



Artix™

Release Notes
Version 3.0, October 2005

IONA Technologies PLC and/or its subsidiaries may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this publication. Except as expressly provided in any written license agreement from IONA Technologies PLC, the furnishing of this publication does not give you any license to these patents, trademarks, copyrights, or other intellectual property. Any rights not expressly granted herein are reserved.

IONA, IONA Technologies, the IONA logo, Orbix, Orbix Mainframe, Orbix Connect, Artix, Artix Mainframe, Artix Mainframe Developer, Mobile Orchestrator, Orbix/E, Orbacus, Enterprise Integrator, Adaptive Runtime Technology, and Making Software Work Together are trademarks or registered trademarks of IONA Technologies PLC and/or its subsidiaries.

Java and J2EE are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. CORBA is a trademark or registered trademark of the Object Management Group, Inc. in the United States and other countries. All other trademarks that appear herein are the property of their respective owners.

IONA Technologies PLC makes no warranty of any kind to this material including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. IONA Technologies PLC shall not be liable for errors contained herein, or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

“While the information in this publication is believed to be accurate, IONA Technologies PLC makes no warranty of any kind to this material including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. IONA shall not be liable for errors contained herein, or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

COPYRIGHT NOTICE

No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, photocopying, recording or otherwise, without prior written consent of IONA Technologies PLC. No third-party intellectual property right liability is assumed with respect to the use of the information contained herein. IONA Technologies PLC assumes no responsibility for errors or omissions contained in this publication. This publication and features described herein are subject to change without notice.

Copyright © 1999-2006 IONA Technologies PLC. All rights reserved.

All products or services mentioned in this publication are covered by the trademarks, service marks, or product names as designated by the companies that market those products.”

Updated: 30-Nov-2006

Contents

All Artix 3.0 Releases	1
Upgrading from Previous Versions	1
Licenses	1
Java Development	1
C++ Development	2
Java and C++ Development	3
Artix Services	4
Artix Security	4
Artix Configuration	5
Artix Designer	5
Known Issues in All 3.0 Releases	5
Installation	6
Type Support	7
xsd:union support	7
Demos	8
Code Generation	9
Tuxedo Plug-in	9
Java	10
Security	10
Transformer	12
J2EE Connector	12
Logging Restrictions	12
Artix Designer	13
Reporting Problems	13
Other Resources	13
Artix 3.0.4	15
New Features in Artix 3.0.4	15
JMS Logging	15
Fixed Bugs in Artix 3.0.4	15
Artix 3.0.3	18
New Features in Artix 3.0.3	18

WSDL Publishing	18
Artix Routing	19
JMS Transport	19
Artix Java	20
Artix for .NET	21
Artix Container	22
Documentation Updates in Artix 3.0.3	22
Artix Configuration Guide	22
Developing Artix Applications in C++	22
Developing Artix Applications in Java	23
Deploying and Managing Artix Solutions	23
Known issues in Artix 3.0.3	23
Artix Security	23
Artix Designer	25
Fixed Bugs in Artix 3.0.3	25
Artix 3.0.2	28
New Features in Artix 3.0.2	28
Artix Locator	28
JCA Transactions	29
Reliable Message Operations	29
Command-Line Tools	29
JMS Transport	29
Performance Enhancements	29
Java API for High Availability	30
Logging per Bus	30
Artix Container	30
Artix Designer	30
Transaction Propagation Across Dissimilar Middleware	31
Java Transactions	31
Documentation Updates in Artix 3.0.2	31
Getting Started with Artix	31
Designing Artix Solutions	31
Developing Artix Applications in C++	32
Developing Artix Plug-Ins with C++	32
Developing Artix Applications in Java	32
Artix for J2EE	32
Artix for CORBA	33
Deploying and Managing Artix Solutions	33

Artix Configuration Guide	33
Artix Security Guide	34
Artix Primer	34
Known Issues in Artix 3.0.2	34
Binary compatibility	34
Compiling on Red Hat Linux	35
64-Bit Linux	35
J2EE Connector	36
Circular Dependency Between it_afc and it_bus libraries	36
Artix Designer	36
Demos	37
Fixed Bugs in Artix 3.0.2	38
Artix 3.0.1	42
New Features in Artix 3.0.1	42
J2EE Connector	43
Code Generation	43
Artix Designer	43
Artix Java	43
SOAP Support	44
Type Support	44
Internationalization	44
Logging	44
Transports	44
Artix Container	45
High Availability	45
Service Shutdown	45
Artix Locator	45
Finding WSDL Contracts and References	46
UDDI Support	46
CA WSDM Support	46
Command Line Tools	46
JMS Transport	47
HTTP Transport	47
User Credential Propagation from SOAP to CORBA Bindings	48
Other	48
Documentation Updates in Artix 3.0.1	48
Getting Started with Artix	49
Designing Artix Solutions	49

CONTENTS

Developing Artix Applications in C++	49
Developing Artix Plug-ins with C++	49
Developing Artix Applications in Java	49
Artix for J2EE	50
Artix for CORBA	50
Deploying and Managing Artix Solutions	50
Artix Configuration Reference	50
Artix Technical Use Cases	50
Learning About Artix	50
Known Issues in Artix 3.0.1	50
Type Support	51
Threading Model Configurations	51
SOAPScope	52
Java Custom SOAP Headers	52
HTTP	52
Imported Schema Serialization	53
Demos	53
Missing Demos and Documentation	54
Artix Designer	54
Fixed Bugs in Artix 3.0.1	55

All Artix 3.0 Releases

In these release notes

These release notes contain the following sections:

Upgrading from Previous Versions	page 1
Known Issues in All 3.0 Releases	page 5
Reporting Problems	page 13
Other Resources	page 13

Upgrading from Previous Versions

When upgrading from a previous version of Artix, bear the following in mind.

- [Licenses](#)
- [Java Development](#)
- [C++ Development](#)
- [Java and C++ Development](#)
- [Artix Services](#)
- [Artix Security](#)
- [Artix Configuration](#)
- [Artix Designer](#)

Licenses

Acquire and install new licenses. If you have not received your new license, please contact your IONA representative.

Java Development

The following changes have been made to Artix Java. See [Developing Artix Applications in Java](#) for more details:

Anonymous types The Artix Java code generator now generates anonymous types as in-line sub-classes.

User-defined faults The Java code generator has changed how it names the generated class for user-defined faults. If a user-defined complex type is used as a fault message, a class called `typeName_Exception.java` is generated to support the throwable exception.

Observer interface removed The `Observer` interface used for Java instrumentation has been replaced with the JMX compliant `Notification` interface.

Transport APIs All Java transport APIs now use streams instead of `ByteBuffers`.

HandlerFactory API The Java `HandlerFactory` API now provides `HandlerInfo` as a parameter onto which the user can set the correct handler `className` and any additional configuration information.

References API In the Java `references` API, the `com.ionaschemas.references.Port` class has been renamed to `com.ionaschemas.references.ReferencePort`.

C++ Development

The following changes have been made to Artix C++. See [Developing Artix Applications in C++](#) for more details:

Servant registration The `register_servant()` function no longer returns a `Service`.

New methods In the `ServerMessageInterceptor` API, `intercept_dispatch()` has been replaced by two new functions.

```
virtual void message_received(BinaryBuffer& in_message,
                             DispatchInfo& dispatch_context);
virtual void send_message(BinaryBuffer& out_message,
                          DispatchInfo& dispatch_context);
```

Accessing the Port object Due to a change in the Artix servant implementation, user code can no longer get the `Port` object directly from within the generated server implementation object. Previously users required access to the `Port` object to access contextual message data. This feature is now supported using the `Context` API.

Changes to registry interface The implementation of the `registry` interface has been changed for Artix 3.0. Any user code that makes use of it needs to be updated to match the new implementation. In particular, the return type of `register_entry()` and `deregister_entry()` have been changed to `bool`. Also, the operations are no longer defined as throwable.

Changes to reference API In the `reference` API, the following changes have been made to the `IT_Bus::ReferencePort` class:

- The return type from `get_name()` has changed from `IT_Bus::String` to `IT_Bus::NCName`.
- The `get_properties()` function has been renamed to `get_any()` and the return type has changed from `IT_Bus::ElementListTAnyHolder` to `IT_Bus::AnyList`.

Java and C++ Development

The following changes apply to both C++ and Java development:

Current API removed The `Current` API has been removed from both C++ and Java.

This was used in Artix server code to access the `ServerOperation` that was in progress at that time. Users can now access this contextual information through the `Context` API. The code used to access the `ServerOperation` in Artix 3.0 is shown below.

```
ContextRegistry* context_registry =
    get_bus()->get_context_registry();
ContextCurrent& context_current =
    context_registry->get_current();
ContextContainer* context_container =
    context_current.request_contexts();

ServerOperation * operation = 0;
Context& context_data =
    context_container->get_context (SERVER_OPERATION_CONTEXT,
                                   true);
ServerOperationContext* operation =
    IT_DYNAMIC_CAST (ServerOperationContext*, context_data);
ServerOperation& server_op = operation->get_data();
```

Custom transport implementations Due to updates in the server transport implementation, custom transport implementations need to be updated. For more information, see either [Developing Artix Plug-ins in C++](#) or [Developing Artix Applications in Java](#).

RegistryException removed `RegistryException` has been removed from Artix. Any user code that catches this exception needs to be updated.

Artix Services

The following changes have been made to Artix services:

Container service The `itartix_service` has been replaced with a new container-based service. This change impacts how all Artix services are configured. For detailed information on the impact on your Artix configuration, see [Deploying and Managing Artix Solutions](#).

Finding WSDL contracts and references The mechanism that Artix uses to find WSDL contracts and references has changed. For detailed information on the impact on your Artix configuration, see [Deploying and Managing Artix Solutions](#).

New bus_loader plug-in CORBA applications that load Artix plug-ins need to add a new plug-in called `bus_loader` to their `orb_plugins` list. This plug-in loads the Artix bus in applications that do not call `Bus::init()`.

If you are using the `itartix_service`, add the `bus_loader` plug-in to its `orb_plugins` list.

Artix Security

Security certificates Due to changes in the Artix HTTPS implementation, Artix no longer accepts certificates in the `pem` format. You need to convert your certificates to the `pkcs12` format. The standard way of doing this is shown below.

```
cat cacert.pem keycert.pem privkey.pem > SomeName.pem
openssl pkcs12 -export -in SomeName.pem -out
    SomeOtherName.p12
```

Certificate chain length The X.509 certificate chain length is now limited to 2 by default. Previously, for the HTTPS protocol in Artix, there was no chain length limit. If any of your certificate chains have a length greater than 2, you will need to override the default limit by setting the following configuration variable in your Artix configuration:

```
# Artix Configuration File
policies:max_chain_length_policy = "ChainLength";
```

OpenSSL libraries removed Artix 3.0 no longer includes the OpenSSL libraries. The only SSL libraries included with Artix 3.0 are the Baltimore libraries.

Artix Configuration

The following changes have been made to Artix configuration. For more details, see the [Artix Configuration Guide](#):

Custom interceptors The configuration for using custom interceptors has been changed from the format used in Artix 2.1. For more information, see [Deploying and Managing Artix Solutions](#).

Refactored references The references passed by Artix have been refactored in 3.0. The new references are incompatible with older implementations. If you require backward compatibility you can set `bus:reference_2.1_compat = "true";` in your `artix.cfg` file.

Artix Designer

Due to the redesign of Artix Designer, you need to create your pre-Artix 3.0 workspaces using the new Eclipse-based Artix Designer.

Known Issues in All 3.0 Releases

For known issues in specific releases, see the relevant notes on each release.

The following are known issues in all Artix 3.0 releases:

- [Installation](#)
- [Type Support](#)
- [xsd:union support](#)
- [Demos](#)

- [Code Generation](#)
- [Tuxedo Plug-in](#)
- [Java](#)
- [Security](#)
- [Transformer](#)
- [J2EE Connector](#)
- [Logging Restrictions](#)
- [Artix Designer](#)

Installation

The following are known issues with the installation of Artix 3.0.

Previous versions Artix 3.0 cannot be installed in the same directory tree as an Artix 1.x installation or Artix 2.x installation. We recommend that you completely remove any previous Artix installations from your system before installing Artix 3.0.

Installing on Windows When installing Artix on Windows platforms, the setting of `PATH` and `CLASSPATH` in `artix_env.bat` and the demo build scripts will be incorrect if you install into a folder that has a space in the pathname (for example, `C:\Program Files\IONA`).

Uninstalling on Windows The Artix uninstaller, like most Windows uninstallers, does not remove files that its companion installer did not place on the disk. This means that files generated after the initial installation are not removed by the uninstaller.

The Artix uninstaller reports the failure to remove some directories and files, but does not report others. The directories and files usually left in place by the Artix uninstaller include the following:

- `InstallDir\artix\3.0\bin`, containing `artix_comp.bat` generated by `artix_env.bat`
- `InstallDir\artix\3.0\etc`, containing `artix_cxx.mk.bak`
- `InstallDir\artix\3.0\demos*`, containing any files generated when compiling the Java and C++ demo code

Installing and uninstalling on Windows Server 2003 When installing or uninstalling Artix on Windows Server 2003, run the installer or uninstaller program in XP compatibility mode.

Also, if you receive an error saying “The memory could not be written”, check your Data Execution Prevention settings, as follows:

1. Select **Start|Control Panel|System**.
2. Click the **Advanced** tab.
3. In the Performance section, click **Settings**.
4. In the Performance Options dialog box, click the **Data Execution Prevention** tab.
5. If **Turn on DEP for all programs and services except those I select** is selected, click **Add** to add the Artix installer (`artix.exe`) to the list.

Environment variables on Windows 2000 and Windows 2003 Server When the installer sets the path environment variable for the current user, it also appends the value of the system path environment variable.

In addition, if the system path entries are removed from the user path prior to uninstalling, the uninstaller fails with an alert dialog with the following text: “The uninstallation could not complete due to an error.”

Type Support

The following are known issues in Artix type support.

Floats and doubles Due to platform-specific limits, floats and doubles are limited to the range supported by `FLT_MIN-FLT_MAX` and `DBL_MIN-DBL_MAX`, as defined by the C++ compiler in `float.h`. This range may not completely reflect the range required by XML Schema. This also affects the Java runtime as it is also limited by range of the C++ runtime.

Tibco binding and `xsd:dateTime` When reading an `xsd:dateTime` using the Tibco binding, Artix creates a 0 UTC time zone offset instead of a local time. In addition, Artix generates a warning if a local time is specified.

`xsd:union` support

Unions are not supported in the CORBA transport, the Tib/RV transport, or the Transformer.

Demos

The following are known issues with the demos included with all Artix 3.0 releases (*InstallDir/artix/3.0/demos*).

Located router The start-up time for the located router demo (*advanced/located_router*) may be slow. Users should wait until the following lines appear in the command window.

```
IONA Artix container server
IONA Artix container server ready
```

JMS On HP-UX, the JMS demos require loading the `jvm_manager` plug-in, which depends on the `libjvm.sl` JRE library. Loading `libjvm.sl` is problematic for the following reason documented on HP's website:

http://www.hp.com/products1/unix/java/java2/sdkrt14/infolibrary/sdk_rnotes_1.4.2.02.html#libjvm

After updating to the proper patch, you can run the JMS demos by calling

```
export LD_PRELOAD=jre_lib/lib/PA_RISC/server/libjvm.sl
```

When running the JMS demos, the following statement appears:

```
(IT_CORE:1) E - could not load plug-in jms: Using configuration scope
of "demos.oneway", configuration variable
"plugins:jms:shlib_name" is not set or is set to an empty string. Use
"-ORBname" to specify appropriate configuration, if necessary.
```

This statement can be ignored.

Tuxedo With Tuxedo 6.5 the directory paths used in the Tuxedo demos are too long. You must shorten the directory path by relocating the demo directory.

Tagged to Tuxedo On Unix, to get the tagged binding over Tuxedo routing demo (*routing/tagged_tuxedo_soap_http*) to work, you must run the following commands:

```
export IT_DOMAIN_NAME=tagged_tuxedo_soap_http
export IT_CONFIG_DOMAINS_DIR=../../etc
```

SOAP over MQ On AIX, to run the SOAP over MQ demo (*transports/soap_over_mq*), set `export EXTSHM=ON` before running the demo.

MQ to CORBA On AIX, to run the Java client of the MQ to CORBA routing demo (`routing/soap_mq_corba`), set `EXTSHM=ON` before running the client.

SOAP over IIOP tunnel The SOAP over IIOP tunnel demo (`transports/soap_over_iiop_tunnel`) only works with the advanced version of Artix. It does not work if you are using a standard license.

iSF cluster security The iSF cluster demo (`security/isf_cluster`) only demonstrates iSF federation features.

Code Generation

There are the following issues with code generation.

Batch files on UNIX The code generator produces Windows batch files to start the Artix router on UNIX platforms.

To overcome this problem, replace the five lines of script beneath the `#!/bin/sh` directive with the following:

```
source ./artix_env
```

This will cause the script to correctly source the locally generated Artix environment script.

IDL to WSDL on UNIX There are some issues using the CORBA IDL to WSDL functionality on UNIX platforms.

On some occasions the IDL compiler may not run correctly to convert the IDL to WSDL. This typically happens when the physical disk that Eclipse is installed on one physical disk and `/tmp` is on another.

To work around this, use the `corbatowsdl` command and then CORBA-enable this WSDL from within Artix Designer.

Tuxedo Plug-in

Tuxedo does not allow you to have a space in the path name of any path referenced in the `tux.env` script.

You need to use the short file name format for Windows environments. You can display the short filenames using `dir /x`.

Java

The following are known issues with Artix Java.

THREAD-LOCAL `THREAD-LOCAL` should not be used when developing Artix Java server applications.

Xerces/Xalan conflict on AIX On AIX, there is a possible conflict between the version of `xerces` and `xalan` in the AIX JDK and the version that ships with Artix. You may need to add the following flags when running Artix Java applications:

- `$JDK_ENDORSED_DIRS`
- `$JDK_BOOTSTRAP_CLASSPATH`

WSDL generation Artix does not support the generation of WSDL from Java operations that return an array of a base type like `int`, `long`, or `String`.

C++ interceptors Artix Java applications do not support the use of C++ interceptors built using Visual C++ v7.1.

log4j error A `log4j` initialization error is occasionally displayed when Artix Java applications are started. You can ignore this message. All logging systems will work as expected.

Security

The following are known issues with Artix security.

Settings in port element ignored Due to continuing upgrades to the Artix HTTPS implementation, the HTTPS transport ignores any settings made in the WSDL `port` element.

Messages in HP-UX During startup, the security server prints the following messages on HP-UX:

```
/usr/lib/dld.sl: Unresolved symbol:
  oop_iterate__7oopDescFP10OpClosure (code) from
  /install_dir/jre/lib/PA_RISC/server/libjvm.sl
/usr/lib/dld.sl: Unresolved symbol:
  _adjust_pointer__9MarkSweepSFPP7oopDescb (code) from
  /install_dir/jre/lib/PA_RISC/server/libjvm.sl
/usr/lib/dld.sl: Unresolved symbol:
  do_oop_nv__16FilteringClosureFPP7oopDesc (code) from
  /hp/install_dir/jre/lib/PA_RISC/server/libjvm.sl
/usr/lib/dld.sl: Unresolved symbol:
  oop_iterate__7oopDescFP10OpClosure (code) from
  /install_dir/jre/lib/PA_RISC/server/libjvm.sl
/usr/lib/dld.sl: Unresolved symbol:
  _adjust_pointer__9MarkSweepSFPP7oopDescb (code) from
  /install_dir/jre/lib/PA_RISC/server/libjvm.sl
/usr/lib/dld.sl: Unresolved symbol:
  do_oop_nv__16FilteringClosureFPP7oopDesc (code) from
  /install_dir/jre/lib/PA_RISC/server/libjvm.sl
```

The messages occur due to dependency on the classic JVM, which is deprecated in JDK1.4.x, and can be safely ignored.

Log file message Using the Artix security plug-in causes the following message to be displayed in your log file:

```
SecurityPLugin, failed to register security context,
  reason:ContextException
```

You can safely ignore this message.

Secure Artix Configuration On Windows systems, the secure Artix configuration file, *InstallDir\artix\3.0\ext\domains\artix-secure.cfg* may contain entries for the property `plugins:is2_authorization:action_role_mapping` that have spaces.

The spaces must be replaced by `%20`. For example, change the entry

```
plugins:is2_authorization:action_role_mapping="file://C:\Program
  Files\IONA\artix\3.0\demos\security\full_security\etc";
```

to

```
plugins:is2_authorization:action_role_mapping="file://C:\Program%20
Files\IONA\artix\3.0\demos\security\full_security\etc";
```

Transformer

The transformer does not support complex types derived from other complex types.

J2EE Connector

The following are known issues with the J2EE Connector inbound functionality.

WebSphere When using the inbound functionality with WebSphere, the resource adapter is not able to start up unless a connection factory is explicitly created by another application.

WebLogic The inbound functionality with WebLogic 8.1 SP3 on Windows platforms requires the use of Sun's JDK or JRockit 1.4.2_05.

JRockit 1.4.2_05 is available to download from BEA via the following link:
http://commerce.bea.com/products/weblogicjrockit/other_versions.jsp

JRockit 1.4.2_04, which ships with WebLogic 8.1 SP3, has a known issue (CR ID:172574) that precludes its use.

Logging Restrictions

Log buffering has an effect on application code. Application code must invoke `Bus::shutdown` to flush the buffer to written log. However, server mainlines do not include termination handlers and the `Bus::shutdown` call is not executed.

The default configuration in `artix.cfg` suppresses buffered logging. If you want to employ buffered logging, set the value of the `plugins:xmlfile_log_stream:buffer_file` to `true`.

Be sure to add `Bus::shutdown` to the client and server mainline applications.

During shutdown, the Bus drops some messages because the logging subsystem is shut down before the other Bus plug-ins.

Artix Designer

Using Artix Generator to generate C++ does not preserve existing code modifications.

To investigate the C++ merge facility, go to **Window | Preferences | Artix Designer | C++** and select the **Merge generated code** radio button.

Reporting Problems

Contact customer support at <http://www.iona.com/support/contact/>.

Other Resources

If you need further help please use the following resources:

- [Artix TechZone](http://www.iona.com/devcenter/artix) (<http://www.iona.com/devcenter/artix>) is a free online forum where IONA developers, your peers and other professionals come to share tips on Artix Web Services development. Visit the Artix TechZone today to start making the most of your Artix development experience.
- [IONA University](http://www.iona.com/info/services/ps/) (<http://www.iona.com/info/services/ps/>) delivers practical and insightful courses that cover technical and product issues as well as standards-based best practices gleaned from real-world projects.
- [IONA Professional Services](http://www.iona.com/info/services/consulting/) (<http://www.iona.com/info/services/consulting/>) provide product expertise and consulting solutions that empower end-users, system integrators and software vendors with the knowledge to fully leverage IONA products. Together, IONA consultants and products equip you with a single platform for integrating and developing extremely reliable, scalable, and secure e-Business systems.
- [IONA Security Mailing List](mailto:security-alert@iona.com) (security-alert@iona.com): The mailing list provides security updates associated with all IONA products. To receive security updates from IONA send mail to listserver@iona.com with no subject and the body text `subscribe security-alert youremail`.

Note: Please do not post queries to this e-mail alias; it has been set up only to notify you of security alerts.

- [Online Documentation](http://www.iona.com/support/docs/index.xml) (<http://www.iona.com/support/docs/index.xml>): The latest updates to the Artix documentation are posted on-line.

[Knowledge base articles](http://www.iona.com/support/index.xml) (<http://www.iona.com/support/index.xml>): A database that contains practical advice on specific development issues, contributed by IONA developers, support specialists, and customers.

Artix 3.0.4

In these release notes

The Artix 3.0.4 release notes contain the following sections:

New Features in Artix 3.0.4	page 15
Fixed Bugs in Artix 3.0.4	page 15

New Features in Artix 3.0.4

The following new feature has been added in Artix 3.0.4:

- [JMS Logging](#)

JMS Logging

Artix 3.0.4 supports logging of JMS headers. To use this feature, set the `event_log:filters` configuration variable to `INFO_HIGH`. For example:

```
event_log:filters = ["*=FATAL+ERROR+WARNING+INFO_HI"];
```

For details on Artix logging, see the [Artix Configuration Guide](#).

Fixed Bugs in Artix 3.0.4

The following bugs have been closed in Artix 3.0.4:

Bug #	Description
70138	Tagged binding alias bug in Artix 3.0.2.
70229	If Single Sign On is enabled, the <code>BusContext</code> is not updated with the principal.
70274	<code>wsdltocorba</code> produces <code>java.lang.StackOverflowError</code> with no further error message.
70277	Allow Artix users to configure JMS temporary queue pool parameters.

Bug #	Description
70278	Artix to clean up JMS temporary queues upon timeout.
70298	Artix 3.0.2 client has problem handling non-standard SOAP fault code from third-party server.
70320	SOAP interceptor not thread-safe on client side.
70321	Artix server logs indefinitely when processing invalid SOAP request.
70339	Writing an <code>any</code> with an undefined namespace prefix causes the JVM to core dump.
70347	Artix does not support the <code>default=</code> attribute. It does not handle WSDL containing this attribute correctly for <code>enum</code> , and fails to compile the generated Java code due to incorrect code generation.
70356	Artix cannot read WSDL from Windows on a UNIX file system.
70358	Request handlers do not update output object values.
70375	IBM WebSphere shuts down gracefully due to <code>com.iona.jbus.TransportException</code> .
70390	<code>schemavalidator</code> gives false error messages.
70402	Unable to create new types or edit existing types from Artix Designer.
70404	Can not configure HTTPS information programatically on the server side.
70415	Retrofit Artix 4.0 JCA connection caching mechanism for JMS to Artix 3.0.3.
70443	SOAP messages with <code>&</code> in <code>xsd:any</code> cause JVM crash.
70446	<code>SOAPFault</code> raised by Artix Java server is wrapped by a JNI exception.
70452	<code>wSDL_publish</code> and secure ports.

Bug #	Description
70459	Artix 3.0.3 JMS transport mismatches responses if multiple concurrent clients use the same permanent queue.
70473	Artix fails to unmarshal a user exception correctly.
70487	Setting <code>authorizationRealm</code> in a WSDL file is not picked up at runtime.
70488	Setting system property inappropriately during BEA WebLogic startup.
70491	Incorrect behavior in <code>schemavalidator</code> .
70532	Artix 3.0.3 throws <code>NullPointerException</code> when MTOSI header is missing non-mandatory attribute.
70562	Supporting <code>useMessageIDasCorrelationID</code> on the client side.
70566	Problem of static order initialization.
70603	Tagged binding behaves inconsistently with nillable types.
SR 280984	Client-side Java CORBA binding.

Artix 3.0.3

In these release notes

These release notes contain the following sections:

New Features in Artix 3.0.3	page 18
Documentation Updates in Artix 3.0.3	page 22
Known issues in Artix 3.0.3	page 23
Fixed Bugs in Artix 3.0.3	page 25

New Features in Artix 3.0.3

The following new features have been added in Artix 3.0.3:

- [WSDL Publishing](#)
- [Artix Routing](#)
- [JMS Transport](#)
- [Artix Java](#)
- [Artix for .NET](#)
- [Artix Container](#)

WSDL Publishing

The following updates have been made to Artix's WSDL publishing capabilities in Artix 3.0.3:

- WSDL imports are preserved.
- When a WSDL or XML Schema file is imported from a local file system, the `wSDL_publish` plug-in inserts an HTTP URL from which the imported content can be accessed into the `location` or `schemaLocation` attribute for the imports.
- Artix specific elements can be removed from a published contract.
- Server-side elements are removed from published contracts by default.
- Artix can be configured to customize the processing of contracts before they are published.
- An HTML menu has been added.

- You can query Artix for a WSIL document containing information about active web services.
- You can query Artix for the contracts of Artix specific services.

For more information on loading the `wSDL_publish` plug-in [Developing Artix Applications in C++](#).

For details on the `wSDL_publish` configuration variables, see the [Artix Configuration Guide](#).

Artix Routing

The following changes have been made to the Artix routing plug-in:

Improved scalability Solutions that use references in the Artix routing plug-in are now more scalable. The router now handles requests on objects that share the same interface by implementing a single default servant.

The default servant processes all requests on a set of objects, while references are used to identify individual objects. References can either be proxified or unproxified.

Two new configuration variables, `plugins:routing:proxy_cache_size` and `plugins:routing:reference_cache_size`, have been added to handle the number of proxified and unproxified references on the server. For more details, see the [Artix Configuration Guide](#).

Router Bypass for CORBA Services Artix now allows you to use CORBA location forwarding to connect CORBA clients directly to CORBA servers, thus bypassing the Artix routing plug-in.

To enable router bypass for CORBA services, set the `plugins:routing:use_bypass` configuration variable to `true`. For more details see the [Artix Configuration Guide](#).

JMS Transport

The following optional attributes have been added to the `jms:address` WSDL extensor:

jndiReplyDesinationName Specifies a permanent destination for getting or sending replies. Clients place this value in the `ReplyTo` field of all request headers. On the server-side, the service uses this destination if the client did not specify a `ReplyTo` value in the request header.

Previously, clients created a temporary queue, which they referenced in the `ReplyTo` header. Servers, in turn, sent their response to the temporary queue.

connectionUserName Specifies a username for connecting to a JMS broker with security turned on.

connectionPassword Specifies a password for connecting to a JMS broker with security turned on.

Note: We do not recommend setting a username and password in the WSDL file directly other than when using the Artix router. Instead, set the username and password programmatically using the `JMSSecurityInfo` context value.

Artix Java

The following features have been added to Artix Java:

High performance Java CORBA binding You can now use an alternative Java CORBA binding, offering improved performance, in Artix server applications.

You can enable the new binding via the system properties as follows:

- By passing the command line argument
`-Djava.corba.service.enabled=true`
- By setting the following API call in the application:

```
System.setProperty("java.corba.service.enabled", "true");
```

Alternatively, if you are using multiple buses, you can enable the binding for an individual bus by passing the properties to `Bus.init()`, as follows:

```
Hashtable props = new Hashtable();
props.setProperty("java.corba.service.enabled", "true");
Bus.init(args, props);
```

While the new Java CORBA binding offers improved performance, it also has the following limitations:

- It can be enabled on the server side only.
- You cannot use the Artix C++ locator, session manager, router, or security plug-ins when the binding is enabled.
- The binding supports the following types only:
 - ◆ Primitives

- ◆ The `sequence` and all complex types
- ◆ Simple types

For an example of the new Java CORBA binding, see the CDR over IIOP demo (`demos\transports\cdr_over_iiop`).

Exception handling Artix Java's handling of CORBA system exceptions has been improved thanks to the addition of a `FaultException` class to the `com.ionajbus` package.

The class's methods can return a `FaultCategory` that corresponds to a CORBA system exception, a `FaultCompletionStatus`, and a `FaultSource`.

For more details, see [Developing Artix Applications in Java](#).

Artix for .NET

The following changes have been made to the `Bus.services.dll` library, used by Microsoft .NET clients:

WSE 2.0 compliancy The Artix .NET plug-in now complies with Microsoft Web Service Enhancements (WSE) 2.0.

This affects how a client interacts with the Artix session manager. Proxies must now inherit from the WSE 2.0 proxy class,

`Microsoft.Web.Services2.WebServicesClientProtocol`, which is generated by default when WSE 2.0 Service Pack 3 is installed.

In addition, you can no longer call `setSession()` on the proxy. Instead, you add a `SessionFilter` to the proxy `OutputFilters` pipeline as follows:

```
proxy.Pipeline.OutputFilters.Add(new
    bus.Services.SessionFilter(Session));
```

New classes The following customer filter classes have been added:

- `SessionFilter`—enables users to set a session by creating a new instance of this class and adding it to the SOAP `OutputFilters` collection.
For more details, see the .NET Session Manager demo (`demos\advanced\session_management\dotnet`).
- `KerberosFilter`—enables users to use Kerberos security by creating a new instance of this class and adding it to the SOAP `OutputFilters` collection

Removed classes The following classes have been removed:

- `SessionIdExtensionAttribute`
- `SessionManagedSoapHttpClientProtocol`

For more details see the Using Artix with .NET chapter of the [Artix Technical Use Cases](#) book.

Artix Container

You can now install an instance of the Artix container server as a Windows service.

The following command line arguments have been added to `it_container` as a result:

- `-service install`—installs container as a Windows service
- `-svcName ServiceName`—assigns a name to the Windows service
- `-displayName DisplayName`—sets the name of the service as it will appear in the Windows Services window
- `-service uninstall`—uninstalls container as a Windows service

For more details, see [Deploying and Managing Artix Solutions](#).

Documentation Updates in Artix 3.0.3

The following changes have been made to the Artix documentation for version 3.0.3:

- [Artix Configuration Guide](#)
- [Developing Artix Applications in C++](#)
- [Developing Artix Applications in Java](#)
- [Deploying and Managing Artix Solutions](#)

Artix Configuration Guide

Details on variables for the following features have been added:

- WSDL Publishing
- Optimized Router Proxification
- Router Bypass for CORBA Services

Developing Artix Applications in C++

The section on loading the `wSDL_publish` plug-in has been updated.

Developing Artix Applications in Java

A section on the new `FaultException` class has been added.

Deploying and Managing Artix Solutions

This guide now contains a section on running the container as a Windows service.

Known issues in Artix 3.0.3

The following are known issues in Artix 3.0.3:

- [Artix Security](#)
- [Artix Designer](#)

Artix Security

The following are known issues with Artix security:

Initializing client proxies from Artix references

When initializing client proxies *directly* from Artix references, C++ and Java clients will fail to connect if the server is secure and SSL configuration information is unavailable locally.

When instantiating Java and C++ client proxies directly from Artix references, SSL configuration information must come from the local Artix configuration file. Make sure this information is available at the relevant Artix configuration scope, as shown below:

```
my_application_config_scope
{
    ...
    principal_sponsor:use_principal_sponsor = "true";
    principal_sponsor:auth_method_id = "pkcs12_file";
    principal_sponsor:auth_method_data =
["filename=PathToClientPKCS#12KeyAndCertificateFile",
 "password=PrivateKeyPassword"];
    policies:trusted_ca_list_policy =
"PathToX.509TrustDatabaseFile";
    ...
};
```

For more information about SSL configuration parameters, refer to the [Artix Security Guide](#).

Secure clients must not download WSDL from remote sources

Currently, Artix does not provide any secure mechanisms for obtaining WSDL contracts remotely. Therefore, secure clients should use only a local, protected copy of the WSDL contract.

If you want a client to be secure, you *must not* download a remote WSDL contract over an insecure channel. Specifically, this security hazard affects WSDL contracts downloaded either from the WSDL publish plug-in or from an Artix container process. For example, a snooper program could intercept a HTTP packet containing the WSDL contract and modify the connection details in a WSDL port element.

The following Java example shows a problematic piece of code that would be vulnerable to this security hazard:

```
// Java

import javax.xml.namespace.QName;
import javax.xml.rpc.*;

// WARNING!
// Do not do this in a secure application!
...
SimpleService impl = null;
java.net.URL wsdlLocation = null;

wsdlLocation = new java.net.URL("http://remote_host...");

javax.xml.rpc.ServiceFactory factory = ServiceFactory.newInstance();
javax.xml.rpc.Service service = factory.createService(
    wsdlLocation,
    serviceName
);
impl = (SimpleService) service.getPort(portName,
    SimpleService.class);
```

Avoid publishing security information in WSDL contracts

WSDL publish is an Artix plug-in that serves up WSDL contracts to remote clients over HTTP. This can be used as a convenient mechanism for providing remote clients with an up-to-date copy of the WSDL contract. However, if the

WSDL contract contains any security data, the WSDL publish plug-in becomes a security hazard. Hostile clients can connect to the WSDL publish port, which is not secure, and gain access to the security data in the WSDL contract.

To avoid this security hazard, when using the WSDL publish plug-in, you *must not* include any security data in the published WSDL contracts. Put the security data into a local copy of the Artix configuration file instead. Configuring security using the Artix configuration file is described in detail in the *Artix Security Guide*.

Artix Designer

The Eclipse update mechanism does not work for Artix Designer on Windows. Until this is fixed, Windows users will be able to download future updates of Artix Designer in ZIP format from the IONA web site.

Fixed Bugs in Artix 3.0.3

The following bugs have been closed in Artix 3.0.3:

Bug #	Description
69300	Support for fully qualified hostname in SOAP and HTTP addresses
69518	<code>wSDL_publish</code> is broken for contracts that contain <code>import</code> statements
69596	<code>wSDL_publish</code> should have a setting to only publish standard WSDL
70009	The JMS transport should automatically try reconnecting when an Artix process starts up and the initial connection attempt fails
70017	Cannot catch exception with <code>wSDL_publish</code> plug-in loaded
70022	Implement <code>IT_CSI::set_received_credentials(rec_creds)</code>
70033	<code>wSDL_publish</code> can not handle the WSDL import correctly
70034	Parsing messages with one-character length string generates errors

Bug #	Description
70051	MQ transport should not call <code>MQINQ</code> on remote queue for backout count threshold attribute
70054	WSDL generation tools can not handle elements with fixed value in a <code>choice</code> complex type
70079	<code>wsdltoservice -transport corba -pst</code> flag does not work
70109	Problem routing between two <code>port</code> elements in same <code>service</code>
70126	Adding a SOAP binding using <code>wsdltosoap</code>
70130	Artix Java cannot handle escape character within <code>Any</code> object
70142	String error when <code>maxLength=1</code> with <code>wsdltojava</code>
70145	WSDL <code>import</code> puts the element into the wrong target namespace
70146	<code>com.ionajbus.ReadException</code> received at the server
70157	Artix runtime should generate more meaningful error message for WSDL bootstrapping failure
70159	Need to provide attributes for JMS broker credentials in WSDL
70165	Artix Designer needs to be enhanced so that when the Edit Port dialog is brought up, there are fields available for adding the username and password
70170	<code>wsdltorouting</code> ignores any existing value for the <code>persistence</code> property and sets it always to <code>false</code>
70182	Allow the user to set JMS <code>ReplyTo</code> header on one-way calls.
70191	Artix dumps core on exit
70195	An <code>xs:choice</code> with <code>minOccurs=0</code> will not be marshalled if empty
70196	Performance disparity between Orbix Java and Artix 3.0 Java
70198	Need functionality to selectively catch CORBA <code>TIMEOUT</code> and <code>TRANSIENT</code> exceptions from an Artix Java client

Bug #	Description
70199	Included schema with null target namespace should inherit the namespace of the “includer”
70202	SSO token received by server even though the client hadn't sent one
70209	MQ-WSAT Transaction Interposing has an undesired effect
70212	When setting <code>policies:bindings:corba:token_propagation="true"</code> in the Artix router, the SSO token will be sent on as the <code>ITTPPrincipalName</code> in an outgoing CORBA request
70216	Artix core dumps in constructor of class <code>ConnectKeyBase</code>
70225	Multi-threaded Artix client crashes when trying to invoke on CORBA server
70226	When method <code>getGepardNr</code> is called, the router throws an “Unknown Exception”
70258	HTTP plug-in failed to handle garbage data
70259	Artix Java server crashes if client exited before server could send back reply

Artix 3.0.2

In these release notes

These release notes contain the following sections:

New Features in Artix 3.0.2	page 28
Documentation Updates in Artix 3.0.2	page 31
Known Issues in Artix 3.0.2	page 34
Fixed Bugs in Artix 3.0.2	page 38

New Features in Artix 3.0.2

The following new features have been added in Artix 3.0.2:

- [Artix Locator](#)
- [JCA Transactions](#)
- [Reliable Message Operations](#)
- [Command-Line Tools](#)
- [JMS Transport](#)
- [Performance Enhancements](#)
- [Java API for High Availability](#)
- [Logging per Bus](#)
- [Artix Container](#)
- [Artix Designer](#)
- [Transaction Propagation Across Dissimilar Middleware](#)
- [Java Transactions](#)

Artix Locator

High availability in the Artix locator service has been improved. Support for the forwarding of write request from a slave locator to a master locator is now more robust. The forwarding mechanism is now fully automated and no longer requires any client-side coding. For more information on using the highly available locator, see [Deploying and Managing Artix Solutions](#).

JCA Transactions

The Artix J2EE Connector now supports local J2EE Connector Architecture (JCA) transactions. For more details see [Artix for J2EE](#).

Reliable Message Operations

Customers can now use Artix to reliably get or put messages onto an underlying message transport.

For more details see [Developing Artix Applications in C++](#) and [Developing Artix Applications in Java](#).

Note: This feature is restricted to use with IBM WebSphere MQ.

Command-Line Tools

The following changes have been made to the Artix command-line tools:

Consistent behavior All tools now behave consistently in the face of proper, invalid and missing parameters, and invalid or missing licenses.

New `wSDLtojava` options The following options have been added to the `wSDLtojava` command:

- `-servlet` does the same as the `-plugin` option, but it also adds two lines of code that are specific to a plug-in being run in a servlet environment.
- `-ser` allows you to generate Java types as serializable objects.

JMS Transport

The following enhancements have been made to the Artix JMS transport:

- Exceptions raised by the underlying JMS provider are returned to the code that invoked it on the Artix JMS transport
- Support for reconnection attempt if underlying JMS provider fails
- Support for JMS message acknowledgements

Performance Enhancements

Artix 3.0.2 boasts significant improvements in performance over version 3.0.1.

Latency, or the measure of time for a single invocation to cycle between a client and server, has fallen by an average of 59 percent for SOAP over HTTP across a range of client numbers.

Throughput, or the total number of messages that an Artix server can handle under a given client/message load, has almost doubled for SOAP over HTTP.

Java API for High Availability

Artix 3.0 introduced support for developing highly available Artix services. In version 3.0.1, this feature was exposed through a C++ API. In Artix 3.0.2, a Java API has been added for high availability support. See [Developing Artix Applications in Java](#) for more details.

Logging per Bus

Logging is now configurable per bus.

Artix Container

The Artix container service now retains persistent state information about the services that have been deployed within it. This enables the container to reload services dynamically upon restart. See [Deploying and Managing Artix Solutions](#) for more details.

Artix Designer

The following enhancements have been made to Artix Designer:

CDT Support for the Eclipse C/C++ Development Tools (CDT), allowing users to compile and debug their C++ code from within the Eclipse environment.

WSDL Editor The following features have been added to the WSDL Editor

- Support for additional schema validation
- improved validation when deleting WSDL elements
- Undo/redo support

Cheat sheets Cheat sheets have been added demonstrating how to create an Artix Designer project, create a WSDL file and generate a client application.

Welcome page An improved Welcome page links users to Artix tutorials, release notes, documentation, and online support

Transaction Propagation Across Dissimilar Middleware

See [Developing Artix Applications in C++](#) for more details.

Java Transactions

Java transactions are now supported in Artix 3.0.2.

Documentation Updates in Artix 3.0.2

The following changes have been made to the Artix documentation for version 3.0.2:

- [Getting Started with Artix](#)
- [Designing Artix Solutions](#)
- [Developing Artix Applications in C++](#)
- [Developing Artix Plug-Ins with C++](#)
- [Developing Artix Applications in Java](#)
- [Artix for J2EE](#)
- [Artix for CORBA](#)
- [Deploying and Managing Artix Solutions](#)
- [Artix Configuration Guide](#)
- [Artix Security Guide](#)
- [Artix Primer](#)

Getting Started with Artix

The book has been updated to reflect Artix Designer's integration with the Eclipse CDT.

Designing Artix Solutions

This has been updated to include details on the following:

- JMS reliable messaging
- Changes to the FML and Tibco bindings
- Artix Designer's integration with the Eclipse CDT

Developing Artix Applications in C++

This book has been updated to include information on the following:

- Enabling write request forwarding in persistent maps
- Reliable message operations (MQ transactions)
- Support for restriction of complexContent
- Transaction propagation across dissimilar middleware

Developing Artix Plug-Ins with C++

This book has been updated to include the following:

- Expanded discussion of transport plug-ins
- API for logging per bus

Developing Artix Applications in Java

This book now includes information on the following:

- Java Management eXtensions (JMX)
- Java persistent datastores
- A new chapter on the Artix session manager
- Client threading models
- Incompatibility of the `-service` and `-plugin` parameters in the `wSDLtojava` command
- JMS Broker username/password support
- JMS transactionality settings
- JMS message acknowledgment
- Setting timeouts
- Support for restriction of complexContent

Artix for J2EE

This book has been updated to include information on the following:

- JCA local transactions
- Updates to the “Exposing Artix Web Services from a Servlet Container” chapter, including:
 - ♦ Addition of a WebLogic example
 - ♦ Use of `wSDLtojava -servlet` option

- ◆ Explanation of the relationship between Artix JARs, the `artix_ce.xml`, the application WAR classes and the relevant classloaders
- ◆ Updates to “Configuring Servlet Container to Run an Artix Application” section
- ◆ Updates to the example `web.xml` file
- Addition of JBoss 4.x deployment descriptor examples

Artix for CORBA

Information on the latest WSDL-to-IDL mapping has been added to this book.

Deploying and Managing Artix Solutions

This book has been updated to include information on the following:

- Deploying services on restart (persistent deployment)
- Forwarding of write requests from slave to master replicas
- High availability in the Artix locator
- The `wadd` command
- Specifying services in a chain

In addition, all configuration information has been moved to the *Artix Configuration Guide*.

Artix Configuration Guide

This book has been renamed and expanded from the 3.0.1 *Configuration Reference* to include the following:

- All configuration information previously in the *Deploying and Managing Artix Solutions* book
- Overriding configuration on the command line
- Specifying arbitrary symbols in configuration
- CA-WSDM configuration variables
- New variables for JMS enhancements
- New variables for persistent deployment
- All other new configuration variables added in 3.0.2

Artix Security Guide

This has been updated to include the following:

- The `<http-conf:client>` and `<http-conf:server>` security tags
- Security clustering

Artix Primer

This book is deprecated.

Known Issues in Artix 3.0.2

The following are known issues in Artix 3.0.2:

- [Binary compatibility](#)
- [Compiling on Red Hat Linux](#)
- [64-Bit Linux](#)
- [J2EE Connector](#)
- [Circular Dependency Between `it_afc` and `it_bus` libraries](#)
- [Artix Designer](#)
- [Demos](#)

Note: See also [“Known Issues in All 3.0 Releases”](#) on page 5.

Binary compatibility

Binary compatibility was broken between Artix 3.0.1 and 3.0.2. Applications built with Artix 3.0.1 need to be recompiled and relinked.

The table below explains the binary compatibility changes in more detail.

File	Class	Change
<code>it_bus_config/*.h</code> <code>it_bus_pdk/*.xml*.h</code>	<code>XMLSaxHandler</code>	Improves performance by reducing the number of unicode transcodings performed by the SOAP and XML bindings
<code>it_bus/server_service.h</code>	<code>IT_Bus::ServerService</code>	New pure virtual methods added to abstract class

File	Class	Change
it_bus/port.h	IT_Bus::Port	New pure virtual methods added to abstract class
it_bus_pdk/messaging_transport.h	IT_Bus::ServerTransport	New virtual methods added to abstract class.
it_bus/part_list.h	IT_Bus::PartList	New data member added for improved performance
it_schema_model/type_restriction.h	IT_XMLSCHEMA_Typenodes::TypeRestriction	New pure virtual methods added to abstract class to support complexContent restriction
it_bus/reflect/complex_content.h	IT_Reflect::ComplexContent	New pure virtual methods added to abstract class to support complexContent restriction

Compiling on Red Hat Linux

If you are running Red Hat Linux and are running your C++ compiler with the optimizer turned on (-O), any code that contains a method called `gettext` will not compile. This is caused by a bug in the `/usr/include/libintl.h` file.

You can work around the problem in one of the following ways:

- Modify the WSDL (or user code) to not have a method named `gettext`.
- Modify the `/usr/include/libintl.h` file to change the line:

```
#if defined __OPTIMIZE__
to:
#if defined __OPTIMIZE__ && !defined __cplusplus
```

64-Bit Linux

The following limitations apply to Artix on 64-bit Linux (Red Hat Enterprise Linux AS 3.0):

- SOAP with image attachments are currently not supported.
- The Eclipse CDT is only partially supported, as a fully 64-bit version is not yet available.

J2EE Connector

When deploying the Artix J2EE Connector to WebLogic, the following error are displayed on the console:

```
330 [ExecuteThread: '3' for queue: 'weblogic.kernel.System'] ERROR
com.iona.jca.artix.LocalTransactionImpl - failed to initialise
properly, no bus available
```

You can ignore this error.

Circular Dependency Between `it_afc` and `it_bus` libraries

If an application links with the `it_afc` library but not the `it_bus` library, you may see some unsatisfied symbol warnings. The `reflection/print_random` demo may exhibit this problem.

The work around is to link with the `it_bus` library also. Almost any Artix application will require this library in any case.

Artix Designer

The following are known issues with Artix Designer in version 3.0.2:

WSDL from Dataset The WSDL from Dataset wizard supports fixed data formats only, not Cobol Copybook or tagged.

Container generation Sometimes, creating a container configuration in the Artix Generator does not generate the required `.cfg` files in the `etc` folder, with the result that all start scripts fail. This can happen if you have edited a configuration and applied your changes a number of times. To work around this issue, create a brand new container configuration.

WSDL Editor When using the Undo feature in the WSDL editor, you need to select **Undo** (press **Ctrl+Z**) twice when using the Diagram view.

Plug-in generation When creating a plug-in/container configuration in Artix Generator, the generated service start script does not have execute permissions, resulting in a “permission denied” error when you try to run the script.

CORBA projects When creating a CORBA Web services project, the generated WSDL file contains a `definitions` element with an empty `name` attribute. This can cause problems in other Web services development tools. To work around this issue, enter an arbitrary value for the `name` attribute in the WSDL file.

XML bindings When creating an XML binding, the output messages are not listed in the Binding Operations edit page. Also, changing the input or output message namespace value does not change the value in the model or View Summary page.

Code completion The `IT_THROW_DECL` clause that is part of all Artix C++ method declarations fools the Eclipse code completion functionality. Eclipse considers `IT_THROW_DECL` to be the name of the method and this becomes the entry in the code completion drop-down and the Outline view. Consequently, you cannot use the code completion functionality or move through the source code file by clicking on the method name in the Outline view.

Demos

In order to run the CA WSDM Observer demo (`integration\ca_wsdm\observer`), you will need to do the following:

1. In the `etc` folder, open the `WSDM31MMI.wsdl` file.
2. Go to the `wsdl:operation` element.
3. In each of the child `wsdl:input` elements, remove the `Request` suffix from the `name` attribute. For example, change

```
<wsdl:input message="impl:sendAlertsRequest"
  name="sendAlertsRequest"/>
```

to

```
<wsdl:input message="impl:sendAlertsRequest" name="sendAlerts"/>
```

4. Go to the `wsdl:binding` element.
5. Repeat step 3 for each of the child `wsdl:input` elements.

Fixed Bugs in Artix 3.0.2

The following bugs have been closed in Artix 3.0.2:

Bug #	Description
69051	Basic authentication header to be returned on authentication failure
69057	XML inputter/outputter functions for schema types
69341	Provide an option that enables generation of Artix Java Beans such that they implement <code>java.io.Serializable</code>
69378	Proxies are not thread safe
69554	IS2 Scales badly on concurrent requests
69652	Artix bus does not throw JMS exceptions
69708	Logging system cannot support multiple buses
69735	Configure CORBA router to reread IOR
69775	WSDL compiler allows boolean enumeration with invalid values
69788	<code>idltoxsd</code> does not deal correctly with <code>typedef</code> objects in a <code>struct</code>
69795	If XML Schema contains a document type definition XML parsing error occurs at runtime
69803	If the Artix container is stopped and restarted, deployed services are lost
69804	JVM crash occurs during post-processing of Artix request-level interceptors
69814	Artix does not process <code>wsse:Password</code> element in WSSE usernametoken correctly if the type attribute specifies <code>"#PasswordText"</code>
69817	Exceptions thrown from JAX-RPC message handlers not handled correctly
69822	Security domain not considered during token's local authentication

Bug #	Description
69825	Adding a choice to a restriction of a complexType
69828	Artix client using schema types that are defined as XSD extensions fails to invoke its request with nullpointerexception
69836	Plugins: jms Shlib_name not set or set to an empty string
69840	Artix opens a listener on all network interfaces
69841	Artix router coredumps with mismatched IDL/WSDL
69855	Artix HTTP context cannot handle UTF-8 characters inside cookies
69855	Artix servlet demo doesn't support running multiple web apps
69857	Revocation lists not supported in Artix
69861	WSDL compiler has problems with schema includes
69863	wsdltojava -plugin will only register a single port in generated code
69864	Artix Java client takes too long to deliver first request
69867	Add a -servlet option to wsdltojava
69868	Artix prints an error message whenever SOAP message context is used
69871	CORBA endpoint socket closed results in irrecoverable failure
69872	Include callback mechanism in Artix where the application can receive JMS disconnect notification
69876	Invalid node IDs left in locator when running in replicated mode
69877	Unable to point to multiple locator replicated groups
69879	Artix bus expects to find a C++ plug-in library for JMS transport plug-in
69880	Incorrect namespace serialization in Artix 3.0.1
69881	Artix Java handler does not work in the Tomcat environment

Bug #	Description
69884	Include support for JMS message acknowledgement
69890	Occurrence constraints on sequences or choices in a complexType cause a failed build in C++
69906	<code>wSDLtojava</code> produces incorrect <code>toString</code> for schema array
69909	<code>wSDLtojava</code> has problems when <code>minlength/maxlength</code> is set within a string restriction
69924	TLS handshake fails between iSF client and IS2 on Solaris when IS is configured for old OWEB servlet engine
69925	WSDL model runs out of memory in Artix Java loading recursive WSDL imports
69932	Occurrence constraints on choices in a complexType cause a failed build in C++
69934	Using the Artix XSLT transformer on a complexType that contains a sequence or a choice with a <code>minOccurs=0</code> and <code>maxOccurs="unbounded"</code> results in a "not implemented" error
69939	Artix JMS has no username/password login facility for connection setup
69958	Different classloaders used in registering a factory
69979	<code>java.lang.ArrayIndexOutOfBoundsException: 0</code> returned from an Artix Java server when the client sends an <code>any</code>
69980	Any attempts to validate the element type when <code>processContents</code> set to "skip" or "lax"
69985	Artix Java API <code>SOAPMessageContext.setMessage()</code> when used in interceptor
69989	Failure in case of multiple endpoints with different ports but same url context
69995	Unable to access bus from Artix JCA (J2EE)
N/A	SOAPscope doesn't work with Artix 3.0

Bug #	Description
N/A	Uncompilable type factory when exception and type has same name

Artix 3.0.1

In these release notes

These release notes contain the following sections:

New Features in Artix 3.0.1	page 42
Documentation Updates in Artix 3.0.1	page 48
Known Issues in Artix 3.0.1	page 50
Fixed Bugs in Artix 3.0.1	page 55

New Features in Artix 3.0.1

The following new features have been added in Artix 3.0.1:

- [J2EE Connector](#)
- [Code Generation](#)
- [Artix Designer](#)
- [Artix Java](#)
- [SOAP Support](#)
- [Type Support](#)
- [Internationalization](#)
- [Logging](#)
- [Transports](#)
- [Artix Container](#)
- [High Availability](#)
- [Service Shutdown](#)
- [Artix Locator](#)
- [Finding WSDL Contracts and References](#)
- [UDDI Support](#)
- [CA WSDM Support](#)
- [Command Line Tools](#)
- [JMS Transport](#)
- [HTTP Transport](#)

- [User Credential Propagation from SOAP to CORBA Bindings](#)
- [Other](#)

J2EE Connector

Artix now includes a connector that allows you to service-enable J2EE applications. In addition, this connector allows you to connect Artix applications to communicate with applications hosted inside a J2EE application server.

Code Generation

The Artix code generators have the following new features:

- You can now generate code for multiple WSDL ports in a single pass.
- Artix now supports code generation of deployment descriptors, allowing generated plug-ins to be dynamically deployed into the Artix Container.

Artix Designer

Artix Designer is completely redesigned to work with the Eclipse Framework. When installed along with the Eclipse Java Development Tools (JDT), Artix automatically takes advantage of Eclipse's features.

Artix Java

Artix Java has the following new features:

- Artix handlers can now directly manipulate SOAP messages using the `SOAPMessageContext`.
- Artix handlers can now access the parts of a message in request level handlers.
- Artix handlers can now throw exceptions that are propagated through the messaging chain.
- Artix now provides a mechanism for determining whether an operation is oneway.
- Artix now provides message correlation inside of its handlers.
- Configuration contexts are supported.
- Artix now includes JAX-RPC `DLL Call` support.
- Artix can now resolve custom resources in Java.

SOAP Support

Support for Soap With Attachments has been added to MQ transport.

Type Support

Artix 3.0 now supports the following XML Schema features:

- `xsd:union`
- `xsd:substitutionGroup`
- `xsd:group`
- `xsd:attributeGroup`
- `minOccurs` and `maxOccurs` for choice complex types
- `minOccurs` and `maxOccurs` for sequence complex types
- `xsd:include`
- Recursive import and include

Internationalization

Artix 3.0 has added the following internationalization support:

- Internationalization for the fixed length record binding.
- Codeset conversion for transports that do not have their own concept of headers (for example, IBM Websphere MQ, BEA Tuxedo, and Tibco Rendezvous). This is implemented using an Artix message interceptor and WSDL port extensors.

Logging

Logging is now easier to use and more flexible. For more details, see [Deploying and Managing Artix Solutions](#).

Transports

The following updates have been made in Artix's transport support:

- Both the Java and C++ transport APIs now give the developer more control over the threading model used by server-side transports.
- Artix provides a Java servlet transport that enables you to run Artix applications inside a servlet container.
- Artix fully supports JMS correlation styles.

- Artix now fully supports oneway calls over HTTP. Control is now returned to the client as soon as the oneway operation successfully reaches the service endpoint.

Artix Container

The Artix generic service has been replaced by the Artix container. The container provides a number of advanced service management features, including dynamic loading of service plug-ins and full control of the lifecycle of services. For more information see [Deploying and Managing Artix Solutions](#).

High Availability

Artix now provides the ability to create highly available clusters of Artix services. These clusters provide data replication and failover support to Artix.

Service Shutdown

The Artix runtime now gracefully shuts down Artix services without interrupting any outstanding operations. Previously, a call to shutdown a service would leave any operations that were in progress in an indeterminate state. In 3.0 we guarantee the completion of all outstanding operations. This feature is provided for the following transports:

- HTTP
- IIOP Tunnel
- CORBA

Artix Locator

The following functionality has been added to the locator:

Endpoint listing The locator client C++ API has been extended to include an operation, `list_all_endpoints()`, that returns a list of all endpoints registered with a locator instance.

Service registration The locator now allows you to chose which services to register. Previously, the configuration of the `locator_endpoint` plug-in was on a per-bus basis while now you can load that plug-in and explicitly include or exclude certain endpoints from being registered with the locator. For more information see [Deploying and Managing Artix Solutions](#).

Finding WSDL Contracts and References

Artix 3.0 allows you to resolve an application's contract location and a client's references at run time. You can now use configuration settings or bus start-up arguments to set these values.

The C++ API now offers two new methods:

```
IT_Bus::Boolean IT_Bus::resolve_initial_reference(
    QName& service_name,
    Reference &endpoint_reference)
IT_WSDL:WSDLService* IT_Bus::get_service_contract(
    QName& service_name)
```

The Java API now offers two new methods:

```
Reference Bus.resolve_initial_reference(QName serviceName)
String Bus.getServiceWSDL(QName serviceName)
```

For more information see [Deploying and Managing Artix Solutions](#).

UDDI Support

Artix now provides UDDI support out of the box. This support includes:

- The JUDDI service is available as a standalone service.
- Client-side UDDI resolver support that allows users to bootstrap their proxies using WSDL from the UDDI Service with zero coding. This can be achieved using UDDI URLs.

CA WSDM Support

Artix now provides full support for CA Web Services Distributed Management (CA WSDM). This support comes in the form of an interceptor that updates the CA WSDM console with all the runtime information that CA WSDM needs to know about Artix services. Although CA WSDM was designed with SOAP/HTTP services in mind, the Artix integration provides support for all Artix bindings and transports. The integration is provided on both the client and server side.

Command Line Tools

The following changes have been made to the Artix command line tools. For more information see the [Artix Command Line Reference](#):

New tools The following command line tools have been added:

- `xsdtoxsd`
- `wsdltoservice`
- `wsdltorouting`
- `it_container_admin`
- [itartix_version](#)
- [WSDD](#)

itartix_version The new `itartix_version` command prints the version and build date of your Artix installation.

WSDD Deployment descriptors can be generated from both `wsdltocpp` and `wsdltojava`. However, the descriptors these tools can generate cannot include all the possible information descriptors can have, such as `provider_namespace`. See the `advanced/container/deploy_routes` demo for an example.

The Web Services Deployment Descriptor utility (WSDD) can be used as an alternative, and it should be used for more complex deployment descriptors. See [Deploying and Managing Artix Solutions](#) for more details.

New flags The `idltowsdl` tool has been upgraded to use `-inline` flag, `idltowsdl`, `javatowsdl` and `coboltowsdl` tools have been upgraded to use `-qualified`.

JMS Transport

The Artix JMS transport has been updated to include support for the following:

- Durable subscriptions
- Message selectors
- Correlation IDs
- Message priority

For more information on configuring a JMS transport, see [Designing Artix Solutions](#).

HTTP Transport

The Artix HTTP stack has been redesigned. The new HTTP stack has the following new features:

- IP-6 support

- Enhanced performance
- Baltimore SSL libraries

User Credential Propagation from SOAP to CORBA Bindings

Username/password or an ISF token sent from SOAP binding could be propagated into CORBA binding as csiv2 credentials for future authentication in the CORBA server from the router.

The following configuration parameters are added to enable or disable those features.

```
policies:bindings:corba:token_propagation="true";  
policies:bindings:corba:gssup_propagation="true";
```

Other

Artix 3.0 also includes the following new features:

- Support for WSDL comments
- WS-I compliance for SOAP headers
- Partial compliance with SOAP with Attachments API for Java (SAAJ) 1.2
- Support for WS-Atomic Transaction
- Changes to the API used to manage transaction boundaries from application code
- Support for Java Management Extensions (JMX)
- Support for the Field Manipulation Language (FML) binding
- A more robust and complete TIBRV binding

Documentation Updates in Artix 3.0.1

The following changes have been made to the Artix documentation for version 3.0.1:

- [Getting Started with Artix](#)
- [Designing Artix Solutions](#)
- [Developing Artix Applications in C++](#)
- [Developing Artix Plug-ins with C++](#)
- [Developing Artix Applications in Java](#)
- [Artix for J2EE](#)

- [Artix for CORBA](#)
- [Deploying and Managing Artix Solutions](#)
- [Artix Configuration Reference](#)
- [Artix Technical Use Cases](#)
- [Learning About Artix](#)

Getting Started with Artix

This book provides an overview of Artix, its features, and its basic concepts. It takes you through the process of creating a WSDL file and generating starting point code in both C++ and Java using the Artix Designer development tool. It also provides a brief overview of the Artix documentation set.

Designing Artix Solutions

This book has been reorganized to provide a more consistent view of Artix and how it is used when building Artix contracts.

Developing Artix Applications in C++

This book has been updated to include information on the new data types supported in Artix 3.0.

Developing Artix Plug-ins with C++

This is a new book in the Artix library. It discusses how to write Artix plug-ins. Features that you can implement in plug-ins include message interceptors and custom transports.

Developing Artix Applications in Java

This book has been updated to include information on:

- The new data types supported in Java
- Writing message handlers
- Developing Artix plug-ins
- Developing custom Java transports
- The extended context support

Artix for J2EE

This is a new book in the Artix library. It discusses how to expose J2EE applications to services outside the application server using Artix. It also discusses how to use Artix to allow other services to access applications being run in a J2EE application server.

Artix for CORBA

This is a new book in the Artix library. It focuses on using Artix in a CORBA environment.

Deploying and Managing Artix Solutions

This has been updated to include information on the Artix container. A number of previously undocumented configuration settings have been added to this guide. In addition, it has been updated with information on the new features included for 3.0.1.

Artix Configuration Reference

This is a new book in the Artix library. It is a reference guide of all the configuration variables that can be set in an Artix configuration domain.

Artix Technical Use Cases

This is a new book in the Artix library. It provides step-by-step walk-throughs of common Artix use cases such as Web service enabling a backend service.

Learning About Artix

This book is deprecated.

Known Issues in Artix 3.0.1

The following are known issues in Artix 3.0.1:

- [Type Support](#)
- [Threading Model Configurations](#)
- [SOAPscope](#)
- [Java Custom SOAP Headers](#)

- [HTTP](#)
- [Imported Schema Serialization](#)
- [Demos](#)
- [Missing Demos and Documentation](#)
- [Artix Designer](#)

Note: See also [“Known Issues in All 3.0 Releases”](#) on page 5.

Type Support

The Java runtime doesn't correctly support use of inheritance for anonymous types.

Threading Model Configurations

Thread control and configuration is undergoing a re-factoring that is not complete in Artix 3.0.1. Currently, the MQ and TIBRV transports use a different threading model from the HTTP and IIOp transports. Service-level configuration of thread pools is not supported for the HTTP and IIOp transports but is still supported for the MQ and TIBRV transports.

Thread pool sizes are set through the `thread_pool...` configuration variables, although only a subset of these threads are used to deliver requests to the servant.

MQ and TIBRV use the “port driven” thread model in which the thread pool is maintained by either the Service or Bus. Use the `policy:messaging_transport:concurrency` configuration variable to specify the number of threads that will be used to deliver requests to the servant.

HTTP and IIOp use the “external driven” thread model in which the thread pool is maintained by the ORB. Use the `plugins:http:pool:min_threads` and `plugins:iio:pool:min_threads` configuration variables to specify the number of threads that will be reserved for I/O tasks. The remaining threads (specified through the `thread_pool...` configuration variables) will be available to deliver requests to the servant.

The re-factoring process will continue for Artix 3.0.2.

SOAPscope

Due to a bug in how the platform handles the `ProxyServer` HTTP attribute, Mindreef SOAPscope cannot be deployed on the same computer as both the client and server, or on a different computer from both the client and server. You must deploy SOAPscope on a computer that also hosts either the client or server application.

Java Custom SOAP Headers

If a context, which holds a custom SOAP header, is used in multiple invocations, only the first invocation succeeds.

To reset the header content, you must remove the context that holds the header information. However, the Java implementation of the registry does not have a `remove_context` method. Consequently, an application cannot make multiple invocations once the custom SOAP header has been initialized.

HTTP

The following are known issues with Artix' HTTP support:

Exceptions Using the default configuration, Artix clients that invoke on invalid or non-existent services throw a `DeserializationException` instead of the standard HTTP "404 Not Found" exception. To receive the standard HTTP exception change the `policies:http:browser_navigation:enabled` value to `false`.

Port numbers If a server has two HTTP endpoints that use different port numbers and the same URL context, the server does not function correctly in some cases. For example, if a server has the following port definitions, messages may get redirected to the second port even if they were intended for the first.

```
<port name="..." binding="...">
  <soap:address location="http://localhost:12345/foo/blah"/>
</port>
<port name="..." binding="...">
  <soap:address location="http://localhost:56789/foo/blah"/>
</port>
```

Imported Schema Serialization

When you serialize WSDL with multiple levels of imported schemas using `WSDLDefinitions::write()`, the serialized WSDL contains types from the directly-imported schema, and import statements for the indirectly imported schemas.

If the import statements use relative file URLs and you are using the serialized WSDL from a directory other than the directory of the original WSDL, the WSDL parser is not able to find the imports.

Demos

The following are known issues with the demos included with Artix 3.0.1 (`InstallDir/artix/3.0/demos`):

Located router To run the located router demo successfully on UNIX platforms, you need to modify the `run_cxx_client` script to add `-BUSinitial_reference locator.ref` after `.././etc` to the command that runs the client executable.

Locator endpoints There is a relative path issue in the configuration file for the list locator endpoints demo (`advanced/locator_list_endpoints`). The `include` directive in `etc/locator_list_endpoints.cfg` is as follows:

```
include "../../../etc/domains/artix.cfg"
```

Session management The C++ client in the session management demo (`advanced/session_management`) sometimes crashes on Linux and HPUX when stopped with Control-C.

High availability locator In the high availability locator demo (`advanced/high_availability_locator`) open the configuration file (`etc/high_availability_locator.cfg`) and remove the entry `plugins:locator:peer_timeout` from the server scope.

Axis client Artix reference There is no `build.xml` file in the `integration/axis/axis_client_artix_reference_axis` directory, and thus the Java client doesn't get built with ant. Therefore, the `run_axis_client` script fails.

Servlet container The servlet container demo (j2ee/servlet_container) fails on UNIX platforms. To work around this add the following to the demo's start_tomcat script:

```
export IT_DOMAIN_NAME=tomcat

export IT_CONFIG_DOMAINS_DIR=$IT_PRODUCT_DIR/artix/3.0/demos/j2ee
servlet_container/etc
```

SOAP MQ to CORBA On UNIX platforms, the routing/soap_mq_corba/bin/start_router script should be modified to replace itartix_service with it_container.

Missing Demos and Documentation

Demos and documentation on the following new features and enhancements are not included in Artix 3.0.1.

- Support for WS-Atomic Transaction
- Changes to the transaction API
- JMX support
- FML binding support
- Improvements to the TIBRV binding

Artix Designer

The following are known issues with Artix Designer:

CDT The Eclipse C/C++ Development Toolkit (CDT) plug-in is not supported out-of-the-box in Artix 3.0.1.

Container scripts The script files generated by a container application on UNIX/Linux are of Windows format.

Fixed Bugs in Artix 3.0.1

The following bugs have been closed in Artix 3.0.1:

Bug #	Description
68228	Artix Designer silently adds <code>MaxOccurs="1"</code> to the WSDL
68676	Support for clustered IS2 instances.
68816	Artix 2.0 Designer does not update a WSDL after it has been edited by an external editor
68929	Process needs to gracefully exit if the port number is already used.
68949	Accessing the SOAP/XML message at runtime as a string
69001	Add command line utilities to add a SOAP binding to an Artix contract
69012	Add support for headers in Java
69039	Enhance <code>wsdltocorba</code> to add a new switch to pass a value for <code>corba:address</code>
69072	<code>itartix_service</code> should allow user to specify alternative key for window service
69118	Artix Logging to XML file will include subsystem name
69138	<code>artix_env</code> script requires improvement in the way <code>-verbose</code> flags function
69140	Support custom headers and interceptors
69141	Cross protocol exception mapping needed
69179	<code>wsdltojava</code> unable to use circular imports of XSD
69214	Artix 2.1: Unable to print help contents from Designer
69219	Method for error handling needed in <code>IT_Bus::ServerRequestInterceptor</code>

Bug #	Description
69229	Bogus "no parts matching the return type name" error message
69245	Artix XML logging does not having rolling logfile capabilities
69252	Incorrect type names generated with <code>wsdltocpp</code> and <code>wSDLtojava</code> . <code>WS</code> is used instead of <code>WSExceptionType</code>
69253	<code>_setProperty</code> causing Artix to core on Windows
69264	Schema validator needs to support port-range
69270	Artix router service uses one bus configuration (orb) for both loaded WSDL and administration servant
69283	Pulling in URL WSDL while in the editor mode
69295	Artix throws incorrect exception on authorization failure
69300	Support fully qualified hostname in SOAP & HTTP addresses
69303	Artix documentation should explain the usage of <code>to_string()</code> and the required library to print <code>anyTypes</code> in Artix C++
69317	Save Project as... option
69323	Correct typo in XSLT demo's configuration
69332	Default behavior of C++ <code>to_string</code> method should be changed
69334	Documentation of C++ <code>to_string()</code> method needs to be updated and moved into mainstream product documentation
69339	When the GIOP principal is set with the GIOP current, it is ignored
69371	Enhancement to list active endpoints in locator
69375	Add a command line tool to build WSDL from Schema
69377	<code>com.ionajbus.ntv.BusLocal.getCurrentBus()</code> reverts to the default Bus if it has never been set
69394	Artix interceptor does not work while running within the Artix Router

Bug #	Description
69395	<code>StackOverflowError</code> when importing SOAP schema
69396	Artix should throw an exception if it encounters a <code>SOAPHeader</code> that it can't understand and where the attribute <code>mustUnderstand</code> is set to 1
69403	Artix server-side configuration context settings should be scoped at the Servant/port level
69405	Artix Java needs JAX-RPC DII <code>Call</code> interfaces
69406	Bus init with java Properties
69411	Remove Artix plug-in order dependency
69414	Artix Java handlers should support the JAX-RPC <code>SOAPMessageContext</code>
69415	Allow the user to set the SOAP address programmatically within an Artix Java server
69423	Artix can't call <code>registerContext()</code> with an anonymous element type
69426	Artix should cache SSO tokens rather than just using the IS2 cache
69427	Artix CORBA server fails to start when configured for indirect persistence and as a <code>locator_endpoint</code> at the same time
69435	Make the Java interceptor list in the Artix config independent from the declarations in the WSDL
69453	Artix client crashes with dimensionless SOAP encoded arrays
69478	Artix 2.1.4 Java server crashes on Windows if an empty string is sent in an <code>xsd:datetime</code> type
69481	Add the ability for different plug-ins to be able to discover each other within an Artix process
69485	Reduction in the number of threads used within a multi-wsdl Artix process

Bug #	Description
69511	A standalone IS2 client specifying its <code>caCertDir</code> pointing to a <code>.pem</code> file, fails to set up its keystore with JDK 1.4.2
69521	Namespace containing node import results in Java name collision
69542	IS2 LDAP adapter not thread safe
69543	Bus response monitor does not appear to work in some configurations
69547	Error message for missing shared libraries is logged inconsistently (and might crash Artix at <code>Bus.init()</code>)
69548	Artix 2.1.5 <code>xmlfile_log_stream</code> subsystem does not report TLS related exceptions whereas the <code>local_log_stream</code> subsystem does
69549	Unnecessary <code>IT_Bus::Exception: ContextException</code> messages when full logging is turned on
69555	Artix should warn the user if the WSDL import <code>targetName</code> matches the actual value of the enclosing WSDL document <code>targetNamespace</code>
69557	Artix <code>wSDL:import</code> & <code>schema:import</code> allows namespace coercion
69559	Artix can not support chameleon <code>xsd:include</code>
69563	<code>idltowSDL</code> should not encode the absolute pathname of the idl file into the <code>targetNamespace</code> of the generated WSDL document
69571	<code>MinOccur=0</code> causes "deserialize called too many times" in Artix Java
69585	Java message interceptor eats up the transport exception
69608	Soap with attachments using a message handler doesn't work
69616	<code>idltowSDL</code> needs option to generate <code>elementFormDefault</code> on schema

Bug #	Description
69634	Artix servers do not like SOAP messages with multiple namespaces
69654	<code>toString</code> displaying incorrect output for choice elements in Artix for Java
69656	With <code>minOccurs</code> set to 0 for an <code>xs:choice</code> within another <code>xs:choice</code> , Artix 2.1.5 for Java complains if this element is not set or present
69667	<code>idltowSDL</code> should generate WSDL with version comments
69682	Artix <code>xml_log_stream</code> exhibits poor performance compared to <code>local_log_stream</code>
69753	Artix router sends wrong Reference
69766	Artix server core dumps on a SOAP Message with an empty <code>DateTime</code> field in the message
69773	MQ Oneway operation with <code>Transaction=internal</code> does not perform rollback on application exception
69782	Artix logging does not work correctly for Artix 3.0.
69783	null pointer exception thrown in the artix java message interceptor while receiving a soap fault
69792	Artix HTTP Context cannot handle UTF-8 characters inside cookies
69425/ 69287	Exception is not handled properly in the HTTP plug-in.