

# ArjunaCore 3.0

---

## Installation Guide

AC-IG-8/4/03



## **Legal Notices**

The information contained in this documentation is subject to change without notice.

Arjuna Technologies Limited makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Arjuna Technologies Limited 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.

Java™ and J2EE is a U.S. trademark of Sun Microsystems, Inc. Microsoft® and Windows NT® are registered trademarks of Microsoft Corporation. Oracle® is a registered U.S. trademark and Oracle9™, Oracle9 Server™ Oracle9 Enterprise Edition™ are trademarks of Oracle Corporation. Unix is used here as a generic term covering all versions of the UNIX® operating system. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

Arjuna is a trademark of Hewlett-Packard Company.

## **Software Version**

ArjunaCore 3.0

## **Restricted Rights Legend**

Use, duplication, or disclosure is subject to restrictions as set forth in contract subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software clause 52.227-FAR14.

Arjuna Technologies Limited  
Nanotechnology Centre  
Herschel Building  
Newcastle Upon Tyne  
NE1 7RU  
United Kingdom

© Copyright 2003 Arjuna Technologies Limited

# Contents

## **About This Guide ..... 5**

What This Guide Contains.....	5
Audience .....	5
Prerequisites.....	5
Organization .....	5
Documentation Conventions.....	5
Additional Documentation.....	6
Contacting Us .....	6

## **Installing ArjunaCore..... 7**

Preparing your system .....	7
ObjectStore management.....	7
Additional jar requirements .....	7
Setting properties .....	8
Licensing.....	8





# About This Guide

## What This Guide Contains

---

The Installation Guide contains information on how to use ArjunaCore 3.0.

## Audience

---

This guide is most relevant to engineers who are responsible for installing ArjunaCore 3.0 installations.

## Prerequisites

---

None.

## Organization

---

This guide contains the following chapters:

- **Chapter 1, Installing ArjunaCore:** This chapter contains information on the way to install ArjunaCore.

## Documentation Conventions

---

The following conventions are used in this guide:

Convention	Description
<i>Italic</i>	In paragraph text, italic identifies the titles of documents that are being referenced. When used in conjunction with the Code text described below, italics identify a variable that should be replaced by the user with an actual value.
<b>Bold</b>	Emphasizes items of particular importance.
Code	Text that represents programming code.
<b>Function   Function</b>	A path to a function or dialog box within an interface. For example, “Select File   Open.” indicates that you should select the Open function from the File menu.
( ) and	Parentheses enclose optional items in command syntax. The vertical bar separates syntax items in a list of choices. For example, any of the following three items can be entered in this syntax:  <code>persistPolicy (Never   OnTimer   OnUpdate   NoMoreOftenThan)</code>
<b>Note:</b>	A note highlights important supplemental information.
<b>Caution:</b>	A caution highlights procedures or information that is necessary to avoid damage to equipment, damage to software, loss of data, or invalid test results.

Table 1      Formatting Conventions

---

## Additional Documentation

---

In addition to this guide, the following guides are available in the ArjunaCore 3.0 documentation set:

- *ArjunaCore 3.0 Release Notes*: Provides late-breaking information about ArjunaCore 3.0.
- *ArjunaCore 3.0 Administration Guide*: This guide provides instructions for administering ArjunaCore 3.0.
- *ArjunaCore 3.0 Programmer’s Guide*: Provides guidance for writing applications.

---

## Contacting Us

---

Questions or comments about ArjunaCore 3.0 should be directed to our support team. Send email to [support@arjuna.com](mailto:support@arjuna.com).

# Installing ArjunaCore

## Preparing your system

---

Before installing the *ArjunaCore* software, we recommend the following administrative steps be taken, assuming a default configuration for *ArjunaCore*:

- Install the distribution into the required location.
- *ArjunaCore* requires a minimum object store for storing the outcome of transactions in the event of system crashes. The location of this should be specified in the properties file using the `com.arjuna.ats.arjuna.objectstore.objectStoreDir` environment variable, e.g., `java -Dcom.arjuna.ats.arjuna.objectstore.objectStoreDir=C:\temp foo`.
- By default, all object states will be stored within the `defaultStore` subdirectory of the object store root, e.g., `/usr/local/Arjuna/TransactionService/ObjectStore/defaultStore`. However, this subdirectory can be changed by setting the `com.arjuna.ats.arjuna.objectstore.localOSRoot` property variable accordingly.

## ObjectStore management

---

Within the transaction service installation, the object store is updated regularly whenever transactions are created, or when *Transactional Objects for Java* is used. In a failure free environment, the only object states which should reside within the object store are those representing objects created with the *Transactional Objects for Java* API. However, if failures occur, transaction logs may remain in the object store until crash recovery facilities have resolved the transactions they represent. As such it is very important that the contents of the object store are not deleted without due care and attention, as this will make it impossible to resolve in doubt transactions. In addition, if multiple users share the same object store it is important that they realise this and do not simply delete the contents of the object store assuming it is an exclusive resource.

## Additional jar requirements

---

In order to fully utilize all of the facilities available within *ArjunaCore*, it will be necessary to add all of the jar files contained in the `lib` directory of the distribution.

## Setting properties

---

*ArjunaCore* has been designed to be highly configurable at runtime through the use of various property attributes, which will be described in subsequent sections. Although these attributes can be provided at runtime on the command line, it is possible (and may be more convenient) to specify them through a single properties file. At runtime *ArjunaCore* looks for the file `arjuna-properties.xml` in the following order:

1. the current working directory, i.e., where the application was executed from.
2. the user's home directory.
3. the `etc` directory of the *ArjunaCore* installation; this must be placed in your CLASSPATH.

If found, all entries within this file will be added to the system properties. Obviously non-*ArjunaCore* specific properties can also be specified in this file. For example:

```
<property
  name="com.arjuna.ats.arjuna.coordinator.asyncCommit"
  value="NO"/>
<property
  name="com.arjuna.ats.arjuna.objectstore.objectStoreDir"
  value="c:\temp\ObjectStore"/>
```

The name of the properties file can be overridden at runtime by specifying a new file using the `com.arjuna.ats.arjuna.common.propertiesFile` attribute variable.

## Licensing

---

*ArjunaCore* requires a license key in order to operate. A valid license key must be provided to every *ArjunaCore* application at runtime as the property `ArjunaCore_LicenceKey`

The *Transactional Objects for Java* framework is separately licensed, as is the complete two-phase commit protocol engine. As such, certain licenses may only allow a single transactional resource to be registered with a transaction, or may not allow *Transactional Objects for Java* to be used.