPS system is started and the **DatabaseSubrole** parameter in the [SyncDbService] of the JADE initialization file is incorrectly specified.

Action

Your JADE messages (**jommsg.log**) log file may provide information that helps you to resolve the error.

Enter your license key again. It must match the license key that was supplied with your JADE software. If the error still occurs, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5503 You have exceeded the number of Development Licenses

Cause

This error occurs if you attempt to start or run more copies of the JADE development environment than are permitted by the terms of your license.

Action

If you need to increase the number of JADE development environments permitted under your application license, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5504 You have exceeded the number of Process Licenses

Cause

This error occurs if you attempt to start or run more JADE processes than the maximum number of processes permitted by the terms of your license.

For a detailed description of JADE processes, see "[JADE Object Manager Distributed Processing](#)", in Chapter 1 of the *JADE Object Manager Guide*. For details about registering your JADE license, see "[Reregistering JADE with a New License](#)", in Chapter 1 of the *JADE Initialization File Reference*.

Action

If you need to increase the number of processes permitted under your application license, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5505 This is an unregistered copy

Cause

This error occurs if you attempt to run an SDS or RPS node without first entering the specific SDS or RPS license key.

Action

If you want to purchase a JADE license or you believe there is a problem with your existing license, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5506 The expiry date of this copy has passed

Cause

This error occurs if you attempt to use JADE after the expiry date of its license.

Action

If you want to purchase a JADE license or you believe there is a problem with your existing license, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5507 A Primary Database license key is required for installation

Cause

This error occurs if you attempt to use a secondary database license key to register a primary database.

Action

Check that you are using the correct license key. If you believe there is a problem with your existing license, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5510 Free Developer License cannot be used to register this system

Cause

This error occurs if you attempt to apply a Free Developer License to a system that cannot be registered with a Free Developer License.

Action

Register the system with a Full license or an Enterprise license.

5512 An unknown error occurred while trying to register the database

Cause

This error is most-likely returned by a file error.

Action

Check that you have the correct file access permissions for the database that you are trying to register. If the problem persists, contact your local JADE support center or JADE Support if your JADE licenses include support.

5513 One or more of the parameters specified was invalid

Cause

This error occurs if one or more of the specified registration parameters is invalid.

Action

Check the parameters and correct any that are invalid.

5517 License does not match version of Jade

Cause

This error occurs if the license used is for a different version of JADE from the one being run.

Action

Obtain a license for the current version of JADE. If you believe there is a problem with your current license, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5518 Non-compact license can not be used on compact device

Cause

This error occurs if you attempt to run a JADE application on a compact device, or the compact desktop version, using a license for non-compact platforms. The license used to run JADE on a compact device is different from the license required for a non-compact device.

Action

Obtain a license for the compact version of JADE. If you believe there is a problem with your current license, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5519 Compact license can not be used on non-compact device

Cause

This error occurs if you try to run a JADE application on a non-compact platform using a license for the compact JADE platform. The license used to run JADE on a non-compact device is different from the license required for a compact device.

Action

Obtain a license for the non-compact version of JADE. If you believe there is a problem with your current license, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

5520 Invalid License Type

Cause

This error occurs if a licensing module attempts to restore or retrieve a license with an unsupported license type value.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5700 through 5799 JADE Object Manager Data Interchange Errors

This section describes the errors that can be returned in the JADE Object Manager during data interchange.

5700 No memory for Object Description

Cause

This error indicates a data interchange problem.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5701 Jom DataInterchange Invalid ClassId

Cause

This error indicates a data interchange problem.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5702 Jom DataInterchange Invalid Host Description Handle

Cause

This error indicates a data interchange problem.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5703 Jom DataInterchange Invalid Data Description Handle

Cause

This error indicates a data interchange problem.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5704 Jom DataInterchange not array definition

Cause

This error indicates a data interchange problem.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5705 Jom DataInterchange generic conversion error, see jommsg.log

Cause

This error occurs if a problem occurs that cannot be handled automatically when converting data between heterogeneous environments (or between ANSI and Unicode formats during a database conversion).

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

5706 Jom DataInterchange Invalid codepage translation

Cause

This error occurs when performing a database conversion between ANSI and Unicode formats and a character is encountered that is not valid within the selected codepage.

Action

Select a new codepage when converting your database or alter the character to be valid within the current codepage.

5707 Jom DataInterchange String size decreased

Cause

This error occurs when converting string data between heterogeneous environments (or between ANSI and Unicode formats during a database conversion) and the size of the string has become smaller. This can occur when multi-byte sequences are translated to a single Unicode character or when Unicode surrogates are encountered.

Action

No action should be required, as JADE should handle this situation automatically.

If the problem stops your system running, report this to JADE support.

5708 Jom DataInterchange String buffer too small

Cause

This error occurs when converting string data between heterogeneous environments (or between ANSI and Unicode formats during a database conversion) and the size of the string has become too large to fit inside the fixed-length string. This can occur when a Unicode character is translated to multi-byte sequence during a database conversion from Unicode to ANSI or when Unicode surrogates are encountered across heterogeneous environments.

Action

Increase the maximum size of the fixed length string property or decrease the amount of data stored in the fixed length string property.

5709 Jom DataInterchange Object has multiple problems with strings

Cause

This error occurs when a combination of error [5706 - Jom DataInterchange Invalid codepage translation](#) and error [5708 - Jom DataInterchange String buffer too small](#) occurs.

Action

Review actions of both exceptions and if the problem still occurs and your JADE licenses include support, contact your local JADE support center or JADE Support.

5710 Jom DataInterchange Internal detection of possible translation failure

Cause

This is an internal error in which a character is not valid within the current codepage and there has been a NULL (0 byte) character before the invalid character.

If the string is a fixed-length property, this is not an error. However, if the string is variable-length (that is, a slob), the error is converted to a 5706 error because NULL characters are allowed in slob, whereas in fixed-length string properties, a NULL terminates the string.

Action

Although this error should not occur, if it does so and your JADE licenses include support, contact your local JADE support center or JADE Support.

5711 Jom DataInterchange Buffer too small

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

5900 through 5999 JADE Object Manager Internal Management Message Exchange Errors

This section describes the errors that can be returned in the JADE Object Manager during message exchange.

5900 MgmtMsg Invalid request group

Cause

This error indicates a problem with the internal management message exchange.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5901 MgmtMsg Invalid request

Cause

This error indicates a problem with the internal management message exchange.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5902 MgmtMsg data size too big

Cause

This error indicates a problem with the internal management message exchange.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5903 MgmtMsg Invalid priority type

Cause

This error indicates a problem with the internal management message exchange.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5904 MgmtMsg Invalid response type

Cause

This error indicates a problem with the internal management message exchange.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5905 MgmtMsg Target node not found

Cause

This error indicates a problem with the internal management message exchange.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5906 MgmtMsg Target node no ServerBgThread

Cause

This error indicates a problem with the internal management message exchange.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

5907 MgmtMsg limited to server node

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

6000 through 6999 JADE Compiler Errors

This section describes the errors that can be returned during compilation.

6000 Could not open file

Cause

This error occurs if the compiler is unable to open the input file specified for a schema load.

Action

Check that the file exists in the specified location. If the file is on a Local Area Network (LAN) server, you should also check that you have the necessary access rights to open the file.

6001 Error while reading file

Cause

This error occurs if a problem is encountered while reading from the input file in a schema load. The error may have been caused by some other application corrupting or deleting the file during the load.

If the file is on a LAN server volume, the LAN connection may have been lost during the load.

Action

Check that the file is still present and has not been corrupted. If the file is on a server LAN, check that the LAN connection is still valid and that the file is accessible.

6002 Memory allocation failed

Cause

This error occurs if the compiler is unable to allocate enough memory to read in the input file for a schema load.

Action

Close down other applications in order to free up more memory. You may also have to increase the amount of virtual memory available to Windows. Do this by using the **System** application from the Windows Control Panel.

The problem can also be resolved by dividing the schema file into several smaller units.

6003 Cannot load entry point

Cause

This error occurs if the compiler is unable to load and execute an internal entry point. The cause of this is likely to be that incompatible releases of the JADE compiler DLLs have been installed.

Action

Check that the three JADE compiler DLLs (**jlnctact.dll**, **jlncomp.dll**, and **jlnctab.dll**) have the same date and time and all originate from the same release of JADE.

6004 Invalid token

Cause

This error occurs if your source code contains a token that is not valid in the JADE language; for example, "~", "!", "@", "%", or "?" are not part of the defined JADE syntax.

Action

Remove the invalid token from your source code.

6005 Invalid number

Cause

This error occurs when the input contains a literal number that is invalid.

A number in JADE must be in the following format.

`[+|-] nnn[.nnn]`

If a number contains two or more decimal points or contains non-numeric characters, this error will result.

Action

Change the number in your source code. If you attempt to define a hexadecimal number, you should precede the number with a number, or hash, sign (#).

6006 Invalid timestamp

Cause

This error occurs if the input contains a timestamp that is not valid. A timestamp in JADE must be the following format.

`yyyy:MM:dd:hh:mm[:ss[.mmm]]`

The timestamp must also represent a valid date and time. This error occurs if these conditions are not met.

Action

Correct the timestamp in your source code so that it has a valid format and represents a valid date and time.

6007 Token too long

Cause

This error occurs if an identifier or number exceeds the maximum allowed length.

In JADE, all identifiers are limited to a length of 100 characters and numbers are limited to a length of 30 digits. This error can also be caused by a typing error; for example, a space or period is omitted between two identifiers.

Action

Change your source code to reduce the length of the identifier to 100 characters or fewer, or the length of the number to 30 digits or fewer. If a space or period has been omitted, correct the input accordingly. Although an external function and external method entry point has a maximum length of 255, if it is longer than 100 characters, the namespace or entry point must be enclosed in single quote (') or double quote (") characters.

6008 Hex integer too long

Cause

This error occurs if a hexadecimal integer is greater than eight characters in length.

Action

Change your source code to reduce the length of the integer to fewer than eight characters. If you intended to define a hexadecimal string rather than an integer, enclose the string (following the hash sign (#)) in double quotation (") or single quotation (') characters.

6009 Hex character exceeds maximum character value

Cause

This error occurs if any of the component characters of a hexadecimal string exceed the allowable range for character values.

Hexadecimal strings in JADE are represented as a sequence of space-separated hexadecimal characters; for example:

```
WhiteSpace = #"09 0A 0B 0C 0D 20";
```

The allowable range for character values is:

00 - FF	In an ANSI environment
00 - FFFF	In a Unicode environment

Action

Change your source code to ensure that the hexadecimal characters are within the permitted range. Insert spaces if they were accidentally omitted between characters.

6010 Hex literal contains non-hex character

Cause

This error occurs if a hexadecimal string or a numeric literal includes illegal characters. Hexadecimal string and numeric literals in JADE can contain only the characters 0 through 9, a through f, and A through F. They cannot contain decimal points (hexadecimal Real or Decimal quantities cannot be defined).

Action

Change your source code to remove the non-hexadecimal characters from the string or numeric literal.

6011 String delimiter mismatch

Cause

This error occurs if your entry contains an opening string delimiter (double quote or single quote character) with no closing delimiter.

You may have omitted the closing delimiter or you may have placed it in a comment. Alternatively, you may not have been consistent with your choice of delimiter; the same quote mark character must be used for the opening and closing delimiter.

Action

Determine the correct placement of the closing delimiter and then change your source code accordingly.

6012 Cannot compile user method in RootSchema context

Cause

This error occurs if you compile a method, or add a global constant, translatable string, or external function with the **RootSchema** selected in the Schema Browser.

Action

Select another user schema in the Schema Browser and then recompile the method, or add the global constant, translatable string, or external function.

6013 Type method cannot be updating

Cause

This error occurs if you attempt to use the **updating** method option with a type method.

Action

Remove the **updating** method option or make the method an instance method.

6014 Type method cannot be a conditional

Cause

This error occurs if you attempt to use the **condition** method option with a type method.

Action

Remove the **condition** method option or make the method an instance method.

6015 Type method cannot be a partition method

Cause

This error occurs if you attempt to use the **partitionMethod** method option with a type method.

Action

Remove the **partitionMethod** method option or make the method an instance method.

6017 Type method cannot be lockReceiver

Cause

This error occurs if you attempt to use the **lockReceiver** method option with a type method.

Action

Remove the **lockReceiver** method option or make the method an instance method.

6018 Option should be assigned a value

Cause

This error occurs if the schema file defines an option that requires an associated value and the value is not present.

Action

This error should not normally occur if JADE extracted the schema file correctly. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6019 Invalid option

Cause

This error occurs if you specify an option that is not valid for the element that you are defining.

When you define types, methods, and certain other elements in a JADE schema file or method, the definition can be followed by a list of options. These options are in the form of a list of modifiers separated by commas, and they specify additional attributes of the item being defined. Examples include the **protected**, **updating**, and **abstract** options on a method definition (for details, see "[Method Options](#)", in Chapter 1 of the *JADE Developer's Reference*) and the **transient** option on a class definition.

This error can also result from a misplaced identifier or some other syntactic error, causing the compiler to interpret an identifier as an option when an option was not intended.

Action

Change the invalid option, or correct the method syntax if this is the cause of the error.

6020 Unknown schema

Cause

This error occurs if a schema file has specified that the schema being loaded should be created as a subschema of another schema but this superschema does not exist. This can result from an incorrect schema name in the schema file, or it can be because the superschema does not yet exist in the current JADE database.

Action

Correct the schema name in the schema file, if it is incorrect. If the name is correct but the superschema does not yet exist, create it in JADE or load it from another schema file.

6021 Unknown library

Cause

This error occurs if an external method references a library that has not been defined.

The definition of an external method includes the name of the library and entry point that implement the external method. All libraries that are to be used in external methods must first be defined within the current schema or a superschema.

Action

Define the library by using the Library Browser in the JADE development environment, or add its definition to the schema file that is being loaded.

6022 Unknown class

Cause

This error occurs if the input refers to a class that does not exist.

Action

Correct the class name, if it is incorrect. If the class name is correct but the class has not been defined, define it by using the JADE Class Browser in the JADE development environment or by using a schema file.

6023 Unknown type

Cause

This error occurs if the input refers to a type (primitive type or class) that does not exist.

Action

Correct the type name, if it is incorrect. If the type name is correct but the type has not been defined, define it by using the JADE Class Browser in the JADE development environment or by using a schema file.

6024 Unknown property

Cause

This error occurs if the input refers to a property that does not exist.

Action

Correct the property name, if it is incorrect. If the property name is correct but the property has not been defined, define it by using the JADE Class Browser in the JADE development environment or by using a schema file.

6025 Unknown method

Cause

This error occurs if the input refers to a method that does not exist.

If this error occurs against the signature of a method implementation during a schema file load, it indicates that the method was not declared in the **JadeMethodDefinitions** section of the schema file.

Action

Correct the method name, if it is incorrect. If the method name is correct but the method has not been defined, define it by using the JADE Class Browser in the JADE development environment or by using a schema file.

6026 Unknown property or method

Cause

This error occurs if the input refers to a property or method that does not exist.

Action

Correct the property or method name, if it is incorrect. If the property or method name is correct but the property or method has not been defined, define it by using the JADE Class Browser in the JADE development environment or by using a schema file.

6027 Unknown identifier

Cause

This error occurs if you use an unknown identifier in a method. An identifier in a method must be resolved as one of the following:

- A property or method of the receiver
- A parameter
- A local variable
- A class or primitive type name
- A constant (global, class, or local)

Action

Correct the identifier name, if it is incorrect. If the identifier is correct but the object it references has not been created, create the required object.

6028 Type method cannot be notImplemented

Cause

This error occurs if you attempt to use the **notImplemented** method option with a type method.

Action

Remove the **notImplemented** method option or make the method an instance method.

6029 Unknown translatable string or user format

Cause

This error occurs if the input refers to a translatable string or user format (preceded by a dollar sign) that could not be found.

Action

Correct the name of the translatable string or user format, if it is incorrect. If it is correct and the translatable string or user format has not been defined, define it by using the appropriate browser in the JADE development environment or by using a schema file.

6030 Unknown external function

Cause

This error occurs if the input refers to an external function that does not exist.

Action

Correct the external function name, if it is incorrect. If the external function name is correct but the function has not been defined, define it by using the JADE External Functions Browser or by using a schema file.

6031 Unknown database file

Cause

This error occurs if the schema file refers to a database (map) file that has not been defined.

Action

Replace the invalid database file name with one that is defined.

6033 Type method cannot be a unit test

Cause

This error occurs if you attempt to use the **unitTest** method option with a type method.

Action

Remove the **unitTest** method option or make the method an instance method.

6034 Class already defined with a different superclass

Cause

This error occurs if your schema file attempts to redefine the superclass of an existing class. You cannot change the superclass of an existing class in this way.

Action

Correct the schema so that the class retains its original superclass.

6035 Constant already defined in another schema

Cause

This error occurs if your schema file attempts to define a constant that is already defined in another schema.

Action

Remove the duplicated constant definition.

6036 Constant already defined in another class

Cause

This error occurs if your schema file attempts to define a class constant that is already defined in another class.

Action

Remove the duplicated constant definition.

6037 Constant already defined in another primitive type

Cause

This error occurs if your schema file attempts to define a constant that is already defined in another primitive type.

Action

Remove the duplicated constant definition.

6038 Translatable string already defined in another locale

Cause

This error occurs if your schema file attempts to define a translatable string that is already defined in another locale.

Action

Remove the duplicated translatable string definition.

6039 Locale format already defined in another schema

Cause

This error occurs if your schema file attempts to define a locale format that is already defined in another schema.

Action

Remove the duplicated locale format.

6040 Same name as existing type

Cause

This error occurs if you attempt to define a constant or type with the same name as an existing type.

Action

Rename the constant or type that you are attempting to define, or delete the existing type.

6041 Same name as existing class

Cause

This error occurs if you attempt to define a constant or type with the same name as an existing class.

Action

Rename the constant or type that you are attempting to define, or delete the existing class.

6042 Same name as existing class constant

Cause

This error occurs if you attempt to define a constant with the same name as an existing class constant.

Action

Rename the constant that you are attempting to define, or delete the existing constant.

6043 Same name as existing type constant

Cause

This error occurs if you attempt to define a constant with the same name as an existing type constant.

Action

Rename the constant that you are attempting to define, or delete the existing constant.

6044 Same name as existing global constant

Cause

This error occurs if you attempt to define a constant or type with the same name as an existing global constant.

Action

Rename the constant or type that you are attempting to define, or delete the existing global constant. (You cannot delete a system global constant.)

6045 Same name as property

Cause

This error occurs if you attempt to define a parameter or local variable with the same name as a property.

JADE does not allow the definition of parameters or local variables with the same name as a property of the receiver. This avoids any confusion that might arise from such a duplication of names.

Action

Delete or rename the parameter or local variable that you are attempting to define, or delete or rename the existing property.

6046 Same name as system variable

Cause

This error occurs if you attempt to define a parameter or local variable with the same name as a system variable. (For details, see "[System Variables](#)", in Chapter 1 of the *JADE Developer's Reference*.)

Action

Delete or rename the parameter or local variable that you are attempting to define.

6047 Same name as existing method

Cause

This error occurs if your schema file attempts to define a property that has the same name as a method in the current class or a superclass.

Action

Remove the property or the existing method.

6048 Method already defined in another schema

Cause

This error occurs if your schema file attempts to define a method that already exists in the same class or primitive type, in a superschema or a subschema of the current schema.

Action

Remove the duplicated method.

6049 Type method cannot use 'self'

Cause

This error occurs if you attempt to use the **self** system variable in a type method. Type methods operate on a type and have no instance. Instance access via the **self** system variable is therefore not possible.

Action

Remove the **self** system variable from the method or make the method an instance method.

Alternatively, if you want to reference the type of the current method receiver, use the **selfType** system variable.

6050 Cannot change name

Cause

This error occurs if you attempt to change the name of a method or external function by editing the header and then recompiling it. This is not permitted in JADE.

Action

To rename a method, select the appropriate method in the Methods List of the Class Browser in the JADE development environment and then select the **Rename** command from the Methods menu. To rename an external function, you must first delete it and then add it again under the new name.

6051 Duplicated identifier

Cause

This error occurs if you attempt to define an object with the same name as an existing object.

This can occur by adding a method that is already present in the latest version of a class to the current version through a schema load, regardless of whether the method has the same signature.

Action

Rename the object being defined or the existing object with the same name.

6052 Duplicated label

Cause

This error occurs if you use the same label in two or more nested iteration loops.

Action

Change your application code to rename or delete one of the duplicated labels.

6053 Label mismatch

Cause

This error occurs if the label name at the start of an iteration loop does not match the label at the end.

When you specify a label at the beginning of an iteration loop, the label name can be restated at the end of the loop. This is primarily for documentation and readability purposes, and does not affect the execution of the loop. However, the compiler checks that the label specified at the start of the loop matches the label at the end of the loop, and this error results if the two do not match.

Action

Change your application code to delete the label at the end of the loop, or correct it to match the label at the start of the loop.

6054 Invalid break or continue label

Cause

This error occurs if your application code contains a **break** or **continue** instruction that specifies a label that does not match the labels on any of the enclosing iteration loops.

Action

Correct or remove the label specified in the **break** or **continue** instruction, or delete the **break** or **continue** instruction if it is not required. If the label on the enclosing loop is incorrect, correct this.

6055 Nothing to break or continue

Cause

This error occurs if your application code contains a **break** or **continue** instruction outside all loops.

A **break** or **continue** instruction dictates that the current iteration loop should be aborted or continued and therefore must always appear within a loop.

Action

Change your application code to remove the **break** or **continue** instruction, or implement an enclosing iteration loop.

6056 This is not a method

Cause

This error occurs if an identifier in your application code is followed by parentheses, indicating a method call, but the identifier refers to an object other than a method (for example, a property).

Action

Change the method name if it is incorrect. If a method call was not intended, remove the parentheses.

6057 Non-constant expression

Cause

This error occurs if a non-constant expression is used to initialize a constant (for example, an expression involving a property of the current class).

The expression used to initialize a constant must yield a constant value; that is, it must be made up of literal values or other constants (or both).

Action

Change the expression so that it refers only to literal values or to other constants, or both literal values and other constants.

6058 Invalid statement

Cause

This error occurs if an invalid instruction is encountered.

Instructions in the JADE language are broadly grouped into the following categories.

- Keyword instructions; for example, **if**, **foreach**, or **beginTransaction** instructions
- Assignment instructions; for example:

```
name := str;
```

- Message send instructions; for example:

```
setName(str);
```

Action

Correct or delete the invalid instruction.

6059 Invalid operator

Cause

This error occurs if the input contains an operator that is not valid in the context in which it has been used.

Action

Correct or delete the expression that caused the error.

6060 Incompatible operand

Cause

This error occurs if the input contains a pair of operands that are not compatible for the operation that is being performed; for example, it is invalid to add a **Real** value to a **Date** or to compare a **Binary** and an **Integer** by using the **<** operator.

For a complete list of primitive type compatibility rules in JADE, see [Chapter 1](#) of the *JADE Developer's Reference*.

Action

Delete the invalid expression, or modify the operator or the operands to make the expression valid.

In some cases, it may be possible to type cast one or other of the operands in order to produce a valid expression.

6061 Non-numeric operand

Cause

This error occurs if your application code attempts to perform a numeric operation (+, -, *, /, ^, **mod**, or **div**) on an operand that is not a number (**Integer**, **Real**, or **Decimal** primitive type).

Action

Delete the invalid expression, or modify the operator or the operand to make the expression valid.

In some cases, it may be possible to type cast the invalid operand to a numeric primitive type in order to produce a valid expression.

6062 Non-boolean operand

Cause

This error occurs if your application code attempts to perform a Boolean operation (**and**, **or**, or **not**) on an operand that is not of type **Boolean**.

Action

Delete the invalid expression, or modify the operator or the operand to make the expression valid.

In some cases, it may be possible to type cast the invalid operand to a **Boolean** primitive type in order to produce a valid expression.

6063 Non-alphabetic operand

Cause

This error occurs if your application code attempts to perform a string concatenation operation (&) on an operand that is not of type **String** or **Character**.

Action

Delete the invalid expression, or modify the operator or the operand to make the expression valid.

In some cases, it may be possible to type cast the invalid operand to a **String** or **Character** in order to produce a valid expression.

6064 This is not a string

Cause

This error occurs if your application code contains a non-string expression where a string is required.

Action

Change the expression so that it is of type **String**. In some cases, it may be possible to type cast the expression to convert it to a string.

6065 Not an integer

Cause

This error occurs if your application code contains a non-integer expression where an integer is required.

Action

Change the expression so that it is of type [Integer](#). In some cases, it may be possible to type cast the expression to convert it to an integer.

6067 Type method cannot use feature

Cause

This error occurs if you attempt to access an instance feature using the **self** system variable. Type methods operate on a type and have no instance. Instance feature access via the **self** system variable is therefore not possible.

Action

Remove the reference to the instance feature from the method or make the method an instance method.

6068 Type/non-type method reimplementation mix

Cause

This error occurs if you attempt to reimplement an instance method with a type method or if you attempt to reimplement a type method with an instance method.

Action

Change the method type of the base method or reimplementation method so that the two methods are both instance methods or both type methods.

6069 Too many parameters

Cause

This error occurs if you attempt to define a method or external function with more than the maximum allowable number of parameters. This maximum is 129 parameters.

Action

Modify the method or external function definition so that there are not more than 129 parameters.

6070 Parameter number mismatch

Cause

This error occurs when the number of parameters in a message send (method call) does not match the number of parameters in the definition of the method.

This error can also occur when you compile mapping methods. Mapping methods must always have two parameters.

Action

Correct the number of parameters in the message send to match the parameters of the method definition.

In some cases, the message send may be correct and the method definition incorrect. In these cases, you should correct the method definition.

If you are compiling a mapping method, ensure that there are two parameters.

A mapping method has the following signature.

```
method-name(set: Boolean; _value: type io);
```

The **set** parameter is **true** if the property is being set and **false** if the value of the property is being read. The **_value** parameter is a variable value of the same type as the property.

6071 Incompatible parameter type

Cause

This error occurs if you are passing a parameter to a method (the *actual* parameter) but the parameter does not match the type of the corresponding parameter in the method definition (the *formal* parameter).

The meaning of this error depends on the definition of the parameter, as follows.

- For parameters defined as **input** or **constant**, it must be possible to assign the *actual* parameter to the *formal* parameter, as follows.
 - For composite (non-primitive) parameters, the class of the actual parameter must be the same as, or a subclass of, the class of the formal parameter.
 - For primitive parameters, the actual parameter must be of a type that is assigned to the type of the formal parameter.
- For parameters defined as **output**, it must be possible to assign the formal parameter to the actual parameter, as follows:
 - For composite (non-primitive) parameters, the class of the formal parameter must be the same as, or a subclass of, the class of the actual parameter.
 - For primitive parameters, the formal parameter must be of a type that is assigned to the type of the actual parameter.
- For parameters defined as **io**, it must be possible to assign in both directions, as follows:
 - For composite (non-primitive) parameters, the class of the formal parameter must be identical to the class of the actual parameter.
 - For primitive parameters, the formal parameter must be of a type that is assigned to the type of the actual parameter, and the reverse must be true.

For full details about assignment compatibility rules, see "[JADE Instructions, Expressions, and Assignments](#)", in Chapter 1 of the *JADE Developer's Reference*.

Action

Make the parameter types compatible, by correcting the parameter being passed or the method definition.

6072 Incompatible types

Cause

This error occurs when the right-hand operand is of a type that cannot be assigned to the left-hand operand.

In general, the class of the right-hand side must be the same as, or a subclass of, the class of the left-hand side for composite (non-primitive) objects. For the rules for assignment compatibility for primitive objects, see "[Assignments](#)", in Chapter 1 of the *JADE Developer's Reference*.

When this error occurs for an array or dictionary subscript (using the bracket (`[]`) subscripting syntax), it indicates that the type of the subscript expression is not compatible with the corresponding key of the array or dictionary.

When this error occurs for the control variable of a [foreach](#) instruction, it indicates that the type of the control variable is not compatible with the type of the collection members.

Action

Change the operands or their types so that the types are compatible. In some cases, it may be possible to add a type cast in order to make the operands compatible.

6073 Length must be specified

Cause

This error occurs if your application code defines an external function parameter or return type with no length specification when a length is required. A length is required for usage **io** or **output** parameters of type [String](#) or [Binary](#), and for **Binary** return primitive types.

Action

Specify the length of the parameter or return primitive type, by including an integer enclosed in brackets (`[]`) after the type name.

6074 Length of Real must be 4 or 8 bytes

Cause

This error occurs if your application code specifies a [Real](#) parameter or return primitive type in an external function with a length other than 4 or 8 bytes. Only these lengths are valid for **Real** parameters and return primitive types.

Action

Correct the length of the **Real** parameter or return primitive type.

6075 Type/non-type method interface implementation mix

Cause

This error occurs if you attempt to implement an instance interface with a type method or if you attempt to implement a type interface with an instance method.

Action

Change the method type of the interface or implementation method so that the interface and implementation method are of the same kind (that is, an instance or a type method).

6076 Operator is incompatible with the method type

Cause

This error occurs if you attempt to use the dot (.) operator to call a type method or the @ operator to call an instance method.

Action

Change the operator to the @ operator to call a type method or to the dot (.) operator to call an instance method.

6077 Type method cannot be a constructor or destructor

Cause

This error occurs if you attempt to use the **typeMethod** method option with a constructor or destructor method. Constructor and destructor methods operate on instances of a type. Type methods operate on types and have no instances, so they cannot be used as constructor or destructor methods.

Action

Remove the **typeMethod** method option from the constructor or destructor method.

6081 Cannot assign to create expression

Cause

This error occurs if your application code attempts to assign an extended create expression as the value of an **io** or **output** parameter. This is because a value is effectively assigned from the called method back to the corresponding parameter in the calling method for parameters of these types; however, a **create** expression is not assignable.

Action

Change your application code to replace the **create** expression with something that is assignable; for example, a local variable, parameter, or property.

It may also be possible to resolve the error by changing the parameter type from **io** or **output** to **constant** (the default) or **input**.

Applies to Version: 2020.0.02 and higher

6082 Cannot assign to compound expression

Cause

This error occurs if your application code assigns a value to a compound expression, for example:

```
a + b := c;
```

As **a+b** does not identify a storage location to which a value is assigned, this syntax is invalid.

This error is also returned if you supply a compound expression as the value of an **io** or **output** parameter. This is because a value is effectively assigned from the called method back to the corresponding parameter in the calling method for parameters of these types.

Action

Change your application code to replace the compound expression with a simple expressive that can act as the target of an assignment.

In the case of method parameters, it may be possible to resolve the error by changing the parameter type from **io** or **output** to **constant** (the default) or **input**.

6083 Cannot assign to type

Cause

This error occurs if your application code assigns a value to a class or primitive type. It is also returned if the operand of a **create** instruction is a class or primitive type.

Action

Remove or correct the invalid assignment. If you are coding a **create** instruction, the first operand should be a local variable or property that is to store the created instance.

6084 Cannot assign to constant

Cause

This error occurs if your application code assigns a value to a constant reference. The constant reference can include the following.

- Global, class, or local constants.
- Method parameters defined as **constant** or **input**.

Method parameters are **constant** by default, so parameters cannot be assigned to unless they are explicitly defined as being **io** or **output**.

- The **self** variable.

You cannot assign to **self**, except in a primitive method.

This error is also returned if your code assigns to an expression that ends in a type cast. You cannot assign to type cast expressions.

Action

Remove or correct the invalid assignment. If you are assigning to a parameter and you intend to export the value of that parameter to the calling method, you should define the parameter as being **io** or **output**.

6085 Cannot assign to method

Cause

This error occurs if the left-hand side of an assignment is a method call (message send). This is not valid, as a method call does not identify a storage location that can act as the target of an assignment.

Action

Correct or remove the invalid instruction.

6086 Cannot assign to exclusive collection

Cause

This error occurs if your application code attempts to assign a value to an exclusive collection.

When you add a reference to a class and that reference is a collection, it is normally treated as an **exclusive** collection. The only exception to this is when the reference has no inverse and you explicitly uncheck the **Exclusive** check box in the Define Reference dialog.

An exclusive collection is inherently bound to its parent object; that is, it is created when the parent object is created and deleted when the parent object is deleted. It cannot be assigned to, and it cannot be referenced from any other object.

Action

Remove the invalid assignment. If the collection has no inverse and you did not intend it to be exclusive, modify it by unchecking the **Exclusive** check box on the Define Reference dialog. The collection will then be shared; that is, you will have to create the collection before you can use it, and it will be possible to reference it from any number of other objects.

6087 Cannot assign to automatic reference

Cause

This error occurs if there is an attempt to update an automatic reference from user code.

Inverse references can be defined as having a manual reference at one end and an automatic reference at the other. Manual references can be updated by user code, and the automatic reference is updated automatically by the JADE Object Manager in response to this. You cannot attempt to update an automatic reference from user code.

Action

Remove the invalid assignment. You can update the manual end of the inverse instead, which causes the automatic reference to be updated automatically.

6088 Cannot update automatic collection

Cause

This error occurs if there is an attempt to update an automatic collection from user code.

Inverse references can be defined as having a manual reference at one end and an automatic reference (that can be a collection) at the other. Manual references can be updated by user code, and the automatic reference is updated automatically by the JADE Object Manager in response to this. You cannot update an automatic collection from user code.

Action

Remove the invalid update of the automatic collection. Alternatively, you can update the manual end of the inverse, which causes the automatic collection to be updated automatically.

6089 Cannot assign to read-only property

Cause

This error occurs if your application code assigns to a read-only property outside the owning class or one of its subclasses.

Properties can be defined as being read-only. The value of a read-only property is accessed by any method in any class, but it is assigned a new value only by methods within the class in which the property is defined, or one of its subclasses.

Action

Change your application code to remove the invalid assignment.

If you did not intend the property to be read-only, modify it by selecting the property in the Class Browser and then selecting the **Change** command from the Properties menu.

6090 Cannot access protected property or method

Cause

This error occurs if your application code references a protected property or method outside the owning class or one of its subclasses. Properties and methods are defined as being protected. A protected property or method is referenced by methods only within the class in which it is defined, or one of its subclasses.

Action

Change your application code to remove the invalid access.

If you did not intend the property or method to be protected, modify it and remove the protected attribute. For properties, this can be done by selecting the property in the Class Browser and then selecting the **Change** command from the Properties menu. For methods, the protected option should be removed from the method header and the method then recompiled.

6091 Cannot delete exclusive collection

Cause

This error occurs if there is an attempt to delete an exclusive collection. Exclusive collections cannot be explicitly deleted.

When you add a reference to a class and that reference is a collection, it is normally treated as an **exclusive** collection. The exception to this is when the reference has no inverse and you explicitly uncheck the **Exclusive** check box in the Define Reference dialog.

An exclusive collection is inherently bound to its parent object; that is, it is created when the parent object is created and deleted when the parent object is deleted. It cannot be assigned to, it cannot be referenced from any other object, and it cannot be explicitly deleted.

Action

Change your application code to remove the invalid deletion.

If the collection has no inverse and you did not intend it to be exclusive, modify it by unchecking the **Exclusive** check box in the Define Reference dialog. The collection is then **shared**; that is, you have to create the collection before you use it and then delete it when it is no longer required.

6092 Cannot assign to variable

Cause

This error occurs in one of the following conditions.

- If your application code attempts to assign to a system variable. The values of system variables are maintained internally by JADE, and you cannot assign a value to them in your application code.
- If your application code passes a system variable type guard expression (for example, **self.Customer**) as a parameter to another method where that parameter is defined as usage **io** or **output**. This is not permitted because the other method can assign to this parameter.
- When an assignment cannot be made to the variable used as the control variable in a **foreach** instruction; for example, this can be an **input** or **constant** parameter.

Action

Change your application code to remove the invalid assignment.

If the error was reported on the passing of a parameter, it may be possible to change the usage of that parameter to **constant** (the default) or **input** in order to resolve the error.

If the error was reported against the control variable of a **foreach** instruction, use a local variable as the control variable.

6093 Cannot assign an exclusive collection to a property

Cause

This error occurs if your application code attempts to assign an exclusive collection to another storage location. An exclusive collection cannot be referenced from any other object.

When you add a reference to a class and that reference is a collection, it is normally treated as an **exclusive** collection. The exception to this is when the reference has no inverse and you explicitly uncheck the **Exclusive** check box in the Define Reference dialog. An exclusive collection is inherently bound to its parent object; that is, it is created when the parent object is created and deleted when the parent object is deleted. It cannot be assigned to, or referenced from, any other object.

Action

Change your application code to remove the invalid assignment. If the collection has no inverse and you did not intend it to be exclusive, modify it by unchecking the **Exclusive** check box in the Define Reference dialog. The collection will then be shared; that is, you will have to create the collection before you use it, and it will be possible to reference it from any number of other objects.

6094 Updating method called for constant value

Cause

This error occurs when an updating method is called for an operand that has a value that cannot be updated.

The operand can be a:

- Property of the receiver object in a non-updating method
- Constant parameter
- Substring
- Read-only primitive attribute

Action

Remove or correct the invalid method call.

If the operand is a property of the receiver and the current method is non-updating, resolve the error by making the current method updating.

If the operand is a **constant** parameter (the default), resolve the error by changing the parameter to **input**.

In some cases, it may be possible to change the method being called from updating to non-updating if it does not actually update the receiver.

6096 Cannot call updating method from non-updating method

Cause

This error occurs if the current method is non-updating and you attempt to call an updating method of the receiver (**self**) or one of the primitive attributes of the receiver. As this method call might update the receiver, it is not permitted in a non-updating method.

Action

Remove the invalid method call, or make the method an updating method by adding the **updating** option to the method header.

Alternatively, it may be possible to make the called method non-updating.

6097 Cannot update property in non-updating method

Cause

This error occurs if your application code attempts to modify a property of the receiver (**self**) in a non-updating method. As non-updating methods cannot modify the receiver object, this is invalid.

This error is also returned if your code passes a property of the receiver as a parameter to another method, where that parameter is defined as usage **io**, **output**, or **input**. As the other method can update this parameter (and would thus indirectly update the receiver), this is not permitted.

Action

Remove the code that attempts to modify the property. If the error is being reported on the passing of a parameter, it may be possible to change the usage of that parameter to **constant** (the default), in order to resolve the error.

6098 Cannot update constant

Cause

This error occurs if your application code attempts to update the value of a constant. This is not permitted.

This error is also returned if your code passes a constant as a parameter to another method, where that parameter is defined as usage **io**, **output**, or **input**. As the other method can update this parameter (and would thus indirectly update the constant), this is not permitted.

Action

Remove the invalid code. If the error is being reported on the passing of a parameter, it may be possible to change the usage of that parameter to **constant** (the default), in order to resolve the error.

6099 Cannot assign to primitive type conversion

Cause

This error occurs if your application code assigns a value to a primitive type conversion. A primitive type conversion can never appear on the left-hand side of an assignment.

This error is also returned if your code passes a primitive type conversion as a parameter to another method, where that parameter is defined as usage **io** or **output**. As the other method can assign a value to this parameter, this is not permitted.

Action

Correct your application code to remove the primitive type conversion. If the error is being reported on the passing of a parameter, it may be possible to change the usage of that parameter to **constant** (the default) or **input**, in order to resolve the error.

6100 Redundant type cast

Cause

This error occurs if your application code uses a type cast on the left-hand side of an assignment. A type cast is always redundant in this situation.

Action

Change your application code to remove the redundant type cast.

6110 Invalid return expression

Cause

This error occurs if the current method is defined as having no return type but a **return** instruction was encountered that attempts to return a value.

Action

Add a return type to the method definition, or remove the return value from the **return** instruction.

6111 Missing return expression

Cause

This error occurs if the current method is defined as having a return type but a **return** instruction was encountered that does not return a value.

Action

Remove the return type from the method definition, or add a return value to the **return** instruction.

6112 Return expression mismatch

Cause

This error occurs if the type of the value returned by a **return** instruction is not compatible with the defined return type of the method.

If the return type of the method is a class, the value that is returned must be an instance of that class or one of its subclasses. If the return type is a primitive type, the value that is returned must be assignment-compatible with that type.

For details about assignment compatibility rules for primitive types, see "[Assignment Compatibility](#)" or "[Instructions, Expressions, and Assignments](#)", in Chapter 1 of the *JADE Developer's Reference*.

Action

Modify the return type of the method or the type of the returned value, in order to make them compatible. In some cases, it may be possible to add a type cast to the **return** expression to make it compatible with the method return type.

6113 Method will always return before this statement is reached

Cause

This error occurs if the current method includes code that may never be executed because it immediately follows a **return** instruction and the method will always return before the code is reached.

Action

Remove the **return** instruction or the code that follows it.

6114 Method does not return a value

Cause

This error occurs if the current method is defined as having a return type but there are no **return** instructions in the method.

Action

Remove the return type from the method signature, or add at least one **return** instruction to the body of the method.

6115 Constructors and destructors cannot have parameters or return type

Cause

This error occurs if you attempt to define a constructor (**create**) or destructor (**delete**) method with parameters or a return type. Constructors and destructors in JADE cannot have parameters or a return type.

Action

Remove the parameters or return type from the method. As parameters cannot be used in a constructor, any initialization of an object using external values must be performed by using an additional **initialize** (or similarly named) method.

6116 This is not a collection expression

Cause

This error occurs if the second operand of a **foreach** instruction is not a collection.

Action

Change your application code to replace the operand with an expression that yields a collection.

6117 This is not a collection class

Cause

This error occurs if the input specifies the membership for a class that is not a collection.

Action

Remove the membership specification (that consists of the keyword **of** followed by the name of the membership type).

6118 Already defined as a JADE method

Cause

This error occurs if you attempt to define an external method through your schema file when a JADE method of the same name already exists.

Action

Remove the external method definition from the schema file, or delete the existing JADE method.

6119 Already defined as an external method

Cause

This error occurs if you attempt to define a JADE method through your schema file when an external method of the same name already exists.

Action

Remove the JADE method definition from the schema file, or delete the existing external method.

6124 Signature of referenced method has changed

Cause

This error occurs when the signature of a method has been changed but methods that reference the changed method have not yet been recompiled.

Action

Change the parameters on the call to the changed method, if necessary, and then recompile the current method.

6125 Signature of superclass method has changed

Cause

This error occurs when the signature of a method has been changed but its subclass reimplementations have not yet been recompiled.

Action

Change the signature of the current method to match the superclass definition, if necessary. Then recompile the current method.

6126 Type has changed

Cause

This error occurs when the type of a property or constant has been changed or a type has been renamed but methods that reference the changed item have not yet been recompiled.

Action

Recompile the current method and correct any errors that result from the change of type.

6127 Type method cannot be a mapping method

Cause

This error occurs if you attempt to use the **typeMethod** method option with a mapping method.

Action

Remove the **typeMethod** method option from the mapping method.

6128 Type method cannot be a Web Service

Cause

This error occurs if you attempt to use the **webService** method option with a type method.

Action

Remove the **webService** method option or make the method an instance method.

6129 Cannot assign to self in non-updating primitive method

Cause

This error occurs when attempting to assign to the **self** system variable in a non-updating primitive method.

Action

If the **self** system variable is intentionally being updated, change the method signature so that the **updating** method option is specified (for example, **m1() updating;**); otherwise change your application code to remove the invalid assignment.

6130 Invalid method name

Cause

This error occurs if you attempt to define a method with the name **create** or **delete** in a primitive type, global class (subclass of the **Global** class), or application class (subclass of the **Application** class). These method names are reserved for class constructors and destructors, and cannot be used as method names in primitive types (which cannot have constructors or destructors) or global or application classes (whose instances are system-created).

Action

Choose a different name for the method. If you intended to define a constructor or destructor for an application class, use the Define Application dialog to specify **initialize** and **finalize** methods.

6131 Method signature conflicts with superclass implementation

Cause

This error occurs if the signature of a method that is reimplemented by a subclass is not compatible with its superclass.

To be compatible, the following conditions must apply.

- Both methods must have the same number of parameters.
- Each parameter in the subclass implementation must be of a type that is compatible with the equivalent parameter in the superclass implementation. This means that the type of the subclass parameter must be the same primitive type, the same class, or a subclass of the type of the superclass parameter.
- Each parameter in the subclass implementation must have the same or stricter usage than the equivalent parameter in the superclass implementation. The following must therefore apply.
 - If the superclass parameter is **constant**, the subclass parameter must also be **constant**.
 - If the superclass parameter is **input**, the subclass parameter can be **input** or **constant**.
 - If the superclass parameter is **output**, the subclass parameter must also be **output**.
 - If the superclass parameter is **io**, the subclass parameter can have any usage.
- Both implementations must have no return type, or their return types must be compatible. To be compatible, the return type of the subclass implementation must be the same primitive type, the same class, or a subclass of the return type of the superclass implementation.
- The **protected**, **mapping**, and **updating** options must be the same (either absent or present) in both implementations.

Action

Change the signature of the superclass or the subclass implementation of the method so that it complies with these rules.

6132 Method signature conflicts with prior definition

Cause

This error occurs if the signature of a method in the **methodImplementations** section of a schema file conflicts with the signature of the method in the current schema or as previously defined in the **JadeMethodDefinitions** section of the same schema file.

Action

Change the method signature to conform to the prior definition.

If you want to change the signature of a method that already exists in the current schema, define the new signature in the **JadeMethodDefinitions** section of the schema file and then provide the new implementation in the **methodImplementations** section.

6133 Method signature conflicts with corresponding control method

Cause

This error occurs when you have defined a method as corresponding to a specific control method but its signature is not compatible with that method. To be compatible, the following conditions must apply.

- Both methods must have the same number of parameters.
- Each parameter in the subclass implementation must be of a type that is compatible with the equivalent parameter in the superclass implementation.

This means that the type of the subclass parameter must be the same primitive type, the same class, or a subclass of the type of the superclass parameter.

- Each parameter in the subclass implementation must have the same or stricter usage than the equivalent parameter in the superclass implementation.

This means that the following must apply.

- If the superclass parameter is **constant**, the subclass parameter must also be **constant**.
- If the superclass parameter is **input**, the subclass parameter can be **input** or **constant**.
- If the superclass parameter is **output**, the subclass parameter must also be **output**.
- If the superclass parameter is **io**, the subclass parameter can have any usage.
- Both implementations must have no return type, or their return types must be compatible.

To be compatible, the return type of the subclass implementation must be the same primitive type, the same class, or a subclass of the return type of the superclass implementation.

- The **protected**, **mapping**, and **updating** options must be the same (absent or present) in both implementations.

Action

Change the signature of the method so that it is compatible with the corresponding control method.

6134 This is not a boolean

Cause

This error occurs when a non-**Boolean** expression is encountered where a **Boolean** expression is required.

Action

Change the expression so that it yields a **Boolean** (**true** or **false**) value.

6135 This is not allowed on JADE methods

Cause

This error occurs if you attempt to define a library or entry point name in a JADE method. These attributes are defined only for external methods.

Action

Remove the invalid library or entry point definitions. If you intended the method to be an external method rather than a JADE method, you must delete it and then add it again as an external method.

6136 This is not an abstract class

Cause

This error occurs if you attempt to define an abstract method in a non-abstract class. Only abstract classes can contain abstract methods.

Action

Remove the **abstract** option from the method signature, or define the class as being abstract.

6137 Abstract methods cannot have a body

Cause

This error occurs if an abstract method includes a method body after the signature. Abstract methods must be defined with a method header only.

Action

Remove the method body. If you did not intend the method to be abstract, remove the **abstract** option from the method signature.

6139 Cannot change class map

Cause

This error occurs if you attempt to redefine a class map.

Specifically, a class map has already been defined indicating which file will hold instances of a particular class and the schema file attempts to redefine the class map to use a different file.

Action

Remove the redefinition of the class map, or modify it so that it matches the existing definition.

Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

6140 No class map defined for class

Cause

This error occurs if your schema file defines a persistent class with no associated class map. Every persistent class must have a class map.

Action

Add a class map definition for the class.

6150 Value cannot be zero

Cause

This error occurs if you use a precision of **0** for a **Decimal** variable. The precision of a **Decimal** variable must be in the range **1** through **23**.

Action

Change the precision to a value in the range **1** through **23**.

6151 Value cannot be greater than 23

Cause

This error occurs if you use a precision greater than **23** for a **Decimal** variable. The precision of a **Decimal** variable must be in the range **1** through **23**.

Action

Change the precision to a value in the range **1** through **23**.

6152 Scale factor cannot exceed precision

Cause

This error occurs if the scale factor of a **Decimal** variable exceeds the precision.

In the definition of a **Decimal** variable, the precision is the first value within the brackets (**⌈**) and the scale factor is the second.

Action

Reduce the scale factor to be less than or equal to the precision.

6153 inheritMethod not valid in constructor or destructor

Cause

This error occurs if you have coded an **inheritMethod** in a constructor or destructor method. You cannot code an **inheritMethod** in a constructor or destructor, because all inherited constructors and destructors are automatically invoked when an object is created or deleted.

When an object is created, the JADE Object Manager executes all inherited constructors, starting with the highest superclass implementation and continuing down to the current class. When an object is deleted, all inherited destructors, starting with the implementation in the current class (if present) and continuing up to the highest superclass implementation.

Action

Change your application code to remove the **inheritMethod** call.

6154 Method not implemented in any superclass

Cause

This error occurs if the input contains an **inheritMethod** call but the method is not implemented in any superclass.

Action

Change your application code to remove the **inheritMethod** call, or implement the method in at least one superclass.

6155 Invalid use of pseudotype

Cause

This error occurs if a pseudotype has been used in a context in which it is not valid; for example, it is invalid to use the pseudotype **MemberType** when the current class is not a collection (and therefore does not have a membership type).

Action

Change your application code to remove or correct the invalid pseudotype.

6156 This is not an explicit inverse reference

Cause

This error occurs if you attempt to define an inverse through a schema file in which one of the references is not an explicit inverse reference. This means that the **explicitInverse** option was not present when the reference was defined in the **referenceDefinitions** section of the schema file.

Action

Remove the inverse definition from the schema file, or add the **explicitInverse** option to the definition of the reference.

6157 Inverse operator mismatch

Cause

This error occurs if an inverse definition with multiple parts contains a conflicting set of operators (**and**, **or**, and **xor**). All operators within a single inverse definition must be the same.

Action

Change any conflicting operators so that all operators within the inverse definition are the same.

6158 Incompatible update modes

Cause

This error occurs if the two references in an inverse definition have incompatible update modes. The update modes must be one of the following pairs.

```
manual - automatic
automatic - manual
manualAutomatic - manualAutomatic
```

Note The default value is **manualAutomatic** if no update mode is specified.

Action

Change one of the update modes so that they are compatible.

6159 Unknown type or constant

Cause

This error occurs if your application has no type cast or constant that matches the input.

Action

Correct the invalid type or constant name.

6160 Invalid type cast

Cause

This error occurs if the input contains a typecast that is not valid. This can be for one of the following reasons.

- The code attempts to cast a class to a primitive type or a primitive type to a class.
- An illegal primitive typecast is being performed (for a complete list of valid and invalid primitive type casts, see "[Type Casts](#)", in Chapter 1 of the *JADE Developer's Reference*).
- An illegal class type cast is being performed.

For a typecast to be legal, the class of the typecast must be the same as, or a subclass of, the defined class of the operand; for example, if **f** is defined to be of class **Form**:

```
base    := f.MyBaseForm // valid
                        // MyBaseForm is a subclass of Form
win     := f.Window     // invalid
                        // Window is not a subclass of Form
button := f.Button      // invalid
                        // Button is not a subclass of Form
```

Action

Change the typecast so that it conforms to the above rules.

6161 Invalid type

Cause

This error occurs if the type of the operand is not valid in the current context. The following examples demonstrate this.

- The operand of a **read** or **write** instruction is a method that does not return a value.
- The operand of a **create** or **delete** instruction is a primitive type or a method that does not return a value.

Action

Change the operand so that its type is valid for the instruction that is being performed.

6162 Cannot define global constant in terms of type constant

Cause

This error occurs if you attempt to define a global constant in terms of a class constant or primitive type constant. This is not supported.

Action

Change the global constant so that it is defined only in terms of literal values or other global constants.

6170 Attributes cannot have abstract type

Cause

This error occurs if an attribute has been defined with an abstract type. As attributes are always created when the parent object is created, they must also have a real (non-abstract) type.

Action

Change the type of the attribute to a real (non-abstract) type. This is likely to be a concrete subclass of the original abstract type.

6171 Cannot modify system basic object

Cause

This error occurs if a schema load attempts to modify a **system basic** object; that is, an object that is part of the JADE hard-coded schema.

Action

Remove the invalid code.

6172 Only RootSchema can have a null superschema

Cause

This error occurs if you have defined a schema as being a subschema of **null** in the **schemaDefinition** section of a schema file. Only the **RootSchema** can be defined as a subschema of **null**.

Action

Modify the schema definition so that it is defined as being a subschema of a valid schema (which can be the **RootSchema**, if no other schemas exist).

6173 Schema already exists with a different superschema

Cause

This error occurs if your schema file attempts to define a schema as being a subschema of one schema when it is already defined as being a subschema of another schema.

Action

Change the schema definition so that it matches the current definition of the schema.

6174 Non system user cannot load into system schema

Cause

This error occurs if a non-system user attempts to load a schema file into a system schema. This is not permitted.

Action

Load the schema file into a user (non-system) schema. Do this by specifying a user schema in the **schemaDefinition** section of the schema file or by setting the current schema in JADE to a user schema before loading the file.

6175 Cannot modify object defined in superschema

Cause

This error occurs if you attempt to apply certain changes to a class that is defined in the superschema.

Only a limited range of operations is permitted on classes that are defined in a superschema. New methods and constants can be added to those classes, but no other changes or additions are permitted.

The following operations are *not* permitted on classes that are defined in a superschema.

- Adding new properties.
- Modifying the signature or the implementation of existing methods.
- Adding or modifying documentation text.
- Adding new keys to a collection class.
- Defining a constructor (**create**) or a destructor (**delete**).

Action

Remove the lines that caused this error. If you need to modify or add to a superschema class, load those changes into the superschema that owns the class.

6176 Cannot subclass superschema form - this schema defines extra locales

Cause

This error occurs if you attempt to subclass a form that resides in a superschema when the current schema defines additional locales over and above those defined in the superschema. This cannot be supported, as JADE attempts to generate copies of the form for each locale in the current schema but is unable to find copies of the superschema form for some locales.

In summary, in order to subclass a superschema form, the set of locales supported by the current schema must be identical to those supported by the superschema.

Action

Remove the invalid subclass. Alternatively, you can add locales to the superschema or remove locales from the current schema until the two sets of locales are identical.

6177 User class cannot be mapped to system map file

Cause

This error occurs if your schema file attempts to map a user class to a system map file (one that is defined in the **RootSchema**).

Action

Correct the schema file so that the class is mapped to a user file (one that is defined in your schema).

6178 A schema cannot have more than one subclass of Application, Global or WebSession

Cause

This error occurs if, in a single schema, you attempt to define more than one application class (subclass of the **Application** class), more than one global class (subclass of the **Global** class), or more than one subclass of the **WebSession** class. Each schema can have only one subclass of **Application**, one subclass of **Global**, and one subclass of **WebSession**.

Action

Remove the second **Application**, **Global**, or **WebSession** subclass. If you are migrating from an earlier JADE release in which multiple **Application**, **Global**, and **WebSession** classes were allowed, you may need to merge the definitions of those classes.

6179 Cannot define subclasses of this class

Cause

This error occurs if you attempt to define a subclass of a class that cannot be subclassed; for example, **Control**, **JadeScript**, and certain abstract **Collection** classes.

Action

Remove or modify the invalid subclass definition.

6180 Cannot define inverse to a property in a superschema

Cause

This error occurs if you attempt to define an inverse reference between a property that is defined in the current schema and one that is defined in a superschema. Inverse references cannot span schema boundaries in this way.

Action

Restructure your schemas so that the properties that are to participate in the inverse relationship both reside in the same schema.

6181 No inverse is defined for this property

Cause

This error occurs if the schema file defines a property as being **explicitInverse** but does not define any inverses for the property. An **explicitInverse** property must always have at least one inverse.

Action

This error should not normally occur if the schema file was extracted correctly by JADE.

Note If you are extracting a class that has an inverse to a property of another class, it is your responsibility to ensure that the other class is extracted as well or that it already exists in the target schema with the appropriate property. If not, a schema load error occurs.

If you suspect that the schema file was not extracted correctly by JADE, contact your local JADE support center or JADE Support if your JADE licenses include support.

6183 Cannot specify entripoint and library for abstract method

Cause

This error occurs if a schema file attempts to define an entry point or library for an abstract external method. It is not valid to specify an entry point or library because an abstract method simply defines a signature and is never executed.

Action

Remove the entry point and library from the external method definition. Alternatively, make the method non-abstract by removing the **abstract** keyword from the method definition.

6184 Entrypoint and library must be specified

Cause

This error occurs if an external method definition omits the specification of the entry point and library identifying where the external method is implemented. These must be specified for external methods.

Action

Add the entry point and library names to the external method definition, in the following format.

```
method-header is entry-point-name in library-name;
```

6185 Invalid mapping method signature

Cause

This error occurs if you have defined a mapping method that does not conform to the required signature for mapping methods. Signatures for mapping methods should be entered in the following format.

```
methodName(set: Boolean; value: type-of-property io) mapping;
```

Action

Change the mapping method header to conform to the above signature. If you did not intend the method to be a mapping method, remove the **mapping** option from the method signature.

6186 No corresponding property for mapping method

Cause

This error occurs if you have defined a mapping method but no property of that name could be found in the current class or any of its superclasses.

Action

Rename the mapping method (by deleting and adding it again) so that it corresponds to the name of an existing property, or add the missing property and then recompile the mapping method. Alternatively, if you did not intend the method to be a mapping method, remove the **mapping** option from the method signature.

6187 Cannot access virtual property with no mapping method

Cause

This error occurs if you attempt to access a virtual property for which no corresponding mapping method exists.

Since virtual properties are not physically stored within the parent database object, they require a mapping method in order to manufacture or obtain a value from an external source.

Action

Define a mapping method corresponding to the virtual property, or remove the virtual attribute from the property.

6190 Class is not a subclass of create variable

Cause

This error occurs if you have used a **create** instruction to specify the class of the object that is to be created but that class is not a subclass of the target variable for the **create** instruction.

Action

Change your application code to change the type of the target variable or the class that is being created, so that the two are compatible.

6191 Cannot create an abstract class

Cause

This error occurs if the class you have specified in a **create** instruction is an abstract class. You cannot create instances of abstract classes.

Action

Change the **create** instruction operand so that it represents a real (non-abstract) class.

6192 This is not a class

Cause

This error occurs when the operand is an expression that is not of type **Class**; that is, it does not evaluate to yield a class. The second operand of a **create** instruction must be an explicit class name or an expression that yields a class.

Action

Remove the invalid operand, or correct it so that it yields a valid class.

6193 This is not a member-key dictionary

Cause

This error occurs if your schema file attempts to define a member key for a class that is not a member key dictionary.

Action

Remove the invalid member key definition, or redefine the class as a member key dictionary.

6194 This is not an external-key dictionary

Cause

This error occurs if your schema file attempts to define an external key for a class that is not an external key dictionary.

Action

Remove the invalid external key definition, or redefine the class as an external key dictionary.

6195 Membership must be specified for collection class

Cause

This error occurs if you have defined a collection class but you have not specified the type of its members. A member type specification is mandatory for all collection classes.

Action

Add the membership specification to the type definition in the **typeDefinitions** section of the schema file. The membership is specified by appending the following after the type name at the beginning of the type definition in the schema file.

of member-type-name

6196 Incorrect number of keys

Cause

This error occurs if your application code uses the bracket (`[]`) notation to access elements in a collection but the number of keys provided does not match the keys defined for the collection.

Action

Check the definition of the collection class to determine the required number and types of keys, and then correct the input accordingly.

6197 Must use add method to add a value to a member-key dictionary

Cause

This error occurs if your application code attempts to assign a value to a member key dictionary at a specified key.

By definition, the elements of a member key dictionary are automatically keyed using one of the properties of the element, so you cannot supply an external key with the element. Instead, the `add` method must always be used to add elements to a member key dictionary.

Action

Change your application code to use the `add` method to add the element to the dictionary. If you want to use an external key to access the element, you may need to redefine the dictionary to be an external key dictionary.

6198 This is not a valid locale number

Cause

This error occurs if your schema file contains a locale number that is not valid.

Action

If the schema file was produced by extracting it from JADE and it has not been manually edited, this error should not normally occur.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

6199 Locale not defined in localeDefinitions section

Cause

This error occurs if a locale in the `translatableStringDefinitions` section of the schema file was not first defined in the `localeDefinitions` section.

Action

Add the definition of the locale to the `localeDefinitions` section.

6200 Cannot load Locale Mapping Dialog

Cause

This error occurs if your schema file contains locales that do not match those already in the database and the Locale Mapping dialog is not available. This occurs when loading a production runtime system with none of the JADE classes present.

Action

Ensure that the same set of locales is defined in the first schema file that you load and in all subsequent schema files.

6204 No keys defined for dictionary class

Cause

This error occurs if the schema file defines a dictionary class but does not define any keys for that class.

A dictionary class must always have at least one key.

Action

Add a key definition for the dictionary class.

6205 Invalid key property

Cause

This error occurs if you attempt to specify as a dictionary key a property that is not valid as a key. This can be for one of the following reasons.

- The property can identify a collection rather than a single object reference or primitive attribute.
- The property is a variable length **String** or **Binary** (String Large Object or Binary Large Object) primitive type.
- The property has the virtual attribute.
- The property is a mapped property that is defined in the **RootSchema**.

Action

Change the key definition to refer to a valid property.

6206 Total length of dictionary key or keys exceeds 512 units

Cause

This error occurs if the concatenated length of all the keys that are defined for a dictionary exceeds 512 units (units are bytes, except in Unicode string and character types, where units are Unicode characters).

Action

Remove keys or reduce their lengths until the total key length is 512 units or fewer.

6209 Exception handler must return an Integer

Cause

This error occurs if you attempt to arm a method that does not return an integer as an exception handler. All exception handler methods must return an integer value indicating how the exception is to be propagated.

Action

Change the definition of the exception handler method so that it returns an integer.

Alternatively, arm a different method as the exception handler.

6210 Recursive definition

Cause

This error occurs if you attempt to define a constant in terms of itself. This is not valid.

Action

Change the constant definition to refer only to literal values or to other constants.

6211 Exception class required

Cause

This error occurs if the first operand of an **on Exception** instruction is not the name of an exception class (that is, of the **Exception** class or one of its subclasses).

Action

Change the first operand so that it is the name of an **Exception** class.

6212 Invalid exception handler

Cause

This error occurs if the second operand of an **on Exception** instruction is not the name of a method that is called when the exception is raised.

Action

Change the second operand so that it identifies a method that will handle the exception.

6213 Non exception expression

Cause

This error occurs if the operand of a **raise** exception instruction is not of type **Exception**; that is, it is not an expression that yields an instance of the **Exception** class or one of its subclasses.

Action

Change the operand so that it is of type **Exception**.

6214 Constant names must begin with uppercase

Cause

This error occurs if you attempt to define a constant beginning with a lowercase letter. JADE enforces a syntactic convention that all constant names must begin with an uppercase letter.

Action

Change the constant name so that it begins with an uppercase letter.

6216 Syntax or feature is not yet supported

Cause

This error occurs if you attempt to use syntax or a feature that is planned for possible inclusion in a future release of JADE but is not yet fully supported.

Action

Remove the invalid syntax from your schema file or method source.

6220 Method is checked out to another delta and cannot be loaded

Cause

This error occurs if you attempt to load a method by using a schema file when that method is currently checked out to another delta. As the method is checked out, it is effectively locked by that user and cannot be modified by a schema load.

Action

Sign on to the delta to which the method is checked out, or ask the owner of the delta to check the method in.

6221 Multiple line comment is not terminated

Cause

This error occurs if you begin a multiple line comment with the `/*` character sequence and there is no following `*/` sequence to terminate the comment.

Action

Terminate the comment using the `*/` character sequence.

6222 New patch version is less than current patch version

Cause

This error occurs if you attempt to install a JADE patch release but the version number of the release you are installing is less than the version you are currently running. This check is intended to prevent an older patch release being installed over the top of a newer one.

Action

Verify that you have the correct patch release files and that you are installing into the correct JADE database. Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

6223 Reorg does not support this change

Cause

This error occurs if your schema file attempts to make a change to the schema that is not supported by reorganization. Refer to [Chapter 14](#) of the *JADE Developer's Reference* for a description of the changes that the reorganization does not support.

Action

Correct the schema so that it no longer attempts to make the invalid change. Alternatively, remove all instances of the affected class so that a reorganization is not required.

6224 Invalid operation for external object

Cause

This error occurs if you attempt to create an instance of an external object or attempt to update an external collection. Since external objects and collections are created and maintained automatically by the external database engine, these operations are not valid.

Action

Remove the invalid code.

6225 Cannot define inverse to a virtual property

Cause

This error occurs if the schema file defines a property as being both **explicitInverse** and virtual. Virtual properties cannot participate in inverse relationships.

Action

Remove **virtual** from the property definition, or remove the inverse definition.

6226 Cannot define class map for superschema class

Cause

This error occurs if the **classMapDefinitions** section of your schema load file contains a class map definition for a superschema class. This is not supported.

Action

Remove the class map definition.

6227 Non-mapping method with same name as property

Cause

This error occurs if you attempt to define a non-mapping method with the same name as an existing property.

Action

Perform one of the following actions.

- If you intend the method to be a mapping method, change the method header to conform to the method signature required for mapping methods. This signature is as follows.

```
methodName(set: Boolean; _value: property-type io) mapping;
```

- If you do not intend the method to be a mapping method, rename the method (by deleting and adding it again) so that it does not have the same name as an existing property.
- Delete or rename the existing property.

6228 Cannot change default class map file

Cause

This error occurs if your schema file attempts to redefine the default class map file.

The default map file specifies the name of the database file that is used to contain instances of a class for which no map file has been defined. This error will result if the default map file in the schema file is different from the existing default map file.

Action

Change the class map file specified in the **defaultFileDefinition** section of your schema file so that it matches the existing definition.

Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

6229 Incompatible schema file format

Cause

This error occurs if you attempt to load a schema file whose format is incompatible with the version of JADE you are currently running.

In order to be compatible, the following conditions must apply.

- The version number in the schema file must be at least the minimum version supported by the release of JADE you are running.
- The version number in the schema file must not be greater than the version of JADE you are running. This is intended to prevent errors that may result from the loading of schema files extracted by newer versions of JADE into older systems.

Action

Check the version number in the **jadeVersionNumber** section of the schema file. Verify that you are loading into the correct JADE database. If you are attempting to load an older version that is no longer supported by the release that you are running, create a compatible schema file by loading and extracting the schema using a compatible older version of JADE.

Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

6230 Non-numeric control variable

Cause

This error occurs if a non-numeric control variable is specified in a **foreach** instruction that iterates over a range of numeric values. The following is an example of this error.

```
vars
  s : String;
begin
  foreach s in 1 to 10 do      // error, s is non-numeric
    write s;
  endforeach;
end;
```

Action

Replace the control variable in the **foreach** instruction with a variable of appropriate numeric primitive type. A numeric variable is of type **Integer**, **Real**, **Decimal**, **Date**, **Time**, or **TimeStamp** (for details about these primitive types, see "**Primitive Types**", in Chapter 1 of the *JADE Encyclopaedia of Primitive Types*).

Alternatively, if the **foreach** instruction is intended to iterate over a collection of objects instead of a range of numeric values, modify the *collection-expression* of the instruction to specify a collection of objects.

6232 Same name as reserved word

Cause

This error occurs if you use a JADE reserved word as an identifier name. This is not allowed.

Action

Change the identifier name to a non-reserved word. For a list of reserved words, see "**Reserved Words**", in Chapter 1 of the *JADE Developer's Reference*.

6233 User encryption library not found

Cause

This error occurs when the schema source encryption feature references a schema encryption hook library (DLL) that cannot be found in the executable directory or within the current path.

Action

Ensure that the **SchemaEncryptionHookLibrary** parameter in the [[JadeSecurity](#)] section of the JADE initialization file specifies the correct library name.

If the library name is correct, ensure that the specified DLL is present in the executable directory or within the current path.

6234 Cannot load user encryption entry point

Cause

This error occurs when the encryption option for extracting schema sources is specified but the predefined encryption entry point in the schema encryption hook library cannot be loaded.

Action

Ensure that the library specified by the [SchemaEncryptionHookLibrary](#) parameter in the [[JadeSecurity](#)] section of the JADE initialization file contains the **encryptSchemaSource** entry point.

6235 User routine encryptSchemaSource failed

Cause

This error occurs during a schema extract when the schema source encryption function fails.

Action

Refer to your JADE messages ([jommsg.log](#)) log file for the error status returned by the **encryptSchemaSource** function.

6236 User routine decryptSchemaSource failed

Cause

This error occurs during a schema load when the schema source decryption function fails.

Action

Refer to your JADE messages ([jommsg.log](#)) log file for the error status returned by the **decryptSchemaSource** function.

Check that the [SchemaEncryptionHookLibrary](#) parameter in the [[JadeSecurity](#)] section of the JADE initialization file specifies the correct library for loading the extract file that contains encrypted method source code.

6237 Invalid encrypted header

Cause

This error occurs when the input contains an encrypted source header that is invalid. An encrypted source header must be the following format.

```
_encrypted[n]
```

In this format, **n** is the number of characters that have been encrypted following the header.

This error can also be returned if there is not a valid **_endEncrypted** trailer following the encrypted characters. For example, this could occur if the source file has been changed by editing an encrypted source file and removing or inserting carriage return characters.

Action

Verify that the header is in the correct format. If this is the case, check that the source file has not been changed since it was originally extracted.

6238 Property already defined in a subclass

Cause

This error occurs if your schema file attempts to define a property that already exists in a subclass in either the current schema or a subschema of the current schema.

Action

Remove the duplicated property.

6239 Cannot load user decryption entry point

Cause

This error occurs during the load of a schema file containing encrypted sources when the compiler cannot load the predefined decryption entry point in the schema encryption hook library (DLL).

Action

Ensure that the library specified by the [SchemaEncryptionHookLibrary](#) parameter in the [[JadeSecurity](#)] section of the JADE initialization file contains the **decryptSchemaSource** entry point.

6240 Only RootSchema can define a userHook method signature

Cause

This error occurs if you have defined a user hook method. Only the **RootSchema** can define user hook methods.

Action

Remove the **userHook** option from the method signature.

6241 JadeSchema cannot reimplement a userHook method

Cause

This error occurs if a JADE schema method is reimplementing a user hook method.

Only non-JADE schema methods can reimplement user hook methods.

Action

Remove the method definition, or remove the **userHook** option from the **RootSchema** method that is being reimplemented.

6242 This is not a valid clone of locale

Cause

This error occurs if the schema file defines a locale as a clone of a locale that is not valid; for example, the locale that is to be the basis for your clone may not exist.

Action

Change the locale number specified in the **_cloneOf** clause in the **localeDefinitions** section to refer to a valid locale.

6243 No Application, Global or WebSession subclass defined in superschema

Cause

This error occurs if you attempt to define an application class (subclass of the **Application** class), global class (subclass of the **Global** class), or a Web session class (subclass of the **WebSession** class) and there is no **Application**, **Global**, or **WebSession** subclass defined in the superschema.

Each schema in a schema hierarchy must have an application and global class. The Web session class is optional, but if it is present, all superschemas in the schema hierarchy must also have a Web session class.

Action

Add an **Application**, **Global**, or **WebSession** subclass to the superschema.

6244 Unknown constant

Cause

This error occurs if the input refers to a constant that does not exist.

Action

Correct the constant name, if it is incorrect. If the constant name is correct but the constant has not been defined, define it by using the Class Browser or by using a schema file.

6245 Database file number must be specified

Cause

This error occurs if a database file definition in the bootstrap schema omitted the specification of a file number. All database files defined in the bootstrap schema must have a file number defined.

Action

Change the database file specified in the **databaseFileDefinitions** section of your bootstrap schema file to specify a file number using the **number = file-number** option.

6246 Invalid database file number

Cause

This error occurs if your schema file defines a database file with a file number that is not valid; for example, the file number may not be in the range allocated for user systems.

Action

Change the database file number specified using the **number** option in the **databaseFileDefinitions** section of your schema file to refer to a valid file number. If the number is already allocated to a different file, the next available file number is used.

Alternatively, remove the **number** option to automatically use the next available file number.

Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

6247 Cannot start name with underscore character

Cause

This error occurs if the input declaration name starts with an underscore character. As this may cause a conflict with system-defined identifiers, it is not permitted.

Action

Remove the underscore character.

6248 Cannot remap system database file

Cause

This error occurs if your schema file attempts to define a remap file for a system database (map) file (that is, one that is defined in the **RootSchema**).

Action

Remove or correct the invalid file name.

6249 Incompatible foreach option

Cause

This error occurs if an incompatible option is specified in the **foreach** instruction.

Action

Change the options to make them compatible.

6250 Cannot define exclusive abstract collection

Cause

This error occurs if the schema file defines a reference as being an exclusive abstract collection type.

Exclusive references are always created when the parent object is created, so they must have a real (non-abstract) type.

Action

Change the type of the reference to a real (non-abstract) collection. This is likely to be a concrete subclass of the original abstract collection.

Alternatively, make the reference to the abstract collection a shared reference rather than an exclusive reference.

6251 Constant value has changed

Cause

This error occurs when the value of a constant has been changed but methods that reference the changed item have not yet been recompiled.

Action

Recompile the current method and correct any errors that result from the change of value.

6252 Schema load is not in restart state

Cause

This error occurs during the schema load when the compiler is called, indicating that the previous load should be restarted, but no restart information exists for the specified schema.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

Reset the schema load restart information by checking the **Clear Restart** check box on the Load Schema dialog.

6253 Invalid text at schema file restart position

Cause

This error occurs during the schema load when the compiler attempts to restart a previous load but the text at the restarting position in the schema file does not match the text saved in the restart information.

The check is performed to prevent integrity problems that might result from restarting a schema file load with a changed schema file.

Action

Restore the original schema file and then restart the load.

If the load cannot be successfully restarted, the schema may be in an inconsistent state. You should delete the schema and load a complete definition of the schema from a schema file.

For deployed systems, check the **Clear Restart** check box on the Load Schema dialog to reset the restart information before loading the complete definition schema file.

6254 Type of method has changed

Cause

This error occurs when an external method has been changed to be a JADE method, or a JADE method has been changed to be an external method but methods that reference the changed item have not yet been recompiled.

Action

Recompile the current method and correct any errors that may result from the changed method.

6255 Cannot load JADE patch release using JADE thin client

Cause

This error occurs if you attempt to install a JADE patch release while running JADE in thin client mode. This is not allowed.

Action

Use the Schema Load utility (**jadload.exe**) to install JADE patch releases.

6256 Property has changed

Cause

This error occurs when the length or access mode of a property has been changed but methods that reference the property have not yet been recompiled.

Action

Recompile the current method and correct any errors that result from the changed property.

6257 Keys have changed

Cause

This error occurs when the keys of a collection have been changed but methods that reference the collection have not yet been recompiled.

Action

Recompile the current method and correct any errors that result from the changed keys.

6258 Cannot update compound expression

Cause

This error occurs when an updating method has been called for an operand value that is the result of a compound expression; for example, consider the following statement.

```
(s1 & s2).firstCharToUpper;
```

As **(s1 & s2)** does not identify a storage location that can be updated, this syntax is invalid.

Action

Change your application code to replace the compound expression with a simple expression that can act as the receiver of an updating method. In some cases, it may be possible to change the method being called from updating to non-updating (if it does not actually update the receiver).

6259 Length must be positive

Cause

This error occurs if the length of a primitive type variable is not positive. The following code fragment shows this error.

```
vars
    maxLength = 50;
constants
    s: String[MaxLength-100];    // error: negative length not valid
```

Action

Change the length to a positive integer value.

6260 Scale factor cannot be negative

Cause

This error occurs if the scale factor of a **Decimal** variable is negative. In the definition of a **Decimal** variable, the precision is the first value within the brackets (**[]**) and the scale factor is the second.

The following code fragment shows this error.

```
vars
    d: Decimal[12, -4];    // error: negative scale factor not valid
```

Action

Change the scale factor to an integer value in the range **1** through **23**, not greater than the precision.

6261 JADE patch release decryption failed

Cause

This error occurs during a load of a JADE patch release when the internal patch decryption function fails. It indicates a corrupt or changed patch file.

Action

Ensure that you are loading an authorized patch file as released by your JADE support center. Alternatively, contact your local JADE support center or JADE Support, if your JADE licenses include support.

6262 JADE patch release must be encrypted

Cause

This error occurs if you attempt to load a JADE patch release that has not been encrypted. This is not permitted.

Action

Ensure that you are loading an authorized patch file as released by your JADE support center. Alternatively, contact your local JADE support center or JADE Support, if your JADE licenses include support.

6263 Cannot execute this function from a non-system node

Cause

This error occurs if you attempt to execute an internal function from a non-system node. This is not permitted.

Action

Ensure that you are attempting to execute the correct function and that the function is valid to be executed from a non-system node. Alternatively, this error should not normally occur, and you should contact your local JADE support center or JADE Support, if your JADE licenses include support.

6264 Cannot define inverse to a dynamic dictionary

Cause

This error occurs if the schema file defines a property of type [DynaDictionary](#) as an explicit inverse reference. Dynamic dictionaries cannot participate in inverse relationships.

Action

Remove the inverse definition.

6265 Invalid use of variable parameter list

Cause

This error occurs if a variable parameter list (pseudotype **ParamListType** or **KeyType**) has been used in a context in which it is not valid. This can be for one of the following reasons.

- Only one variable parameter list is permitted for each method, and this parameter list must occur as the first or the last item defined in the formal parameters.
- Variable parameter lists cannot be passed to external functions.

Action

Correct the use of the variable parameter list so that it conforms to the above rules.

6266 Signature of base control method has changed

Cause

This error occurs when the signature of a control method has been changed but its event method reimplementations have not yet been recompiled.

Action

Correct the signature of the current method to match the base control definition, if necessary. Then recompile the method.

6267 Invalid option value

Cause

This error occurs if the schema file defines an option that requires an associated value and the value is not valid for the option.

Action

This error should not normally occur if the schema file was extracted correctly by JADE.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

6268 Type must be specified

Cause

This error occurs if you attempt to define a constant with the value **null** and you do not include a type specification. Every constant must have a defined type, and if no type is specified, the compiler assumes the type of the constant expression. However, in the case of **null**, no type can be assumed.

The following code fragment is an example of this error when defining a method constant.

```
constants
    NullName = null;           // error, no type defined
```

Action

Specify the type of the constant, as shown in the following example.

```
constants
    NullName : String = null;
```

6269 Hex byte exceeds maximum byte value

Cause

This error occurs if any of the component bytes of a hexadecimal binary literal exceed the allowable range for byte values.

Hexadecimal binary literals in JADE are represented as a sequence of space-separated hexadecimal characters, enclosed in square brackets; for example:

```
DaysInMonth = #[1F 1C 1F 1E 1F 1E 1F 1F 1E 1F 1E 1F];
```

The allowable range for byte values is **00** through **FF**.

The following code fragment is an example of this error when defining a method constant.

```
constants
    InvalidBin = #[41 42 430];    // error, invalid hex byte value 430
```

Action

Change the hexadecimal byte so that it is within the permitted range. Insert spaces if they have been accidentally omitted between bytes.

6270 Binary delimiter mismatch

Cause

This error occurs if your entry contains an opening hexadecimal binary delimiter (left square bracket (**[**)) with no closing delimiter (right square bracket (**]**)).

Hexadecimal binary literals in JADE are represented as a sequence of space-separated hexadecimal characters, enclosed in square brackets; for example:

```
DaysInMonth = #[1F 1C 1F 1E 1F 1E 1F 1F 1E 1F 1E 1F];
```

You may have omitted the closing delimiter, or you may have placed it in a comment.

Action

Determine the correct placement of the closing delimiter and then change the input accordingly.

6271 Invalid exception handler receiver

Cause

This error occurs if you attempt to arm an exception handler method defined on a primitive type. This functionality is not supported. The receiver of an exception must be an object.

Action

Arm an exception handler method defined on a class.

6272 External function already defined in another schema

Cause

This error occurs if your schema file attempts to define an external function that is already defined in either a superschema or a subschema.

Action

Remove the duplicated external function definition.

6273 Cannot change class to abstract

Cause

This error occurs if the schema file attempts to change a class from real to abstract and the change is not valid. This can be for one of the following reasons.

- The class has persistent instances existing.
- The class is a collection class and there are existing exclusive properties referring to this collection type. Exclusive references are always created when the parent object is created, so they must have a real (that is, non-abstract) type.

Action

Refer to your JADE messages (**jommmsg.log**) log file for more details about this error.

Remove any persistent instances before changing the class to abstract.

For collection classes referenced by exclusive properties, change the type of the collection to real. Alternatively, make the reference to the abstract collection a shared reference rather than an exclusive reference.

6274 Unknown condition

Cause

This error occurs if the input refers to a condition that does not exist.

Action

Correct the condition name, if it is incorrect. If the condition name is correct but the condition has not been defined, define it by using the Class Browser, or by using a schema file.

6275 Invalid constraint

Cause

This error occurs if you have defined a constraint on a reference and the constraint is not valid. This can be for one of the following reasons.

- The reference is not an explicit inverse reference.
- The schema type of the constraint is not compatible with the reference type. The reference type (or its member type for collections) must be the same as or a subclass of the constraint schema type.
- The constraint must be a JADE condition method with no parameters.

Action

Correct the constraint so that it conforms to the above rules.

6276 Invalid condition signature

Cause

This error occurs if you have defined a condition method that does not conform to the required signature for condition methods.

Condition methods must have constant parameters (if any) of primitive type, a **Boolean** return primitive type, and be non-updating and non-abstract.

Action

Change the condition method header to conform to the rules for the signature of condition methods, above. If you did not intend the method to be a condition method, remove the **condition** option from the method signature.

6277 This is not allowed in conditions

Cause

This error occurs if there is an attempt to use a JADE language feature that is not allowed in condition methods. Condition methods must have the following syntax.

```
condition-name(optional-parameters): Boolean condition;  
[constants  
  constant-declarations]  
begin  
  condition-instructions;  
end;
```

The *condition-instructions* can specify only the **if** and **return** instructions.

In addition, a condition method cannot perform the following.

- Call (non-condition) methods
- Call condition methods on object references (except **self**)
- Reference properties using a path (for example, **myCompany.name**)

- Reference mapped properties
- Reference system variables (apart from **self**)
- Reference system properties (for example, the **text** property for the **Customer** class)
- Call itself (directly or indirectly)
- Reference an element in an array (for example, **results[4]** or equivalently **results.at(4)**)
- Reference an object in a dictionary using its key value (for example, **customers["Jones"]** or the equivalent **customers.getKey("Jones")**)

This error can also occur for a method with the **conditionSafe** method option. A **conditionSafe** method is often used for a virtual property in an RPS mapping.

Action

Change the condition method to conform to the above rules. If you do not intend the method to be a condition method, remove the **condition** option from the method signature.

6278 Maximum block size too large

Cause

This error occurs if the schema file defines the maximum size of a collection block as a value exceeding the value defined by the **MaximumCollectionBlockSize** global constant in the **SystemLimits** category.

Action

Change the maximum block size so that it is within the permitted limit.

6279 Member type size too large

Cause

This error occurs if the schema file defines the member type size of a collection class as a value exceeding 16,000 bytes in an ANSI environment or 32,000 bytes in a Unicode environment.

Action

Change the size of the member type so that it is within the permitted limit.

6280 Cannot assign to meta object

Cause

This error occurs if there is an attempt to assign a value to a meta object (for example, the property object **Customer::name**). This is not valid, as a meta object does not identify a storage location that can act as the target of an assignment.

Action

Change your application code to remove or correct the invalid assignment.

6281 Structural changes are not allowed

Cause

This error occurs when loading a schema file containing structural changes (for example, a property to be added, deleted, or changed) and the changes are not permitted. This can be for one of the following reasons.

- The schema is being loaded into the current version of the schema
- Running **jadloadb** with the **noStructureChanges** option set to **true**
- A reorganization is currently in progress (that is, a reorganization has started but not completed)

Action

If you require structural changes to your schema, load the schema file when none of the conditions above applies. Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

6283 Cannot load JADE patch release for non-existent schema

Cause

This error occurs if you attempt to install a JADE patch release for a schema that does not exist; for example, you cannot install a patch release for **JadeSchema** in a deployed system.

Action

JADE patch releases can be loaded into systems only where the schema to be patched already exists. Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

6284 Invalid qualified identifier

Cause

This error occurs if an identifier has been incorrectly qualified using the **::** syntax. Classes in packages and properties and methods can be qualified by **::** to reference the metaschema object. For example, **Customer::name** refers to the **name** property object of the **Customer** class. However, **Customer::name::x** would not be a valid identifier.

Action

Correct or delete the invalid identifier.

6285 Number too big

Cause

This error occurs when the input contains a numeric literal that exceeds the maximum allowed value. In JADE, the maximum value of a numeric literal is represented by a decimal number with a limit of 23 digits before the decimal point.

Action

Reduce the number of digits before the optional decimal point to 23 or fewer.

6286 Invalid membership

Cause

This error occurs if you attempt to specify the membership of a collection and the membership is not valid; for example, for a non-primitive collection, the member type must be a subclass of the member type of the superclass of the collection.

Action

Change the membership definition to refer to a valid member type.

6287 Cannot remove condition option - condition used as a constraint

Cause

This error occurs if you attempt to remove the **condition** option from a method signature and the condition method is used as a constraint on a reference.

Action

Before removing the **condition** option, update the associated conditional reference definitions so that the method is no longer used as a constraint.

6288 Invalid update mode for constraint

Cause

This error occurs if you attempt to define an inverse for a conditional reference and the update mode is not **automatic**. Constraints can be specified only on automatic references and not on **manual** or **manual/automatic** references.

Action

Change the inverse definition so that the update mode of the conditional reference is automatic.

6289 Cannot specify systemOnly option for user schema entity

Cause

This error occurs if you attempt to specify the **systemOnly** option for a user schema entity. This option is reserved for internal, system-only use.

Action

Remove the **systemOnly** option from the schema entity definition.

6290 Cannot specify unpublished option for user schema entity

Cause

This error occurs if you attempt to specify the **unpublished** option for a user schema entity. This option is reserved for internal, system-only use.

Action

Remove the **unpublished** option from the schema entity definition.

6291 Cannot specify subschemaHidden option for system schema entity

Cause

This error occurs if you attempt to specify the **subschemaHidden** option for a system schema entity. This option is reserved for user schemas only.

Action

Remove the **subschemaHidden** option from the schema entity definition.

6292 Incompatible lifetime option - check superclass lifetime restrictions

Cause

This error occurs if the lifetime you have specified in a class definition is not compatible with the lifetime options of its superclass. For example, a class cannot specify the **persistentAllowed** option if its superclass does not have the **subclassPersistentAllowed** option set.

Action

Correct the class definition so that the lifetimes are compatible.

6293 Cannot define external condition

Cause

This error occurs if you attempt to specify the **condition** option for an external method. User schemas are not permitted to define external condition methods.

Action

Remove the **condition** option from the method signature.

6294 This is not allowed in report scripts

Cause

This error occurs if you have attempted to use a JADE language feature that is not allowed in JADE Report Writer scripts.

The following JADE language features are not allowed in report scripts.

- Transaction control instructions (for example, [beginTransaction](#))
- Exception handling instructions (for example, [on Exception](#))
- The [terminate](#) instruction
- The [create](#) and [delete](#) instructions
- Accessing methods and properties on non-JADE Report Writer classes, except for system features on the application and environmental classes ([app](#), [global](#), [node](#), [process](#), [system](#), and [currentSession](#))

Action

Change the JADE Report Writer script to conform to the above rules.

6295 Patch control security check failed - refer to the JADE log file

Cause

This error occurs during a schema load when your development patch control security routine disallows the updating of a schema entity.

Action

Refer to your JADE messages ([jommsg.log](#)) log file for more details about the attempted update operation.

6296 Package already defined

Cause

This error occurs the schema file attempts to define a package that is already defined within the schema branch.

Action

Remove the duplicated package definition.

6297 Unknown package

Cause

This error occurs if the schema file refers to an exported package that does not exist.

Action

Correct the package name, if it is incorrect. If the package name is correct but the package has not been defined, define it by using the Packages Export Browser or by using a schema file.

6299 Cannot add parameters - condition used as a constraint

Cause

This error occurs if you attempt to add a parameter to the signature of a condition method and the condition method is used as a constraint on a reference. Only condition methods with no parameters can be used as a constraint on a reference.

Action

Before you can add a parameter, you must update the associated conditional reference definitions so that the condition method is no longer used as a constraint.

6300 Receiver type is not an imported class

Cause

This error occurs if you attempt to use the **importMethod** keyword to call an imported method and the type of the receiver is not an imported class. The type of the receiver object of an **importMethod** call must be a class in an imported package.

Action

Change your application code to change the type of the receiver to an imported class, possibly using a type cast.

6301 Cannot define translatable string in terms of type constant

Cause

This error occurs if you attempt to define a translatable string in terms of a class constant or primitive type constant. This is not supported.

Action

Change the translatable string so that it does not refer to class or primitive type constants.

6302 Ambiguous imported class

Cause

This error occurs if you attempt to reference an imported class that has been imported from more than one package.

Action

Prefix the imported class name with the imported package name; for example:

```
Logging::LogManager
```

6303 Cannot import package - class already imported from another package - see jommsgs

Cause

This error occurs if you attempt to import a package containing a class that has already been imported from another package. To prevent potentially ambiguous or conflicting local method definitions, this is not allowed.

The name of the imported package and class can be found in your JADE messages (**jommsg.log**) log file.

Action

Remove the conflicting package definition.

6304 Method cannot be final -- has implementors

Cause

This error occurs if you attempt to specify the **final** option for a method and the method has been reimplemented in a subclass. This is not permitted, because the **final** option specifies that the method cannot be reimplemented.

Action

Remove the **final** option from the method signature. Alternatively, remove all reimplementations of the method.

6305 Method cannot be subschemaFinal -- has subschema implementors

Cause

This error occurs if you attempt to specify the **subschemaFinal** option for a method and the method has been reimplemented in a subschema. This is not permitted.

Action

Remove the **subschemaFinal** option from the method signature. Alternatively, remove all reimplementations of the method in subschemas.

6306 Cannot reimplement a final method

Cause

This error occurs if you attempt to reimplement a method that has the **final** option specified. This is not permitted, because the **final** method option specifies that the method cannot be reimplemented.

Action

If you want to reimplement the method, remove the **final** option from the signature of the super-method.

6307 Cannot reimplement a subschema final method in a subschema

Cause

This error occurs if you attempt to reimplement a method that has the **subschemaFinal** option specified in a superschema. This is not permitted, because the **subschemaFinal** method option specifies that the method cannot be reimplemented in a subschema.

Action

If you want to reimplement the method, remove the **subschemaFinal** option from the signature of the super-method.

6308 Cannot create instance with this lifetime -- check class lifetime restrictions

Cause

This error occurs if the lifetime specified in a **create** instruction is not permitted by the lifetime options defined for the class.

Action

Change your application code or the class definition so that the lifetimes are compatible.

6309 Class cannot be final -- subclasses exist

Cause

This error occurs if you attempt to specify the **final** option for a class and subclasses of that class exist. This is not permitted, because the **final** class option specifies that the class cannot be subclassed.

Action

Remove the **final** option from the class definition. Alternatively, remove all subclasses of the class.

6310 Class cannot be subschemaFinal -- subclasses exist in subschema

Cause

This error occurs if you attempt to specify the **subschemaFinal** option for a class and subclasses of that class exist in a subschema. This is not permitted, because the **subschemaFinal** class option specifies that the class cannot be subclassed in a subschema.

Action

Remove the **subschemaFinal** option from the class definition. Alternatively, remove all subclasses of the class in subschemas.

6311 Cannot reimplement method in a subschema copy class

Cause

This error occurs if you attempt to reimplement a method in a subschema copy class and the method has been defined with the **subschemaCopyFinal** option. This is not permitted.

The **subschemaCopyFinal** option prevents superschema subclasses from inheriting subschema reimplementations, which can potentially cause unexpected behavior; for example, reimplementing the **getAtKey** method from the **Dictionary** class in a **MemberKeyDictionary** subschema copy class is not permitted.

Under normal circumstances, you cannot reimplement a **subschemaCopyFinal** method in a subschema copy class. A warning message box is displayed before the method can be added, to inform you that the action is not permitted.

Action

Remove the invalid method definition. It may be possible to reimplement the method in a subclass rather than a subschema copy class.

6312 Cannot load an exported package definition into a system schema

Cause

This error occurs if you attempt to define an exported package in a system schema. This is not permitted.

Action

Load the schema file into a user (non-system) schema. Do this by specifying a user schema in the **schemaDefinition** section of the schema file.

6313 Cannot load an imported package definition into a system schema

Cause

This error occurs if you attempt to define an imported package in a system schema. This is not permitted.

Action

Load the schema file into a user (non-system) schema. Do this by specifying a user schema in the **schemaDefinition** section of the schema file.

6315 Cannot specify html option for non-HTML class

Cause

This error occurs if your schema file specifies the **html** option for a property in a class that is not a subclass of the **JadeHTMLClass** class. This is not allowed.

Action

Remove the **html** option from the property definition. Alternatively, redefine the class as a subclass of the **JadeHTMLClass** class.

This error should not normally occur if the schema file was extracted correctly by JADE.

If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6316 Invalid option or comma missing between options

Cause

This error occurs if you specify an option that is not valid for the element that you are defining. This error can also result from specifying two or more options without separating them with a comma.

When you define types, methods, and certain other elements in a JADE schema file or method, the definition can be followed by a list of options.

These options are in the form of a list of modifiers separated by commas, and they specify additional attributes of the item being defined. Examples include the **protected**, **updating**, and **abstract** options on a method definition (for details, see "[Method Options](#)", in Chapter 1 of the *JADE Developer's Reference*) and the **transient** option on a class definition.

This error can also result from a misplaced identifier or some other syntactic error, causing the compiler to interpret an identifier as an option when an option was not intended.

Action

Correct the invalid option or correct the syntax if this is the cause of the error.

6317 Method cannot be subschemaCopyFinal -- has subschema copy implementors

Cause

This error occurs if you attempt to specify the **subschemaCopyFinal** option for a method and the method has been reimplemented in a subschema copy class. This is not permitted.

Action

Remove the **subschemaCopyFinal** option from the method signature. Alternatively, remove all reimplementations of the method in subschema copy classes.

6318 Invalid database file name

Cause

This error occurs if your schema file defines a database file with a blank file name. A valid file name cannot consist entirely of space characters.

Action

Change the database file name to a valid (non-blank) file name.

6319 Duplicated feature number

Cause

This error occurs if you attempt to define a class feature with the same number as an existing feature. This error is for system schemas only and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

6320 Class options have changed

Cause

This error occurs when the options of a class have been changed but methods that reference the class have not yet been recompiled.

Action

Recompile the current method and correct any errors that result from the changed options.

6321 Expecting TranslatableString name

Cause

This error occurs if a translatable string name is expected but one is not provided.

In the following example, an attempt is made to add a translatable string with an invalid name **greeting** (translatable strings must begin with an uppercase letter), using the [addCompileTranslatableString](#) method of the [Schema](#) class.

```
currentSchema.addCompileTranslatableString('greeting="Hello World"',  
                                           errCode, errOffset, errLength);
```

Action

Correct your code.

6322 webService option must be specified for a Web Services method

Cause

This error occurs if the schema file contains a Web service method definition that omits the [webService](#) option. This option must be specified for Web service methods.

Action

Add the **webService** option to the Web service method definition.

This error should not normally occur if the schema file was extracted correctly by JADE. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6323 `webService` option can only be specified for a Web Services method

Cause

This error occurs if the schema file contains a JADE or external method definition that includes the `webService` option. This option can be specified only for Web service methods.

Action

Remove the `webService` option from the method definition.

This error should not normally occur if the schema file was extracted correctly by JADE. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6324 This is not a Web Services class

Cause

This error occurs if your schema file attempts to specify the value of a Web service property for a class that is not a Web service class.

Action

Remove the invalid property specification.

This error should not normally occur if the schema file was extracted correctly by JADE. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6325 This is not a Web Services method

Cause

This error occurs if your schema file attempts to specify the value of a Web service property for a method that is not a Web service method.

Action

Remove the invalid property specification. This error should not normally occur if the schema file was extracted correctly by JADE. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6326 Cannot reimplement a condition

Cause

This error occurs if you attempt to reimplement a condition; that is, a method that has the `condition` option specified. This is not permitted, since condition methods cannot be called polymorphically.

Action

If you want to reimplement the method, remove the `condition` option from the signature of the super-method.

6327 Validation access check failed

Cause

This error is raised by the `validateMethod` method of the `JadeMetadataAnalyzer` class when a user-defined security check fails.

Action

Change your method code to remove the invalid access. For example, the following methods check that input method source code does not reference the:

- `beginTransaction` instruction
- `Employee` class
- `deposit` method of the `Account` class

First, create a subclass of the `JadeMetadataAnalyzer` class (called `MyAnalyzer`) and reimplement the `canAccessStatement` and `canAccessSchemaEntity` methods, as follows.

```
canAccessStatement(statement: String; err: String output): Boolean;
begin
    if statement = Statement_BeginTransaction then
        err := 'transactions are not allowed';
        return false;
    endif;
    return true;
end;

canAccessSchemaEntity(se: SchemaEntity; err: String output): Boolean;
begin
    if se.isKindOf(Class) then
        if se.name = 'Employee' then
            err := 'access to class is not allowed';
            return false;
        endif;
    elseif se.isKindOf(Method) then
        if se.Method.schemaType.name = 'Account' and se.name = 'deposit' then
            err := 'access to method is not allowed';
            return false;
        endif;
    endif;
    return true;
end;
```

The method in the following example then validates the method source.

```
checkMethod(source: String; schemaType: Type);
vars
    analyzer      : MyAnalyzer;
    err, pos, len : Integer;
    msg           : String;
begin
    create analyzer;
    analyzer.validateMethod(source, schemaType, currentSchema, err, pos,
                           len, msg);
```

```
if err = 0 then
    write 'method validated successfully';
else
    write 'method validation failed - ' & process.getErrorText(err);
    if err = JadeMetadataAnalyzer.AccessCheckFailed then
        write msg;
    endif;
endif;
epilog
    delete analyzer;
end;
```

6328 Schema default locale cannot be clone

Cause

This error occurs if the schema file contains a locale definition that has both **schemaDefaultLocale** and **cloneOf** clauses.

Action

Remove the **schemaDefaultLocale** clause. This error should not normally occur if the schema file was extracted correctly by JADE. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6329 Schema default locale already defined

Cause

This error occurs if the schema file contains more than one locale definition that is defined with **schemaDefaultLocale**.

Action

Remove the **schemaDefaultLocale** clause from all but one of the locale definitions. This error should not normally occur if the schema file was extracted correctly by JADE. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6330 Schema default locale is undefined

Cause

This error occurs if the schema file does not contain one locale definitions which is defined with **schemaDefaultLocale**.

Action

Add the **schemaDefaultLocale** clause to one of the non-clone locale definitions. This error should not normally occur if the schema file was extracted correctly by JADE. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6331 Parameter count does not match default locale translation

Cause

This error occurs if a translation of a translatable string does not contain the same number of parameters as the translation for the default locale of the schema.

Action

Change the parameter list to match. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6332 Default locale has no translation of this string

Cause

This error occurs if the default locale of the schema does not have a translation of the translatable string that you are attempting to compile.

Action

Delete the translatable string and then recreate it to ensure that it is correctly added to all locales. This error should not normally occur if the schema file was extracted correctly by JADE. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6333 Schema load was missing TranslatableString source for this locale

Cause

This error occurs if the schema file does not contain a translation for the locale of a newly added translatable string.

Action

Update the translation to include appropriate source and then compile. This error should not normally occur if the schema file was extracted correctly by JADE. If you require assistance and your JADE licenses include support, contact your local JADE support center or JADE Support.

6334 Translatable string already defined in another schema

Cause

This error occurs if a translatable string already exists in the schema hierarchy with the given name. It can be in a schema above or below the current one.

Action

Choose a different name for the translatable string.

6335 Cannot add a mapping method to a subschema hidden property

Cause

This error occurs if you attempt to create a mapping method for a property that is declared in a superschema and marked as subschema-hidden.

Action

If you want to implement the mapping method then remove the subschema-hidden attribute from the property.

6336 Cannot add a mapping method to a property used in a condition

Cause

This error occurs if you attempt to create a mapping method for a property that is referred to in a condition method.

Action

Perform one of the following actions.

- If you intend the method to be a mapping method, remove the reference to the property in the condition method.
- If you do not intend the method to be a mapping method, rename the method (by deleting and adding it again) so that it does not have the same name as an existing property.

6337 Cannot change the exporter schema or exported package of an imported package

Cause

This error occurs if you attempt to change the exporting schema name or the exported package name associated with an imported package during a schema load.

Action

Load a schema that deletes the imported package and then load a schema that adds the imported package with the new settings.

6338 Unknown interface

Cause

This error occurs if the input refers to a JADE interface that does not exist.

Action

Correct the JADE interface name if it is incorrect. If the interface name is correct but the interface has not been defined, define it by using the JADE Interface Browser or by loading a schema file.

6339 Same name as existing interface

Cause

This error occurs if you attempt to define a constant or type with the same name as an existing JADE interface.

Action

Rename the constant or type that you are attempting to define, or delete the existing interface.

6340 Method signature conflicts with implemented interface method

Cause

This error occurs if the signature of the implementation of an interface method is not compatible with the signature of the interface method as specified in the JADE interface.

Action

Ensure that the signature of the implemented method is compatible with the signature of the interface method as specified in the JADE interface.

6341 Signature of interface method has changed

Cause

This error occurs when the signature of a JADE interface method has been changed but methods that reference the changed method have not yet been recompiled.

Action

Change the parameters on the call to the changed method, if necessary. Then recompile the current method.

6342 Attempted recursive call of the compiler

Cause

This error occurs when a process attempts to invoke the compiler when it is already running. Attempting to debug the [canAccessStatement](#) or [canAccessSchemaEntity](#) method of the [JadeMetadataAnalyzer](#) class also causes this error.

Action

Do not run the debugger against methods that invoke the compiler. Do not create transient methods in a process that is executing the compiler, for example, by using the [JadeMetadataAnalyzer](#) class.

6343 Cannot specify clientExecution or serverExecution on primitive methods

Cause

This error occurs when the signature of a method on a primitive type specifies **clientExecution** or **serverExecution**. Methods on primitive types are always executed in the node of the calling method.

Action

Change the signature of the method to remove **clientExecution** and **serverExecution** and recompile the method.

6344 Web Service method cannot return type Any

Cause

This error occurs when the return type of a Web service method is type **Any**.

Action

Change the return type of the method to specify the type returned and recompile the method.

6345 Web Service method cannot have parameters of type Any

Cause

This error occurs when a parameter of a Web service method is type **Any**.

Action

Change the parameters of the method to specify the type passed and recompile.

6346 Interface method mapping not defined for this class

Cause

This error occurs when class implementing a JADE interface does not include all methods in the JADE interface in the implementation specification.

Action

Change the implementation specification for the class to include all methods defined in the JADE interface.

6347 Superinterface has not been implemented for this class

Cause

This error occurs when the class implementing a JADE interface does not implement a JADE interface that the original interface extends.

Action

Change the class to implement all JADE interfaces that the original interface extends.

6348 Cannot delete a method which is checked out (see jommsgs)

Cause

This error occurs if you try to delete a method that has been checked out. The name of the method can be found in your JADE messages (**jommsg.log**) log file.

Action

Undo the method check out or do not delete the method.

6349 Method already defined in superinterface or subinterface

Cause

This error occurs when the method name being defined in a JADE interface already exists in a sub-interface (an interface that extends this one) or a super-interface (an interface that this one extends).

Action

Change the method name to be unique.

6350 Expression type should be ApplicationContext

Cause

This error occurs if the expression does not return an instance of type [ApplicationContext](#).

Action

Correct the expression.

6351 Cannot specify a method name here

Cause

This error occurs if the expression cannot include method calls.

Action

Remove any method calls from the expression.

6352 Constants or TranslatableStrings were left with errors (see jommsgs)

Cause

This error occurs when one or more constants or translatable strings were left in error after the schema load. The entities in error are listed in your JADE messages (**jommsg.log**) log file.

Action

Fix the entities that are in error and extract the schema file again.

6353 Preceding locale TranslatableString parameter count does not match default locale

Cause

This error occurs when the parameter count for this translatable string in the default locale differs from the parameter count for the same translatable string in another locale already loaded.

Action

Correct the schema file and load it again.

6354 Structural changes are not allowed - reorg in progress

Cause

This error occurs if you attempt to load a schema file that includes structural changes while a database reorganization is in progress.

Action

Complete the reorganization and then load the schema file.

6355 Cannot delete entity - see jommsgs

Cause

This error occurs if an attempt to delete an entity by using a JADE command **.jcf** file failed.

Action

Examine your JADE messages (**jommsg.log**) log file to determine why the delete operation failed and then take appropriate action.

6356 Invalid conditionSafe signature

Cause

This error occurs if the [conditionSafe](#) method option is detected on a method that is not a virtual property mapping method.

Action

Remove the **conditionSafe** option from the method signature.

6358 ConditionSafe only allowed on user virtual property mapping methods

Cause

This error occurs if a user method has the [conditionSafe](#) option in its signature and the method is not a mapping method for a virtual property.

Action

Remove the **conditionSafe** option from the method signature.

6361 Schema load disallowed - reorg or logical certify is active

Cause

This error occurs if you attempt a schema load into the latest version of the schema or to run the logical certifier while a reorganization is in progress.

Action

Retry the schema load or rerun the logical certifier when the reorganization has completed.

6362 Schema load disallowed - reorg is incomplete

Cause

This error occurs if you attempt a schema load into the latest version while a reorganization is incomplete.

Action

Restart or abort the reorganization then attempt the schema load again. For details, see "[Restarting Reorganizations](#)", in Chapter 14 of the *JADE Developer's Reference*.

6364 Method exists in a superclass in the latest schema context

Cause

This error occurs if an attempt to add a method to a class in the current schema context failed because the name was the same as a method in a superclass in the latest schema context.

Action

Rename the method.

6365 Method exists in a subclass in the latest schema context

Cause

This error occurs if an attempt to add a method to a class in the current schema context failed because the name was the same as a method in a subclass in the latest schema context.

Action

Rename the method.

6366 Class marked as webService but is not in an ExposedList

Cause

This error occurs during a schema load when a Web service class definition is incomplete.

This error should not normally occur if the schema file is produced by extracting from JADE. Errors can be introduced by manually editing the file in an incorrect way.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

6367 Constant needs compile

Cause

This error occurs when a method or constant refers to another constant that has not been successfully compiled.

Action

Recompile the referenced constant.

6368 Class already exists in the latest schema context

Cause

This error occurs if an attempt is made to add a class to the current schema context when a class with same name already exists in the latest schema context.

Action

Rename the class.

6369 Sending method is checked out to another delta and cannot be updated

Cause

This error occurs if a delta is set when compiling a method and a sending method must be recompiled and that sender method is already checked out.

Action

Check in the sending method.

6370 Implementing method is checked out to another delta and cannot be updated

Cause

This error occurs if a delta is set when you compile a method that is reimplemented in a subclass and the reimplementation, which must be recompiled, is already checked out.

Action

Check in the reimplementing method.

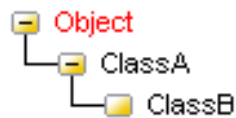
6371 Method exists with a different signature in a subclass in another schema

Cause

This error occurs if the method being compiled, which is defined on a subschema copy class, does not have a compatible signature with a method of the same name which exists in a subclass of the subschema copy class in some other schema.

The other method exists in the schema containing the root type of method being compiled or in one of its subschemas, as shown in the following example:

In the following example of a **SomeSchema** schema, **ClassB** implements a **meth(): String;** method.



In a subschema of the **SomeSchema** schema in the following example, **ClassA** attempts to implement a **meth(): Boolean;** method.



Action

Change the method signature to be compatible with the signature of the existing method.

6372 Cannot import package - exported by a superschema or subschema

Cause

This error occurs if you attempt to import a package from a subschema, which is not allowed.

You do not need to import a package from a superschema, as the classes are available through the normal class inheritance mechanism.

Action

Do not import the package.

6373 Invalid UNICODE character value

Cause

This error occurs if a **StringUtf8** literal contains a hexadecimal value that does not correspond to a valid Unicode code point (character). Refer to the "*The Unicode Standard (version 4.1)*", which you can download from <http://www.unicode.org>.

Action

Correct the literal value.

6374 StringUtf8 literal text incorrectly formatted

Cause

This error occurs if a **StringUtf8** literal contains an invalid character entity. The entity format could be wrong (for example, a hexadecimal value could include non-hexadecimal characters) or the entity name could be unrecognized.

Action

Correct the literal value.

6375 Schema file localeDefinitions section is empty

Cause

This error occurs if the localeDefinitions section of the schema file does not contain any locale definitions. It is possible the schema file may be corrupted.

Action

Check the locale definitions in the original environment for correctness and extract the **.scm** and **.ddb** or **.ddx** files again.

6376 StringUtf8 literal text contains a non-US-ASCII character

Cause

This error occurs if you attempt to include non-US-ASCII characters in the definition of a **StringUtf8** literal, as shown in the following code fragment example.

```
str8 := @"Copyright ©";           // correct is   str8 := @"Copyright &copy;";
```

Action

Replace the non-US-ASCII characters with appropriate character entities or escaped decimal or hexadecimal Unicode code points.

6377 StringUtf8 literal entity name is missing a trailing semicolon

This error occurs if you omit a trailing semicolon from a character entity in the definition of a **StringUtf8** literal, as shown in the following code fragment example.

```
str8 := @"Copyright &copy";       // correct is   str8 := @"Copyright &copy;";
```

Action

Add the missing semicolon at the end of the character entity.

6378 StringUtf8 literal entity name is unrecognised

This error occurs if you misspell a character entity in the definition of a **StringUtf8** literal, as shown in the following code fragment example

```
str8 := @"Copyright &cpy;";       // correct is   str8 := @"Copyright &copy;";
```

Action

Ensure that the character entity is spelled correctly.

6379 The declared type number does not match the defined type number

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

6380 This is not a OID

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

6381 Schema add denied - global class of a superschema needs reorg

Cause

This error occurs if you attempt to add or load a schema and the global class of a superschema has been versioned.

Action

Reorganize or un-version the superschema before retrying to add or load the schema.

6382 Called method needs recompile to rebuild formal parameter list

Cause

The formal parameter list of the method being called is incomplete and must be rebuilt before the calling method can be compiled successfully.

Action

Recompile the called method and then retry compiling the caller method.

6384 Proxy webservice class already exposed

Cause

This error occurs if an attempt is made to add a [JadeWebService](#) class to an exposure list when it is already present in another list.

Action

Check and fix the exposure list in the JADE development environment and then extract the schema again.

6385 Dictionary keys definition does not match superclass

Cause

This error occurs if you attempt to load a dictionary class that has different dictionary keys defined from those of a superclass.

Action

Ensure that the keys defined for each class in a hierarchy of dictionary classes are the same.

6386 Dictionary keys definition does not match subclass

Cause

This error occurs if you attempt to load a dictionary class that has different dictionary keys defined from those of a subclass.

Action

Ensure that the keys defined for each class in a hierarchy of dictionary classes are the same.

6387 Error occurred copying dictionary keys to subclasses

Cause

When the keys in a dictionary class are changed, the same changes must be made to all the subclasses of the dictionary. This error occurs if the process fails.

Action

Additional details can be found in your JADE messages (**jommsg.log**) log file. If your JADE licenses include support, contact your local JADE support center or JADE Support.

6388 Method cannot be both abstract and notImplemented

Cause

This error occurs if you attempt to include the method options **abstract** and **notImplemented** in the signature of a method.

Action

Remove the **notImplemented** option from the method signature.

6389 Methods marked with `notImplemented` cannot have a body

Cause

This error occurs if an attempt is made to include a method body in a method that has the `notImplemented` method option.

Action

Remove or comment out the method body.

6390 Cannot call a method marked with `notImplemented`

Cause

This error occurs if a method includes a call to a method that has the `notImplemented` method option.

Action

Remove the method call to the method that has the `notImplemented` method option.

6391 First actual parameter for `PseudoMethodCallType` must be object reference

Cause

This error occurs when you compile a method that makes an asynchronous method call using the `invoke` method of the `JadeMethodContext` class and the first parameter is not an object reference.

Action

Ensure that the first actual parameter to the `invoke` method is an object reference.

6392 Second actual parameter for `PseudoMethodCallType` must be valid method name

Cause

This error occurs when you compile a method that makes an asynchronous method call using the `invoke` method of the `JadeMethodContext` class and the second parameter is not the name of a method that can be invoked on the object referenced by the first parameter.

Action

Ensure that the second parameter to the `invoke` method is a method that can be invoked on the object reference that is the first parameter.

6393 PseudoMethodCallType parameter prevents use of ApplicationContext clause

Cause

This error occurs when you compile a method that makes an asynchronous method call using the `invoke` method of the `JadeMethodContext` class that includes an `in applicationContext` clause.

Action

Do not include an `in applicationContext` clause.

6394 PseudoMethodCallType target method formal parameters cannot be output or io

Cause

This error occurs when you compile a method that makes an asynchronous method call using the `invoke` method of the `JadeMethodContext` class and method being called asynchronously has parameters defined with a usage of `output` or `io`.

Action

Ensure that any method you call asynchronously using the `invoke` method does not have parameters with usage `output` or `io`.

6395 Class already exists in a superschema or subschema

Cause

This error occurs if you attempt to add a class when one already exists with the same name in a superschema or subschema.

Action

Rename the existing class or rename the new class.

6396 Cannot delete ExternalDatabase entity - see jommsgs

Cause

This error occurs if you carry out a partial load of an `ExternalDatabase` with the option to delete classes that are not in the schema file and the deletion of one or more `ExternalDatabase` classes is not allowed. For example, the class could be used as a reference or in a non-`ExternalDatabase` collection class.

Action

The classes which cannot be deleted are listed your JADE messages (`jommsg.log`) log file. Delete the uses of these `ExternalDatabase` classes and then retry the partial load of the `ExternalDatabase`.

6397 Non-static translatableString

Cause

This error should not normally occur. It is raised when a method on the [TranslatableString](#) class is called for a non-static translatable string (that is, a translatable string that has parameters or contains another translatable string) and the method receiver must be a static translatable string.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

6398 Same name as format

Cause

This error occurs if you attempt to add a [TranslatableString](#) with the same name as an existing date, time, number, or currency format. In the following example, an attempt is made to add a translatable string with the name **DateFormat** using the [addCompileTranslatableString](#) method of the [Schema](#) class. The attempt fails because there is already a format defined in the schema with the name **DateFormat**.

```
currentSchema.addCompileTranslatableString('DateFormat="Hello World"',  
                                           errCode, errOffset, errLength);
```

Action

Select a name for the translatable string that is not one of the defined formats for the schema.

6399 Cannot update the latest version

Cause

This error occurs if you attempt to add a [TranslatableString](#) to the latest schema version or an attempt to update a translatable string within the latest schema version (for example, by running the [addCompileTranslatableString](#) method of the [Schema](#) class or the [updateCompile](#) method of the [TranslatableString](#) class).

Action

Do not attempt to add or update a [TranslatableString](#) in the latest schema version.

6400 Unexpected token

Cause

This error occurs if the end of the input was expected but additional input tokens were encountered.

Action

Remove the additional tokens from the input. In some cases, an earlier syntactic error (such as a misplaced **end** in a method) may have caused this error. In this case, the earlier syntactic error should be found and corrected.

6401 Invalid error code

Cause

This error occurs if an error occurs that cannot be found in the list of known error codes.

Action

This error should not normally occur. If your JADE licenses include support, contact your local JADE support center or JADE Support.

6402 Expecting:

Cause

This error occurs if a specific input token is expected but was not found. In some cases, the highlighted token may be the source of the error or there may be an error earlier in the input, with no syntactic inconsistency being detected until the highlighted token. In other cases, the token that is reported as being expected by the compiler may not be what you intended.

Consider the following code fragments.

```
// ERROR! "do" omitted below
foreach cust in customerDict
    write cust.name;
endforeach;
```

In the above code fragment, the compiler reports **Expecting: do** on the token **write** that correctly identifies the error.

```
// ERROR! "where" misspelt below
foreach cust in customerDict wher cust.name > "S" do
    write cust.name;
endforeach;
```

In the above code fragment, the compiler reports **Expecting: do** on the misspelled token **wher**. Although the real error is that **where** has been misspelled, the compiler is unable to ascertain that this is the source of the error. However, it knows that **do** is the next compulsory symbol that must appear in the input, and so this is the error that is reported.

Action

Find the position of the error and change your application code accordingly.

6403 Called method's type is incompatible with this method's type

Cause

This error occurs if you attempt to call an instance method from a type method or if you attempt to call a type method from an instance method without specifying the receiver of the method.

Action

Change the method type of the calling method or called method so that the two methods are both instance methods or both type methods.

Alternatively, you can specify the method receiver using the appropriate operator (that is, the dot (.) operator for an instance method or the @ operator for a type method).

6406 Unknown ExecuteFlag name

Cause

This error occurs if the *executeFlag* name specified in the **executeWhen** instruction is not recognized as a **Boolean** global constant.

```
executeWhen Debug;
  write obj.name;
endExecuteWhen;
```

Action

Ensure that you have defined a **Boolean** global constant.

6407 Invalid ExecuteFlag name

Cause

This error should not normally occur. It indicates that the *executeFlag* name specified in the **executeWhen** instruction is broken in some way.

```
executeWhen Debug;
  write obj.name;
endExecuteWhen;
```

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

6408 ExecuteFlag name is not a Boolean global constant

Cause

This error occurs when the *executeFlag* name specified in the **executeWhen** statement is not a **Boolean** global constant. In the following example, **Debug** is a **Boolean** constant in the user-defined **DemoFlagCategory** category of global constants.

```
executeWhen Debug;
  write obj.name;
endExecuteWhen;
```

Action

Ensure that you have defined a **Boolean** global constant.

6409 Schema load modified a current version object during full versioning

Cause

This error occurs if the current version of a script (a method, constant, or translatable string) is marked for recompilation during a schema load in which everything is versioned. Scripts in this state are reported by the JADE Logical Certifier utility.

Action

As the script is marked as being in error, you must load the script or recompile it. If this error persists and you are unable to resolve it by other means, contact your local JADE support center or JADE Support if your JADE licenses include support.

6410 This type is not allowed for a WebService method parameter or return type

Cause

This error occurs if an attempt is made to compile a Web service method (that is, a method in a [JadeWebServiceProvider](#) subclass with the [webService](#) method option) that has a parameter or return type of [JadeDynamicObject](#), [DynaDictionary](#) or a subclass or one of these classes.

Action

You can add a non-**webService** method to a **JadeWebServiceProvider** subclass. The method must be compiled successfully without the **webService** method option before **JadeDynamicObject** and **DynaDictionary** parameters and return types can be specified.

6411 Structural changes are not allowed - class is in use

Cause

This error occurs during a schema load where the [loadStyle](#) parameter has the value **onlyStructuralVersioning**, if you attempt to change a class (for example, by adding a control to a form) and the class has no persistent instances but is in use at the time (for example, the form could be running).

Action

Retry the schema load when the class is not in use.

6412 Cannot update the current version

Cause

This error occurs if you attempt a schema load into the current version of a schema that adds or removes an interface to a class, but the version of the interface in the latest version is different. This restriction on changes that are allowed in the current version of a schema is also enforced in the JADE development environment.

Action

Do not attempt such changes in the current version of a schema, or apply the changes to the latest version.

6413 The type of this property is undefined within this schema

Cause

This error occurs when the compiler is unable to find the type (class or interface) associated with a property. The most-common cause is when the type of an imported property is not included in the same package as the property and is not declared in the **RootSchema**.

Action

If the cause is an imported property, add its type to the package; otherwise, run the metadata logical certifier.

6414 Cannot add a mapping method to an imported class

Cause

This error occurs if you attempt to create a mapping method for an imported property on an imported class.

Action

If you want to implement the mapping method, implement it on the property in the schema that exports the class.

6415 Cannot combine multiple unitTest options on a method

Cause

This error occurs if you include more than one of the following method options in the signature of a method involved in unit testing.

- `unitTestBeforeClass`
- `unitTestAfterClass`
- `unitTestBefore`
- `unitTestAfter`
- `unitTest`
- `unitTestIgnore`

For details, see "[Writing Unit Tests](#)", in Chapter 17 of the JADE *Developer's Reference*.

Action

Correct your coding.

6416 Unit test methods must have no parameters and no return type

Cause

This error occurs if you include parameters or a return type in the signature of a unit test method. A unit test method cannot have parameters or return a value.

Action

Correct your coding.

6417 Type is incomplete imported type as a result of circular package dependencies

Cause

This error occurs if your schema file imports a package in a schema that has not yet been exported (that is, the package has a circular dependency with another package in the schema).

For details about circular packages, see "[Specifying Advanced Load Options](#)", in Chapter 10 of the *JADE Development Environment User's Guide*, see "[Specifying Advanced Load Options](#)", in Chapter 10 of the *JADE Development Environment User's Guide*, and "[Specifying Additional Load Options](#)", in the *Jade Schema Load User's Guide*.

Action

Load the exporting schema to ensure that the exporting package is complete and then reload the importing schema file.

6418 Web Service method cannot return Interface type

Cause

This error occurs when the return type of a Web service method is an Interface type.

Action

Change the return type of the method to specify the type returned as a class and recompile the method.

6419 Web Service method cannot have parameters of type Interface

Cause

This error occurs when a parameter of a Web service method is an Interface type.

Action

Change the parameters of the method to specify the type passed as a class and recompile.

6420 Interface is already implemented by this class in a subschema or superschema

Cause

This error occurs if you attempt to implement an interface by a class when that interface is already implemented by a class of the same name in a superschema or a subschema.

Action

Do not attempt to implement an interface for the same class in more than once, regardless of whether you are attempting this in a subschema, or a superschema.

6421 Cannot reuse existing class map for a partitioned file

Cause

This error occurs if a new database file is defined that shares the same name as a file in a different schema and the partitionable attribute is set to **true** for one of the files.

Action

Change the name of the file, to avoid the conflict.

6422 Can only map a single class to a partitioned file

Cause

This error occurs if more than one class is mapped to a database file that has the **partitionable** attribute set to **true**.

Action

Change the class map definition.

6423 Cannot map a collection class to a partitioned file

Cause

This error occurs if a collection class is mapped to a database file that has the **partitionable** attribute set to **true**.

Action

Change the class map definition.

6424 Cannot unpartition a partitioned file via a schema load

Cause

This error occurs if an existing database file is partitioned and the schema definition does not specify that the value of the `partitionable` property is `true`.

Action

Determine the reason for the inconsistency and take corrective action.

If a database file has been partitioned in a deployed database, the value of the `partitionable` property should be set to `true` in the development database to avoid this inconsistency.

6425 Invalid partition method signature

Cause

This error occurs if you have defined a partition method that does not conform to the required signature for partition methods.

Partition methods must meet the following requirements:

- Type of any parameter must be a JADE primitive type
- Usage of any parameter must be `constant`
- Return type must be `Integer`
- Method options cannot include `updating` or `abstract`
- Method options must include `partitionMethod`

Action

Change the partition method header to conform to the rules for the signature of partition methods. If you did not intend the method to be a partition method, remove the `partitionMethod` option from the method signature.

6426 This is not allowed in a partition method

Cause

This error occurs if you attempt to use a JADE language feature that is not allowed in a partition method; that is, a method with the `partitionMethod` method option specified.

Partition methods:

- Can use `if` and `return` instructions only.
- Can use the `self` system variable only (for example, `app` and `global` are not allowed).
- Cannot declare and use local option variables.
- Cannot use expressions with property de-references (for example, `myCustomer.name` is not allowed) or property expressions such as `array[i]` or `customer["name"]`.

Action

Change the partition method to conform to the partition method restrictions. If you do not intend the method to be a partition method, remove the `partitionMethod` method from the method signature.

6428 Cannot access versioned feature in current version of method

Cause

This error occurs when you attempt to compile a method that exists in the current version but not in the latest version and that method refers to a property or method that is *different* in the latest version; for example, a property with a different type or a method with a different signature.

Action

Extract the method that cannot be compiled from the current version and load it into the latest version. Alternatively, remove references to the property or method that is *different* in the latest version.

6429 Cannot mark the default map as partitionable

Cause

This error occurs when you attempt to load a schema with the default map file marked as partitionable, which is not allowed.

Action

Remove the partitionable attribute from the map file definition in the `databaseFileDefinitions` section of the schema file and then load the schema again.

6431 Cannot use option browserExecution on method in this class [Obsolete]

Cause

This error occurs when you attempt to compile a method with the `browserExecution` option in which the option is not allowed.

You can only use this option for methods in primitive types (including primitive arrays) and the `Application` class.

Action

Remove the `browserExecution` option from the signature.

6432 Cannot have multiple execution location options on a method

Cause

This error occurs when the signature of a method contains more than one execution location; for example, [serverExecution](#) and [clientExecution](#)).

The metaschema can only store one execution location for a method, and it makes no sense to have multiple locations.

Action

If there are multiple locations, decide on the single location where the method should execute. If a method could execute in more than one location, define separate methods with a single execution location.

6435 Cannot remove subclassPersistentAllowed option as subclass instances exist

Cause

This error occurs if you attempt to change the **Class Lifetime** setting or the **Subclass Lifetime** setting that specifies that persistent instances for a class or its subclasses are not allowed, when persistent instances of the class (or its subclasses) already exist.

Action

Remove the existing persistent instances of the class (or its subclasses), which are preventing the change to the **Class Lifetime** setting or the **Subclass Lifetime** setting being made.

6436 ActiveX Exposures are no longer valid

Cause

This error occurs when you attempt to load a schema that contains a JADE ActiveX exposure. JADE ActiveX exposures are not supported in JADE versions after version 6.3.

Action

Remove the exposure in the JADE 6.3 system, and then re-extract the schema files

6439 A runtime dynamic property with the same name already exists

Cause

This error occurs if a class definition containing a non-dynamic property with the same name as a runtime dynamic property is encountered during a schema load.

Action

Change the dynamic property name or the property name in the schema file, to avoid the conflict.

6440 A user class with the same name already exists

Cause

This error occurs if a class definition containing a non-user class with the same name as a user class is encountered during a schema load.

Action

Change the user class name or the class name in the schema file, to avoid the conflict.

6441 The expression is too complex

Cause

This error occurs if you attempt to compile a method that has a large expression that has greater than 4,000 consecutive operand/operator pairs; for example, a method containing a **Binary** constant that is the result of concatenating more than 5,000 literal values.

Action

Avoid the compile error by changing the expression to concatenate a series of subexpressions enclosed in parentheses, where each subexpression has significantly fewer than 4,000 operands.

The following string expression can be changed to use subexpressions, as is shown in the second example.

```
"a" & "b" & "c" & "d" & "e" & "f" & "g" & "h" & "i" & "j" & "k" & "l"
```

```
("a" & "b" & "c") & ("d" & "e" & "f") & ("g" & "h" & "i") & ("j" & "k" & "l")
```

6442 An abstract class method cannot have a unitTest option

Cause

This error occurs if you attempt to compile a method in an abstract **JadeTestCase** subclass and the method includes a *unitTest* option; that is, **unitTestBeforeClass**, **unitTestBefore**, **unitTest**, **unitTestIgnore**, **unitTestAfter**, or **unitTestAfterClass**.

Action

Remove the *unitTest* option from the method signature.

6443 A static property cannot be converted to a dynamic property and vice versa

Cause

This error occurs when a schema load attempts to change a static property to be dynamic, or a dynamic property to be static.

Action

Correct the schema files to avoid changing static properties to be dynamic, or dynamic properties to be static.

6444 A dynamic property cannot have its cluster name changed

Cause

This error occurs when a schema load attempts to change the cluster to which a dynamic property is assigned.

Action

Correct the schema files to avoid assigning dynamic properties to different clusters.

6445 Unsupported dynamic property type change

Cause

This error occurs when a schema load attempts to change the type of a dynamic reference to a class that is not a superclass type.

Action

Correct the schema files to avoid changing the type of a dynamic references to a class other than a superclass.

6446 A Dynamic property cannot be virtual

Cause

This error occurs when a schema load attempts to change a dynamic property to be virtual.

Action

Correct the schema files to avoid changing dynamic properties to be virtual.

6447 A Dynamic property cannot be changed from non-embedded to embedded

Cause

This error occurs when a schema load attempts to change a dynamic property that is a subobject to one that is embedded in the object; for example, changing a **Binary** with more than 540 bytes to a **Binary** with fewer than 540 bytes.

Action

Correct the schema files to avoid changing dynamic properties subobjects to being embedded in the object.

6448 Changes to Dynamic properties can only be made in the latest version

Cause

This error occurs when a schema load attempts to change dynamic properties in the current schema version.

Action

Load into the latest version of the schema.

6449 Java exposures are no longer valid

Cause

This error occurs when you attempt to load a schema that contains a JADE Java exposure. JADE Java exposures are not supported in JADE versions after version 7.0.

Action

Remove the exposure in the JADE 7.0 system, and then re-extract the schema files.

6450 Constructor not exported from Package

Cause

This error occurs when attempting to create an instance of an exported class that has a constructor with parameters in an importing schema and the **create** method has not been exported in a package.

Action

If you export a class with a constructor with parameters from a package, export the **create** method as well.

6451 Constructor has parameters

Cause

This error occurs when a constructor definition has parameters that are not specified in the **create** instruction.

Action

Specify parameters in the **create** instruction that are compatible with those of the constructor.

6452 Constructor does not call superclass constructor

Cause

This error occurs when the immediate superclass implementation has parameters and **::super** is not specified in the constructor.

Action

Specify **::super** and the parameters defined in the superclass constructor in your **create** method.

6453 Superclass constructor has parameters

Cause

This error occurs when a superclass **create** method has parameters and the **create** method of an immediate subclass does not provide the parameters required by the superclass **create** method.

Action

A **create** method must be implemented on the subclass of the class that has a constructor with parameters. This method must specify **::super** and provide the parameters required by the superclass **create** method.

6454 Only valid in constructor

Cause

This error occurs when **::super** is specified a method that is not a **create** method.

Action

Remove the **::super(required-parameters)** definition.

6455 Superclass constructor does not have parameters

Cause

This error occurs if the superclass **create** method does not have parameters and **::super(required-parameters)** is defined in the **create** method of an immediate subclass.

Action

Define parameters in the **create** method of the superclass, or remove the **::super(required-parameters)** definition in the **create** method of the immediate subclass.

6456 Constructor formal parameters cannot be output or io

Cause

This error occurs when you compile a constructor that has parameters defined with a usage of **output** or **io**.

Action

Ensure that the constructor does not have parameters with usage **output** or **io**.

6457 Constructor with parameters not allowed on this class

Cause

This error occurs when you declare a parameter on the **create** method of a class that derives from a RootSchema class other than **Object**.

Action

Delete the parameter declaration.

6458 Constructor does not have parameters

Cause

This error occurs when a constructor definition does not have parameters that are specified in the **create** instruction.

Action

Specify parameters in the **create** instruction that are compatible with those of the constructor.

6459 Invalid receiverByReference signature

Cause

This error occurs when a **receiverByReference** method is defined that does not conform to the required **receiverByReference** method option syntax.

The **receiverByReference** method option cannot be specified in a method signature:

- With a primitive type method
- With any one of the **condition**, **conditionSafe**, **lockReceiver**, **partitionMethod**, **mapping**, **typeMethod**, and **webservice** method options
- On unit test methods
- On a notification or event method
- With interface methods

Action

Change the **receiverByReference** method header to conform to the rules for the signature of **receiverByReference** methods. If you did not intend the method to have **receiverByReference** behavior, remove the option from the method signature.

Applies to Version: 2020.0.01 and higher

6460 Type is not a sub-type of collection membership

Cause

This error occurs if an explicit type specified in a **foreach** instruction **as** clause expression is not a sub-type of the membership type of the collection.

Action

Change your application code so that the **as** clause explicit type is a subtype of the membership type of the collection.

Applies to Version: 2020.0.01 and higher

6500 Cancelled by user

Cause

This is a non-error status code that occurs when you cancel a schema load; for example, by clicking a **Cancel** button.

Action

None.

7000 through 7999 Method and Schema File Syntax Errors

The error messages in this section indicate syntax errors in your method or schema file; for example, you could have omitted a semicolon, used a reserved word inappropriately, or ordered the clauses of a JADE instruction incorrectly.

The error numbers and the corresponding syntax error can change from one release to the next. The code is based on the internal number used for each symbol.

However, the error text is available by calling the [getErrorText](#) method of the [Process](#) class, as in the following code fragment.

```
write process.getErrorText(7023);  
// outputs "Expecting: TimeStamp" in JADE 2016
```

See Chapter 1, "[JADE Language Reference](#)", of the *JADE Developer's Reference* for all syntax descriptions.

8000 through 8255 External Database Errors

This section describes errors that can be returned if you encounter problems when using external databases.

8000 External database ODBC error

Cause

This error reports any general unexpected ODBC error.

Action

For more details, see "[ODBCException Class](#)", in Chapter 1 of the *JADE Encyclopaedia of Classes*.

8001 External database open failed

Cause

This error occurs if an attempt to open an external database is unsuccessful.

Action

Check the ODBC configuration for the data source. For certain errors, additional information may be recorded in the JADE messages log file ([jommsg.log](#)).

8002 External database not open

Cause

This error occurs if your application code attempts to access an external database that has not been opened.

Action

Ensure your application code opens the external database before it attempts to access it.

8003 External database create failed

Cause

This error occurs if an attempt to create an external database is unsuccessful.

Action

Check the ODBC configuration for the data source. For certain errors, additional information may be recorded in the JADE messages log file ([jommsg.log](#)).

8004 External database already in transaction state

Cause

This error occurs if your application code attempts to enter transaction state for an external database that is already in transaction state.

Action

Change your application code to ensure that this does not occur.

8005 External database not in transaction state

Cause

This error occurs if your application code attempts to end transaction state for an external database that is not in transaction state.

Action

Change your application code to ensure that this does not occur.

8006 External database illegal use of forward only cursor

Cause

This error occurs if an illegal operation is attempted; for example, backward scrolling when your ODBC driver or data source supports only forward-only cursors.

Action

Avoid the operation that resulted in the error, or obtain an ODBC driver that supports the operation.

8007 External database keyset driven cursor not supported by ODBC driver

Cause

This error occurs if you explicitly request keyset-driven cursor behavior but the ODBC driver does not support this.

Action

Avoid setting the keyset-driven option, or obtain an ODBC driver that supports this behavior.

8008 External database static cursor not supported by ODBC driver

Cause

This error occurs if you explicitly request static cursor behavior but the ODBC driver does not support this.

Action

Avoid setting the static option, or obtain an ODBC driver that supports this behavior.

8009 External database ODBC driver has inadequate API conformance level

Cause

This error occurs if the ODBC driver that you are using does not conform to the required ODBC API conformance level 1.

Action

Obtain and use an ODBC driver that satisfies the minimum conformance level.

8010 External database ODBC driver inadequate SQL conformance level

Cause

This error occurs if the ODBC driver that you are using does not conform to the minimum conformance level defined in the ODBC standard.

Action

Obtain and use an ODBC driver that satisfies the minimum conformance requirements.

8011 External database ODBC driver version 2 or higher required

Cause

This error occurs if your ODBC driver is not the correct version.

Action

Obtain and use an ODBC driver that is version 2 or higher.

8012 External database dynamic cursor not supported by ODBC driver

Cause

This error occurs if you explicitly request dynamic cursor behavior but the ODBC driver does not support this.

Action

Avoid setting the dynamic option, or obtain an ODBC driver that supports this behavior.

8013 External database persistent external collection not supported

Cause

This error occurs if your application code attempts an operation that uses a persistent external collection. This is not supported for external databases.

Action

Redefine any external collection references so that at runtime they are properties of a transient class such as [app](#), a form, or the external database instance.

8014 External database operation failed

Cause

This error occurs if an operation on an external object, collection, or database could not be processed.

Action

Refer to the JADE messages log file ([jommsg.log](#)) for more details about this error.

8022 External database multiple records updated

Cause

This error occurs most-commonly if the external table does not have a set of columns that uniquely identifies a row in the result set, so that an update or delete on an external object affects multiple rows in the external database.

Action

Ensure that tables mapped into a class for update or delete have one or more columns that uniquely identify a row in the table.

8023 External database no record updated

Cause

This error occurs if an update or delete operation on an external object failed to update the external database. This usually means that a concurrent user deleted the record.

Action

Change your application code to avoid or handle this error.

8024 Cannot access an External database opened by a different Node

Cause

This error occurs when an attempt is made to access an external database on a different node to the one that opened the external database; for example, when an external database is opened on a client node and a server method attempts to read or write to the external database.

Action

Change your code to ensure that when external database objects are employed on different nodes, the external database is opened on the same node that performs the external database operations.

8025 Invalid mapping method for attribute: not blob or slob

Cause

This error occurs if there is a mapping method for an attribute corresponding to a column in the external relational database and the size of the string (or binary) data in the column is not unbounded.

A column containing string or binary values with a length that is unbounded or above a certain value (that is, a slob or blob) requires a mapping method for the corresponding attribute. The mapping method, which is created by the External Database wizard, calls the `_getVirtualString`, `_setVirtualString`, `_getVirtualBinary`, and `_setVirtualBinary` methods.

A column containing string or binary values with a fixed length should not have a mapping method for the corresponding attribute.

Action

Remove the invalid mapping method for the attribute.

8256 through 8499 JADE ODBC Errors

This section describes errors that can occur when calling the ODBC driver to access a JADE database. These errors may or may not be displayed by the calling program.

It is recommended that if you begin to encounter these errors, you also turn on SQL debugging, accessed from the ODBC Data Source Administrator dialog. For more details, see your *ODBC Programmer's Reference* from Microsoft.

8256 SQLConnect requires a User ID

Cause

This error occurs if **SQLConnect** is called with no user name.

Action

Connect again with a valid user name specified.

8257 Unexpected token: '%s' at %d

Cause

This error occurs if an unexpected token was found in the SQL statement, for example:

```
SELECT &num FROM TABLE
```

Action

Correct the SQL statement and call **SQLPrepare** again.

8258 Expecting: '%s'; found '%s' at %d

Cause

This error occurs if the specified token was found at the specified position in the SQL statement but another was expected.

Action

Correct the SQL statement and call **SQLPrepare** again.

If the error message is *Expecting statement; found select at 1*, the JADE ODBC driver is set to expect the SQL keywords in uppercase only. Check the setting of keywords in the Data Source Name (DSN).

8259 Internal JADE ODBC error

Cause

This is an internal error.

Action

Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

8262 Memory allocation failed

Cause

This error occurs if the memory allocation request fails.

Action

Perform one of the following actions.

- Shut down other applications that are not required.
- Increase virtual memory.
- Add more memory.

8263 Class '%s' not found in the database

Cause

This error occurs if the table specified was found in the relational view but the class does not exist in the JADE database.

Action

Check that the JADE database, the relational view, and the SQL table definitions are all current and up to date.

8264 Column number specified (%d) is equal to 0 or is greater than maximum (%d)

Cause

This error occurs if the column number specified does not exist in the current statement.

Action

Correct the SQL statement and call **SQLPrepare** again.

8265 Data item truncated; buffer was too small

Cause

This error occurs if the buffer specified by the calling program is too small for the data being returned. The data is truncated to fit the user buffer and a non-fatal warning is returned.

Action

This is a non-fatal error. It is issued as a warning for your information. It may be necessary to increase the buffer specified by the calling program.

8266 Unknown column attribute: %d

Cause

This error occurs if an unknown column attribute is passed to **SQLColAttribute**.

Action

Correct the attribute and call **SQLColAttribute** again.

8267 Relational Table has no columns

Cause

This is an internal error.

Action

Refer to the JADE messages log file (**jommmsg.log**) for more details about this error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

8268 SQL statement not supported in read-only database: %s

Cause

This error occurs if the SQL statement modifies the database. This is not supported, since the JADE database is read-only through ODBC.

The following SQL statements are not supported.

- CREATE
- DELETE
- DROP
- INSERT
- UPDATE

Action

Correct the SQL statement accordingly.

8269 Table '%s' was not found

Cause

This error occurs if the specified table in the table list was not found.

Action

Correct the SQL statement and call **SQLPrepare** again.

8270 Table correlation name '%s' is not unique

Cause

This error occurs if the specified table correlation name is already in use as a table or correlation name; for example:

```
SELECT oid FROM Ferry QX, Port QX
```

Action

Correct the SQL statement and call **SQLPrepare** again.

8271 Maximum name length (%d) exceeded: '%s'

Cause

This error occurs the name specified exceeds the maximum table or column name length of 80 characters.

Action

Correct the SQL statement and call **SQLPrepare** again.

8272 Property corresponding to column '%s' not found

Cause

This error occurs if the specified column in the select list or **WHERE** clause was not found in any table specified in the table list; for example:

```
SELECT num FROM Ferry
```

Action

Correct the SQL statement and call **SQLPrepare** again.

8273 Column name '%s' ambiguous

Cause

This error occurs if the specified column in the select list or **WHERE** clause was found in more than one of the tables specified in the table list.

Action

Correct the SQL statement by specifying the table from which the column is to be selected and call **SQLPrepare** again.

8274 Table name '%s' ambiguous

Cause

This is an internal error. The specified table name cannot be found.

Action

Refer to the JADE messages log file (**jommsg.log**) for more details about this error.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

8275 Invalid combination of types compared: %s and %s

Cause

This error occurs if the types being compared in the **where** clause are not valid; for example:

```
<time> = <date>
```

Action

Correct the SQL statement and call **SQLPrepare** again.

8276 Invalid comparison operator used for type %s

Cause

This error occurs if the comparison operator specified is not valid for the type; for example:

```
<date> LIKE <date>
```

Action

Correct the SQL statement and call **SQLPrepare** again.

8277 Illegal date value

Cause

This error occurs if:

- A character column of the result set is bound to a **Date** primitive type, but the value of the column is not a valid date.
- The parameter supplied for a **Date** primitive type is not valid.

Action

Correct the illegal date value accordingly.

8278 Time must be in the format hh:mm:ss

Cause

This error occurs if the parameter supplied for a **Time** primitive type is not in the format *hh:mm:ss*.

Action

Correct the parameter accordingly.

8279 Illegal time value

Cause

This error occurs if the parameter supplied for a **Time** primitive type is not a valid time.

Action

Correct the parameter accordingly.

8280 Illegal timestamp value

Cause

This error occurs if the parameter supplied for a **TimeStamp** primitive type is not a valid timestamp.

Action

Correct the parameter accordingly.

8281 Illegal type conversion from type %s to type %s

Cause

This error occurs if a statement specifies an illegal type conversion for the specified type.

Action

Correct the SQL statement and call **SQLPrepare** again.

8282 Buffer for character conversion of type %s too small (%d required)

Cause

This error occurs if you specify a buffer for character output that is too small to contain the **Date**, **Time**, or **TimeStamp** value.

Action

Correct the buffer size accordingly.

8283 Buffer for binary conversion of type %s too small (%d required)

Cause

This error occurs if you specify a buffer for the binary output that is too small to contain the date, time, or timestamp value.

Action

Correct the buffer size accordingly.

8284 RelationalTableRelationship (%s) missing vital information

Cause

This is an internal error. This error occurs if a specified relational table relationship is missing vital information.

Action

Refer to your JADE messages (**jommsg.log**) log file for more details about this error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

8285 Numeric argument in an ORDER BY clause must be an integer value

Cause

This error occurs when you specify an argument in the **ORDER BY** clause that is not an integer value; for example:

```
SELECT c03BINA,c06DECI FROM Q010101 ORDER BY 22.5
```

Action

Change the **ORDER BY** clause to use an integer value.

8286 Numeric argument in an ORDER BY clause is out of range

Cause

This error occurs if you specify a numeric argument in the **ORDER BY** clause that is greater than the number of columns specified; for example:

```
SELECT c03BINA,c06DECI FROM Q010101 ORDER BY 5
```

Action

Change the **ORDER BY** clause accordingly.

8287 Column '%s' in ORDER BY clause not found in the select list

Cause

This error occurs if you specify a column in the **ORDER BY** clause that cannot be found in any of the tables specified in the table clause; for example:

```
SELECT c03BINA,c09DATE FROM Q010101 ORDER BY c10TIME
```

Action

Change the **ORDER BY** clause accordingly.

8288 Operand of unary operation '%s' not numeric

Cause

This error occurs when the operand of a unary operation (+,-) is not numeric; for example:

```
SELECT c10TIME FROM Q010101 WHERE -c10TIME < 8
```

Unary operations must have numeric operands.

Action

Correct the specified operand accordingly.

8289 Operand of binary operation '%s' not numeric

Cause

This error occurs when the operand of a binary operation (+,-) is not numeric; for example:

```
SELECT c10TIME,c06DECI FROM Q010101 WHERE c06DECI - c10TIME < 5
```

Binary operations must have numeric operands.

Action

Correct the specified operand accordingly.

8290 Both operands of binary operation '%s' dynamic, cannot determine expression type

Cause

This error occurs when both operands of a binary operation (+,-) are dynamic parameters; for example:

```
SELECT c10TIME,c06DECI FROM Q010101 WHERE c06DECI - c10TIME < 5
```

Both operands of a binary operation cannot be dynamic parameters.

Action

Correct the specified operands accordingly.

8291 Ill-formed expression found; cannot contain a boolean operation

Cause

This error occurs if an expression is found to contain a **Boolean** operation.

Action

Correct the expression accordingly.

8292 Ill-formed search condition found

Cause

This error occurs if an expression is found to contain an invalid search condition.

Action

Correct the SQL statement and call **SQLPrepare** again.

8293 Both operands of a comparison operation cannot be dynamic

Cause

This error occurs if both operands of a comparison are dynamic parameters; for example:

```
SELECT c10TIME,c06DECI FROM Q010101 WHERE ? <= ?
```

Both operands of a comparison cannot be dynamic parameters.

Action

Correct the specified operands accordingly.

8294 Operand of unary operation '%s' cannot be dynamic

Cause

This error occurs if the operand of a unary (+,-) operation is a dynamic parameter; for example:

```
SELECT c10TIME,c06DECI FROM Q010101 WHERE c10TIME < -?
```

Unary operations cannot have a dynamic parameter as an operand.

Action

Correct the specified operand accordingly.

8295 Item in the select list cannot be a dynamic parameter

Cause

This error occurs if an item in the select list is a dynamic parameter; for example:

```
SELECT ?,c10TIME,c06DECI FROM Q010101
```

Select lists cannot contain dynamic parameters.

Action

Correct the select list accordingly.

8296 Parameter index specified (%d) greater than number of parameters (%d)

Cause

This error occurs if the parameter supplied does not exist in the current SQL statement.

Action

Correct the SQL statement and call **SQLPrepare** again.

8297 Parameter %d does not have a value bound to it

Cause

This error occurs if a SQL statement contains a parameter number but a value has not been bound to that parameter.

A value is required for all parameters; for example:

```
SELECT * FROM Table WHERE count = ?
```

A value must be supplied at run time for the ? parameter.

Action

Ensure that a value is supplied at run time for the specified parameter.

8298 Invalid parameter type for parameter %d; Only parameters of type SQL_PARAM_INPUT accepted

Cause

This error occurs if the parameter type of a specified parameter is invalid. Input parameters only are valid. Procedure calls are not supported.

Action

Ensure that valid parameter types (input parameters) only are used.

8299 Cannot convert from C type %s to SQL type %s for parameter %d

Cause

This error occurs if the value supplied for a specified parameter cannot be converted from the C type specified by the statement to the SQL type required.

Action

Correct the SQL statement and call **SQLPrepare** again.

8300 Multiple calls have been made to SQLPutData for parameter %d

Cause

This error occurs if multiple calls have been made to **SQLPutData** for a specified parameter. Multiple calls are supported only for character or binary data.

Action

Avoid making multiple calls to **SQLPutData** for the specified parameter (for other than character or binary data).

8301 SQL_NULL_DATA or SQL_DEFAULT_PARAM not supported by SQLPutData for parameter %d

Cause

This error occurs if **SQLPutData** is called with a null data or default parameter. **SQLPutData** does not support null data and default parameters.

Action

Correct the **SQLPutData** call accordingly.

8302 Invalid string or buffer length passed to SQLPutData for parameter %d

Cause

This error occurs if an invalid length is passed to **SQLPutData** for a specified parameter.

Action

Correct the **SQLPutData** call accordingly.

8303 Length of parameter %d not supplied

Cause

This error occurs if a length is required for a specified parameter but a length is not supplied.

Action

Correct the **SQLPutData** call accordingly.

8304 Binary data invalid size; size supplied %d; required %d

Cause

This error occurs if the binary data supplied for a specified parameter is not the required size.

Action

Ensure that the binary data supplied for the specified data is of the required size.

8305 Buffer for type conversion too short (%d supplied; %d required)

Cause

This error occurs if the character data supplied for a specified parameter is not the required size.

Action

Ensure that the character data supplied for the specified data is of the required size.

8306 Overflow occurred converting numeric data

Cause

This error occurs if an overflow occurs while converting numeric data.

Action

Ensure that the buffer for the specified data is sufficient for the data being returned.

8307 String being converted not a valid number format

Cause

This error occurs if the conversion of a string value to a numeric or date-time format cannot be completed because the string is not a valid number or date-time.

This may occur when returning column data or evaluating parameters.

Action

Ensure that the string that is to be converted represents a valid number or date-time value.

8308 ParameterValuePtr passed to SQLParamData cannot be NULL

Cause

This error occurs if **SQLParamData** is called with a **null** value **ParameterValuePtr**.

Action

Ensure that **SQLParamData** is called with a valid, non-null, **ParameterValuePtr** value.

8309 Operation cancelled before it completed

Cause

This error occurs if **SQLCancel** is called before the operation completed.

Action

Ensure that **SQLCancel** is not called before the operation completes.

8310 Date-time type %s invalid

Cause

This error occurs if the date/time escape clause in the SQL statement is not valid; for example:

```
SELECT c10TIME FROM Q010101 WHERE c10TIME = {tt invalid}
```

Action

Correct the SQL statement and call **SQLPrepare** again.

8311 Invalid cursor state

Cause

This error occurs if **SQLCursor** has been called when no cursor is open on the statement.

Action

Ensure that a cursor is open on the statement before **SQLCursor** is called.

8312 Property '%s' of class '%s' not found

Cause

This error occurs if the specified property of the specified class is not found in the JADE database. This is an internal error.

Action

Refer to the JADE messages log file (**jommsg.log**) for more details about this error.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

8313 String to be converted to binary contains non-hexadecimal characters

Cause

This error occurs if the conversion of a hex string value to binary format could not be completed because the string contains non-hexadecimal characters; for example:

```
SELECT c03BINA FROM Q010101 WHERE c03BINA = #0f0g
```

This can occur when evaluating parameters or literals in the **where** clause.

Action

Ensure that the string to be converted contains valid hexadecimal characters only.

8314 Requested information type not supported

Cause

This error occurs if the requested information type is not supported.

Action

None.

8315 Schema does not contain the requested Relational View %s

Cause

This error occurs if the relational view specified in the Data Source Name (DSN) does not exist in the specified JADE schema.

Action

Check the DSN definition of the schema and relational view.

8317 Already connected to another schema; Current: %s Requested: %s

Cause

This error occurs if an attempt is made to connect to a schema while already connected to another schema. The JADE ODBC driver can be connected to one schema at a time only.

Action

The current schema must be disconnected before the new one can be connected.

8318 Catalog and schemas not supported

Cause

This error occurs when a Catalog function requests a specific Catalog or Schema. The JADE database does not support the ODBC concept of Catalog and Schemas.

Action

Remove the Catalog or Schema name.

8319 Invalid descriptor field

Cause

This error occurs if an unknown descriptor field is passed to a **SQLGetDescField** or **SQLSetDescField** call.

Action

Correct the **SQLGetDescField** or **SQLSetDescField** call.

8320 Invalid attempt to modify an implementation row descriptor

Cause

This error occurs if there is an invalid attempt to modify an implementation row descriptor. You cannot modify an implementation row descriptor.

Action

Ensure that you do not attempt to modify an implementation row descriptor.

8321 Inconsistent descriptor information set

Cause

This error occurs if the information set for the descriptor is inconsistent.

Action

Ensure that the information set for the descriptor is consistent.

8322 Optional feature not implemented

Cause

This error occurs if an attempt is made to use an optional feature that is not implemented. The following **SQLSetStmtAttribute** feature is not supported.

- ON value for **SQL_ATTR_ASYNC_ENABLE** (**SQL_ATTR_ASYNC_ENABLE_OFF** only is supported)

The following **SQLSetConnectAttribute** features are not supported.

- SQL_ATTR_ASYNC_ENABLE
- SQL_AUTOCOMMIT
- SQL_ATTR_CONNECTION_TIMEOUT
- SQL_ATTR_CURRENT_CATALOG
- SQL_ATTR_LOGIN_TIMEOUT
- SQL_ATTR_PACKET_SIZE
- SQL_ATTR_TRANSLATE_LIB
- SQL_ATTR_TRANSLATE_OPTION
- SQL_TXN_ISOLATION

Action

None.

8323 Option value changed

Cause

This error occurs when the option value passed by the call is not supported by the JADE ODBC driver. The value is changed to a supported value and a non-fatal warning is returned.

Action

No action is required for this error. It is issued as a warning for your information only.

8324 Indicator variable required but not supplied

Cause

This error occurs if a null value was found but this information cannot be passed back because an **Indicator** variable has not been supplied.

Action

Ensure that an **Indicator** variable is supplied.

8325 Invalid argument for FetchOrientation

Cause

This error occurs if an invalid argument is supplied for **FetchOrientation**. **SQL_FETCH_NEXT** only is supported for **FetchOrientation**.

Action

Change the argument supplied to **FetchOrientation** accordingly.

8326 Invalid cursor position

Cause

This error occurs if an invalid cursor position is specified for **SQLGetData**. Only a forward-only cursor with row set of one is implemented.

Action

Specify a valid cursor position on **SQLGetData** accordingly.

8327 Directory location of the Jade ODBC library in the DSN is not the Jade Install Directory

Cause

This error occurs if the location of the JADE ODBC library is not the same as that of the JADE install directory. The location of the JADE ODBC library must be the same as the JADE install directory.

Action

Change the value of the location of the JADE ODBC library in the Data Source Name (DSN) so that it is the same as the JADE install directory.

8328 Database contains invalid date value - current date returned

Cause

This error occurs if the JADE **Date** attribute or the date portion of a JADE **TimeStamp** being returned contains an invalid date value. The current date value is returned and a non-fatal warning is returned.

Action

No action is required for this error. It is issued as a warning for your information only.

8329 Parameter value evaluation resulted in the truncation

Cause

This error occurs if a parameter that is passed to the JADE ODBC driver is truncated to the length allocated for that parameter type.

Action

Check the parameter values.

8330 Columns in a UNION must have the same type

Cause

This error is raised by the JADE ODBC driver if the types of columns joined in the **UNION** of an SQL **SELECT** statement do not match.

Action

Check the types of the columns being joined, modify the SQL **SELECT** statement to join only columns of the same type with a **UNION**, and then resubmit the statement.

8331 All queries in UNION must have the same number of columns

Cause

This error is raised by the JADE ODBC driver if the number of columns joined in the **UNION** of an SQL **SELECT** statement does not match.

Action

Check the columns being joined, modify the SQL **SELECT** statement to join the same number of columns in the **UNION**, and then resubmit the statement.

8332 Divide by zero

Cause

This error is raised by the JADE ODBC driver if a computation in the SQL **SELECT** statement results in a division by zero (0).

Action

Check the values and computations in the SQL **SELECT** statement, correct the values, and then resubmit the statement.

8333 Column '%s' in ORDER BY clause is not unique in the select list

Cause

This error is raised by the JADE ODBC driver if the column that is specified in the **ORDER BY** clause of an SQL **SELECT** statement is not unique.

Action

Check the names of the columns specified in the **ORDER BY** clause and ensure that the name is unique in all tables used in the select statement.

If necessary, modify the SQL **SELECT** statement to alias the column or to qualify the column name with the table name in the **ORDER BY** clause and then resubmit the statement.

8334 Columns of a table derived from a Jade method can only be ANDed and compared for equality (=)

Cause

This error is raised by the JADE ODBC driver if the column of a table derived from a JADE method are used in the **SELECT** statement **WHERE** clause in any way other than an **AND** or an equality comparison (for example, using **OR**, **>=**, and so on).

Action

Check the **SELECT** statement, modify the **WHERE** clause to use only **AND** or **=** operations for columns of tables derived from JADE methods, and then resubmit the statement.

8335 Schema '%s' does not exist

Cause

This error occurs if the schema specified in the DSN does not exist.

Action

Check the schema specified in the DSN for correctness.

8336 RPC does not support usage IO or OUTPUT parameters of this type

Cause

This error occurs if the ODBC Client driver makes a Remote Procedure Call (RPC) using IO or OUTPUT parameters that are not supported in an RPC.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

8337 Column %d invalid because it is not an aggregate and not in GROUP BY

Cause

This error is raised by the JADE ODBC driver if an SQL **SELECT** statement contains a **GROUP BY** clause and one of the **SELECT** columns has been omitted from the **GROUP BY** clause.

Action

Check that the columns in the **GROUP BY** clause match the **SELECT** columns exactly.

8338 Column '%s' in GROUP BY clause not found in the select list

Cause

This error is raised by the JADE ODBC driver if a column in the **GROUP BY** clause is not one of the **SELECT** columns in the SQL statement.

Action

Check that the columns in the **GROUP BY** clause match the **SELECT** columns exactly.

8339 Column '%s' in GROUP BY is not unique

Cause

This error is raised by the JADE ODBC driver if a column is duplicated in the **GROUP BY** clause of an SQL statement.

Action

Check that the columns in the **GROUP BY** clause match the **SELECT** columns exactly.

8340 Client communication protocol is not compatible with the server

Cause

This error occurs if a thin client attempts to connect to a node that is running JADE software with which the thin client is not compatible.

Action

Ensure that the client is attempting to connect to the correct ODBC service.

Upgrade the thin client software, if required. To initiate a download of the latest thin client software, start a JADE smart thin client application.

8342 Communication link failure

Cause

This error occurs if a network error causes a JADE ODBC thin client to disconnect from the server while processing a query.

Action

Close and then reopen the database connection from your ODBC application. If the problem persists, contact your network administrator.

8345 Table name already used

Cause

This error occurs if a call to the [addUserTable](#) method of the [RelationalView](#) class specifies a table name which is already in use in the [RelationalView](#) class.

Action

Specify a unique table name, and then re-execute the code.

8346 Relational Table not found

Cause

This error occurs if a call to the [addUserAttribute](#), [removeColumn](#), or [removeTable](#) method of the [RelationalView](#) class specifies a table name which does not exist in the [RelationalView](#) class.

Action

Specify an existing table name, and then re-execute the code.

8347 Attribute name already used as column in the selected table

Cause

This error occurs if a call to the [addUserAttribute](#) method of the [RelationalView](#) class specifies a column name which is already used in the specified table.

Action

Specify a unique column name, and then re-execute the code.

8348 No SQL Mapping for JADE Type defined by JadeRelationalAttribute::getJadeType

Cause

This error occurs if a call to the [getJadeType](#) method for the [JadeRelationalAttributeIF](#) interface that is specified in a call to the [addUserAttribute](#) method of the [RelationalView](#) class returns a JADE type which cannot be mapped in a relational view.

Action

Do not map the attribute. Alternatively, map to another JADE type which *can* be mapped in a relational view (for example, [String](#)).

8349 Column not found in Relational Table

Cause

This error occurs if a call to the [removeColumn](#) method of the [RelationalView](#) class specifies a column name which does not exist in the specified table.

Action

Specify an existing column name, and then re-execute the code.

8350 ODBC Server running in different schema: Server %s ThinClient: %s

Cause

This error occurs if the ODBC Server that is specified in the JADE ODBC thin client Data Source Name (DSN) is running in a different schema from the schema specified in the DSN.

Action

Specify the correct schema, or modify the connection options, in the DSN.

8351 Soft attributes may only be added to Soft Tables

Cause

This error occurs if a call to the [addUserAttribute](#) method of the [RelationalView](#) class specifies a table name which is not a user-defined *soft table*.

Action

Only attempt to add user-defined soft attributes to user-defined-soft tables.

Add *soft tables* using the [addUserTable](#) method of the [RelationalView](#) class.

8352 No JADE Type defined for this Attribute

Cause

This error occurs if a call to the [getJadeType](#) interface method for the [JadeRelationalAttributeIF](#) interface that is specified in a call to the [addUserAttribute](#) method of the [RelationalView](#) class returns a **null** type.

Action

Modify the interface method to return a valid JADE type that can be mapped in a relational view.

8353 Must be ODBC Relational View, not RPS Mapping

Cause

This error occurs if a call to the [addUserAttribute](#) or the [addUserTable](#) or method of the [RelationalView](#) class is made on a [RelationalView](#) object which is an RPS mapping.

Action

Use a **RelationalView** object which is not an RPS mapping.

8354 Invalid Username or Password specification

Cause

This error occurs if the user name or password that is passed to the JADE ODBC driver is longer than the maximum 30 characters allowed for a valid JADE user name and password.

Action

Specify a valid user name and password, and then reconnect.

8355 Client library version differs from server version

Cause

This error occurs if a thin client attempts to connect to a node that is running JADE software with which the thin client is not compatible because the client and server have different versions of one or more of the following libraries:

- **jadodbc_c**
- **jomos**
- **jomjdi**
- **jadcnet**
- **jomutil**

Action

Ensure that the client is attempting to connect to the correct ODBC service.

Upgrade the thin client software, if required. To initiate a download of the latest thin client software, start a JADE smart thin client application.

8356 Column name cannot be changed

Cause

This error occurs if a call to the **changeColumnName** method of the **RelationalView** class specifies a column name that is not a user-defined property or method. For example, the **oid** column name cannot be changed.

Action

Do not attempt to change the name of a column that is not a user-defined property or method.

8357 Null value eliminated in aggregate function

Cause

This error occurs if an aggregate function in a **SELECT** statement includes **NULL** values that were ignored when computing the result of the aggregate function.

Action

This is an informational message only, so no user action is required.

8360 Query timeout expired

Cause

This error occurs when the time taken to execute an ODBC query exceeds the time specified in the **QueryTimeout** parameter in the [\[JadeOdbc\]](#) section of the JADE initialization file. The query timeout can also be set by the ODBC tool submitting the query, which overrides the JADE initialization file setting for that statement only.

Action

Increase the timeout value by making a change to the value of the **QueryTimeout** parameter in the JADE initialization file or to a setting in the ODBC query tool.

8500 through 8599 Schema Load-Specific Errors

This section describes errors that can be returned if you encounter problems when using the JADE schema load functionality.

8500 Superschema not specified or found

Cause

This error occurs when using batch JADE schema load functionality to load and rename a schema, if the superschema of the target schema is not specified or not found.

Action

Edit the command line of the batch schema load executable to specify the superschema of the target schema, by setting the **subschemasOf** parameter with a valid superschema.

8501 Reorganisation during load failed

Cause

This error occurs if the JADE Schema Load utility (**jadload.exe**) is unable to reorganize a class or subclass following a schema file load.

Action

Refer to the reorganization log file (**jomreorg.log**) in the directory identified by the **TEMP** environment variable or the current directory.

The log file contains the class in which the error occurred and enables you to correct your data before actioning a successful reorganization.

8502 Cannot load into read only schema

Cause

This error occurs if the JADE Schema Load utility (**jadload.exe**) attempts to load into a read-only schema.

Action

Set the **ReadOnlySchema** parameter to **false** in the appropriate [**JadeServer**] or [**JadeClient**] section of the JADE initialization file.

8503 Unable to generate Forms from their descriptions

Cause

This error occurs when JADE encounters a problem in generating the forms from their definitions. This error could be caused when a user cancels the load operation.

Action

Try the operation again or contact your local JADE support center or JADE Support if your JADE licenses include support.

8504 The Report Writer is not installed in this system

Cause

This error occurs when attempting to load JADE Report Writer files when the **JadeReportWriterSchema** schema has not been loaded.

Action

Load the **JadeReportWriterSchema** schema.

8505 The specified Report file could not be loaded

Cause

This error occurs when a specified report view file cannot be found. The most-common reasons for a file not being found are:

- The path or directory name cannot be found.
- Access to the file is denied.

Action

Check the file path name and permissions for the required report view file.

8506 The specified Report Definitions file could not be loaded

Cause

This error occurs when a specified report definitions file cannot be found. The most-common reasons for a file not being found are:

- The path or directory name cannot be found.
- Access to the file is denied.

Action

Check the file path name and permissions for the required report definitions file.

8507 The specified Report Definitions file could only be partially loaded

Cause

This error occurs when an error is found in some reports in a report definition file.

Action

Check the JADE Report Writer log (**JadeReportWriter**<*n*>.log) for details.

8508 The specified schema could not be found

Cause

This error occurs when a specified schema cannot be found.

Action

Refer to the standard output file for details. Check the schema name of the required schema.

8509 The specified schema could not be deleted

Cause

This error occurs if the JADE Schema Load utility (**jadloadb.exe**) is unable to delete a schema; for example, a schema cannot be deleted if it has any subschemas defined.

Action

Check the standard output file for details. If necessary, delete any subschemas and then retry the operation.

8510 Methods in error were detected by the schema load

Cause

This error occurs if the JADE Schema Load utility (**jadloadb.exe**) finds any methods in error during the schema load. It is issued as a warning for your information only.

The schema load has been completed.

Action

If the methods should have compiled cleanly, you should fix the errors and rerun the schema load.

8511 Reorganisation is required

Cause

This error occurs when one or more of the schemas that are loaded require reorganization.

The JADE Schema Load utility (**jadloadb.exe**) has not initiated the reorganization because the option to suppress reorganizations has been specified.

Action

The reorganization must be initiated manually from the JADE development environment or **JadeReorgApp**.

8512 Load cancelled

Cause

This error occurs when you request that a schema load operation is to be aborted.

Action

None.

8513 Invalid command file format

Cause

This error occurs if a command file header is not in the format of a valid JADE command file, or is in a format later than the version of JADE you are using.

Action

Check the **JadeVersionNumber** in the command file header.

Refer to your JADE messages (**jommsg.log**) log file for more details.

8514 One or more commands in command file did not complete successfully - refer to jommsg.log [obsolete]

Cause

This error occurs if one or more commands in a command file are not processed successfully by the JADE Schema Load utility (**jadloadb.exe**). It is a warning code only.

Action

Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

8515 Command file processing failed - refer to jommsg.log

Cause

This error occurs if the batch JADE Schema Load (**jadloadb**) executable, the **JadeSchemaLoader** application in **jadclient**, **jade**, or the **Application** class **startApplicationWithParameter** method detects a fatal error when processing a JADE command file.

Action

Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

8525 Reorganisation is required prior to DDB load

Cause

This error occurs if an HTML document was versioned or if you run **jadload** or **jadloadb** with **suppressReorg** set to **true** and a reorganization of control classes is required before the form and data definition (**.ddb** or **.ddx**) files can be loaded.

Action

Perform the reorganization and then load the HTML documents or form and data definition (**.ddb** or **.ddx**) files.

8526 DDB load failed

Cause

This internal error can occur during a DDB or DDX form and data definitions file load that is initiated by **jadload** or **jadloadb**.

Action

Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

8527 Load results in one or more incomplete schemas

Cause

This error occurs after loading multiple schema files, if references to package schemas remain unresolved. This can happen if export package definitions do not match the corresponding import package definitions.

Action

Refer to your JADE messages (**jommsg.log**) log file for more details about this error.

8600 through 8699 JADE Source Control Errors

This section describes errors that can be returned if you experience problems when using the JADE source control.

8602 General Git operation error

Cause

This error occurs when an unspecified error occurs during a Git operation. It usually indicates an unexpected error.

Action

Try closing the any Git client dialogs that are open before repeating the operation that caused this error.

8603 Ambiguous Git reference, object, or path

Cause

This error occurs when a Git reference, Git object, or path wildcard is ambiguous and cannot uniquely identify an item. It can occur if a specified Secure Hash Algorithm (SHA) is too short or more than one file matches a path wildcard.

Action

Repeat the failing operation with a longer SHA or an unambiguous path wildcard.

8604 Operation requires a working directory

Cause

This error occurs when an operation that requires a working directory is performed against a bare repository. Operations such as staging, merging, and rebasing require a working directory.

Action

The JADE Git client is not intended to be used against a bare local Git repository. Clone a local copy of the Git repository with a working directory.

8605 Checkout prevented by conflicting changes

Cause

This error occurs when a checkout operation cannot be performed because of a conflicting change that is staged in the repository index or unstaged in the working directory.

Action

Resolve any outstanding conflicts of files in your working directory.

8606 Commit operation would create an 'empty' Commit

Cause

This error occurs when a commit operation has the effect of not changing the content of the Git repository; that is, the files being committed are identical to the existing ones. This would create an 'empty' commit.

Action

If you want to explicitly create an 'empty' commit, use the advanced features of another Git client.

8608 Specification is incorrect

Cause

This error occurs when the provided specification is incorrect; for example:

- It is syntactically incorrect
- The specification refers to a Git object that is of an incorrect type (for example, it is asking to create a branch from a Git blob)
- Specifying an incorrect branch name or a URL that does not specify a repository (for example, <http://github.com> by itself)

Action

Correct the specified branch or URL.

8609 Attempting to open a locked file

Cause

This error occurs when an operation attempts to open a locked file such as the repository **index.lock** file.

Action

Determine which file is being locked and which process is locking it.

Either wait for that process to release the locked file, or cancel the operation that is being attempted by closing the any open Git client dialogs. (Alternatively, try to free the locked file by using another mechanism.)

8610 Duplicate name for a Git resource

Cause

This error occurs when a Git branch, Git tag, Git reference, or Git remote with the same name already exists in the repository; for example, a branch called **HEAD** cannot be created, as it would clash with the symbolic Git reference **HEAD**.

Action

Change the resource name so that it no longer clashes.

8611 Push operation would lose commits

Cause

This error occurs when a push operation cannot be performed against the remote repository without losing commits. It occurs when a non-fast forward merge is required but an attempt is made to force the merge through.

Action

Prior to the push operation, perform a non-fast forward merge of any files required.

8612 Specified Git resource does not exist

Cause

This error occurs when an attempt is made to use a Git resource (for example, a commit, tag, remote, and so on) that does not exist.

Action

Check that any specified Git resource (commit, tag, remote, and so on) actually exists and that it is spelled correctly.

8613 Required Fetch for Merge was not part of a Pull operation

Cause

This error occurs when the specified reference with which to merge was unexpectedly not fetched as part of a pull operation.

Action

Attempt the fetch and pull operations again before repeating the merge operation.

8614 Git object cannot be found from Tag

Cause

This error occurs when a tag cannot be unwrapped to locate the associated Git object. This is usually caused by an incorrectly specified Git reference.

Action

Specify the tag name correctly.

8616 Repository path is invalid

Cause

This error occurs when a Git repository is being built with a path that does not point to a valid Git directory or working directory.

Action

Change the specified path to be a valid Git directory or working directory.

8617 Operation requires a non-empty Branch

Cause

This error occurs when an operation such as a checkout that requires an existing branch is performed against a branch that has not yet been created. There is therefore nothing for the operation to work on.

Action

Specify a non-empty branch for the operation or create and populate the specified branch.

8618 Explicit paths could not be matched

Cause

This error occurs when an explicit path or a list of explicit paths could not be matched by an operation that requires one or more valid paths or files.

Action

Check that the paths and files specified by the operation exist.

8619 Operation requires a fully merged index

Cause

This error occurs when an operation such as a commit or rebase operation that requires a fully merged repository Index is performed against a repository index that contains files that have not been merged.

Action

Merge any outstanding un-merged files before repeating your original operation.

8620 Repository authentication

Cause

This error occurs when an authentication error occurred accessing a Git repository.

It *may* correspond to an HTTP or HTTPS status code of 401, if either of these protocols is being used.

Action

Check that your authentication credentials are correct and that you can successfully access any remote repository, perhaps by using other tools.

8621 Network protocol or transfer error

Cause

This error occurs when a network protocol or transfer error is detected.

It *may* be caused by a non-zero HTTP or HTTPS status code (other than code 401, 403, or 404) if either of these protocols is being used.

Action

Check that you can still successfully access any remote repository, perhaps by using other tools.

8622 Repository authorization issue

Cause

This error occurs when the requested repository was not found or it was not accessible because of an authorization issue.

It *may* correspond to an HTTP or HTTPS status code of 403 or 404, if either of these protocols is being used.

Action

Check that you can successfully access the URL of the remote repository, perhaps by using other tools.

8623 Cannot access the repository server due to a certificate problem

Cause

This error occurs when the SSL certificate of the repository server fails validation on the client. There are number of potential underlying causes, including:

- A self-signed certificate is installed on the repository-hosting server. Self-signed certificates are not trusted, because they are generated by the server; not by a Certificate Authority (CA).
- The server's certificate has been revoked or it has expired.
- The common name to which the SSL certificate is issued (for example, **git.acme.com**) does not exactly match the host name part of the repository URL.

The error report includes additional information about the server certificate such as the issuer and subject, when this is available.

Action

Contact the administrator of your Git repository server (or a security specialist in your organization) for advice and a solution.

If the server has a self-signed certificate, one approach to solving this issue is to add the required certificates to the Windows Certificate Store as a trusted certificate.

Note The JADE Git client does *not* observe the Git **http.sslVerify** configuration setting, which is a workaround to allow Git clients to ignore server certificate validation.

8700 through 8799 JADE Upgrade Errors

This section describes errors that can be returned if you experience problems when using the JADE upgrade utility.

8701 General Upgrade error code

Cause

This error occurs when part of the upgrade process is not recognized.

Action

Check your JADE messages (**jommsg.log**) log file for information about why the upgrade failed.

If you cannot resolve the error and your JADE licenses include support, contact your local JADE support center or JADE Support.

8703 Method could not be recompiled - no source present

Cause

This error occurs when a method needs to be recompiled as part of the upgrade validation and the method source is not present.

Action

The methods that need to be recompiled are listed in the **jadeupgrade.log** file.

8705 Outstanding reorganisations were detected in this database

Cause

This error occurs if there are classes that need reorganizing in this database before the upgrade can proceed.

Action

Reorganize any schemas and their classes that need reorganizing before proceeding with the upgrade.

8706 The parameter <parameter> specified for the Upgrade is invalid

Cause

This error occurs if the specified parameter was found to be invalid.

Action

Check that the parameter specified is correct, and correct it if necessary.

8710 Checked out methods were detected in this database

Cause

This error occurs if there are methods still checked out in the database to be upgraded.

Action

Check in any checked out methods before proceeding with the upgrade.

8711 Uncompiled or in error methods were detected in this database

Cause

This error occurs if there are uncompiled methods or methods in error detected in the database to be upgraded.

Action

Correct or comment out the offending code before proceeding with the upgrade.

8722 Cannot upgrade an Ansi database with Unicode files or vice versa

Cause

This error occurs when an attempt is made to upgrade an ANSI database with a Unicode version of JADE or an attempt is made to upgrade a Unicode database with an ANSI version of JADE.

Action

Contact JADE to obtain the appropriate version of the product for your type of database.

8723 Upgrade validation failed: refer to jommsg.log

Cause

This error occurs when the validation of an upgrade fails.

Action

Use the information contained in your JADE messages (**jommsg.log**) log file to determine the cause of the failure and then take corrective action before restarting the validation.

If the validate fails (for example, because a method conflict is detected), you can use the JADE development environment to correct the error.

When the development environment starts, it checks for an incomplete upgrade and prompts you to respond if you want to restart an incomplete validate.

In general, the steps required to fix up a failed validate are to start the JADE development environment, correct the problem, restart the JADE development environment, and then click the **Yes** button when prompted to specify if the validate should be restarted.

8724 - Cannot upgrade if JadeReportWriterSchema is present

Cause

This error occurs if the **JadeReportWriterSchema** schema exists in the JADE system when the upgrade process checks for the **JadeReportWriterSchema** schema when it starts up, before starting the migration. If that schema exists, the upgrade is aborted.

Action

Before upgrading to the current (latest) JADE feature release:

- If the **JadeReportWriterSchema** has been installed, you must extract the data and then delete the schema before upgrading; that is:
 - a. In the JADE Report Configuration application, select the **Unload All** command from the View menu to unload all of your report data to a **.rwa** file. In the **File Name** text box of the Unload All dialog, specify the name and location of the file to which you want to extract all of your report writer data; for example:

```
d:\jade\rpts\alldata.rwa
```

Alternatively, you can use the **executeMethod** in the batch JADE Load utility to extract all reports, as shown in the following example.

```
jadloadb path=d:\jade\system ini=d:\jade\myjade.ini  
schema=JadeReportWriterSchema executeSchema=JadeReportWriterSchema  
executeClass=JadeReportWriterGlobal executeMethod=unloadAllToFile  
executeParam=d:\jade\rpts\alldata.rwa
```

- b. Delete the **JadeReportWriterSchema** schema by using the **Remove** command in the Schema menu of the JADE development environment or by using the **deleteSchema** parameter in the batch JADE Load utility; for example:

```
jadloadb path=d:\jade\system ini=d:\jade\myjade.ini  
deleteSchema=JadeReportWriterSchema
```

When the upgrade to the current (latest) JADE feature release has finished:

- If the **JadeReportWriterSchema** is required, reload the **JadeReportWriterSchema** from the previous release and then reload the reports and data extracted from the previous release, by using one of the following actions.
 - In the JADE Report Configuration application, select the **Load All** command from the View menu to load all report data from your extracted **.rwa** file.

In the **File Name** text box of the Load All dialog, specify the name and location of the report data extract file you want to load; for example:

```
d:\jade\rpts\alldata.rwa
```

- Use the **reportLoadAllFile** parameter in the batch load utility to specify the fully qualified name of a single unload (extract) file that contains all JADE Report Writer view, folder, system option, user, and report definitions that you want to load, as shown in the following example.

```
jadloadb path=d:\jade\system reportLoadAllFile=d:\jade\rpts\alldata.rwa  
ini=d:\jade\myjade.ini
```

8900 through 8999 XML Processing Errors

This section describes errors that can be returned if you encounter problems when performing Extensible Markup Language (XML) processing.

8900 XML parser could not be created

Cause

This error occurs if an instance of the XML parsing engine could not be created.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

8901 XML parser error

Cause

This error occurs when an error occurs during the parsing of an XML document; for example, if the document is not well-formed.

Action

The **extendedErrorText** property of this exception contains the description of the parsing error. The **errorItem**, **fileName**, **lineNumber**, and **columnNumber** properties identify the item in error.

Correct your document and parse it again.

8902 XML root element is already defined

Cause

This error occurs when you attempt to add a root (top-level) element to an XML document and one exists already. An XML document must have only one root element.

Action

Change your application code to ensure that this does not occur.

8903 XML Document Type is already defined

Cause

This error occurs when you attempt to add a document type declaration to an XML document and one exists already. An XML document can have only one document type declaration.

Action

Change your application code to ensure that this does not occur.

8904 XML node cannot be null

Cause

This error occurs you attempt to pass a null node reference to an XML processing method and the parameter cannot be null; for example, specifying a null value as the destination position when moving a node in the document tree.

Action

Change your application code to ensure that this does not occur.

8905 XML hierarchy request is invalid

Cause

This error occurs when you attempt to add a node to an XML document at an invalid position; for example, moving an element to before a document or attribute.

Action

Change your application code to ensure that this does not occur.

8906 XML Jade string to UTF8 conversion failed

Cause

This error occurs when you attempt to parse an XML document that cannot be converted from JADE native format to UTF8 format.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

8907 XML UTF8 to Jade string conversion failed

Cause

This error occurs when you attempt to parse an XML document that cannot be converted from UTF8 format to JADE native format.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

8908 Cannot access an XML parser created by a different node

Cause

This error occurs when an attempt is made to access an XML parser object on a different node from the one that created the parser; for example, when a parser is opened on a client node and a server method attempts to use the parser, this exception is raised.

Action

Change your application code to ensure that when parsing XML documents on different nodes, the parser object is created on the same node that performs the parsing.

8909 XML class mapping is invalid

Cause

This error occurs when you attempt to set an invalid mapping for a [JadeXMLNode](#) class. The mapping is used when instances are created during the parsing of an XML document.

The [setClassMapping](#) of the [JadeXMLDocumentParser](#) class, which is used to define the mapping, has the following signature.

```
setClassMapping(nodeClass: Class; mappedClass: Class) updating;
```

The **nodeClass** parameter must be a valid [JadeXMLNode](#) system subclass (excluding the [JadeXMLDocument](#) class) and the **mappedClass** parameter must inherit from the subclass specified in the **nodeClass** parameter.

This error can also be returned if all concrete [JadeXMLNode](#) classes have not been mapped to user subclasses when parsing a persistent document.

Action

Correct your application code to ensure that this does not occur.

8910 Cannot parse XML document into a persistent object

Cause

This error occurs when you attempt to parse an XML document into a persistent object tree using the [parseFile](#) or [parseString](#) method of the [JadeXMLDocument](#) class, which is not allowed.

To parse persistent documents, you must use the [JadeXMLDocumentParser](#) class and set up a mapping of node classes to user subclasses.

Action

Use the [JadeXMLDocumentParser](#) class to parse persistent XML documents.

8950 General RegexLibrary failure

Cause

This error occurs as the result of an internal regular expression (Regex) failure.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

8951 Regex failed during an operation

Cause

This error occurs when a problem is encountered during a **match**, **replaceAll**, or **replaceFirst** method call.

Action

Check the extended error information for the cause of the problem and change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

8952 Regex failed with an invalid pattern

Cause

This error occurs when coding mistakes are detected within the specified Regex pattern string.

Action

Check the extended error information for the cause of the problem and change your code to avoid this error; for example:

- Are you using PCRE2 syntax?
- Try pasting your Regex string into an online Regex validator.
- If you want to use the ECMAScript-style Regex, did you set the **JadeRegexPattern** class **setUnicodeECMAScriptDialect** method to **true** before compilation?

8953 Attempt to use Regex from a different node

Cause

This error occurs when you use a **JadeRegexPattern** object from a different node to that on which it was created.

Action

If you want to use the pattern on a node that differs from the node where it was created, recompile the pattern.

8954 Pattern or search text UTF8 validity checks failed

Cause

This error occurs when the specified text or Regex pattern did not pass validity checks.

Action

Check the extended error information for the cause of the problem and change your code to avoid this error. The method and source position containing the expression at which the exception was raised are reported in the error description.

8955 Pattern was not compiled before doing a Regex operation

Cause

This error occurs when a Regex operation such as a **scan** or **isMatch** method was run for a **JadeRegexPattern** that was not compiled first.

Action

Compile your Regex pattern before calling a scan or a match method with the **JadeRegexPattern** class **compile** method.

If you are calling these methods asynchronously from the **compile** method, you can call the **isCompiled** method to check the compilation state of the **JadeRegexPattern** object.

8956 A Regex operation took too long and timed out

Cause

This error occurs when a Regex operation took longer than the timeout period specified by the **JadeRegexPattern** class **setTimeoutValue** method. As the timeout value defaults to zero (0), this error occurs only if you specify a timeout (defined in milliseconds).

Action

If this error occurs, you can:

- Make the Regex pattern more efficient.
- Give the Regex operation a smaller subject string.
- If you are letting users define Regex patterns, check that they are not defining malicious patterns, in which case doing nothing would be the best thing, as this prevents your process being hung by malicious patterns.
- Increase the timeout value or remove it.

8957 A Regex operation was either interrupted or forced off

Cause

This error occurs when a Regex process was forced off or it was interrupted by the JADE Monitor or programmatically by another process.

Action

Find where **System** class **forceOffUser** or **interruptUser** method could be used in your application to cause the process to be terminated or interrupted.

If a Regex operation is interrupted, it cannot be resumed but it can be retried.

9002 and 9005 JADE Database Utility Exit Codes

Exit code 9002 or 9005 can be returned if you encounter problems with the JADE Database utility, indicating that the batch operation failed.

Batch database utility program operations that consist of a number of steps or that take longer than a few seconds (for example, **backup**, **certify**, or **restore**) can optionally display a progress report of each specified operation while that operation is running, unless status output is suppressed.

Error information is output to **stderr**. At the end of the operation, the running display provides a completion report, highlighting the number of files processed and the number of files found with error.

Detailed results of the operations are written to the appropriate log file in your specified database directory, to enable you to investigate further, if required.

11000 through 11099 Web Services Errors

This section describes errors that can be returned if you experience problems with JADE Web services.

11001 Web Service class %s does not exist

Cause

This error occurs if a Web service client is requesting a Web service that does not exist.

Action

Correct the Web service class name.

11002 Web Service method %s does not exist

Cause

This error occurs if the Web service method does not exist. It can also occur if the Web service request message does not contain a **<body>** element.

Action

Correct the Web service method name.

11003 Parameter %s to method %s must be of type %s

Cause

This error occurs if the parameter sent to the method was not of the correct type.

Action

The Web Services Description Language (WSDL) that was imported does not match the service request.

Get the latest copy of the WSDL, re-import and try the operation again. If this still fails, contact the Web service provider.

11004 Number of licenses exceeded

Cause

This error occurs if you attempt to start or run more JADE Web clients than the maximum number of Web client (HTML) licenses permitted by the terms of your license.

Action

If you need to increase the number of Web client (HTML) licenses permitted under your application license, contact JADE Licensing.

For details about contacting JADE Licensing or ordering a license, contact jadelicensing@jadeworld.com or access our Web site at <https://www.jadeworld.com/jade-platform/developer-centre/pricing-licensing>.

11005 Minimum response time exceeded for this request

Cause

This error occurs if the Web service client user does not receive a response from the JADE application within the configured time limit.

When the timer event occurs, a default message is sent back to the client and the current request is terminated.

Action

Resubmit the request or change your session minimum response time, by specifying the minimum response time (in seconds) in the **Minimum Response Time** text box on the **Web Options** sheet of the Define Application dialog, as required.

A minimum response time value of zero (0) indicates no minimum (infinity).

11006 Session has been ended

Cause

This error occurs if a session is programmatically ended. This error occurs only when session handling is enabled.

Action

Start a new session.

11007 Session has timed out

Cause

This error occurs if the Web session has ended because it received no requests within the specified timeout period. This error occurs only when session handling is enabled.

Action

Start a new session or change the session timeout configuration, by specifying the minimum session timeout time (in minutes) in the **Session Timeout** text box on the **Web Options** sheet of the Define Application dialog, as required.

A minimum response time value of zero (0) indicates no minimum (infinity).

11008 Service is currently unavailable

Cause

This error occurs if the requested Web service is currently unavailable.

Action

Check with your network administrator or Web service provider. It may be that the request can be resubmitted at a later time.

11009 There is a version mismatch for the requested service

Cause

This error occurs if the application version specified in the incoming target namespace does not match that of the target namespace of the application. If these do not match and version control checking is active, a Simple Object Access Protocol (SOAP) fault is raised.

Action

Change your application code to use the correct (matching) versions.

11010 Web Service parameter %s of method %s does not exist

Cause

This error occurs if a request to a JADE Web service provider contains a parameter that does not exist in the current version of the Web service. This can occur if the Web service consumer is using an outdated version of the WSDL for the service.

A SOAP error response is sent back to the consumer and processing continues.

Action

Check that the Web service consumer is using an up-to-date version of the WSDL for the service.

11051 Unrecognised response from service

Cause

This error occurs if the Web service provider is not available.

Action

Check the request end-point. If this is correct, you may need to contact your Web service provider.

11052 The service returned a fault message

Cause

This error occurs if a fault is returned from the Web service provider.

Action

Check that the parameters to the method are correct.

11053 Property value for an enumerated property is not valid

Cause

This error occurs if the value being set for a property in an imported Web service class does not match the list of enumerated values.

Action

The [extendedErrorText](#) property of the exception that is raised contains the list of valid values for this property. Ensure that the value to which you are setting this property is one of these values.

11054 String or Binary property has exceeded its maximum defined length

Cause

This error occurs to a Web service client if a [String](#) or [Binary](#) value that is sent to a Web service exceeds the maximum length defined for it.

Action

The extended error text of the error message defines the maximum length for the [String](#) or [Binary](#) value in question.

Change your Web service client application code so that the length of the [String](#) passed to the Web service does not exceed the specified length.

11055 Result of expression overflows Decimal precision

Cause

This error occurs to a Web service client if a [Decimal](#) value sent to a Web service exceeds the number of decimal places defined for it.

Action

The extended error text of the error message defines the length and the maximum number of decimal places for this value.

Change your Web service client application code so that these values are not exceeded.

11056 The requested message type is invalid

Cause

This error occurs if an incoming Web message does not conform to the supported message types.

The most-likely cause is that you are using the wrong version of the [jadehttp](#) module. When communicating with JADE 2020 applications, you must use the [jadehttp](#) module supplied with JADE release 2020.

Action

Check that you are using the correct **jadehttp** module. You may have to restart the Web service for the new module to take effect. Refer to your JADE documentation for more details.

11057 Web Service exposed list does not exist

Cause

This error occurs if an incoming Web message refers to an exposure list that does not exist in the current version of the Web service. This can occur if the Web service user is using an outdated version of the WSDL for the service.

A SOAP error response is sent back to the Web service user, and processing continues.

Action

Check that the Web service user is using an up-to-date version of the WSDL for the service.

11058 A property referenced a sub-class instance that is not in the exposure

Cause

This error occurs if a property instance was encountered in a Web service where the property instance is a subclass of the defined property type and that subclass is not included in the Web service exposure.

Action

Update the exposure to include the subclass.

11059 Result of expression overflows Integer precision

Cause

This error occurs to a Web service client if a **Integer** value that is sent to a Web service exceeds the maximum length defined for it.

Action

The extended error text of the error message defines the maximum length for the **Integer** value in question. The property name and class in which the overflow occurred is output to the respective **Exception** class **errorItem** and **extendedErrorText** properties.

Change your Web service client application code so that the length of the **Integer** passed to the Web service does not exceed the specified length.

11081 WSDL Generation Failed

Cause

This error occurs when a WSDL generate is unsuccessful. The extended error text contains details of the error.

Action

This is an internal error.

If your JADE licenses include support, contact your local JADE support center or JADE Support.

11091 Submitted HTML form is out of sequence

Cause

This error occurs in a Web-enabled application occurs when the **usePageSequencing** option is set to **true** and the HTML page that is returned has a sequence number that is less than the last sequence number that was used.

The error message contains details of the actual and expected sequence numbers and the incoming message.

Action

Trap the exception and return a suitable message to the Web browser user.

11100 through 11199 JADE REST Service Errors

This section describes errors that can be returned if you experience problems with JADE REST-based Web services.

11101 Class referenced in Rest Services request does not exist

Cause

This error occurs if the class defined by the application for REST services does not exist.

Action

Fix the application definition.

11102 Requested Rest Service method does not exist

Cause

This error occurs if the method to be called by the REST service does not exist in the [JadeRestService](#) subclass used by the application.

Action

Add the method or fix the calling client logic.

11103 A parameter of the called method is usage output

Cause

This error occurs when a method to be called by the REST service has a usage **output** parameter, which is not allowed.

Action

Fix the method signature by changing the parameter usage.

11104 A parameter type of the called method is not supported by Rest Services

Cause

This error occurs when a method to be called by the REST service has a parameter type not supported by REST services, or a **ParamTypeList** parameter is not the last parameter.

Action

Fix the method signature.

11105 The object parameter required by called Rest Services method was not supplied

Cause

This error occurs when a method to be called by the REST service requires an object parameter that was not supplied in an XML or JSON script attached to the request.

Action

Fix the calling client.

11106 The signature of the called Rest Services method does not match the supplied url entities

Cause

This error occurs when the signature of a method to be called by the REST service does not match the parameters supplied in the URI path of the request.

Action

Fix the calling client.

11107 The Rest Service response xml cannot be generated because circular reference handling is disabled

Cause

This is an internal JADE error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

11108 The Rest Service response json cannot be generated because the same object is referenced twice

Cause

This error occurs when the object returned by the called method to be passed back to the client has circular references or multiple references to the same object. For this type of structure, the response can be handled only using the Newtonsoft JSON format.

Action

Change the structure of the objects being returned. Alternatively, switch to the Newtonsoft JSON format.

11109 The Rest service returned a fault message

Cause

This error can occur in the following cases.

- The XML being parsed contains a **Fault** type object.
- The XML content is faulty; for example, it contains a reference to an object that does not exist.
- There are internal issues in parsing the XML.

Action

Check the additional exception information and the passed XML to determine why the error occurred, and then fix the issue.

11110 The Rest service requested message is invalid

Cause

This error occurs when the class type specified in the JSON or XML for an object is not a subclass of the expected type. It can also occur if the URI passed is empty, and therefore does not contain a *noun* combined with the HTTP verb to construct the name of the method to be called.

Action

Fix the calling client.

11111 Unrecognised response from Rest service

Cause

This error occurs when the XML parser finished the parsing without finding the complete content, or the required method object parameter could not be found.

Action

Check the XML content sent from the client and the method signature.

11112 Cannot access a Rest Services object created by a different node

Cause

This could be an internal error. In addition, it could occur if a **JadeRestService** object is used outside the REST services framework.

Action

Check the context where the error occurred. **JadeRestService** classes can be used only by the REST services framework.

11113 The called Rest Service method can only have one object parameter

Cause

This error occurs when the signature of the called REST service method has more than one object parameter.

Action

Fix the signature of the method.

11114 Invalid numeric value encountered in Json

Cause

This error occurs when the JSON script contains an invalid numeric value; for example, **1h** or it is not followed by an expected delimiter.

Action

Fix the JSON construction in the calling client.

11115 Invalid string value encountered in Json

Cause

This error occurs when the JSON script contains a badly-formed string; for example, an invalid hexadecimal value, an invalid escape sequence **\<c>**, no trailing quote, or it is not followed by the expected delimiter.

Action

Fix the JSON construction in the calling client.

11116 Invalid identifier encountered in Json

Cause

This error occurs when the JSON script contains a bad identifier, which is not **true**, **false**, **null**, a string enclosed in quotes, or it is not followed by the expected delimiter.

Action

Fix the JSON construction in the calling client.

11117 Unmatched array delimiter encountered in Json

Cause

This error occurs when the JSON script contains a closed bracket character (**]**) that does not match the starting delimiter of the sequence.

Action

Fix the JSON construction in the calling client.

11118 Unmatched delimiter encountered in Json

Cause

This error occurs when the JSON script contains a closing brace character (}) that does not match the starting delimiter of the sequence.

Action

Fix the JSON construction in the calling client.

11119 Unexpected character encountered in Json

Cause

This error occurs when the JSON parser encounters an unexpected character or character sequence.

Action

Fix the JSON construction in the calling client.

11120 Json definition is incomplete

Cause

This error occurs when the JSON script is incomplete. The JSON text did not have matching closing characters (}, /, or]) for all of the opening characters ({, /, or [, respectively).

Action

Fix the JSON construction in the calling client.

11121 Json binary value is invalid

Cause

This error occurs when the JSON script contains an invalid binary value that starts with the opening bracket character ([) character.

Action

Fix the JSON construction in the calling client.

11122 Json date value is invalid

Cause

This error occurs when the JSON script contains an invalid date value.

Action

Fix the JSON construction in the calling client.

11123 The reference tag in the Json text was not previously encountered

Cause

This error occurs when the JSON script contains a reference to an object where the tag had not previously been encountered.

Action

Fix the JSON construction in the calling client.

11124 Duplicate reference tag in the Json text

Cause

This error occurs when the JSON script contains an object id that has previously been used in the JSON for another object.

Action

Fix the JSON construction in the calling client.

11125 Requested Rest Service method is protected

Cause

This error occurs when the method called by the REST service on the user [JadeRestService](#) subclass used by the application is protected.

Action

Fix the JSON construction in the calling client.

11126 A Rest Service method was called but the service was never initialised

Cause

This error occurs if the application is already attached to a JADE Web Service Manager, because the JADE REST service has not been initialized and the logic assumes it has (for example, when creating a transient instance of a **JadeRestService** subclass and calling the **processRequest** method).

Action

Change your logic. In a JADE REST service application, the **processRequest** method can be initiated only by the JADE REST services framework.

11127 JadeRestService.reply was called but there is no web message to reply to

Cause

This error occurs when the **JadeRestService** class **reply** method is called and its processing is not associated with a received web message.

Action

Re-implement the **reply** method on the **JadeRestService** subclass that is being used.

Note When using manual testing processing, the **reply** method should not call **inheritMethod**.

11128 Requested Rest Service method is a Type Method and cannot be called from Rest Services

Cause

This error occurs when a method called as the endpoint of a REST service request is a type method.

Action

Fix your application code, by removing or renaming the type method so that it cannot be called from a REST service.

11151 Cannot access a JadeJson object created by a different node

Cause

This could be an internal error. In addition, it could occur if a **JadeJson** object is created by a different node.

Action

Check the context where the error occurred, and fix your logic.

11152 The json cannot be generated because the same object is referenced twice

Cause

This error occurs when the object returned by the called method to be passed back to the client has circular references or multiple references to the same object and the format is **Format_Json_Microsoft**. For this type of structure, the response can be handled only using the Newtonsoft JSON format.

Action

Change the structure of the objects being returned. Alternatively, switch to the Newtonsoft JSON format.

11153 The type found in the Json is unknown

Cause

This error occurs when the type of an object specified in the JSON is not a known JADE class.

Action

Fix the task generating the JSON to specify the correct JADE class.

11154 The type of the object to decode is invalid

Cause

This error occurs if the value of the **type** parameter of a parse method (for example, **parsePrimitive**) in the **JadeJson** class is null.

Action

Fix your logic.

12000 through 12999 JADE Remote Access Program Errors

The errors in this section can occur when running the JADE Remote Access Program (**jadrap.exe**).

12002 Open Failed

Cause

This error occurs if the JADE server failed to open the network connection type in listening mode.

Action

Check that all configured network types have been installed on the server. Check your JADE messages (**jommsg.log**) log file for more details.

12006 Server node name cannot exceed 100 characters

Cause

This error occurs if the specified server name is too long.

Action

Shorten the server name specification to a maximum of 100 characters (for example, in the **NodeName** parameter in the [NonGuiClient] section of the JADE initialization file).

12008 Invalid transport type

Cause

This error occurs if an unknown network type identifier has been specified as the *transport-type* value of the **NetworkSpecification<specification-number>** parameter in the [JadeServer] section of the JADE initialization file.

This error can occur if the *transport-type* value of the **NetworkSpecification<specification-number>** parameter in the [JadeServer] section of the JADE initialization file is maintained manually and an unrecognized network identifier is specified.

Action

Maintain this parameter only by using the TCP/IP Network dialog of the JADE Remote Access Program.

12009 No network active

Cause

This error occurs if no recognized network type was specified as the transport mechanism.

Action

Specify at least one network type for the JADE server.

12019 Service already exists

Cause

This error occurs if you attempt to create a service that already exists.

Action

Use a different JADE service node name for the current service that you are creating.

12020 Cannot rename running service

Cause

This error occurs if you attempt to rename the JADE server node while the service is running.

Action

Shut down the JADE service before renaming the server node.

12024 - Cannot unregister running service

Cause

This error occurs if you attempt to unregister a service while it is still running.

Action

Shut down the JADE service before unregistering.

14000 through 14999 JADE User Interface Errors

The following sections describe JADE user interface errors. JADE user interface errors are those usually associated with runtime window actions.

14001 Out of memory

Cause

This error occurs if a memory allocation request fails. This occurs only if the application had reached the stage where a huge amount of memory has been allocated, or the size of your system Page file is inadequate.

Action

Check the size of your paging file (the virtual memory) from the Windows Control Panel **System** command, by clicking the **Performance Options** button on the **Advanced** sheet of the System Properties dialog.

If the total paging file size for all drives displayed in the Virtual memory group box on the Performance Options dialog is inadequate, click the **Change** button and specify the appropriate initial and maximum paging file sizes in the Virtual Memory dialog. (See Windows help for notes about managing your computer's performance by adjusting the paging file size.)

14002 Property can only be accessed after the window is created

Cause

This error occurs when a property that is a directive to the physical window associated with a form or control is accessed and there is no such associated window; for example, accessing the [itemText](#) array property of the [ListBox](#) class has no meaning if there is no associated window.

Action

Check the window object that is being accessed. To get this error, the object is not a running form or a control that is attached to a running form.

14003 Method can only be called after the window is created

Cause

This error occurs when a method that is a directive to the physical window associated with a form or control is accessed and there is no such associated window; for example, calling the **unload** method for a form has no meaning if there is no associated running form.

This error occurs if the object is not a running form, or the object is not a control that is attached to a running form.

Action

Check the window object that is being accessed.

14004 Required schema must be included on the Command line using schema=<name>

Cause

This error occurs if JADE cannot determine the schema to which your application belongs.

When initiating JADE, the command line must contain the application schema (except when running the JADE application).

Action

Include the schema argument on the command line, as follows.

```
schema = name
```

For example:

```
schema = manufacturing
```

14005 Window associated with this Jade object cannot be located

Cause

This error occurs if your code has gained access to the window object of another application. This object has a physical window attached, and your code attempts to modify that physical window. As this can cause problems (for example, Windows deadlocks), JADE prevents you from doing this.

Action

Identify where this is happening in your application code and then change your code to prevent this.

14006 Property value type or Method parameter type differs from schema definition

Cause

This error occurs when the wrong type of property value or method parameter is received internally. The property type or method parameter type does not match the schema definition type.

Action

If you are accessing a property or method from an external method, check the format being set in the [DskParam](#) parameters. For other situations, contact your local JADE support center or JADE Support if your JADE licenses include support.

14007 This pseudo array method is not implemented

Cause

This error occurs if you attempt to use methods that are not implemented for a control, form, or menu item when accessing arrays of GUI-related information. Examples of such arrays are the **backColor** property values for each item in a list box, the list of child controls of a parent control, and the list of child menu items of a menu item. These arrays are held in the JADE run time module and not in the database, which is why they are referred to as pseudo arrays.

For the arrays associated with list boxes, combo boxes, and tables (for example, **itemBackColor**), the only methods that are implemented are **at**, **atPut** (which enables you to use the square brackets notation to access the elements), **createIterator** (which allows logic to do a **foreach** over the array), **size**, and **size64**. All other array methods are not implemented and return this 14007 exception.

For the arrays associated with control and menu item children (for example, the **Window** class **allControlChildren** and **MenuItem** class **children** properties), the only methods that are implemented are **at** (which allows the use of square brackets to access the elements), **createIterator** (which allows logic to do a **foreach** over the array), **size**, and **size64**. These arrays are of fixed size based on the content, and cannot be directly added to or removed from. The array entry is created when the control, menu item, or list item is created and deleted when that entity is deleted.

All other array methods are not implemented and return this 14007 exception.

Action

Change your application code to use only the available methods and properties of the control. Do not use the other array methods.

14008 Invalid property value

Cause

This error occurs when the value of the property is not valid for the window or for the window in a specific state; for example, setting the **mousePointer** property to a value (for example, **100**) outside the range of the valid values.

Action

Check the documentation for the valid values for that property and then change your application code accordingly.

14009 Method called for non-running application object

Cause

This error occurs when a method has been called for an application object that is not actually running.

Action

Change your application code to use the correct application object for the method.

14010 Method does not apply to this control

Cause

This error occurs when the called method does not apply to the window object used.

Action

This error should not normally occur. If your JADE licenses include support, contact your local JADE support center or JADE Support.

14011 Invalid property array index

Cause

This error occurs if the index used to access an array property of a control is out of range; for example, `listbox1.itemText(0)`.

Action

Change your application code to ensure that a valid array index is used; for example, if a list box is empty, accessing any array values will generate this error.

14012 dragMode set to 2 for a window not in dragMode

Cause

This error occurs when code attempts to turn off drag mode for a window that is not in drag mode.

Action

Change your application code to prevent this.

14013 Picture is not an Icon

Cause

This error occurs when an attempt is made to set the application icon to an image that is not an icon.

Action

Change your code to ensure that the picture file is valid. Use the `app.checkPictureFile` method to check that the picture file is valid.

14014 Picture file open failed

Cause

This error occurs when a specified picture file cannot be opened. The file is no longer present, the file is protected from any access, or there is a physical problem with the file.

Action

Check that the physical file is valid and available.

14015 File does not contain an image type that can be handled

Cause

This error occurs when an attempt is made to load a picture from a file that does not contain a picture image recognized by JADE. It also occurs when using the [convertPicture](#) method or the [convertToFile](#) method of the [Binary](#) primitive type with binary data that does not contain valid image data.

Action

Use the [app.checkPictureFile](#) method to check that the selected file contains a valid picture image before attempting to load the picture.

14016 Wrong image type for this property

Cause

This error occurs when setting a [picture](#), [icon](#), or [mouseCursor](#) property to a picture type that is not appropriate for that property; for example, setting the forms icon to a bitmap image.

Action

Use the [app.checkPictureFile](#) method to check that the type of the picture being assigned is valid.

14017 Menu item not found

Cause

This error occurs when the physical menu associated with a menu object cannot be found.

Action

This error should not normally occur. If your JADE licenses include support, contact your local JADE support center or JADE Support.

14018 Menu item insertion failed

Cause

This error occurs if the insertion of a menu item into the physical menu fails.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

14019 Menu item modification failed

Cause

This error occurs if the modification of a menu item in the physical menu fails.

Action

This error should not normally occur. If your JADE licenses include support, contact your local JADE support center or JADE Support.

14020 Menu item removal failed

Cause

This error occurs if the deletion of a menu item from the physical menu fails.

Action

This error should not normally occur. If your JADE licenses include support, contact your local JADE support center or JADE Support.

14021 Item cannot be expanded because it is not visible

Cause

This error occurs if code attempts to expand an item of a list box that is not currently visible; that is, the item is a subitem and its parent item is not expanded.

Action

Change your application code to avoid this situation.

14022 Item level not incremental

Cause

This error occurs when an attempt is made to set the level of an item in a list box to an invalid hierarchical value; for example, setting the level to **3** when the level of the previous item is **1**.

Action

Change your application code to prevent this.

14023 Item Level can only be 1 through 63

Cause

This error occurs if an attempt is made to set the level of a list box item to a value not in the range **1** through **63**. A maximum of 63 levels are supported.

Action

Change your application code to prevent this.

14024 First item must be level 1

Cause

This error occurs if an attempt is made to set the level of the first item in the list box to a value other than **1**. Using the level features of a list box requires that the items are defined in a hierarchical structure, with the first item always being at level 1.

Action

Change your application code to prevent this.

14025 Maximum number of ListBox entries exceeded

Cause

This error occurs when your application code attempts to load more than 32,000 entries into a combo box or list box control.

Action

Change your application code so that fewer entries are loaded into the combo box or list box control.

14026 You are not licensed for this product

Cause

This error occurs if you attempt to use an OLE or ActiveX control product and there is no evidence of a license for the product on your workstation.

Action

Ensure that a proper licensed installation of the product has been performed on this workstation.

14027 This action is not valid from within the displayEntry or displayRow method

Cause

This error occurs if your application code attempts to perform an action that cannot be handled within the [displayEntry](#) or [displayRow](#) events. This is not allowed because it will affect the interaction of the collection already in progress.

Action

Change your code to circumvent this problem.

14028 The form control list contains an invalid control entry.

Cause

This error occurs if a control instance is encountered while building a form at run time that is not valid or is not a subclass of a non-abstract JADE **Control** class.

Additional information is logged in the JADE messages (**jommsg.log**) log file, listing the object identifier (OID) and the class name of the object, if possible.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support. The workaround is to reload the form from a **.scm** and **.ddb** or **.ddx** file.

14029 'Freelimage' library not present or failed to initialize

Cause

This error occurs if the **Freelimage** library is missing or fails to initialize.

Action

This library is normally installed in the JADE **bin** directory. Attempt to restore the missing files, or reinstall JADE.

14030 An MDI form cannot be Modal

Cause

This error occurs if a form is created as an MDI form and your application code then attempts to call the **showModal** method for it.

Action

Ensure that the form is not declared as an MDI form, or call the **show** method rather than the **showModal** method.

14031 Application start request failed because the thread limit was reached

Cause

This error occurs if you have exceeded the operating system resource limit on the number of application threads.

A JADE client can have the number of threads up to an operating system resource-defined limit. An attempt has been made to start another application that exceeded that limit.

Action

Change your code to prevent the application limit for a client from being exceeded.

14032 No startup form defined.

Cause

This error occurs if the application does not have a start-up form defined.

Action

Use the **Startup Form** combo box on the **Application** sheet of the Define Application dialog to define the start-up form for the application.

14033 Invalid parent for window.

Cause

This error occurs if your application code attempts to set the parent of a window control to an invalid value. This can be for one of the following reasons.

- The parent object does not have an attached window
- The parent object is a control that cannot have children

Action

Change your code accordingly.

14034 An MDI or Modal form cannot have its parent changed

Cause

This error occurs if the [setFormParent](#) method of the [Form](#) class attempts to change the parent of an MDI or modal form. These types of forms cannot have their form parent changed.

Action

Correct your application code to prevent this.

14035 Form not found in this application

Cause

This error occurs when the creation of a form fails to find the persistent definition of the form stored in the [Form](#) class.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

14036 The update parameter of the listCollection or displayCollection method must be false for transient collections

Cause

This error occurs if your application code attempts to call the **listCollection** list box method or the **displayCollection** table method requesting update of the list box contents or the table contents when the collection changes.

The **listCollection** or **displayCollection** request cannot be fulfilled because JADE does not provide notifications on transient objects. When the collection object is transient, the only valid value for the **update** parameter is **false**.

Action

Change the **listCollection** or **displayCollection** call to pass the **update** parameter with a value of **false**. You must then update the collection yourself; for example, by causing a user notification on the list box or table.

14037 The form was previously unloaded

Cause

This error occurs if an attempt is made to display a form that has been unloaded but has yet to be deleted.

Action

Change your application code to prevent this.

14038 Invalid control index

Cause

This error occurs when accessing the controls on a form by using **form.controls(index)** and the index value is invalid (<1 or > **form.controlCount**).

Action

Change your code accordingly.

Note **Controls** is a dynamic array. Deleting or adding entries from the form changes the **controlCount** and potentially the position of a control in this array.

14039 No conversion available for that picture type

Cause

This error occurs if your application code attempts to convert a non-standard picture type. Only bitmaps, TIFF (tagged image file format), PNG (portable network graphics), and JPEG (Joint Photographic Experts Group) type picture images can be converted to another picture type.

Action

Change your application code to ensure you have a picture of the required type before calling the [convertPicture](#) or the [convertToFile](#) method of the [Binary](#) primitive type.

14040 Conversion to that Picture type is not available

Cause

This error occurs if your application code requests a non-standard picture conversion. The type of picture conversion requested by the [convertPicture](#) or [convertToFile](#) method call of the [Binary](#) primitive type is not available. JADE supports picture conversion to bitmap, TIFF (tagged image file format), PNG (portable network graphics), or JPEG (Joint Photographic Experts Group) types only.

Action

Change the method parameter to use a valid conversion picture type.

14041 Decimals not valid for alpha data type

Cause

This error occurs if your application code attempts to set the [decimals](#) property of a text box control to a non-zero value when the [dataType](#) property is set to alphanumeric.

Action

You must first set the [dataType](#) property to indicate that the text box will accept only numeric primitive type data and then set the [decimals](#) property.

14042 Decimals must be 0 through 17

Cause

This error occurs if the [decimals](#) property of a text box control is set outside the valid range **0** through **17**.

Action

Change your application code so that a valid [decimals](#) property value is supplied.

14043 Decimals must be less than maxLength

Cause

This error occurs if the number of decimal places allowed in a text box is greater than or equal to the maximum length of the text that can be entered. If this is allowed, you will not be able to enter any digits before the decimal place, and you will not be able to enter the entire decimal.

Action

Ensure that the [maxLength](#) property exceeds the number of decimal places allowed and will allow the correct number of digits to be entered before the decimal place.

14044 Collection must be of an object type membership

Cause

This error occurs if your application code attempts to attach a collection to a table and the collection is not an object type collection.

Action

Change your application code accordingly.

14045 Invalid or invisible sheet requested

Cause

This error occurs for a folder or table control if an attempt is made to access a sheet that does not exist.

For a table, the error is also returned if an attempt is made to delete all sheets of the table.

Action

Change your application code accordingly.

14046 This action is not valid from within the displayRow method

Cause

This error occurs if you attempt one of the following actions from within the [displayRow](#) event.

- Delete the sheet
- Clear the use of the displayed collection
- Attach a collection to the sheet

Action

Change your application code accordingly.

14047 Property unavailable for this combo style type

Cause

This error occurs if the text selection properties [selText](#), [selStart](#), or [selLength](#) are accessed for a combo box style that does not support text entry (drop-down list and spin box styles).

Action

Change the style of combo box or remove the code.

14048 Invalid number of columns set for Table

Cause

This error occurs if an attempt is made to set the number of columns of a table (the **columns** property) outside the range **0** through **16,000**.

Action

Change your application code accordingly.

14049 Invalid number of rows set for Table

Cause

This error occurs if an attempt is made to set the number of rows of a table (the **rows** property) outside the range **0** through **32,000**; for example, by inserting one or more rows in a table that already contains 32,000 rows.

Action

Change your application code accordingly.

14050 Invalid Table column

Cause

This error occurs if an attempt is made to set the value of the current column for the current sheet outside the range **1** through the value of the **columns** property.

This can occur when using the **accessColumn** method to set the sorting for a column before the value of the **sortOrder** property for the **JadeTableColumn** object has been set. JADE cannot access a table column to set its sort attribute before the **sortOrder** property of that column has been set.

Action

Change your application code accordingly.

When using **accessColumn** to set the sorting for a column, the **sortOrder** value must be set before the other sort attributes (such as, **sortAsc**, **sortCased**, **sortType**) for that column.

14051 Invalid Table row

Cause

This error occurs when:

- The value of the current row for the current sheet is set to a value outside the range **1** through the value of the **rows** property.

```
table1.row := 11;           // Exception raised if table has fewer than 11 rows
```

- A **JadeTableRow** reference is set using the **accessRow** method with an invalid row parameter.

```
table1.accessRow(11);       // Exception raised if table has fewer than 11 rows
```

- A **JadeTableRow** reference becomes invalid because rows are removed from a table.

```
table1.accessRow(5);    // Assume table has more than 5 rows, so no exception
table1.rows := 3;       // Reduce the number of rows in the table
table1.accessedRow.backColor := Red; // Exception as row is no longer valid
```

The error can also occur when accessing a cell using an invalid row value.

Action

Change your application code accordingly.

14053 The index for the Table Sort attributes must be 1 through 6

Cause

This error occurs if the index to the table sort properties is not in the range 1 through 6. Up to six table sort properties can be specified for each sheet.

Action

Change your application code accordingly.

14056 Cannot delete the only sheet of a table

Cause

This error occurs if an attempt is made to delete the only sheet of a table. A table control must always have at least one sheet.

Action

Change your application code to prevent this.

14058 Cell Text does not match an entry in the cell combo list

Cause

This error occurs if a cell of a table has an **inputType** property of **ComboBox** and the cell text is set to a value that does not match any of the entries in the combo box. Setting an **inputType** of **ComboBox** means that the cell text can only be one of the entries in the combo box list. An entry is set by using the **comboBoxIndex** property or by the cell **text** property.

Action

Change your application code to ensure that the text matches an entry in the combo box.

14059 Cell Combo index invalid for cell combo list

Cause

This error occurs when an index `comboIndex` property value is set for a cell of a table that has an `inputType` property of `ComboBox` and that index value is invalid. Setting the `comboIndex` value for the cell selects the specified entry in the combo box, so the index must be in the range 1 through the number of entries in the combo box.

Action

Check the text that is loaded into the combo box (entries separated by tabs), and check that the correct index value is specified.

14060 Table has no current cell (row, column)

Cause

This error occurs if a table control does not have a current cell for the current sheet; that is, one of the row or column property values is invalid, or the sheet has no rows or no columns.

Action

Ensure that the sheet being accessed has `rows` and `columns` and that the correct values of the `row` and `column` properties are set.

14061 Table must have at least one sheet visible

Cause

This error occurs if an attempt is made to hide the only visible sheet on a table control.

Action

Change your application code to ensure that it does not attempt to hide the last visible sheet on a table control.

14062 That sheet of the Table is hidden (cannot be top sheet)

Cause

This error occurs if an attempt is made to set the `topSheet` property of a table to a sheet that is not visible.

Action

Change your application code to prevent this.

14063 Window create failed

Cause

This error occurs if the physical creation of a window fails.

The Windows Graphic Device Interface (GDI) resources have been exhausted for the system. When JADE attempts to add another control or form, Windows fails because there are already too many controls in memory.

Action

While your application may not create a large number of controls, Windows resources are shared by all applications. It may be that another application or report program is consuming resources.

Check the log file for messages reporting the number of applications running and the total number of forms and controls for all applications running in this node, similar to the following:

```
2006/06/29 14:55:53.835 00e60-0834 JadeExe: Error = 14063 : Window create failed
2006/06/29 14:55:53.895 00e60-0834 JadeExe: Failing Window: Jade:Menu Name: Jade
Popup Menu
2006/06/29 14:55:53.945 00e60-0834 JadeExe: Window's error: Not enough storage is
available to process this command. (8)
2006/06/29 14:55:54.006 00e60-0834 JadeExe: Total of 5 Forms & 1055 controls active
in 2 Jade applications
```

GDI is comprised of one 16-bit heap (64K bytes in size) and one 32-bit heap (2M bytes in size). GDI uses its heaps to store fonts, brushes, palettes, bitmaps, and pens (graphical objects). The GDI manages what appears on the screen. It provides graphic support for printers and other output devices, draws graphic primitives, manipulates bitmaps, and interacts with device-independent graphics drivers, including those for display and printer output device drivers.

Shutting down the machine and restarting purges the GDI memory.

You can increase the heap size by editing the Registry.

Caution If you use Registry Editor incorrectly, you may cause serious problems that can require you to reinstall your operating system. Jade Software Corporation cannot guarantee that you can solve problems that result from using Registry Editor incorrectly. Use Registry Editor at your own risk.

To increase the size of the desktop heap:

1. Run Registry Editor (**REGEDT32.EXE**).
2. From the **HKEY_LOCAL_MACHINE** subtree, go to the following key.

```
\System\CurrentControlSet\Control\Session Manager\SubSystems
```
3. Select the **Windows** value.
4. From the Edit menu, select the **String** command.
5. Increase the **SharedSection** parameter.
6. **SharedSection** specifies the system and desktop heaps, using the following format:

```
SharedSection=xxxx,yyyy
```
7. In this format, the **xxxx** value defines the maximum size of the system-wide heap (in kilobytes) and the **yyyy** value defines the size of the heap for each desktop. The default value of the heap for each desktop under Windows NT 3.5 (512K bytes) can support approximately 2,500 windows. Increasing the desktop heap by 256K bytes or 512K bytes normally provides enough memory to correct *Out of Memory* error messages.

Some programs that are more GDI resource-intensive are:

- Graphics applications
- Multimedia utilities
- Internet tools and applications.

Closing down some of the other applications running on the machine may free sufficient resources to complete the operation.

If the problem occurs while printing a report, consider the following options.

- Create one print form only and reuse this for each page of the report. This generates only one instance of the report and its associated controls. Create a reference to the form and use it for each page.
- Use the [createPrintForm](#) method to reduce the amount of GUI windows resources that are used.

Note that if you are explicitly deleting forms after printing, while JADE flags these instances as deleted they still exist and consume Windows resources until JADE encounters a quietpoint in processing in which it tidies up any remaining classes and controls.

If this does not satisfactorily resolve the situation, contact your local JADE support center or JADE Support if your JADE licenses include support. They will assist you in ascertaining what other applications running on the same server could have utilized a large number of GUI resources.

14064 Command ignored while in paint event

Cause

This error occurs if the paint code of a control or form attempts to use the [refresh](#), [refreshNow](#), or [clearGraphics](#) method. These methods cause another paint to be generated that would then cause another paint that would again call these methods, and so on. This would cause a never-ending paint situation that would probably also cause a stack overflow. As the next paint of your control or form will generate another error, the application will probably need to be terminated and then restarted before the code can be corrected.

Action

Remove the calls to these methods for paint events.

14065 ShowModal called for a form already in showModal

Cause

This error occurs if the [showModal](#) method is called for a form that is already displayed as a modal form.

Action

Change your application code to prevent this.

14066 Form is not an MDI frame

Cause

This error occurs when the [moveMdiClient](#) method is called for a form that is not an MDI frame.

Action

Change your application code accordingly.

14067 First character must be alphabetic (see TextBox case property)

Cause

This error occurs if you have not specified an alphabetic character for the first character of the case property of the [TextBox](#) control.

Action

Change your application code accordingly.

14068 Property is read only

Cause

This error occurs if an attempt is made by logic to change the value of a read-only property.

Action

Change your application code to prevent this.

14069 Control with that index has already been loaded

Cause

This error occurs if the [loadControl](#) method is called with an index value for a control copy that has been assigned by a previous [loadControl](#) call.

Action

Change your application code to prevent this.

14070 Menu item passed to popupMenu does not have submenu items

Cause

This error occurs when the [popupMenu](#) form method is called for a menu item that is not a popup menu. The [popupMenu](#) method can be called only for a menu with submenu items.

Action

Change your application code accordingly.

14071 Creation or operation of the popup menu failed

Cause

This error occurs if the windows call **TrackPopupMenu** fails.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

14072 This action is not valid when a collection is attached to a Control

Cause

This error occurs when a collection is attached to a list box, combo box, or table and code attempts to add or remove an entry from the control or to set the **itemObject** value of an entry.

When a collection is associated with a sheet of a control

The **addItem**, **addItemAt**, **removeItem** methods are not available for a list box, combo box, or table. In addition, the **resort**, **moveRow**, and **deleteRow** methods are not available for a table.

Action

When you attach a collection, you cannot add non-collection entries and you cannot remove existing entries. The value of the **itemObject** property for an item is the attached collection object, and therefore it cannot be changed. Change your application code accordingly.

14073 Only transient controls can be added to a form

Cause

This error occurs if the **addControl** method is called for a control that is not a transient object. Runtime controls must always be transient.

Action

Ensure that the control object is transient.

14074 Text of numeric TextBox or JadeEditMask not valid numeric

Cause

This error occurs for a **TextBox** or **JadeEditMask** control under one of the following conditions.

- The **dataType** property of the text box is a numeric primitive type and an attempt is made to set the text to a non-numeric value.
- An attempt is made to set the **dataType** property of the text box to numeric and the existing text is non-numeric.

- An attempt is made to set the **decimals** or **maxLength** property that would leave the text invalid according to the new rules.
- The **mask** property of a **JadeEditMask** specifies numeric data and an attempt is made to set the text with non-numeric data.
- An attempt is made to change the **mask** property of a **JadeEditMask** control that would leave the text invalid according to the new rules.

Action

Change your application code to ensure that the text value is valid according to the implied validation rules for the properties being set.

14075 Too many digits in text of numeric TextBox or JadeEditMask

Cause

This error occurs for a **TextBox** or **JadeEditMask** control under one of the following conditions.

- The **dataType** property of the text box is a numeric primitive type and an attempt is made to set the text with too many leading digits.
- An attempt is made to set the **dataType** property to numeric and the existing text has too many leading digits.
- An attempt is made to set the **decimals** or **maxLength** property that would leave the text invalid according to the new rules.
- The **mask** property of a **JadeEditMask** specifies numeric data and an attempt is made to set the text with too many leading digits.
- An attempt is made to change the **mask** property of a **JadeEditMask** control that would leave the text invalid according to the new rules.

Action

Change your application code to ensure that the text value is valid according to the implied validation rules for the properties being set.

14076 Too many decimals in text of numeric TextBox or JadeEditMask

Cause

This error occurs for a **TextBox** or **JadeEditMask** control under one of the following conditions.

- The **dataType** property of the text box is a numeric primitive type and an attempt is made to set the text to a value with too many decimal places.
- An attempt is made to set the **dataType** property to numeric and the existing text has too many decimal places.
- An attempt is made to set the **decimals** or **maxLength** property that would leave the text invalid according to the new rules.

- The **mask** property of a **JadeEditMask** specifies numeric data and an attempt is made to set the text with too many decimal places.
- An attempt is made to change the **mask** property of a **JadeEditMask** control that would leave the text invalid according to the new rules.

Action

Change your application code to ensure that the text value is valid according to the implied validation rules for the properties being set.

14077 A signed numeric is not allowed in the text for this TextBox

Cause

This error occurs for a **TextBox** control under one of the following conditions.

- The **dataType** property of the text box is a numeric primitive type and an attempt is made to set the text to a value that includes a negative sign.
- An attempt is made to set the **dataType** property to numeric and the existing text has a negative sign.
- An attempt is made to set the **decimals** or **maxLength** property that would leave the text invalid according to the new rules.

Action

Change your application code to ensure that the text value is valid according to the implied validation rules for the properties being set.

14078 A gui action has been requested in a non-gui environment

Cause

This error occurs if the application is not a GUI application.

Action

Change the logic in your application.

14079 The text size exceeds maxLength

Cause

This error occurs if your code is attempting to set the **text** value of a **TextBox** or **JadeRichText** control to a string that is longer than the value of the **maxLength** property.

Action

Change your application code to prevent this, or change the allowable **maxLength** property value.

14080 There is no object in the Ole Control

Cause

This error occurs if OLE-type methods are called on an **OleControl** class that does not have an **oleObject** property loaded.

Action

Ensure that the **OleControl** has an object loaded using the **objectType** method before calling the method.

14081 The ActiveX control has not released all its Ole object references

Cause

This error occurs if the **ActiveXControl** is still holding references to interface objects used in communicating to the ActiveX container supplied by JADE. When the ActiveX control obtains any reference to these objects, a reference count is incremented for each object.

The ActiveX control must call *release* on these objects when it has finished with the object. Failure to do so will cause leakage of memory and resources and can also result in the JADE application crashing.

Action

The creator of the ActiveX control needs to ensure that all interface objects used are released.

14082 The oleObject property for a running OleControl window cannot be changed

Cause

This error occurs if your application code attempts to change the **oleObject** property of a running **OleControl** window. When an **OleControl** control has an associated window, it creates a transient **OleObject** object that is assigned to the **oleObject** property and used during the running life of its Window. This object cannot be changed.

Action

You can use the contents of another **OleObject** instance in the control by using the copy method of the **OleObject** class. For example:

```
oleControl1.oleObject.copy(oleObj);
```

Note To cause that data to take effect, you must also call the **OleControl** class **loadFromDB** method.

14085 Not a valid Ole Class

Cause

This error occurs if an attempt is made to load an OLE object using the **OleObject** class name and that OLE class is unknown on this workstation.

Action

Ensure that the **OleObject** class name is valid and that the **OleObject** class is registered on this workstation.

14086 File not found

Cause

This error occurs if an attempt is made to embed or link to a file in an **OleControl** and that file is not available.

Action

Ensure that the file is available.

14088 The create of the Ole object failed

Cause

This error occurs if the creation of an OLE object fails. A possible cause is that the **ActiveXControl** is not registered on your PC.

Action

Ensure that the OLE object being created is valid. The **errorItem** in the Exception dialog identifies the control that caused the exception.

On Windows 10 64-bit operating systems, some older ActiveX controls no longer register. If this happens, you may need to replace the control with a newer version or with another library.

14089 Ole object verb is unknown

Cause

This error occurs if an attempt is made to use the **applyVerb** method to invoke an unknown verb for the OLE object loaded into the **OleControl**.

Action

Ensure that the **applyVerb** method is called with a valid OLE verb for that object.

14090 File not of recognized Ole type

Cause

This error occurs if an attempt is made to embed an object from a file that is not valid or a known OLE object type.

Action

Ensure that a correct OLE object file is embedded and that the OLE server is registered on this workstation.

14091 Server execution failed

Cause

This error occurs if the creation of an OLE object fails because the OLE server execution has failed. This is usually due to the fact that the OLE server is registered but the executable file cannot be found.

Action

Ensure that the required OLE server executable is valid and present.

14092 MultiMedia function failed.

Cause

This error occurs if you attempt to apply a multimedia action to a device that does not support multimedia. The error message also includes text from the multimedia device that further explains the cause of the error.

Action

Change your application code accordingly.

14093 A menu with that index has already been loaded

Cause

This error occurs when the [loadMenu](#) method is called to add a new menu item and a previous call to the [loadMenu](#) method already created a menu with that index.

Action

Ensure that the index value of the menu being created has not already been assigned.

14094 That menu item not found

Cause

This error occurs if the [getMenuItem](#) method is called for a menu item that does not exist, or if the [loadMenu](#) method is called from a base menu item that must have been deleted.

Action

Ensure that the index to the [getMenuItem](#) method is valid and that the [loadMenu](#) method is not performed on a deleted menu.

14095 Invalid scroll data: invalid scrollBar or scroll range

Cause

This error occurs if the calls to the [getScrollRange](#) method or the [setScrollRange](#) method specify an invalid scroll bar type ([SCROLL_HORZ](#) or [SCROLL_VERT](#)), if the minimum scroll range is greater than the maximum scroll range, or if the small change value is greater than the large change value.

Action

Ensure that the parameters to the **getScrollRange** and the **setScrollRange** methods are valid.

14096 Class of Ole Object not registered

Cause

This error occurs if the OLE class requested for an **OleControl** control or an **ActiveXControl** control is not registered on this workstation.

Action

Ensure that the **OLE** class or **ActiveX** control is registered on this workstation.

14097 User Security does not allow access to that form

Cause

This error occurs if the **show** or **showModal** method is called on a form that has the **app.userSecurityLevel** value greater than the value of the **securityLevelVisible** property of the **Window** class or **MenuItem** class.

Action

Check the security level of the user before allowing the form to be created or displayed.

14098 Property or method name not valid for ActiveX Control

Cause

This error occurs for an **ActiveXControl** control when a method or property previously defined for the **ActiveX** control is no longer available.

Action

Ensure that the correct version of the **ActiveX** control is loaded on this workstation.

14099 Unknown event name

Cause

This error occurs if the call to **setEventMapping** requests an event mapping for an event that is not defined.

Action

Ensure that the event name is defined for the control.

14100 Method does not exist in call to setEventMapping

Cause

This error occurs if the call to [setEventMapping](#) requests an event mapping for an event for which there is no default method defined.

Action

Ensure that the default event method is defined.

14101 Method signature does not match event method in call to setEventMapping

Cause

This error occurs when a call to [setEventMapping](#) requests an event to be mapped to a method that has a method signature that does not match that of the default event method.

Action

Ensure that the method signature of the requested event method matches that of the default event method.

14102 Creating storage for the Ole object failed

Cause

This error occurs if the creation of the temporary storage area used by the **OLE** object or [ActiveXControl](#) control failed. The error could also arise with a [JadeRichText](#) control, as it is possible for this control to contain an **OLE** object or **ActiveXControl** control.

The error text includes the Windows error, which gives more details about the failure.

Action

Perform one or more of the following actions.

- Check whether the disk is full.
- Check the version of the **OLE** server or **ActiveXControl**.
- Check the Windows error information.

14103 Could not get IViewObject2 interface from the Ole Object

Cause

This error occurs if JADE attempts to use the **IViewObject2** interface and the interface is not supported.

JADE calls the OLE object or OCX Windows Component Object Model interface **IViewObject2**, to paint the OLE object image. If this interface is not supported, it cannot be used in JADE.

Action

Report this error to the developer of the OLE server or [ActiveXControl](#) control that is being used.

14104 The Ole object failed to draw itself

Cause

This error occurs if the **IViewObject2** interface fails.

JADE calls the **OLE** object or OCX Windows Component Object Model interface **IViewObject2**, to paint the **OLE** object image. If this interface fails, it is unlikely that the **OLE** object image is available.

Action

Check the Windows error information. If this does not solve the problem, report this error to the developer of the OLE server or ActiveX control that is being used.

14105 Could not get IPersistStorage interface from the Ole object

Cause

This error occurs if the **IPersistStorage** interface is not available.

JADE calls the OLE object or OCX Windows Component Object Model interface **IPersistStorage**, to save data for the OLE object. If this interface is not available, no permanent image of the object is saved in the JADE database.

Action

Report this error to the developer of the **OLE** server or [ActiveXControl](#) control that is being used.

14106 Saving the Ole Object to persistent storage failed

Cause

This error occurs if the **IPersistStorage** interface fails.

JADE calls the **OLE** object or OCX Windows Component Object Model interface **IPersistStorage**, to save data for the **OLE** object.

The error text includes the Windows error, which gives more details about the failure.

Action

Check the Windows error information. If this does not solve the problem, report this error to the developer of the **OLE** server or [ActiveXControl](#) control that is being used.

14107 The attempt to activate the Ole object failed

Cause

This error occurs if the **OLE** object or **ActiveXControl** control user interface fails. The attempt to activate the OLE object or **ActiveXControl** control user interface failed. The error text includes the Windows error, which gives more details about the failure.

Action

Check the Windows error information. If this does not solve the problem, report this error to the developer of the **OLE** server or **ActiveXControl** control that is being used.

14108 Ole method call or property access failed

Cause

This error occurs when access to an ActiveX control property or method fails.

The failure is caused by invalid data being passed to the **ActiveXControl** control (for example, an unacceptable value), or by the wrong version of the **ActiveX** control being used where the property or method definition differs from the JADE database definition created when the **ActiveX** control was loaded.

The error text includes the Windows error, which gives more details about the failure.

Action

Check the Windows error information.

14109 Unrecognised command line argument

Cause

This error occurs if an unrecognized argument is detected in the command line.

Action

Check your command line and then correct the invalid argument. Use the following format.

```
app=name path=name server=name
```

14110 Application to be executed must be included on the command line

Cause

This error occurs if the application name is not included in the command line. When JADE is initiated, it requires the name of the application to be run.

Action

Change the command line definition to include the application argument, as follows.

```
app = name
```

For example:

```
app = JADE
```

14111 Database path to be used must be included on the command line

Cause

This error occurs if the database path is not included in the command line. The command line to initiate JADE must include the database path.

Action

Change the command line definition to include the **path** argument, as follows.

```
path = path
```

For example:

```
path = C:\JADE\DB
```

14112 Control in form controlList is not a subclass of Control

Cause

This error occurs if an invalid entry is found in the control list of the form that is being initiated. The entry is empty, or it is not a control object.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

14113 Windows appears to be in a recursive loop

Cause

This error occurs if Windows messages are being received by JADE before previous messages have been dealt with. This situation has reached a critical stage at which JADE will soon generate a runtime fault, due to the overflow of the runtime stack.

This situation is usually caused by inappropriate use of [app.doWindowEvents](#) or wait code. For example, if the user holds down a key and the [keyDown](#) event code makes an [app.doWindowEvents](#) call, another event is processed, which then does another [app.doWindowEvents](#) call and so on, until the JADE stack overflows.

Action

Change your code to ensure that a recursive loop does not develop.

14114 Incompatible Ole library version

Cause

This error occurs if the OLE 2 version does not match the OLE 2 version of JADE. This means that OLE objects and controls cannot be used.

Action

Ensure that the correct versions of Windows and OLE 2 are loaded.

14115 Ole initialize failed

Cause

This error occurs if the initialization of OLE 2 fails. This means that **OLE** objects and **ActiveXControl** controls cannot be used.

Action

Check the Windows error, and ensure that you have the correct OLE2 DLLs loaded.

14116 Failed to Create Class

Cause

This error occurs if the creation of the Windows internal GUI class fails.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

14117 Method must be reimplemented in subclass

Cause

This error occurs if you have not reimplemented this method in your subclass.

Action

Change your application code to reimplement this method in your subclass.

14118 The start of a second application by a thin client did not end up on the correct application server

Cause

This error occurs when a presentation client has initiated a second application and the TCP connection was made to an application server that differs from the one running the first application. (The applications check that the presentation client has connected to the correct application server.)

Action

The network configuration is not set up correctly for JADE operation. When the first connection succeeds, the application server's IP address is queried. This IP address is subsequently used to ensure the correct application server is reached if another application is initiated. If the use of this IP address did not achieve the expected result and another application server was reached instead, the network configuration should be adjusted accordingly.

14119 A Thin Client reconnection attempt ended up on the wrong application server

Cause

This error occurs when a presentation client connection is lost and an attempted reconnection to the original application server reached a different application server instead. (The applications check that the presentation client has connected to the correct application server.)

Action

If the application server with active users is closed and then quickly restarted, this error is normal (that is, the reconnection found the new application server).

If the application server has not been closed and this error occurs, the network configuration is not set up correctly for JADE operation. When the first connection succeeds, the application server's IP address is queried. This IP address is subsequently used to ensure the correct application server is reached if the connection is lost and an attempted reconnection is initiated. If the use of this IP address did not achieve the expected result and another application server was reached instead, the network configuration should be adjusted accordingly.

14122 Failed to convert picture to file.

Cause

This error occurs when the requested conversion of a picture to another picture format fails.

Action

The additional error description displayed with this error provides the reason for the failure.

14123 Operation is invalid for server execution

Cause

This error occurs because the requested operation is not available in a server method.

Action

Change your application code accordingly.

14124 numberOfColourBits parameter must be 1,4,8 or 24

Cause

This error occurs when calling the [createPicture](#) method if you have not correctly specified the number of color bits that apply to the created picture.

Action

Correct your application code to specify the correct **numberOfColorBits** parameter in the method call.

14125 OLE error

Cause

This error occurs if the creation of an [ActiveXControl](#) control fails. The exact reason cannot be determined from the **ActiveX** control and this error has been returned.

Action

Report the error to the developer of the **ActiveX** control.

14126 Create of OLE picture failed

Cause

This error occurs if passing a JADE picture to a Windows **OLE** picture fails.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

14127 Error accessing ActiveX property page information

Cause

This error occurs if the JADE initiation of or [ActiveXControl](#) control property page fails.

Action

Report the error to the developer of the [ActiveXControl](#) control.

14128 Error accessing ActiveX type information

Cause

This error occurs if the JADE query of [ActiveXControl](#) control type information fails.

Action

Report the error to the developer of the ActiveX control.

14129 Insert OLE object failed

Cause

This error occurs if the [ActiveXControl](#) control **create** operation fails.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

14130 A Tcp Port number must be included on the command line

Cause

This error occurs if a TCP/IP port number of the host client is not included on the command line when running the JADE thin client.

Action

Change the command line entries accordingly.

14131 The path and server commands are not valid for thin client execution

Cause

This error occurs if you have tried to include a path or server option on your command line when running JADE thin client. These are specified by the application server to which the presentation client is connecting.

Action

Change the command line arguments accordingly.

14132 That control is already attached to a form

Cause

This error occurs if your application code attempts to connect a control with a physical window when it is already associated with another physical window; for example, if you attempt to perform a **form.addControl** call more than once.

Action

Change your application code to prevent this.

14133 Referencing a list item that has not yet been assigned

Cause

This error occurs if your application code attempts to reference an entry in a list box or combo box with an attached **listCollection** that is logically correct but the entry is not yet assigned. For example, a list box has only entries **1** through **10** attached so far, but code refers to list entry **11**, which has not yet been generated by a **displayEntry** event.

Action

Change your application code to prevent this.

14134 Text height calculation failed - your fonts may be corrupted

Cause

This error occurs if a standard Windows API for calculating the height of text fails. Previous investigation of this fault has shown that Windows internal font tables have been corrupted.

Action

Restart your workstation and try again.

14135 App Server stack overflow

Cause

This error occurs when a recursive situation between the presentation client and the application server has resulted in the overflow of the message stack of the application server.

Action

Change your code to avoid the recursive situation.

14136 App Server and Jade exe versions differ

Cause

This error occurs if the JADE software versions of the presentation client and the application server are incompatible.

Action

Install the correct JADE version.

14137 The App Server and Jade exe are not Unicode compatible

Cause

This error occurs if the presentation client and the JADE software on the application server are not Unicode compatible.

Action

Ensure that the correct JADE software is installed.

14138 Invalid AppServer port number

Cause

This error occurs if the TCP port number is invalid.

Action

Use a port number in the range **0** through **65535**.

14139 The execution of that application is not available from the connected Application Server

Cause

This error occurs if the application server has been initialized to permit only a restricted list of schemas and applications to be executed using thin client and the requested schema and application is not on that list.

Action

Check and confirm that you have specified allowed schema and application names. For more details, see the [AllowSchemaAndApp](#) parameter in the [[JadeAppServer](#)] section of the JADE initialization file, in the *JADE Initialization File Reference*.

14140 Thin Client Tcp connection failed

Cause

This error occurs if the TCP/IP layers have failed to open the requested thin client connections. It can also occur when a thin client fails to connect to the application server because a Secure Socket Layer (SSL) proxy server is not *automatically* detected.

Action

Perform the following checks.

- The **AppServer** name is correct and mapped to the correct Internet Protocol (IP) number in the host file
- The **AppServerPort** number matches the port number of the target application server

- The application server is running
- Whether a proxy server is in use

If a proxy server is in use, you could avoid automatic detection issues by setting the following values in the JADE initialization file.

```
[JadeThinClient]
SSLProxyDetect=false
SSLProxyHost=<proxy hostname>
SSLProxyPort=<proxy port number>
```

If the value of the **SSLProxyDetect** parameter is not set to **false**, the network proxy must be formally declared in the PC browser settings of the thin client. This step ensures proxy information is added to the registry on the thin client.

14141 Thin Client AppServer could not be located

Cause

This error occurs if the **AppServer** name was not available on your Domain Name Service (DNS) server.

Action

Specify the correct **AppServer** name.

14142 Thin Client Tcp request was cancelled

Cause

This error occurs if a network operation was interrupted.

Action

Check that the presentation client and the application are still running, and that your network connection is still available.

14143 Thin Client Tcp connection was closed

Cause

This error occurs if the thin client connection was closed by the presentation client or the application server while an operation was still in progress.

Action

Check that the presentation client and the application are still running, and that your network connection is still available.

14144 Thin Client Tcp connection read error

Cause

This error occurs if the thin client TCP/IP receiver encountered an error.

Action

Check the log file for a more-specific message or error number, and then contact your network administrator.

14145 Thin Client Tcp connection write error

Cause

This error occurs if the thin client TCP/IP sender encounters an error.

Action

Check the log file for a more-specific message or error number, and then contact your network administrator.

14146 Thin Client Tcp connection read operation is already pending

Cause

This error occurs if there is an attempt to perform a read operation when an existing read is already pending.

Action

Change your application code so that multiple reads cannot occur on the same network object.

14147 Thin Client Tcp connection is not open

Cause

This error occurs if there is an attempt to perform a read or write operation when the thin client connection is not open.

Action

If this message occurs unexpectedly, check your application and ensure that it does not close the network. If it is not a programmatic problem, check with your network administrator why the thin client connection is not open.

14148 Thin Client Tcp connection listen port error

Cause

This error occurs if the application server attempts to listen for a thin client connection on the nominated TCP port and the listen fails. The most-likely causes are that the port is already in use, or that there is a network problem.

Action

Try a different TCP port. If this does not help, ask your network administrator to check the network.

14149 Thin Client Tcp connection accept socket problem

Cause

This error occurs if the TCP listen process was terminated abnormally. The most-likely cause is that the port is already in use or there is a network problem.

Action

Try a different TCP port. If this does not help, ask your network administrator to check the network.

14150 QueryInterface failed

Cause

This error occurs if the interface you are requesting is not an interface of the object.

Action

Change your application code accordingly.

14151 Error saving IJadeAutoPicture

Cause

This error occurs if the automation picture object cannot be saved as the specified file.

Action

Check that the path is valid or that the disk is not full.

14152 Error loading Binary into JadeAutoPicture

Cause

This error occurs if an automation picture object cannot be created from the specified file.

Action

Check that the path is valid or that the disk is not full.

14153 Class must be a subclass of IDispatch

Cause

This error occurs if the class applied to the **class** parameter of the [getInterface](#) method is not a subclass of the **IDispatch** class.

Action

Change your application code accordingly.

14154 ActiveX object has already been created

Cause

This error occurs if there is an attempt to create an ActiveX object that already exists.

Action

Change your application code to prevent this.

14155 Invalid parameter

Cause

This error occurs if a parameter is found to be invalid.

Action

Change your application code to use a correct parameter. For details about parameter usage, see "[Parameters](#)", in Chapter 1 of the JADE *Developer's Reference*.

14156 Not an Automation Event class

Cause

This error occurs if a parameter is not of a class that corresponds to an automation event class.

Action

Change your application code accordingly.

14157 App Server had problems initializing the SSL library

Cause

This error occurs if the application server is unable to initialize the Secure Sockets Layer (SSL) library successfully.

Action

Check that the SSL library is present and valid.

14158 Thin Client had problems initializing the SSL library

Cause

This error occurs if the application server is unable to initialize the Secure Sockets Layer (SSL) library successfully.

Action

Check that the SSL library is present and valid.

14159 ActiveX meta data missing

Cause

This error occurs if information that describes the [ActiveXControl](#) object is missing.

This can result if a **.ddb** file for an **ActiveX** object was not loaded during the JADE schema extract or load process or **ActiveX** meta data does not exist when the [ActiveXAutomation](#) class [createAutomationObject](#) method is called.

Action

Check that your schema load or extract loaded the **.ddb** or **.ddx** file for the **ActiveX** object.

14160 ActiveX object must be created before its properties can be accessed

Cause

This error occurs if there is an attempt to access an [ActiveXControl](#) object before it has been created.

Action

Ensure you have created an **ActiveX** object before attempting to access it.

14161 Jade Dll version mismatch:

Cause

This error occurs if you attempt to run JADE with a mixture of JADE binary modules from different releases.

Action

Ensure that you have installed and are using the correct JADE binaries.

14162 The thin client download directory is not valid

Cause

This error occurs if the directory specified in the JADE initialization file on the presentation client into which downloaded or pre-downloaded files will be written is not valid.

Action

Specify a valid directory in the JADE initialization file on the presentation client.

14163 The thin client download failed because file to be downloaded from the application server could not be successfully accessed

Cause

This error occurs if a file to be downloaded from the application server cannot be accessed; for example, the file is not present, or access is denied by security restrictions.

Action

Refer to your JADE messages (**jommsg.log**) log file for more information, and correct the problem identified with the download file.

14164 The thin client download installation program Jaddinst.exe failed to execute

Cause

This error occurs if the download installation program (**jaddinst.exe**) is missing or did not initiate successfully.

Action

Install the correct **jaddinst.exe** program from the JADE release.

14165 The thin client software is currently being downloaded, please wait and try again shortly. The user performing the download is:

Cause

This error occurs if the shared thin client files that you are attempting to use are currently being updated by another user download. These binary files are not available until that download is complete.

Action

Try again later. How much later depends on the line speed of the network you are using for the thin client connection.

14166 Another user is performing a Jade version check. However, this process has taken over 30 seconds. Check that the file thinlock.fil in the binary directory is not blocking the signon process

Cause

This error occurs if the shared thin client files you are attempting to use are currently being updated by another user download. In this case, over 30 seconds have elapsed, and therefore it is likely that the download may have failed and left these files locked.

Action

Check whether another user is performing a download of the shared files you are attempting to use. If that is the case, try again later. If another user download is not taking place, remove the file **thinlock.fil** from the JADE binary directory and then try again.

14167 An application server download directory is not valid

Cause

This error occurs if you have specified an invalid download or pre-download directory in your JADE initialization file on the application server client.

Action

Check your JADE initialization file and make sure you have specified a valid directory.

14168 A new version of the thin client software must be downloaded to complete this request - unfortunately the maximum number of simultaneous downloads permitted by this installation has already been reached. Please try again shortly.

Cause

This error occurs if the thin client software you are attempting to use is out of date. A download cannot be issued at this time because the site-defined limit on simultaneous downloads in progress has already been reached by other users.

Action

Try again later.

14169 A new version of the thin client software must be downloaded to complete this request - proceed with the download?

Cause

This error occurs if the thin client software you are attempting to use is out of date. In this case, an automatic upgrade of this software will be performed unless you reject the request.

Action

Click the **Yes** button to proceed with the software upgrade or click the **No** button to reject the upgrade.

14170 There are download files available for a future release. Do you wish to perform the download now?

Cause

This message is displayed if a future release of the thin client software is available for pre-downloading. These files have been made available so that you can perform the download in advance and hence avoid a flood of download requests when that software goes live. The downloaded files are stored temporarily until required.

Action

Click the **Yes** button to perform the pre-download now. Alternatively, click the **No** button to defer the download until later.

This request will be repeated each time a thin client session is initiated until that pre-download is finally completed.

14171 A new version of the thin client software must be installed from files already downloaded before you can continue - proceed with the install?

Cause

This error occurs if the thin client software you are attempting to use is out of date. In this case, the new files required are already available on the thin client workstation (pre-downloaded). These files must now be installed before you can continue.

Action

Click the **Yes** button to accept the installation or click the **No** button to reject the installation.

14172 The thin client automatic download version check cannot create the lock file required for that process

Cause

This error occurs when using a file to create a single user function when another user is already accessing that file.

Action

Check whether there are other users holding the required file locked.

14173 The thin client automatic download cannot revert to this old Jade version

Cause

This error occurs if a presentation client attempts to connect to an application server that is running an older version of JADE and the automatic download process is not supported across those releases. From JADE 6.3 onwards, the automatic thin client download supports only reverting the presentation client to the same or the prior major release (that is JADE 6.3 can revert to JADE 6.2 but not to JADE 6.1).

Action

If you want the presentation client to revert to an earlier version of JADE, re-install the JADE thin client software for the required version, or perform the automatic download in stages by connecting to application servers running the prior major releases, one after the other. For example, to revert from JADE 6.3 to 6.1, connect the presentation client to a JADE 6.2 application server and then to a JADE 6.1 application server.

14174 This session has been terminated by a force off request

Cause

This error occurs if a user or administrator manually requests that your presentation client application is forcibly closed through a JADE Monitor request, or by the use of the `forceOffUser` method in the `System` class.

Action

Contact your administrator and then restart the application.

14175 The thin client automatic download cannot upgrade from this old Jade version

Cause

This error occurs if a presentation client attempts to connect to an application server that is running a newer version of JADE and the automatic download process is not supported across those releases.

From JADE 6.3 onwards, the automatic thin client download supports only upgrading the presentation client to the same or the next major release (that is, a JADE 6.2 presentation client can be upgraded from a JADE 6.3 application server but a JADE 6.1 presentation client cannot).

Action

If you want to upgrade the presentation client to an earlier version of JADE, reinstall the JADE thin client software for the required version, or perform the automatic download in stages by connecting to application servers running the next major releases, one after the other. For example, to upgrade from JADE 6.1 to 6.3, first connect the presentation client to a JADE 6.2 application server and then to a JADE 6.1 application server.

14176 Timed out waiting to send a management message to thin client

Cause

When a method is called on a **Process** object that performs an action on a thin client, a message is sent to the thin client. If the thin client is busy and cannot accept messages, a timeout can occur and the error is raised.

This error is more likely to occur if the method is executing for a **Process** object other than the current process. For example, the target process could be blocked because an Exception dialog, or other modal dialog, is being displayed.

Action

Avoid or delay calling methods on **Process** objects other than the current process. Avoid leaving thin clients processes in a modal state for extended periods of time. If errors occur frequently, contact your local JADE support center or JADE Support if your JADE licenses include support.

14177 Management message ignore because thin client in reconnect state

Cause

When a method is called on a **Process** object and that method needs to perform an action on a thin client, a message is sent to the thin client. If the thin client has had a network disconnect from the application server, an exception is raised.

Action

If the results of the method are not critical, you can trap the error and ignore it.

Alternatively, retry the operation later, as the thin client could have reconnected to the application server or the application server could have terminated the process because the thin client failed to reconnect within the configured time limit.

14178 The language in use on this pc is not supported on the application server you are connecting to. You must change the language in control panel.

Cause

This error occurs when a presentation client PC is set to a locale that is not installed on the application server. As a result, the application server refuses the connection because it cannot perform locale-aware handling for the presentation client.

Action

Change the locale set on the presentation client PC or install the required locale on the application server.

14179 Attempts to display the entries in a collection repeatedly failed - the iterator may not be valid

Cause

This error occurs when calling the **displayCollection** or **listCollection** method resulted in recursive errors.

Action

There is an issue with the collection being used or there is an issue with a user subclass of the **Iterator** class that is being used.

Investigate the collection (perform a schema validation) and if the **Iterator** class is subclass, check the logic.

14180 A print form cannot be shown

Cause

This error occurs when an attempt is made to **show** a non-GUI print form or to change its **visible** property in GUI mode. A non-GUI print form cannot be displayed in GUI mode.

Action

Change your application code so that the required form is not created as a print form.

14181 ActiveX invoke caused an exception

Cause

This error occurs when an **ActiveXControl** object raises an exception error during a property access or a method invocation.

This error has occurred within the **ActiveX** object itself, and the details of the error include the error number generated and passed by the **ActiveX** object.

Action

Refer to the **ActiveX** documentation. You may have to modify the call sequence or change the values passed as parameters to the **ActiveX** object.

14182 A non-gui app cannot create a form

Cause

This error occurs when a non-GUI client attempts to handle a form.

Action

Make the application GUI or GUI, no forms.

14183 The collection attached to a control via displayCollection or listCollection has been deleted

Cause

This error occurs when the [displayCollection](#) or [listCollection](#) method has been called on a control without update notification and the view has subsequently been scrolled but the collection has been found to be deleted.

Action

When the collection is deleted, the control using the collection must be cleared from being associated with the collection.

14184 A virtual collection cannot be used with displayCollection

Cause

This error occurs when the [displayCollection](#) method has been called attempting to assign a virtual collection to a table or list box, which is not allowed for this method.

Action

For a [ListBox](#) or [ComboBox](#), the [listCollection](#) method can be used instead. For a [Table](#), you must manually add entries to the table.

14185 The requested media name is not valid

Cause

This error occurs if logic has requested that a multimedia control load the contents of a file, and the file cannot be located or accessed.

Action

Change your logic to avoid this error.

14186 The requested media is not a supported type

Cause

This error occurs if logic has requested that a multimedia control load the contents of a file and that file is not a valid multimedia file, or the media type is not supported.

Action

Supply a file containing a supported media format.

14187 The requested action is only available in a GUI application and form

Cause

This error occurs when the **animateWindow** method has been called for a non-GUI application or a form that is not a JADE form.

The **animateWindow** method is available only in GUI applications and for JADE forms (that is, it is not available for Web forms and Web applications).

Action

Change your logic.

14190 Invalid day of the month for a JadeEditMask control

Cause

This error occurs if your application code attempts to set the text of a **JadeEditMask** control so that the characters of the text corresponding to the day part of the mask are invalid; for example, a **mask** value of **dd** and text of **a1**.

Action

Change your application code to prevent this.

14191 Invalid month for a JadeEditMask control

Cause

This error occurs if your application code attempts to set the text of a **JadeEditMask** control so that the characters of the text corresponding to the month part of the mask are invalid; for example, a **mask** value of **MMM** and text of **Jum**.

Action

Change your application code to prevent this.

14192 Invalid year for a JadeEditMask control

Cause

This error occurs if your application code attempts to set the text of a **JadeEditMask** control so that the characters of the text corresponding to the year part of the mask are invalid; for example, a **mask** value of **yyyy** and text of **12a1**.

In addition, note that years in the **JadeEditMask** control must consist of four numeric values for a **mask** of **yyyy** and two numeric values for a **mask** of **yy** to be considered valid.

Action

Change your application code to prevent this.

14193 The text of JadeEditMask control has too much data

Cause

This error occurs if your application code attempts to set the text of a **JadeEditMask** control so that there are more characters in the text than are required by the mask; for example, a **mask** value of **999** and text of **1234**.

Action

Change your application code to prevent this.

14194 Invalid numeric data in a JadeEditMask control

Cause

This error occurs if your application code attempts to set the text of a **JadeEditMask** control so that a character of the text corresponding to a numeric part of the mask is invalid; for example, a **mask** value of **999** and text of **a12**.

Action

Change your application code to prevent this.

14195 Invalid second or hour value for a JadeEditMask control

Cause

This error occurs if your application code attempts to set the text of a **JadeEditMask** control so that the characters of the text corresponding to the second or hour part of the mask are invalid; for example, a **mask** value of **ss** and text of **61**.

Action

Change your application code to prevent this.

14196 Invalid hour value for a JadeEditMask control

Cause

This error occurs if your application code attempts to set the text of a **JadeEditMask** control so that the characters of the text corresponding to the hour part of the mask are invalid; for example, a **mask** value of **hh** and text of **25**.

Action

Change your application code to prevent this.

14197 A text character is not a valid member of the list of valid characters for a JadeEditMask control

Cause

This error occurs if your application code attempts to set the text of a **JadeEditMask** control so that a character of the text corresponding to an alpha or alphanumeric part of the mask is not a member of the list of valid characters supplied in the mask; for example, a **mask** value of **a|abc|** and text of **z**.

Action

Change your application code to prevent this.

14198 A JadeEditMask control mask directive is missing the numeric part of the definition

Cause

This error occurs if your application code is attempting to set the **mask** property of a **JadeEditMask** control such as a field width and the numeric width is missing; for example, a **mask** value of **Waa** (which should be **W32aa**).

Action

Change your application code to prevent this.

14199 The JadeEditMask control 'Or' list missing terminating '|' character or is empty

Cause

This error occurs if your application code is attempting to set the **mask** property of a **JadeEditMask** control so that the end vertical bar character (|) of a list of valid characters for an alpha or alphanumeric part of the mask is missing, or there are no members of the list; for example, a **mask** value of **a|abc**.

Action

Change your application code to prevent this.

14200 A JadeEditMask control mask does not have any definitions for a field

Cause

This error occurs if your application code attempts to set the **mask** property of a **JadeEditMask** control so that there are no definitions for a field. This is usually caused by having two @ field termination characters consecutively or by having a @ character at the end of the mask. The @ character terminates a field within the control and starts a new label or text box portion.

Action

Change your application code to prevent this.

14201 Invalid alphabetic character

Cause

This error occurs if your application code attempts to set the text of a **JadeEditMask** control so that a character of the text corresponding to an alpha in the mask is not a valid alpha character; for example, a **mask** value of **a** and text of **.**.

Action

Change your application code to prevent this.

14202 Invalid literal in a JadeEditMask control

Cause

This error occurs if your application code attempts to set the **mask** property of a **JadeEditMask** control so that the end quote of a literal string is missing; for example, a **mask** value of **dd"/**.

Action

Change your application code to prevent this.

14203 Invalid range defined in the mask for a JadeEditMask control

Cause

This error occurs if your application code attempts to set an invalid range for a numeric field in the mask of a **JadeEditMask** control; for example, a **mask** value **99[12-]**. A range must follow an integer mask definition and be of the form [*start-range*]-<*end-range*>]; for example, **999[1-127]**.

Action

Change your application code to prevent this.

14204 Numeric value is not in the range specified in the mask

Cause

This error occurs if your application code attempts to set the text of a [JadeEditMask](#) control so that characters of the text corresponding to a numeric part of the mask are of a valid numeric but are outside of specified range part of the mask; for example, a [mask](#) value of [99\[12-36\]](#) and text of [40](#).

Action

Change your application code to prevent this.

14205 A numeric range in the mask for a JadeEditMask control can only be for a integer numeric

Cause

This error occurs if your application code attempts to set the range for a real numeric in the mask of a [JadeEditMask](#) control but a range can only be applied to integer numeric fields; for example, a [mask](#) value of [99.9\[12-36\]](#).

Action

Change your application code to prevent this.

14206 JadeEditMask control does not have a date mask for handling setTextFromDate or getTextAsDate

Cause

This error occurs for a [JadeEditMask](#) control when the [mask](#) property of the control does not indicate that the data is a single full date field and [getTextAsDate](#) or [setTextFromDate](#) is called.

Action

Change your application code to avoid using these methods, or modify the mask property to indicate that data is a date field.

14207 JadeEditMask control does not have a time mask for handling setTextFromTime or getTextAsTime

Cause

This error occurs for a [JadeEditMask](#) control when the [mask](#) property of the control does not indicate that the data is a single full time field and [getTextAsTime](#) or [setTextFromTime](#) is called.

Action

Change your application code to avoid using these methods, or modify the mask property to indicate that data is a time field.

14208 The data in a JadeEditMask control is incomplete

Cause

This error occurs when you attempt to retrieve the value of the text in a **JadeEditMask** control converted to a specific type and the text is incomplete; for example, if you call the **getTextAsTime** method of the **JadeEditMask** class and the hours or minutes part of the text is empty.

Action

Change your logic to check whether the text is valid according to the edit mask by calling the **isValid** method of the **JadeEditMask** class.

14210 Textbox case conflicts with textbox dataType

Cause

This error occurs when a created numeric field text box attempts to test for uppercase or lowercase input.

Action

Change your application code so that a numeric text box does not test for text case.

14211 Attaching to remote automation server not supported

Cause

This error occurs if an attempt is made to attach to an automation server on a remote machine (defined by the **remoteServerName** property of the **ActiveXAutomation** class). This is not supported, as you can attach only to local servers.

Action

You cannot attach to an automation server on a remote machine (when the **remoteServerName** property is set). Change your application code accordingly.

14212 Failed to attach to Automation server

Cause

This error occurs if an attempt to attach to an automation server fails. This could be for several reasons; for example:

- The specified server name is invalid.
- The classes are not registered on the server.
- There is no running object to attach to (and the **createIfNone** parameter passed to the **attachAutomationObject** method of the **ActiveXAutomation** class is **false**).

Action

Determine the cause of the failure and correct your application code accordingly.

14213 Failed to create Automation object

Cause

This error occurs if the **ActiveX** automation object create operation fails. This is mainly due to the classes not being registered or an invalid Globally Unique Identifier (GUID).

Action

Ensure that the **ActiveX** automation object being created is valid.

14214 Control type cannot be skinned

Cause

This error occurs if the **setSkin(skin)** method of the **Control** class is called on a control class that cannot handle skinning. Controls that are drawn entirely by an external routine such as **ActiveXControl**, **Ocx**, or **MultiMedia** cannot be skinned.

Action

Change your application code accordingly.

14215 Incompatible control skin set

Cause

This error occurs if the **setSkin(skin)** method of the **Control** class is called on a control with an incompatible skin type. Each control type that can be skinned has a corresponding **JadeSkin** type that must be used; for example, **setSkin(btnSkin)** invoked for a **Button** object must pass a **JadeSkinButton** parameter.

Action

Change your application code accordingly.

14216 This action is only valid when a collection is attached to the Control

Cause

This error occurs if the **refreshEntries** method has been called on a table, list box, or combo box to refresh the displayed entries from the collection attached to the control using the **displayCollection** or **listCollection** methods but a collection has not been attached.

Action

Change your application code accordingly.

14217 This feature is not available in this environment

Cause

This error occurs if a Windows-specific feature is used.

It can also occur if you attempt to clone instances of the [Printer](#) class or one of its subclasses

Action

No work-around is available.

14218 Internal InputType ComboBox cannot have entries directly added to the combo

Cause

This error occurs if logic is attempting to add entries directly to the internal combo box attached to a table where the value of the [inputType](#) property of the table is [Table.InputType_ComboBox](#). This combo box is automatically filled with the contents of the [comboList](#) property associated with the cell over which the combo box is placed. The text of the cell is used to select the entry in the combo. Using logic to add entries to the combo would cause disruption of this automatic process.

Action

Change your logic to avoid this error.

14219 Failed to insert OLE object - possibly an invalid insertion point

Cause

This error occurs if insertion of an object (for example, an image) into the contents of a [JadeRichText](#) control failed. Windows has not provided a reason for the failure. The most-likely cause is that the insert was attempted at a point not permitted by the [JadeRichText](#) control; for example, to the right of a table in the rich text control contents.

Action

Repeat the insert at a valid point in the contents. (A valid point also permits text to be entered at that location.)

14220 Failed to insert OLE object into text

Cause

This error occurs if an attempt to insert an **OLE** object into a rich text document fails.

Action

Consult the extended error text for more details about the error.

Check that the server of the object you are attempting to insert has been installed correctly. Consult the documentation for the software that created the object you are attempting to insert.

14221 OLE common dialog not found

Cause

This error occurs if the **oledlg.dll** DLL cannot be loaded or found.

Action

Ensure that the DLL is present and check your Windows installation, if required.

14222 Riched20.dll (version 3 or later) not found

Cause

This error occurs if the **riched20.dll** (version 3 or later) cannot be loaded or found.

Action

Ensure that the DLL is present and that the version is correct (version 3 or later is required; that is, **riched20.dll** 5.3.x.x or later). Check your Windows installation, if required.

14223 No object selected for the Object Properties Dialog

Cause

This error occurs if there is an attempt to display the Object Properties dialog and no object is selected. In practice, this error should occur rarely, if at all.

Action

Ensure that an object is selected before attempting to display the Object Properties dialog.

14224 The Number of columns widths doesn't match column count

Cause

This error is raised by the **insertTable** method if the number of widths specified by the **colWidths** parameter does not match the number of columns specified by the **cols** parameter; that is, the number of width values in the array does not tally with the number of columns.

Action

Change your application code to ensure that the number of widths and number of columns specified by the parameters match.

14225 Invalid line number specified

Cause

This error occurs if the **lineNumber** parameter specified for the **getLine** method of the **JadeRichText** class is less than **1** or greater than the number of lines.

Action

Change your application code to ensure that the line number specified by the parameter is in the valid range (**1** through to the total number of lines, where line one is the first line of the text).

14226 Load from file failed

Cause

This error is raised by the [JadeRichText](#) class [loadFromFile](#) method if a system error occurs when loading a Rich Text Format (RTF) file into a control.

The error text includes the Windows error description, which gives more details about the failure.

Action

Check the Windows error information and take the appropriate steps to correct the error.

14227 Save in file failed

Cause

This error is raised by the [JadeRichText](#) class [saveInFile](#) method if a system error occurs when saving the content of a Rich Text Format (RTF) control to a file.

The error text includes the Windows error description, which gives more details about the failure.

Action

Check the Windows error information and take the appropriate steps to correct the error.

14242 The create of the win32 interop host for a .NET control failed

Cause

This error occurs if the creation of the Microsoft control required to host a .NET control in a **win32** application fails.

Action

Perform the following actions, if required.

- Check that the version of the .NET control involved is correct.
- Check that the version of .NET installed is correct, and possibly try to re-install .NET.
- Restart the application, as the .NET control failure could be a resource issue.

14254 A recursive call to displayEntry or displayRow has been generated

Cause

This error occurs if the logic processed in a [displayRow](#) method or a [displayEntry](#) method on a list box or table results in that event method being called recursively. The most probable cause is that the method has called [app.doWindowEvents](#) or it has created a form. This method should not invoke such calls and should invoke only simple logic that results in the return of the string to be displayed within the list box or table entry.

Action

Adjust your logic to avoid the recursion.

14255 Application terminated after a failure of the TCP connection to the application server

Cause

This error occurs if a JADE application is terminated because the thin client TCP connection to the application server fails (usually as a result of a network failure).

Action

Restart the application. If the problem persists, seek assistance from your network management group.

14256 The `setPictureDescription` method can only get called on a cell that has an image assigned

Cause

This error occurs if the `setPictureDescription` method of the `JadeTableCell` class is called for a cell that does not have an assigned image.

Action

Ensure that the picture is assigned to the cell before the description is attached.

14257 A control or menu does not have the required reference property defined on its form class

Cause

This error occurs if a control on a form does not have the required associated reference property defined. Every control that is painted on a form requires a property reference, using the name of the control, to be added to the form class.

This problem usually occurs because the `.ddb` or `.ddx` (form and data definition) file associated with a schema has not been loaded.

Action

Load the `.ddb` or `.ddx` file, and then identify the problem forms by performing a schema validation on the forms.

If a mismatch still exists for a form, replace the form with a prior valid version of the form from a database backup.

14258 Batch mode drawing requires autoRedraw to be set

Cause

This error occurs if you call the [beginBatchDrawing](#) method on a window for which the [autoRedraw](#) property has not previously been set to **true**.

Action

Correct your logic.

14401 ActiveX properties/parameters can only be primitives, Arrays, or ActiveXInterfaces

Cause

This error occurs if you attempt to pass a JADE object that is not of an instance of an [Array](#) subclass or an [ActiveXInterface](#) subclass to a method parameter or property of type [Any](#). Whilst you can pass an instance of an [Array](#) subclass, it cannot be an attribute of another object.

Action

Change your application code to pass only primitives, **Array** subclass objects, or **ActiveXInterface** subclass objects. A common error (if you are copying a Visual Basic example) is to attempt to pass an [ActiveXAutomation](#) subclass object when your code should pass an **ActiveXInterface** subclass object.

14402 ActiveX Parameter description doesn't match actual parameters

Cause

This error occurs if there is an error in the generated code.

Action

Check that you have not modified the generated code. If this is the case, contact your local JADE support center or JADE Support if your JADE licenses include support.

14403 Event interface is not defined for this object

Cause

This error occurs if the [beginNotifyAutomationEvent](#) method is called with an incompatible **eventClassRefName** parameter. The type of the event reference is not compatible with the ActiveX object.

Action

Change your application code to only specify events of the object.

14404 ActiveX object not created or already deleted

Cause

This error occurs as a result of a coding error; for example, a method is called on an ActiveX control on a form that has previously been unloaded.

Action

Change your application code so that methods are not called on ActiveX controls after the containing form has been unloaded.

14544 A concurrent process has already opened the same database

Cause

This error occurs if you attempt to open a database that is already open in single user (exclusive) mode.

Action

Determine in which mode the database should be opened; that is, single user or multiuser mode.

14566 The network connection to the Jade Application Server was lost (possibly a network error) - attempting to reconnect

Cause

This system message is information passed to the user on the status of the connection. This message is included in the **jadmsgs.eng** file, to allow the text to be translated in the same way as the text for exceptions can be translated. For details, see "[Translating Messages](#)", in Chapter 11 of the *JADE Development Environment User's Guide*.

The situation that gave rise to the information message may subsequently result in an exception.

Action

No specific action is required.

14567 Attempted reconnect failed - The Jade Application Server cannot be reached

Cause

This system message is simply information passed to the user on the status of the connection. This message is included in the **jadmsgs.eng** file, to allow the text to be translated in the same way as the text for exceptions can be translated. For details, see "[Translating Messages](#)", in Chapter 11 of the *JADE Development Environment User's Guide*.

The situation that gives rise to this message may subsequently result in an exception.

Action

Restart your application.

14568 Attempted reconnect failed - The Jade Application Server cannot continue session

Cause

This system message is simply information passed to the user on the status of the connection. This message is included in the **jadmsgs.eng** file, to allow the text to be translated in the same way as the text for exceptions can be translated. For details, see "[Translating Messages](#)", in Chapter 11 of the *JADE Development Environment User's Guide*.

The situation that gives rise to this message may subsequently result in an exception.

Action

Restart your application.

14569 Attempted reconnect failed - msg sequence error

Cause

This system message is simply information passed to the user on the status of the connection. This message is included in the **jadmsgs.eng** file, to allow the text to be translated in the same way as the text for exceptions can be translated. For details, see "[Translating Messages](#)", in Chapter 11 of the *JADE Development Environment User's Guide*.

The situation that gives rise to this message may subsequently result in an exception.

Action

Restart your application.

14570 Attempted reconnect failed during initial write

Cause

This system message is simply information passed to the user on the status of the connection. This message is included in the **jadmsgs.eng** file, to allow the text to be translated in the same way as the text for exceptions can be translated. For details, see "[Translating Messages](#)", in Chapter 11 of the *JADE Development Environment User's Guide*.

The situation that gives rise to this message may subsequently result in an exception.

Action

Restart your application.

14571 Recursive reconnect attempts aborted

Cause

This system message is simply information passed to the user on the status of the connection. This message is included in the **jadmsgs.eng** file, to allow the text to be translated in the same way as the text for exceptions can be translated. For details, see "[Translating Messages](#)", in Chapter 11 of the *JADE Development Environment User's Guide*.

The situation that gives rise to this message may subsequently result in an exception.

Action

Restart your application.

14572 The network connection to the Jade Application Server has been lost (possibly a network error) and the reconnect facility has been disabled - restart your application

Cause

This system message is simply information passed to the user on the status of the connection. This message is included in the **jadmsgs.eng** file, to allow the text to be translated in the same way as the text for exceptions can be translated. For details, see "[Translating Messages](#)", in Chapter 11 of the *JADE Development Environment User's Guide*.

The situation that gave rise to this message may subsequently result in an exception.

Action

Restart your application.

14574 Unknown member for a .NET type

Cause

This error occurs if a member (method, property, or field) of the specified name does not exist for the current .NET type.

Action

If you have manually altered the classes, methods, and properties that were generated by the .NET import wizard, make sure the member names specified in your code match members that actually exist in the .NET type.

14575 Error during conversion between Jade and .NET types

Cause

This is an internal error arising from an exception raised by .NET when converting between .NET and JADE data types.

Action

The extended error text of the JADE exception object contains details about the conversion that caused the error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

14576 Error in .NET object during member call

Cause

This error occurs if an unhandled exception occurs inside the imported assembly when accessing a member (method, property, or field).

Action

Check the documentation for your imported components and that you are making correct use of the functionality provided by those components. The extended error text of the JADE exception object contains details of the .NET exception.

14577 Failed to create .NET object

Cause

This error occurs if JADE cannot create the required .NET object (component or class). The most-likely scenario is that the assembly in which the type being created is defined could not be loaded or found.

Action

Fix your code to correctly identify the type and assembly.

14578 No Jade control matching the .NET control

This is an internal error.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

14579 Not a valid .NET object for this method

Cause

This error occurs if you use one of the helper functions [getPicture](#), [getFont](#), or [getColor](#) on a [JadeDotNetType](#) instance that does not represent a matching .NET type; for example, using [getColor](#) on a .NET object that is not a [System::Drawing::Color](#) object.

Action

Correct your logic.

14580 Create of .NET control failed

Cause

This error occurs when an attempt to create a .NET control object fails during the creation of a runtime form or when loading a form into the JADE Painter.

Action

The extended error text for the exception contains the .NET exception text that explains the cause of the error. For example, the following extended error text indicates a missing DLL.

```
Create of .NET control failed
Exception [Could not load file or assembly 'SimpleDemo, Version=1.0.0.0,
Culture=neutral, PublicKeyToken=null' or one of its dependencies. The system cannot
find the file specified.]
```

14581 Failed to import .NET assembly

Cause

This error occurs when a .NET assembly is imported and indicates a problem with the assembly or one of its dependencies.

Action

The extended error text for the exception contains the .NET exception text that explains the cause of the error, which could possibly be a missing file.

14582 .NET object already created

Cause

This error occurs when you call the [createDotNetObject](#) method in the [JadeDotNetType](#) class more than once on the same JADE proxy object.

Note The [createDotNetObject](#) method is called automatically when you use the [createPicture](#), [createFont](#), or [createColor](#) method in the [JadeDotNetType](#) class.

It is also called when you call a method or access a property on a [JadeDotNetType](#) object without previously calling the [createDotNetObject](#) method.

Action

Remove one of the calls to the [createDotNetObject](#) method.

14583 .NET object still to be created

Cause

This is an internal error.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

14584 Failed to dispose .NET object

Cause

This error occurs when the **Dispose** method of a .NET object raises an exception.

Action

Arm an appropriate JADE exception handler. If required, report the issue to the developer of the imported .NET assembly.

15000 through 15099 JADE Print Errors

This section describes the errors that you could encounter when printing with JADE.

15001 Print must be of a frame

Cause

This error is raised by the `print` method of the `Printer` class when there is an attempt to print a control other than a frame.

Action

Change your application code to ensure that the control to be printed is a frame.

15002 New page on printer failed

Cause

This error is raised by the printer driver when printing a new page fails. The printer may not be online, or it may be configured incorrectly.

Action

Check that the printer is online. If it is online, check that it is configured correctly.

15003 Printer not open

Cause

This error occurs when you attempt to close a printer that is not currently open.

Action

Use the `isPrinterOpen` method of the `Printer` class to determine if the printer is currently open.

15004 Textout to printer failed

Cause

This error is raised by the printer driver when printing a frame fails. The printer may not be online or it may be configured incorrectly.

Action

Check that the printer is online. If it is online, check that it is configured correctly.

15005 Open of printer failed

Cause

This error is raised when JADE is unable to open the required printer.

The required printer could be the one specified when using the `setPrinter` method of the `Printer` class or the default printer as defined by the operating system for the current user. The reason for the failure is logged to the JADE messages (`jommsg.log`) log file. These failures include *printer not on line* and *default printer not found* when attempting to use the default printer.

The error can also be raised if you are printing to a **File:** device and you press the **Cancel** button on the dialog that is displayed for you to enter an output file name.

Action

Check that the printer is online. If it is online, check that it is configured correctly.

15006 Header and footer larger than page depth

Cause

This error occurs if the print output uses a header or footer frame (or both) that has exceeded the depth of the page (height).

Action

Reduce the height of the header or footer frame (or both).

15007 Frame larger than page depth

Cause

This error occurs if there is an attempt to print a frame that has a greater height than the page depth. This error can also be returned if the frame height is greater than the remaining depth on the page; for example, after the header frame has been printed.

Action

Reduce the frame depth (height).

15008 Printing started - change of printPreview ignored

Cause

This error occurs when an attempt to change the `printPreview` property of the `Printer` class is made after printing has begun (a call has already been made to the `print` method or the `newPage` method).

Action

Change your application code to ensure that the `printPreview` property is assigned to before the `print` or `newPage` method call.

15010 Printing started - change of copies ignored

Cause

This error occurs when an attempt to change the **copies** property of the **Printer** class is made after printing has begun (a call has already been made to the **print** method or the **newPage** method).

Action

Change your application code to ensure that the **copies** property is assigned to before the **print** or **newPage** method call.

15011 Orientation value must be Portrait (1) or Landscape (2)

Cause

This error occurs if you have assigned a value other than **Print_Portrait** (1) or **Print_Landscape** (2) to the **orientation** property of the **Printer** class.

Action

Change your application code to assign the correct **orientation** property.

15012 Printer internal construction of a page failed

Cause

This error occurs if the internal printer code fails while attempting to construct a page of output during printing.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

15013 Printer is currently open

Cause

This error occurs if there is an attempt to change the printer name during printing.

Action

Change your application code to ensure that the set printer call (using the **setPrinter** method of the **Printer** class) is made prior to the first print frame being executed.

15014 Failed to obtain local and favorite printers

Cause

This error occurs if the call to obtain the current printers available to this workstation fails.

Action

Ensure that at least one printer is available.

15015 Printing cancelled

Cause

This message is displayed if you click the **Cancel** button on the Print dialog while printing is in progress.

Action

Ensure that your application terminates printing if the **Cancel** button is clicked, as the cancel code closes the current document and aborts the printing.

15016 Printing stopped

Cause

This message is displayed if you click the **Stop** button on the Print dialog while printing is in progress.

Action

Ensure that your application terminates printing if the **Stop** button is clicked, as the stop code closes the current document. The following text is appended to the existing output.

```
*** User Stopped Printing ***
```

15017 Supplied printer does not match local and favorite printers

Cause

This error occurs if you attempt to set to a printer that is not available.

Action

Ensure that the printer is installed correctly or select a valid printer.

15021 No default printer exists for this user

Cause

This error occurs if you do not have a printer defined.

Action

Add a printer by using the Printers dialog, displayed by using the **Printers** command from the Settings submenu on your Windows Start menu, or from the Printers dialog on your Windows Control Panel.

15023 Printing started - change of printer ignored

Cause

This error occurs if there is an attempt to change the printer when printing has already been initiated. This error can occur if there is an attempt to change the printer in use before any printing has occurred, or if some printing has already been performed on the current printer and there is an attempt to change the printer during printing.

Action

Change your application code to prevent this.

If printing has occurred, the printer must be closed before starting new output on a different printer.

15024 The print position must be ≥ 0 and less than `printer.pageHeight`

Cause

This error occurs if your application code attempts to set the current position outside the page boundaries.

Action

Change your application code to prevent this.

15025 The printUnformatted request failed

Cause

This error occurs if the `printUnformatted` method of the `Printer` class fails.

Action

Refer to your JADE messages (`jommmsg.log`) log file for more details. Check that the output string being sent to the printer is valid. Check that the printer is capable of handling pass-through direct printing.

15027 PaperSource value is invalid

Cause

This error occurs if the `paperSource` value of the `Printer` class is not valid for the current printer.

Action

Change your application code to prevent this. Use the `getAllPaperSources` method to validate the required `paperSource`.

15028 Duplex value is invalid

Cause

This error occurs if the requested value of the `duplex` property of the `Printer` class is not valid.

Action

Check your application code for the value assigned to the `duplex` property.

15029 Printing started - change of duplex option ignored

Cause

This error occurs if your application code attempts to change the **duplex** property of the **Printer** class after printing has been initiated.

Action

Change your application code to prevent this.

15030 Printing started - change of collate option ignored

Cause

This error occurs if your application code attempts to change the **collate** property of the **Printer** class after printing has been initiated.

Action

Change your application code to prevent this.

15031 Printer object in use in print preview : cannot be reused

Cause

This error occurs when a printer object is in print preview mode and an attempt is made to use the printer object again.

Action

Change your application code to prevent this.

15032 DocumentType is invalid

Cause

This error occurs when the value of the **documentType** property of the **Printer** class is set to **Print_Custom_Paper**.

Action

Enter a valid value for the **documentType** property of the **Printer** class or use the **setCustomPaperSize** method.

15033 Error occurred during playback of print meta file

Cause

This internal error occurs when attempting to play back a print meta file (for print preview or to do the actual print).

Action

Refer to your JADE messages (**jommmsg.log**) log file for help in working around the problem.

As this is an internal error, if your JADE licenses include support, contact your local JADE support center or JADE Support.

15034 Failed to get page dimensions

Cause

This error occurs when the width or the height is not returned in response to a query from JADE for page dimensions.

Action

Install the latest printer drivers for the target device. If the problem persists, use another printer.

15600 through 15699 JadeTextEdit Control Errors

This section contains errors that can occur when using the [JadeTextEdit](#) control.

15602 Memory allocation failed

Cause

This error occurs if resource memory cannot be allocated.

Action

Perform one or more of the following actions:

- Shut down other applications that are not required
- Install more memory
- Increase virtual memory

15607 JadeTextEdit support library not found

Cause

This error occurs if the **jadedit.dll** is not present in the **bin** directory.

Action

Copy the library to the appropriate location.

15608 JadeTextEdit support library incomplete

Cause

This error occurs if a required entry point is missing from the **jadedit.dll**.

Action

Copy the correct version of the library to the appropriate location.

15609 JadeTextEdit support library does not match jade runtime

Cause

This error occurs if the version of the **jadedit.dll** does not match the JADE run time executable (**jade**).

Action

Copy the correct version of the library to the appropriate location.

15611 Invalid value type for parameter

Cause

This error occurs when a [JadeTextEdit](#) method is called with a parameter of the wrong type.

Action

Fix the portion of code that resulted in the error in your application program.

15612 Invalid value for parameter

Cause

This error occurs when a [JadeTextEdit](#) method is called with a parameter outside the valid range.

Action

Fix the portion of code that resulted in the error in your application program.

15622 Keyword list number must be 1 to 9

Cause

This error occurs when an invalid keyword list number parameter is passed to the [changeKeywords](#) method of the [JadeTextEdit](#) class.

Action

Fix the portion of code that resulted in the error in your application program.

15624 Property is readOnly

Cause

This error occurs when a property cannot be updated.

Action

Remove the property assignment.

15625 Unknown language number

Cause

This error occurs when the value assigned to the [language](#) property of the [JadeTextEdit](#) class does not match a known language.

Action

Fix the portion of code that resulted in the error in your application program.

15626 Unknown language number

Cause

This error occurs when the value assigned to the [language](#) property of the [JadeTextEdit](#) class does not match a known language.

Action

Fix the portion of code that resulted in the error in your application program.

15628 Unknown Named Attribute

Cause

This error occurs when the name parameter passed to the [getNamedAttribute](#) or [setNamedAttribute](#) method of the [JadeTextEdit](#) class is not recognized.

Action

Fix the portion of code that resulted in the error in your application program.

15629 Wrong value type for Named Attribute

Cause

This error occurs when the type of the value parameter passed the [getNamedAttribute](#) or [setNamedAttribute](#) method of the [JadeTextEdit](#) class does not match the required type for the given attribute name.

Action

Fix the portion of code that resulted in the error in your application program.

15630 Invalid value for Named Attribute

Cause

This error occurs if the value of the value parameter passed the [getNamedAttribute](#) or [setNamedAttribute](#) method of the [JadeTextEdit](#) class is outside the valid range for the specified attribute name.

Action

Fix the portion of code in your application program that resulted in the error.

15631 Named Attribute is readOnly

Cause

This error occurs if the specified named attribute cannot be passed to the [setNamedAttribute](#) method of the [JadeTextEdit](#) class.

Action

Fix the portion of code in your application program that resulted in the error.

15632 Named Attribute is writeOnly

Cause

This error occurs if the specified named attribute cannot be passed to the [getNamedAttribute](#) method of the [JadeTextEdit](#) class.

Action

Fix the portion of code in your application program that resulted in the error.

15636 Command number cannot be bound to a key stroke

Cause

This error occurs when a command number passed to the [bindKeyToCommand](#) method of the [JadeTextEdit](#) class is outside the range of bindable commands (2,000 through 2,999).

Action

Fix the portion of code in your application program that resulted in the error.

15637 List array has more than 125 entries

Cause

This error occurs when an array passed to the [setLinemarkLines](#) method of the [JadeTextEdit](#) class has more than 125 entries.

Action

Limit the number of entries in the array to 125.

15638 Attempt to assign too much text to the control

Cause

This error occurs if a file passed to the [loadTextFromFile](#) method of the [JadeTextEdit](#) class is too large. The maximum size of a file that you can load into the control by using the [loadTextFromFile](#) method is the smaller of 50M bytes or a tenth of the physical memory.

Action

Use another tool to edit the file or split it into smaller pieces.

15639 Filename is empty

Cause

This error occurs if a **filename** parameter that is an empty string is passed to the [loadTextFromFile](#) method or the [saveTextToFile](#) method of the [JadeTextEdit](#) class.

Action

Fix the portion of code in your application program that resulted in the error.

15640 Character outside decimal range of zero to 127

Cause

This error occurs if one of the *characterSet* strings passed to [setWordCharacterSets](#) of the [JadeTextEdit](#) class contains a character whose decimal value is greater than 127.

Action

Fix the portion of code in your application program that resulted in the error.

15643 Unrecognised action parameter value

Cause

This error occurs if a value passed in the **options** parameter to the [addText](#) method or the **action** parameter to the [changeKeywords](#), [doLinemarker](#), or [moveCaret](#) method of the [JadeTextEdit](#) class is invalid.

Action

Fix the portion of code that resulted in the error in your application program.

15645 Text contains an invalid Unicode character

Cause

This error occurs in a Unicode JADE environment if a string value is assigned to a property or is passed as a parameter to a method and the string value includes a character that is not a valid Unicode character.

Action

Fix the portion of code in your application program that resulted in the error.

15646 Clip buffer number out of range

Cause

This error occurs if a **buffnum** parameter passed to the [getClipBuffer](#) or [setClipBuffer](#) method of the [JadeTextEdit](#) class is not in the range zero (0) through 9.

Action

Fix the portion of code in your application program that resulted in the error.

15647 JadeTextEdit already interlocked

Cause

This error occurs if the [JadeTextEdit](#) receiver for a [shareDocumentFrom](#) method call has been used as the receiver or the parameter in a previous [shareDocumentFrom](#) method call.

In the following example, the exception would be raised if **control2** had participated in a previous **shareDocumentFrom** method call.

```
// control1 and control2 are instances of JadeTextEdit
control2.shareDocumentFrom(control1);
```

Action

Correct your coding.

15648 JadeTextEdit already interlocked

Cause

This error occurs if the **JadeTextEdit** parameter to a **shareDocumentFrom** method call has been passed as the receiver or the parameter in a previous **shareDocumentFrom** method call.

In the following example, the exception would be raised if **control1** had participated in a previous **shareDocumentFrom** method call.

```
// control1 and control2 are instances of JadeTextEdit
control2.shareDocumentFrom(control1);
```

Action

Correct your coding.

15649 JadeTextEdit cannot play keyboard macro while recording

Cause

This error occurs if you attempt to play a keystroke macro while recording is in progress.

Action

Change the code.

15650 JadeTextEdit macro aborted - caret moved outside document

Cause

This error occurs if you attempt to move the caret (cursor) before the first character of text or after the last character of text. The macro replay stops when this occurs.

Action

None required.

15651 JadeTextEdit macro aborted - no match for find

Cause

This error occurs during the replaying of a **JadeTextEdit** macro if a *find* or *replace* command does not find any matching text. The macro replay stops when this occurs.

Action

None required.

15652 JadeTextEdit invalid autoComplete cache file content - ignored

Cause

This error occurs if the contents of the JADE AutoComplete cache file are not valid. The AutoComplete cache file contains a list of **RootSchema** class definitions that were previously loaded by the AutoComplete process from the database.

Action

No action is required. The cache file is discarded and the information is reloaded from the database, as required.

15901 through 15999 Thin Client Connection Balancing Errors

This section describes the errors relating to JADE thin client connection balancing.

15901 TCCB Node is not an App Server

Cause

This error occurs if the [Application](#) class [enableThinClientConnBalancing](#) method is invoked on a node which is not an application server.

Action

Change your application code to avoid calling this method on a node that is not an application server.

15902 TCCB AppServerGroupName missing or invalid

Cause

This error occurs if the [AppServerGroupName](#) parameter in the [\[JadeAppServer\]](#) section of the JADE initialization file is missing or does not have a specified value.

Action

Add or correct the [AppServerGroupName](#) parameter in the [\[JadeAppServer\]](#) section of the JADE initialization file.

15903 TCCB [JadeAppServer]NodeName missing or invalid

Cause

This error occurs if the [NodeName](#) parameter in the [\[JadeAppServer\]](#) section of the JADE initialization file is missing or does not have a specified value.

Action

Add or correct the [NodeName](#) parameter in the [\[JadeAppServer\]](#) section of the JADE initialization file.

15904 TCCB Unencrypted Address,Port missing or invalid

Cause

This error occurs if the [ExternalAppServerAndPort](#), [AppServerPort](#), or [AppServer](#) parameter in the [\[JadeAppServer\]](#) section of the JADE initialization file is invalid, or the [AppServer](#) and [AppServerPort](#) parameters are both invalid.

Action

Add or correct the appropriate parameters in the JADE initialization file.

15905 TCCB SSL Address,Port missing or invalid

Cause

This error occurs if the [ExternalAppServerAndPort](#), [AppServerPort](#), or [SSLSecurePort](#) parameter in the [\[JadeAppServer\]](#) section of the JADE initialization file is invalid, or the [AppServer](#) and the [SSLSecurePort](#) parameters are both invalid.

Action

Add or correct the appropriate parameters in the JADE initialization file.

15906 TCCB DLL (jadegm) missing or entrypoint not found

Cause

This error occurs if the [jadegm.dll](#) is missing, or one of its entry points is not available.

Action

Ensure that the [jadegm.dll](#) is present, and that its version is compatible with [jadapp](#) and [jadappb](#).

15907 TCCB Invoke picker method test error - see jommsg.log

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

15908 TCCB All picker processors have terminated

Cause

This error occurs when all processes in the node which called the [Application](#) class [enableThinClientConnBalancing](#) method have terminated. This node no longer redirects connections to other application servers.

Action

Correct your application code so that at least one process which calls the [Application](#) class [enableThinClientConnBalancing](#) method never terminates.

15909 TCCB No AppServers registered with specified group name

Cause

This is an internal error and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support, providing appropriate error logs.

15910 TCCB AppServer group cannot accept additional ThinClients

Cause

This error occurs when all application servers in the group have reached their maximum user count, and additional thin client connections are rejected.

Action

Increase the value specified in the [MaxLocalProcesses](#) parameter in the JADE initialization file to allow more processes to sign on when the node is next initialized. Alternatively set the value to zero (**0**), to remove the restriction on the number of processes that are signed on.

Alternatively, add another application server to the group.

15911 TCCB Timeout selecting AppServer

Cause

This error occurs when a thin client connection times out after waiting for 30 seconds for an application server to be chosen.

Action

Retry connecting to an application server.

16000 through 19999 JADE Development Environment Errors

This section describes errors that can occur when using the JADE development environment.

16000 A JADE user cancelled an operation

Cause

This error occurs when a user cancels an action; for example, by clicking a **Cancel** button.

Action

Ignore this error if it occurs as a handled exception.

If it occurs as an unhandled error, contact your local JADE support center or JADE Support if your JADE licenses include support.

16001 A call was made to an obsolete method

Cause

This is an internal JADE error that should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

16002 A call was made to a console method that does not have a defined callback

Cause

This is an internal JADE error that should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

16102 The class was not found in the current schema branch

Cause

This error occurs when loading a **.ddb** or **.ddx** (form and data definition) file if the specified class is referred to but cannot be found in the current schema or its superschemas.

Action

Check that the class exists in the schema branch into which you are loading the **.ddb** or **.ddx** (form and data definition) file. Ensure that the class name is identical. If the class does not exist, define it by using the Class Browser or by using a schema file.

16103 Expected token

Cause

This error occurs when loading a **.ddb** or **.ddx** (form and data definition) file if the specified token is expected but it is not the next token read from the file.

Action

The **.ddb** or **.ddx** (form and data definition) file may have been altered or corrupted. Try a fresh extraction from your JADE database and then repeat the load action.

16104 The specified file is not available

Cause

This error occurs when attempting to load a **.ddb** or **.ddx** (form and data definition) file if the specified file cannot be found.

Action

Check that the file exists in the specified location. If the file is on a Local Area Network (LAN) server volume, you should also check that you have the necessary access rights to open the file.

16105 The class is invalid in this context

Cause

This error occurs when loading a **.ddb** or **.ddx** (form and data definition) file if the specified class is referred to but it is of an invalid class in this context; for example, a collection class when a non-collection class is expected.

Action

Check that the class definition is correct in the schema branch into which you are loading the **.ddb** or **.ddx** (form and data definition) file.

16106 The forms have invalid entries

Cause

This error occurs after loading a **.ddb** or **.ddx** (form and data definition) file if a check on the validity of form definitions detects that the forms, their controls, and their menus are not consistent with each other.

Action

Do a selective extract from the original system of the forms that had errors reported for them and then reload them into the new system. For assistance or if the error still occurs, contact your local JADE support center or JADE Support if your JADE licenses include support.

16107 Unexpected token

Cause

This error occurs while loading a **.ddb** or **.ddx** (form and data definition) file if the specified token is encountered but it was not expected at this point in the file. The **.ddb** or **.ddx** (form and data definition) file may have been altered or corrupted.

Action

Attempt a fresh extraction from your JADE database. If the error still occurs, contact your local JADE support center or JADE Support if your JADE licenses include support.

16108 The form will be assigned the definitions of the default locale

Cause

This error occurs when loading a **.ddb** or **.ddx** (form and data definition) file if a mismatch is detected between the locales defined for the target schema and the locales in the file.

Action

No action is required for this error. It is issued only as a warning, for your information. The additional locales in the schema are assigned the definitions of the default file locale.

16109 Unknown references were detected in the loaded definition

Cause

This error occurs when loading a **.ddb** or **.ddx** (form and data definition) file if references to objects that are not defined in the file are read.

Action

Remove the offending references that are logged in your JADE messages (**jommsg.log**) log file from the original system. Extract the **.ddb** or **.ddx** (form and data definition) file again and then load it into the new system.

16110 The property of your Global object has an invalid reference

Cause

This error occurs when loading a **.ddb** or **.ddx** (form and data definition) file if a property of the **Global** object references an object that was not found in the file.

Action

Set the offending references to **null** in the original system. Extract the **.ddb** or **.ddx** (form and data definition) file again and load it into the new system.

16111 The form for the application is missing from the loaded definition

Cause

This error occurs when loading a **.ddb** or **.ddx** (form and data definition) file if the specified forms definition was not found in the file but is referenced by the specified application.

Action

Extract the application again, including the specified form.

16112 The parent of the control on the form is missing from the loaded definition

Cause

This error occurs when loading a **.ddb** or **.ddx** (form and data definition) file if the parent property of the specified control references a control or form that was not found in the file.

Action

Extract the form containing the control again, including the referenced form or control.

16113 The format of the DDB file is incompatible with this version of Jade

Cause

This error occurs if the format of the **.ddb** (form and data definition) file being loaded is a **.ddb** format from a release prior to JADE release 5.0, or is later than the version of JADE that you are using.

This error also occurs if you attempt to load a **.ddx** file into a JADE release earlier than JADE 18.0.01.

Action

Load your **.ddb** files from your prior JADE version into a JADE 5.0 system along with their **.scm** (schema) files and then extract them as JADE 5.0 format files, and then load these into your system. If the **.ddb** or **.ddx** files are a later version than the version of JADE you are using, upgrade your version of JADE to be the same version or later than that of the **.ddb** or **.ddx** files.

16114 Expected more input than was available

Cause

This error occurs if a problem is encountered while reading from the input file in a **.ddb** or **.ddx** (form and data definition) load. The error may have been caused by some other application corrupting or deleting the file during the load. If the file is on a Local Area Network (LAN) server volume, the LAN connection may have been lost during the load.

Action

Check that the file is still present and has not been corrupted. If the file is on a LAN server volume, check that the LAN connection is still valid and that the file is accessible.

16116 Patch Control Check failed

Cause

This error occurs when patch control is on and patch control extensions is on and an entry point in a user-defined library specified by the **DevelopmentSecurityLibrary** parameter in the [\[JadeSecurity\]](#) section of the JADE initialization file is called for an entry point that exists and returns a value of **false**.

Action

Check that the user-defined entry point is doing what is expected.

16117 Application used in one or more packages and cannot be deleted

Cause

This error occurs when a load attempts to delete an application that is defined as being used in one or more packages. Applications that are used this way cannot be deleted.

Action

Change the package definitions so that the application is no longer used then re-extract the schema or include the application as part of the extract

16118 Versioning failed

Cause

This error occurs when a schema load cannot complete because a reorganization is currently in progress, or a reorganization has failed or is waiting for the transition to be carried out.

Action

Wait for the reorganization to complete, abort the reorganization, or carry out the transition.

16119 The RPS Mapping columns do not match the Jade definitions

Cause

This error occurs when an RPS mapping is updated by loading a form and data definition file (.ddb or .ddx) and the mapping is not consistent with the current definitions of JADE attributes.

Action

Ensure that changes to JADE attribute definitions are loaded before loading an RPS mapping that depends on the updated definitions.

16120 The ddb file specifies a schema that does not exist

Cause

This error occurs when a form and data definition file (.**ddb** or **.ddx**) is loaded and a control does not have a control reference on the form to handle form translations.

Action

Ensure that the schema metadata file (**.scm**) and form and data definition file (.**ddb** or **.ddx**) match.

31000 through 31099 TCP/IP Connection Errors

This section describes the errors that can occur when using a TCP/IP connection.

31000 Connection failed to initialize

Cause

This error occurs if the TCP/IP network fails to initialize.

Action

Install the TCP/IP protocol.

31001 Connection no target host name specified

Cause

This error occurs if you have not specified the **Connection** class **name** property.

Action

Set the value of the **name** property to the target host name, as specified in the host file.

31002 Connection target host not found

Cause

This error occurs if the specified value in the **Connection** class **name** property cannot be found on the host file, or is not available on your Domain Name Service (DNS) server.

Action

Specify the correct **name** property or enter the correct target name in the host file.

31003 Connection invalid port specification

Cause

This error occurs if the **TcpIpConnection** class **port** property is invalid.

Action

Enter a value for the **port** property that is greater than zero (0).

31004 Connection failed to connect

Cause

This error occurs if the TCP/IP layers fail to open the requested connections.

Action

Check that the host name is mapped to the correct IP number in the host file. Check that its port number matches the port number of the target listener.

31005 Connection bind failed

Cause

This error occurs if the TCP/IP bind function fails. This error can occur if the specified listener port is already in use by another listener.

Action

Ensure that only one listener is running on this port for every workstation. Check your JADE messages (**jommsg.log**) log file for a more-specific message or error number, and then contact your network administrator.

31006 Connection listen failed

Cause

This error occurs if the TCP/IP listen function fails.

Action

Check the JADE messages log file (**jommsg.log**) for a more-specific message or error number and then contact your network administrator.

31007 Connection accept failed

Cause

This error occurs if the TCP/IP accept function fails.

Action

Check your JADE messages (**jommsg.log**) log file for a more-specific message or error number and then contact your network administrator.

31008 Connection read error

Cause

This error occurs if the TCP/IP receiver encounters an error.

Action

Check your JADE messages (**jommsg.log**) log file for a more-specific message or error number and then contact your network administrator.

31009 Connection write error

Cause

This error occurs if the TCP/IP sender encounters an error.

Action

Check your JADE messages (**jommsg.log**) log file for a more-specific message or error number and then contact your network administrator.

31010 Connection close error

Cause

This error occurs if an error is detected during the closing of a TCP/IP network connection.

Action

Check your JADE messages (**jommsg.log**) log file for a more-specific message or error number and then contact your network administrator.

31011 Connection invalid state for requested operation

Cause

This error occurs if the requested operation is not valid for the current state of the network object.

Action

Ensure that the current network state is valid for this operation.

31012 Connection read operation is already pending

Cause

This error occurs if a read operation is already in progress for this connection.

Action

Change your application code so that a multiple read cannot occur on the same network object.

31013 Connection attribute cannot be modified directly

Cause

This error occurs if there is an attempt to modify the **Connection** class **state** property or the **TcplpConnection** class **remoteName** or **remotelpAddress** property.

Action

Change your application code to ensure you do not attempt to modify the **state**, **remoteName**, or **remotelpAddress** property.

31014 Connection PostThreadMessage for event notification failed

Cause

This error occurs if the internal event processing fails.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

31015 Connection memory allocation failed

Cause

This error occurs if the system failed to allocate memory.

Action

Perform one of the following actions.

- Shut down other applications that are not required.
- Increase virtual memory.
- Add more memory.

31017 Connection was closed

Cause

This error occurs if the connection was closed by the remote device or by your application while an operation was still in progress.

Action

If this message occurs unexpectedly, check your application and ensure that it does not close the network.

31018 Connection request cancelled

Cause

This error occurs if a network operation was interrupted.

Action

Check that the network was not closed from another location running your application.

31019 Connection no message name for asynchronous method

Cause

This error occurs if the message **name** parameter is of zero length.

Action

Specify a valid method name for the [Connection](#) class **name** parameter.

31020 Connection receiver does not have specified method

Cause

This error occurs if the target object for the callback event does not have a method with the specified name, or the object specification is invalid.

Action

Change the message name to an existing name, or change the target object to one that has the specified method.

31021 Connection receiver method must not return a value

Cause

This error occurs if the notification method returns a value.

Action

Remove the return value from the specified method.

31022 Connection receiver object class validation error

Cause

This error occurs if a specified receiver object class cannot be validated.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

31023 Connection receiver object specification is NULL

Cause

This error occurs if a **null** value is specified for the receiver object.

Action

Specify a valid receiver object.

31024 Connection receiver method validation error

Cause

This error occurs if the validation of the specified receiver object method name fails.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

31025 Connection receiver method return type validation error

Cause

This error occurs if the validation of the specified receiver object method return type fails.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

31026 Connection failed to load dynamic library

Cause

This error occurs if the [TcplpConnection](#) class failed to load the DLL.

Action

Check the name and path of the specified library and that it is a 32-bit version of the library.

31027 Connection failed to find method in library

Cause

This error occurs if the [TcplpConnection](#) class fails to obtain the address of the specified method in the appropriate library.

Action

Check that the name of the specified method matches the name of the method in the library. The match must be exact, and the comparison is case-sensitive.

31028 Connection authentication failure

Cause

This error occurs if the authentication execution fails.

Action

Check that authentication has been set up correctly on both the client and the server side of the application.

Check the JADE messages ([jommsg.log](#)) log files on the client and server for specific return values of the authentication methods.

31029 Connection encryption error

Cause

This error occurs if the encryption of a network message fails.

Action

Check that the encryption description has been set up correctly on both the client and the server side of the application.

Check the JADE messages (**jommsg.log**) log files on the client and server for specific return values of the encryption method.

31030 Connection decryption error

Cause

This error occurs if the decryption of a network message fails.

Action

Check that the decryption description has been set up correctly on both the client and the server side of the application.

Check the JADE messages (**jommsg.log**) log files on the client and server for specific return values of the decryption method.

31031 Connection invalid connection

Cause

This error occurs if the **TcplpConnection** class encounters an internal reference pointer.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

31032 Connection may not be a persistent object

Cause

This error occurs if you create the **TcplpConnection** class as a persistent object.

Action

Create **TcplpConnection** as a transient object.

31033 Connection invalid parameter type

Cause

This error occurs if a **TcplpConnection** class parameter is not of the correct type.

Action

Check your application code and ensure that all of your parameters are correct.

31034 Connection negative or zero buffer size invalid

Cause

This error occurs if you have specified a negative or zero-length buffer size when performing an asynchronous **readBinary** operation.

Action

Ensure that you specify a buffer size of positive length with the **readBinary** method.

31035 Connection listen operation is already pending

Cause

This error occurs if a listen operation is already in progress for this connection.

Action

Change your application code so that a multiple listen cannot occur on the same network object.

31036 Connection read timeout

Cause

This error occurs if the connection times out during a read operation.

Action

The timeout period is user initiated, and it is the responsibility of the user to decide the action to be taken.

31037 Connection write timeout

Cause

This error occurs if the connection times out during a write operation.

Action

The timeout period is user-initiated, and it is the responsibility of the user to decide the action to be taken.

31038 Connection listen timeout

Cause

This error occurs if the connection times out during a listen operation.

Action

The timeout period is user-initiated, and it is the responsibility of the user to decide the action to be taken.

31039 Connection invalid invocation

Cause

This error occurs when an asynchronous method of a [TcplpConnection](#) object is executed from a **serverExecution** method.

Asynchronous operations are not valid from a **serverExecution**, since the calling thread is not a valid JADE process thread and is therefore unable to execute callbacks on behalf of the JADE process.

Action

Change your application code so that it does not use asynchronous operations from **serverExecution** methods.

31040 Connection delimiter size invalid

Cause

This error is raised by the [readUntil](#) or [readUntilAsynch](#) methods of the [TcplpConnection](#) class when the delimiter is an invalid size. The specified delimiter must be of non-zero length.

Action

Change your application code to specify a non-**null** delimiter in the method signature.

31041 Connection readUntil buffer full

Cause

This error is for internal use only and should not normally occur.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

31043 Connection proxy mismatch or proxy not defined

Cause

This error occurs if an object reference mismatch is detected between the proxy server and the proxy specified in the **tcpipConnection** parameter, or if the attempt to establish a connection fails.

Action

Change your application code to specify the correct connection reference to the [TcplpConnection](#) object specified in the **tcpipConnection** parameter of the [JadeTcplpProxy](#) class [connect](#) method.

31048 Connection SSL no cipher

Cause

This error occurs if no common cipher is found between client and server when initializing the connection.

Action

Specify a list of ciphers on the client and server with at least one entry in common. Alternatively, specify no cipher and let SSL work out the strongest cipher.

31049 Connection SSL no certificate

Cause

This error occurs if no certificate is supplied when certificate verification is enabled.

Action

Supply the SSL certificate or switch off certificate verification.

31050 Connection SSL invalid method type

Cause

This error occurs if an unknown method type is specified.

Action

Supply one of the documented method types.

31051 Connection method called with empty parameter list

Cause

This is an internal error involving the JADE Web Services framework, and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

31100 through 31199 MultiWorkerTransport Errors

This section describes error messages that can occur when connecting using the multiple-worker TCP transport mechanism.

31102 MultiWorkerTransport property is readOnly after beginListening

Cause

This error occurs if the property cannot be changed after the `beginListening` method of the `JadeMultiWorkerTcpTransport` class has been called.

Action

Fix the portion of code that resulted in the error in your application program so that the property is assigned before the `beginListening` method is invoked.

31103 MultiWorkerTransport beginListening has not been called

Cause

This error occurs if the `beginListening` method of the `JadeMultiWorkerTcpTransport` class must be called before the method that caused the exception.

Action

Fix the portion of code in your application program that resulted in the error.

31106 MultiWorkerTransport connection not assigned to worker

Cause

This error occurs if a worker process attempts to access a connection that is not assigned to that worker.

Action

Fix the portion of code in your application program that resulted in the error.

31107 MultiWorkerTransport connection already bound to another worker

Cause

This error occurs if an attempt is made to bind a connection that is already bound to another worker.

Action

Fix the portion of code in your application program that resulted in the error.

31108 MultiWorkerTransport userObject cannot be nonSharedTransient

Cause

This error occurs when the object assigned to the [userGroupObject](#) property of the [JadeMultiWorkerTcpTransport](#) class does not have a lifetime of **sharedTransient** or **persistent**.

Action

Fix the portion of code in your application program that resulted in the error.

31116 MultiWorkerTransport different MWTT class/subclass

Cause

This error occurs if the transport group specified by [listenHostname](#) and [listenPortnumber](#) is originally created using a different subclass of [JadeMultiWorkerTcpTransport](#).

Action

Fix the portion of code in your application program that resulted in the error.

31117 MultiWorkerTransport beginListening refused by validateServerProcess

Cause

This error occurs if the attempt by this worker process to call the [beginListening](#) method of the [JadeMultiWorkerTcpTransport](#) class is rejected by the [validateServerProcess](#) method.

Action

Fix the portion of code in your application program that resulted in the error.

31122 MultiWorkerTransport connection not bound to worker

Cause

This error occurs if the worker process calling the connection unbind method is not the worker that is currently bound to the connection.

Action

Fix the portion of code in your application program that resulted in the error.

31123 MultiWorkerTransport unknown connectionId

Cause

This error occurs if the `connectionId` passed to the `causeUserEventOnConnId` method of the `JadeMultiWorkerTcpTransport` class does not match an existing connection.

Action

Fix the portion of code in your application program that resulted in the error.

31124 MultiWorkerConnection writeD8 message exceeds 99,999,999 bytes

Cause

This is an internal error involving the JADE Web services framework and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

31125 MultiWorkerConnection readD8Buffered found invalid length

Cause

This is an internal error involving the JADE Web services framework and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

31126 MultiWorkerConnection getBufferedChunk not preceded by successful readBuffered

Cause

This is an internal error involving the JADE Web services framework and should not normally occur.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support.

31129 Connection has timed out and has been closed

Cause

This error occurs if a connection times out after a Web service message has been read but before the message has been processed. Incoming messages for the connection are discarded.

Action

Increase the value of the **MessageTimeout** parameter setting in the **jadehttp.ini** file or increase the number of copies of the Web service provider application.

Another possibility is that the processing of the request is taking longer than expected. Investigate whether this is the case and change the code as required.

31250 through 31299 Service Event Log Errors

Errors and warnings in the range 31250 through 31299 can be reported to an event log (for example, the **jadehttp.log** file) when running a service.

Detailed results of the service events are written to the appropriate log file in your specified directory, to enable you to investigate further, if required.

31300 through 31399 JADE Serial Port Errors

This section describes error messages that can occur when connecting through a serial port.

31300 Numeric serial port number out of range

Cause

This error occurs when the supplied serial port number is outside the valid range of **0** through **255**, as shown in the following code fragment.

```
serialport.name := "COM493:";
```

Action

Specify a valid port number.

31301 Serial port device is already open

Cause

This error occurs if you call the [open](#) or [openAsynch](#) method of the [JadeSerialPort](#) class and the connection has previously been opened.

Action

Check that the value of the [state](#) property for the connection is not **Connected (2)** before attempting to open the serial port.

31302 Specified port name is not valid, or does not exist

Cause

This error occurs if you specify a serial port number that does not exist on your computer. For example, if you have two serial ports (COM1 and COM2) only, the following code fragment raises an exception.

```
serialport.name := "COM3";  
serialport.open;
```

Action

Ensure that the serial port name is valid and that it exists. Note that dialogs for adding printers can list serial COM ports that do not exist. Use the **mode COMx** command, where **x** is the port number, to see if the port is still valid.

31303 Obtaining Serial I/O singleton failed

Cause

This error is for internal use only and should not normally occur.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

31304 Serial port is not open

Cause

This error occurs if you attempt to communicate through the serial port (for example, using the [listen](#) or [readBinary](#) method of the [JadeSerialPort](#) class) and the serial port has not been opened.

Action

Open the serial port connection before attempting to use it. Check that the value of the [state](#) property for the connection is **Connected (2)**.

31305 Serial port open failed, check jommsg log for further details

Cause

This error occurs if the [open](#) or [openAsynch](#) method of the [JadeSerialPort](#) class fails to open the serial port.

Action

Examine your JADE messages ([jommsg.log](#)) log file for further details of the failure.

31306 Serial port listen failed, check jommsg log for further details

Cause

This error occurs if the [listen](#) or [listenAsynch](#) method of the [JadeSerialPort](#) class fails.

Action

Examine your JADE messages ([jommsg.log](#)) log file for further details of the failure.

31307 Problem occurred closing device

This error occurs if the [close](#) or [closeAsynch](#) method of the [JadeSerialPort](#) class fails to close the serial port.

Action

Examine your JADE messages ([jommsg.log](#)) log file for further details of the failure.

31308 Reading from the device failed

Cause

This error occurs if the [readBinary](#), [readBinaryAsynch](#), [readUntil](#), or [readUntilAsynch](#) method of the [JadeSerialPort](#) class fails to read from the serial port.

Action

Examine your JADE messages ([jommsg.log](#)) log file for further details of the failure.

31309 Reading from the device failed

Cause

This error occurs if the [readBinary](#), [readBinaryAsynch](#), [readUntil](#), or [readUntilAsynch](#) method of the [JadeSerialPort](#) class fails to read from the serial port.

Action

Examine your JADE messages ([jommsg.log](#)) log file for further details of the failure.

31310 Writing to the device failed

Cause

This error occurs if the [writeBinary](#) or [writeBinaryAsynch](#) method of the [JadeSerialPort](#) class fails to write to the serial port.

Action

Examine your JADE messages ([jommsg.log](#)) log file for further details of the failure.

31311 C++ pointer to SerialPort object is invalid

Cause

This error is for internal use only and should not normally occur.

Action

This is an internal error. If your JADE licenses include support, contact your local JADE support center or JADE Support.

31312 Invalid timeout value (value < zero)

Cause

This error occurs if the [timeout](#) property of the [Connection](#) class is set to a negative value, as shown in the following code fragment.

```
serialport.timeout := -10;
```


Action

Set the value of the **timeout** property to a positive number to represent the number of milliseconds before a listen, read, or write operation times out, or to zero (0) if there is no timeout.

31313 Operation was attempted when no presentation client serial port exists

Cause

This error occurs if the [usePresentationClient](#) property of the [JadeSerialPort](#) class is set **true** (so that the serial port on the presentation client rather than the application server is to be used) and no such port is available.

Action

Investigate why a serial port was not available on the presentation client.

31314 Serial port an asynchronous operation timeout

Cause

This error occurs if the [openAsync](#), [listenAsync](#), [readBinaryAsync](#), [readUntilAsync](#), [writeBinaryAsync](#), or [closeAsync](#) method of the [JadeSerialPort](#) class times out.

Action

Investigate why the operation timed out and consider using exception handling for these errors.

31315 Serial port open operation timeout

Cause

This error occurs if the [open](#) or [openAsync](#) method of the [JadeSerialPort](#) class times out.

Action

Investigate why the operation timed out and consider using exception handling for these errors.

31316 Serial port listen operation timeout

Cause

This error occurs if the [listen](#) or [listenAsync](#) method of the [JadeSerialPort](#) class times out.

Action

Investigate why the operation timed out and consider using exception handling for these errors.

31317 Serial port close operation timeout

Cause

This error occurs if the [close](#) or [closeAsync](#) method of the [JadeSerialPort](#) class times out.

Action

Investigate why the operation timed out and consider using exception handling for these errors.

31318 Serial port read operation timeout

Cause

This error occurs if the [readBinary](#), [readBinaryAsynch](#), [readUntil](#), or [readUntilAsynch](#) method of the [JadeSerialPort](#) class times out.

Action

Investigate why the operation timed out and consider using exception handling for these errors.

31319 Serial port write operation timeout

Cause

This error occurs if the [writeBinary](#) or [writeBinaryAsynch](#) method of the [JadeSerialPort](#) class times out.

Action

Investigate why the operation timed out and consider using exception handling for these errors.

31320 Serial port communications error found, parity error etc.

Cause

This error occurs if a parity error is found indicating a corruption of the signal, or some other communications error is found.

Action

Consider using exception handling for these errors.

31321 Serial port an asynchronous operation failed

Cause

This error occurs if the [openAsynch](#), [listenAsynch](#), [readBinaryAsynch](#), [readUntilAsynch](#), [writeBinaryAsynch](#), or [closeAsynch](#) method of the [JadeSerialPort](#) class fails.

Action

Investigate why the operation failed and consider using exception handling for these errors.

31322 Serial port blocking communications due to one of CTS/DSR/RSLD state

Cause

This error occurs when the serial port is waiting to receive a Clear To Send (CTS) signal, Data Set Ready (DSR), or Receive Signal Line Detect (RSLD) signal. These signals are required before information can be transmitted from the serial port.

Action

Investigate why the CTS, DSR, or RSLD signal has not been received.

31323 Serial port closed during I/O operation

Cause

This error occurs when a serial port is closed while data is being received or transmitted using the [readBinary](#), [readBinaryAsynch](#), [readUntil](#), [readUntilAsynch](#), [writeBinary](#), or [writeBinaryAsynch](#) method of the [JadeSerialPort](#) class.

Action

Investigate why the serial port was closed.

31324 Serial port operation attempted while asynchronous close in progress

Cause

This error occurs if an asynchronous operation was attempted while an existing asynchronous close operation is still in progress.

Action

Change the code to wait until the current asynchronous close operation completes before initiating a new operation.

31325 Serial port operation attempted while asynchronous open in progress

Cause

This error occurs if an asynchronous operation was attempted while an existing asynchronous open operation is still in progress.

Action

Change the code to wait until the current asynchronous open operation completes before initiating a new operation.

31326 Serial port operation attempted while asynchronous listen in progress

Cause

This error occurs if an asynchronous operation was attempted while an existing asynchronous listen operation is still in progress.

Action

Change the code to wait until the current asynchronous listen operation completes before initiating a new operation.

31327 Serial port operation attempted while asynchronous read in progress

Cause

This error occurs if an asynchronous operation was attempted while an existing asynchronous read operation is still in progress.

Action

Change the code to wait until the current asynchronous read operation completes before initiating a new operation.

31328 Serial port operation attempted while asynchronous write in progress

Cause

This error occurs if an asynchronous operation was attempted while an existing asynchronous write operation is still in progress.

Action

Change the code to wait until the current asynchronous write operation completes before initiating a new operation.

31500 through 31599 Network Proxy Errors

This section describes error messages that can occur when connecting with a network proxy.

31500 NetworkProxy memory allocation failed

Cause

This error occurs if the network proxy could not allocate resource memory.

Action

Perform one or more of the following actions.

- Shut down other applications that are not required
- Install more memory
- Increase virtual memory

31501 NetworkProxy proxyType not (yet) supported

Cause

This error occurs if the proxy communications protocol specified by the [proxyType](#) property of the [JadeTcpIpProxy](#) class is out of range or not currently supported. Only the HTTP proxy type and the connect part of the SOCKS V4 protocol are supported.

Action

Specify the correct proxy communication protocol in the [proxyType](#) property of the [JadeTcpIpProxy](#) class object.

31502 NetworkProxy proxy request was cancelled

Cause

This error occurs if a network proxy request for connection has failed for some reason. The request was not granted by the proxy.

Action

Check with your network administrator to establish the reason for the cancelled request.

31503 NetworkProxy network read failed

Cause

This error occurs if the data interchange has failed in the process of creating a connection. This is most-likely due to a low-level network connection problem.

Action

Check with your network administrator to establish the reason for the failed network operation.

31504 NetworkProxy network write failed

Cause

This error occurs if the data interchange has failed in the process of creating a connection. This is most-likely due to a low-level network connection problem.

Action

Check with your network administrator to establish the reason for the failed network operation.

31505 NetworkProxy Authentication package not supported

Cause

This error is reserved for future use.

Action

None.

31506 NetworkProxy Bad credentials, domain/username/password wrong

Cause

This error occurs if authentication required by the proxy server fails because a connection attempt is made specifying an incorrect domain, user name, or password.

Action

Change your application code to specify the correct [domain](#), [password](#), and [userName](#) properties.

31507 NetworkProxy Unexpected error from Security API

Cause

This error is raised by an authentication problem when attempting to connect to a Microsoft domain. The execution of the normal handshake has failed, and messages have not been exchanged in the expected order.

Action

This error indicates an internal network problem. Contact your network administrator. It may be possible to disable domain-level security on the proxy server. Alternatively, contact your local JADE support center or JADE Support if your JADE licenses include support.

31508 NetworkProxy Bad handshake protocol

Cause

This error is raised by a failure in the authentication process when attempting to connect to a Microsoft domain. The proxy server has failed to supply the expected information.

Action

This error indicates an internal network problem. Contact your network administrator. Alternatively, contact your local JADE support center or JADE Support if your JADE licenses include support.

31509 NetworkProxy Unable to obtain credentials

Cause

This error occurs when an attempt to obtain the user credentials for sign-on to the proxy server fails, usually due to an internal resource problem.

This typically occurs when the user is unable to sign on normally and the application attempts to obtain a user name and password by creating a dialog. The dialog is not displayed, due to a resource failure.

Action

Consider reimplementing your application code to supply the correct user name and password, or check with your network administrator to establish the reason for the resource failure.

31510 NetworkProxy User does not want to supply credentials

Cause

This error occurs when an attempt to obtain the user name and password of the user for sign-on to the proxy server cannot proceed, because the user will not or cannot supply the information.

Action

Enter the correct user name and password and try again, contact your network administrator to obtain a valid user name and password, or check that you have access to this domain.

31512 NetworkProxy no network specified

Cause

This error is reserved for future use.

Action

None.

31513 NetworkProxy unsupported browser type

Cause

This error occurs if the browser type specified by the **JadeTcplpProxy** class **browserType** property is not supported.

Action

Change your application code to specify the correct **browserType** property of the **JadeTcplpProxy** class object. At present, the Netscape Navigator and Microsoft Internet Explorer Web browsers are supported.

31600 through 31699 WebSocket Network Errors

This section describes error messages that can occur when connecting with a WebSocket protocol.

31600 WebSocket Protocol Error

Cause

This error occurs if an error in the WebSocket protocol is detected.

Action

Further information about the error is stored in the **Exception** class **extendedErrorText** property.

If this error persists and you are unable to resolve it by other means, contact your local JADE support center or JADE Support if your JADE licenses include support.

31700 through 31899 Messaging Framework Errors

This section describes error messages that can occur when systems communicate using messages and message queues.

31700 Method and/or parameter unsupported

Cause

This error occurs if you attempt to set a property of a [JadeGenericQueue](#) instance using the [setQueueProperty](#) method of the [JadeGenericQueue](#) class, where the property is specified by the **propname** parameter and the value is specified by the **value** parameter.

It is intended that in a future release you will be able to use the **setQueueProperty** method to set transport-specific properties.

Action

Do not use the **setQueueProperty** method until the intended functionality is implemented.

31701 Memory allocation failed

Cause

This error occurs if all virtual memory available for queues and messages has been exhausted; for example, if there are many queues containing many messages.

Action

It may be possible to free up memory by processing existing messages in the queue. A more drastic option, which loses the messages in queue, is to restart the node.

31702 Object must be process transient

Cause

This error should not occur. It is to ensure that instances of the [JadeMessagingFactory](#) class and other JADE messaging framework classes are only created as transient objects. The allowed lifetime of these classes is **transient**, so the error should never arise.

Action

No action is required.

31704 Handle address space error

Cause

This error occurs when a [JadeMessagingFactory](#) instance or [JadeGenericQueue](#) instance is created on one node and an attempt is made to use that object on another node by executing a [serverExecution](#) method. This is not supported.

Action

Correct your code.

31706 String parameter has embedded null

Cause

This error occurs if a string parameter contains a null character where none is expected. For example, if the [fullName](#) of a [JadeGenericQueue](#) instance class is used as a parameter, it is not permitted to contain an embedded null character.

Action

Correct your code.

31707 String parameter has non-US-ASCII characters

This error occurs if a string parameter contains a character with a [Decimal](#) value outside the range **0** through **255** where such a character is not expected.

Action

Correct your code.

31708 String parameter has invalid characters

Cause

This error occurs when a string parameter contains characters that are invalid for the purpose of the method. In the following example, the first parameter of the [openQueue](#) method of the [JadeMessagingFactory](#) class is not in the correct *transport_name :// queue_manager_name / queue_name* format (because a forward slash character is missing).

```
myReplyQueue := factory.openQueue("JadeMQ:/localnode/TestQ",  
                                   "Access=Public;Usage=Put");
```

Action

Correct your code.

31709 Queue fullname format invalid

Cause

This error occurs if the **fullName** property of the **JadeGenericQueue** class is not specified in the correct format. The correct format is **transport_name :// queue_manager_name / queue_name**, as shown in the following example.

```
JadeMQ://localnode/TestQ
```

Action

Correct your code.

31710 Transport name unknown

Cause

This error occurs when the transport name is not supported. In the following example, the first parameter of the **openQueue** method of the **JadeMessagingFactory** class has the correct format, **transport_name :// queue_manager_name / queue_name**, but **BogusMQ** is not a valid transport name.

```
myReplyQueue := factory.openQueue("BogusMQ://localnode/TestQ",  
                                   "Access=Public;Usage=Put");
```

Action

Correct your code.

31713 Option name too long

Cause

This error occurs if the length of the name of an option exceeds the limit of **100** characters.

Action

Correct your code.

31715 Option name has invalid character

Cause

This error occurs if the characters in the name of an option do not conform to the expected format.

- The first character must be a letter (**A** through **Z**, or **a** through **z**)
- Subsequent characters must be letters or the underscore character

Action

Correct your code.

31716 Option name not recognized

Cause

This error occurs if the **options** parameter passed to method is not recognized. In the following example, the second parameter of the **beginMessage** method of the **JadeGenericQueue** class has incorrectly been entered as **Nowait** instead of **NoWait**.

```
myQueue.beginMessage(msg, "Nowait"); // Option should be "NoWait"
```

Action

Correct your code.

31717 Option must have value

Cause

This error occurs if an option is provided that is expected to specify a value and none is provided.

In the following example, the second parameter of the **beginMessage** method of the **JadeGenericQueue** class does not provide a value for the **Timeout** option (for example, **Timeout=10000**).

```
myQueue.beginMessage(msg, "Timeout=");
```

Action

Correct your code.

31718 Option value not recognized

Cause

This error occurs if an option specified in the **options** parameter passed to method has a value that is not recognized.

In the following example, the second parameter of the **openQueue** method of the **JadeMessagingFactory** class does not have a valid name value for the **Usage** option (for example, **Usage=Put**).

```
myQueue := fcty.openQueue("JadeMQ://localnode/TestQ",  
                           "Access=Public;Usage=UnKnown");
```

Action

Correct your code.

31720 Quoted text has no matching trailing quote character

Cause

This error occurs when an option takes a quoted text value and a trailing quote character has been omitted.

In the following example, the second parameter of the `openQueue` method of the `JadeMessagingFactory` class has an option `AccessPassword` that takes a quoted value. The option should have been entered as `AccessPassword='secret'` but the trailing quote character is missing.

```
myQueue := factory.openQueue("JadeMQ://localnode/TestQ",  
                              "Access=Protected;AccessPassword='secret");
```

Action

Correct your code.

31721 Option value has invalid character

Cause

This error occurs if the first character of an option in the `options` parameter passed to method is not a valid alphanumeric character. In the following example, the second parameter of the `openQueue` method of the `JadeMessagingFactory` class has a `Usage` option that incorrectly begins with a comma.

```
myQueue := factory.openQueue("JadeMQ://localnode/TestQ", "Usage=,All");
```

Action

Correct your code.

31722 Option expects name value

This error occurs if an option in the `options` parameter passed to method is expected to have a specific name value but none has been specified. In the following example, the second parameter of the `openQueue` method of the `JadeMessagingFactory` class does not have a valid name value for the `Access` option (for example, `Access=Public`).

```
myQueue := fcty.openQueue("JadeMQ://localnode/TestQ", "Access=1;Usage=Put");
```

Action

Correct your code.

31723 Option expects numeric value

Cause

This error occurs if the `options` parameter passed to method is expected to have a numeric value but none has been specified. In the following example, the second parameter of the `beginMessage` method of the `JadeGenericQueue` class does not have a numerical value specified for the `Timeout` option (for example, `Timeout=10000`).

```
myQueue.beginMessage(msg, "Timeout=millisecs");
```

Action

Correct your code.

31724 Option expects quoted text value

Cause

This error occurs when an option requires a quoted text value and one has not been provided. In the following example, the second parameter of the `openQueue` method of the `JadeMessagingFactory` class has an option `AccessPassword` that takes a quoted value. The option should have been entered as `AccessPassword='999'` but the quote characters are missing.

```
myQueue := factory.openQueue("JadeMQ://localnode/TestQ",  
                              "Access=Protected;AccessPassword=999");
```

Action

Correct your code.

31725 Option value is invalid

Cause

This error occurs when an option requires a value in a specific range and the value is outside the allowed range.

Action

Correct your code.

31727 Messaging module is closing down

Cause

This error should not normally occur. It could possibly arise if the node that is hosting the messaging queue is closing down.

Action

Wait until the node hosting the messaging queue is restarted.

31728 Message length exceeds transport limit

Cause

This error occurs if the message length exceeds the maximum length allowed by the transport. The maximum value is the value of the `maxMessageLength` property of the `JadeGenericQueue` class if it is set, or the maximum value permitted by the transport.

Action

Reduce the size of payload of the message, which is contained in the `body` property of the `JadeGenericMessage` object. Alternatively, split the message content into a number of smaller messages.

31729 Property is readonly

Cause

This error occurs when an attempt is made to set the value of a read-only property of a JADE messaging framework entity.

In the following example, an attempt is made to alter the read-only `expiresWhen` property of a `JadeGenericMessage` object by using `setPropertyValue` property of the `Object` class.

```
msg.setPropertyValue("expiresWhen",ts);           // ts is a TimeStamp value
```

Action

Do not use `setPropertyValue` method to change a read-only property of a JADE messaging framework entity.

31730 Negative time offset

Cause

This error occurs when the `expiresWhen` property of a `JadeGenericMessage` instance is set to a date and time value that has already expired.

In the following example, the `expiresWhen` property is set using the `setExpiryRelativeToNow` method with a negative value for the `lifeTimeSeconds` parameter.

```
msg.setExpiryRelativeToNow(-10);
```

Action

Correct your code.

31731 Cannot modify message attributes

Cause

This error occurs if you attempt to change the value of the property of a `JadeGenericMessage` instance after calling the `beginMessage` or `putMessage` method of the `JadeGenericQueue` class.

In the following example, an attempt is made to set the `replyQueueFullName` property of a `JadeGenericMessage` instance after calling the `beginMessage` method. (The property should be set before calling the `beginMessage` method.)

```
myQueue.beginMessage(msg, "Timeout=10000");  
msg.replyQueueFullName := myQueue.fullName;
```

Action

Correct your code.

31732 Cannot append to message body

Cause

This error occurs if you attempt to append information to the body of a `JadeGenericMessage` instance that is incompatible with the format of the message.

In the following example, an attempt is made to append a string to a message by using the [appendString](#) method of the **JadeGenericMessage** class after starting to build the message in the **JadeTpls** format.

```
msg.format := "JadeTpls";  
msg.appendBodyTuple("name", "wilbur");  
msg.appendString("Hello World");
```

Action

Correct your code.

31733 Message class is not compatible with queue transport

Cause

This error occurs if you create a [JadeGenericMessage](#) instance for one transport (for example, the WebSphere MQ transport) and attempt to put the message to a [JadeGenericQueue](#) instance that was created for a different transport (for example, the JadeMQ transport).

Action

Ensure a message is sent to a queue that is created for the same transport as the message.

31734 Queue is not open

Cause

This error occurs if you attempt an operation on a queue after the queue has been closed using the [close](#) method of the [JadeGenericQueue](#) class.

Action

Correct your code.

31735 Only localnode queue manager is currently supported

Cause

This error occurs when using the JadeMQ transport and you attempt to access the queue manager executing on another node. In the following example, the first parameter of the [openQueue](#) method of the [JadeMessagingFactory](#) class specified the URI of the queue. (The only permissible location for the queue manager is **localnode**; that is, "**JadeMQ://localnode/queuename**".)

```
myQueue := factory.openQueue("JadeMQ://remotenode/TestQ", null);
```

Action

Correct your code.

31737 Option must not have value

Cause

This error occurs if an option is provided that is expected not to specify a value and one is provided. In the following example, the second parameter of the `beginMessage` method of the `JadeGenericQueue` class provides a value for the `NoWait` option. (The option should simply be specified as `"NoWait"`.)

```
myQueue.beginMessage(msg, "NoWait=true");
```

Action

Correct your code.

31738 Queue does not exist

Cause

This error occurs if you try to open a queue with the `create=Never` option (or equivalently the `MustExist` option) and the queue does not in fact exist.

In the following example, the `openQueue` method of the `JadeMessagingFactory` class is used to open a queue and the queue does not exist.

```
vars
  factory : JadeMessagingFactory;
begin
  create factory transient;
  myQueue := factory.openQueue("JadeMQ://localnode/TestQ", "MustExist");
epilog
  delete factory;
end;
```

Action

Correct your code.

31739 Queue already exists - create refused

This error occurs if you try to open a queue with the `Create=Always` option (or equivalently the `MustCreate` option) and the queue already exists.

In the following example, the `openQueue` method of the `JadeMessagingFactory` class is used to open a queue and the queue already exists.

```
vars
  factory : JadeMessagingFactory;
begin
  create factory transient;
  myQueue := factory.openQueue("JadeMQ://localnode/TestQ", "MustCreate");
epilog
  delete factory;
end;
```

Action

Correct your code.

31740 Queue is private - access refused

Cause

This error occurs if you try to open a queue with the **Access=Private** option and the queue has already been opened by another process. A queue that is opened with this option can only be accessed by the process that opened it.

In the following example, the `openQueue` method of the `JadeMessagingFactory` class is used to open a queue and the queue has already been opened by another process.

```
vars
    factory : JadeMessagingFactory;
begin
    create factory transient;
    myQueue := factory.openQueue("JadeMQ://localnode/TestQ",
                                "Access=Private");
epilog
    delete factory;
end;
```

Action

Correct your code.

31741 Queue is protected - access refused

This error occurs if you try to open a queue with the **Access=Protected** option and you do not specify the correct **AccessPassword**. To access a queue that is created with the **Access=Protected** option you must specify the correct password.

In the following example, the `openQueue` method of the `JadeMessagingFactory` class is used to create a queue with the password specified by the option **AccessPassword='East'**.

```
myQueue := factory.openQueue("JadeMQ://localnode/TestQ",
                             "Access=Protected; AccessPassword='East'");
```

Subsequently the following attempt is made to access the queue using an incorrect password.

```
myQueue := factory.openQueue("JadeMQ://localnode/TestQ",
                             "Access=Protected; AccessPassword='West'");
```

Action

Correct your code.

31742 Queue is guarded - access refused

This error occurs if you try to open a queue that was created with the **Access=Guarded** option (which is currently not implemented) and you fail to satisfy the specified access requirements.

In the following example, the `openQueue` method of the `JadeMessagingFactory` class is used to create a queue **Access=Guarded** option.

```
myQueue := factory.openQueue("JadeMQ://localnode/TestQ", "Access=Guarded");
```

Action

Correct your code.

31744 ReplyQueueName transport must match message transport

Cause

This error occurs if the transport type specified in the [replyQueueFullName](#) property of a [JadeGenericMessage](#) object differs from the transport type for the message itself; that is, you cannot send a message on a JadeMQ queue and request a reply to be sent to a WebSphere MQ queue.

Action

Correct your code.

31745 CorrelationID value is too long

Cause

This error occurs if the [correlationID](#) property exceeds the maximum length allowed for the transport, which is 24 bytes on WebSphere MQ and 64 bytes on JadeMQ. This error should not normally occur as the value of the [correlationID](#) property of a reply message is obtained from the value of the [messageID](#) property of the request message.

Action

Correct your code.

31746 Message format value is too long

This error occurs if the [format](#) property exceeds the maximum length allowed for the transport, which is 8 characters on WebSphere MQ and 16 characters on JadeMQ.

Action

Correct your code.

31747 Message not initialized for put

Cause

This error occurs if you attempt to *put* (that is, send) a message to a message queue and the message queue has not been initialized for that operation.

In the following example, an attempt is made to send a message using the [putMessage](#) method of the [JadeGenericQueue](#) class but the queue has been opened with the [Usage=Get](#) option.

```
myQueue := fcty.openQueue("JadeMQ://localnode/TestQ",  
                          "Access=Public;Usage=Get");  
msg := myQueue.createMessage(true);
```

```
msg.appendString("Hello World");  
myQueue.putMessage(msg, null);
```

Action

Correct your code.

31748 Message not initialized for get

This error occurs if you attempt to *get* (that is, retrieve) a message from a message queue and the message queue has not been initialized for that operation. In the following example, an attempt is made to retrieve a message using the **getMessage** method of the **JadeGenericQueue** class but the queue has been opened with the **Usage=Put** option.

```
myQueue := fcty.openQueue("JadeMQ://localnode/TestQ",  
                           "Access=Public;Usage=Put");  
msg := myQueue.createMessage(true);  
myQueue.getMessage(msg, null);
```

Action

Correct your code.

31749 Message parameter is null

Cause

This error occurs if you attempt to *get* (that is, retrieve) a message from a message queue or *put* (that is, send) a message to a message queue and the msg parameter is null.

In the following example, an attempt is made to retrieve a message using the **getMessage** method of the **JadeGenericQueue** class but the queue msg variable is not initialized.

```
// msg := myQueue.createMessage(true);  
myQueue.getMessage(msg, null);
```

Action

Correct your code.

31752 PutMessage timeout waiting for queue space

Cause

This error occurs if you attempt to send a message to a JadeMQ queue that would cause the memory used by the queue to exceed the value of the **maxMemoryInuse** property for the queue, or would cause the number of messages to exceed the value of the **maxMessageCount** property for the queue. This error does not apply to messages sent to a WebSphere MQ queue.

Action

You could use exception handling to handle these situations.

31753 Attribute value too small

Cause

This error occurs if you set the value of the [maxMemoryInuse](#) property or the [maxMessageCount](#) property for a queue to a value less than one.

Action

Correct your code.

31754 Masked queue name too long

Cause

This error occurs when a queue name is specified with a trailing asterisk (*) indicating a mask from which a unique name is generated when the queue is opened (the queue name is part of the first parameter of the [openQueue](#) method of the [JadeMessagingFactory](#)) and the number of characters exceeds the maximum allowed length for the transport type. The maximum length for a masked queue name is 100 characters for the WebSphere MQ transport and 70 characters for the JadeMQ transport.

Action

Correct your code.

31755 Queue name too long

Cause

This error occurs when a queue name is specified without a trailing asterisk (*) indicating a mask (the queue name is part of the first parameter of the [openQueue](#) method of the [JadeMessagingFactory](#) class) and the number of characters exceeds the maximum allowed length for the transport type. The maximum length for an unmasked queue name is 40 characters for the WebSphere MQ transport and 120 characters for the JadeMQ transport.

Action

Correct your code.

31756 ReplyQueueName cannot include masked queue name

Cause

This error occurs if you attempt to set the [replyQueueFullName](#) property of a [JadeGenericMessage](#) object with a trailing asterisk (*) indicating a mask from which a unique name is generated when the queue is opened. A masked queue name is only permitted when a queue name is specified as part of the first parameter of the [openQueue](#) method of the [JadeMessagingFactory](#) class.

Action

Correct your code.

31757 Only the queue creator can change this property

Cause

This error occurs when an existing JadeMQ queue is opened and an attempt is made to change the values of the [maxMemoryInuse](#), [maxMessageCount](#), and [maxMessageLength](#) properties. Only the process that originally created the queue is permitted to change the values of these properties.

Action

Correct your code.

31758 Queue not opened for put

Cause

This error occurs if an attempt is made to send a message to a queue that was opened by using the [openQueue](#) method of the [JadeMessagingFactory](#) class with the **options** parameter containing the option **Usage=Get** or other value that does not allow messages to be sent.

Action

Correct your code.

31759 Queue not opened for get

Cause

This error occurs if an attempt is made to retrieve a message from a queue that was opened by using the [openQueue](#) method of the [JadeMessagingFactory](#) class with the **options** parameter containing the option **Usage=Put** or other value that does not allow messages to be retrieved.

Action

Correct your code.

31760 Queue not opened for inquiries

Cause

This error occurs if an attempt is made to inquire on information about a queue that was opened by using the [openQueue](#) method of the [JadeMessagingFactory](#) class with the **options** parameter containing the option **Usage=Put** or other value that does not allow inquiries to be made about messages in the queue; for example, attempting to execute the [countQueuedMessages](#) method raises this exception.

Action

Correct your code.

31761 Option not supported

Cause

This error should not normally occur. It indicates that you are specifying an option for a queue that is not valid for the transport of the queue.

Action

Correct your code.

31762 Property is writeonly

Cause

This error occurs if you attempt to display the value of the `defaultGetMessageOptions` or `defaultPutMessageOptions` property of a `JadeGenericQueue` instance. These properties are used for convenience in setting default values for `getMessage` and `putMessage` method calls.

The following code fragment raises the exception.

```
queue := factory.openQueue("JadeMQ://localnode/TestQ", "");  
write queue.defaultGetMessageOptions;           // raises the exception
```

Action

Do not attempt to display the value of the `defaultGetMessageOptions` or the `defaultPutMessageOptions` property.

31763 Expected a semicolon between options

Cause

This error occurs if an `options` parameter passed to method has not separated one option from the next with a semicolon character. In the following code fragment, the second parameter of the `beginMessage` method of the `JadeGenericQueue` class does not have a semicolon between the parameters `NoWait` and `Timeout=10000`.

```
myQueue.beginMessage(msg, "Nowait Timeout=10000");  
// Options parameter should be "Nowait;Timeout=10000"
```

Action

Correct your code.

31764 Queue already opened with Usage=Get

Cause

This error occurs when a queue is created with `SingleUser` access and another process attempts to open the queue.

Action

Correct your code.

31767 Message length exceeds queue limit

Cause

This error occurs if you attempt to add a message to the queue that exceeds the maximum allowed length for messages in the queue. The upper limit for the message length depends on the transport, but this value can be reduced for a particular queue by setting the value of the `maxMessageLength` property the `JadeGenericQueue` class.

Action

Correct your code.

31768 Invalid property name

Cause

This error occurs if the name of the queue or message property is not two through 100 characters in length, or it does not start with a letter followed by letters, digits, and underscore characters.

Action

Correct your code.

31769 Unrecognized property name

Cause

This error should not normally occur. It indicates that a system property has been incorrectly specified.

Action

Correct your code.

31770 Unsupported property type

Cause

This error occurs if the `setMessageProperty` property of the `JadeGenericMessage` class is used to set a user property for a message in a JadeMQ queue and the type of the user property is an unsupported primitive type such as `MemoryAddress`.

Action

Correct your code.

31771 User property exceeds available space

Cause

This error occurs if total memory allocated to user properties in a message in a JadeMQ queue, which are defined by using the `setMessageProperty` property of the `JadeGenericMessage` class, exceeds 32K bytes.

Action

Correct your code.

31772 User property value exceeds 9999 bytes

Cause

This error occurs if memory allocated to a user property in a message in a JadeMQ queue, which is defined by using the [setMessageProperty](#) property of the [JadeGenericMessage](#) class, exceeds 9999 bytes.

Action

Correct your code.

31776 Transport table is full

Cause

This error should not normally occur with the currently supported transports.

Action

No action is required.

31777 Type of user property is incompatible with value parameter

Cause

This error occurs if the type of a variable used to retrieve the value of a user property in a message in a JadeMQ queue, which is retrieved by using the [getMessageProperty](#) property of the [JadeGenericMessage](#) class, is not compatible with the type of the **value** parameter; for example, if the property is of type [String](#) and the variable used as the **value** parameter is of type [Integer](#).

Action

Correct your code.

31778 Correlation ID list has too many entries

Cause

This error occurs if you use the [getMessageByCorrelationID](#) method of the [JadeGenericQueue](#) class to retrieve a message from the queue where the value of the [correlationID](#) property of the message matches one of the elements of the **correlIdList** parameter.

The value of the **correlIdList** parameter is either a single [Binary](#) value or a [BinaryArray](#) value. In WebSphere MQ, the **correlIdList** parameter can only be a single **Binary** value. In JadeMQ, the **correlIdList** parameter can be a **BinaryArray** list but the number of entries must not exceed 64.

Action

Correct your code.

31779 Invalid body tuple name

Cause

This error occurs if a message is composed of *name-value* tuples and the name does not conform to the following requirements.

- First character must be an alphabetic character
- Subsequent characters can be alphanumeric, underscores, or periods
- Length must not exceed 90 characters

Action

Change your code.

31780 Unsupported body tuple type

Cause

This error occurs if you attempt to append a body tuple value of type **MemoryAddress**. This is the only primitive type that is not valid as a tuple value.

Action

Do not attempt to append a body tuple value of type **MemoryAddress**.

31782 Body text is not valid tuple format

Cause

This error occurs if you are unsuccessful in an attempt to build the payload of a **JadeGenericMessage** object by directly assigning to the **body** property.

Action

Use the **appendBodyTuple** method to build the message payload.

31783 Type of bodyTuple is incompatible with value parameter

Cause

This error occurs if the **value** parameter of the **getBodyTuple** method is not compatible with the type of the data in the tuple (for example, if the data in the tuple is a **String** and the **value** parameter is type **Integer**).

Action

The incompatibilities can be avoided if the **value** parameter is declared as type **Any**.

31785 IBMWMQ exception

Cause

This error occurs if the WebSphere MQ application is unable to carry out an operation. A completion code and reason code are returned that JADE stores in the **errorItem** property of the exception. For example, an unknown channel name specified in the **openQueue** option MQSERVER clause causes exception 31785 with an **errorItem** property of "**completion=2, reasonCode=2059**".

Action

After determining the completion code and reason code from the JADE exception, see the **WebSphere MQ Messages** manual for more details.

31786 IBMWMQ does not support this feature

Cause

This error occurs if you attempt to use a feature that is not supported by the WebSphere MQ transport (for example, user properties are only available with the JadeMQ transport).

Action

Change your code.

31788 IBMWMQ invalid MQSERVER option value

Cause

This error occurs if the information specified to connect a WebSphere MQ client to the queue manager is invalid. When you open a WebSphere MQ queue you must include an **MQSERVER** clause as part of the **openQueue** method options, embedded in the **fullName** parameter or as one of the options.

For more details, see "[Opening a WebSphere MQ Queue](#)", in Chapter 15 of the *JADE Developer's Reference*. For details about the client interface, see the **WebSphere MQ Client Manual**.

Action

Correct your code.

31789 QueueManager is unknown

Cause

This error should not occur if you are using a standard documented interface.

Action

Contact your local JADE support center or JADE Support if your JADE licenses include support.

31790 QueueManager name too long

Cause

This error occurs if the name of the queue manager specified in the `openQueue` method of the `JadeMessagingFactory` class exceeds the limit for the transport. For example, the queue manager name for the WebSphere MQ transport cannot exceed 40 characters.

Action

Correct your code.

31791 IBMWMQ queue open must have explicit Usage option

Cause

This error occurs if the `options` parameter in the `openQueue` method of the `JadeMessagingFactory` class does not contain an explicit `Usage` option describing how the queue is to be used; for example, `Usage=Get`.

Action

Correct your code.

31792 IBMWMQ unable to start background thread

Cause

This error occurs when the limit on the number of threads that the WebSphere MQ application server is configured to support is exceeded.

Action

This is a WebSphere MQ configuration issue.

31796 IBMWMQ Model open option is required with masked queue name

Cause

This error occurs when using the WebSphere MQ transport if you specify a queue name with an asterisk (*) and the temporary model queue has not been created. WebSphere MQ requires a masked queue to be based on an existing queue.

You must define a model queue and create the actual queue based on the model.

Action

Change your code.

31797 Body must be empty for first appendBodyTuple() call

Cause

This error occurs if the [appendBodyTuple](#) method of the [JadeGenericMessage](#) class is called when the body of the message contains binary data either by using the [appendBinary](#), [appendString](#), [appendStringAsUtf8](#), or [appendStringUtf8](#) methods or directly assigning a value to the [body](#) property.

Action

Do not call the [appendBinary](#), [appendString](#), [appendStringAsUtf8](#), or [appendStringUtf8](#) methods or assign to the [body](#) property if you intend to call the [appendBodyTuple](#) method.

31798 Message not initialized for put or begin

Cause

This error occurs if a queue is opened to retrieve or inquire on messages, using the [openQueue](#) method of the [JadeMessagingFactory](#) class, and an attempt is made to use the [putMessage](#) method or the [beginMessage](#) method of the [JadeGenericQueue](#) class.

Action

Change your code.

31799 beginMessage already called for this message

Cause

This error occurs if you call the [beginMessage](#) method of the [JadeGenericQueue](#) class for the same message more than once.

Action

Change your code.

31800 putMessage following beginMessage must have empty options parameter

Cause

This error occurs if you call the [putMessage](#) method of the [JadeGenericQueue](#) class with a non-null [options](#) parameter after previously calling the [beginMessage](#) method for the same message. In this scenario, the options for the message are specified in the [beginMessage](#) call and cannot be changed when the message construction is completed and the message is *put* in the queue by using the [putMessage](#) method.

Action

Change your code.

31801 putMessage following beginMessage must use the same queue

Cause

This error occurs if the receiver of the [putMessage](#) method is a different [JadeGenericQueue](#) instance from the one that was used as the receiver for the [beginMessage](#) method call. When the [beginMessage](#) method is called, the message is associated with a queue (the receiver of the [beginMessage](#) method) and that association cannot be changed.

Action

Change your code.

31802 IBMWMQ getMessageByCorrelId is limited to one correlationID value

Cause

This error occurs if the [getMessageByCorrelationID](#) method of the [JadeGenericQueue](#) class is called for the WebSphere MQ queue with a [BinaryArray](#) as the [correlIdList](#) parameter.

In WebSphere MQ, the [correlIdList](#) parameter can be a single [Binary](#) value only. (A JadeMQ queue allows a [BinaryArray](#) with up to 64 correlation ID values as the [correlIdList](#) parameter.)

Action

Change your code.

31804 Attribute value must be greater than or equal to zero

Cause

This error occurs if the [feedback](#), [type](#), or [report](#) property of a [JadeGenericMessage](#) instance is set to a negative value. The value of these numeric properties must be greater than or equal to zero (0).

Action

Correct your code.

31806 Reply QueueManager name is unknown

Cause

This error should not occur if you are using a standard documented interface.

Action

Contact your local JADE support center or JADE Support if your JADE licenses include support.

31807 Reply Queue name is unknown

Cause

This error should not occur if you are using a standard documented interface.

Action

Contact your local JADE support center or JADE Support if your JADE licenses include support.

31809 Queue name is prohibited

Cause

This error occurs if you use a reserved queue name prefix; for example, **Jade*** or **JADE***. These prefixes are reserved for JADE to create internal queues.

Action

Change your code.

31813 Queue not opened for modify

Cause

This error occurs if you attempt to modify the [maxMemoryInuse](#), [maxMessageCount](#), or [maxMessageLength](#) property of a [JadeGenericQueue](#) instance and the queue was not opened, using the [openQueue](#) method of the [JadeMessagingFactory](#) class, with the option of modification.

Action

Open the queue with the **options** parameter including **Usage=Mod** or **Usage=All**, which enables you to modify queue properties.

31820 Option takes a single value

Cause

This error occurs if you specify more than one value for an option of the **beginMessage**, **putMessage**, or **getMessage** method in the [JadeGenericQueue](#) class and the option expects a single value, as shown in the following incorrect code fragment.

```
requestQueue.putMessage(msg, "Timeout=5000,10000");
```

Action

Correct your code.

31832 IBMWMQ QueueManager disconnected

Cause

This error occurs when a WebSphereMQ API call returns one of the following IBM error codes:

- MQRC_CONNECTION_BROKEN
- MQRC_CONNECTION_QUIESCING
- MQRC_CONNECTION_STOPPING

Action

Close the message queue. If necessary, re-open it after you investigate the cause of the unexpected Queue Manager disconnect.

31833 PutMessage interrupted waiting for queue space

Cause

This error occurs when the JADE process is interrupted while waiting to insert a message into a **JadeGM** queue that is full. Interruptions are usually the result of user action in the JADE Monitor (force off user or interrupt process).

Action

If the interrupt was deliberate, determine why the queue is full; otherwise, determine the cause of the interrupt.

31900 through 31999 Asynchronous Method Call Errors

This section describes error messages that can occur when making asynchronous method calls.

31900 Method and/or parameter unsupported

Cause

This is an internal error and should not normally occur.

Action

Contact your local JADE support center or JADE Support if your JADE licenses include support.

31901 Memory allocation failed

Cause

This error occurs if a memory allocation request fails.

Action

Shut down one or more applications to regain memory and then retry the request. If this problem persists, it may be necessary to add additional physical memory or reduce the cache size limits.

For details, see the **TransientCacheSizeLimit** parameter in the [\[JadeClient\]](#) or [\[JadeServer\]](#) section of your JADE initialization file, or the **ObjectCacheSizeLimit** parameter in the [\[PersistentDb\]](#) section. See also the *JADE Initialization File Reference*.

31902 Object must be process transient

Cause

This error indicates that an attempt has been made to create a non-shared transient instance of the [JadeMethodContext](#) class. The **lifetime** restrictions on the class should prevent this from happening.

Action

No action.

31903 Handle address space error

Cause

This error occurs if you create an instance of the [JadeMethodContext](#) class in a [serverExecution](#) method and attempt to use it in a [clientExecution](#) method, or the reverse.

Action

Use a [JadeMethodContext](#) instance in the node in which it is created.

31904 Handle mismatch

Cause

This is an internal error and should not normally occur.

Action

Contact your local JADE support center or JADE Support if your JADE licenses include support.

31906 MethodContext handle invalid

Cause

This error indicates some kind of memory corruption and should not normally occur.

Action

Contact your local JADE support center or JADE Support if your JADE licenses include support.

31907 Messaging module is closing down

Cause

This error should not normally occur. It could possibly occur if the node that is executing the asynchronous method calls is closing down.

Action

Wait until the node is restarted.

31908 Property is readonly

Cause

This error indicates that an attempt has been made to change the value of a read-only property of a [JadeMethodContext](#) instance (perhaps by using the [setProperty](#) of the [Object](#) class); for example, attempting to set the value of the [state](#) property.

Action

Do not attempt to change read-only properties.

31910 MethodContext not initialized

Cause

This error occurs if you attempt to set a property of a [JadeMethodContext](#) instance (for example, the [timeout](#) property) when the associated asynchronous method call has already been started by calling the [invoke](#) method.

Action

Set properties of a [JadeMethodContext](#) instance before calling the [invoke](#) method.

31914 Unsupported method parameter type

Cause

This error occurs in an asynchronous method call, using the [invoke](#) method of the [JadeMethodContext](#) class, if the method has a parameter of type [MemoryAddress](#) or another unsupported type.

In the following code fragment, the [getHistory](#) method has a parameter of type [MemoryAddress](#), which is not supported.

```
context.invoke(cust, getHistory, memAddr);
```

Action

Correct your code.

31916 Target method is null

Cause

This is an internal error and should not normally occur.

Action

Contact your local JADE support center or JADE Support if your JADE licenses include support.

31917 Invoke called with too many method parameters

Cause

This is an internal error and should not normally occur.

Action

Contact your local JADE support center or JADE Support if your JADE licenses include support.

31918 Timeout cannot be less than zero

Cause

This error occurs if the [timeout](#) property of a [JadeMethodContext](#) instance is set to a negative value. The value of this property must be greater than or equal to zero, as shown in the following code fragment.

```
create context transient;  
context.timeout := -1;
```

Action

Correct your code.

31919 Invoke receiver object is null

Cause

This error occurs in an asynchronous method call, using the [invoke](#) method of the [JadeMethodContext](#) class, if the receiver object is a null reference.

In the following code fragment, the error occurs if the **cust** object is a null reference.

```
jadeMethodContext.invoke(cust, getHistory, date);
```

Action

Correct your code.

31920 WorkerAppName is null

Cause

This error occurs in an asynchronous method call, using the [invoke](#) method of the [JadeMethodContext](#) class, if the worker application associated with the context object has not been specified.

In the following code fragment, the instruction specifying the [workerAppName](#) property has been commented out to raise the exception.

```
// Create the JadeMethodContext instance
create context transient;
//context.workerAppName := "Server";
context.invoke(cust, getHistory, date);
```

Action

Correct your code.

31923 Worker application is not running

Cause

This error occurs if a copy of the application specified by the [workerAppName](#) property of the [JadeMethodContext](#) instance is not running or did not execute the [asyncInitialize](#) method of the [Application](#) class when it was initialized.

Action

Correct your code.

31924 Too many active methodContexts in wait list

Cause

This error occurs if there are already 64 active [JadeMethodContext](#) instances. An active method context is one for which the [invoke](#) method has been called but has not completed.

A method context is completed when it is returned as the return value from the [waitForMethods](#) method on the [Process](#) class.

Action

Wait until some of the active method contexts have completed before calling the [invoke](#) method again.

31926 Call not invoked

Cause

This error occurs if the [getReturnValue](#) method of the [JadeMethodContext](#) class is called before the [invoke](#) method has been called, as shown in the following code fragment.

```
create context transient;  
write context.getReturnValue.String;
```

Action

Correct your code.

31927 Call inprogress

Cause

This error occurs if the [getReturnValue](#) method of the [JadeMethodContext](#) class is called after the [invoke](#) method has been called but before the method context has completed and been returned by the [waitForMethods](#) method on the [Process](#) class, as shown in the following code fragment.

```
create context transient;  
context.invoke(cust, getHistory, date);  
write context.getReturnValue.String;
```

Action

Correct your code.

31928 Call timedout

Cause

This error occurs if an asynchronous method call times out before it completes. In the following example, the [getErrorNumber](#) method is called on a [JadeMethodContext](#) instance that has timed out.

```
context.timeout := 1; // call times out after 1 millisecond  
context.invoke(cust, getHistory, date);  
completedContext :=process.waitForMethods(context);  
write completedContext.getReturnValue.String;
```

Action

You can use the [isTimedOut](#) method of the [JadeMethodContext](#) class to test for this situation.

31932 Target method has no return type

Cause

This error occurs if you attempt to get the return value from an asynchronous method call, by using the `getReturnValue` method of the `JadeMethodContext` class, but the method being called does not return a value.

In the following code fragment, an exception is raised if the method call associated with the `completedContext` instance does not return a value.

```
// Obtain context of last asynchronous call to complete
completedContext := process.waitForMethods(context);
// Check how things went and obtain the result
write "Return value: " & completedContext.getReturnValue.String;
```

Action

Correct your code.

31934 Invoke receiver object is non-shared transient

Cause

This error occurs in an asynchronous method call, using the `invoke` method of the `JadeMethodContext` class, if the receiver object is a reference to a non-shared transient object.

In the following code fragment, the error occurs if the `cust` object is a non-shared transient object.

```
jadeMethodContext.invoke(cust, getHistory, date);
```

Action

Correct your code.

31935 Method parameter is a non-shared transient object

Cause

This error occurs in an asynchronous method call, using the `invoke` method of the `JadeMethodContext` class, if an object that is used as a parameter to the method is created as a non-shared transient object.

In the following code fragment, the `paramObj` used in the asynchronous method call is incorrectly created as a transient object, which results in an exception being raised.

```
// Create the JadeMethodContext instance
create context transient;
// Create the parameter object
create paramObj transient; // should not be a process transient
//context.workerAppName := "Server";
context.invoke(cust, getHistory, paramObj);
```

Action

Correct your code.

32500 through 32550 X509 Certificate Errors

This section describes error messages that can occur when using X509 certificates.

32500 X509Certificate memory allocation failed

Cause

This error occurs if the machine ran out of memory and virtual memory.

Action

Shut down any applications that you do not require. Increase memory or virtual memory size, or increase both memory and virtual memory size, if required.

32501 X509Certificate attribute is ReadOnly

Cause

This error occurs if an attempt was made to modify a read-only property of an X509 certificate.

Action

Modify your code so that it does not attempt to modify any read-only properties.

32503 X509Certificate invalid certificate

Cause

This error occurs if a certificate was used with an unknown contents type.

Action

Use X509 certificates in PEM-format.

32505 X509Certificate invalid Certificate file name or contents

Cause

This error occurs if the specified certificate file contains a certificate of unknown format.

Action

Use X509 certificates in PEM-format.

32506 X509Certificate invalid PrivateKey file name or contents

Cause

This error occurs if the specified private key file contains a private key of unknown format.

Action

Use X509 private keys in PEM-format.

32507 X509Certificate private and public key mismatch

Cause

This error occurs if a pair of private keys and public keys (issued as a pair) was supplied that did not match.

Action

Supply matching private and public keys.

32511 X509Certificate invalid privatekey

Cause

This error occurs if a private key of unknown format was used.

Action

Provide a correctly formatted private key.

32512 X509Certificate invalid certificate data

Cause

This error occurs if a certificate with unknown contents was used.

Action

Provide a certificate with correct contents.

32513 X509Certificate invalid privatekey data

Cause

This error occurs if a private key with unknown contents was used.

Action

Provide a private key with correct contents.

32515 X509Certificate file read operation failed

Cause

This error occurs if a file read operation failed.

Action

Ensure that the file path exists and that it is accessible.

32518 X509Certificate failed to open specified file

Cause

This error occurs if the specified file was not found or it could not be accessed.

Action

Correct the file name or change the security settings.

35000 through 35600 Inter-Node Communications Errors

This section describes error messages that can occur when using the Inter Node Communications (INC) module for communications in the Synchronized Database Service (SDS) environment.

These errors are normally logged to the JADE messages log file (**jommsg.log**).

35000 INC - no memory

Cause

This error occurs if there is insufficient memory to establish an Inter Node Communications (INC) connection.

Action

Perform one or more of the following actions.

- Shut down other applications that are not required
- Install more memory
- Increase virtual memory

35003 INC - listener setup failed

Cause

This error occurs if the Inter Node Communications (INC) connection listen function fails.

Action

Check the **NetworkSpecification<n>** parameter in the [[ConnectionParams.<name>](#)] section of the JADE initialization file on the primary server. If necessary, check the JADE messages log files (**jommsg.log**) for a more-specific message or error number and then contact your network administrator.

35004 INC - failed to establish network connection

Cause

This error occurs if a network connection could not be opened.

Action

Check the following:

- The primary JADE Synchronized Database Service (SDS) server node is running
- Network parameters on the secondary and the primary servers are the same

Ensure that the **ServerNodeSpecifications** parameter in the [\[ConnectionParams.<name>\]](#) section of the JADE initialization file is a valid host name (or alias) for the primary server workstation and that when using TCP/IP, the host name to Internet Protocol (IP) address mapping is correctly defined in your network configuration (local host file, Domain Name Service (DNS) or Windows Internet Naming Services (WINS)). See your network administrator if you are unsure about this.

35005 INC - no listener network was specified

Cause

This error occurs if the primary server JADE initialization file does not include at least one network specification.

Action

Check the **NetworkSpecification<n>** parameter in the [\[ConnectionParams.<name>\]](#) section of the JADE initialization file on the primary server to ensure that at least one network is specified.

35008 INC - invalid or unsupported network type

Cause

This error occurs on the primary or secondary server if an invalid or unsupported network type is specified.

Action

Check the network specifications in the [\[ConnectionParams.<name>\]](#) section of the JADE initialization file. A common error is the misspelling of **Tcplp**.

35009 INC - invalid network specification

Cause

This error occurs if you specify a network specification label for a secondary server that cannot be found in the JADE initialization file.

Action

Correct the network specification label or create an entry in the JADE initialization file for this label.

35010 INC - invalid method call for this network type

Cause

This error occurs if an invalid method call for the network type occurs in the Inter Node Communications (INC) module.

Action

Check your JADE messages (**jommmsg.log**) log file for more details. If this error persists and you are unable to resolve it by other means, contact your local JADE support center or JADE Support if your JADE licenses include support.

35011 INC - invalid state for operation

Cause

This error occurs if the requested operation is not valid for the current state of the network object.

Action

Check your JADE messages (**jommsg.log**) log file for more details. If your JADE licenses include support, contact your local JADE support center or JADE Support.

35012 INC - network initialize failed

Cause

This error occurs if the Inter Node Communications (INC) network failed to initialize.

Action

Check your JADE messages (**jommsg.log**) log file for more details. If this error persists and you are unable to resolve it by other means, contact your local JADE support center or JADE Support if your JADE licenses include support.

35014 INC - protocol version mismatch

Cause

This error occurs if there is an attempt to connect using incompatible versions of the Inter Node Communications (INC) module.

Action

Use compatible protocol versions.

35015 INC - invalid handle

Cause

This error occurs if the contents of the handle pointer are invalid.

Action

Check your JADE messages (**jommsg.log**) log file for more details. If this error persists and you are unable to resolve it by other means, contact your local JADE support center or JADE Support if your JADE licenses include support.

35016 INC - communication error

Cause

This error occurs if there is a communication error in the Inter Node Communications (INC) network.

Action

Check your JADE messages (**jommsg.log**) log file for more details. If this error persists and you are unable to resolve it by other means, contact your local JADE support center or JADE Support if your JADE licenses include support.

35017 INC - remote host disconnected

Cause

This error occurs if the network connection has been closed by the remote host.

Action

Terminate the application.

35018 INC - invalid value for minimum channels

Cause

This error occurs if the specified number of logical channels for the connection initialize function is invalid.

Action

Check your JADE messages (**jommsg.log**) log file for more details. If this error persists and you are unable to resolve it by other means, contact your local JADE support center or JADE Support if your JADE licenses include support.

35019 INC - remote requested finalize

Cause

This error occurs if the Inter Node Communications (INC) finalize function is requested by the remote server.

Action

Check your JADE messages (**jommsg.log**) log file for more details. If this error persists and you are unable to resolve it by other means, contact your local JADE support center or JADE Support if your JADE licenses include support.

35020 INC - invalid channel option

Cause

This error occurs if an invalid channel option is specified. This is an internal error.

Action

Check your JADE messages (**jommsg.log**) log file for more details. If your JADE licenses include support, contact your local JADE support center or JADE Support.

35021 INC - access denied

Cause

This error occurs if Inter Node Communications (INC) network access is denied, due to the value of the **RestrictedHostAccess** parameter specified in the [ConnectionParams.<name>] section of the JADE initialization file for a primary server. When this parameter is set to **true**, the primary server will refuse connections from any host that is not included in the **AllowedHost** list.

Action

Check the relevant parameters in the [ConnectionParams.<name>] section of the JADE initialization file.

Check your JADE messages (**jommsg.log**) log file for more details. If this error persists and you are unable to resolve it by other means, contact your local JADE support center or JADE Support if your JADE licenses include support.

35023 INC - node compatibility mismatch

Cause

This error occurs if the primary and secondary nodes are running binaries for incompatible architecture styles; for example, Intel 32-bit and Intel 64-bit.

Action

Change the binaries of one of the nodes to a set that is compatible with the other node.

35503 FileTransfer - data corrupted (MD5) during transfer

Cause

This error occurs if data corruption problems are encountered during the file transfer from the database to the node to automatically upgrade the JADE software on the node.

Action

Contact your local JADE support center or JADE Support if your JADE licenses include support.

35504 Jadeadmin - auto upgrade found a protocol error

Cause

This error occurs if an unexpected situation is encountered in sending internal messages between the **jadeadmin** utility and the database.

Action

If your JADE licenses include support, contact your local JADE support center or JADE Support attaching the JADE messages (**jommsg.log**) log files from the **jadeadmin** and database nodes.