



Sun Cluster Data Service for MaxDB Guide for Solaris OS



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 819-2982-11
May 2007, Revision A

Copyright 2007 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2007 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs brevets américains ou des applications de brevet en attente aux États-Unis et dans d'autres pays.

Cette distribution peut comprendre des composants développés par des tierces personnes.

Certains composants de ce produit peuvent être dérivées du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux États-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux États-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux États-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de cette publication et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des États-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

Contents

Preface	5
Installing and Configuring Sun Cluster HA for MaxDB	9
Sun Cluster HA for MaxDB Overview	10
Overview of the Installation and Configuration Process for Sun Cluster HA for MaxDB	10
Planning the Sun Cluster HA for MaxDB Installation and Configuration	12
Configuration Requirements	12
Supported Configurations of This Data Service	13
Configuration Considerations	15
Configuration Planning Questions	16
Installing and Configuring MaxDB	16
▼ How to Install and Configure MaxDB	16
▼ How to Enable MaxDB to Run in a Cluster	18
Verifying the MaxDB Installation and Configuration	20
▼ How to Verify MaxDB Installation and Configuration on Each Node	20
Installing the Sun Cluster HA for MaxDB Packages	22
▼ How to Install the Sun Cluster HA for MaxDB Packages	22
Configuring the HASStoragePlus Resource Type to Work With Sun Cluster HA for MaxDB ...	24
▼ How to Register and Configure an HASStoragePlus Resource	24
Registering and Configuring Sun Cluster HA for MaxDB	25
Setting Sun Cluster HA for MaxDB Extension Properties	26
Administering SAP xsriver as a User Other Than Root	26
▼ How to Register and Configure an SAP xsriver Resource	27
▼ How to Register and Configure a MaxDB Resource	29
Tuning the Sun Cluster HA for MaxDB Fault Monitors	32
Factors That Affect the Interval Between Fault Monitor Probes	33
Operations by the Sun Cluster HA for MaxDB Fault Monitors During a Probe	33
Faults Detected by the Sun Cluster HA for MaxDB Fault Monitors	34

Forcing the MaxDB Database Instance to Be Restarted if the Parent Kernel Process Is Terminated	35
Verifying the Sun Cluster HA for MaxDB Installation and Configuration	35
▼ How to Verify the Operation of the MaxDB Fault Monitor	36
▼ How to Verify the Operation of the SAP xserver Fault Monitor	37
Upgrading the SUNW.sap_xserver Resource Type	38
Information for Registering the New Resource Type Version	39
Information for Migrating Existing Instances of the Resource Type	39
A Sun Cluster HA for MaxDB Extension Properties	41
SUNW.sapdb Extension Properties	41
SUNW.sap_xserver Extension Properties	44
Index	47

Preface

Sun Cluster Data Service for MaxDB Guide for Solaris OS explains how to install and configure Sun™ Cluster HA for MaxDB on both SPARC® based systems and x86 based systems.

Note – This Sun Cluster release supports systems that use the SPARC and x86 families of processor architectures: UltraSPARC, SPARC64, and AMD64. In this document, the label x86 refers to systems that use the AMD64 family of processor architectures.

This document is intended for system administrators with extensive knowledge of Sun 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 Solaris™ Operating System (Solaris OS) and expertise with the volume-manager software that is used with Sun Cluster software.

Note – Sun Cluster software runs on two platforms, SPARC and x86. The information in this document pertains to both platforms unless otherwise specified in a special chapter, section, note, bulleted item, figure, table, or example.

Using UNIX Commands

This document contains information about commands that are specific to installing and configuring Sun 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 Solaris Operating System
- 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% you have mail.</code>
AaBbCc123	What you type, contrasted with onscreen computer output	<code>machine_name% su</code> 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 the C shell, Bourne shell, and Korn shell.

TABLE P-2 Shell Prompts

Shell	Prompt
C shell	<code>machine_name%</code>
C shell for superuser	<code>machine_name#</code>
Bourne shell and Korn shell	<code>\$</code>
Bourne shell and Korn shell for superuser	<code>#</code>

Related Documentation

Information about related Sun Cluster topics is available in the documentation that is listed in the following table. All Sun Cluster documentation is available at <http://docs.sun.com>.

Topic	Documentation
Data service administration	<i>Sun Cluster Data Services Planning and Administration Guide for Solaris OS</i> Individual data service guides
Concepts	<i>Sun Cluster Concepts Guide for Solaris OS</i>
Overview	<i>Sun Cluster Overview for Solaris OS</i>
Software installation	<i>Sun Cluster Software Installation Guide for Solaris OS</i>
System administration	<i>Sun Cluster System Administration Guide for Solaris OS</i>
Hardware administration	<i>Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS</i> Individual hardware administration guides
Data service development	<i>Sun Cluster Data Services Developer's Guide for Solaris OS</i>
Error messages	<i>Sun Cluster Error Messages Guide for Solaris OS</i>
Command and function reference	<i>Sun Cluster Reference Manual for Solaris OS</i>

For a complete list of Sun Cluster documentation, see the release notes for your release of Sun Cluster at <http://docs.sun.com>.

Related Third-Party Web Site References

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

Note – Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun 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. Sun 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, Support, and Training

The Sun web site provides information about the following additional resources:

- [Documentation](http://www.sun.com/documentation/) (<http://www.sun.com/documentation/>)
- [Support](http://www.sun.com/support/) (<http://www.sun.com/support/>)
- [Training](http://www.sun.com/training/) (<http://www.sun.com/training/>)

Getting Help

If you have problems installing or using Sun 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 Solaris Operating System (for example, Solaris 10)
- The release number of Sun Cluster (for example, Sun Cluster 3.2)

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>SPARC: prtdiag -v</code>	Displays system diagnostic information
<code>/usr/cluster/bin/clnode show-rev</code>	Displays Sun Cluster release and package version information

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

Installing and Configuring Sun Cluster HA for MaxDB

This chapter explains how to install and configure Sun Cluster HA for MaxDB. From version 7.5, SAP DB is distributed under the name MaxDB for MySQL (MaxDB). In this book, all versions of this database are referred to as MaxDB.

Note – If you are using the Solaris 10 OS, install and configure this data service to run only in the global zone. At publication of this document, this data service is not supported in non-global zones. For updated information about supported configurations of this data service, contact your Sun service representative.

This chapter contains the following sections.

- “Sun Cluster HA for MaxDB Overview” on page 10
- “Overview of the Installation and Configuration Process for Sun Cluster HA for MaxDB” on page 10
- “Planning the Sun Cluster HA for MaxDB Installation and Configuration” on page 12
- “Installing and Configuring MaxDB” on page 16
- “Verifying the MaxDB Installation and Configuration” on page 20
- “Installing the Sun Cluster HA for MaxDB Packages” on page 22
- “Configuring the HAS_{torageP}lus Resource Type to Work With Sun Cluster HA for MaxDB” on page 24
- “Registering and Configuring Sun Cluster HA for MaxDB” on page 25
- “Tuning the Sun Cluster HA for MaxDB Fault Monitors” on page 32
- “Verifying the Sun Cluster HA for MaxDB Installation and Configuration” on page 35
- “Upgrading the SUNW.sap_xserver Resource Type” on page 38

Sun Cluster HA for MaxDB Overview

To eliminate single points of failure in an MaxDB system, Sun Cluster HA for MaxDB provides the following features:

- Fault monitoring and automatic failover for the MaxDB application. You must configure Sun Cluster HA for MaxDB as a failover data service.
- Fault monitoring and automatic restart for SAP xserver. You must configure SAP xserver as a scalable data service.

For conceptual information about failover data services and scalable data services, see the *Sun Cluster Concepts Guide for Solaris OS*.

Each component of MaxDB has data service that protects the component when the component is configured in Sun Cluster. See the following table.

Note – The files that are associated with the `SUNW.sap_xserver` resource type are supplied with the Sun Cluster HA for SAP liveCache data service. The Sun Cluster HA for SAP liveCache data service is installed when you install Sun Cluster HA for MaxDB data service.

TABLE 1 Protection of MaxDB Components by Sun Cluster Data Services

MaxDB Component	Data Service
MaxDB	Sun Cluster HA for MaxDB The resource type is <code>SUNW.sapdb</code> .
SAP xserver	Sun Cluster HA for SAP liveCache The resource type is <code>SUNW.sap_xserver</code> .
NFS file system	Sun Cluster HA for NFS The resource type is <code>SUNW.nfs</code> . For more information about this data service, see <i>Sun Cluster Data Service for NFS Guide for Solaris OS</i> .

Overview of the Installation and Configuration Process for Sun Cluster HA for MaxDB

The following table summarizes the tasks for installing and configuring Sun Cluster HA for MaxDB and provides cross-references to detailed instructions for performing these tasks. Perform the tasks in the order that they are listed in the table.

TABLE 2 Tasks for Installing and Configuring Sun Cluster HA for MaxDB

Task	Cross-Reference
Plan the Sun Cluster HA for MaxDB installation and configuration	Your SAP documentation. <i>Sun Cluster Data Services Planning and Administration Guide for Solaris OS</i> “Planning the Sun Cluster HA for MaxDB Installation and Configuration” on page 12
Install and configure MaxDB	“Installing and Configuring MaxDB” on page 16
Verify the MaxDB installation and configuration	“Verifying the MaxDB Installation and Configuration” on page 20
Install the Sun Cluster HA for MaxDB packages	“Installing the Sun Cluster HA for MaxDB Packages” on page 22
Configure the HASto ragePlus resource to work with Sun Cluster HA for MaxDB	“Relationship Between Resource Groups and Device Groups” in <i>Sun Cluster Data Services Planning and Administration Guide for Solaris OS</i> “Synchronizing the Startups Between Resource Groups and Device Groups” in <i>Sun Cluster Data Services Planning and Administration Guide for Solaris OS</i> “Configuring the HASto ragePlus Resource Type to Work With Sun Cluster HA for MaxDB” on page 24
Register and configure the Sun Cluster HA for MaxDB data service	“Registering and Configuring Sun Cluster HA for MaxDB” on page 25
(Optional) Tune the Sun Cluster HA for MaxDB fault monitors	“Tuning the Sun Cluster HA for MaxDB Fault Monitors” on page 32
(Optional) Use an alternate project identifier (ID)	“Data Service Project Configuration” in <i>Sun Cluster Concepts Guide for Solaris OS</i>
Verify the Sun Cluster HA for MaxDB installation and configuration	“Verifying the Sun Cluster HA for MaxDB Installation and Configuration” on page 35
(Optional) Upgrade the SUNW.sap_xserver resource type	“Upgrading the SUNW.sap_xserver Resource Type” on page 38

Planning the Sun Cluster HA for MaxDB Installation and Configuration

This section contains the information that you need to plan your Sun Cluster HA for MaxDB installation and configuration.

Note – Before you begin, consult your MaxDB documentation for configuration restrictions and requirements that are not imposed by Sun Cluster software. For information about restrictions that the Sun Cluster software imposes, see the Sun Cluster documentation.

Configuration Requirements

The configuration requirements in this section apply only to Sun Cluster HA for MaxDB.



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

For requirements that apply to all data services, see “Configuration Guidelines for Sun Cluster Data Services” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

MaxDB Software Version Requirements

Use MaxDB version 7.4 or compatible versions.

Sun Cluster HA for MaxDB Configuration Requirements

Configure Sun Cluster HA for MaxDB as a failover data service. You cannot configure Sun Cluster HA for MaxDB as a scalable data service. For more information, see the following sections:

- “How to Enable MaxDB to Run in a Cluster” on page 18
- “How to Register and Configure a MaxDB Resource” on page 29

SAP xserver Configuration Requirements

To enable client applications to access Sun Cluster HA for MaxDB, you must use SAP xserver. Configure SAP xserver as a scalable data service. Do *not* configure SAP xserver as a failover data service.

Configure SAP xserver so that SAP xserver starts on all nodes to which the MaxDB resource can fail over. To implement this configuration, ensure that the node list of the SAP xserver resource group contains all nodes that are in the node list of the MaxDB resource group. For more information, see “How to Register and Configure an SAP xserver Resource” on page 27.

Supported Configurations of This Data Service

The Sun Cluster HA for MaxDB data service supports configurations that conform to the requirements in [“Configuration Requirements” on page 12](#).

If you plan to use MaxDB with other highly available SAP applications, you must also configure the Sun Cluster data services for those applications. For more information, see the following table.

SAP Application	Sun Cluster Data Service	Associated Document
SAP R/3	Sun Cluster HA for SAP	<i>Sun Cluster Data Service for SAP Guide for Solaris OS</i>
SAP liveCache	Sun Cluster HA for SAP liveCache	<i>Sun Cluster Data Service for SAP liveCache Guide for Solaris OS</i>
SAP Web Application Server	Sun Cluster HA for SAP Web Application Server	<i>Sun Cluster Data Service for SAP Web Application Server Guide for Solaris OS</i>

The examples that follow show these supported configurations of Sun Cluster HA for MaxDB:

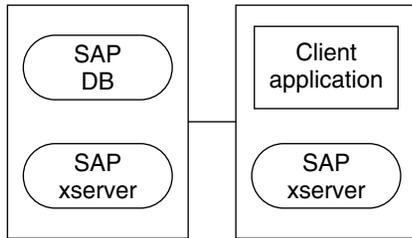
- Two-node cluster configuration
- Four-node cluster configuration with SAP R/3
- Four-node cluster configuration with SAP R/3 and SAP liveCache

Note – Sun Cluster HA for MaxDB might support additional configurations. However, you must contact your Sun service provider for information about additional configurations.

EXAMPLE 1 Two-Node Configuration

This example shows a two-node configuration in which a client application accesses the MaxDB resource through the SAP xserver resource. The characteristics of this configuration are as follows:

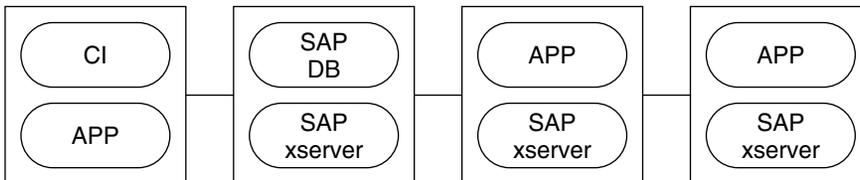
- The MaxDB resource is configured as a failover data service.
- The SAP xserver resource is configured as a scalable data service.



EXAMPLE 2 Four-Node Configuration With SAP R/3

This example shows a four-node configuration in which MaxDB is used with SAP R/3. This configuration uses multiple Advanced Planner & Optimizer (APO) application servers. The characteristics of this configuration are as follows:

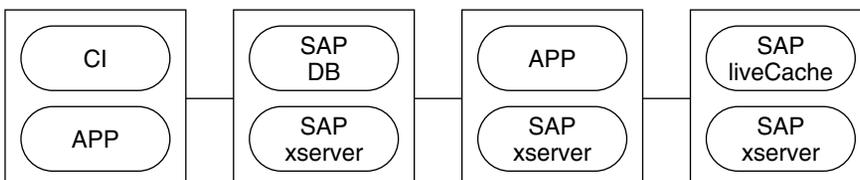
- The SAP APO Central Instance (CI) resource is configured as a failover data service.
- The MaxDB resource is configured as a failover data service.
- The SAP xserver resource is configured as a scalable data service.
- APO application server (APP) resources are configured as scalable data services.



EXAMPLE 3 Four-Node Configuration With SAP R/3 and SAP liveCache

This example shows a four-node configuration in which MaxDB is used with SAP R/3 and SAP liveCache. This configuration uses multiple APO application servers. The characteristics of this configuration are as follows:

- The SAP APO CI resource is configured as a failover data service.
- The MaxDB resource is configured as a failover data service.
- The SAP xserver resource is configured as a scalable data service.
- APP resources are configured as scalable data services.
- The SAP liveCache resource is configured as a failover data service.



Configuration Considerations

The configuration considerations in the subsections that follow affect the installation and configuration of Sun Cluster HA for MaxDB.

Device Group for the MaxDB Application

Ensure that you create a device group for the MaxDB application as follows:

- Install MaxDB on its own global device group. For more information, see [“Installing and Configuring MaxDB” on page 16](#). This separate global device group for MaxDB ensures that the MaxDB resource can depend on the HASStoragePlus resource only for MaxDB.
- Create an HASStoragePlus resource for the global device group on which MaxDB is installed. For more information, see [“Configuring the HASStoragePlus Resource Type to Work With Sun Cluster HA for MaxDB” on page 24](#).
- Ensure that the resource for MaxDB depends on the HASStoragePlus resource for the global device group on which MaxDB is installed. For more information, see [“Registering and Configuring Sun Cluster HA for MaxDB” on page 25](#).

Dependencies of the MaxDB Application on SAP xserver

Configure MaxDB so that MaxDB starts only on a node where SAP xserver is running. To implement this configuration, configure resources and resource groups as follows:

- Ensure that the resource for MaxDB depends on the resource for SAP xserver.
- Create on the MaxDB resource group a strong positive affinity for the SAP xserver resource group.

For more information, see [“Registering and Configuring Sun Cluster HA for MaxDB” on page 25](#).

Administration of SAP xserver by a User Other Than Root

You might be required to administer SAP xserver as a user other than root. In this situation, you must create and define that user as follows:

- You must create that user on all cluster nodes that master SAP xserver.
- You must define that user when you register and configure Sun Cluster HA for MaxDB. For more information, see [“Registering and Configuring Sun Cluster HA for MaxDB” on page 25](#).

Configuration Planning Questions

Answer the questions in this section to plan the installation and configuration of Sun Cluster HA for MaxDB. Write the answers to these questions in the space that is provided on the data service worksheets in Appendix D, “Data Service Configuration Worksheets and Examples,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

- Which resource group will you use for the MaxDB application resource and for the logical host name for the MaxDB resource?
Use the answer to this question when you perform the procedure “[How to Enable MaxDB to Run in a Cluster](#)” on page 18.
- What is the logical host name for the MaxDB resource? Clients access the data service through this logical host name.
Use the answer to this question when you perform the following procedures:
 - “[How to Install and Configure MaxDB](#)” on page 16
 - “[How to Enable MaxDB to Run in a Cluster](#)” on page 18
- Where will the system configuration files reside?
See *Sun Cluster Data Services Planning and Administration Guide for Solaris OS* for the advantages and disadvantages of using the local file system instead of the cluster file system.

Installing and Configuring MaxDB

To enable Sun Cluster HA for MaxDB to make MaxDB highly available, additional installation and configuration operations are required. These operations supplement the standard installation and standard configuration of MaxDB.

During a standard installation, MaxDB is installed with a physical host name. To enable MaxDB to run in a cluster, you must modify MaxDB to use a logical host name.

For information about the standard installation and standard configuration of MaxDB, see the following documentation:

- If you are using MaxDB with SAP R/3, see the SAP R/3 documentation for information about how to install and configure SAP R/3 with MaxDB.
- If you are using MaxDB independently of SAP R/3, see the MaxDB documentation.

▼ How to Install and Configure MaxDB

- 1 **On one node of the cluster, install the MaxDB software.**
Ensure that you install MaxDB on its own global device group.

- 2 **Perform a standard configuration of MaxDB.**
- 3 **Create the `.XUSER.62` file in the home directory of the operating system (OS) user who administers the MaxDB instance.**

Note – If you are using MaxDB 7.5 or MaxDB 7.6, the UNIX user identity of the OS user who administers the MaxDB database is `sdb`. You need to specify this user when you create the MaxDB resource in a later step. Otherwise, the MaxDB fault monitor cannot probe the MaxDB database.

- a. **Create a plain text file that contains information about the database user who administers the MaxDB database instance.**

For information about the content of this file, see the MaxDB documentation. For the name of the server on which the database is running, specify the logical host name for the MaxDB resource that you specified in “[Configuration Planning Questions](#)” on page 16.

For an example of the content of this file, see [Example 4](#).

- b. **As MaxDB admin user, generate the `.XUSER.62` file from the plain text file that you created in [Step a](#).**

Use the MaxDB command `xuser` for this purpose.

```
# xuser -b user-info-file
```

`-b user-info-file` Specifies the plain text file from which you are generating the `.XUSER.62` file

- 4 **Copy the `/usr/spool/sql` directory and its contents from the node on which you installed MaxDB to all nodes where resources for MaxDB and SAP xserver will run.**

To ensure that the same owner owns the directory and its contents on all nodes, use the `tar(1)` command and the `rsh(1)`

```
# tar cfB - /usr/spool/sql | rsh destination tar xfB -
```

`destination` Specifies the node to which you are copying the `/usr/spool/sql` directory and its contents

The following example shows a plain text file that contains information about a database user who administers an MaxDB instance.

- 5 **If you are installing MaxDB 7.5 or MaxDB 7.6, perform this step. If you are installing a previous version, proceed to the next step.**

Copy the `/etc/opt/sdb` directory and its contents from the node on which you installed MaxDB to all nodes where resources for MaxDB and SAP xserver will run.

To ensure that the same owner owns the directory and its contents on all nodes, use the `tar(1)` command and the `rsh(1)`

```
# tar cfb - /etc/opt/sdb | rsh destination tar xfb -
```

destination Specifies the node to which you are copying the `/etc/opt/sdb` directory and its contents

Example 4 Information About a Database User Who Administers a MaxDB Instance

```
DEFAULT
dbm
dbm
TST
svr-1
  blank line
  blank line
-1
-1
  blank line
```

This example specifies the following information about a database user who administers an MaxDB instance:

- The user key that is used to address this combination of XUSER data is named DEFAULT.
- The user name and password of the database user are dbm.
- The name of the MaxDB instance is TST.
- The logical host name for the MaxDB resource is svr-1.
- No structured query language (SQL) mode is specified.
- The default time-out value of the MaxDB instance is used.
- The default isolation level of the MaxDB instance is used.

For more information, see the MaxDB documentation.

▼ How to Enable MaxDB to Run in a Cluster

- 1 Create a failover resource group to contain the MaxDB application resources and the logical host name for the MaxDB resource.

Use the resource group that you identified when you answered the questions in “[Configuration Planning Questions](#)” on page 16.

```
# clresourcegroup create -n nodelist sapdb-rg
```

sapdb-rg Specifies that the resource group that you are creating is named *sapdb-rg*.

-n *nodelist* Specifies a comma-separated list of nodes where the resource group can be brought online. Specifying a node list is optional. If you omit the node list, the resource group can be brought online on all cluster nodes.

2 Ensure that all network resources that you intend to you use are added to your name service database.

3 Add a logical host name resource to the failover resource group that you created in [Step 1](#).

```
# clreslogicalhostname create -g sapdb-rg \  
-h sapdb-logical-hostname sapdb-logical-hostname
```

-g *sapdb-rg*

Specifies that the logical host name's database resource is to be added to the failover resource group that you created in [Step 1](#).

-h

Specifies the host name list. You must use this option either when more than one logical host needs to be associated with the new MaxDB resource or when the logical host does not have the same name as the MaxDB resource itself. MaxDB is the resource for the *sapdb-logical-hostname* that you specified in “[Configuration Planning Questions](#)” on [page 16](#).

sapdb-logical-hostname

Specifies the logical host name of the server on which the database is running. This host name must be the logical host name for the MaxDB resource that you specified in “[Configuration Planning Questions](#)” on [page 16](#).

Note – If you require a fully qualified hostname, you must specify the fully qualified name with the -h option and you cannot use the fully qualified form in the resource name.

4 Enable the resource group that you created in [Step 1](#).

```
# clresourcegroup online -emM sapdb-rg
```

-emM Enables the resource group *sapdb-rg* created in [Step 1](#) and moves it to the MANAGED state.

Example 5 Enabling MaxDB to Run in a Cluster

This example shows the sequence of commands that are required to enable MaxDB to run in a cluster. The commands are run on only one cluster node.

1. The following command creates a failover resource group to contain the MaxDB application resources and the logical host name for the MaxDB resource. The resource group is named *sapdbrg*. The *sapdbrg* resource group can be brought online on all cluster nodes.

```
# clresourcegroup create sapdbrg
```

- The following command adds a logical host name resource to the sapdbrg resource group. The logical host name of the server on which the database is running is `svr-1`. When the sapdbrg resource group is brought online, an attempt is made for each node to discover a network interface on the subnet that the host name list identifies.

```
# clreslogicalhostname create -g sapdbrg -h sdrvr-1 -d sapdbl
```

Note – If you require a fully qualified hostname, you must specify the fully qualified name with the `-h` option and you cannot use the fully qualified form in the resource name.

- The following command moves the sapdbrg resource group to the MANAGED state, brings the resource group online, and monitors the resources in the resource group.

```
# clresourcegroup online -emM sapdbrg
```

Verifying the MaxDB Installation and Configuration

Before you install the Sun Cluster HA for MaxDB packages, verify that the MaxDB software is correctly installed and configured to run in a cluster. This verification does *not* verify that the MaxDB application is highly available because the Sun Cluster HA for MaxDB data service is not yet installed.

▼ How to Verify MaxDB Installation and Configuration on Each Node

Perform this procedure on each node that can master the MaxDB resource group.

- Log in as superuser to a node that can master the MaxDB resource group.**
- Switch the MaxDB resource group to the node that you logged in to in [Step 1](#).**

```
# clresourcegroup switch -n node sapdb-rg
```

`-n node` Specifies the node to which the MaxDB resource group is to be switched

`sapdb-rg` Specifies that the MaxDB resource group `sapdb-rg` is to be switched to another node

3 Confirm that the MaxDB database can be started and be stopped.

a. Become the OS user who administers the MaxDB database.

```
# su - os-sapdb-adm-user
```

os-sapdb-adm-user Specifies the UNIX user identity of the OS user who administers the MaxDB database. This user's home directory contains the `.XUSER.62` file that was created during the installation and configuration of MaxDB. For more information, see [“Installing and Configuring MaxDB” on page 16](#). You specify this user when you set the `DB_User` extension property as explained in [“How to Register and Configure a MaxDB Resource” on page 29](#).

If you are using MaxDB 7.5 or MaxDB 7.6, the UNIX user identity of the OS user who administers the MaxDB database is `sdb`.

b. Start the SAP xserver.

```
$ x_server start
```

c. Manually start the MaxDB database instance on the node that you logged in to in [Step 1](#).

```
$ dbmcli -U sapdb-adm-key db_online
```

-U sapdb-adm-key Specifies that the `dbmcli` command is run with the user key of the database user who administers the MaxDB instance. This user key is created during the installation and configuration of MaxDB. For more information, see [“Installing and Configuring MaxDB” on page 16](#). You specify this user key when you set the `User_Key` extension property as explained in [“How to Register and Configure a MaxDB Resource” on page 29](#).

d. Confirm that the MaxDB database instance is started.

e. Manually stop the MaxDB database instance.

```
$ dbmcli -U sapdb-adm-key db_offline
```

-U sapdb-adm-key Specifies that the `dbmcli` command is run with the user key that you used for starting the database in [Step c](#)

f. Confirm that the MaxDB database instance is stopped.

Installing the Sun Cluster HA for MaxDB Packages

If you did not install the Sun Cluster HA for MaxDB packages during your initial Sun Cluster installation, perform this procedure to install the packages. To install the packages, use the Sun Java™ Enterprise System Installation Wizard.

▼ How to Install the Sun Cluster HA for MaxDB Packages

Perform this procedure on each cluster node where you are installing the Sun Cluster HA for MaxDB packages.

You can run the Sun Java Enterprise System Installation Wizard 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 – Install the packages for this data service in the global zone.

Before You Begin Ensure that you have the Sun Java Availability Suite DVD-ROM.

If you intend to run the Sun Java Enterprise System Installation Wizard 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 Sun Java Availability Suite DVD-ROM into the DVD-ROM drive.

If the Volume Management daemon `volmgr(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 Sun Java Enterprise System 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 Sun Java Enterprise System Installation Wizard.

```
# ./installer
```

5 When you are prompted, accept the license agreement.

If any Sun Java Enterprise System components are installed, you are prompted to select whether to upgrade the components or install new software.

6 From the list of Sun Cluster agents under Availability Services, select the data service for MaxDB.

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 Sun Java Enterprise System 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 Sun Java Enterprise System Installation Wizard with the CLI, omit this step

11 Exit the Sun Java Enterprise System Installation Wizard.

12 Unload the Sun Java Availability Suite DVD-ROM 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
```

Next Steps Go to “Configuring the HASStoragePlus Resource Type to Work With Sun Cluster HA for MaxDB” on page 24.

Configuring the HASStoragePlus Resource Type to Work With Sun Cluster HA for MaxDB

For maximum availability of the MaxDB database, resources that Sun Cluster HA for MaxDB requires must be available before the MaxDB database instance is started. An example of such a resource is the file system where programs and libraries for the MaxDB runtime environment reside. To ensure that these resources are available, configure the HASStoragePlus resource type to work with Sun Cluster HA for MaxDB.

For information about the relationship between resource groups and disk device groups, see “Relationship Between Resource Groups and Device Groups” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

Configuring the HASStoragePlus resource type to work with Sun Cluster HA for MaxDB involves the following operation:

- Registering and configuring an HASStoragePlus resource

▼ How to Register and Configure an HASStoragePlus Resource

Perform this procedure on any one cluster node.

1 Register the SUNW.HASStoragePlus resource type.

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

2 Create an HASStoragePlus resource for the global device group on which MaxDB is installed.

Create this resource in the MaxDB resource group. This resource must perform an affinity switchover for all global devices that are defined for this resource.

```
# clresource create -d -g sapdb-rg \  
-t SUNW.HASStoragePlus -p filesystemmountpoints=mountpoint-list \  
-p globaldevicepaths=sapdb-device-group \  
-p affinityon=TRUE hsp-resource
```

-d

Specifies that a new resource is to be created in a disabled state.

-g sapdb-rg

Specifies that the resource is to be added to the MaxDB resource group.

-t SUNW.HASStoragePlus

Specifies that the resource is an instance of the SUNW.HASStoragePlus resource type.

- p filesystemmountpoints=*mountpoint-list*
Specifies a list of valid mount points for the file system.
- p globaldevicepaths=*sapdb-device-group*
Specifies the name of the global device group on which the MaxDB software is installed.
- p affinityon=TRUE
Specifies that this resource performs an affinity switchover for all global devices that are defined for this resource.

hsp-resource

Specifies that the resource that you are creating is named *hsp-resource*.

The resource is created in the enabled state.

Example 6 Creating an HASStoragePlus Resource

```
# clresource create -g sapdbrg \  
-t SUNW.HASStoragePlus -p filesystemmountpoints=/global/sapbdbdata \  
-p globaldevicepaths=sapdbdg -p affinityon=TRUE hsprs
```

This example shows the creation of a SUNW.HASStoragePlus resource that has the following characteristics:

- The resource is named *hsprs*.
- The resource is a member of a resource group that is named *sapdbrg*. The creation of this resource group is shown in [Example 5](#).
- The resource is an instance of the SUNW.HASStoragePlus resource type. The registration of this resource type is not shown in this example.
- The mount point for the file system is */global/sapbdbdata*.
- The MaxDB software is installed on a global device group that is named *sapdbdg*.
- The *hsprs* resource performs an affinity switchover for all global devices that are defined for this resource.

Next Steps Go to [“Registering and Configuring Sun Cluster HA for MaxDB”](#) on page 25.

Registering and Configuring Sun Cluster HA for MaxDB

To enable Sun Cluster HA for MaxDB to make MaxDB highly available, configure Sun Cluster data services as follows:

- Configure SAP xserver as a scalable data service.
- Configure Sun Cluster HA for MaxDB as a failover data service.

Before you perform this procedure, ensure that the Sun Cluster HA for MaxDB data service packages are installed.



Caution – One SAP xserver serves multiple MaxDB instances and, if SAP liveCache is used, multiple SAP liveCache instances in the cluster. Therefore, do *not* configure more than one SAP xserver resource on the same cluster. If more than one SAP xserver resource runs on the same cluster, conflicts between the SAP xserver resources occur. These conflicts cause all SAP xserver resources to become unavailable. If you attempt to start the SAP xserver a second time, the attempt fails. The error message `Address already in use` is also displayed.

Setting Sun Cluster HA for MaxDB Extension Properties

The sections that follow contain instructions for registering and configuring resources. These instructions explain how to set *only* extension properties that Sun Cluster HA for MaxDB requires you to set. For information about all Sun Cluster HA for MaxDB extension properties, see [Appendix A](#). You can update some extension properties dynamically. You can update other properties, however, only when you create or disable a resource. The Tunable entry indicates when you can update a property.

To set an extension property of a resource, include the following option to modify the resource:

`-p property=value`

`-p property` Identifies the extension property that you are setting

`value` Specifies the value to which you are setting the extension property

You can also use the procedures in Chapter 2, “Administering Data Service Resources,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS* to configure resources after the resources are created.

Administering SAP xserver as a User Other Than Root

You might be required to administer SAP xserver as a user other than root. In this situation, you must create and define that user as follows:

- You must create that user on all cluster nodes that master SAP xserver.
- You must define that user when you register and configure an SAP xserver resource. To define the user who administers SAP xserver, set the `Xserver_User` extension property when you create an SAP xserver resource. For more information about the `Xserver_User` extension property, see “[SUNW.sap_xserver Extension Properties](#)” on page 44.

▼ How to Register and Configure an SAP xserver Resource

1 Become superuser on a cluster node.

2 Register the `SUNW.sap_xserver` resource type.

```
# clresourcetype register SUNW.sap_xserver
```

3 Create a scalable resource group for the SAP xserver resource.

Configure SAP xserver so that SAP xserver starts on all nodes to which the MaxDB resource can fail over. To implement this configuration, ensure that the node list of the SAP xserver resource group contains all nodes that are in the node list of the MaxDB resource group. This resource group is created when the procedure “[How to Enable MaxDB to Run in a Cluster](#)” on page 18 is performed.

```
# clresourcegroup create -n nodelist \
-p Maximum primaries=nodes-in-sapdb-rg \
-p Desired primaries=nodes-in-sapdb-rg xserver-rg
```

-n *nodelist*

Specifies a comma-separated list of nodes where the resource group can be brought online. Ensure that this node list contains all nodes that are in the node list of the MaxDB resource group.

-p *Maximum primaries=nodes-in-sapdb-rg*

Specifies the maximum number of nodes on which the SAP xserver resource can start. This number is equal to the number of nodes that are in the node list of the MaxDB resource group. You must specify the same number as the value of the `Desired primaries` property.

-p *Desired primaries=nodes-in-sapdb-rg*

Specifies the desired number of nodes on which the SAP xserver resource can start. This number is equal to the number of nodes that are in the node list of the MaxDB resource group. You must specify the same number as the value of the `Maximum primaries` property.

xserver-rg

Specifies that the resource group that you are creating is named *xserver-rg*.

4 Add the `HASStoragePlus` resource to the SAP xserver resource group.

```
# clresource create -g xserver-rg \
-t SUNW.HASStoragePlus \
-p filesystemmountpoints=mountpoints \
-p affinityon=false xserver-storage-resource
```

-g *xserver-rg*

Specifies that the resource that you are creating is added to the resource group *xserver-rg*.

- t SUNW.HASStoragePlus
Specifies that the resource type of the resource you are creating is SUNW.HASStoragePlus.
- p filesystemmountpoints=*mountpoint*, ...
Specifies the mount points for the resource group you are creating.
- p affinityon=false
Indicates that the SAP xserver resource does not have to be colocated with the device group.

xserver-storage-resource

Specifies that the resource that you are creating is named *xserver-storage-resource*.

The resource is created in the enabled state.

For more details on how to set up an HASStoragePlus resource, see “Enabling Highly Available Local File Systems” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

5 Create an SAP xserver resource in the resource group that you created in [Step 3](#).

```
# clresource create -d -g xserver-rg \  
-t SUNW.sap_xserver \  
-p resource_dependencies=xserver-storage-resource xserver-resource
```

-g *xserver-rg*

Specifies that the resource is to be added to the resource group that you created in [Step 3](#)

-t SUNW.sap_xserver

Specifies that the resource is an instance of the SUNW.sap_xserver resource type

-p resource_dependencies=*xserver-storage-resource*

Sets a resource dependency between *xserver-storage-resource* and the HASStoragePlus resource we created in [Step 4](#).

xserver-resource

Specifies that the resource that you are creating is named *xserver-resource*.

The resource is created in the enabled state.

6 Enable the resource group that you created in [Step 3](#).

```
# clresourcegroup online -emM xserver-rg
```

-emM Enables the resource group and moves it to the MANAGED state.

xserver-rg Specifies the name of the resource group.

Example 7 Configuring a SUNW.sap_xserver Resource

This example shows the sequence of commands that are required to configure an SAP xserver resource. The commands are run on only one cluster node.

1. The following command creates a scalable resource group to contain an SAP xserver resource for a four-node cluster. The resource group is named `xsrvrrg`. The `xsrvrrg` resource group can be brought online on all cluster nodes.

```
# clresourcegroup create \  
-p Maximum primaries=4 \  
-p Desired primaries=4 xsrvrrg
```

2. The following command adds the `HASStoragePlus` resource to the SAP xserver resource group. The `xserver-storage-resource` is called `xsstorrs` and the mount point, `mtp1`.

```
# clresource create -d -g xsrvrrg \  
-t SUNW.HASStoragePlus \  
-p filesystemmountpoints=mtp1 \  
-p affinityon=false xsstorrs
```

The resource is created in the enabled state.

3. The following command creates an SAP xserver resource that is named `xsvrrrs` in the `xsrvrrg` resource group. The SAP xserver resource is an instance of the `SUNW.sap_xserver` resource type. The `xserver-storage-resource` is `xsstorrs`. The registration of this resource type is not shown in this example.

```
# clresource create -d -g xsrvrrg -t SUNW.sap_xserver \  
-p resource_dependencies=xsstorrs xsvrrrs
```

The resource is created in the enabled state.

4. The following command moves the `xsrvrrg` resource group to the `MANAGED` state and brings the resource group online.

```
# clresourcegroup online -emM xsrvrrg
```

▼ How to Register and Configure a MaxDB Resource

- 1 Register the `SUNW.sapdb` resource type.

```
# clresourcetype register SUNW.sapdb
```

- 2 Create a MaxDB resource in the MaxDB resource group.

Ensure that the MaxDB resource depends on the following resources:

- The `HASStoragePlus` resource for the global device group on which MaxDB is installed.
- The SAP xserver resource.

When you create this resource, specify the following information about the MaxDB database instance. This information is created when MaxDB is installed and configured as explained in [“Installing and Configuring MaxDB” on page 16](#).

- The name of the MaxDB database.
- The UNIX user identity of the OS user who administers the MaxDB database instance. If you are using MaxDB 7.5 or MaxDB 7.6, the UNIX user identity of the OS user who administers the MaxDB database is `sdb`.
- The user key of the database user who administers the MaxDB database instance.

```
# clresource create -d -g sapdb-rg \  
-t SUNW.sapdb \  
-p DB_Name=db-name \  
-p DB_User=os-sapdb-adm-user \  
-p User_Key=sapdb-adm-key \  
-p resource_dependencies=hsp-resource,xserver-resource sapdb-rs
```

-d

Specifies that the resource that you are creating is not immediately enabled.

-g *sapdb-rg*

Specifies that the resource is to be added to the MaxDB resource group.

-t SUNW.sapdb

Specifies that the resource is an instance of the SUNW.sapdb resource type.

-p DB_Name=*db-name*

Specifies the name of the MaxDB database instance in uppercase.

-p DB_User=*os-sapdb-adm-user*

Specifies the UNIX user identity of the OS user who administers the MaxDB database. This user's home directory contains the `.XUSER.62` file that was created during the installation and configuration of MaxDB. For more information, see [“Installing and Configuring MaxDB” on page 16](#).

-p User_Key=*sapdb-adm-key*

Specifies the user key of the database user who administers the MaxDB database instance. This user key is created during the installation and configuration of MaxDB. For more information, see [“Installing and Configuring MaxDB” on page 16](#).

-p resource_dependencies=*hsp-resource,xserver-resource*

Specifies that the MaxDB resource depends on the following resources.

- The HASToragePlus resource for the global device group on which MaxDB is installed
- The SAP xserver resource

sapdb-rs

Specifies that the resource that you are creating is named *sapdb-rs*.

The resource is created in the enabled state.

3 Ensure that the MaxDB resource group is brought online only on a node where the SAP xserver resource group is online.

To meet this requirement, create on the MaxDB resource group a strong positive affinity for the SAP xserver resource group.

```
# clresourcegroup set -p rg_affinities=++xserver-rg sapdb-rg
```

```
-p rg_affinities=++xserver-rg
```

Specifies that the MaxDB resource group declares a strong positive affinity for the SAP xserver resource group.

```
sapdb-rg
```

Specifies that the MaxDB resource group is to be modified.

4 Enable the MaxDB resource group.

```
# clresourcegroup online -emM sapdb-rg
```

```
-emM      Enables the MaxDB resource group and moves it to the MANAGED state.
```

```
sapdb-rg  Specifies the name of the MaxDB resource group to be brought online.
```

5 (Optional) Consider configuring your cluster to prevent noncritical resource groups from being brought online on the same node as the MaxDB resource group.

You might plan to run lower-priority services on a node to which the MaxDB resource can fail over. In this situation, consider using resource group affinities to shut down the noncritical services when the MaxDB resource fails over to the node.

To specify this behavior, declare on the resource group for each noncritical service a strong negative affinity for the MaxDB resource group.

```
# clresourcegroup set -p rg_affinities=--sapdb-rg noncritical-rg
```

```
-p
```

Sets the resource group property to the specified value.

```
noncritical-rg
```

Specifies the name of the noncritical resource group.

Example 8 Creating a SUNW.sapdb Resource

```
# clresource create -d -g sapdb-rg -t SUNW.sapdb \
-p DB_Name=TST -p DB_User=dbadmin \
-p User_Key=DEFAULT -p resource_dependencies=hsprsr,xsrrvrs sapdb-rg
```

The resource is created in the enabled state.

This example shows the creation of a SUNW.sapdb resource that has the following characteristics:

- The resource is named `sapdbrs`.
- The resource is a member of a resource group that is named `sapdbrg`. The creation of this resource group is shown in [Example 5](#).
- The resource is an instance of the `SUNW.sapdb` resource type. The registration of this resource type is not shown in this example.
- The MaxDB database instance that is associated with this resource is named `TST`.
- The UNIX user identity of the OS user who administers the MaxDB database is `dbadmin`.
- The user key of the database user who administers the MaxDB database is `DEFAULT`.
- The MaxDB resource depends on the following resources:
 - An `HASStoragePlus` resource that is named `hsprs`. The creation of the `hsprs` resource is shown in [Example 6](#).
 - A `SUNW.sap_xserver` resource that is named `xsvrrs`. The creation of the `xsvrrs` resource is shown in [Example 7](#).

Tuning the Sun Cluster HA for MaxDB Fault Monitors

Fault monitoring for the Sun Cluster HA for MaxDB data service is provided by the following fault monitors:

- The MaxDB fault monitor
- The SAP xserver fault monitor

Each fault monitor is contained in a resource whose resource type is shown in the following table.

TABLE 3 Resource Types for Sun Cluster HA for MaxDB Fault Monitors

Fault Monitor	Resource Type
MaxDB	<code>SUNW.sapdb</code>
SAP xserver	<code>SUNW.sap_xserver</code>

System properties and extension properties of these resources control the behavior of the fault monitors. The default values of these properties determine the preset behavior of the fault monitors. The preset behavior should be suitable for most Sun Cluster installations. Therefore, you should tune the Sun Cluster HA for MaxDB fault monitors *only* if you need to modify this preset behavior.

Tuning the Sun Cluster HA for MaxDB fault monitors involves the following tasks:

- Setting the interval between fault monitor probes
- Setting the timeout for fault monitor probes

- Defining the criteria for persistent faults
- Specifying the failover behavior of a resource

For more information, see “Tuning Fault Monitors for Sun Cluster Data Services” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*. Information about the Sun Cluster HA for MaxDB fault monitors that you need to perform these tasks is provided in the subsections that follow.

Sun Cluster HA for MaxDB also enables you to control how the fault monitor responds if the MaxDB parent kernel process is not running. For more information, see [“Forcing the MaxDB Database Instance to Be Restarted if the Parent Kernel Process Is Terminated” on page 35](#).

Tune the Sun Cluster HA for MaxDB fault monitors when you register and configure Sun Cluster HA for MaxDB. For more information, see [“Registering and Configuring Sun Cluster HA for MaxDB” on page 25](#).

Factors That Affect the Interval Between Fault Monitor Probes

To determine whether SAP xserver and the MaxDB database instance are operating correctly, the Sun Cluster HA for MaxDB fault monitors probe these resources periodically. The optimum interval between fault monitor probes depends on the time that is required to respond to a fault in a resource. This time depends on how the complexity of the resource affects the time that is required for operations such as restarting the resource.

For example, SAP xserver is a much simpler resource and can be restarted much quicker than MaxDB. Therefore, the optimum interval between fault monitor probes of SAP xserver is shorter than the optimum interval between probes of MaxDB.

Operations by the Sun Cluster HA for MaxDB Fault Monitors During a Probe

The optimum timeout for fault monitor probes depends on the operations that a fault monitor performs to probe the resource.

Operations by the MaxDB Fault Monitor During a Probe

During a probe, the MaxDB fault monitor performs the following operations:

1. The MaxDB fault monitor determines whether the MaxDB database instance is online.

2. If the MaxDB database instance is online, the MaxDB fault monitor determines whether the parent kernel process of the MaxDB database instance is running. You can control how the fault monitor responds if the parent kernel process is not running. For more information, see [“Forcing the MaxDB Database Instance to Be Restarted if the Parent Kernel Process Is Terminated”](#) on page 35.
3. The MaxDB fault monitor determines whether SAP xserver is available. This fault monitoring supplements the fault monitoring that the SAP xserver fault monitor provides.

Operations by the SAP xserver Fault Monitor During a Probe

During a probe, the SAP xserver fault monitor determines whether SAP xserver is available.

Faults Detected by the Sun Cluster HA for MaxDB Fault Monitors

Faults that each Sun Cluster HA for MaxDB fault monitor detects are described in the subsections that follow.

Faults Detected by the MaxDB Fault Monitor

The MaxDB fault monitor detects the following faults in MaxDB:

- A status of the MaxDB database instance that is not ONLINE, for example, OFFLINE or ADMIN
- Unexpected termination of the parent kernel process of the MaxDB database instance

The MaxDB fault monitor also detects the unavailability of SAP xserver. This fault monitoring supplements the fault monitoring that the SAP xserver fault monitor provides.

Note – If the MaxDB fault monitor detects that SAP xserver is unavailable twice within the retry interval, the MaxDB fault monitor restarts MaxDB. By restarting MaxDB, the fault monitor ensures that the MaxDB database fails over to another node when SAP xserver is persistently unavailable.

Faults Detected by the SAP xserver Fault Monitor

The SAP xserver fault monitor detects following faults:

- **Unavailability of SAP xserver.** Unavailability of SAP xserver is also detected by the MaxDB fault monitor.
- **Persistent system errors.** A persistent system error is a system error that occurs four times within the retry interval. If a persistent system error occurs, the fault monitor restarts SAP xserver.

Recovery Actions in Response to Detected Faults

To minimize the disruption that transient faults in a resource cause, a fault monitor restarts the resource in response to such faults. For persistent faults, more disruptive action than restarting the resource is required:

- For the MaxDB resource, the fault monitor fails over the resource to another node. The MaxDB resource is a failover resource.
- For the SAP xserver resource, the fault monitor takes the resource offline. The SAP xserver is a scalable resource.

Forcing the MaxDB Database Instance to Be Restarted if the Parent Kernel Process Is Terminated

By default, unexpected termination of the parent kernel process does *not* cause the MaxDB fault monitor to restart the MaxDB database instance. The MaxDB database instance can continue to function without the parent kernel process. Restarting the MaxDB database instance in this situation might cause unnecessary unavailability of the MaxDB database instance. Therefore, you should force the MaxDB database instance to be restarted *only* if you require a feature that the parent kernel process provides. An example of such a feature is maintaining the integrity of the log history.

To force the MaxDB database instance to be restarted if the parent kernel process is terminated, set the `Restart_if_Parent_Terminated` extension property of the `SUNW.sapdb` resource to `True`.

Verifying the Sun Cluster HA for MaxDB Installation and Configuration

After you install, register, and configure Sun Cluster HA for MaxDB, verify the Sun Cluster HA for MaxDB installation and configuration. Verifying the Sun Cluster HA for MaxDB installation and configuration determines if the Sun Cluster HA for MaxDB data service makes the MaxDB application highly available.

Verifying the Sun Cluster HA for MaxDB installation involves verifying the operation of the following fault monitors:

- The MaxDB fault monitor
- The SAP xserver fault monitor

▼ How to Verify the Operation of the MaxDB Fault Monitor

Perform this procedure on each node where MaxDB can run.

1 Log in as superuser to a node that can master the MaxDB resource group.

2 Switch the MaxDB resource group to the node that you logged in to in [Step 1](#).

```
# clresourcegroup switch -n node sapdb-rg
```

node Specifies the node to which the MaxDB resource group is to be switched

sapdb-rg Specifies the name of the MaxDB resource group is to be switched to another node

3 Abnormally terminate MaxDB.

a. Determine the process identities of all kernel processes for the MaxDB database instance that you are running.

```
# ps -ef | grep kernel | grep db-name
```

db-name Specifies the name of the MaxDB database instance in uppercase

b. Kill all kernel processes for the MaxDB database instance that you are running.

```
# kill -9 sapdb-kernel-pid
```

sapdb-kernel-pid Specifies the process identities of the MaxDB kernel processes that you determined in [Step a](#)

4 Confirm that the Sun Cluster HA for MaxDB fault monitor performs the appropriate operation from the following list:

- Restarting the MaxDB resource
- Failing over the MaxDB resource to another node

The expected behavior of the fault monitor depends on the failure history of the resource and the value of the `Failover_enabled` extension property. For more information, see the following sections:

- [“Tuning the Sun Cluster HA for MaxDB Fault Monitors”](#) on page 32
- [“SUNW.sapdb Extension Properties”](#) on page 41

5 Terminate MaxDB normally.

a. Become the OS user who administers the MaxDB database.

```
# su - os-sapdb-adm-user
```

os-sapdb-adm-user Specifies the UNIX user identity of the OS user who administers the MaxDB database. This user's home directory contains the `.XUSER.62` file that was created during the installation and configuration of MaxDB. For more information, see [“Installing and Configuring MaxDB” on page 16](#). You specify this user when you set the `DB_User` extension property as explained in [“How to Register and Configure a MaxDB Resource” on page 29](#). If you are using MaxDB 7.5 or MaxDB 7.6, the UNIX user identity of the OS user who administers the MaxDB database is `sdb`.

b. Manually stop the MaxDB database instance.

```
# dbmcli -U sapdb-adm-key db_offline
```

-U sapdb-adm-key Specifies that the `dbmcli` command is run with the user `key` of the database user who administers the MaxDB database. This user `key` is created during the installation and configuration of MaxDB. For more information, see [“Installing and Configuring MaxDB” on page 16](#). You specify this user `key` when you set the `User_Key` extension property as explained in [“How to Register and Configure a MaxDB Resource” on page 29](#).

6 Confirm that the Sun Cluster HA for MaxDB fault monitor performs the appropriate operation from the following list:

- Restarting the MaxDB resource
- Failing over the MaxDB resource to another node

The expected behavior of the fault monitor depends on the failure history of the resource and the value of the `Failover_enabled` extension property. For more information, see the following sections:

- [“Tuning the Sun Cluster HA for MaxDB Fault Monitors” on page 32](#)
- [“SUNW.sapdb Extension Properties” on page 41](#)

▼ How to Verify the Operation of the SAP xserver Fault Monitor

Perform this procedure on each node that can master SAP xserver.

1 Log in to a node that can master SAP xserver.

2 Abnormally terminate SAP xserver.**a. Determine the process identities of all SAP xserver processes.**

```
# ps -ef | grep vserver
```

b. Kill all SAP xserver processes.

```
# kill -9 xserver-pid
```

xserver-pid Specifies the process identities of the SAP xserver processes that you determined in [Step a](#)

3 Confirm that the SAP xserver fault monitor restarts the SAP xserver resource.**4 Terminate SAP xserver normally.****a. Become the OS user who administers SAP xserver.**

```
# su - os-sapxsrvr-adm-user
```

```
os-sapxsrvr-adm-user
```

Specifies the UNIX user identity of the OS user who administers SAP xserver. By default, this user is root. You can specify this user by setting the Xserver_User extension property. For more information, see “[SUNW.sap_xserver Extension Properties](#)” on [page 44](#).

b. Manually stop the SAP xserver.

```
# x_server stop
```

5 Confirm that the SAP xserver fault monitor restarts the SAP xserver resource.

Upgrading the SUNW.sap_xserver Resource Type

The SUNW.sap_xserver resource type is supplied with the Sun Cluster HA for SAP liveCache data service. The Sun Cluster HA for SAP liveCache data service is installed when you install Sun Cluster HA for MaxDB data service. Upgrade the SUNW.sap_xserver resource type if all conditions in the following list apply:

- You are using version 3.0 5/02 asynchronous release, version 3.1 5/03, version 3.1 10/03, or 3.1 4/04 of the Sun Cluster HA for SAP liveCache data service.
- You plan to use the Sun Cluster HA for MaxDB data service with your existing version of the Sun Cluster HA for SAP liveCache data service.
- You need to use the new features of the SUNW.sap_xserver resource type.

For general instructions that explain how to upgrade a resource type, see “Upgrading a Resource Type” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*. The information that you need to complete the upgrade of the SUNW.sap_xserver resource type is provided in the subsections that follow.

Information for Registering the New Resource Type Version

The relationship between a resource type version and the release of Sun Cluster data services is shown in the following table. The release of Sun Cluster data services indicates the release in which the version of the resource type was introduced.

Resource Type Version	Sun Cluster Data Services Release
1.0	3.0 5/02 asynchronous release
2	3.1 4/04
3.1	3.2

To determine the version of the resource type that is registered, use the `clresource list` command.

The resource type registration (RTR) file for this resource type is `/opt/SUNWsc/c/xserver/etc/SUNW.sap_xserver`.

Information for Migrating Existing Instances of the Resource Type

The information that you need to migrate instances of the SUNW.sap_xserver resource type is as follows:

- You can perform the migration only when the resource is unmonitored.
- If you need to use the new features of the SUNW.sap_xserver resource type, the required value of the `Type_version` property is 2.
- If you need to specify the directory that contains programs and libraries for the SAP xserver runtime environment, set the `Independent_Program_Path` extension property. For more information, see [“SUNW.sap_xserver Extension Properties” on page 44](#).

The following example shows a command for editing an instance of the SUNW.sap_xserver resource type.

EXAMPLE 9 Editing an Instance of the SUNW.sap_xserver Resource Type During Upgrade

```
# clresource set -p Independent_Program_Path=/sapdb/indep_prog \  
-p Type_version=2 sapxserver-rs
```

This command edits a SUNW.sap_xserver resource as follows:

- The SUNW.sap_xserver resource is named sapxserver-rs.
- The Type_version property of this resource is set to 2.
- The independent program path is /sapdb/indep_prog.

Sun Cluster HA for MaxDB Extension Properties

Extension properties for Sun Cluster HA for MaxDB resource types are described in the following sections.

- “[SUNW.sapdb Extension Properties](#)” on page 41
- “[SUNW.sap_xserver Extension Properties](#)” on page 44

For details about system-defined properties, see the `r_properties(5)` man page and the `rg_properties(5)` man page.

SUNW.sapdb **Extension Properties**

The `SUNW.sapdb` resource type represents the MaxDB application in a Sun Cluster configuration. The extension properties of this resource type are as follows:

`dbmcli_Start_Option`

The option that is passed to the `dbmcli` command to start the MaxDB database instance.

Note – For MaxDB version 7.4.3, set this property to `db_online`.

Data type	String
Default	<code>db_online</code>
Range	Not applicable
Tunable	When disabled

`DB_Name`

The name of the MaxDB database instance in uppercase. This name is created when MaxDB is installed and configured as explained in “[Installing and Configuring MaxDB](#)” on page 16.

Data type String

Default No default defined

Range Not applicable

Tunable When disabled

DB_User

The UNIX user identity of the operating system (OS) user who administers the MaxDB database instance. This user's home directory contains the `.XUSER.62` file that was created during the installation and configuration of MaxDB. For more information, see [“Installing and Configuring MaxDB” on page 16](#).

Data type String

Default No default defined

Range Not applicable

Tunable When disabled

Failover_enabled

Specifies whether the fault monitor fails over the MaxDB resource if the number of attempts to restart exceeds `Retry_count` within the time that `Retry_interval` specifies. The possible values of this extension property are as follows:

- `True` – Specifies that the fault monitor fails over the MaxDB resource
- `False` – Specifies that the fault monitor does *not* fail over the MaxDB resource

Data type Boolean

Default `True`

Range Not applicable

Tunable Any time

Independent_Program_Path

The full path to the directory that contains the following programs and libraries for the MaxDB application:

- Programs that are independent of the database software version
- Libraries for the client runtime environment

Sun Cluster HA for MaxDB determines the path to the `dbmcli` command from the value of this property. The `dbmcli` command resides in the `bin` subdirectory of the directory that this property specifies.

Data type String

Default `/sapdb/programs`

Range Not applicable

Tunable When disabled

Monitor_retry_count

The maximum number of restarts by the process monitor facility (PMF) that are allowed for the fault monitor.

Data type	Integer
Default	4
Range	No range defined
Tunable	Any time

Monitor_retry_interval

The period of time in minutes during which the PMF counts restarts of the fault monitor.

Data type	Integer
Default	2
Range	No range defined
Tunable	Any time

Pid_Dir_Path

The full path to the directory under which files that store the process identities of MaxDB kernel processes are created. The process identities of MaxDB kernel processes are stored in the following files:

- *pid-dir/ppid/db-name*
- *pid-dir/pid/db-name*

The replaceable items in these file paths are as follows:

- *pid-dir* is the directory that the `Pid_Dir_Path` extension property specifies
- *db-name* is the name of the MaxDB database instance that the `DB_Name` extension property specifies

Data type	String
Default	<code>/var/spool/sql</code>
Range	Not applicable
Tunable	When disabled

Probe_timeout

The timeout value in seconds that the fault monitor uses to probe an MaxDB database instance.

Data type	Integer
Default	90
Range	30–99,999

Tunable Any time

Restart_if_Parent_Terminated

Determines whether the fault monitor restarts the MaxDB database instance if the parent kernel process is terminated. The possible values of this extension property are as follows:

- **True** – Specifies that the fault monitor restarts the MaxDB database instance if the parent kernel process is terminated
- **False** – Specifies that the fault monitor does *not* restart the MaxDB database instance if the parent kernel process is terminated

Data type Boolean

Default False

Range Not applicable

Tunable Any time

User_Key

The user key of the database user who administers the MaxDB database instance. This user key is created when MaxDB is installed and configured as explained in [“Installing and Configuring MaxDB” on page 16](#).

Data type String

Default No default defined

Range Not applicable

Tunable When disabled

SUNW.sap_xserver **Extension Properties**

The SUNW.sap_xserver resource type represents SAP xserver in a Sun Cluster configuration. The extension properties of this resource type are as follows:

Confdir_List

The full path to the directory that contains the MaxDB software and MaxDB database instance.

Data type String

Default /sapdb

Range Not applicable

Tunable At creation

Independent_Program_Path

The full path to the directory that contains the following programs and libraries for SAP xserver:

- Programs that are independent of the database software version
- Libraries for the client runtime environment

Sun Cluster HA for MaxDB determines the path to the `x_server` command from the value of this property. The `x_server` command resides in the `bin` subdirectory of the directory that this property specifies.

Data type	String
Default	No default defined
Range	Not applicable
Tunable	When disabled
Introduced in release	3.1 4/04

Monitor_retry_count

The maximum number of restarts by the PMF that are allowed for the fault monitor.

Data type	Integer
Default	4
Range	No range defined
Tunable	Any time

Monitor_retry_interval

The period of time in minutes during which the PMF counts restarts of the fault monitor.

Data type	Integer
Default	2
Range	No range defined
Tunable	Any time

Probe_timeout

The timeout value in seconds for fault monitor probes.

Data type	Integer
Default	120
Range	No range defined
Tunable	Any time

Soft_Stop_Pct

The percentage of the Stop method timeout that is used to stop SAP xserver by using the SAP utility `x_server stop`. If this timeout is exceeded, the SIGKILL signal is used to stop all SAP xserver processes.

Data type	Integer
Default	50
Range	1–100
Tunable	When disabled

Xserver_User

The UNIX user identity of the OS user who administers SAP xserver.

Data type	String
Default	root
Range	Not applicable
Tunable	At creation

Index

A

- Address already in use error message, 26
- affinities
 - resource groups
 - creating, 31
 - planning, 15
- affinity switchover, 24

C

- caution notice, multiple SAP xserver resources, 26
- clnode command, 8
- clusters, running MaxDB in, 18-20
- commands, node information, 8
- Confdir_List extension property, SUNW.sap_xserver
 - resource type, 44
- configuring
 - device groups
 - performing, 24-25
 - planning, 15
 - file systems
 - performing, 24-25
 - planning, 15
 - HASStoragePlus resource
 - performing, 24-25
 - planning, 15
 - MaxDB application, 16-20
 - Sun Cluster HA for MaxDB
 - overview, 10-11
 - performing, 25-32
 - planning, 12-16

creating

- HASStoragePlus resource, 24
- MaxDB resource, 29
- MaxDB resource group, 18
- SAP xserver resource, 28
- SAP xserver resource group, 27
- .XUSER.62 file, 17

D

- database instance, *See* MaxDB database instance
- database users
 - identity of OS user, 42
 - sample definition, 18
 - user key for, 44
- DB_Name extension property, 41
- DB_User extension property, 42
- dbmcli command
 - path to, 42
 - start option, 41
- dbmcli_Start_Option extension property, 41
- defining
 - MaxDB instance user, 17
 - SAP xserver user
 - planning for nondefault, 15
 - specifying nondefault, 26
- dependencies
 - device groups
 - creating, 29
 - planning, 15

dependencies (*Continued*)

- MaxDB resource
 - creating, 29
 - planning, 15

device groups

- configuring
 - performing, 24-25
 - planning, 15
- installation of MaxDB, 16
- MaxDB application, 15

directories

- See also* paths
- /usr/spool/sql, 17

E

editing, resource type instances, 39-40

enabling

- MaxDB resource group, 31
- SAP xserver resource group, 28

error messages, cluster, 8

extension properties

- See also* properties
- See also* system properties
- SUNW.sap_xserver resource type, 44-46
- SUNW.sapdb resource type, 41-44

F

Fai_lover_enabled extension property, SUNW.sapdb resource type, 42

fault monitors

- faults detected by, 34-35
- MaxDB
 - faults detected, 34
 - resource type for, 32
 - verifying operation of, 36-37
- probe interval, 33
- probe timeout, 33-34
- response to faults, 34-35
- SAP xserver
 - faults detected, 34
 - resource type for, 32

fault monitors, SAP xserver (*Continued*)

- verifying operation of, 37-38
- tuning, 32-35

faults

- recovery actions, 35
- responses to, 34-35

file systems

- configuration
 - performing, 24-25
 - planning, 15
- protection by data services, 10

files

- process identity, 43
- RTR, 39
- system configuration, 16

G

global zone, 9, 22

H

HASStoragePlus resource

- configuration
 - performing, 24-25
 - planning, 15
- dependency of MaxDB resource on
 - creating, 29
 - planning, 15
- MaxDB device group, 15
- resource group for, 24
- help, 8

I

Independent_Program_Path extension property

- SUNW.sap_xserver resource type, 44
- SUNW.sapdb resource type, 42

installing

- MaxDB application, 16-20
- Sun Cluster HA for MaxDB, 22-23
- overview, 10-11

installing, Sun Cluster HA for MaxDB (*Continued*)

verifying installation, 35-38

intervals, fault monitor probes, 33

K

kernel processes

identities, 43

termination of parent

extension property, 44

restarting MaxDB after, 35

L

libraries

MaxDB

SUNW.sap_xserver resource type, 44

SUNW.sapdb resource type, 42

SAP xserver, 45

local zones, *See* non-global zones

locations, *See* paths

log history, preserving, 35

logical host names

resource group for

creating, 18

enabling, 20

planning, 16

resources

creating, 19

planning, 16

M

MaxDB application

configuring, 16-20

device group for, 15

fault monitor, 32

installing, 16-20

paths to programs and libraries

SUNW.sap_xserver resource type, 44

SUNW.sapdb resource type, 42

MaxDB application (*Continued*)

processes

paths to process identity files, 43

termination of parent, 35, 44

protection by data services, 10

resource group for

creating, 18

enabling, 20

planning, 16

running in cluster, 18-20

software versions, 12

termination of parent kernel process, 35

verifying installation and configuration, 20-21

MaxDB database instance

database user, 44

name, 41

operating system user, 42

starting

dbmcli command options, 41

MaxDB verification, 21

stopping, 21

MaxDB resource

creating, 29

dependency on other resources

creating, 29

planning, 15

MaxDB resource group

affinity for SAP xserver resource group

creating, 31

planning, 15

maximum values

nodes for SAP xserver, 27

restarts

SUNW.sap_xserver resource type, 45

SUNW.sapdb resource type, 43

messages file, 8

migrating, resource type instances, 39-40

Monitor_retry_count extension property

SUNW.sap_xserver resource type, 45

SUNW.sapdb resource type, 42

Monitor_retry_interval extension property

SUNW.sap_xserver resource type, 45

SUNW.sapdb resource type, 43

N

names, MaxDB database instance, 41
network addresses, *See* logical host names
NFS file system, protection by data services, 10
nodes, for SAP xserver and MaxDB, 12
non-global zones, 9, 22

O

operating system users
 of MaxDB database instance
 defining, 17
 extension property for, 42
 sample definition, 18
 of SAP xserver
 defining, 15
 extension property for, 46
overriding, default SAP xserver users, 26

P

packages, 22-23
parent kernel processes
 extension property, 44
 restarting MaxDB after termination of, 35
paths
 dbmcli command, 42
 MaxDB programs and libraries
 SUNW.sap_xserver resource type, 44
 SUNW.sapdb resource type, 42
 process identity files, 43
 SAP xserver programs and libraries, 45
 x_server command, 45
persistent faults, recovery actions, 35
Pid_Dir_Path extension property, 43
Probe_timeout extension property
 SUNW.sap_xserver resource type, 45
 SUNW.sapdb resource type, 43
processes
 identities, 43
 termination of parent
 extension property, 44
 restarting MaxDB after, 35

programs

 MaxDB
 SUNW.sap_xserver resource type, 44
 SUNW.sapdb resource type, 42
 SAP xserver, 45
properties
 See also extension properties
 See also system properties
 Type_version, 39
protection, MaxDB application, 10
prtconf -v command, 8
prtdiag -v command, 8
psrinfo -v command, 8

R

recovery actions, after faults, 35
registering
 SUNW.HAStoragePlus resource type, 24-25
 SUNW.sap_xserver resource type
 during initial set up, 27-29
 during upgrade, 39
 SUNW.sapdb resource type, 29-32
resource groups
 HAStoragePlus resource, 24
 logical host name
 creating, 18
 enabling, 20
 logical host names
 planning, 16
 MaxDB
 creating, 18
 enabling, 20
 planning, 16
 SAP xserver
 creating, 27
 enabling, 28
 planning, 12
resource type registration (RTR) file, 39
resource types
 fault monitors, 32
 migrating instances of, 39-40
 SUNW.HAStoragePlus
 See also HAStoragePlus resource

- resource types, SUNW.HAStoragePlus (*Continued*)
 - registering, 24-25
 - SUNW.nfs, 10
 - SUNW.sap_xserver
 - extension properties, 44-46
 - initial registration, 27-29
 - instantiating, 28
 - protection by, 10
 - registration during upgrade, 39
 - relation to Sun Cluster HA for SAP liveCache, 10
 - upgrading, 38-40
 - SUNW.sapdb
 - extension properties, 41-44
 - initial registration, 29-32
 - instantiating, 29
 - protection by, 10
- resources
 - logical host name
 - creating, 19
 - planning, 16
 - MaxDB
 - creating, 29
 - SAP xserver
 - creating, 28
- responses, to faults, 34-35
- Restart_if_Parent_Terminated extension property
 - SUNW.sapdb resource type, 44
 - tuning, 35
- restarts
 - interval between
 - SUNW.sap_xserver resource type, 45
 - SUNW.sapdb resource type, 43
 - maximum allowed
 - SUNW.sap_xserver resource type, 45
 - SUNW.sapdb resource type, 43
- restrictions
 - zones, 9,22
- RTR (resource type registration) file, 39
- S**
- SAP liveCache application, use with Sun Cluster HA for MaxDB, 13-14
- SAP R/3 application, use with Sun Cluster HA for MaxDB, 13-14
- SAP xserver
 - administration by nondefault users
 - planning, 15
 - specifying, 26
 - caution notice, 26
 - fault monitor, 32
 - multiple instances of, 26
 - nodes for, 12
 - operating system user, 46
 - paths to programs and libraries, 45
 - protection by data services, 10
 - resource group for
 - creating, 27
 - enabling, 28
 - planning, 12
 - service configuration, 12
 - starting, 21
 - stopping
 - command for, 38
 - time allowed for, 46
 - upgrading, 38-40
- SAP xserver resource, 28
 - dependency of MaxDB resource on
 - creating, 29
 - planning, 15
- SAP xserver resource group
 - affinity of MaxDB resource group for
 - creating, 31
 - planning, 15
- show - rev subcommand, 8
- showrev -p command, 8
- SIGKILL signal, 46
- Soft_Stop_Pct extension property, 45
- software packages, 22-23
- starting
 - MaxDB database instance
 - dbmcli command options, 41
 - MaxDB verification, 21
 - SAP xserver, 21
- stopping
 - MaxDB database instance, 21

stopping (*Continued*)

- SAP xserver
 - command for, 38
 - time allowed for, 46
- Sun Cluster HA for MaxDB
 - overview, 10
 - configuration
 - performing, 25-28
 - planning, 12-16
 - with HASStoragePlus resource, 24-25
 - fault monitors, 32-35
 - installing, 22-23
 - verifying installation, 35-38
 - SAP liveCache application, use with, 13-14
 - SAP R/3 application, use with, 13-14
 - service configuration, 12
 - software packages
 - installing, 22-23
- Sun Cluster HA for SAP liveCache, relation to SUNW.sap_xserver resource type, 10
- SUNW.HASStoragePlus resource type
 - See also* HASStoragePlus resource
 - registering, 24-25
- SUNW.nfs resource type, 10
- SUNW.sap_xserver resource type
 - extension properties, 44-46
 - instantiating, 28
 - protection by, 10
 - registering
 - during initial setup, 27-29
 - during upgrade, 39
 - relation to Sun Cluster HA for SAP liveCache, 10
 - resource type versions, 39
 - upgrading, 38-40
- SUNW.sapdb resource type
 - extension properties, 41-44
 - instantiating, 29
 - protection by, 10
 - registering, 29-32
- system configuration files, location, 16
- system properties, effect on fault monitors, 32

T

- technical support, 8
- timeouts
 - fault monitor
 - guidelines for setting, 33-34
 - SUNW.sap_xserver resource type, 45
 - SUNW.sapdb resource type, 43
 - stop method, 46
- transient faults, recovery actions, 35
- tuning, fault monitors, 32-35
- Type_version property, 39

U

- upgrading, SAP xserver, 38-40
- User_Key extension property, 44
- users
 - of MaxDB database instance
 - database, 44
 - operating system, 42
 - sample definition, 18
 - of SAP xserver, 46
 - planning for nondefault, 15
 - specifying nondefault, 26
- /usr/spool/sql directory, 17

V

- /var/adm/messages file, 8
- verifying, MaxDB application, 20-21
- version requirements, MaxDB application, 12
- versions, resource types, 39

X

- x_server command, path to, 45
- Xserver_User extension property
 - specifying, 26
 - SUNW.sap_xserver resource type, 46
- .XUSER.62 file
 - creating, 17
 - and DB_User extension property, 42

.XUSER.62 file (*Continued*)
example, 18

Z

zones, 9, 22

