



Release Notes

Version 2.0.4 , June 2004

IONA, IONA Technologies, the IONA logo, Orbix, Orbix/E, Orbacus, Artix, Orchestrator, Mobile Orchestrator, Enterprise Integrator, Adaptive Runtime Technology, Transparent Enterprise Deployment, and Total Business Integration 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.

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 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.

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 book. This publication and features described herein are subject to change without notice.

Copyright © 2001–2003 IONA Technologies PLC. All rights reserved.

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

Updated: 22-Sep-2005

Contents

Upgrading from Previous Versions	1
New Features	2
Features Added in 2.0	3
Features Added in 2.0.2	5
Features Added in 2.0.4	5
Documentation Updates	6
Updates for 2.0	6
Updates for 2.0.2	7
Updates for 2.0.3	7
Known Problems	7
Installation	8
Type Support	8
Demos	8
Artix Designer	9
Tuxedo Plugin	10
Java	10
JMS	10
Security Server	11
Imported Schema Serialization	11
Secure Artix Configuration	11
XSLT Engine	12
Fixed Bugs	12
Fixed in 2.0	12
Fixed in 2.0.1	14
Fixed in 2.0.2	15
Fixed in 2.0.4	15
Reporting Problems	16
Other Resources	16

CONTENTS

Artix 2.0 Release Notes

In this document

This document contains the following sections:

Upgrading from Previous Versions	page 1
New Features	page 2
Documentation Updates	page 6
Known Problems	page 7
Fixed Bugs	page 12
Reporting Problems	page 16
Other Resources	page 16

Upgrading from Previous Versions

When upgrading from a 1.x version of Artix you need to do the following:

- acquire and install new licences. If you have not received your new license please contact your IONA representative.
- recreate any existing Artix 1.x projects as the Artix 2.0 project format and the Artix 1.x project format are not compatible.
- regenerate all code generated from WSDL.
- due to changes in the way Artix generates C++ code for complex types containing elements with `minOccurs=0` or `maxOccurs>1`, you will need to change any Artix application code that explicitly uses `ElementListT`. The code will need to be modified as shown by the following example:

```
// Existing code:
IT_Bus::ElementListT<SomeType, &SomeComplexType_x_qname, 0, 10>&
    list = something.getx();
// Correct code:
IT_Vector<SomeType>& list = something.getx()
```

- due to a change in the Artix C++ servant implementation, user code can no longer get the `Port` object directly from within the generated server `Impl` object. You must now use the `Current` object to get the `Port` object in Artix server code. The code for getting a port using the `Current` object is shown below:

```
void TestImpl::do_stuff() IT_THROW_DECL((IT_Bus::Exception))
{
    Current& current = get_bus()->get_current();
    Port& port = current.get_operation().get_port();
}
```

- due to a change in the way attributes are mapped in C++ optional attributes can no longer be specified in WSDL messages. Attributes are optional by default, unless otherwise specified in the schema.
- due to a change in the way attributes are mapped in C++, attribute getter and setter methods use pointers in 2.0. The code generated for attributes in 2.0 resembles the following:

```
Int* Foo::geta();
Foo::seta(const Int*);
Foo::seta(const Int&);
```

As a result of this change, all applications that use `get()` to retrieve attribute values must be changed to check that the value is set and not a null pointer.

- check the relevant development guides for other changes to Artix generated code that will impact your applications.
- recompile all applications that contain generated code or reference generated header files.

New Features

This section discusses the following topics:

Features Added in 2.0	page 3
Features Added in 2.0.2	page 5
Features Added in 2.0.4	page 5

Features Added in 2.0

The following new features have been added for Artix 2.0:

- [<complexContent> Support](#)
- [COBOL Copybook Importing](#)
- [Callbacks and Artix References](#)
- [JMS Pub/Sub Support](#)
- [Artix Designer](#)
- [CORBA Binding](#)
- [Command Line Tools](#)
- [Java Type Support](#)
- [BMC Patrol Integration](#)
- [CORBA Oneway Support](#)

<complexContent> Support

Artix 2.0 supports XMLSchema types defined by extension using `<complexContent>`. Types defined this way are mapped using language specific inheritance mechanisms.

COBOL Copybook Importing

Artix 2.0 will import COBOL copybooks and create WSDL representations of the data. For information on using this feature see *Designing Artix Solutions from the Command Line*.

Callbacks and Artix References

Artix 2.0 now fully supports callbacks. An Artix callback is an implementation pattern, whereby a client implements a WSDL service (thus exhibiting hybrid client/server behavior). Because the server initially does not know about the client's service, the client must send details about its service to the server. Once the server receives these details, it is able to call back on the client's service at any later time.

In Artix, the details about a service are encapsulated in an *Artix reference*. Hence, the extended support for references in Artix 2.0 holds the key to using callbacks effectively. In particular, callbacks rely on the following features of Artix 2.0 references:

- References have been refactored so that they are identified with a particular WSDL service, rather than a WSDL port as previously.

- References can be generated easily by calling the new `IT_Bus::Service::get_reference()` function.
- References are now fully supported by Artix routers. Every reference that passes through a router (whether as an input or an output parameter) is now automatically *proxified*. Proxification means that the router automatically creates a new route for the references that pass through it.

For information on using callbacks and Artix references read *Developing Artix Applications in C++*.

JMS Pub/Sub Support

The Artix JMS transport now supports the publish & subscribe messaging domain for oneway operations. The `<jms:address> destinationStyle` attribute may be set to `topic`.

Artix Designer

The Artix Designer has been completely redesigned for Artix 2.0. It now organizes projects using a workspace paradigm. The new designer includes extended support for importing COBOL copybooks, Java code generation, and deployment bundle generation.

CORBA Binding

The CORBA binding has been updated to include support for the following:

- Artix references.
- recursive data types.
- attributes specified in `<simpleType>` elements.
- improved mapping of types defined using `<complexContent>`.

Command Line Tools

Artix 2.0 adds the following command line tools:

- `wSDLtoacl` generates an ACL file from an Artix contract.
- `wSDLtosoap` generates a SOAP binding from the logical portion of an Artix contract.
- `coboltowSDL` generates an Artix contract with a fixed binding from a COBOL copybook.

For information on using the new command line tools see *Designing Artix Solutions from the Command Line*.

Java Type Support

Artix Java supports `xsd:anyType`.

BMC Patrol Integration

Artix 2.0 supports integration with the BMC Patrol enterprise application management system. For information using on using BMC Patrol to manage Artix applications see the *IONA BMC Patrol Integration Guide*.

CORBA Oneway Support

Artix 2.0 generates CORBA oneway operations for WSDL operations that do not have an output message.

On-Demand Plugin Loading

For 2.0, Artix loads transport and payload format plugins as needed. The Artix bus determines which plugins it needs from the services and binding information in the loaded Artix contract. As a result, you no longer need to specify Artix transport or payload format plugins in the `orb_plugins` list of an Artix applications configuration scope. For more information on plugins and Artix configuration see *Deploying and Managing Artix Solutions*.

Single Sign-On

Single Sign-On allows clients to present credentials for authentication once, and reuse the SSO token thus obtained for all subsequent authentications. For information about using single sign-on read the *Artix Security Guide*.

Features Added in 2.0.2

The following new features have been added for Artix 2.0.2:

- an XSLT engine that allows user to transform messages as they are passed between Artix applications.
- i18n compliance.
- support for CORBA `wstring`.
- the ability to print WSDL documents from Artix Designer.
- added support for openssl-0.9.7d.

Features Added in 2.0.4

Artix now supports Visual C++ v7.1 for all Artix C++ applications.

Documentation Updates

This section discusses the following topics:

Updates for 2.0	page 6
Updates for 2.0.2	page 7
Updates for 2.0.3	page 7

The Artix documentation is updated regularly on the IONA Support Web Site. Please check there regularly for the latest documentation.

Updates for 2.0

The following changes have been made to the Artix documentation for Artix 2.0:

- [Designing Artix Solutions](#)
- [Designing Artix Solutions from the Command Line](#)
- [Developing Artix Applications in Java](#)
- [Developing Artix Applications in C++](#)
- [Artix Tutorial](#)

Designing Artix Solutions

This book has been split into two books. *Designing Artix Solutions* deals exclusively with using the Artix Designer. All command line tools and discussions of detailed type mappings have been moved into *Designing Artix Solutions from the Command Line*.

Designing Artix Solutions from the Command Line

This book discusses how Artix extends WSDL to map data and application interfaces into the different transports supported by Artix. It also discusses the Artix routing extensions and the command line tools that Artix delivers. The content has been updated to reflect the support for `<complexContent>` in the CORBA type mapping and the support for Artix references in the CORBA binding.

Developing Artix Applications in Java

This book has been modified to provide detailed information about how Artix maps data types from WSDL to Java. It also discusses how to create user defined exceptions in Artix.

Developing Artix Applications in C++

This book has been updated to include:

- details about the updated message attribute interface in Artix C++.
- examples of creating callbacks in Artix C++.
- using data types described using `<complexContent>`.
- examples of using the updated server registration methods in Artix C++.

Artix Tutorial

The *Artix Tutorial* has been completely rewritten for 2.0. It is now independent from the Artix demos and provides examples using the Artix designer.

Updates for 2.0.2

The following documentation updates have been made for Artix 2.0.2:

- *Deploying and Managing Artix Solutions* has been updated to include information on using the internationalization features of Artix.
- *Deploying and Managing Artix Solutions* has been updated to include information on using the XSLT engine.
- *Developing Artix Applications in C++* has been updated to reflect changes in the Artix `IT_String` API.

Updates for 2.0.3

The following updates have been made for Artix 2.0.3:

- *Designing Artix Solutions from the Command Line* has been updated to clarify the use of the discriminator option in the fixed format binding and to reflect updates in the tuxedo service specification.
- A command line tools reference guide has been added.

Known Problems

The following are known problems in this release:

- [Installation](#)
- [Type Support](#)
- [Demos](#)
- [Artix Designer](#)

- [Tuxedo Plugin](#)
- [Java](#)
- [JMS](#)
- [Security Server](#)
- [Imported Schema Serialization](#)
- [Secure Artix Configuration](#)
- [XSLT Engine](#)

Installation

The following are known issues with the installation of Artix 2.0:

- Artix 2.0.x cannot be installed in the same directory tree as an Artix 1.x installation. It is recommended that you completely remove any Artix 1.x installations from your system before installing Artix 2.0.x.
- Uninstalling Artix leaves behind a number of files that prevent successful re-installation into the same directory. It is advised that you manually delete all files left behind by the uninstaller before attempting to reinstall.
- The `javatowsdl` tool is not installed on AIX.
- The `wSDLtojava` tool is not installed on AIX.

Type Support

Due to limits in the C++ compilers, 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 XSchema. This also affects the Java runtime as it is also limited by range of the C++ runtime.

Demos

The following are known issues with the demos included with Artix:

- Currently JMS demos on HP will not run. JMS demos require loading the `jvm_manager` plugin which depends on the jre library `libjvm.sl`. Loading `libjvm.sl` is problematic for the following reason as documented on HP's website (http://www.hp.com/products1/unix/java/java2/sdkrte14/infolibrary/sdk_notes_1.4.2.02.html#libjvm).

After updating to the proper patch, simply calling `export LD_PRELOAD=jre_lib/lib/PA_RISC/server/libjvm.sl` will allow you to run the JMS demos.

- To run the demo, `switch\run_outside`, you have to put the `mq` plugin before the `routing` plugin in the `orb_plugins` list as follows:

```
iona_services
{
  artix_service
  {
    orb_plugins = ["xmlfile_log_stream", "iiop_profile", "giop",
                  "iiop", "soap", "http", "ws_orb",
                  "mq", "routing"];
    ...
  }
}
```

- On Linux systems you need to use the following command to source the demo environment scripts:

```
source ./env
```

- In order to get the `\routint\tagged_tuxedo_soap_http` demos to work on Unix platforms you must run the following commands:

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

- The `transports/soap_over_mq` demo does not work on AIX.
- The Java client of the `routing/soap_mq_corba` demo does not work on AIX.

Artix Designer

The following are known issues when using Artix Designer:

- Deployment profiles cannot be created on Linux. You must use the default C++ deployment profile that is created for the workspace.
- The `artix_env` script included in the deployment bundle is invalid.
- You cannot edit a deployment profile or a deployment bundle once it is created. You must delete it and create a new one if you want to use different settings.

- The code generator for Java creates both client and server code regardless of which application type you selected when creating the bundle definition.
- The script generated to start a Java server is incorrect.
- The `artix_env` script generated using the **Fasttrack Dialog C++ Client** or **C++ Client & Server** choices always points to the `artix.cfg` that comes with the install rather than to the one that ultimately gets generated by **Run Deployer**.

Tuxedo Plugin

Tuxedo does not allow you to have a space in the pathname 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 problems with Artix Java:

- `THREAD-LOCAL` should not be used when developing Artix Java applications. They are not maintained by the Artix runtime.
- On AIX, there is a possible conflict between the version of `xerces` and `xalan` in the AIX JDK and the version that is shipped with Artix you may need to add the following flags when running Artix Java applications:
 - ♦ `$JDK_ENDORSED_DIRS`
 - ♦ `$JDK_BOOTSTRAP_CLASSPATH`
- Artix does not support the generation of WSDL from Java operations that return an array of a base type like `int`, `long`, or `String`.
- Artix Java applications do not support the use of C++ interceptors built using Visual C++ v7.1.

JMS

Artix C++ applications that use the Artix JMS transport do not support the use of Visual C++ v7.1. They must be built using Visual C++ v6.0.

Security Server

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

```
/usr/lib/dld.sl: Unresolved symbol: oop_iterate__7oopDescFP10oopClosure (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__7oopDescFP10oopClosure (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.

Imported Schema Serialization

There is an issue using WSDL with imported schemas that also import other schemas. When you serialize the WSDL with multiple levels of imported schemas using `WSDLDefinitions::write()`, the serialized WSDL will contain 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 different directory to the original WSDL, then the WSDL parser will not be able to find the imports.

Secure Artix Configuration

On Windows systems, the secure Artix configuration file, `install_dir\artix\2.0\ext\domains\artix-secure.cfg` may contain entries for the property `plugins:is2_authorization:action_role_mapping` that have spaces. The spaces, if any must be replaced by `%20`. For example, the entry

```
plugins:is2_authorization:action_role_mapping="file://C:\Program
Files\IONA\artix\2.0\demos\security\full_security\etc";
```

must be changed to

```
plugins:is2_authorization:action_role_mapping="file://C:\Program%20Files\IONA\artix\2.0\demos\security\full_security\etc";
```

XSLT Engine

The XSLT engine does not support complex types derived from other complex types.

Fixed Bugs

This section discusses the following topics:

Fixed in 2.0	page 12
Fixed in 2.0.1	page 14
Fixed in 2.0.2	page 15
Fixed in 2.0.4	page 15

Fixed in 2.0

The following bugs have been closed in Artix 2.0:

Bug #	Description
68525	Artix locator returns the exact contents of <soap address=...> for <code>lookupEndpointResponse</code> .
68529	Located session server core dumps on startup if <code>event_log:filter</code> is set to <code>"**"</code> .
68548	Fixed binding support for the <code>minOccurs</code> and <code>maxOccurs</code> attributes does not work for elements of basic types.
68551	Having multiple servers on the same HTTP port does not work when Keep-Alive is enabled.
68561	MQ intercommunication does not work when client and server are on separate machines.

Bug #	Description
68623	Artix Designer code generator does not map Java package names to directories before generating code.
68625	<code>artix_env</code> does not prepend the <code>PATH</code> variable with <code>JAVA_HOME/bin</code> .
68626	Artix Designer writes the wrong path to the WSDL in generated Java server and sample code.
68637	Artix <code>xsd</code> attribute does not support special characters (e.g. carriage returns).
68646	Artix client fails when null is provided for a value with <code>minOccurs=0</code> in the WSDL.
68651	Uncaught exceptions cause session-managed servers to core dump.
68653	Core dump due to unknown SOAP Fault.
68654	Crash in locator service due to undeclared exceptions.
68655	[Enhancement] Confirm project save when Designer is closing and the project is not saved.
68656	[Enhancement] Automatically validate when applying WSDL edits.
68663	Session managed service core dumps in <code>IT_Bus::ServiceFactoryImpl::create_service()</code> .
68664	Session managed service core dumps in <code>IT_Bus_Services::EndpointManagerDetails::register_endpoint()</code> .
68665	Artix locator core dumps on <code>IT_Mutex::lock()</code> .
68697	Artix Java SOAP client throws a <code>NullPointerException</code> in the native code for “ref” members.
68707	Reconnect required on Artix client for server Keep-Alive timeout.

Bug #	Description
68713	[Enhancement] <code>artix_env</code> should be self-contained and not dependant on the shell environment.
68715	Memory leak when creating a proxy.

Fixed in 2.0.1

The following bugs have been closed in Artix 2.0.1:

Bug#	Description
68719	When <code>dateTime</code> type is passed on the wire (as a part of a complex type), call fails with a <code>com.iona.jbus.WriteException</code> . (Java only)
68680	The Artix Validator does not detect WSDL error when an anonymous type is specified explicitly and an element refers to it anonymously.
68665	Locator service core dumps on <code>IT_Mutex::lock()</code> .
68664	Session managed server core dumps in <code>IT_Bus_Services::EndpointManagerDetails::register_endpoint</code> .
68663	Session managed server core dumps in <code>IT_Bus::ServiceFactoryImpl::create_service</code> .
68656	Enhance layout of menus, sub menus and command buttons on the WSDL edit tab.
68655	Confirmation dialog box when the Designer is closed and the project is not saved.
68654	Crash in locator service.
68653	Server core dumps due to uncaught exception.
68651	Core dump in the session-managed server.
68626	Artix Designer outputs a wrong path to the WSDL in the generated Java server and demo code.

Bug#	Description
68625	Enhance <code>artix_env[.bat]</code> to append/prepend the <code>PATH</code> variable with <code>\$JAVA_HOME/bin</code> .
68623	Artix Designer code generator does not map Java package names to directories before generating the code.
68619	Artix core dump when permissions not set for directory (logs, dynamic wsdl) being written to.
68611	Unable to determine a null attribute.

Fixed in 2.0.2

The following bugs were closed in Artix 2.0.2:

Bug #	Description
68790	Session-managed server: coredump due to unexpected exception thrown in <code>begin_session</code> .
68715	Memory Leak when creating a proxy and invoking in Artix 1.2.
68713	Enhancement request: <code>artix_env[.bat]</code> script needs to be self contained and not dependent on the shell environment.
68707	Reconnect required on Artix client and server Keep-Alive timeout.

Fixed in 2.0.4

The following bugs were closed in Artix 2.0.4:

Bug #	Description
68982	Code generator produces uncomparable C++ for type inheritance.
69006	It is difficult to determine the service pack number of Artix installed.
68962	Artix Designer puts hard-coded namespace values when it generates WSDL from IDL.

Bug #	Description
68956	server_address_mode_policy:publish_hostname does not work for https.
68919	Artix 2.0.3 Advanced Edition does not install wsdltojava and javatowsdl.
68935	wsdltosoap adds duplicate reference imports.
68966	Some log messages are logged as warning messages but should be information messages.
68924	wsdltoCORBA generates invalid IDL.
68975	wsdltoCPP or wsdltojava generated code does not compile because of generated class name collision.
68835	Change the log level of IT_BUS:0) I - HttpUrlEndpoint::connection_wait, Entering message from INFO_HIGH to INFO_LOW.
68846	The same artix switch should support authentication by username/password and the sso token.
69086	SoapContextContainer needs to cleanup contexts.
69051	HTTP - Basic authentication header to be returned on authentication failure.

Reporting Problems

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

Other 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 today.

- [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 try to post queries to this e-mail alias; it has been set up only to notify you of security alerts.

- [IONA Security Advisory Archive](http://www.iona.com/support/docs/artix/security_advisories/index.xml) (http://www.iona.com/support/docs/artix/security_advisories/index.xml): This page archives all of the Artix-related security updates that have been sent to the the IONA security mailing list.
- [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.