



# Artix™ Mainframe

---

## C++ SDK Installation Guide

Version 5.1, December 2007

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 logos, Orbix, Artix, Making Software Work Together, Adaptive Runtime Technology, Orbacus, IONA University, and IONA XMLBus 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.

---

#### 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 © 2001-2007 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: December 14, 2007

# Contents

<b>Preface</b>	<b>5</b>
<b>What is Covered in This Book</b>	<b>5</b>
<b>Who Should Read This Book</b>	<b>5</b>
<b>Prerequisites</b>	<b>5</b>
<b>The Artix Mainframe Library</b>	<b>5</b>
<b>Getting the Latest Version</b>	<b>7</b>
<b>Searching the Artix Library</b>	<b>7</b>
<b>Artix Online Help</b>	<b>7</b>
<b>Artix Glossary</b>	<b>8</b>
<b>Additional Resources</b>	<b>8</b>
<b>Document Conventions</b>	<b>8</b>
<b>Installing the C++ SDK</b>	<b>11</b>
<b>Before You Install</b>	<b>12</b>
<b>Installation Steps</b>	<b>13</b>
<b>Uninstalling Artix Mainframe C++ SDK</b>	<b>20</b>
<b>For More Information</b>	<b>21</b>
<b>Index</b>	<b>23</b>

## CONTENTS

# Preface

## What is Covered in This Book

This book describes the prerequisites and procedures for installing the C++ SDK component as an add-on to an existing Artix Mainframe installation.

## Who Should Read This Book

This book is intended for z/OS systems programmers who want to install the C++ SDK component as an add-on to Artix Mainframe.

## Prerequisites

You must have already installed and customized Artix Mainframe on your system, as described in the Artix Mainframe *Installation Guide*.

## The Artix Mainframe Library

The Artix Mainframe documentation library includes the following books:

- [C++ SDK Programmer's Guide](#)
- [Introduction to Artix Mainframe](#)
- [Getting Started](#)
- [Service-Enabling Existing z/OS Applications](#)
- [Creating New z/OS Applications from WSDL](#)
- [Creating New z/OS Applications from IDL](#)
- [Service-Enabling DB2 SQL](#)
- [Common User Tasks](#)
- [Administrator's Guide](#)

## C++ SDK Programmer's Guide

The *C++ SDK Programmer's Guide* describes how to develop application code to customize the behavior of your Artix Mainframe deployments.

### **Introduction to Artix Mainframe**

The *Introduction to Artix Mainframe* guide provides an introductory overview of Artix Mainframe in terms of its components and the various integration solutions that it supports. It also provides an introductory overview of Web services and CORBA technology in general.

### **Getting Started**

The *Getting Started* guide is intended to help you get started quickly with Artix Mainframe. It provides demonstration walkthroughs of the various integration solutions that Artix Mainframe supports.

### **Service-Enabling Existing z/OS Applications**

The *Service-Enabling Existing z/OS Applications* guide explains how to use Artix Mainframe to expose existing mainframe applications as Web services or CORBA servers or both.

### **Creating New z/OS Applications from WSDL**

The *Creating New z/OS Applications from WSDL* guide explains how to use Artix Mainframe to create CICS or IMS-based COBOL or PL/I Web services starting from WSDL. It also explains how to create z/OS-based COBOL or PL/I Web service clients starting from WSDL.

### **Creating New z/OS Applications from IDL**

The *Creating New z/OS Applications from IDL* guide explains how to use Artix Mainframe to create z/OS-based COBOL or PL/I Artix CORBA clients starting from IDL.

### **Service-Enabling DB2 SQL**

The *Service-Enabling DB2 SQL* guide explains how to use Artix Mainframe to expose DB2 SQL statements and stored procedures as Web services.

### **Common User Tasks**

The *Common User Tasks* guide explains how to use Artix Mainframe to perform various common tasks and implement various features that are not specific to any particular integration solution. It is provided as an addendum to the other user guides.

### **Administrator's Guide**

The *Administrator's Guide* describes how to set up and configure the various features of Artix Mainframe. It also discusses the various levels of security that can be used in the product.

## Getting the Latest Version

The latest updates to the Artix Mainframe documentation can be found at <http://www.iona.com/support/docs>.

Compare the version dates on the web page for your product version with the date printed on the copyright page of the PDF edition of the book you are reading.

## Searching the Artix Library

You can search the online documentation by using the **Search** box at the top right of the documentation home page:

<http://www.iona.com/support/docs>

To search a particular library version, browse to the required index page, and use the **Search** box at the top right, for example:

<http://www.iona.com/support/docs/artix/mainframe/5.1/index.xml>

You can also search within a particular book. To search within a HTML version of a book, use the **Search** box at the top left of the page. To search within a PDF version of a book, in Adobe Acrobat, select **Edit | Find**, and enter your search text.

## Artix Online Help

Artix Designer and Artix Orchestration Designer include comprehensive online help, providing:

- Step-by-step instructions on how to perform important tasks
- A full search feature
- Context-sensitive help for each screen

There are two ways that you can access the online help:

- Select **Help | Help Contents** from the menu bar. The help appears in the contents panel of the Eclipse help browser.
- Press **F1** for context-sensitive help.

In addition, there are a number of cheat sheets that guide you through the most important functionality in Artix Designer and Artix Orchestration Designer. To access these, select **Help | Cheat Sheets**.

## Artix Glossary

The [Artix Glossary](#) is a comprehensive reference of Artix terms. It provides quick definitions of the main Artix components and concepts. All terms are defined in the context of the development and deployment of Web services using Artix.

## Additional Resources

The [IONA Knowledge Base](http://www.iona.com/support/knowledge_base/index.xml) ([http://www.iona.com/support/knowledge\\_base/index.xml](http://www.iona.com/support/knowledge_base/index.xml)) contains helpful articles written by IONA experts about Artix and other products.

The [IONA Update Center](http://www.iona.com/support/updates/index.xml) (<http://www.iona.com/support/updates/index.xml>) contains the latest releases and patches for IONA products.

If you need help with this or any other IONA product, go to [IONA Online Support](http://www.iona.com/support/index.xml) (<http://www.iona.com/support/index.xml>).

Comments, corrections, and suggestions on IONA documentation can be sent to [docs-support@iona.com](mailto:docs-support@iona.com).

## Document Conventions

### Typographical conventions

This book uses the following typographical conventions:

*Fixed width*

Fixed width (Courier font) in normal text represents portions of code and literal names of items such as classes, functions, variables, and data structures. For example, text might refer to the `IT_Bus::AnyType` class.

Constant width paragraphs represent code examples or information a system displays on the screen. For example:

```
#include <stdio.h>
```

*Fixed width italic*

Fixed width italic words or characters in code and commands represent variable values you must supply, such as arguments to commands or path names for your particular system. For example:

```
% cd /users/YourUserName
```

*Italic*

Italic words in normal text represent *emphasis* and introduce *new terms*.

**Bold** Bold words in normal text represent graphical user interface components such as menu commands and dialog boxes. For example: the **User Preferences** dialog.

### Keying Conventions

This book uses the following keying conventions:

No prompt	When a command's format is the same for multiple platforms, the command prompt is not shown.
%	A percent sign represents the UNIX command shell prompt for a command that does not require root privileges.
#	A number sign represents the UNIX command shell prompt for a command that requires root privileges.
>	The notation > represents the MS-DOS or Windows command prompt.
...	Horizontal or vertical ellipses in format and syntax descriptions indicate that material has been eliminated to simplify a discussion.
[ ]	Brackets enclose optional items in format and syntax descriptions.
{ }	Braces enclose a list from which you must choose an item in format and syntax descriptions.
	In format and syntax descriptions, a vertical bar separates items in a list of choices enclosed in { } (braces).  In graphical user interface descriptions, a vertical bar separates menu commands (for example, select <b>File Open</b> ).

## PREFACE

# Installing the C++ SDK

---

**Overview**

This document explains how to install the C++ SDK component of Artix Mainframe 5.1.

---

**In this document**

This document discusses the following topics:

<a href="#">Before You Install</a>	<a href="#">page 12</a>
<a href="#">Installation Steps</a>	<a href="#">page 13</a>
<a href="#">Uninstalling Artix Mainframe C++ SDK</a>	<a href="#">page 20</a>
<a href="#">For More Information</a>	<a href="#">page 21</a>

---

# Before You Install

## Prerequisites

---

The z/OS-based component of Artix Mainframe 5.1 must already be installed and configured correctly. See the Artix Mainframe *Installation Guide* for more details. The C++ SDK component may then be installed on top of Artix Mainframe.

## C++ SDK installation details

---

The C++ SDK installation package consists of a SEQ file, `ARTIX.SDK.SEQ`, which is shipped as an `IEBCOPY` backup file that has been compressed using the TSO `XMIT` command.

---

# Installation Steps

## Overview

This subsection describes the steps you must follow to install the C++ SDK component of Artix Mainframe on z/OS.

**Note:** You must complete all the steps in this subsection in the order in which they are presented.

## Step 1: Preallocate a data set

Preallocate a z/OS sequential data set with the following information:

Space Units	Tracks
Primary quantity	1300
Secondary quantity	100
Directory blocks	0
Record format	FB
Record length	80
Block size	3120

## Step 2: Copy the ARTIX.SDK.SEQ file

Copy the `ARTIX.SDK.SEQ` file from your product CD into the z/OS data set that you preallocated in the preceding step. How you copy the file depends on the type of machine the CD-ROM drive is on. The most convenient way is to use FTP.

The following is an example of the FTP command sequence to transmit the `ARTIX.SDK.SEQ` file into the preallocated data set, where the CD drive letter is `d:` and `XXXX.XXXX` represents the name of the data set:

```
d:
ftp zOShost
ftp> binary
ftp> put ARTIX.SDK.SEQ 'XXXX.XXXX'
```

### Step 3: Unpack the PDS

---

After the `ARTIX.SDK.SEQ` file has been copied to z/OS, use the TSO `RECEIVE` command to unpack the PDS (where `XXXX.XXXX` represents the exact name of the PDS data set that is to be received):

```
RECEIVE INDSN('XXXX.XXXX')
```

Because the preceding command is a TSO command, you must enter it on an ISPF command screen.

You are prompted with restore parameters similar to the following:

```
To receive the Artix Mainframe C++ SDK PDS,
please specify the following:
DA('HLQ.ARTIX51.SDK.PDS') SPACE(1172,100) REL
replacing the HLQ as appropriate.
INMR901I Dataset HLQ.ARTIX51.SDK.PDS from JOE on NODENAME
INMR906A Enter restore parameters or 'DELETE' or 'END' +
```

**Note:** The high-level qualifier that you specify in this PDS name must match the high-level qualifier that you specified when installing Artix Mainframe. For example, if you chose `HLQ.ARTIX51` as the high-level qualifier for your Artix Mainframe installation, you must choose `HLQ.ARTIX51.SDK.PDS` in this case. The high-level qualifier can be up to 21 characters, including one or more periods.

The sequential data set, `XXXX.XXXX`, can now be deleted.

---

### Step 4: Expand the PDS

The `artixhlq.SDK.PDS($FIRST)` member contains JCL to expand the other PDS members into the full C++ SDK installation. You must use the same high-level qualifier that you specified when installing Artix Mainframe. To ensure that you are using the correct high-level qualifier, edit the `$FIRST` member, using the following command in ISPF (where `artixhlq` represents the high-level qualifier for your Artix Mainframe installation):

```
C 'HLQ.ARTIX51' 'artixhlq' ALL
```

Now submit `artixhlq.SDK.PDS($FIRST)` to install Artix Mainframe C++ SDK.

**Note:** This step might take several minutes to complete.

### Step 5: Customize the Artix default installation HLQ

You must ensure that the high-level qualifier used throughout the C++ SDK package (that is, in JCL member, PROCS, readmes, and configuration files) matches the high-level qualifier that you specified when installing Artix Mainframe.

The `artixhlq.SDK.PDS($SECOND)` member contains JCL to convert all the references of `HLQ.ARTIX51` in the C++ SDK package to match the high-level qualifier for your Artix Mainframe installation. To enable the `$SECOND` job to do this, perform the following steps:

1. Edit the `artixhlq.SDK.PDS($SECOND)` member, using the following command in ISPF:

```
C 'INSTALHLQ' 'artixhlq' ALL
```

In the preceding command, `artixhlq` must match the high-level qualifier that you specified when installing Artix Mainframe.

2. Submit `$SECOND` to convert all the references of `HLQ.ARTIX51` in the C++ SDK package to match the high-level qualifier for your Artix Mainframe installation.

### Step 6: Customize your locale (if necessary)

*This is only relevant if you want to run Artix in a locale other than the default locale IBM-1047, and your system and compiler are also running in a locale other than IBM-1047.*

Artix include files and demonstration sources are coded by default in locale IBM-1047. Follow these steps if you do not want to run Artix in the default IBM-1047 locale, and your system and compiler are also running in a locale other than IBM-1047:

1. In `artixhlq.SDK.PDS($THIRD)`, use the following command in ISPF to change the default high-level qualifier, to make it match your installation value (where `artixhlq` represents your high-level qualifier):

```
C 'HLQ.ARTIX51' 'artixhlq' ALL
```

2. In `artixhlq.SDK.PDS`, use the following command in ISPF to change the value of the `TO` variable, to make it match the locale codeset you want to use (where `IBM-xxx` represents your codeset):

```
C 'IBM-500' 'IBM-xxx' ALL
```

The preceding command lets you simultaneously change all occurrences of the default to make it match your codeset.

3. Submit `$THIRD` to convert the files to match your installation.

### Step 7: Set ITLOCALE and CPPLCALE (if necessary)

---

*This is only relevant if you want to run Artix Mainframe in a locale other than IBM-1047, and your system and compiler are running in a locale other than the locale in which you want to run Artix Mainframe.*

If you plan to run Artix Mainframe in a locale other than IBM-1047, and your system and compiler are running in a locale other than the locale in which you want to run Artix Mainframe, set the following variables in `artixhlg.PROCLIB(ARTXVARS)`:

ITLOCALE	This is the locale in which you want to run Artix Mainframe. For example, to have Artix Mainframe run in the Swiss German locale, set ITLOCALE as follows:  <code>SET ITLOCALE='LC_ALL=DE_CH.IBM-500'</code>  <b>Note:</b> If you have set the ITTIMEZ variable, you must include a comma before the ITLOCALE setting as follows:  <code>SET ITLOCALE=',LC_ALL=DE_CH.IBM-500'</code>
CPPLCALE	This is the locale in which you want to run the C++ compiler. For example, to have the C++ compiler run in the Swiss German locale, set CPPLCALE as follows:  <code>SET CPPLCALE='LOCALE(DE_CH.IBM-500)'</code>

See the “Passing Program Parameters” section of the *Artix Mainframe Administrator’s Guide* for more details of alternative methods of passing program parameters in Artix Mainframe.

**Step 8: Check installed data sets**

Compare your list of installed data sets with the list shown in [Table 1](#)

**Note:** These data sets are installed alongside those already installed with Artix Mainframe 5.1. If there are name clashes between low-level qualifiers, data sets installed with the C++ SDK component are prefixed with an SDK qualifier (for example, `artixhlq.LOADLIB` and `artixhlq.SDK.LOADLIB`).

**Table 1:** *Data sets installed with the C++ SDK*

Data Set	Description
<code>artixhlq.DEMOS.CPP.BLD.JCLLIB</code>	Sample JCL to run the IDL compiler and compile a C++ plug-in.
<code>artixhlq.DEMOS.CPP.GEN</code>	Used to store IDL-generated stub/skeleton code.
<code>artixhlq.DEMOS.CPP.H</code>	Contains sample include files.
<code>artixhlq.DEMOS.CPP.HH</code>	Used to store IDL-generated include files.
<code>artixhlq.DEMOS.CPP.LOADLIB</code>	Contains output executables.
<code>artixhlq.DEMOS.CPP.SRC</code>	Contains sample C++ source.

**Table 1:** *Data sets installed with the C++ SDK*

Data Set	Description
<code>artixhlg.INCLUDE.H</code>	Artix Mainframe C++ SDK API (IDL and C++)
<code>artixhlg.INCLUDE.IT@CAL.H</code>	
<code>artixhlg.INCLUDE.IT@DSA.CXX</code>	
<code>artixhlg.INCLUDE.IT@DSA.H</code>	
<code>artixhlg.INCLUDE.IT@ERR.H</code>	
<code>artixhlg.INCLUDE.IT@ITL.CXX</code>	
<code>artixhlg.INCLUDE.IT@ITL.H</code>	
<code>artixhlg.INCLUDE.IT@OSS.H</code>	
<code>artixhlg.INCLUDE.IT@TS.H</code>	
<code>artixhlg.INCLUDE.IT@TSDSA.H</code>	
<code>artixhlg.INCLUDE.OMG.H</code>	
<code>artixhlg.INCLUDE.OMG.HH</code>	
<code>artixhlg.INCLUDE.OMG.IDL</code>	
<code>artixhlg.INCLUDE.ORBIX.H</code>	
<code>artixhlg.INCLUDE.ORBIX.HH</code>	
<code>artixhlg.INCLUDE.ORBIX.IDL</code>	
<code>artixhlg.INCLUDE.ORBIX@PD.H</code>	
<code>artixhlg.INCLUDE.ORBIX@PD.HH</code>	
<code>artixhlg.INCLUDE.ORBIX@PD.IDL</code>	
<code>artixhlg.INCLUDE.ORBIX@SY.CXX</code>	
<code>artixhlg.INCLUDE.ORBIX@SY.H</code>	
<code>artixhlg.INCLUDE.ORBIX@XT.HH</code>	
<code>artixhlg.INCLUDE.ORBIX@XT.IDL</code>	

**Table 1:** *Data sets installed with the C++ SDK*

Data Set	Description
<code>artixhlq.SDK.CONFIG</code>	SDK-specific configuration to run the IDL compiler.
<code>artixhlq.SDK.LKED</code>	Artix Mainframe C++ SDK sidedecks.
<code>artixhlq.SDK.LOADLIB</code>	Additional Artix Mainframe C++ runtime DLLs.
<code>artixhlq.SDK.PROCLIB</code>	SDK-specific PROCs for running the IDL compiler and C++ compiler.

**Note:** The supplied IDL file, `SIMPLE`, is installed into the existing Artix Mainframe `artixhlq.DEMOS.IDL` PDS. If this data set has been removed, the `SIMPLE` IDL file is installed into a new `artixhlq.DEMOS.SDK.IDL` PDS instead.

## Step 9: Verifying your SDK installation

To verify that the C++ SDK has been installed correctly, the following sample jobs have been provided in `artixhlq.DEMOS.CPP.BLD.JCLLIB` to perform an IDL compile and an Artix C++ compile/link

IDL	This invokes the IDL procedure to perform an IDL-to-C++ compilation step, using the sample IDL source file that is located by default in <code>artixhlq.DEMOS.IDL(SIMPLE)</code> .
PLUGIN	This compiles some sample C++ source files and performs a pre-link/link to create an executable. This illustrates how one might build a custom C++ plug-in to be loaded by the transformer service component of Artix Mainframe.

---

# Uninstalling Artix Mainframe C++ SDK

---

## Steps

To uninstall Artix Mainframe C++ SDK, delete all the data sets that are listed in [Table 1](#) which starts on page 17 of this guide.

---

# For More Information

---

## Release Notes

See the Artix Mainframe *Release Notes* at:

[http://www.iona.com/support/docs/artix/mainframe/5.1/release\\_notes/release\\_notes.pdf](http://www.iona.com/support/docs/artix/mainframe/5.1/release_notes/release_notes.pdf)

---

## Knowledge base

Review IONA Knowledge Base articles for Artix at:

<http://www.iona.com/support/kb/index.jspa>

---

## Technical support

E-mail technical support with questions and suggestions at:

[support@iona.com](mailto:support@iona.com)

**For More Information**

# Index

## Symbols

\$FIRST 14  
\$SECOND 15  
\$THIRD 16

## A

ARTIX.SEQ 13, 14  
artixhlq.PDS(\$FIRST) 14  
artixhlq.PDS(\$SECOND) 15

## C

CPPLCALE 16

## D

data set  
    installed with Artix for z/OS 17  
    PDS 14  
    sequential 13

## F

FTP 13

## I

IBM-1047 locale 15  
IONA  
    knowledge base 21  
    technical support 21  
ISPF 14, 15  
ITLOCALE 16

## J

JCL 14, 15

## K

knowledge base 21

## P

PDS(\$FIRST) 14  
PDS(\$SECOND) 15  
PDS(\$THIRD) 15  
PDS data set 14  
preallocate 13

## R

RECEIVE 14  
Release Notes 21

## S

sequential data set 13  
support, IONA 21

## T

technical support 21  
TSO  
    RECEIVE 14

## U

uninstalling 20

