

Oracle® Solaris Cluster Data Service for Apache Tomcat Guide

License Restrictions Warranty/Consequential Damages Disclaimer

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

Warranty Disclaimer

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Restricted Rights Notice

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Hazardous Applications Notice

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Trademark Notice

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group in the United States and other countries.

Third Party Content, Products, and Services Disclaimer

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. UNIX est une marque déposée concédée sous licence par X/Open Company, Ltd.

Contents

Preface	7
1 Installing and Configuring Oracle Solaris Cluster HA for Apache Tomcat	13
Installing and Configuring Oracle Solaris Cluster HA for Apache Tomcat	13
Oracle Solaris Cluster HA for Apache Tomcat	14
Planning the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration ...	15
Apache Tomcat and Solaris Containers	15
About Horizontal Scalability	15
Configuration Restrictions	17
Configuration Requirements	18
Installing and Configuring Apache Tomcat	22
▼ How to Enable Apache Tomcat to run in a Global Zone Configuration	23
▼ How to Install and Configure Apache Tomcat in the Global Zone	26
▼ How to Enable Apache Tomcat to run in a Zone Configuration	29
▼ How to Install and Configure Apache Tomcat in a Zone	32
▼ How to Enable Apache Tomcat to run in Failover Zone Configuration	34
▼ How to Install and Configure Apache Tomcat in a Failover Zone	36
Verifying the Installation and Configuration of Apache Tomcat	38
▼ How to Verify the Installation and Configuration of Apache Tomcat	38
▼ How to Deploy the Apache Tomcat Application	39
Installing the HA for Apache Tomcat Packages	40
▼ How to Install the HA for Apache Tomcat Packages	40
Registering and Configuring Oracle Solaris Cluster HA for Apache Tomcat	42
▼ How to Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a Failover Data Service	42
▼ How to Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a Multiple Masters Data Service	46
▼ How to Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a Scalable	

Data Service	50
Configuring Oracle Solaris Cluster HA for Apache Tomcat in Zones	54
Configuring Oracle Solaris Cluster HA for Apache Tomcat in a Failover Zone	54
Configuring Oracle Solaris Cluster HA for Apache Tomcat in a Zone	57
Verifying the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration ..	58
▼ How to Verify the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration	58
Understanding the Oracle Solaris Cluster Apache Tomcat HA Parameter File	59
Structure of the Apache Tomcat HA parameter file	59
Strategy to Choose the TestCmd and the ReturnString Variable	60
Understanding Oracle Solaris Cluster HA for Apache Tomcat Fault Monitor	61
Resource Properties	61
Probing Algorithm and Functionality	61
Debugging HA for Apache Tomcat	62
▼ How to Activate Debugging for HA for Apache Tomcat	62
 A Deployment Example: Installing Apache Tomcat in the Global Zone	65
Target Cluster Configuration	65
Software Configuration	65
Assumptions	66
Installing and Configuring Apache Tomcat on Global Storage in the Global Zone	66
▼ Example: Preparing the Cluster for Apache Tomcat	66
▼ Example: Configuring Cluster Resources for Apache Tomcat	67
▼ Example: Installing the Apache Tomcat Software on Shared Storage	67
▼ Example: Modifying the Apache Tomcat Configuration Files	68
▼ Example: Enabling the Apache Tomcat Software to Run in the Cluster	69
 B Deployment Example: Installing Apache Tomcat in a Failover Zone	71
Target Cluster Configuration	71
Software Configuration	71
Assumptions	72
Installing and Configuring Apache Tomcat on Global Storage in the Failover Zone	72
▼ Example: Preparing the Cluster for Apache Tomcat	72
▼ Example: Configuring Cluster Resources for Apache Tomcat	73
▼ Example: Configuring the Failover Zone	73

▼ Example: Installing the Apache Tomcat Software on Shared Storage	75
▼ Example: Modifying the Apache Tomcat Configuration Files	76
▼ Example: Enabling the Apache Tomcat Software to Run in the Cluster	76
C Deployment Example: Installing Apache Tomcat in a Non-Global Zone	77
Target Cluster Configuration	77
Software Configuration	77
Assumptions	78
Installing and Configuring Apache Tomcat on Shared Storage in a Non-Global Zone	78
▼ Example: Preparing the Cluster for Apache Tomcat	78
▼ Example: Configuring the Zone	78
▼ Example: Configuring Cluster Resources for Apache Tomcat	80
▼ Example: Installing the Apache Tomcat Software on Local Storage	80
▼ Example: Modifying the Apache Tomcat Configuration File	81
▼ Example: Enabling the Apache Tomcat Software to Run in the Cluster	82
Index	83

Preface

Oracle Solaris Cluster Data Service for Apache Tomcat Guide explains how to install and configure Oracle Solaris Cluster data services.

Note – This Oracle Solaris Cluster release supports systems that use the SPARC and x86 families of processor architectures: UltraSPARC, SPARC64, AMD64, and Intel 64. In this document, x86 refers to the larger family of 64-bit x86 compatible products. Information in this document pertains to all platforms unless otherwise specified.

This document is intended for system administrators with extensive knowledge of Oracle software and hardware. Do not use this document as a planning or presales guide. Before reading this document, you should have already determined your system requirements and purchased the appropriate equipment and software.

The instructions in this book assume knowledge of the Oracle Solaris Operating System and expertise with the volume-manager software that is used with Oracle Solaris Cluster software.

Using UNIX Commands

This document contains information about commands that are specific to installing and configuring Oracle Solaris Cluster data services. The document does *not* contain comprehensive information about basic UNIX commands and procedures, such as shutting down the system, booting the system, and configuring devices. Information about basic UNIX commands and procedures is available from the following sources:

- Online documentation for the Oracle Solaris Operating System
- Oracle Solaris Operating System man pages
- Other software documentation that you received with your system

Typographic Conventions

The following table describes the typographic conventions that are used in this book.

TABLE P-1 Typographic Conventions

Typeface	Meaning	Example
AaBbCc123	The names of commands, files, and directories, and onscreen computer output	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. <code>machine_name%</code> you have mail.
AaBbCc123	What you type, contrasted with onscreen computer output	<code>machine_name%</code> su Password:
<i>aabbcc123</i>	Placeholder: replace with a real name or value	The command to remove a file is <i>rm filename</i> .
<i>AaBbCc123</i>	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the <i>User's Guide</i> . <i>A cache</i> is a copy that is stored locally. Do <i>not</i> save the file. Note: Some emphasized items appear bold online.

Shell Prompts in Command Examples

The following table shows the default UNIX system prompt and superuser prompt for shells that are included in the Oracle Solaris OS. Note that the default system prompt that is displayed in command examples varies, depending on the Oracle Solaris release.

TABLE P-2 Shell Prompts

Shell	Prompt
Bash shell, Korn shell, and Bourne shell	\$
Bash shell, Korn shell, and Bourne shell for superuser	#
C shell	machine_name%
C shell for superuser	machine_name#

Related Documentation

Information about related Oracle Solaris Cluster topics is available in the documentation that is listed in the following table. All Oracle Solaris Cluster documentation is available at <http://www.oracle.com/technetwork/indexes/documentation/index.html>.

Topic	Documentation
Data service administration	<i>Oracle Solaris Cluster Data Services Planning and Administration Guide</i> Individual data service guides
Concepts	<i>Oracle Solaris Cluster Concepts Guide</i>
Software installation	<i>Oracle Solaris Cluster Software Installation Guide</i>
System administration	<i>Oracle Solaris Cluster System Administration Guide</i>
Hardware administration	<i>Oracle Solaris Cluster 3.3 Hardware Administration Manual</i> Individual hardware administration guides
Data service development	<i>Oracle Solaris Cluster Data Services Developer's Guide</i>
Error messages	<i>Oracle Solaris Cluster Error Messages Guide</i>
Command and function reference	<i>Oracle Solaris Cluster Reference Manual</i>

For a complete list of Oracle Solaris Cluster documentation, see the release notes for your release of Oracle Solaris Cluster at <http://www.oracle.com/technetwork/indexes/documentation/index.html>.

Related Third-Party Web Site References

Third-party URLs that are referenced in this document provide additional related information.

Note – Oracle is not responsible for the availability of third-party web sites mentioned in this document. Oracle does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Oracle will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Documentation and Support

See the following web sites for additional resources:

- [Documentation \(http://www.oracle.com/technetwork/indexes/documentation/index.html\)](http://www.oracle.com/technetwork/indexes/documentation/index.html)
- [Support \(http://www.oracle.com/us/support/systems/index.html\)](http://www.oracle.com/us/support/systems/index.html)

Oracle Software Resources

[Oracle Technology Network \(http://www.oracle.com/technetwork/index.html\)](http://www.oracle.com/technetwork/index.html) offers a range of resources related to Oracle software:

- Discuss technical problems and solutions on the [Discussion Forums \(http://forums.oracle.com\)](http://forums.oracle.com).
- Get hands-on step-by-step tutorials with [Oracle By Example \(http://www.oracle.com/technetwork/tutorials/index.html\)](http://www.oracle.com/technetwork/tutorials/index.html).

Getting Help

If you have problems installing or using Oracle Solaris Cluster, contact your service provider and provide the following information:

- Your name and email address (if available)
- Your company name, address, and phone number
- The model number and serial number of your systems
- The release number of the Oracle Solaris Operating System (for example, Oracle Solaris 10)
- The release number of Oracle Solaris Cluster (for example, Oracle Solaris Cluster 3.3)

Use the following commands to gather information about each node on your system for your service provider.

Command	Function
<code>prtconf -v</code>	Displays the size of the system memory and reports information about peripheral devices
<code>psrinfo -v</code>	Displays information about processors
<code>showrev -p</code>	Reports which patches are installed
<code>prtdiag -v</code>	Displays system diagnostic information
<code>/usr/cluster/bin/clnode show-rev</code>	Displays Oracle Solaris Cluster release and package version information

Also have available the contents of the `/var/adm/messages` file.

Installing and Configuring Oracle Solaris Cluster HA for Apache Tomcat

This chapter explains how to install and configure HA for Apache Tomcat.

This chapter contains the following sections.

- “Installing and Configuring Oracle Solaris Cluster HA for Apache Tomcat” on page 13
- “Oracle Solaris Cluster HA for Apache Tomcat” on page 14
- “Planning the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration” on page 15
- “Installing and Configuring Apache Tomcat” on page 22
- “Verifying the Installation and Configuration of Apache Tomcat” on page 38
- “Installing the HA for Apache Tomcat Packages” on page 40
- “Registering and Configuring Oracle Solaris Cluster HA for Apache Tomcat” on page 42
- “Configuring Oracle Solaris Cluster HA for Apache Tomcat in Zones” on page 54
- “Verifying the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration” on page 58
- “Understanding the Oracle Solaris Cluster Apache Tomcat HA Parameter File” on page 59
- “Understanding Oracle Solaris Cluster HA for Apache Tomcat Fault Monitor” on page 61
- “Debugging HA for Apache Tomcat” on page 62

Installing and Configuring Oracle Solaris Cluster HA for Apache Tomcat

Table 1–1 lists the tasks for installing and configuring Oracle Solaris Cluster HA for Apache Tomcat. Perform these tasks in the order they are listed.

TABLE 1–1 Task Map: Installing and Configuring Oracle Solaris Cluster HA for Apache Tomcat

Task	For Instructions, Go To
1. Plan the installation.	“Planning the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration” on page 15

TABLE 1-1 Task Map: Installing and Configuring Oracle Solaris Cluster HA for Apache Tomcat
(Continued)

Task	For Instructions, Go To
2. Install and configure Apache Tomcat.	“How to Install and Configure Apache Tomcat in the Global Zone” on page 26
3. Verify installation and configuration.	“How to Verify the Installation and Configuration of Apache Tomcat” on page 38
4. Install Oracle Solaris Cluster HA for Apache Tomcat packages.	“Installing the HA for Apache Tomcat Packages” on page 40
5.1 Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a failover data service.	“How to Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a Failover Data Service” on page 42
5.2 Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a multiple masters data service.	“How to Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a Multiple Masters Data Service” on page 46
5.3 Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a scalable data service.	“How to Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a Scalable Data Service” on page 50
5.4 Register and configure Oracle Solaris Cluster HA for Apache Tomcat in a failover zone.	“Configuring Oracle Solaris Cluster HA for Apache Tomcat in Zones” on page 54
6. Verify Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration.	“How to Verify the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration” on page 58
7. Understanding the Oracle Solaris Cluster Apache Tomcat HA parameter file.	“Understanding the Oracle Solaris Cluster Apache Tomcat HA Parameter File” on page 59
8. Understanding the Oracle Solaris Cluster HA for Apache Tomcat Fault Monitor.	“Understanding Oracle Solaris Cluster HA for Apache Tomcat Fault Monitor” on page 61
9. How to debug Oracle Solaris Cluster HA for Apache Tomcat.	“How to Activate Debugging for HA for Apache Tomcat” on page 62

Oracle Solaris Cluster HA for Apache Tomcat

Apache Tomcat acts as a servlet engine behind an Apache web server, or you can configure it as a standalone web server that includes the servlet engine.

Apache Tomcat is freely available under Apache Software License and can be downloaded from <http://jakarta.apache.org>.

The Oracle Solaris Cluster HA for Apache Tomcat data service provides a mechanism for orderly startup and shutdown, fault monitoring, and automatic failover of the Apache Tomcat service.

The following table describes the relation between the application components and the related Oracle Solaris Cluster data service.

TABLE 1–2 Protection of Components

Component	Protected by
Apache Tomcat	Oracle Solaris Cluster HA for Apache Tomcat
Failover Zone	Oracle Solaris Cluster HA for Containers

Planning the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration

This section contains the information you need to plan your Oracle Solaris Cluster HA for Apache Tomcat installation and configuration.

Apache Tomcat and Solaris Containers

Oracle Solaris Cluster HA for Apache Tomcat is supported in Solaris Containers, Oracle Solaris Cluster is offering two concepts for Solaris Containers.

- Zones are containers which are running after a reboot of the node. These containers, combined with resource groups having the nodename *nodename:zonename* as a valid “nodename” in the resource groups nodename list.
- Failover Zone containers are managed by the Solaris Container agent, and are represented by a resource of a resource group.

About Horizontal Scalability

Oracle Solaris Cluster includes a concept of horizontal scalability for data services called scalable service. IP based load-balancing algorithms are integrated in this concept. Because of this reason you can scale horizontally without using hardware loadbalancers. For a more detailed discussion of this scalable service see [Oracle Solaris Cluster Concepts Guide](#).

Before using the Oracle Solaris Cluster HA for Apache Tomcat in a scalable configuration, it is necessary to have a closer look at the infrastructure of the cluster and the clients.

If your clients access your application using proxies, determine whether the used proxy stays the same during a session context. This is true for an intranet.

If the proxy changes during a session context it means, from the load-balancing point of view, that the source IP address is changing. This will spoil every IP based load-balancing, whether it is hardware or software.

Especially when your client accesses the server over the Internet, it is *not* guaranteed, that the source IP address remains the same during a session context.

Oracle Solaris Cluster HA for Apache Tomcat can be configured two ways to get horizontal scalability.

1. Using Oracle Solaris Cluster HA for Apache Tomcat in a purely scalable configuration.
2. Using Oracle Solaris Cluster HA for Apache Tomcat in a multiple masters configuration.

The difference between the two configuration is in the way of accessing the nodes.

- A client accesses the scalable configuration by its shared address. In this case, Oracle Solaris Cluster does the load-balancing.
- A client accesses a multiple masters configuration using each node's physical address. Load-balancing must be done outside of the cluster.

For more information about scalable data services, see *Oracle Solaris Cluster Concepts Guide*.

Both configuration options are discussed in the sections “Oracle Solaris Cluster HA for Apache Tomcat as a scalable configuration” on page 16 and “Oracle Solaris Cluster HA for Apache Tomcat as a multiple master configuration” on page 17.

Oracle Solaris Cluster HA for Apache Tomcat as a scalable configuration

You can use Oracle Solaris Cluster HA for Apache Tomcat in a scalable configuration if it is guaranteed, that the source IP address remains the same during a session context. This guarantee is achieved in example in an intranet.

If the source IP address might change during a session context and a scalable configuration is required, Apache Tomcat needs to be configured with session replication. This can be done by the application using a global file system or a database.



Caution – You will get a performance penalty from this approach.

You will get better performance using Tomcats inbound memory session replication for a scalable configuration with changing source IP addresses.

- You can achieve the session replication for Apache Tomcat 4.x by following the instructions at <http://www.theserverside.com/news/1365188/Clustering-Technologies-In-Memory-Session-Replication-in-Tomcat-4>.

- For Apache Tomcat 5.x, inbound memory session replication as a built-in feature, is on the feature list.
- For Apache Tomcat 3.x the inbound memory session replication is not configurable.

Oracle Solaris Cluster HA for Apache Tomcat as a multiple master configuration

You can use Oracle Solaris Cluster HA for Apache Tomcat in a multiple master configuration in the other scenarios. External load-balancing is required. A typical configuration uses the scalable Apache web server as a load balancer and configures one physical host name of an Apache Tomcat instance behind each instance of the Apache web server. Another option is to use a hardware load balancer, which handles the session context.

The difference between scalable and multiple masters configuration is only in the way, the clients access the cluster nodes. In a scalable configuration, they access the shared address. Otherwise the clients access the physical zone or hostnames.

Configuration Restrictions

The configuration requirements in this section apply only to Apache Tomcat.



Caution – If your data service configuration does not conform to these requirements, the data service configuration might not be supported.

Restriction to deploy Oracle Solaris Cluster HA for Apache Tomcat in a scalable configuration

Deploy a scalable Oracle Solaris Cluster HA for Apache Tomcat configuration only if either session replication, or reliable source IP addresses are achieved. Otherwise the behavior of the application becomes unpredictable.

Restriction for the Load_balancing_policy

Setting the resource parameter `Load_balancing_policy` to `LB_STICKY` is strictly required, if Oracle Solaris Cluster HA for Apache Tomcat is deployed in a scalable configuration with reliable source IP addresses when no session replication is configured. Otherwise, the behavior of the application becomes unpredictable. In every other scalable configuration the `Sticky Load_balancing_policy` helps to get the more cache hits out of your caches.

Restriction for Scalable Services and Solaris Containers

Oracle Solaris Cluster HA for Apache Tomcat can be deployed in scalable configurations in Solaris Containers only if you use the zone features of Oracle Solaris Cluster.

Restriction for the Apache Tomcat smf Service Name in a Failover Zone

The Apache Tomcat configuration in a failover zone uses the smf component of Oracle Solaris Cluster HA for Solaris Containers. The registration of the Apache Tomcatdata service in a failover zone defines an smf service to control the Apache Tomcat database. The name of this smf service is generated in this naming scheme:

`svc:/application/sczone-agents:resource-name`. No other smf service with exactly this name can exist.

The associated smf manifest is automatically created during the registration process in this location and naming scheme:

`/var/svc/manifest/application/sczone-agents/resource-name.xml`. No other manifest can coexist with this name.

Restriction for Apache Tomcat 5.5.28, 6.0.29, and 7.06

Starting with Apache Tomcat versions 5.5.28, 6.0.29, and 7.06, you must use the wget probe algorithm. This bypasses the normal probe, enabling the `TestUrl` parameter to work correctly. You can choose to use the wget probe algorithm for earlier Apache Tomcat versions, or continue to use the original mconnect probe algorithm.

Configuration Requirements

These requirements apply to Oracle Solaris Cluster HA for Apache Tomcat only. You must meet these requirements before you proceed with your Oracle Solaris Cluster HA for Apache Tomcat installation and configuration.



Caution – Your data service configuration might not be supported if you do not adhere to these requirements.

Location of the Tomcat Home Directory for Scalable or Multiple Masters Configurations

If you intend to install Apache Tomcat in a scalable resource group, create the Tomcat Home directory and its dynamic data on local storage.

This required because Apache Tomcat uses the directory structure to store its configuration, logs, deployed applications and so on.

If your local storage is not big enough, you can use a Global File System on the shared storage.



Caution – In this scenario, the deployment of Tomcat applications needs to occur on every node where Apache Tomcat is hosted.

Load Balancing for Multiple Master Configurations

If you intend to install Apache Tomcat in a multiple master configuration, an external load balancer is required.

Location of the Tomcat Home Directory for Failover Configurations

If you intend to install Apache Tomcat in a failover resource group, create the Tomcat Home directory on the shared storage. The location for the Tomcat Home directory can reside on a Global File System (GFS) or it can reside on a Failover File System (FFS) with an HAStoragePlus resource. It is best practice to store it on a FFS.

This is required because Apache Tomcat uses the directory structure to store its configuration, logs, deployed applications, and so on. It is not recommended to store the binaries local and the dynamic parts of the data on the shared storage.

Note – It is best practice to mount Global File Systems with the `/global` prefix and to mount Failover File Systems with the `/local` prefix.

Location of the wget Command for Apache Tomcat 5.5.28, 6.0.29, and 7.06

To support the wget probe algorithm for Apache Tomcat 5.5.28, 6.0.29, and 7.06, the wget command must be available from every node of the cluster.

Apache Tomcat Component Dependencies

You can configure the Oracle Solaris Cluster HA for Apache Tomcat data service to protect one or more Apache Tomcat instances. Each instance needs to be covered by one Apache Tomcat resource. The dependencies between the Apache Tomcat resource and other needed resources are described in [Table 1–3](#), [Table 1–4](#), [Table 1–5](#), or [Table 1–6](#).

TABLE 1–3 Dependencies Between HA for Apache Tomcat Components in Failover Configurations

Component	Dependency
Apache Tomcat resource in the global zone or a zone.	SUNW.HAStoragePlus This dependency is required only, if the configuration uses a failover file system or file systems in a zone.
	SUNW.LogicalHostName

TABLE 1-3 Dependencies Between HA for Apache Tomcat Components in Failover Configurations
(Continued)

Component	Dependency
Apache Tomcat resource in a failover zone.	Oracle Solaris Cluster HA for the Solaris Container boot resource. SUNW.HAStoragePlus SUNW.LogicalHostName — This dependency is required only if the zones boot resource does not manage the zone's IP address.

Because of the special requirements of a scalable configuration, you need no dependencies to storage or addresses, as long as every Apache Tomcat and its parameter file `pf file` is stored on the root file system. Otherwise follow [Table 1-4](#).

TABLE 1-4 Dependencies Scalable

Component	Description
Storage Resource	This resource can be a SUNW.HAStorage or a SUNW.HAStoragePlus resource type.
Apache Tomcat (Mandatory)	→ Storage resource The Storage resource manages the Apache Tomcat File System Mount points and ensures that Apache Tomcat is not started until they are mounted.
Shared address (Mandatory)	The shared address resource resides in a separate resource group on which the scalable resource group depends.

A multiple master configuration is running on more than one nodes like a scalable configuration, but without a shared address. Because of the special requirements of a multiple masters configuration, you need no dependencies to storage or addresses, as long as every Apache Tomcat and its parameter file `pf file` is stored on the root file system. Otherwise follow [Table 1-5](#).

TABLE 1-5 Dependencies Between HA for Apache Tomcat Components in Failover Configurations

Component	Dependency
Apache Tomcat resource in the global zone	SUNW.HAStoragePlus. This dependency is required only, if the configuration uses a failover file system.
Apache Tomcat resource in a failover zone.	Oracle Solaris Cluster HA for the Solaris Container boot resource. SUNW.HAStoragePlus.

If more elaborate dependencies are required, see the `r_properties(5)` and `rg_properties(5)` man pages for further dependencies and affinities settings.

Note – For more detailed information about Apache Tomcat, refer to the <http://jakarta.apache.org> web page.

A SUNW.HAStoragePlus resource requires a Resource_offline_restart dependency type, while all other resources require a strong dependency type called Resource_dependencies. You must define the Resource_offline_restart dependency for the SUNW.HAStoragePlus resource if the resource type version is at least version 9. See Table 1–6 for the dependency type needed for each resource. The following resources are examples and you should evaluate the dependency to other resources on a case-by-case basis.

TABLE 1–6 Dependency Types for HA for Apache Tomcat Resources

Resource Name	Dependency Type
SUNW.HAStoragePlus	Resource_offline_restart
SUNW.HAStorage	Resource_dependencies
SUNW.LogicalHostName	Resource_dependencies

Apache Tomcat Configuration and Registration Files

Apache Tomcat component has configuration and registration files in the directory /opt/SUNWsctomcat/util. These files let you register the Apache Tomcat component with Oracle Solaris Cluster.

Within these files, the appropriate dependencies have been applied.

```
# cd /opt/SUNWsctomcat
#
# ls -l util
total 4
-rwx--x--- 1 root    other      709 Apr 29 11:57 sctomcat_config
-rwx--x--- 1 root    other      561 Apr 29 11:58 sctomcat_register
# more util/*g
::::::::::::
util/sctomcat_config
::::::::::::
#
# Copyright 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#

#ident    "@(#)sctomcat_config.ksh 1.2      06/03/08"

# This file will be sourced in by sctomcat_register and the parameters
# listed below will be used.
#
# These parameters can be customized in (key=value) form
#
#      RS - name of the resource for the application
```

```
#      RG - name of the resource group containing RS
#      PORT - name of the port number
#      LH - name of the LogicalHostname SC resource
# SCALABLE - true for a scalable resource or false for a failover resource
# NETWORK - false or true, false for multiple master configurations without
#            shared address, in this case SCALABLE will be ignored
#
#      PFILE - absolute path to the parameter file for the Tomcat resource
#      HAS_RS - name of the HASStoragePlus SC resource
#              (it can be a , separated list for the dependencies)
#
# The following variables need to be set only if the agent runs in a
# local zone
#      ZONE - the zone name where the Apache Tomcat should run in
#            Optional
#      ZONEBT - The resource name which controls the zone.
#            Optional
#      PROJECT - A project in the zone, that will be used for this service
#               specify it if you have an su - in the start stop or probe,
#               or to define the smf credentials. If the variable is not set,
#               it will be translated as :default for the smf manifest
#            Optional

RS=
RG=
PORT=
LH=
NETWORK=false
SCALABLE=false
PFILE=
HAS_RS=

# local zone specific options

ZONE=
ZONE_BT=
PROJECT=
```

Installing and Configuring Apache Tomcat

This section contains the procedures you need to install and configure Apache Tomcat.

Note – For more information about Apache Tomcat, refer to the <http://jakarta.apache.org> web page.

Determine how Apache Tomcat will be deployed in Oracle Solaris Cluster:

- Determine which version of Apache Tomcat will be deployed.
- Determine how many Apache Tomcat instances will be deployed.
- Determine which Cluster File System will be used by each Apache Tomcat instance.
- Determine the type of the target zone where you will install Apache Tomcat. Valid zone types are, the global zone, the failover zone, or a zone.

To install and configure Apache Tomcat in a *global zone* configuration, complete the following tasks:

- [“How to Enable Apache Tomcat to run in a Global Zone Configuration” on page 23](#)
- [“How to Install and Configure Apache Tomcat in the Global Zone” on page 26](#)

To install and configure Apache Tomcat in a zone configuration, complete the following tasks:

- [“How to Enable Apache Tomcat to run in a Zone Configuration” on page 29](#)
- [“How to Install and Configure Apache Tomcat in a Zone” on page 32](#)

To install and configure Apache Tomcat in a failover zone configuration, complete the following tasks:

- [“How to Enable Apache Tomcat to run in Failover Zone Configuration” on page 34](#)
- [“How to Install and Configure Apache Tomcat in a Failover Zone” on page 36](#)

You will find installation examples for each zone type in:

- [Appendix A, “Deployment Example: Installing Apache Tomcat in the Global Zone”](#)
- [Appendix B, “Deployment Example: Installing Apache Tomcat in a Failover Zone”](#)
- [Appendix C, “Deployment Example: Installing Apache Tomcat in a Non-Global Zone”](#)

▼ How to Enable Apache Tomcat to run in a Global Zone Configuration

- 1 **Become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization on one of the nodes in the cluster that will host Apache Tomcat.**

- 2 **Register the `SUNW.gds` and `SUNW.HAStoragePlus` resource type.**

```
# clresourcetype register SUNW.gds SUNW.HAStoragePlus
```

- 3 **Create a failover resource group.**

Note – Perform this step only if you create a failover data service.

```
# clresourcegroup create ApacheTomcat-failover-resource-group
```

- 4 **(Optional) Create a resource for the Apache Tomcat Disk Storage.**

Note – Perform this step only if you create a failover data service.

```
# clresource create \
> -g Apache-Tomcat-failover-resource-group \
> -t SUNW.HAStoragePlus \
> -p FilesystemMountPoints=Apache Tomcat- instance-mount-points Apache-Tomcat-has-resource
```

5 (Optional) Create a resource for the Apache Tomcat Logical Hostname.

Note – Perform this step for a failover data service.

```
# clreslogicalhostname \
> -g Apache Tomcat-failover-resource-group \
> -h Apache Tomcat-logical-hostname \
> Apache Tomcat-lh-resource
```

6 Enable the failover resource group that now includes the Apache Tomcat Disk Storage and Logical Hostname resources.

Note – Perform this step for a failover data service.

```
# clresourcegroup online -M -n current-node Apache Tomcat-failover-resource-group
```

7 (Optional) Create the resource group for the multiple masters data service.

Note – Perform this step only if you create a multiple masters data service.

```
# clresourcegroup create \
> -p Maximum_primaries=2 \
> -p Desired_primaries=2 \
> Apache-Tomcat-Scalable-resource-group
```

If you need more nodes, adjust Maximum_primaries and Desired_primaries to the appropriate value.

8 (Optional) Create a resource for the Apache Tomcat Disk Storage if it is not in the root file system.

Note – Perform this step only if you create a multiple masters data service.

```
# clresource create \
> -g Apache-Tomcat-failover-resource-group \
> -t SUNW.HAStoragePlus \
> -p FilesystemMountPoints=Apache Tomcat- instance-mount-points Apache-Tomcat-has-resource
```

9 Enable the failover resource group that now includes the Apache Tomcat Disk Storage and Logical Hostname resources.

Note – Perform this step only if you create a multiple masters data service.

```
# clresourcegroup online -M Apache Tomcat-failover-resource-group
```

10 Create a failover resource group for the SharedAddress resource.

Note – Perform this step only if you create a scalable data service.

```
# clresourcegroup create Apache-Tomcat-SharedAddress-resource-group
```

11 Create the SharedAddress resource.

Note – Perform this step only if you create a scalable data service.

```
# clressharedaddress create \
> -g Apache-Tomcat-SharedAddress-resource-group \
> -h Apache-Tomcat-SharedAddress-hostname \
> Apache-Tomcat-SharedAddress-resource
```

12 Online the SharedAddress resource group.

Note – Perform this step only if you create a scalable data service.

```
# clresourcegroup online -M Apache-Tomcat-SharedAddress-resource-group
```

13 Create the resource group for the scalable service.

Note – Perform this step only if you create a scalable data service.

```
# clresourcegroup create \
> -p Maximum_primaries=2 \
> -p Desired_primaries=2 \
> -p RG_dependencies=Apache-Tomcat-SharedAddress-resource-group \
> Apache-Tomcat-Scalable-resource-group
```

If you need more nodes, adjust *Maximum_primaries* and *Desired_primaries* to the appropriate value.

14 Create a resource for the Apache Tomcat Disk Storage if it is not in the root file system.

Note – Perform this step only if you create a scalable data service.

```
# clresource create Apache-Tomcat-has-resource \
> -g Apache-Tomcat-scalable-group \
> -t SUNW.HASStoragePlus \
> -p FilesystemMountPoints=Apache Tomcat- instance-mount-points
```

- 15 Enable the failover resource group that now includes the Apache Tomcat Disk Storage and Logical Hostname resources.

Note – Perform this step only if you create a scalable data service.

```
# clresourcegroup online -M Apache Tomcat-resource-group
```

▼ How to Install and Configure Apache Tomcat in the Global Zone

- 1 Determine how Apache Tomcat will be deployed in Oracle Solaris Cluster
 - Determine whether you will use Apache Tomcat as a failover a multiple master, or a scalable data service.
For conceptual information on scalable and failover data services, see [Oracle Solaris Cluster Concepts Guide](#).
 - Determine which user name will run Apache Tomcat.
 - Determine how many Apache Tomcat versions and instances will be deployed.
 - If more than one instance of a version will be deployed, determine whether they share the binaries.
 - Determine which Cluster File System will be used by each Apache Tomcat instance.
- 2 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.
- 3 Create user and group if required.



Caution – In this scenario, the deployment of Tomcat group and user needs to occur on every node where Apache Tomcat is hosted.

If Apache Tomcat is to run under a non root user, you have to create the appropriate user, and the appropriate group. For these tasks use the following commands.

a. Create the group by running the following command:

```
# groupadd -g 1000 tomcat
```

b. Create the user by running the following command:

```
# useradd -u 1000 -g 1000 -d /global/tomcat -s /bin/ksh tomcat
```

4 If you are not logged in as root, switch to the appropriate user name.

```
# su - user-name
```

5 Install Apache Tomcat.

If you deploy Apache Tomcat as a failover data service, install Apache Tomcat onto a shared file system within Oracle Solaris Cluster.

It is recommended that you install Apache Tomcat onto shared disks. For a discussion of the advantages and disadvantages of installing the software on a local versus a cluster file system, see “[Determining the Location of the Application Binaries](#)” in *Oracle Solaris Cluster Data Services Planning and Administration Guide*.

If you will deploy Apache Tomcat as a failover data service install the Apache Tomcat binaries on the shared storage on **one** node. If Apache Tomcat will be deployed as a scalable or a multiple master data service, install the Apache Tomcat binaries on the local storage on **every** node, that will host the Apache Tomcat data service.

Note – Refer to <http://tomcat.apache.org/index.html> for instructions about installing Apache Tomcat.

If you deploy Apache Tomcat as a scalable data service for a scalable or a multiple masters configuration, repeat the following step at every node that will host Apache Tomcat.

6 Create the environment script.

Create an Korn shell or a C shell script (dependent on the login-shell of your Apache Tomcat user name) to set the environment variables for Apache Tomcat. You must set the environment variables in a shell script and not in the users profile.

With this mechanism you can install and run multiple Apache Tomcat versions and instances under one user name.



Caution – These shell scripts must be available on every node that can host the Apache Tomcat data service. For a failover configuration, store them on the shared storage of the node or in the target zone. For a scalable or a multiple masters configuration, store them on the local file system of every node or on the shared storage. These scripts must not be different on the various nodes.

Example 1–1 Korn shell script to set the environment variables for Apache Tomcat 3.x/3.3

```
# more env.ksh
#!/usr/bin/ksh
#
# Environment for Tomcat
#
JAVA_HOME=/usr/j2se
export JAVA_HOME
JAKARTA_HOME=/global/mnt1/jakarta-3.3
export JAKARTA_HOME
TOMCAT_HOME=$JAKARTA_HOME
export TOMCAT_HOME
```

Example 1–2 C shell script to set the environment variables for Apache Tomcat 3.3/3.3

```
# more env.csh
#!/usr/bin/csh
#
# Environment for Tomcat
#

setenv JAVA_HOME /usr/j2se
setenv JAKARTA_HOME /global/mnt1/jakarta-3.3
setenv TOMCAT_HOME $JAKARTA_HOME
```

Example 1–3 Korn shell script to set the environment variables for Apache Tomcat 4.1.x

```
# more env.ksh
#!/usr/bin/ksh
#
# Environment for Tomcat
#
JAVA_HOME=/usr/j2se
export JAVA_HOME
CATALINA_HOME=/global/mnt1/jakarta-tomcat-4.1.18
export CATALINA_HOME
```

Example 1–4 C shell script to set the environment variables for Apache Tomcat 4.1.x

```
# more env.csh
#!/usr/bin/csh
#
# Environment for Tomcat
```

```
#
setenv JAVA_HOME /usr/j2se
setenv CATALINA_HOME /global/mnt1/jakarta-tomcat-4.1.18
```

The environment variables are version and configuration dependent.

▼ How to Enable Apache Tomcat to run in a Zone Configuration

- 1 Become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization on one of the nodes in the cluster that will host Apache Tomcat.
- 2 Create and boot your zone *Apache Tomcat-zone* on all the nodes to host your Apache Tomcat data base.
- 3 Register the *SUNW.gds* and *SUNW.HAStoragePlus* resource type.

```
# clresourcetype register SUNW.gds SUNW.HAStoragePlus
```
- 4 Create a failover resource group.

Note – Perform this step only if you create a failover data service.

```
# clresourcegroup create -n node1:Apache Tomcat-zone,node2:Apache Tomcat-zone \
Apache Tomcat-failover-resource-group
```

- 5 (Optional) Create a resource for the Apache Tomcat Disk Storage.

Note – Perform this step only if you create a failover data service.

```
# clresource create \
> -g Apache-Tomcat-failover-resource-group \
> -t SUNW.HAStoragePlus \
> -p FilesystemMountPoints=Apache Tomcat- instance-mount-points Apache-Tomcat-has-resource
```

- 6 (Optional) Create a resource for the Apache Tomcat Logical Hostname.

Note – Perform this step for a failover data service.

```
# clreslogicalhostname \
> -g Apache Tomcat-failover-resource-group \
> -h Apache Tomcat-logical-hostname \
> Apache Tomcat-lh-resource
```

- 7 Enable the failover resource group that now includes the Apache Tomcat Disk Storage and Logical Hostname resources.**

Note – Perform this step for a failover data service.

```
# clresourcegroup online -M -n current-node Apache Tomcat-failover-resource-group
```

- 8 (Optional) Create the resource group for the multiple masters data service.**

Note – Perform this step only if you create a multiple masters data service.

```
# clresourcegroup create \  
> -p Maximum_primaries=2 \  
> -p Desired_primaries=2 \  
> -n node1:Apache Tomcat-zone,node2:Apache Tomcat-zone\  
> Apache-Tomcat-Scalable-resource-group
```

If you need more nodes, adjust Maximum_primaries and Desired_primaries to the appropriate value.

- 9 (Optional) Create a resource for the Apache Tomcat Disk Storage if it is not in the root file system.**

Note – Perform this step only if you create a multiple masters data service.

```
# clresource create \  
> -g Apache-Tomcat-failover-resource-group \  
> -t SUNW.HASStoragePlus \  
> -p FilesystemMountPoints=Apache Tomcat- instance-mount-points Apache-Tomcat-has-resource
```

- 10 Enable the failover resource group that now includes the Apache Tomcat Disk Storage and Logical Hostname resources.**

Note – Perform this step only if you create a multiple masters data service.

```
# clresourcegroup online -M Apache Tomcat-failover-resource-group
```

- 11 Create a failover resource group for the SharedAddress resource.**

Note – Perform this step only if you create a scalable data service.

```
# clresourcegroup create -n node1:Apache Tomcat-zone,node2:Apache Tomcat-zone\  
> Apache-Tomcat-SharedAddress-resource-group
```

12 Create the SharedAddress resource.

Note – Perform this step only if you create a scalable data service.

```
# clressharedaddress create \
> -g Apache-Tomcat-SharedAddress-resource-group\
> -h Apache-Tomcat-SharedAddress-hostname \
> Apache-Tomcat-SharedAddress-resource
```

13 Online the SharedAddress resource group.

Note – Perform this step only if you create a scalable data service.

```
# clresourcegroup online -M Apache-Tomcat-SharedAddress-resource-group
```

14 Create the resource group for the scalable service.

Note – Perform this step only if you create a scalable data service.

```
# clresourcegroup create \
> -p Maximum_primaries=2 \
> -p Desired_primaries=2 \
> -p RG_dependencies=Apache-Tomcat-SharedAddress-resource-group \
-n node1:Apache Tomcat-zone,node2:Apache Tomcat-zone\
Apache-Tomcat-Scalable-resource-group
```

If you need more nodes, adjust Maximum_primaries and Desired_primaries to the appropriate value.

15 Create a resource for the Apache Tomcat Disk Storage if it is not in the root file system.

Note – Perform this step only if you create a scalable data service.

```
# clresource create Apache-Tomcat-has-resource \
> -g Apache-Tomcat-scalable-group \
> -t SUNW.HAStoragePlus \
> -p FilesystemMountPoints=Apache Tomcat- instance-mount-points
```

16 Enable the failover resource group that now includes the Apache Tomcat Disk Storage and Logical Hostname resources.

Note – Perform this step only if you create a scalable data service.

```
# clresourcegroup online -M Apache Tomcatscalable-resource-group
```

▼ How to Install and Configure Apache Tomcat in a Zone

1 Determine how Apache Tomcat will be deployed in Oracle Solaris Cluster

- Determine whether you will use Apache Tomcat as a failover a multiple master, or a scalable data service.
For conceptual information on scalable and failover data services, see [Oracle Solaris Cluster Concepts Guide](#).
- Determine which user name will run Apache Tomcat.
- Determine how many Apache Tomcat versions and instances will be deployed.
- If more than one instance of a version will be deployed, determine whether they share the binaries.
- Determine which Cluster File System will be used by each Apache Tomcat instance.

2 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.

3 Enter the target zone

```
# zlogin tomcat-zone
```

4 Create user and group if required.



Caution – In this scenario, the deployment of Tomcat applications needs to occur on every zone where Apache Tomcat is hosted.

If Apache Tomcat is to run under a non root user, you have to create the appropriate user, and the appropriate group. For these tasks use the following commands.

a. Create the group by running the following command:

```
# groupadd -g 1000 tomcat
```

b. Create the user by running the following command:

```
# useradd -u 1000 -g 1000 -d /global/tomcat -s /bin/ksh tomcat
```

5 If you are not logged in as root, switch to the appropriate user name.

```
# su - user-name
```

6 Install Apache Tomcat.

If you deploy Apache Tomcat as a failover data service, install Apache Tomcat onto a shared file system within Oracle Solaris Cluster.

It is recommended that you install Apache Tomcat onto shared disks. For a discussion of the advantages and disadvantages of installing the software on a local versus a cluster file system, see “[Determining the Location of the Application Binaries](#)” in *Oracle Solaris Cluster Data Services Planning and Administration Guide*.

If you will deploy Apache Tomcat as a failover data service install the Apache Tomcat binaries on the shared storage on **one** zone. If Apache Tomcat will be deployed as a scalable or a multiple master data service, install the Apache Tomcat binaries on the local storage in **every** target zone, that will host the Apache Tomcat data service.

Note – Refer to <http://tomcat.apache.org/index.html> for instructions about installing Apache Tomcat.

If you deploy Apache Tomcat as a scalable data service for a scalable or a multiple masters configuration, repeat the following step at every target zone that will host Apache Tomcat.

7 Create the environment script.

Create an Korn shell or a C shell script (dependent on the login-shell of your Apache Tomcat user name) to set the environment variables for Apache Tomcat. You must set the environment variables in a shell script and not in the users profile.

With this mechanism you can install and run multiple Apache Tomcat versions and instances under one user name.



Caution – These shell scripts must be available on every node that can host the Apache Tomcat data service. For a failover configuration, store them on the shared storage in the target zone. For a scalable or a multiple masters configuration, store them on the local file system or on the shared storage of every target zone. These scripts must not be different on the various zones.

Example 1–5 Korn shell script to set the environment variables for Apache Tomcat 3.x/3.3

```
# more env.ksh
#!/usr/bin/ksh
#
# Environment for Tomcat
#
JAVA_HOME=/usr/j2se
export JAVA_HOME
JAKARTA_HOME=/global/mnt1/jakarta-3.3
export JAKARTA_HOME
TOMCAT_HOME=$JAKARTA_HOME
export TOMCAT_HOME
```

Example 1–6 C shell script to set the environment variables for Apache Tomcat 3.3/3.3

```
# more env.csh
#!/usr/bin/csh
#
```

```
# Environment for Tomcat
#

setenv JAVA_HOME /usr/j2se
setenv JAKARTA_HOME /global/mnt1/jakarta-3.3
setenv TOMCAT_HOME $JAKARTA_HOME
```

Example 1–7 Korn shell script to set the environment variables for Apache Tomcat 4.1.x

```
# more env.ksh
#!/usr/bin/ksh
#
# Environment for Tomcat
#
JAVA_HOME=/usr/j2se
export JAVA_HOME
CATALINA_HOME=/global/mnt1/jakarta-tomcat-4.1.18
export CATALINA_HOME
```

Example 1–8 C shell script to set the environment variables for Apache Tomcat 4.1.x

```
# more env.csh
#!/usr/bin/csh
#
# Environment for Tomcat
#
setenv JAVA_HOME /usr/j2se
setenv CATALINA_HOME /global/mnt1/jakarta-tomcat-4.1.18
```

The environment variables are version and configuration dependent.

▼ How to Enable Apache Tomcat to run in Failover Zone Configuration

- 1 Become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization on one of the nodes in the cluster that will host Apache Tomcat.

- 2 As superuser register the `SUNW.HASStoragePlus` and the `SUNW.gds` resource types.

```
# clresourcetype register SUNW.HASStoragePlus SUNW.gds
```

- 3 Create a failover resource group.

```
# clresourcegroup create Apache Tomcat-resource-group
```

- 4 Create a resource for the Apache Tomcat zone's disk storage.

```
# clresource create -t SUNW.HASStoragePlus \
-p FileSystemMountPoints=Apache Tomcat-instance-mount-points \
Apache Tomcat-has-resource
```

- 5 (Optional) If you want the protection against a total adapter failure for your public network, create a resource for the Apache Tomcat's logical hostname.

```
# clreslogicalhostname create -g Apache Tomcat-resource-group \
-h logical-hostname \
Apache Tomcat-logical-hostname-resource-name
```

- 6 Place the resource group in the managed state.

```
# clresourcegroup online -M Apache Tomcat-resource-group
```

- 7 Install the zone.

Install the zone according to the Oracle Solaris Cluster HA for Solaris Containers agent documentation, assuming that the resource name is *Apache Tomcat-zone-rs* and that the zone name is *Apache Tomcat-zone*.

- 8 Verify the zone's installation.

```
# zoneadm -z Apache Tomcat-zone boot
# zoneadm -z Apache Tomcat-zone halt
```

- 9 Register the zone's boot component.

- a. Copy the container resource boot component configuration file.

```
# cp /opt/SUNWsczone/sczbt/util/sczbt_config zones-target-configuration-file
```

- b. Use a plain text editor to set the following variables:

```
RS=Apache Tomcat-zone-rs
RG=Apache Tomcat-resource-group
PARAMETERDIR=Apache Tomcat-zone-parameter-directory
SC_NETWORK=true|false
SC_LH=Apache Tomcat-logical-hostname-resource-name
FAILOVER=true|false
HAS_RS=Apache Tomcat-has-resource
Zonename=Apache Tomcat-zone
Zonebootopt=zone-boot-options
Milestone=zone-boot-milestone
Mounts=
```

- c. Create the parameter directory for your zone's resource.

```
# mkdir Apache Tomcat-zone-parameter-directory
```

- d. Execute the Oracle Solaris Cluster HA for Solaris Container's registration script.

```
# /opt/SUNWsczone/sczbt/util/sczbt_register -f zones-target-configuration-file
```

- e. Enable the Solaris Container resource

```
# clresource enable Apache Tomcat-zone-rs
```

- 10 Enable the resource group.

```
# clresourcegroup online Apache Tomcat-resource-group
```

▼ How to Install and Configure Apache Tomcat in a Failover Zone

1 Determine how Apache Tomcat will be deployed in Oracle Solaris Cluster

- Determine whether you will use Apache Tomcat as a failover a multiple master, or a scalable data service.

For conceptual information on scalable and failover data services, see [Oracle Solaris Cluster Concepts Guide](#).

- Determine which user name will run Apache Tomcat.
- Determine how many Apache Tomcat versions and instances will be deployed.
- If more than one instance of a version will be deployed, determine whether they share the binaries.
- Determine which Cluster File System will be used by each Apache Tomcat instance.

2 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.

3 (Optional) If you deploy Apache Tomcat in a non global zone, enter the target zone

```
# zlogin tomcat-zone
```

4 Create user and group if required.

If Apache Tomcat is to run under a non root user, you have to create the appropriate user, and the appropriate group. For these tasks use the following commands.

a. Create the group by running the following command:

```
# groupadd -g 1000 tomcat
```

b. Create the user by running the following command:

```
# useradd -u 1000 -g 1000 -d /global/tomcat -s /bin/ksh tomcat
```

5 If you are not logged in as root, switch to the appropriate user name.

```
# su - user-name
```

6 Install Apache Tomcat.

If you deploy Apache Tomcat as a failover data service, install Apache Tomcat onto a shared file system or in a failover zone within Oracle Solaris Cluster.

It is recommended that you install Apache Tomcat onto shared disks. For a discussion of the advantages and disadvantages of installing the software on a local versus a cluster file system, see “Determining the Location of the Application Binaries” in [Oracle Solaris Cluster Data Services Planning and Administration Guide](#).

If you will deploy Apache Tomcat as a failover data service install the Apache Tomcat binaries on the shared storage on **one** node or in the failover zone. If Apache Tomcat will be deployed as a multiple master data service, install the Apache Tomcat binaries on the local storage in **every** target zone, that will host the Apache Tomcat data service.

Note – Refer to <http://tomcat.apache.org/index.html> for instructions about installing Apache Tomcat.

If you deploy Apache Tomcat as a multiple masters data service, repeat the following step at every node or target zone that will host Apache Tomcat.

7 Create the environment script.

Create an Korn shell or a C shell script (dependent on the login-shell of your Apache Tomcat user name) to set the environment variables for Apache Tomcat. You must set the environment variables in a shell script and not in the users profile.

With this mechanism you can install and run multiple Apache Tomcat versions and instances under one user name.



Caution – These shell scripts must be available on every zone that can host the Apache Tomcat data service. For a failover configuration, store them on the shared storage of the node or in the target zone. For a multiple masters configuration, store them on the local file system or on the shared storage of the target zone. These scripts must not be different on the various zones. For an installation in a failover zone any place where the tomcat user has access is appropriate.

Example 1–9 Korn shell script to set the environment variables for Apache Tomcat 3.x/3.3

```
# more env.ksh
#!/usr/bin/ksh
#
# Environment for Tomcat
#
JAVA_HOME=/usr/j2se
export JAVA_HOME
JAKARTA_HOME=/global/mnt1/jakarta-3.3
export JAKARTA_HOME
TOMCAT_HOME=$JAKARTA_HOME
export TOMCAT_HOME
```

Example 1–10 C shell script to set the environment variables for Apache Tomcat 3.3/3.3

```
# more env.csh
#!/usr/bin/csh
#
# Environment for Tomcat
#

setenv JAVA_HOME /usr/j2se
```

```
setenv JAKARTA_HOME /global/mnt1/jakarta-3.3
setenv TOMCAT_HOME $JAKARTA_HOME
```

Example 1–11 Korn shell script to set the environment variables for Apache Tomcat 4.1.x

```
# more env.ksh
#!/usr/bin/ksh
#
# Environment for Tomcat
#
JAVA_HOME=/usr/j2se
export JAVA_HOME
CATALINA_HOME=/global/mnt1/jakarta-tomcat-4.1.18
export CATALINA_HOME
```

Example 1–12 C shell script to set the environment variables for Apache Tomcat 4.1.x

```
# more env.csh
#!/usr/bin/csh
#
# Environment for Tomcat
#
setenv JAVA_HOME /usr/j2se
setenv CATALINA_HOME /global/mnt1/jakarta-tomcat-4.1.18
```

The environment variables are version and configuration dependent.

Verifying the Installation and Configuration of Apache Tomcat

This section contains the procedure you need to verify the installation and configuration of Apache Tomcat.

▼ How to Verify the Installation and Configuration of Apache Tomcat

This procedure does not verify that your application is highly available because you have not installed your data service yet.

1 Start the Apache Tomcat.

Switch to the Tomcat user name in the target zone (in the following example, it is root) and change to the directory where the environment script is located. In the following example the Tomcat version is 4.1.24.

Note – The output messages of the start and shutdown commands are highly version dependent.

```
# . ./env.ksh
# cd $CATALINA_HOME/bin
# ./startup.sh
Using CATALINA_BASE:   /tomcat/jakarta-tomcat-4.1.24
Using CATALINA_HOME:   /tomcat/jakarta-tomcat-4.1.24
Using CATALINA_TMPDIR: /tomcat/jakarta-tomcat-4.1.24/temp
Using JAVA_HOME:       /usr/j2se
```

2 Check the Installation

Start a web browser and connect to the cluster node with `http://nodename:8080`. If you see the default Tomcat homepage everything is working correctly.

3 Stop the Apache Tomcat.

```
# ./shutdown.sh
Using CATALINA_BASE:   /tomcat/jakarta-tomcat-4.1.24
Using CATALINA_HOME:   /tomcat/jakarta-tomcat-4.1.24
Using CATALINA_TMPDIR: /tomcat/jakarta-tomcat-4.1.24/temp
Using JAVA_HOME:       /usr/j2se
```

▼ How to Deploy the Apache Tomcat Application

If you configure Apache Tomcat as a failover resource, execute the following steps on one node. If you deploy Apache Tomcat as a scalable configuration, repeat the following steps on every node. If you deploy Apache Tomcat in a non global zone, regardless if it is a failover zone or a zone perform all the steps below in the target zone.



Caution – It is vital for a scalable configuration that the Apache Tomcat configuration does not differ on the Tomcat hosts.

1 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.

2 (Optional) Enter the non global target zone.

```
# zlogin zone-name
```

3 Modify the configuration files `server.xml` and the Tomcat user configuration to the desired content.

For further information refer to <http://tomcat.apache.org/index.html>.

4 Start Apache Tomcat as described in “How to Verify the Installation and Configuration of Apache Tomcat” on page 38.

5 Deploy your Application.

For further information refer to <http://tomcat.apache.org/index.html>.

6 Test your application with appropriate methods.

For example, connect with a web browser to the Host with the appropriate port and select the start page of your application.

7 Stop Apache Tomcat as described in “[How to Verify the Installation and Configuration of Apache Tomcat](#)” on page 38.

The port on which Apache Tomcat is serving is required for the configuration of the Oracle Solaris Cluster Apache Tomcat resource.

It is best practice to use the start page of the deployed application for the Fault Monitoring. In this case the TestCmd is described in “[Understanding the Oracle Solaris Cluster Apache Tomcat HA Parameter File](#)” on page 59 can be get /start_page. With this procedure, you monitor Apache Tomcat and the application it is serving.

8 (Optional) Leave the non global zone.

Installing the HA for Apache Tomcat Packages

If you did not install the HA for Apache Tomcat packages during your initial Oracle Solaris Cluster installation, perform this procedure to install the packages. To install the packages, use the installer program.

Note – You need to install the HA for Apache Tomcat packages in the global cluster and not in the zone cluster.

▼ How to Install the HA for Apache Tomcat Packages

Perform this procedure on each cluster node where you want the HA for Apache Tomcat software to run.

You can run the installer program with a command-line interface (CLI) or with a graphical user interface (GUI). The content and sequence of instructions in the CLI and the GUI are similar.

Note – Even if you plan to configure this data service to run in non-global zones, install the packages for this data service in the global zone. The packages are propagated to any existing non-global zones and to any non-global zones that are created after you install the packages.

Before You Begin Ensure that you have the Oracle Solaris Cluster installation media.

If you intend to run the installer program with a GUI, ensure that your DISPLAY environment variable is set.

1 On the cluster node where you are installing the data service packages, become superuser.

2 Load the Oracle Solaris Cluster installation media into the DVD-ROM drive.

If the Volume Management daemon `vol(1M)` is running and configured to manage DVD-ROM devices, the daemon automatically mounts the DVD-ROM on the `/cdrom` directory.

3 Change to the installation wizard directory of the DVD-ROM.

- **If you are installing the data service packages on the SPARC platform, type the following command:**

```
# cd /cdrom/cdrom0/Solaris_sparc
```

- **If you are installing the data service packages on the x86 platform, type the following command:**

```
# cd /cdrom/cdrom0/Solaris_x86
```

4 Start the installation wizard.

```
# ./installer
```

5 When you are prompted, accept the license agreement.

6 From the list of Oracle Solaris Cluster agents under Availability Services, select the data service for Apache Tomcat.

7 If you require support for languages other than English, select the option to install multilingual packages.

English language support is always installed.

8 When prompted whether to configure the data service now or later, choose Configure Later.

Choose Configure Later to perform the configuration after the installation.

9 Follow the instructions on the screen to install the data service packages on the node.

The installation wizard displays the status of the installation. When the installation is complete, the wizard displays an installation summary and the installation logs.

10 (GUI only) If you do not want to register the product and receive product updates, deselect the Product Registration option.

The Product Registration option is not available with the CLI. If you are running the installation wizard with the CLI, omit this step.

11 Exit the installation wizard.

12 Unload the installation media from the DVD-ROM drive.

a. To ensure that the DVD-ROM is not being used, change to a directory that does *not* reside on the DVD-ROM.

b. Eject the DVD-ROM.

```
# eject cdrom
```

Registering and Configuring Oracle Solaris Cluster HA for Apache Tomcat

This section contains the procedures you need to configure Oracle Solaris Cluster HA for Apache Tomcat.

▼ How to Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a Failover Data Service

This procedure assumes that you installed the data service packages.

If you did not install the Oracle Solaris Cluster HA for Apache Tomcat packages, go to [“Installing the HA for Apache Tomcat Packages” on page 40](#).

It is assumed that the file system of Apache Tomcat will be mounted as a failover file system.

1 Prepare the parameter file, which is required by the Oracle Solaris Cluster HA for Apache Tomcat.

```
# cd /opt/SUNWscotomcat/bin
# cp pfile desired-place
```

Edit the parameter file `pfile` and follow the comments within that file. For example:

```
#!/usr/bin/ksh
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the License).
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/CDDL.txt
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/CDDL.txt.
# If applicable, add the following below this CDDL HEADER, with the
```

```

# fields enclosed by brackets \[\] replaced with your own identifying
# information: Portions Copyright \[yyyy\] \[name of copyright owner\]
#
# CDDL HEADER END
#
# Copyright (c) 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#

#ident    "@(#)pfile.ksh 1.6      10/08/16"

# Set the Apache Tomcat specific environment variables which the start, stop
# and check fuctions will use
#
# EnvScript      Script to set runtime environment for tomcat
# User           Apache Tomcat User
# Basepath       Absolute path to Tomcat Home directory i.e. 4.x CATALINA_HOME
#                or TOMCAT_HOME for 3.x
# Host           Hostname to test Apache Tomcat
# Port           Port where Apache Tomcat is configured
# TestCmd        Apache Tomcat test command
#
# Some versions of Apache Tomcat do not react on the probe algorithm with the
# http get command. Because of this we provide an alternative probe method
# which uses wget. If wget is not installed on your system you must install it
# manually and provide the WgetPath variable. If the WgetPath variable is not set,
# it defaults to /usr/sfw/bin/wget.
# You can use the wget algorithm for https probe as well which are not possible
# with the http get command.
#
# Use the variables TestUrl and WgetPath as a replacement for:
# Port, Host, TestCmd
# You can specify only one set of variables.
#
# TestUrl        Url where Tomcat serves a web site. This can be done either
#                by the http or https  protocol
#                Format: http://hostname:port/startpage
#                Example: http://localhost:8080/
#                You might want to include options here like:
#                "--no-cookies http://localhost:8080/"
#                The requirement here is that wget -O - ${TestUrl} produces
#                the output containing the ReturnString.
# WgetPath        Abolute path to wget, leaving this variable empty defaults to:
#                /usr/sfw/bin/wget
# Startwait      Sleeping $Startwait seconds after completion of the
#                start command
# ReturnString    This string must be present in the output of the http get command
#                or in the output of the wget command.
EnvScript=
User=
Basepath=
Host=
Port=8080
TestCmd="get /index.jsp"
TestUrl=
WgetPath=
ReturnString="CATALINA"
Startwait=20

```

The following is an example for Apache Tomcat 4.1.24.

```
EnvScript=/global/mnt1/tomtest/env.ksh
User=tomtest
Basepath=/global/mnt1/tomtest/jakarta-tomcat-4.1.24
Host=tomcat-lh
Port=8080
TestCmd="get /index.jsp"
TestUrl=
WgetPath=
ReturnString="CATALINA"
Startwait=20
```

This example for Apache Tomcat 4.1.24 configures a standalone web server with the default start page `index.jsp`. The start page contains the string `CATALINA`.

To use the alternative `wget` probe, assuming `wget` is available under `/usr/sfw/bin/wget`, instead set the following variables:

```
EnvScript=/global/mnt1/tomtest/env.ksh
User=tomtest
Basepath=/global/mnt1/tomtest/jakarta-tomcat-4.1.24
Host=
Port=
TestCmd=
TestUrl="http://tomcat-lh:8080/"
WgetPath=
ReturnString="CATALINA"
Startwait=20
```



Caution – The parameter files must be available on every node that can host the Apache Tomcat data service. For a scalable or multiple-masters configuration, store the parameter files on the local file system of every node or non-global zone or on the shared storage. The parameter files cannot differ for any instance of Apache Tomcat on the various nodes.

Repeat [Step 1](#) for every Apache Tomcat instance you need.

2 Create and register each required Apache Tomcat component.

```
# cd /opt/SUNWsctomcat/util
# cp sctomcat_config desired place
```

Edit the `sctomcat_config` file and follow the comments within that file. For example:

```
#
# Copyright (c) 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#

#ident    "@(#)sctomcat_config.ksh 1.2    06/03/08"

# This file will be sourced in by sctomcat_register and the parameters
# listed below will be used.
#
# These parameters can be customized in (key=value) form
```

```

#
#      RS - name of the resource for the application
#      RG - name of the resource group containing RS
#      PORT - name of the port number
#      LH - name of the LogicalHostname SC resource
# SCALABLE - true for a scalable resource or false for a failover resource
# NETWORK - false or true, false for multiple master configurations without
#            shared address, in this case SCALABLE will be ignored
#
#      PFILE - absolute path to the parameter file for the Tomcat resource
#      HAS_RS - name of the HASToragePlus SC resource
#               (it can be a , separated list for the dependencies)
#
# The following variables need to be set only if the agent runs in a
# local zone
#      ZONE - the zone name where the Apache Tomcat should run in
#            Optional
#      ZONEBT - The resource name which controls the zone.
#            Optional
#      PROJECT - A project in the zone, that will be used for this service
#               specify it if you have an su - in the start stop or probe,
#               or to define the smf credentials. If the variable is not set,
#               it will be translated as :default for the smf manifest
#            Optional

RS=
RG=
PORT=
LH=
NETWORK=false
SCALABLE=false
PFILE=
HAS_RS=

# local zone specific options

ZONE=
ZONE_BT=
PROJECT=

```

The following is an example for Apache Tomcat.

```

RS=tomcat-res
RG=tomcat-rg
PORT=8080
LH=tomcat-lh
SCALABLE=false
PFILE=/global/mnt1/pfile
NETWORK=true
HAS_RS=tomcat-hastplus-res

```

3 After editing sctomcat_config register the resource.

```
# ksh ./sctomcat_register -f desired place/sctomcat_config
```

Repeat Step 9,10, and 11 for each Apache Tomcat instance you need.

4 Enable each Apache Tomcat resource.

Repeat this step for each Apache Tomcat resource created in the step 10.

```
# clresource status
```

```
# clresource enable Apache-Tomcat-resource
```

▼ How to Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a Multiple Masters Data Service

This procedure assumes that you installed the data service packages during your initial Oracle Solaris Cluster installation.

If you did not install the Oracle Solaris Cluster HA for Apache Tomcat packages as part of your initial Oracle Solaris Cluster installation, go to [“Installing the HA for Apache Tomcat Packages” on page 40](#).

- 1 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.**
- 2 Prepare the parameter file, which is required by the Oracle Solaris Cluster HA for Apache Tomcat.**

```
# cd /opt/SUNWscotomcat/util
# cp scotomcat_config desired-place
```

Edit the parameter file and follow the comments within that file. For example:

```
#!/usr/bin/ksh
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the License).
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/CDDL.txt
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/CDDL.txt.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets \[\] replaced with your own identifying
# information: Portions Copyright \[yyyy\] \[name of copyright owner\]
#
# CDDL HEADER END
#
#
```

```

#
#
# Copyright (c) 2006,2010, Oracle and/or its affiliates. All rights reserved.
#
#ident    "@(#)pfile.ksh 1.6      10/08/16"

# Set the Apache Tomcat specific environment variables which the start, stop
# and check fuctions will use
#
# EnvScript      Script to set runtime environment for tomcat
# User           Apache Tomcat User
# Basepath       Absolute path to Tomcat Home directory i.e. 4.x CATALINA_HOME
#               or TOMCAT_HOME for 3.x
# Host           Hostname to test Apache Tomcat
# Port           Port where Apache Tomcat is configured
# TestCmd        Apache Tomcat test command
#
# Some versions of Apache Tomcat do not react on the probe algorithm with the
# http get command. Because of this we provide an alternative probe method
# which uses wget. If wget is not installed on your system you must install it
# manually and provide the WgetPath variable. If the WgetPath variable is not set,
# it defaults to /usr/sfw/bin/wget.
# You can use the wget algorithm for https probe as well which are not possible
# with the http get command.
#
# Use the variables TestUrl and WgetPath as a replacement for:
# Port, Host, TestCmd
# You can specify only one set of variables.
#
# TestUrl        Url where Tomcat serves a web site. This can be done either
#               by the http or https protocol
#               Format: http://hostname:port/startpage
#               Example: http://localhost:8080/
#               You might want to include options here like:
#               "--no-cookies http://localhost:8080/"
#               The requirement here is that wget -O - ${TestUrl} produces
#               the output containing the ReturnString.
# WgetPath        Absolute path to wget, leaving this variable empty defaults to:
#               /usr/sfw/bin/wget
# Startwait      Sleeping $Startwait seconds after completion of the
#               start command
# ReturnString    This string must be present in the output of the http get command
#               or in the output of the wget command.EnvScript=
EnvScript=
User=
Basepath=
Host=
Port=8080
TestCmd="get /index.jsp"
TestUrl=
WgetPath=
ReturnString="CATALINA"
Startwait=20

```

The following is an example for Apache Tomcat 4.1.24.

```

EnvScript=/tomcat/env.ksh
User=tomcat
Basepath=/tomcat/jakarta-tomca-4.1.24

```

```
Host=localhost
Port=8080
TestCmd="get /index.jsp"
TestUrl=
WgetPath=
ReturnString="CATALINA"
Startwait=20
```

This example for Apache Tomcat 4.1.24 configures a standalone web server with the default start page `index.jsp`. The start page contains the string `CATALINA`.

Depending on the selected probe method, the `Host` parameter or the IP alias in the `TestUrl` parameter depends on the location of the `pfile`:

- If the `pfile` is stored on the shared storage, you must use `localhost`.
- If the `pfile` is stored on the local storage, it can be either `localhost` or the node's host name.

To use the alternative `wget` probe, assuming `wget` is available under `/usr/sfw/bin/wget`, set the following variables:

```
EnvScript=/global/mnt1/tomtest/env.ksh
User=tomtest
Basepath=/global/mnt1/tomtest/jakarta-tomcat-4.1.24
Host=
Port=
TestCmd=
TestUrl="http://localhost:8080/"
WgetPath=
ReturnString="CATALINA"
Startwait=20
```



Caution – The parameter files must be available on every node that can host the Apache Tomcat data service. For a scalable or a multiple-masters configuration, store the parameter files on the local file system of every node or non-global zone or on the shared storage. The parameter files must not differ for any instance of Apache Tomcat on the various nodes.

Repeat this Step for every Apache Tomcat instance and every node you need.

3 Create and register each required Apache Tomcat component.

```
# cd /opt/SUNWscTomcat/util
# cp sctomcat_config desired place
```

Edit the `sctomcat_config` file and follow the comments within that file, for example:

```
#
# Copyright 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#

#ident    "@(#)sctomcat_config.ksh 1.2    06/03/08"

# This file will be sourced in by sctomcat_register and the parameters
```



```

# listed below will be used.
#
# These parameters can be customized in (key=value) form
#
#     RS - name of the resource for the application
#     RG - name of the resource group containing RS
#     PORT - name of the port number
#     LH - name of the LogicalHostname SC resource
#     SCALABLE - true for a scalable resource or false for a failover resource
#     NETWORK - false or true, false for multiple master configurations without
#               shared address, in this case SCALABLE will be ignored
#
#     PFILE - absolute path to the parameter file for the Tomcat resource
#     HAS_RS - name of the HASStoragePlus SC resource
#              (it can be a , separated list for the dependencies)
#
# The following variables need to be set only if the agent runs in a
# local zone
#     ZONE - the zone name where the Apache Tomcat should run in
#            Optional
#     ZONEBT - The resource name which controls the zone.
#              Optional
#     PROJECT - A project in the zone, that will be used for this service
#               specify it if you have an su - in the start stop or probe,
#               or to define the smf credentials. If the variable is not set,
#               it will be translated as :default for the smf manifest
#               Optional

RS=
RG=
PORT=
LH=
NETWORK=false
SCALABLE=false
PFILE=
HAS_RS=

# local zone specific options

ZONE=
ZONE_BT=
PROJECT=

```

The following is an example for Apache Tomcat.

```

RS=tomcat-res
RG=tomcat-rg
PORT=8080
LH=tomcat-lh
SCALABLE=false
SNETWORK=falsePFILE=/local/pfile
HAS_RS=Apache-Tomcat-has-resource

```

Omit the LH parameter here, it will not be used. Set the parameter HAS_RS only when another file system (GFS or local file system) is used and the appropriate Apache-Tomcat-has-resource is created. The ZONE, ZONE_BT and PROJECT variables are needed only if the multiple masters container resources are zones managed by the container agent.

4 After editing `sctomcat_config` register the resource.

```
# ksh ./sctomcat_register -f desired place/sctomcat_config
```

Repeat Steps 7, 8, and 9 for every Apache Tomcat instance you need.

5 Enable each Apache Tomcat resource.

Repeat this step for each Apache Tomcat resource created in the previous step.

```
# clresource status
```

```
# clresource enable Apache-Tomcat-resource
```

▼ How to Register and Configure Oracle Solaris Cluster HA for Apache Tomcat as a Scalable Data Service

This procedure assumes that you installed the data service packages during your initial Oracle Solaris Cluster installation.

If you did not install the Oracle Solaris Cluster HA for Apache Tomcat packages as part of your initial Oracle Solaris Cluster installation, go to [“Installing the HA for Apache Tomcat Packages” on page 40](#).

- 1 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.**
- 2 Prepare the parameter file, which is required by the Oracle Solaris Cluster HA for Apache Tomcat.**

```
# cd /opt/SUNWsctomcat/bin
cp pfile desired-place
```

Edit the parameter file `pfile` and follow the comments within that file. For example:

```
#!/usr/bin/ksh

# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the License).
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/CDDL.txt
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/CDDL.txt.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets \[\] replaced with your own identifying
# information: Portions Copyright \[yyyy\] \[name of copyright owner\]
#
```

```

# CDDL HEADER END
#

#
# Copyright (c) 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#

#ident "@(#)pfile.ksh 1.6      10/08/16"

# Set the Apache Tomcat specific environment variables which the start,
# stop and check functions will use
#
# EnvScript      Script to set runtime environment for tomcat
# User           Apache Tomcat user name
# Basepath       Absolute pathname to Tomcat Home directory i.e. 4.x CATALINA_HOME
#               or TOMCAT_HOME for 3.x
# Host           Hostname to test Apache Tomcat
# Port           Port where Apache Tomcat is configured
# TestCmd        Apache Tomcat test command
#
#
# Some versions of Apache Tomcat do not react on the probe algorithm with the
# http get command. Because of this we provide an alternative probe method
# which uses wget. If wget is not installed on your system you must install it
# manually and provide the WgetPath variable. If the WgetPath variable is not set,
# it defaults to /usr/sfw/bin/wget.
# You can use the wget algorithm for https probe as well which are not possible
# with the http get command.
#
# Use the variables TestUrl and WgetPath as a replacement for:
# Port, Host, TestCmd
# You can specify only one set of variables.
#
# TestUrl        Url where Tomcat serve a web site. This can be done either
#               by the http or https protocol
#               Format: http://hostname:port/startpage
#               Example: http://localhost:8080/
#               You might want to include options here like:
#               "--no-cookies http://localhost:8080/"
#               The requirement here is that wget -O - ${TestUrl} produces
#               the output containing the ReturnString.
# WgetPath        Absolute path to wget, leaving this variable empty defaults to:
#               /usr/sfw/bin/wget
# Startwait      Sleeping $Startwait seconds after completion of the
#               start command
# ReturnString    This string must be present in the output of the http get command
#               or in the output of the wget command.
EnvScript=
User=
Basepath=
Host=
Port=8080
TestCmd="get /index.jsp"
TestUrl=
WgetPath=
ReturnString="CATALINA"
Startwait=20

```

To use the alternative wget probe, assuming wget is available under `/usr/sfw/bin/wget`, set the following variables:

```
EnvScript=/global/mnt1/tomtest/env.ksh
User=tomtest
Basepath=/global/mnt1/tomtest/jakarta-tomcat-4.1.24
Host=
Port=
TestCmd=
TestUrl="http://localhost:8080/"
WgetPath=
ReturnString="CATALINA"
Startwait=20
```

This example for Apache Tomcat 4.1.24 configures a standalone web server with the default start page `index.jsp`. The start page contains the string `CATALINA`.

```
EnvScript=/tomcat/env.ksh
User=tomcat
Basepath=/tomcat/jakarta-tomcat-4.1.24
Host=localhost
Port=8080
TestCmd="get /index.jsp"
TestUrl=
WgetPath=
ReturnString="CATALINA"
Startwait=20
```

Depending on the selected probe method, the `Host` parameter or the IP alias in the `TestUrl` parameter depends on the location of the `pfile`.

- If the `pfile` is stored on the shared storage, you must use `localhost`.
- If the `pfile` is stored on the local storage, it can be either `localhost` or the node's host name.

Do not use the SharedAddress here.



Caution – The parameter files must be available on every node that can host the Apache Tomcat data service. For a scalable or a multiple-masters configuration, store the parameter files on the local file system of every node or on the shared storage. The parameter files must not differ for any instance of Apache Tomcat on the various nodes.

Repeat this Step for every Apache Tomcat instance and every node you need.

3 Create and register each required Apache Tomcat component.

```
# cd /opt/SUNWscotomcat/util
# cp scotomcat_config desired place
```

Edit the `scotomcat_config` file and follow the comments within that file, for example:

```
#
# Copyright 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#
```

```
#ident "@(#)sctomcat_config.ksh 1.2      06/03/08"

# This file will be sourced in by sctomcat_register and the parameters
# listed below will be used.
#
# These parameters can be customized in (key=value) form
#
#     RS - name of the resource for the application
#     RG - name of the resource group containing RS
#     PORT - name of the port number
#     LH - name of the LogicalHostname SC resource
#     SCALABLE - true for a scalable resource or false for a failover resource
#     NETWORK - false or true, false for multiple master configurations without
#               shared address, in this case SCALABLE will be ignored
#
#     PFILE - absolute path to the parameter file for the Tomcat resource
#     HAS_RS - name of the HASStoragePlus SC resource
#              (it can be a , separated list for the dependencies)
#
# The following variables need to be set only if the agent runs in a
# local zone
#     ZONE - the zone name where the Apache Tomcat should run in
#            Optional
#     ZONEBT - The resource name which controls the zone.
#             Optional
#     PROJECT - A project in the zone, that will be used for this service
#              specify it if you have an su - in the start stop or probe,
#              or to define the smf credentials. If the variable is not set,
#              it will be translated as :default for the smf manifest
#              Optional

RS=
RG=
PORT=
LH=
NETWORK=false
SCALABLE=false
PFILE=
HAS_RS=

# local zone specific options

ZONE=
ZONE_BT=
PROJECT=
```

The following is an example for Apache Tomcat.

```
RS=tomcat-res
RG=tomcat-rg
PORT=8080
LH=tomcat-lh
SCALABLE=true
NETWORK=true
PFILE=/tomcat/pfile
HAS_RS=Apache-Tomcat-has-resource
```

Configure the parameter `LH` need to be configured as the name of the shared address. Set the parameter `HAS_RS` only when another file system (GFS or local file system) is used and the appropriate Apache-Tomcat-has-resource is created.

4 After editing `sctomcat_config` register the resource.

```
# ksh ./sctomcat_register desired place/sctomcat_config
```

Repeat Steps 9, 10, an 11 for every Apache Tomcat instance you need.

5 Enable each Apache Tomcat resource.

Repeat this step for each Apache Tomcat resource created in the previous step.

```
#
```

```
# clresource enable Apache-Tomcat-resource
```

Configuring Oracle Solaris Cluster HA for Apache Tomcat in Zones

The Oracle Solaris Cluster HA for Apache Tomcat can be configured in a failover zone and in a zone. This section describes the additional steps which are necessary to configure Oracle Solaris Cluster HA for Apache Tomcat in both zone types.

Configuring Oracle Solaris Cluster HA for Apache Tomcat in a Failover Zone

▼ How to Register Oracle Solaris Cluster HA for Apache Tomcat in a Failover Zone

This procedure uses the configuration file to register Oracle Solaris Cluster HA for Apache Tomcat in a failover zone.

Before You Begin

Before you register Oracle Solaris Cluster HA for Apache Tomcat in a failover zone, perform the following tasks. Follow procedures in the [Oracle Solaris Cluster Data Service for Solaris Containers Guide](#) and [Oracle Solaris Cluster Data Service for Apache Tomcat Guide](#).

- Install Oracle Solaris Cluster HA for Containers and Oracle Solaris Cluster HA for Apache Tomcat data-service packages in the global zone of each cluster node that you want to run the failover zone.
- Create the failover zone.
- Configure the boot component to control your failover zone.

- 1 **Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.**
- 2 **Make a copy of the `/opt/SUNWsctomcat/util/sctomcat_config` file to use as your configuration file.**

You can place this configuration file in any directory that is accessible to the failover zone.

- 3 **Open your configuration file for editing and modify the variables for non-global zones.**

These variables are used only when the data service is installed in a non-global zone. Set the values of these variables as follows:

ZONE	Set the name of the zone where Oracle Solaris Cluster HA for Apache Tomcat should run.
ZONE_BT	Set the name of the resource that controls the zone.
PROJECT	Set the name of the Sun Resource Manager Project in the failover zone. If the PROJECT variable is omitted from the configuration file, the data service uses the default project of the user, <code>:default</code> , in a Solaris Service Management Facility (SMF) context. This project name is placed in the <code>start/project</code> property and the <code>stop/project</code> property of the SMF service.

Note – Keep this modified configuration file for possible future use.

- 4 **Register Oracle Solaris Cluster HA for Apache Tomcat with SMF.**

```
# ksh /opt/SUNWsctomcat/util/sctomcat_register -f filename
```

`-f` Specifies the configuration file name.

filename The configuration file name.

The registration of Oracle Solaris Cluster HA for Apache Tomcat generates an SMF manifest and registers an SMF service in the failover zone that is specified by the `ZONE` variable in the configuration file. The manifest is named *resourcename.xml*, where *resource* is the name that is specified by the `RS` variable in the configuration file. This manifest is placed in the `/var/svc/manifest/application/sczone-agents/` directory. The SMF service is named `svc:/application/sczone-agents/resource`.

The Oracle Solaris Cluster HA for Apache Tomcat data service uses the SMF component of the Oracle Solaris Cluster HA for Containers data service to control and probe this SMF service of the Oracle Solaris Cluster HA for Apache Tomcat. It registers a Oracle Solaris Cluster HA for Containers SMF component on top of the already-created SMF service. The name of the script that starts and stops the manifest as well as probes the SMF component is `control_sctomcat`.

▼ How to Modify Parameters in the Oracle Solaris Cluster HA for Apache Tomcat Manifest

Perform this task to change parameters in the Oracle Solaris Cluster HA for Apache Tomcat manifest and to validate the parameters in the failover zone. Parameters for the Oracle Solaris Cluster HA for Apache Tomcat manifest are stored as properties of the SMF service. To modify parameters in the manifest, change the related properties in the SMF service then validate the parameter changes.

- 1 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations on the zones console.
- 2 Change the Solaris Service Management Facility (SMF) properties for the Oracle Solaris Cluster HA for Apache Tomcat manifest.

```
# svccfg svc:/application/sczone-agents:resource
```

For more information, see the [svccfg\(1M\)](#) man page.

- 3 Validate the parameter changes.

```
# /opt/SUNWscTomcat/bin/control_sctomcat validate resource
```

Messages for this command are stored in the `/var/adm/messages/` directory of the failover zone.

- 4 Disconnect from the failover zone's console.

▼ How to Remove a Oracle Solaris Cluster HA for Apache Tomcat Resource From a Failover Zone

- 1 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.
- 2 Disable and remove the resource that is used by the Oracle Solaris Cluster HA for Apache Tomcat data service.

```
# clresource disable resource
# clresource delete resource
```

- 3 Log in as superuser to the failover zone's console.
- 4 Unregister Oracle Solaris Cluster HA for Apache Tomcat from the Solaris Service Management Facility (SMF) service.

```
# /opt/SUNWscTomcat/util/sctomcat_smf_remove -f filename
```

-f Specifies the configuration file name.

filename The name of the configuration file that you used to register Oracle Solaris Cluster HA for Apache Tomcat with the SMF service.

Note – If you no longer have the configuration file that you used to register Oracle Solaris Cluster HA for Apache Tomcat with the SMF service, create a replacement configuration file:

- a. Make a copy of the default file, `/opt/SUNWsctomcat/util/sctomcat_config`.
 - b. Set the `ZONE` and `RS` parameters with the values that are used by the data service.
 - c. Run the `sctomcat_smf_remove` command and use the `-f` option to specify this configuration file.
-

- 5 Disconnect from the failover zone's console.

Configuring Oracle Solaris Cluster HA for Apache Tomcat in a Zone

▼ How to Register Oracle Solaris Cluster HA for Apache Tomcat in a Zone

This procedure uses the configuration file to register Oracle Solaris Cluster HA for Apache Tomcat in a zone.

Before You Begin Before you register Oracle Solaris Cluster HA for Apache Tomcat in a zone, perform the following tasks. .

- Install Oracle Solaris Cluster HA for Apache Tomcat data-service packages in the global zone of each cluster node that you want to run the zone.
- Create the zone on each node to host Oracle Solaris Cluster HA for Apache Tomcat.

Note – The zones configuration parameter `autoboot` **must** be set to `true`

- 1 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.
- 2 Make a copy of the `/opt/SUNWsctomcat/util/sctomcat_config` file to use as your configuration file.
- 3 Open your configuration file for editing, but do not set the variables for non-global zones.
These variables are used only when the data service is installed in a non-global zone managed by the Container Agent.

Note – Keep this modified configuration file for possible future use.

4 Register Oracle Solaris Cluster HA for Apache Tomcat with SMF.

```
# ksh /opt/SUNWscTomcat/util/sctomcat_register -f filename
```

-f Specifies the configuration file name.

filename The configuration file name.

Verifying the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration

This section contains the procedure you need to verify that you installed and configured your data service correctly.

▼ How to Verify the Oracle Solaris Cluster HA for Apache Tomcat Installation and Configuration

1 Become superuser or assume a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` RBAC authorizations.

2 Ensure all the Apache Tomcat resources are online with `scstat`.

```
# cluster status
```

For each Apache Tomcat resource which is not online, use the `clresource` command as follows.

```
# clresource enable Apache-Tomcat-resource
```

3 If you deployed a failover data service run the `clresourcegroup` command to switch the Apache Tomcat resource group to another cluster node, such as `node2` or `node2:zone1` with the command alternatives described below.

```
# clresourcegroup switch -n node2 Apache-Tomcat-failover-resource-group
```

```
# clresourcegroup switch -n node2:zone1 Apache-Tomcat-failover-resource-group
```

Understanding the Oracle Solaris Cluster Apache Tomcat HA Parameter File

Use the information to understand the contents of the Oracle Solaris Cluster Apache Tomcat HA parameter file. This section describes the structure and the content of the Oracle Solaris Cluster Apache Tomcat HA parameter file, as well as the strategy to chose some of its variables.

Structure of the Apache Tomcat HA parameter file

Apache Tomcat HA uses a parameter file to pass parameters to the start, stop and probe command. This parameter file needs to be a valid Korn shell script which sets several variables. The structure of this file appears in [Table 1–7](#). For examples of the parameter file refer to “[Registering and Configuring Oracle Solaris Cluster HA for Apache Tomcat](#)” on page 42.

TABLE 1–7 Structure of the Apache Tomcat HA parameter file

Variable	Explanation
EnvScript	This is a ksh script or a csh script depending on the login-shell of the user name, which owns Apache Tomcat. The purpose of this script is to set the Apache Tomcat specific environment variables, which are needed to start and stop the Apache Tomcat instance.
User	The owner of the Apache Tomcat instance.
Basepath	Basepath is the absolute pathname to the directory where the Tomcat bin directory resides. Typically, it is the TOMCAT_HOME (3.x) or the CATALINA_HOME (4.x).
Host	The Host variable is the Host to test the functionality of the Apache Tomcat server process. The Test is done via a connection to Host:Port.
Port	A Port where Apache Tomcat is serving. This Port is used together with the Host to test the functionality of the Apache Tomcat server process.
TestCmd	This variable represents the command which is passed to the Apache Tomcat server process to test it's sanity.
ReturnString	The variable ReturnString represents the string which must be present in the answer to the TestCmd. It can not be Connection refused because this string is in the answer when the Apache Tomcat server process is not running.

TABLE 1–7 Structure of the Apache Tomcat HA parameter file *(Continued)*

Variable	Explanation
Startwait	This variable stands for the number of seconds to wait after the Apache Tomcat start command is completed. It lasts until the Apache Tomcat server process is fully operational. The absolute number of seconds depends on the speed and the load of the Hardware. A good strategy is to start with 10-20 seconds.
TestUrl	This variable presents a complete http or https URL, which is passed to the wget command to test the Tomcat server's sanity.
WgetPath	This variable presents the absolute path to the wget binary, if the wget binary is not located under /usr/sfw/bin/wget.

The parameters in [Table 1–7](#) can be changed at any time. The only difference is when changes take effect.

The following parameters of the Apache Tomcat HA parameter file are used for starting and stopping Apache Tomcat. Changes to these parameters take effect at every restart or disabling and enabling of a Apache Tomcat resource.

- EnvScript
- User
- Basepath
- Startwait
- TestUrl
- WgetPath

The following parameters of the Apache Tomcat HA parameter file are used within the fault monitor. Changes to these parameters take effect at every `Thorough_probe_interval`.

- Host
- Port
- TestCmd
- ReturnString

Strategy to Choose the TestCmd and the ReturnString Variable

The following alternatives are available to choose the ReturnString variable:

- Take the start page of your application and set the TestCmd to `get /start_page`. If you picked the wget probe method, set TestUrl to `http://ipalias:port/start_page`. With this strategy, you are monitoring that Apache Tomcat is serving your application.

- Take the Apache Tomcat default start page and set the TestCmd to get /default-startpage. If you picked the wget method, set TestUrl to http://ipalias:port/default-startpage. In this case, set the ReturnString to a string contained in the start page. This string depends on the deployed Apache Tomcat version; for 3.x it is Tomcat home and for 4.x and 5.x it is CATALINA. With this strategy, you are monitoring that Apache Tomcat is serving its default application.
- Deploy a test application (which is not provided with the HA for Apache Tomcat agent) to Apache Tomcat. Set the TestCmd to get startpage_of_the_application. If you use the wget method, set TestUrl to http://ipalias:port/startpage_of_the_application. In this case, set the ReturnString to a string contained in the start page. With this strategy, you are monitoring that Apache Tomcat is serving your test application.
- If none of the above is appropriate, set the TestCmd to get /a-page-which-does-not-exists. In this case, set the ReturnString to a string contained in the Error Page. With this strategy, you are monitoring that Apache Tomcat is operational, because it registers that it must deliver a page that does not exist.

You can evaluate the different pages by connecting using a browser with hostname:port and specifying the different pages.

Understanding Oracle Solaris Cluster HA for Apache Tomcat Fault Monitor

This section describes the Oracle Solaris Cluster HA for Apache Tomcat fault monitor's probing algorithm and functionality, states the conditions, messages, and recovery actions associated with unsuccessful probing, and states the conditions and messages associated with unsuccessful probing.

For conceptual information on fault monitors, see [Oracle Solaris Cluster Concepts Guide](#).

Resource Properties

The Oracle Solaris Cluster HA for Apache Tomcat fault monitor uses the same resource properties as the resource type SUNW.gds. Refer to the [SUNW.gds\(5\)](#) man page for a complete list of resource properties used.

Probing Algorithm and Functionality

The following steps are executed to monitor the sanity of Apache Tomcat.

1. Sleeps for Thorough_probe_interval.
2. Pings the Host, which is configured in the Apache Tomcat HA parameter file unless the TestUrl parameter is specified in the Tomcat parameter file.

3. Connects to the Apache Tomcat via Host and Port. If the connection is successful it sends the TestCmd and tests whether the ReturnString comes back. If it fails, it is rescheduled after 5 seconds. If this fails again, then the probe will restart the Apache Tomcat.



Caution – The ReturnString cannot be **Connection refused** because this string will be returned if no connection is possible.

4. If the Apache Tomcat process has died, pmf will interrupt the probe to immediately restart the Apache Tomcat.
5. If the Apache Tomcat is repeatedly restarted and subsequently exhausts the Retry_count within the Retry_interval, then a failover is initiated for the resource group onto another node. This is done if the resource property Failover_enabled is set to TRUE.

Debugging HA for Apache Tomcat

HA for Apache Tomcat has a file named config that enables you to activate debugging for Apache Tomcat resources. This file is in the /opt/SUNWscotomcat/etc directory.

▼ How to Activate Debugging for HA for Apache Tomcat

- 1 **Determine whether you are in a global zone or in a failover zone configuration.**

If your operating system is Oracle Solaris 10 and your Apache Tomcat resource is dependent on a Solaris Container boot component resource, you are in a failover zone configuration. In any other case, especially on a Solaris 9 system, you are in a global zone configuration.

- 2 **Determine whether debugging for HA for Apache Tomcat is active.**

```
# grep daemon /etc/syslog.conf
*.err;kern.debug;daemon.notice;mail.crit      /var/adm/messages
*.alert;kern.err;daemon.err                    operator
#
```

If debugging is inactive, daemon.notice is set in the file /etc/syslog.conf of the appropriate zone.

- 3 **If debugging is inactive, edit the /etc/syslog.conf file in the appropriate zone to change daemon.notice to daemon.debug.**

- 4 **Confirm that debugging for HA for Apache Tomcat is active.**

If debugging is active, daemon.debug is set in the file /etc/syslog.conf.

```
# grep daemon /etc/syslog.conf
*.err;kern.debug;daemon.debug;mail.crit      /var/adm/messages
```

```
*.alert;kern.err;daemon.err          operator
#
```

5 Restart the syslogd daemon in the appropriate zone.

```
# svcadm refresh svc:/system/system-log:default
```

6 Edit the /opt/SUNWsczone/sczbt/etc/config file to change the DEBUG= variable according to one of the examples:

- `DEBUG=ALL`
- `DEBUG=resource name`
- `DEBUG=resource name,resource name, ...`

```
# cat /opt/SUNWscPostgreSQL/etc/config
#
# Copyright 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#
# Usage:
#     DEBUG=<RESOURCE_NAME> or ALL
#
DEBUG=ALL
#
```

Note – To deactivate debugging, repeat step 1 to 6, changing `daemon.debug` to `daemon.notice` and changing the `DEBUG` variable to `DEBUG=`.

Deployment Example: Installing Apache Tomcat in the Global Zone

This appendix presents a complete example of how to install and configure the Apache Tomcat application and data service in the global zone. It presents a simple two-node cluster configuration. If you need to install the application in any other configuration, refer to the general-purpose procedures presented elsewhere in this manual. For an example of Apache Tomcat installation in a non-global zone, see [Appendix B, “Deployment Example: Installing Apache Tomcat in a Failover Zone,”](#) or [Appendix C, “Deployment Example: Installing Apache Tomcat in a Non-Global Zone,”](#) according to your zone type.

Target Cluster Configuration

This example uses a two-node cluster with the following node names:

- phys - schost - 1 (a physical node, which owns the file system)
- phys - schost - 2 (a physical node)

This configuration also uses the logical host name `ha - host - 1`.

Software Configuration

This deployment example uses the following software products and versions:

- Oracle Solaris 10 9/10 software for SPARC or x86 platforms
- Oracle Solaris Cluster 3.3 core software
- HA for Apache Tomcat
- A minimum of Apache Tomcat version 5.5.17
- The `wget` binary is available under `/usr/sfw/bin/wget`.
- Your preferred text editor

This example assumes that you have already installed and established your cluster. It illustrates installation and configuration of the data service application only.

Note – The steps for installing Apache Tomcat in a cluster that runs on Solaris version 9 OS are identical to the steps in this example.

Assumptions

The instructions in this example were developed with the following assumptions:

- **Shell environment:** All commands and the environment setup in this example are for the Korn shell environment. If you use a different shell, replace any Korn shell-specific information or instructions with the appropriate information for your preferred shell environment.
- **User login:** Unless otherwise specified, perform all procedures as superuser or assume a role that provides `solaris.cluster.admin`, `solaris.cluster.modify`, and `solaris.cluster.read` RBAC authorization.

Installing and Configuring Apache Tomcat on Global Storage in the Global Zone

The tasks you must perform to install and configure Apache Tomcat in the global zone are as follows:

- “Example: Preparing the Cluster for Apache Tomcat” on page 66
- “Example: Configuring Cluster Resources for Apache Tomcat” on page 67
- “Example: Installing the Apache Tomcat Software on Shared Storage” on page 67
- “Example: Modifying the Apache Tomcat Configuration Files” on page 68
- “Example: Enabling the Apache Tomcat Software to Run in the Cluster” on page 69

▼ Example: Preparing the Cluster for Apache Tomcat

- 1 **Install and configure the cluster as instructed in [Oracle Solaris Cluster Software Installation Guide](#).**

Install the following cluster software components on both nodes.

- Oracle Solaris Cluster core software
- Oracle Solaris Cluster data service for Apache Tomcat

- 2 **Beginning on the node that owns the file system, add the tomcat user.**

```
phys-schost-1# groupadd -g 1000 tomcat
phys-schost-2# groupadd -g 1000 tomcat
phys-schost-1# useradd -g 1000 -d /global/mnt3/tomcat -m -s /bin/ksh tomcat
phys-schost-2# useradd -g 1000 -d /global/mnt3/tomcat -m -s /bin/ksh tomcat
```

▼ Example: Configuring Cluster Resources for Apache Tomcat

- 1 Register the necessary data types on both nodes.

```
phys-schost-1# clresourcetype register SUNW.gds SUNW.HAStoragePlus
```

- 2 Create the Apache Tomcat resource group.

```
phys-schost-1# clresourcegroup create RG-TOM
```

- 3 Create the logical host.

```
phys-schost-1# clreslogicalhostname create -g RG-TOM ha-host-1
```

- 4 Create the HAStoragePlus resource in the RG-TOM resource group.

```
phys-schost-1# clresource create -g RG-TOM -t SUNW.HAStoragePlus -p AffinityOn=TRUE \
-p FilesystemMountPoints=/global/mnt3,/global/mnt4 RS-TOM-HAS
```

- 5 Enable the resource group.

```
phys-schost-1# clresourcegroup online -M RG-TOM
```

▼ Example: Installing the Apache Tomcat Software on Shared Storage

These steps illustrate how to install the Apache Tomcat software in the directory `/global/mnt3/tomcat`. As long as only one node is mentioned it needs to be the node where your resource group is online.

- 1 Install the Apache Tomcat binaries on.

```
phys-schost-1# su - tomcat
phys-schost-1# gzcatt apache-tomcat-5.5.17.tar.gz|tar xvf -
phys-schost-1# gzcatt apache-tomcat-5.5.17-compat.tar.gz|tar xvf -
```

- 2 Create your environment script `/global/mnt3/tomcat/env.ksh` with the following contents.

```
cat env.ksh
#!/usr/bin/ksh
JAVA_HOME=/usr/j2se
CATALINA_HOME=/global/mnt3/tomcat/apache-tomcat-5.5.17
export JAVA_HOME CATALINA_HOME
```

▼ Example: Modifying the Apache Tomcat Configuration Files

- 1 **Copy the Apache Tomcat configuration file from the agent directory to its deployment location.**

```
phys-schost-1# cp /opt/SUNWsctomcat/util/sctomcat_config /global/mnt3
phys-schost-1# cp /opt/SUNWsctomcat/bin/pfile /global/mnt3
phys-schost-1# chown tomcat:tomcat /global/mnt3/pfile
```

- 2 **Add this cluster's information to the sctomcat_config configuration file.**

The following listing shows the relevant file entries and the values to assign to each entry.

```
.
.
.

RS=RS-TOM
RG=RG-TOM
PORT=8080
LH=ha-host-1
NETWORK=true
SCALABLE=false
PFILE=/global/mnt3/pfile
HAS_RS=RS-TOM-HAS
```

- 3 **Save and close the file.**
- 4 **Add this cluster's information to the parameter file /global/mnt3/pfile.**

The following listing shows the relevant file entries and the values to assign to each entry.

```
EnvScript=/global/mnt3/tomcat/env.ksh
User=tomcat
Basepath=/global/mnt3/tomcat/apache-tomcat-5.5.17
Host=ha-host-1
Port=8080
TestCmd="get /index.jsp"
ReturnString="CATALINA"
Startwait=20
```

As an alternative, you can use the following set of variables:

```
EnvScript=/global/mnt3/tomcat/env.ksh
User=tomcat
Basepath=/global/mnt3/tomcat/apache-tomcat-5.5.17
TestUrl="http://ha-host-1:8080/"
ReturnString="CATALINA"
Startwait=20
```

- 5 **Save and close the file.**

▼ Example: Enabling the Apache Tomcat Software to Run in the Cluster

- 1 Run the `sctomcat_register` script to register the resource.

```
phys-schost-1# ksh /opt/SUNWsctomcat/util/sctomcat_register \  
-f /global/mnt3/sctomcat_config
```

- 2 Enable the resource.

```
phys-schost-1# clresource enable RS-TOM
```


Deployment Example: Installing Apache Tomcat in a Failover Zone

This appendix presents a complete example of how to install and configure the Apache Tomcat application and data service in a non — global failover zone. It presents a simple two-node cluster configuration. If you need to install the application in any other configuration, refer to the general-purpose procedures presented elsewhere in this manual. For an example of Apache Tomcat installation in a global zone, see [Appendix A, “Deployment Example: Installing Apache Tomcat in the Global Zone,”](#) or for an installation in a zone see [Appendix C, “Deployment Example: Installing Apache Tomcat in a Non-Global Zone,”](#) according to your zone type.

Target Cluster Configuration

This example uses a two-node cluster with the following node names:

- phys-schost-1 (a physical node, which owns the file system)
- phys-schost-2 (a physical node)
- clu1 the zone to be failed over

This configuration also uses the logical host name ha-host-1.

Software Configuration

This deployment example uses the following software products and versions:

- Oracle Solaris 10 9/10 software for SPARC or x86 platforms
- Oracle Solaris Cluster 3.3 core software
- HA for Apache Tomcat
- Oracle Solaris Cluster HA for Solaris Container
- A minimum of Apache Tomcat version 5.5.17
- The wget binary is available under /usr/sfw/bin/wget.
- Your preferred text editor

This example assumes that you have already installed and established your cluster. It illustrates installation and configuration of the data service application only.

Assumptions

The instructions in this example were developed with the following assumptions:

- **Shell environment:** All commands and the environment setup in this example are for the Korn shell environment. If you use a different shell, replace any Korn shell-specific information or instructions with the appropriate information for your preferred shell environment.
- **User login:** Unless otherwise specified, perform all procedures as superuser or assume a role that provides `solaris.cluster.admin`, `solaris.cluster.modify`, and `solaris.cluster.read` RBAC authorization.

Installing and Configuring Apache Tomcat on Global Storage in the Failover Zone

The tasks you must perform to install and configure Apache Tomcat in the failover zone are as follows:

- “[Example: Preparing the Cluster for Apache Tomcat](#)” on page 72
- “[Example: Configuring Cluster Resources for Apache Tomcat](#)” on page 73
- “[Example: Configuring the Failover Zone](#)” on page 73
- “[Example: Installing the Apache Tomcat Software on Shared Storage](#)” on page 75
- “[Example: Modifying the Apache Tomcat Configuration Files](#)” on page 76
- “[Example: Enabling the Apache Tomcat Software to Run in the Cluster](#)” on page 76

▼ Example: Preparing the Cluster for Apache Tomcat

- Install and configure the cluster as instructed in [Oracle Solaris Cluster Software Installation Guide](#).

Install the following cluster software components on both nodes.

- Oracle Solaris Cluster core software
- Oracle Solaris Cluster data service for Apache Tomcat
- Oracle Solaris Cluster data service for Solaris Containers

▼ Example: Configuring Cluster Resources for Apache Tomcat

- 1 Register the necessary data types on both nodes.

```
phys-schost-1# clresource type register SUNW.gds SUNW.HASStoragePlus
```

- 2 Create the Apache Tomcat resource group.

```
phys-schost-1# clresourcegroup create RG-TOM
```

- 3 Create the HASStoragePlus resource in the RG-TOM resource group.

```
phys-schost-1# clresource create -g RG-TOM -t SUNW.HASStoragePlus -p AffinityOn=TRUE \
-p FilesystemMountPoints=/global/mnt3,/global/mnt4 RS-TOM-HAS
```

- 4 Enable the resource group.

```
phys-schost-1# clresourcegroup online -M RG-TOM
```

▼ Example: Configuring the Failover Zone

- 1 On shared cluster storage, create a directory for the failover zone root path.

This example presents a sparse root zone. You can use a whole root zone if that type better suits your configuration.

```
phys-schost-1# mkdir /global/mnt3/zones
```

- 2 Create a temporary file, for example /tmp/x, and include the following entries:

```
create -b
set zonepath=/global/mnt3/zones/clu1
set autoboot=false
set pool=pool_default
add inherit-pkg-dir
set dir=/lib
end
add inherit-pkg-dir
set dir=/platform
end
add inherit-pkg-dir
set dir=/sbin
end
add inherit-pkg-dir
set dir=/usr
end
add net
set address=ha-host-1
set physical=hme0
end
add attr
set name=comment
set type=string
```

```
set value="Apache Tomcat cluster zone"      Put your desired zone name between the quotes here.  
end
```

3 Configure the failover zone, using the file you created.

```
phys-schost-1# zonecfg -z clu1 -f /tmp/x
```

4 Install the zone.

```
phys-schost-1# zoneadm -z clu1 install
```

5 Log in to the zone.

```
phys-schost-1# zlogin -C clu1
```

6 Open a new window to the same node and boot the zone?

```
phys-schost-1a# zoneadm -z clu1 boot
```

7 Close this terminal window and disconnect from the zone console.

```
phys-schost-1# ~.
```

8 Copy the containers configuration file to a temporary location.

```
phys-schost-1# cp /opt/SUNWsczone/sczbt/util/sczbt_config /tmp/sczbt_config
```

9 Edit the /tmp/sczbt_config file and set variable values as shown:

```
RS=RS-TOM-ZONE  
RG=RG-TOM  
PARAMETERDIR=/global/mnt3/zonepar  
SC_NETWORK=false  
SC_LH=  
FAILOVER=true  
HAS_RS=RS-TOM-HAS
```

```
Zonename=clu1  
Zonebootopt=  
Milestone=multi-user-server  
Mounts=
```

10 Create the zone according to the instructions in the [Oracle Solaris Cluster Data Service for Solaris Containers Guide](#)[Oracle Solaris Cluster Data Service for Solaris Containers Guide](#).

11 Register the zone resource.

```
phys-schost-1# ksh /opt/SUNWsczone/sczbt/util/sczbt_register -f /tmp/sczbt_config
```

12 Enable the zone resource.

```
phys-schost-1# clresource enable RS-TOM-ZONE
```

▼ Example: Installing the Apache Tomcat Software on Shared Storage

These steps illustrate how to install the Apache Tomcat software in the directory `/tomcat`. As long as only one node is mentioned it needs to be the node where your resource group is online.

1 Log in to the zone.

```
phys-schost-1# zlogin clu1
```

2 Add the tomcat group and user.

```
zone# groupadd -g 1000 tomcat
zone# useradd -g 1000 -d /tomcat -m -s /bin/ksh tomcat
```

3 Install the Apache Tomcat binaries.

```
zone# su - tomcat
zone# gzcac apache-tomcat-5.5.17.tar.gz|tar xvf -
zone# gzcac apache-tomcat-5.5.17-compatible.tar.gz|tar xvf -
```

4 Create your environment script `/tomcat/env.ksh` with the following contents.

```
zone# cat env.ksh
#!/usr/bin/ksh
JAVA_HOME=/usr/j2se
CATALINA_HOME=/tomcat/apache-tomcat-5.5.17
export JAVA_HOME CATALINA_HOME
```

5 Copy the Apache Tomcat configuration file from the agent directory to its deployment location.

```
zone# cp /opt/SUNWsctomcat/bin/pfile /tomcat
zone# chown tomcat:tomcat /tomcat/pfile
```

6 Add this cluster's information to the parameter file `/tomcat/pfile`.

The following listing shows the relevant file entries and the values to assign to each entry.

```
EnvScript=/tomcat/env.ksh
User=tomcat
Basepath=/tomcat/apache-tomcat-5.5.17
Host=ha-host-1
Port=8080
TestCmd="get /index.jsp"
ReturnString="CATALINA"
Startwait=20
```

As an alternative, you can use the following set of variables:

```
EnvScript=/global/mnt3/tomcat/env.ksh
User=tomcat
Basepath=/global/mnt3/tomcat/apache-tomcat-5.5.17
TestUrl="http://ha-host-1:8080/"
ReturnString="CATALINA"
Startwait=20
```

- 7 Save and close the file.
- 8 Leave the zone.

▼ Example: Modifying the Apache Tomcat Configuration Files

- 1 Copy the Apache Tomcat configuration file from the agent directory to its deployment location.

```
phys-schost-1# cp /opt/SUNWsctomcat/util/sctomcat_config /global/mnt3
```

- 2 Add this cluster's information to the sctomcat_config configuration file.

The following listing shows the relevant file entries and the values to assign to each entry.

```
RS=RS-TOM
RG=RG-TOM
PORT=8080
LH=ha-host-1
NETWORK=true
SCALABLE=false
PFILE=/tomcat/pfile
HAS_RS=RS-TOM-HAS
ZONE=clu1
ZONE_BT=RS-TOM-ZONE
PROJECT=
```

- 3 Save and close the file.

▼ Example: Enabling the Apache Tomcat Software to Run in the Cluster

- 1 Run the sctomcat_register script to register the resource.

```
phys-schost-1# ksh /opt/SUNWsctomcat/util/sctomcat_register \
-f /global/mnt3/sctomcat_config
```

- 2 Enable the resource.

```
phys-schost-1# clresource enable RS-TOM
```

Deployment Example: Installing Apache Tomcat in a Non-Global Zone

This appendix presents a complete example of how to install and configure the Apache Tomcat application and data service in a non-global zone. It presents a simple two-node cluster configuration. If you need to install the application in any other configuration, refer to the general-purpose procedures presented elsewhere in this manual. For an example of Apache Tomcat in the global zone, see [Appendix A, “Deployment Example: Installing Apache Tomcat in the Global Zone.”](#) For an installation in failover zone, see [Appendix B, “Deployment Example: Installing Apache Tomcat in a Failover Zone.”](#)

Target Cluster Configuration

This example uses a two-node cluster with the following node names:

- `phys-schost-1` (a physical node, which owns the file system)
- `zone-1` (a zone defined on `phys-schost-1`, which owns the file system)
- `phys-schost-2` (a physical node)
- `zone-2` (a physical node)

Software Configuration

This deployment example uses the following software products and versions:

- Oracle Solaris 10 9/10 software for SPARC or x86 platforms
- Oracle Solaris Cluster 3.3 core software
- Oracle Solaris Cluster Data Service for Apache Tomcat
- A minimum of Apache Tomcat version 5.5.17
- The `wget` binary is available under `/usr/sfw/bin/wget`.

This example assumes that you have already installed and established your cluster. It illustrates installation and configuration of the data service application only.

Assumptions

The instructions in this example were developed with the following assumptions:

- **Shell environment:** All commands and the environment setup in this example are for the Korn shell environment. If you use a different shell, replace any Korn shell-specific information or instructions with the appropriate information for your preferred shell environment.
- **User login:** Unless otherwise specified, perform all procedures as superuser or assume a role that provides `solaris.cluster.admin`, `solaris.cluster.modify`, and `solaris.cluster.read` RBAC authorization.

Installing and Configuring Apache Tomcat on Shared Storage in a Non-Global Zone

These instructions assume that you are installing the Apache Tomcat software as the `tomcat` user in a shared directory.

The tasks you must perform to install and configure Apache Tomcat in the zone are as follows:

- [“Example: Preparing the Cluster for Apache Tomcat” on page 78](#)
- [“Example: Configuring the Zone” on page 78](#)
- [“Example: Configuring Cluster Resources for Apache Tomcat” on page 80](#)
- [“Example: Installing the Apache Tomcat Software on Local Storage” on page 80](#)
- [“Example: Modifying the Apache Tomcat Configuration File” on page 81](#)
- [“Example: Enabling the Apache Tomcat Software to Run in the Cluster” on page 82](#)

▼ Example: Preparing the Cluster for Apache Tomcat

- Install and configure the cluster as instructed in [Oracle Solaris Cluster Software Installation Guide](#).

Install the following cluster software components on both nodes.

- Oracle Solaris Cluster core software
- Oracle Solaris Cluster data service for Apache Tomcat

▼ Example: Configuring the Zone

In this task you will install the Solaris Container on `phys-schost-1` and `phys-schost-2`. Therefore perform this procedure on both hosts.

1 On local cluster storage of , create a directory for the zone root path.

This example presents a sparse root zone. You can use a whole root zone if that type better suits your configuration.

```
phys-schost-1# mkdir /zones
```

2 Create a temporary file, for example /tmp/x, and include the following entries:

```
create -b
set zonepath=/zones/clu1
set autoboot=true
set pool=pool_default
add inherit-pkg-dir
set dir=/lib
end
add inherit-pkg-dir
set dir=/platform
end
add inherit-pkg-dir
set dir=/sbin
end
add inherit-pkg-dir
set dir=/usr
end
add net
set address=zone-1      Choose a different address (zone-2) on the second node.
set physical=hme0
end
add attr
set name=comment
set type=string
set value="Apache Tomcat cluster zone"    Put your desired zone name between the quotes here.
end
```

3 Configure the failover zone, using the file you created.

```
phys-schost-1# zonecfg -z clu1 -f /tmp/x
```

4 Install the zone.

```
phys-schost-1# zoneadm -z clu1 install
```

5 Log in to the zone.

```
phys-schost-1# zlogin -C clu1
```

6 Open a new window to the same node and boot the zone?

```
phys-schost-1# zoneadm -z clu1 boot
```

7 Close this terminal window and disconnect from the zone console.

```
phys-schost-1# ~.
```

▼ Example: Configuring Cluster Resources for Apache Tomcat

- 1 Register the HAStoragePlus resource type.

```
phys-schost-1# clresource type register SUNW.gds SUNW.HAStoragePlus
```

- 2 Create the Apache Tomcat resource group.

```
phys-schost-1# clresourcegroup create -n phys-host-1:clu1,phys-host-2:clu1 RG-TOM
```

- 3 Create the HAStoragePlus resource in the RG-TOM resource group.

```
phys-schost-1# clresource create -g RG-TOM -t SUNW.HAStoragePlus -p AffinityOn=TRUE \
-p FilesystemMountPoints=/global/mnt3,/global/mnt4 RS-TOM-HAS
```

- 4 Enable the resource group.

```
phys-schost-1# clresourcegroup online -M RG-TOM
```

▼ Example: Installing the Apache Tomcat Software on Local Storage

These steps illustrate how to install the Apache Tomcat software in the shared storage. As long as only one node is mentioned it needs to be the node where your resource group is online.

- 1 Log in to the zone on both nodes.

```
phys-schost-1# zlogin clu1
phys-schost-2# zlogin clu1
```

- 2 Add the tomcatgroup and user.

```
zone-1# groupadd -g 1000 tomcat
zone-1# useradd -g 1000 -d /global/mnt3/tomcat -m -s /bin/ksh tomcat
zone-2# groupadd -g 1000 tomcat
zone-2# useradd -g 1000 -d /global/mnt3/tomcat -m -s /bin/ksh tomcat
```

- 3 Install the Apache Tomcat binaries.

```
zone-1# su - tomcat
zone-1$ gzcat apache-tomcat-5.5.17.tar.gz|tar xvf -
zone-1$ gzcat apache-tomcat-5.5.17-compat.tar.gz|tar xvf -
```

- 4 Create your environment script /global/mnt3/tomcat/env.ksh with the following contents.

```
cat env.ksh
#!/usr/bin/ksh
JAVA_HOME=/usr/j2se
CATALINA_HOME=/global/mnt3/tomcat/apache-tomcat-5.5.17
export JAVA_HOME CATALINA_HOME
```

- 5 Leave the zone.

▼ Example: Modifying the Apache Tomcat Configuration File

In this task you will create the configuration file to prepare the Apache Tomcat database. It is assumed, that you are still logged in to this zone, otherwise log into this zone.

1 Switch to the tomcat user.

```
zone# su - tomcat
```

2 Copy the Apache Tomcat configuration file from the agent directory to its deployment location.

```
zone# cp /opt/SUNWsctomcat/bin/pfile /global/mnt3
phys-schost-1# chown tomcat:tomcat /global/mnt3/pfile
```

3 Add this cluster's information to the parameter file /global/mnt3/pfile.

The following listing shows the relevant file entries and the values to assign to each entry.

```
EnvScript=/global/mnt3/tomcat/env.ksh
User=tomcat
Basepath=/global/mnt3/tomcat/apache-tomcat-5.5.17
Host=ha-host-1
Port=8080
TestCmd="get /index.jsp"
ReturnString="CATALINA"
Startwait=20
```

As an alternative, you can use the following set of variables:

```
EnvScript=/global/mnt3/tomcat/env.ksh
User=tomcat
Basepath=/global/mnt3/tomcat/apache-tomcat-5.5.17
TestUrl="http://ha-host-1:8080/"
ReturnString="CATALINA"
Startwait=20
```

4 Save and close the file.

5 Leave the zone.

6 Copy the Apache Tomcat configuration file from the agent directory to its deployment location.

```
phys-schost-1# cp /opt/SUNWsctomcat/util/sctomcat_config /global/mnt3
```

7 Add this cluster's information to the sctomcat_config configuration file.

The following listing shows the relevant file entries and the values to assign to each entry.

```
RS=RS-TOM
RG=RG-TOM
PORT=8080
LH=ha-host-1
NETWORK=true
SCALABLE=false
```

```
PFILE=/global/mnt3/pfile  
HAS_RS=RS-TOM-HAS
```

- 8 Save and close the file.

▼ Example: Enabling the Apache Tomcat Software to Run in the Cluster

This task will register the Apache Tomcat resource, it is essential, that you perform it on one node only.

- 1 Run the `sctomcat_register` script to register the resource.

```
phys-schost-1# ksh /opt/SUNWsctomcat/util/sctomcat_register \  
-f /global/mnt3/sctomcat_config
```

- 2 Enable the resource.

```
phys-schost-1# clresource enable RS-TOM
```

Index

C

clnode command, 10

Command

- clresourcegroup, 24, 26, 30, 31

- clresourcecetype, 23, 29

commands

- clresource, 46, 58

- clresourcegroup, 24, 25, 30, 31

- clresourcegroup, 35

- cluster, 58

- groupadd, 27, 32, 36

- node information, 10

- shutdown.sh, 39

- startup.sh, 38

- su, 27, 32, 36

- useradd, 27, 32, 36

configurations

- failover configuration, 19

- multiple master configuration, 17

- multiple masters configuration, 16

- scalable configuration, 16

D

debugging, HA for Apache Tomcat, 62–63

dependencies, 19

deployment examples

- Apache Tomcat in a failover zone, 77

- Apache Tomcat in a local zone, 77

- Apache Tomcat in a non-global zone, 77

- Apache Tomcat in the global zone, 65–69, 71–76

E

enable resource group, 25, 31

enable Tomcat resource, 46, 50, 54

enabling resource group, 35

environment scripts, 28, 33, 37

examples

- Apache Tomcat in a failover zone, 77

- Apache Tomcat in a local zone, 77

- Apache Tomcat in a non-global zone, 77

- Apache Tomcat in the global zone, 65–69, 71–76

F

fault monitor, 61–62

G

global zone, 40

H

HA for Apache Tomcat

- debugging, 62–63

- installing, 40–42

- software packages, installing, 40–42

help, 10–11

I

install, 26–29, 32–34, 36–38
installing, HA for Apache Tomcat, 40–42

L

load_balancing_policy, 17
local zones, *See* non-global zones

M

messages file, 11

N

non-global zones, 40

P

packages, 40–42
parameter file, 42, 46, 50, 59–60
pfile
 See parameter file
probing, 61–62
prtconf -v command, 10
prtdiag -v command, 10
psrinfo -v command, 10

R

register, 44, 48, 52
registration files, 21
registration scripts, 44, 48, 52
requirements, 18–22
resource group, enabling, 35
Resource Types
 SUNW.gds, 23, 29
resources, Apache Tomcat application
 debugging, 62–63
restrictions, 17–18

restrictions (*Continued*)

 zones, 40

S

scalability, 15–17
show- rev subcommand, 10
showrev -p command, 10
software packages, 40–42
strategy, 60

T

technical support, 10–11

V

/var/adm/messages file, 11

Z

zones, 40