

**Fujitsu SPARC M12 and
Fujitsu M10/SPARC M10**
Domain Configuration Guide



Manual Code: C120-E680-20EN
January 2023

Copyright © 2007, 2023, Fujitsu Limited. All rights reserved.

Oracle and/or its affiliates provided technical input and review on portions of this material.

Oracle and/or its affiliates and Fujitsu Limited each own or control intellectual property rights relating to products and technology described in this document, and such products, technology and this document are protected by copyright laws, patents, and other intellectual property laws and international treaties.

This document and the product and technology to which it pertains are distributed under licenses restricting their use, copying, distribution, and decompilation. No part of such product or technology, or of this document, may be reproduced in any form by any means without prior written authorization of Oracle and/or its affiliates and Fujitsu Limited, and their applicable licensors, if any. The furnishings of this document to you does not give you any rights or licenses, express or implied, with respect to the product or technology to which it pertains, and this document does not contain or represent any commitment of any kind on the part of Oracle or Fujitsu Limited or any affiliate of either of them.

This document and the product and technology described in this document may incorporate third-party intellectual property copyrighted by and/or licensed from the suppliers to Oracle and/or its affiliates and Fujitsu Limited, including software and font technology.

Per the terms of the GPL or LGPL, a copy of the source code governed by the GPL or LGPL, as applicable, is available upon request by the End User. Please contact Oracle and/or its affiliates or Fujitsu Limited. 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 of The Open Group.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

Fujitsu and the Fujitsu logo are registered trademarks of Fujitsu Limited.

SPARC Enterprise, SPARC64, SPARC64 logo and all SPARC trademarks are trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries and used under license.

Other names may be trademarks of their respective owners.

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 END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Disclaimer: The only warranties granted by Oracle and Fujitsu Limited, and/or any affiliate in connection with this document or any product or technology described herein are those expressly set forth in the license agreement pursuant to which the product or technology is provided.

EXCEPT AS EXPRESSLY SET FORTH IN SUCH AGREEMENT, ORACLE OR FUJITSU LIMITED, AND/OR THEIR AFFILIATES MAKE NO REPRESENTATIONS OR WARRANTIES OF ANY KIND (EXPRESS OR IMPLIED) REGARDING SUCH PRODUCT OR TECHNOLOGY OR THIS DOCUMENT, WHICH ARE ALL PROVIDED AS IS, AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING WITHOUT LIMITATION 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. Unless otherwise expressly set forth in such agreement, to the extent allowed by applicable law, in no event shall Oracle or Fujitsu Limited, and/or any of their affiliates have any liability to any third party under any legal theory for any loss of revenues or profits, loss of use or data, or business interruptions, or for any indirect, special, incidental or consequential damages, even if advised of the possibility of such damages.

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, 2023, Fujitsu Limited. Tous droits réservés.

Oracle et/ou ses affiliés ont fourni et vérifié des données techniques de certaines parties de ce composant.

Oracle et/ou ses affiliés et Fujitsu Limited détiennent et contrôlent chacun des droits de propriété intellectuelle relatifs aux produits et technologies décrits dans ce document. De même, ces produits, technologies et ce document sont protégés par des lois sur le droit d'auteur, des brevets, et d'autres lois sur la propriété intellectuelle et des traités internationaux. Ce document, le produit et les technologies afférents sont exclusivement distribués avec des licences qui en restreignent l'utilisation, la copie, la distribution et la décompilation.

Aucune partie de ce produit, de ces technologies ou de ce document ne peut être reproduite sous quelque forme que ce soit, par quelque moyen que ce soit, sans l'autorisation écrite préalable d'Oracle et/ou ses affiliés et de Fujitsu Limited, et de leurs éventuels concédants de licence. Ce document, bien qu'il vous ait été fourni, ne vous confère aucun droit et aucune licence, expresse ou tacite, concernant le produit ou la technologie auxquels il se rapporte. Par ailleurs, il ne contient ni ne représente aucun engagement, de quelque type que ce soit, de la part d'Oracle ou de Fujitsu Limited, ou des sociétés affiliées de l'une ou l'autre entité.

Ce document, ainsi que les produits et technologies qu'il décrit, peuvent inclure des droits de propriété intellectuelle de parties tierces protégés par le droit d'auteur et/ou cédés sous licence par des fournisseurs à Oracle et/ou ses sociétés affiliées et Fujitsu Limited, y compris des logiciels et des technologies relatives aux polices de caractères.

Conformément aux conditions de la licence GPL ou LGPL, une copie du code source régit par la licence GPL ou LGPL, selon le cas, est disponible sur demande par l'Utilisateur Final.

Veuillez contacter Oracle et/ou ses affiliés ou Fujitsu Limited. Cette distribution peut comprendre des composants développés par des parties tierces. Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie.

UNIX est une marque déposée de The OpenGroup.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés.

Fujitsu et le logo Fujitsu sont des marques déposées de Fujitsu Limited.

SPARC Enterprise, SPARC64, le logo SPARC64 et toutes les marques SPARC sont utilisées sous licence et sont des marques déposées de SPARC International, Inc., aux Etats-Unis et dans d'autres pays.

Tout autre nom mentionné peut correspondre à des marques appartenant à leurs propriétaires respectifs.

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 END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Avis de non-responsabilité : les seules garanties octroyées par Oracle et Fujitsu Limited et/ou toute société affiliée de l'une ou l'autre entité en rapport avec ce document ou tout produit ou toute technologie décrits dans les présentes correspondent aux garanties expressément stipulées dans le contrat de licence régissant le produit ou la technologie fournis.

SAUF MENTION CONTRAIRE EXPRESSEMENT STIPULEE AU DIT CONTRAT, ORACLE OU FUJITSU LIMITED ET/OU LES SOCIETES AFFILIEES A L'UNE OU L'AUTRE ENTITE DECLINENT TOUT ENGAGEMENT OU GARANTIE, QUELLE QU'EN SOIT LA NATURE (EXPRESSE OU IMPLICITE) CONCERNANT CE PRODUIT, CETTE TECHNOLOGIE OU CE DOCUMENT, LESQUELS SONT FOURNIS EN L'ETAT. EN OUTRE, TOUTES LES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON, SONT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE. Sauf mention contraire expressément stipulée dans ce contrat, dans la mesure autorisée par la loi applicable, en aucun cas Oracle ou Fujitsu Limited et/ou l'une ou l'autre de leurs sociétés affiliées ne sauraient être tenues responsables envers une quelconque partie tierce, sous quelque théorie juridique que ce soit, de tout manque à gagner ou de perte de profit, de problèmes d'utilisation ou de perte de données, ou d'interruptions d'activités, ou de tout dommage indirect, spécial, secondaire ou consécutif, même si ces entités ont été préalablement informées d'une telle éventualité.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTE AUTRE CONDITION, DECLARATION ET GARANTIE, EXPRESSE OU TACITE, EST FORMELLEMENT EXCLUE, DANS LA MESURE AUTORISEE PAR LA LOI EN VIGUEUR, 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 xi

Chapter 1 Understanding an Overview of Domain Configuration 1

- 1.1 SPARC M12/M10 Domain Configurations 1
 - 1.1.1 Overview of the One-Unit Configuration and Building Block Configuration 1
 - 1.1.2 Overview of the Logical Domain Configuration 4
 - 1.1.3 Features of the Domain Configuration 6
- 1.2 SPARC M12/M10 Hardware Resources 10
- 1.3 What is a Physical Partition? 13
 - 1.3.1 Understanding Physical Partition Components 13
 - 1.3.2 Working With a Physical Partition Configuration 15
- 1.4 What is a Logical Domain? 18
 - 1.4.1 Understanding Logical Domain Components 18
 - 1.4.2 Working With a Logical Domain Configuration 20
 - 1.4.3 Reconfiguration of a Physical I/O Device 22
- 1.5 What is Physical Partition Dynamic Reconfiguration? 23
 - 1.5.1 Overview of Physical Partition Dynamic Reconfiguration 23
 - 1.5.2 Using Physical Partition Dynamic Reconfiguration 23
 - 1.5.3 Dynamic Reconfiguration of CPUs and Memory 24
 - 1.5.4 Software Conditions and Confirmation Methods 25

1.6	Mixed Use of the SPARC64 X+ Processor and SPARC64 X Processor	27
1.6.1	CPU Operation Type and CPU Operational Mode	28
Chapter 2	Conditions and Settings for System Operation	31
2.1	XSCF Status Management	31
2.1.1	Physical Partition Status	31
2.1.2	System Board Status	32
2.1.3	Status Change	34
2.2	XSCF Conditions and Settings	38
2.2.1	Configuration Conditions Required for the XSCF	38
2.2.2	Settings From the XSCF	38
2.2.3	Memory Mirror Mode Considerations	40
2.2.4	Considerations When Adding a CPU Module	40
2.3	Oracle Solaris Conditions and Settings	41
2.3.1	I/O and Software Requirements	41
2.3.2	Swap Space-related Considerations	41
2.3.3	Real-Time Process Considerations	42
2.4	Logical Domain Conditions and Settings	42
2.4.1	Consideration of Logical Domain Configuration	43
2.4.2	Consideration of Logical Domain Reconfiguration	55
2.5	Dynamic Reconfiguration Conditions and Settings	56
2.5.1	Considerations When Configuring the System for Dynamic Reconfiguration	56
2.5.2	Considerations in System Operation for Dynamic Reconfiguration	58
2.5.3	How to Check/Set PPAR DR Mode	68
2.6	Considerations When Using the SPARC64 X+ Processor	71
2.6.1	Updating the XCP firmware	71
2.6.2	CPU Operational Mode Settings	71
2.6.3	Conditions for a Mixed Configuration With the SPARC64 X+ Processor and the SPARC64 X Processor Within a PPAR	75

2.6.4	Relationship Between the CPU Operational Mode and Physical Partition Dynamic Reconfiguration	76
-------	--	----

Chapter 3 Operations for Domain Configuration 79

3.1	Operations and Commands Related to Physical Partition Configurations	79
3.1.1	Checking Physical Partition Configuration Information	80
3.1.2	Checking the Physical Partition Status	83
3.1.3	Checking the System Board Status	84
3.1.4	Checking Device Setting Information	87
3.1.5	Checking the Logical Domain Status	88
3.1.6	Checking the CPU Activation Key Information	90
3.1.7	Checking the Usage Status of CPU Core Resources	90
3.1.8	Checking CPU Activation Registration and Setting Information	92
3.1.9	Checking the Physical Partition Operation Mode	93
3.1.10	Displaying the Logical Domain Configuration Information for a Physical Partition	94
3.1.11	Setting Memory Mirroring	95
3.1.12	Setting Physical Partition Configuration Information	97
3.1.13	Adding a CPU Activation Key	98
3.1.14	Assigning CPU Core Resources Used in a Physical Partition	99
3.1.15	Adding a System Board to a Building Block Configuration	102
3.1.16	Deleting a System Board From a Building Block Configuration	104
3.1.17	Setting the Physical Partition Operation Mode	106
3.1.18	Specifying the Logical Domain Configuration Information for a Physical Partition	108
3.1.19	Starting a Physical Partition	109
3.1.20	Stopping a Physical Partition	109
3.1.21	Connecting to the Control Domain Console	109

3.2	Operations and Commands Related to Logical Domain Configurations	110
3.2.1	Confirming That Logical Domains Manager is Running	112
3.2.2	Checking Services	112
3.2.3	Checking the Number of Virtual CPUs That Can be Assigned Based on CPU Activations	113
3.2.4	Checking the Assignment Status of Resources	113
3.2.5	Checking the Usage Status of Resources	113
3.2.6	Checking the Usage Status of Each Resource Group	115
3.2.7	Checking the Resource Usage Status of Each CPU Socket	116
3.2.8	Checking the Logical Domain Status	117
3.2.9	Displaying the Logical Domain Configuration Information	118
3.2.10	Checking the Usage Status of I/O Devices	118
3.2.11	Starting Delayed Reconfiguration Mode	120
3.2.12	Setting the Default Services	121
3.2.13	Configuring Virtual CPUs	123
3.2.14	Configuring Virtual Memory	127
3.2.15	Setting CPU Socket Restrictions	130
3.2.16	Setting Logical Domain Configuration Information	131
3.2.17	Creating a Logical Domain	132
3.2.18	Configuring an I/O Device	132
3.2.19	Creating or Destroying the SR-IOV Virtual Function	134
3.2.20	Configuring a Virtual Network Device	138
3.2.21	Configuring a Virtual Disk Server	139
3.2.22	Configuring a Virtual Disk	140
3.2.23	Configuring a Virtual Console	141
3.2.24	Configuring a Startup Device	142
3.2.25	Binding Resources	142
3.2.26	Starting a Guest Domain	142
3.2.27	Specifying a Shutdown Group	143
3.2.28	Reconfiguring a Device	143

3.2.29	Enabling Recovery Mode	143
Chapter 4	Configuring a Physical Partition	145
4.1	Physical Partition Configuration Flow	145
4.2	Configuring and Operating a Physical Partition	146
Chapter 5	Logical Domain Configuration Example	155
5.1	Logical Domain Configuration Flow	155
5.2	Example of Logical Domain Configuration Operations	157
5.2.1	Logging In to the Control Domain	158
5.2.2	Setting the Default Services	159
5.2.3	Making Initial Settings for the Control Domain	159
5.2.4	Configuring a Guest Domain	161
5.2.5	Configuring Recovery Mode	162
5.2.6	Saving Logical Domain Configuration Information	163
Chapter 6	Physical Partition Reconfiguration	165
6.1	Flow for Physical Partition Reconfiguration	165
6.1.1	Flow for System Board Addition	165
6.1.2	Flow for System Board Deletion	167
6.1.3	Flow for System Board Move	168
6.1.4	Flow for System Board Replacement	170
6.2	Example of System Board Addition	171
6.2.1	Example of System Board Assignment	172
6.2.2	Example of System Board Incorporation	175
6.2.3	Example of Operations for Reserving the Incorporation of a System Board	178
6.3	Example of Operations for Deleting a System Board	182
6.3.1	Example of Operations for Deleting a System Board Assignment	182
6.3.2	Example of Operations for Releasing a System Board	183
6.3.3	Example of Operations for Reserving a System Board Unassignment	188
6.4	Example of Operations for Moving a System Board	189

6.5	Example of Operations for Replacing a System Board	196
6.6	Considerations When Changing a Physical Partition Number	201
Chapter 7	Migrating a Guest Domain	203
7.1	Overview	203
7.1.1	Requirements for Live Migration	205
7.2	Migrating a Guest Domain	206
7.3	Guest Domain Migration Example	206
Appendix A	Environment Configuration Examples and Procedures Using Physical Partition Dynamic Reconfiguration	209
A.1	Updating a System That Does Not Support Physical Partition Dynamic Reconfiguration to One That Does Support It	210
A.1.1	Configuration Example	210
A.1.2	Configuration Procedure	211
A.2	For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has No Free Space for Hardware Resources	231
A.2.1	Configuration Example	232
A.2.2	Example of the Physical Partition Configuration Procedure	234
A.2.3	Example of the Active Replacement Procedure (for Oracle VM Server for SPARC 3.1.x)	259
A.2.4	Example of the Active Replacement Procedure (When Dynamic Assignment of the PCIe Bus is Available)	283
A.3	For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has Free Hardware Resources	307
A.3.1	Configuration Example	308
A.3.2	Example of the Physical Partition Configuration Procedure	310
A.3.3	Example of the Active Replacement Procedure (for Oracle VM Server for SPARC 3.1.x)	338
A.3.4	Example of the Active Replacement Procedure (When Dynamic Assignment of the PCIe Bus is Available)	354

A.4	For Expansion of a Newly Installed System of Version XCP 2220 or Later from the 1BB to 2BB Configuration	369
A.4.1	Configuration Example	369
A.4.2	Procedure for Expansion (for Oracle VM Server for SPARC 3.1.x)	372
A.4.3	Procedure for Expansion (When Dynamic Assignment of the PCIe Bus is Available)	390
A.5	For Installation of the System Board Configured by the SPARC64 X+ Processor to the Physical Partition Configured Only by the SPARC64 X Processor	408
A.5.1	Configuration Example	408
A.5.2	Configuration Procedure	411
A.6	For Active Replacement of a System Board in a Configuration with Only the Control Domain (for Oracle VM Server for SPARC 3.2 or Later)	446
A.6.1	Configuration Example	447
A.6.2	Example of the Physical Partition Configuration Procedure	448
A.6.3	Example of the Active Replacement Procedure	457
Appendix B	Supplementary Information for Using Dynamic Reconfiguration	465
B.1	Considerations for the XSCF Reboot or Failover Time	465
	Checking the completion of addition	465
	Checking the completion of removal	466
B.2	Supplementary Information on CPU Operational Mode	467
B.3	Other Considerations	467
Appendix C	Meanings of Messages and Corresponding Corrective Action	471
C.1	Command Messages	471
C.1.1	addboard	471
C.1.2	deleteboard	475
Index		481

Preface

This document describes the domain configuration function provided by the SPARC M12/M10 systems from Oracle or Fujitsu. The document is intended for system administrators with advanced knowledge of computer networks and Oracle Solaris.

Fujitsu SPARC M12 is sold as SPARC M12 by Fujitsu in Japan.
Fujitsu SPARC M12 and SPARC M12 are identical products.
Fujitsu M10 is sold as SPARC M10 by Fujitsu in Japan.
Fujitsu M10 and SPARC M10 are identical products.

Audience

This document is intended for system administrators with advanced knowledge of computer networks and Oracle Solaris.

Related Documentation

All documents for your server are available online at the following locations.

- Sun Oracle software-related documents (Oracle Solaris, etc.)
<https://docs.oracle.com/en/>
- Fujitsu documents
Global site
<https://www.fujitsu.com/global/products/computing/servers/unix/sparc/downloads/manuals/>
Japanese site
<https://www.fujitsu.com/jp/products/computing/servers/unix/sparc/downloads/manual/>

For a system using the SPARC M12, see the manuals listed in "[Documentation Related to the SPARC M12.](#)"

For a system using the SPARC M10, see the manuals listed in "[Documentation Related to the SPARC M10.](#)"

Documentation Related to the SPARC M12

Manual Names (*1)

Fujitsu SPARC M12 Product Notes

Fujitsu SPARC M12 Quick Guide

*Fujitsu SPARC M12 Getting Started Guide (*2)*

*Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Important Legal and Safety Information (*2)*

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Safety and Compliance Guide

Software License Conditions for Fujitsu SPARC M12 and Fujitsu M10/SPARC M10

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Security Guide

Fujitsu SPARC Servers/SPARC Enterprise/PRIMEQUEST Common Installation Planning Manual

Fujitsu SPARC M12-1 Installation Guide

Fujitsu SPARC M12-2 Installation Guide

Fujitsu SPARC M12-2S Installation Guide

Fujitsu SPARC M12 PCI Card Installation Guide

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Domain Configuration Guide

*Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 RCIL User Guide (*3)*

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF MIB and Trap Lists

Fujitsu SPARC M12-1 Service Manual

Fujitsu SPARC M12-2/M12-2S Service Manual

Crossbar Box for Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Service Manual

PCI Expansion Unit for Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Service Manual

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Glossary

External USB-DVD Drive user guide

*1 The listed manuals are subject to change without notice.

*2 Printed manuals are provided with the product.

*3 This document applies specifically to the SPARC M12/M10 and FUJITSU ETERNUS disk storage system.

Manual Names (*1)

Fujitsu M10/SPARC M10 Systems Product Notes

Fujitsu M10/SPARC M10 Systems Quick Guide

*Fujitsu M10/SPARC M10 Systems Getting Started Guide (*2)*

*Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Important Legal and Safety Information (*2)*

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Safety and Compliance Guide

Software License Conditions for Fujitsu SPARC M12 and Fujitsu M10/SPARC M10

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Security Guide

Fujitsu SPARC Servers/SPARC Enterprise/PRIMEQUEST Common Installation Planning Manual

Fujitsu M10-1/SPARC M10-1 Installation Guide

Fujitsu M10-4/SPARC M10-4 Installation Guide

Fujitsu M10-4S/SPARC M10-4S Installation Guide

Fujitsu M10/SPARC M10 Systems PCI Card Installation Guide

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Domain Configuration Guide

*Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 RCIL User Guide (*3)*

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF MIB and Trap Lists

Fujitsu M10-1/SPARC M10-1 Service Manual

Fujitsu M10-4/Fujitsu M10-4S/SPARC M10-4/SPARC M10-4S Service Manual

Crossbar Box for Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Service Manual

PCI Expansion Unit for Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Service Manual

Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Glossary

External USB-DVD Drive user guide

*1 The listed manuals are subject to change without notice.

*2 Printed manuals are provided with the product.

*3 This document applies specifically to the SPARC M12/M10 and FUJITSU ETERNUS disk storage system.

Notes on Safety

Read the following documents thoroughly before using or handling the SPARC M12/M10.

- *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Important Legal and Safety Information*

Text Conventions

This manual uses the following fonts and symbols to express specific types of information.

Font/Symbol	Meaning	Example
AaBbCc123	What you type, when contrasted with on-screen computer output. This font is used to indicate an example of command input.	XSCF> adduser jsmith
AaBbCc123	The names of commands, files, and directories; on-screen computer output. This font is used to indicate an example of command output in the frame.	XSCF> showuser -P User Name: jsmith Privileges: useradm auditadm
<i>Italic</i>	Indicates the name of a reference manual.	See the <i>Fujitsu M10-1/SPARC M10-1 Installation Guide</i> .
" "	Indicates the names of chapters, sections, items, buttons, or menus.	See "Chapter 2 Network Connection."

Command Syntax in the Text

While the XSCF commands have a section number of (8) or (1), it is omitted from the text.

For details on the commands, see the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual*.

Syntax of the Command-Line Interface (CLI)

The command syntax is as follows:

- A variable that requires the input of a value is in *Italics*.
- An optional element is enclosed in [].
- A group of options for an optional keyword is enclosed in [] and delimited by |.

Document Feedback

If you have any comments or requests regarding this document, please take a moment to share them with us. Along with the manual code, manual title, and page number, state your points specifically at one of the following websites:

- Global site
<https://www.fujitsu.com/global/contact/>
- Japanese site
<https://www.fujitsu.com/jp/products/computing/servers/unix/sparc/contact/>

Chapter 1

Understanding an Overview of Domain Configuration

This chapter provides an overview of domain configuration.

- [SPARC M12/M10 Domain Configurations](#)
- [SPARC M12/M10 Hardware Resources](#)
- [What is a Physical Partition?](#)
- [What is a Logical Domain?](#)
- [What is Physical Partition Dynamic Reconfiguration?](#)
- [Mixed Use of the SPARC64 X+ Processor and SPARC64 X Processor](#)

1.1 SPARC M12/M10 Domain Configurations

This section describes an overview and features of the domain configurations that can be built with SPARC M12/M10.

1.1.1 Overview of the One-Unit Configuration and Building Block Configuration

This section provides an overview of a system that consists of one SPARC M12/M10 unit and a system that consists of multiple SPARC M12 or SPARC M10 units (a system with a building block configuration).

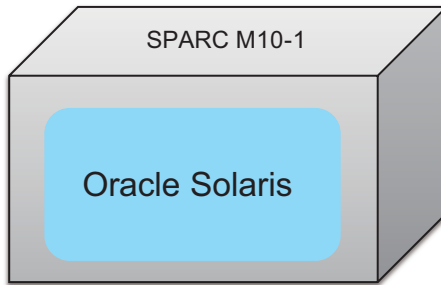
- System with the one-unit configuration
- System with the building block configuration (SPARC M12-2S/M10-4S)

System with the one-unit configuration

[Figure 1-1](#) shows an example of a system configuration with SPARC M10-1. In this

example, the system has one Oracle Solaris running on it.

Figure 1-1 System with the One-Unit Configuration (Example: SPARC M10-1)



System with the building block configuration

With SPARC M12-2S or SPARC M10-4S, a system with the building block (BB) configuration can connect multiple SPARC M12-2S units or SPARC M10-4S units. The system can be built as an independent system or divided into multiple physical partitions in units of BBs.

For the system with the building block configuration, multiple BBs can comprise a single physical partition.

Note - For the building block method, SPARC M12-2S and SPARC M10-4S cannot be mixed to comprise a system.

Figure 1-2 shows an example of a system configuration with multiple SPARC M12-2S units. In this example, the system has multiple SPARC M12-2S units running one Oracle Solaris.

Figure 1-2 System with the Building Block Configuration (Not Divided into Physical Partitions)

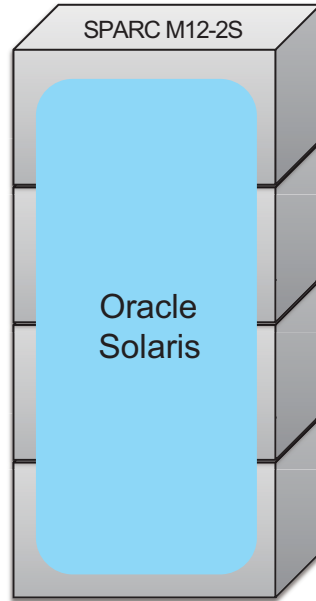
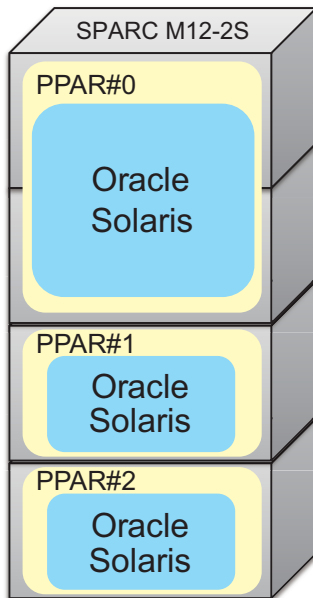


Figure 1-3 shows an example in which four SPARC M12-2S units are connected with the system divided into three physical partitions. In this example, the system is divided into multiple physical partitions, each running one Oracle Solaris.

Figure 1-3 System with the Building Block Configuration (Divided into Physical Partitions)



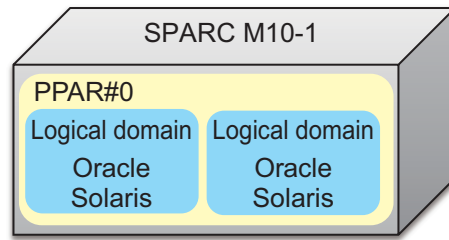
1.1.2 Overview of the Logical Domain Configuration

Configuration with multiple logical domains

Multiple logical domains can be configured in a single physical partition.

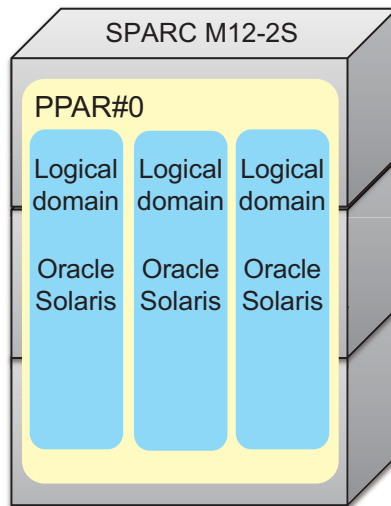
Configuring multiple logical domains in a single physical partition makes it possible to run one Oracle Solaris on each logical domain. Figure 1-4 shows an example of a system configuration with SPARC M10-1. This example shows a system configuration with two logical domains configured in a single physical partition.

Figure 1-4 Configuration with Multiple Logical Domains (One-Unit Configuration)



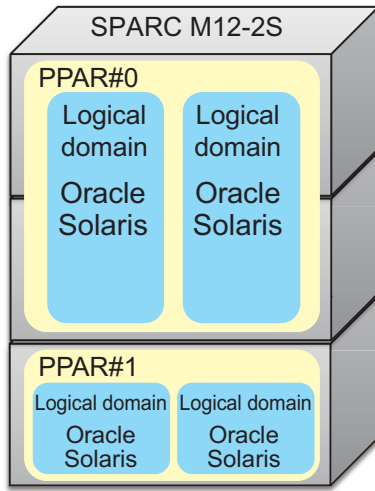
Configuring multiple logical domains in a system with the building block configuration makes it possible to run Oracle Solaris on each logical domain. [Figure 1-5](#) shows an example of a system with the building block configuration in which three SPARC M12-2S units are connected and three logical domains are configured in one physical partition.

Figure 1-5 Configuration with Multiple Logical Domains (Building Block Configuration - 1)



A system with the building block configuration can be divided into multiple physical partitions, with multiple logical domains configured on each physical partition. Furthermore, Oracle Solaris can be run on each logical domain. [Figure 1-6](#) shows an example of a system with the building block configuration in which three SPARC M12-2S units are connected and two logical domains are configured in each of the two physical partitions.

Figure 1-6 Configuration with Multiple Logical Domains (Building Block Configuration - 2)



1.1.3 Features of the Domain Configuration

This section describes the domain configuration, a feature of the SPARC M12/M10 system.

Features of a system consisting of logical domains

The SPARC M12/M10 system allows multiple logical domains to be configured using Oracle VM Server for SPARC and an independent OS to run on each logical domain. Different logical domains can run different versions of the OS, respectively. CPUs, memory, and I/Os can be assigned flexibly to each logical domain, enabling efficient use of hardware resources.

Figure 1-7 shows an example in which two CPU cores and a part of the memory resource are moved between Oracle Solaris instances of different versions.

Figure 1-7 Efficient Use of Hardware Resources between Logical Domains (Example of One-Unit Configuration Using SPARC M10-1)

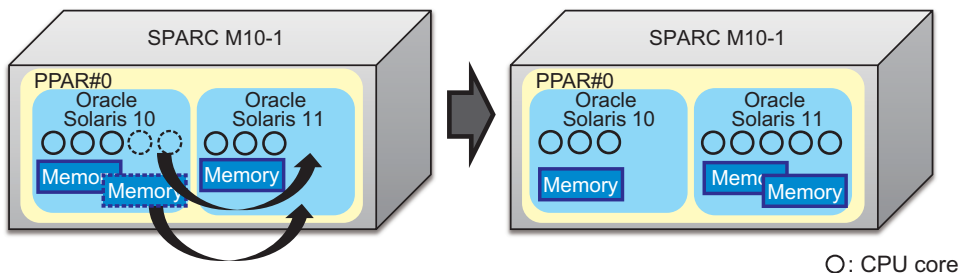
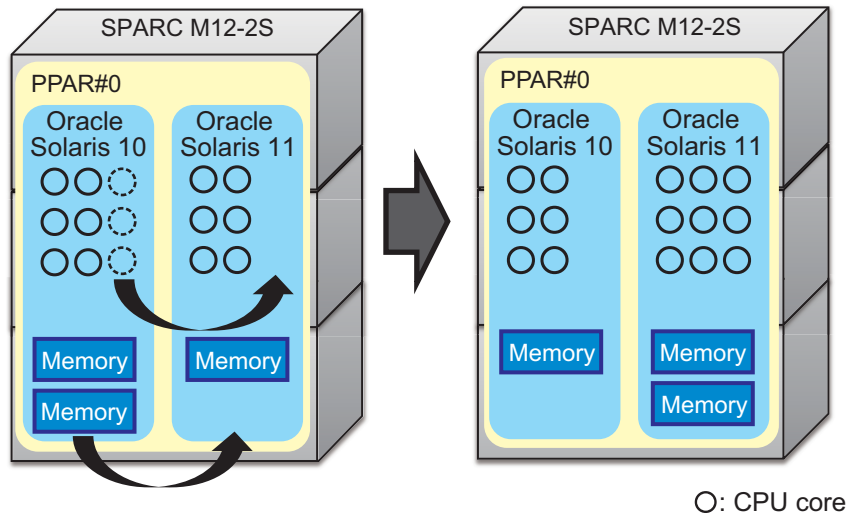


Figure 1-8 shows an example in which three CPU cores and one memory resource are moved between Oracle Solaris instances of different versions.

Figure 1-8 Efficient Use of Hardware Resources between Logical Domains (Building Block Configuration)

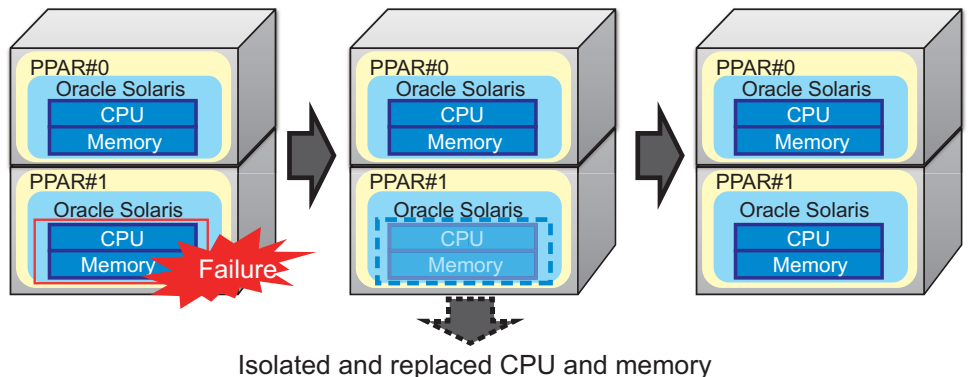


Features of a system consisting of physical partitions

The SPARC M12-2S/M10-4S system allows physical partitions to be configured with one or more chassis and an independent OS to run on each physical partition. A system consisting of physical partitions provides a higher level of hardware independence than a system that consists of logical domains using Oracle VM Server for SPARC.

Figure 1-9 shows an example in which chassis are isolated from the system to replace the CPU and memory.

Figure 1-9 Feature of Physical Partitions (Hardware Independence)

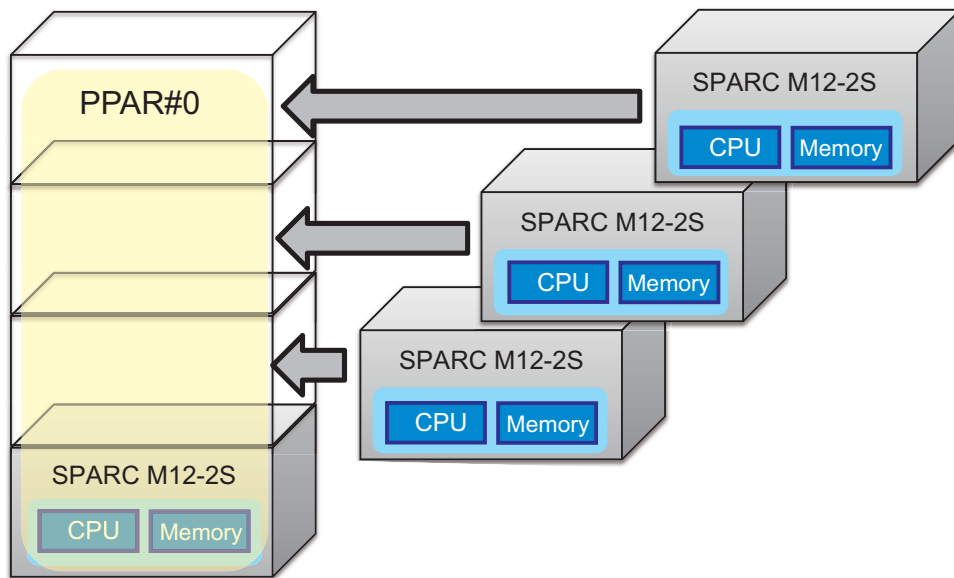


The SPARC M12-2S/M10-4S system allows hardware resources to be expanded by

stacking multiple chassis. This enables a small system to be expanded gradually as necessary for the environment.

Figure 1-10 shows an example in which three SPARC M12-2S units are added and the four units are used as a single physical partition.

Figure 1-10 Feature of Physical Partitions (Hardware Scalability)

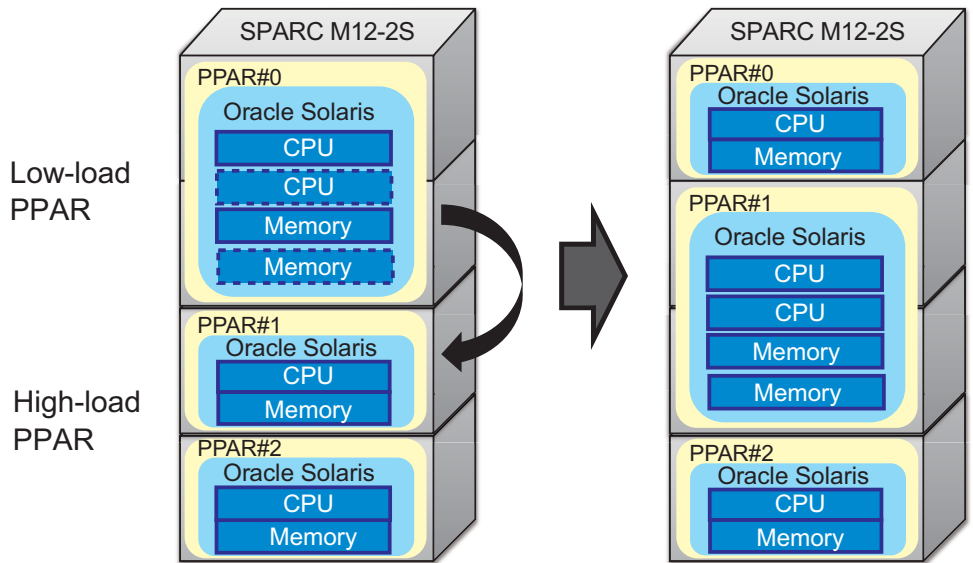


Also, CPUs, memory, and I/Os can be added to and removed from a physical partition in units of building blocks (PSBs).

Moreover, a system consisting of multiple physical partitions allows these physical partitions to be managed in a centralized manner, and hardware resources can be moved from a physical partition with a low load to one with a high load.

Figure 1-11 shows an example in which one chassis's worth of resources is moved from PPAR#0 to PPAR#1.

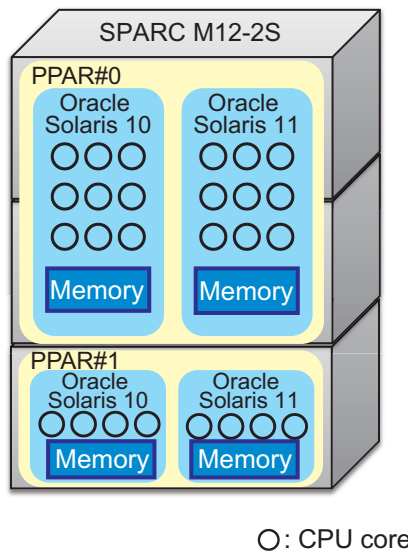
Figure 1-11 Feature of Physical Partitions (Hardware Flexibility)



In addition, logical domains can be configured in each physical partition.

Figure 1-12 shows an example in which three SPARC M12-2S units are connected and two logical domains are configured in each of the two physical partitions.

Figure 1-12 Example of Configuring Logical Domains in Physical Partitions



1.2 SPARC M12/M10 Hardware Resources

This section describes the hardware resources of the SPARC M12/M10 system.

The SPARC M12/M10 system allows hardware resources, such as CPUs, memory, and I/Os, to be assigned flexibly, enabling efficient use of the resources.

Figure 1-13 shows an example of efficient use of the SPARC M12-1/M10-1 hardware resources.

Figure 1-13 Efficient Use of Hardware Resources (SPARC M12-1/M10-1)

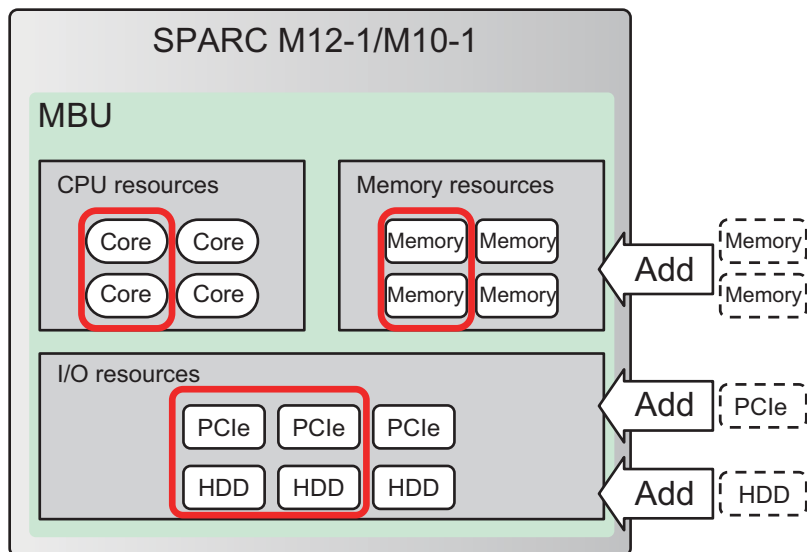
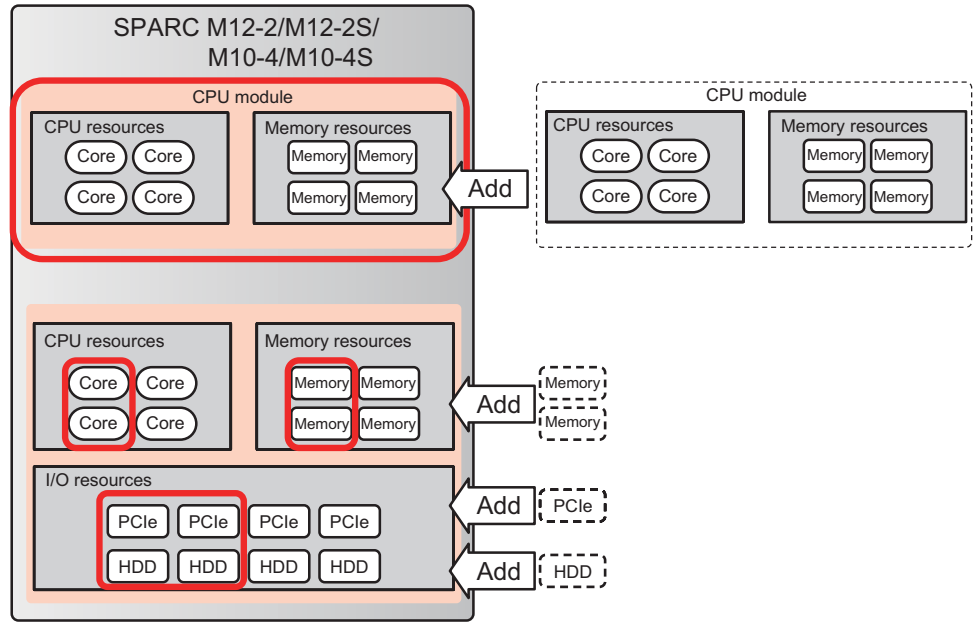


Figure 1-14 shows an example of efficient use of the hardware resources of SPARC M12-2/M12-2S/M10-4/M10-4S.

Figure 1-14 Efficient Use of Hardware Resources (SPARC M12-2/M12-2S/M10-4/M10-4S)



■ CPU resources

[SPARC M12-1]

The resources made available by CPU Activation are a maximum of 6 CPU cores.

[SPARC M12-2/M12-2S]

The resources made available by CPU Activation are a maximum of 24 CPU cores (in a 2-CPU configuration).

[SPARC M10-1]

The resources made available by CPU Activation are a maximum of 16 CPU cores.

[SPARC M10-4/M10-4S]

The resources made available by CPU Activation are a maximum of 32 CPU cores (in a 4-CPU configuration).

■ Memory resources

Memory mirroring can be configured for all models. In that case, the amount of available memory is halved.

[SPARC M12-1]

Up to 1 TB can be mounted.

[SPARC M12-2/M12-2S]

Up to 2 TB can be mounted (in a 2-CPU configuration).

[SPARC M10-1]

Up to 1 TB can be mounted.

[SPARC M10-4/M10-4S]

Up to 4 TB can be mounted (in a 4-CPU configuration).

■ I/O resources

[SPARC M12-1]

Up to three PCIe cards can be mounted. By connecting a PCI expansion unit, up to 33 PCIe cards can be mounted. Up to eight internal storage units can be mounted.

[SPARC M12-2/M10-4]

The SPARC M12-2/M10-4 allows up to 11 PCIe cards to be mounted. By connecting a PCI expansion unit, up to 71 PCIe cards can be mounted. Up to eight internal storage units can be mounted.

The number of PCIe cards can be increased dynamically using the PHP.

[SPARC M12-2S/M10-4S]

The SPARC M12-2S/M10-4S allows up to eight PCIe cards to be mounted. By connecting a PCI expansion unit, up to 58 PCIe cards can be mounted. Up to eight internal storage units can be mounted.

The number of PCIe cards can be increased dynamically using the PHP.

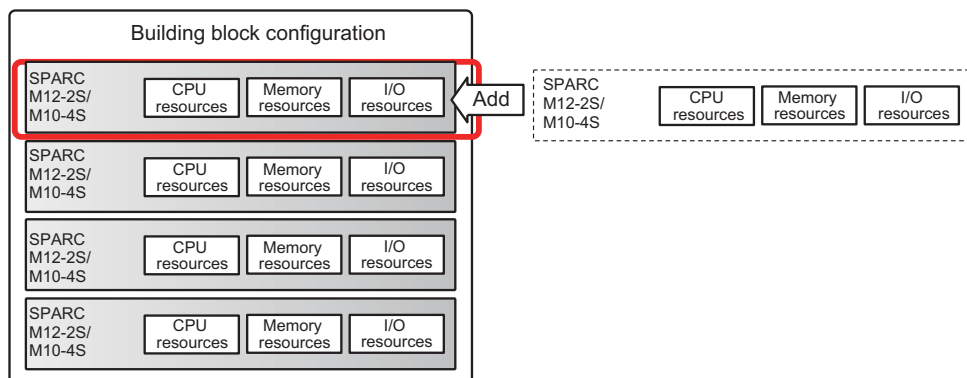
[SPARC M10-1]

The SPARC M10-1 allows up to three PCIe cards to be mounted. By connecting a PCI expansion unit, up to 23 PCIe cards can be mounted. Up to eight internal storage units can be mounted.

■ Building block configuration (SPARC M12-2S/M10-4S)

A system with a building block configuration that consists of multiple SPARC M12-2S/M10-4S units can be expanded gradually from 1 block to up to 16 blocks by stacking blocks, with a chassis used as a block.

Figure 1-15 Efficient Use of Hardware Resources in a Building Block Configuration



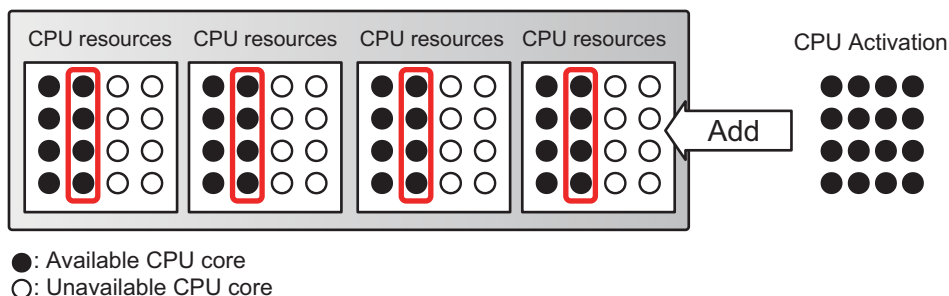
■ CPU Activation

The CPU Activation function can be used to add CPU resources at the granularity of CPU cores, which are a smaller unit than CPU chips.

You can start with the minimum required CPU resources and gradually expand the CPU resources by adding CPU Activations.

For details on CPU Activation, see "Chapter 5 CPU Activation" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

Figure 1-16 CPU Activation (Example of SPARC M10-4/M10-4S)



1.3 What is a Physical Partition?

This section describes the components of a physical partition and physical partition-related operations.

1.3.1 Understanding Physical Partition Components

A physical partition consists of multiple building blocks. In a configuration that is not a building block configuration, one server is one physical partition. The XSCF firmware handles one building block as one physical system board (PSB) when configuring a physical partition. A physical system board consists of three types of resources: CPU, memory, and I/O device. In the physical partition, the hardware resources of the physical system boards are assigned to logical system boards (LSB).

■ Physical system board (PSB)

The PSB consists of physical components (CPU, memory, and I/O) mounted in one SPARC M12/M10 chassis. In the SPARC M12-1/M10-1, a physical system board is a motherboard unit. In the SPARC M12-2/M12-2S/M10-4/M10-4S, a physical system board is a CPU memory unit (including lower (CMUL) and higher (CMUU)). A physical system board may be used as the unit representing a chassis, in maintenance for adding/removing/replacing a SPARC M12/M10 chassis. For a

system with a building block configuration, a physical system board refers to a building block (BB).

- Logical system board (LSB)

The LSB is a logical unit name assigned to a physical system board. Each physical partition has a set of logical system boards assigned to it. With one PSB number assigned to the logical system boards in the physical partition, they can be recognized in the system.

LSB numbers are used to control the assignment of resources such as memory to the physical partition.

- Physical partition number (PPAR-ID)

The PPAR-ID is a number that identifies a physical partition. You can assign a number ranging from 0 to 15. However, the following restrictions apply to physical partition numbers:

- For SPARC M12-1/M12-2/M10-1/M10-4

The physical partition number is fixed to 0. You cannot change the value.

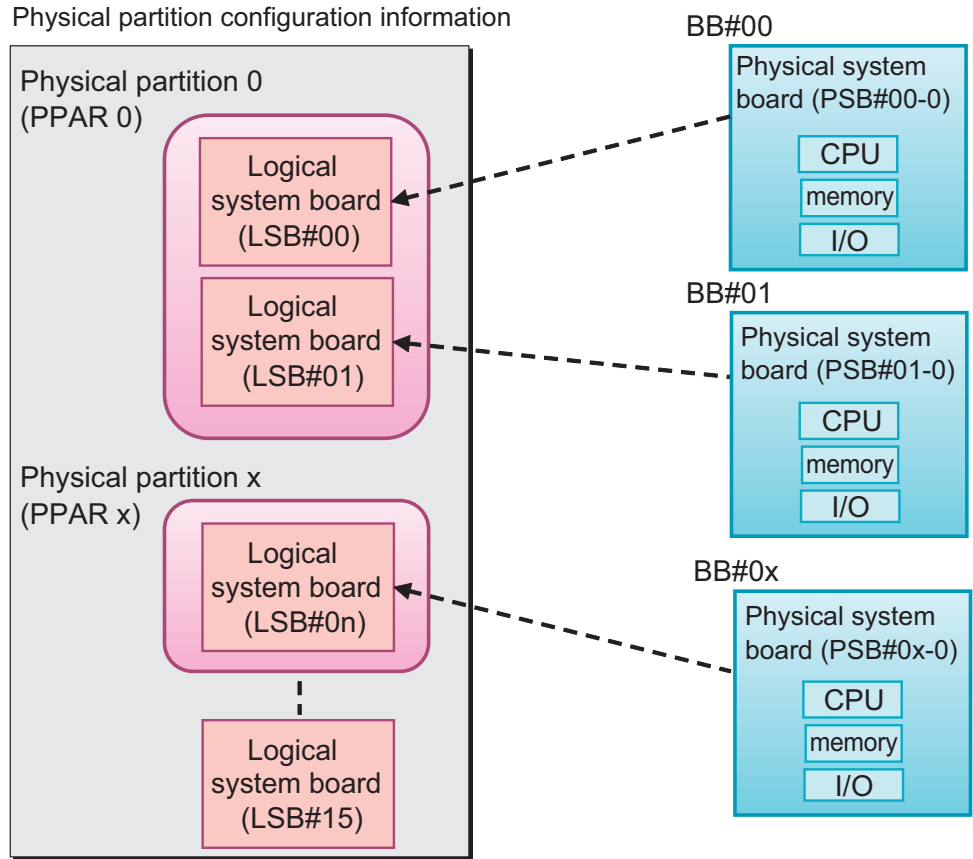
- For SPARC M12-2S/M10-4S

You can assign a physical partition number that is the same as the BB-ID of a chassis mounted on the system. If the BB-ID of an unmounted chassis is specified as a physical partition number, the power-on of the physical partition will fail. For details on the physical partition configuration recommended in consideration of SPARC M12-2S/M10-4S removal, see "[4.2 Configuring and Operating a Physical Partition](#)."

[Example] For a system with BB#0 and BB#1 mounted, 0 or 1 can be assigned as a physical partition number. If you specify 2 as a physical partition number, the power-on of the physical partition will fail.

The mapping between the logical and physical system boards on the physical partition is set in the physical partition configuration information (PPAR configuration information). [Figure 1-17](#) is a conceptual diagram of this mapping.

Figure 1-17 Conceptual Diagram of Mapping Between Logical System Boards and Physical System Boards



All the system boards that are targets for dynamic reconfiguration must be registered in the physical partition configuration information (PPAR configuration information) for the relevant physical partition. Use the `setpcl` command for their registration in the PPAR configuration information.

PPAR configuration information maps logical system boards (LSB) that comprise the physical partition to physical system boards (PSB) which are hardware resource information, and is managed with XSCF. Each piece of PPAR configuration information also stores setting information on the hardware resources of the PSBs mapped to the physical partitions and LSBs.

1.3.2 Working With a Physical Partition Configuration

This section describes how to work with a physical partition configuration with multiple SPARC M12-2S or M10-4S units.

To set the physical partition configuration and work with physical partitions, use the XSCF shell and XSCF Web provided by the XSCF. Operations through them are

collectively managed by the XSCF. In addition, XSCF security management allows only administrators with specific access privileges to perform operations related to a physical partition configuration. For details on the XSCF shell commands for physical partition configurations, see ["3.1 Operations and Commands Related to Physical Partition Configurations."](#)

For XSCF Web operations, see "Chapter 3 Configuring the System" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

Registering/Releasing a system board

"Registering a system board" means providing a physical system board (PSB) that indicates one building block (BB) with a logical system board number (LSB number) and setting it in the physical partition configuration information.

"Releasing a system board" means deleting a PSB (BB) from the physical partition configuration information. The registration of a physical system board (PSB) in physical partition configuration information enables you to perform PSB (BB) operations, such as assigning and incorporating the PSB (BB) into the physical partition.

Adding a system board

"Adding a system board" means incorporating an installed building block (PSB) or unused PSB (BB) that does not belong to any physical partition, into a physical partition. The process of adding a PSB (BB) is performed step-by-step, first with "connect" and then "configure." When you add a PSB (BB), the specified PSB (BB) first connects to the relevant physical partition. That is followed by the process of incorporating the PSB (BB). The addition of the PSB (BB) is completed at this point.

Deleting a system board

"Deleting a system board" means releasing a building block (PSB) that is no longer necessary from the physical partition to be configured. The process of deleting a PSB (BB) is performed step-by-step, first with "unconfigure" and then "disconnect."

To assign the PSB (BB) to a different physical partition, you need to include the unassign operation in the deletion process. When you delete a PSB (BB), the specified PSB (BB) is first released. Then, it is disconnected from the physical partition to which it belongs. The deletion of the PSB (BB) is completed at this time.

Assigning/Unassigning a system board

"Assigning a system board" means assigning an installed building block (PSB) or an unused PSB (BB) that does not belong to any physical partition so that it belongs to a physical partition. "Unassigning a system board" means releasing a building block or PSB from the physical partition to which it belongs.

Assigning a PSB (BB) to the relevant physical partition disables operation of the PSB (BB) from a different physical partition. When you assign a PSB (BB) to a physical partition and power on the physical partition, the PSB (BB) is added. When you power off the physical partition, the PSB (BB) is deleted and assigned to the physical partition.

When you unassign an assigned system board from its physical partition, it no longer

belongs to any physical partition and can be assigned to a different physical partition.

Replacing a system board

"Replacing a system board" means replacing a building block (PSB) for maintenance purposes. You can use this operation when replacing CPU, memory, and I/O device hardware resources. The process of replacing a PSB (BB) is performed step-by-step. When you replace a PSB (BB), the specified PSB (BB) is deleted. Once the deletion of the PSB (BB) is completed, the PSB (BB) is ready to be removed. Remove the PSB (BB) at this point. Then, after performing work such as component replacement, re-install the new PSB (BB) for the replacement. After that, once the installation is completed, the addition of the PSB (BB) completes the replacement.

System board pool function

The system board pool function is a function that puts a specific system board (PSB<BB>) into a state where it does not belong to any physical partition. This function is effective for moving a PSB (BB) among multiple physical partitions as required. For example, you can take a PSB (BB) from the system board pool and then add it to a physical partition that is subject to high CPU and memory loads. You can also return a system board to the system board pool when it is no longer needed.

You can assign a pooled PSB (BB) to a physical partition only if it is registered in physical partition configuration information (PPAR configuration information). In other words, by registering the same PSB (BB) in the configuration information for multiple PPARs, you can flexibly operate the system by incorporating or releasing the PSB (BB) according to the operating status of physical partitions. However the usage of pooled PSB (BB) must be properly managed for such operations.

You can also combine this function with the memory nullification option and I/O nullification option described in ["2.2 XSCF Conditions and Settings"](#) to easily add and delete PSBs (BBs).

Reserving a physical partition configuration change

You not only can dynamically add or delete a system board (PSB<BB>) in a physical partition but also can reserve the reconfiguration performed at the next power on/off or restart time of that physical partition.

For example, you can reserve a physical partition configuration change for the following occasions.

- When dynamic reconfiguration cannot reconfigure hardware resources for business and operational convenience.
- When you do not want to immediately change the physical partition configuration.
- When you want to prevent dynamic reconfiguration operations from changing settings and want to change the configuration immediately after a physical partition restart: An example is when you are deleting a PSB (BB) that has a driver or PCI card that does not support dynamic reconfiguration.

[Table 1-1](#) lists the operations available on the system boards (PSB<BB>) composing a physical partition.

Table 1-1 Operations Available on System Boards

Term	Description
Register	To register a system board in physical partition configuration information
Release	To release the registration of a system board from physical partition configuration information
Add	To add a system board to a physical partition
Delete	To delete a system board from a physical partition
Assign	To assign a system board to a physical partition
Unassign	To unassign a system board from a physical partition
Connect	To connect a system board to a physical partition
Disconnect	To disconnect a system board from a physical partition
Configure	To incorporate a system board into a physical partition configuration
Unconfigure	To release a system board from a physical partition configuration
Reserve	To reserve the physical partition power-on or restart time as the time of incorporation of a system board into a physical partition. This operation similarly reserves the unassignment of a system board from a physical partition.
Install	To insert a SPARC M12/M10 chassis into the system
Remove	To remove a SPARC M12/M10 chassis from the system
Replace	To remove a SPARC M12/M10 chassis for a maintenance check, etc. and to reinstall the chassis or install a new one

1.4 What is a Logical Domain?

This section describes the components of a logical domain and logical domain-related operations.

1.4.1 Understanding Logical Domain Components

A logical domain consists of virtual CPUs, virtual memory, and virtual I/Os.

- **Virtual CPU**

A CPU can be assigned to a logical domain in units of virtual CPUs (threads). In the SPARC M12/M10 system, one physical CPU (i.e., one socket) has multiple cores, each of which has threads. This means that a physical CPU has as many virtual CPUs as the number of threads. You can assign these virtual CPUs to a logical domain. Normally, assign the virtual CPUs to a logical domain in units of cores, considering the logical domain performance.

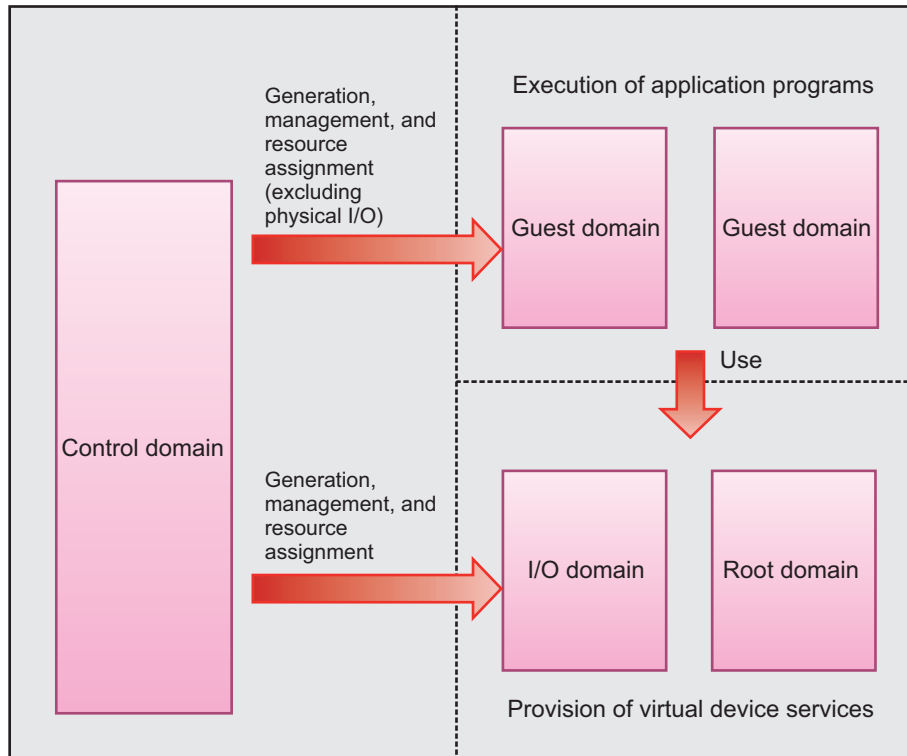
- Virtual memory
Memory can be assigned to a logical domain in units of 256 MB.
- Virtual I/O
I/O can be assigned to a logical domain in units of virtual I/Os. For example, a virtual disk is one virtual I/O, which can be used as a:
 - Physical disk
 - Physical disk slice
 - File from a file system such as ZFS or UFS
 - Logical volume from the ZFS or another volume manager

Logical domains are categorized by role as follows.

- Control domain
A control domain is a logical domain that creates and manages other logical domains and assigns resources to the logical domains. Each physical partition has only one control domain. Oracle VM Server for SPARC is installed on the control domain, where the management software Logical Domains Manager is running.
- I/O domain
An I/O domain is a logical domain that provides virtual device services. The virtual device services include a disk, network, and console.
- Root domain
A root domain is an I/O domain to which a PCIe root complex is assigned. A PCIe root complex is an entire PCIe bus, consisting of the PCIe bus, all PCI switches, and devices. Physical I/O devices belong to the root domain, which accesses them directly.
- Guest domain
A guest domain is a logical domain controlled by the control domain. It uses virtual device services from an I/O domain. Normally, middleware and application programs run on the guest domain. Oracle Solaris runs independently on the guest domain, which can thus start and stop without affecting other guest domains. Virtual CPUs, virtual memory, and virtual I/Os can be dynamically added to and deleted from the guest domain.
- Service domain
Service domain is a generic term for a domain that provides services to a guest domain. Specifically, it includes I/O and root domains.

Figure 1-18 Concept of the Relationship Between Logical Domains

Physical partition



1.4.2 Working With a Logical Domain Configuration

The operations for logical domains use Logical Domains Manager, which is management software for Oracle VM Server for SPARC. You can configure or reconfigure a logical domain by using Logical Domains Manager. For these operations, log in to the control domain, and use the `ldm` command along with various options.

This section describes the available operations for logical domains.

Configuring a logical domain

When configuring a logical domain, perform the following operations by using the `ldm` command of Logical Domains Manager, which is management software for Oracle VM Server for SPARC.

- Create a logical domain.
- Assign hardware resources as a virtual CPU, virtual memory, and a virtual I/O to the logical domain.
- Save the logical domain configuration information.

- Start/Stop the logical domain.

For logical domains other than the control domain, you can also add or delete virtual CPU, memory, and I/O hardware resources while the operating system and applications are running. This reconfiguration is referred to as logical domain dynamic reconfiguration. However, the availability of logical domain dynamic reconfiguration depends on the operating system and applications running in the logical domain.

Reconfigurable hardware resources

The following hardware resources in a logical domain can be reconfigured.

- Virtual CPU
A CPU can be reconfigured in units of virtual CPUs (threads). However, considering performance, reconfiguration is usually in units of cores.
- Virtual memory
Memory can be reconfigured in units of 256 MB.
- Virtual I/O
Virtual I/Os can be reconfigured in units of virtual disks or virtual networks. When reconfiguring virtual I/Os assigned to a guest domain, you do not have to restart the guest domain.

Before removing a virtual disk or virtual network, you need to perform the following tasks.
 - Removing a virtual disk
After unmounting the virtual disk, remove it.
 - Removing a virtual network
After executing unplumb for the virtual network interface, remove the virtual network.
- Physical I/O
Physical I/Os can be assigned to the logical domains in the following units.

Table 1-2 Version of Oracle VM Server for SPARC With Reconfigurable Physical I/O

Type of Physical I/O	Static Reconfiguration	Dynamic Reconfiguration
Virtual function of PCIe SR-IOV (Single Root I/O Virtualization)	3.0 or later	3.1 or later
PCIe endpoint	3.0 or later	3.1.1.1 or later
PCIe root complex	3.1 or later	3.2 or later (*1)

*1 The logical domains which a PCIe root complex can be added to or deleted from is Oracle Solaris 11.2 SRU11.2.8.4.0 or later.

For the requirements to assign devices (including an on-board device) using the SR-IOV, PCIe endpoint, and assignment requirements for a root complex (PCIe bus), see the following manuals.

- Latest *Product Notes* for your server
- *PCI Card Installation Guide* for your server

- *Oracle VM Server for SPARC Administration Guide* of the version used
- *Oracle VM Server for SPARC Release Notes* of the version used

When statically reconfiguring physical I/Os assigned to the control domain and the root domain, switch to delayed reconfiguration mode. Then, restart the control domain and the root domain respectively, and apply the settings for the reconfiguration.

Before deleting a physical I/O, you need to perform the following tasks.

- Deleting a physical I/O
 - If it is a disk, unmount it, and if it is a network, unplumb it. Then, after setting the physical I/O device to unused, delete it.
 - To dynamically move hardware resources of the relevant guest domain, the Logical Domains DR daemon (drd) must be operating on the guest domain.

1.4.3 Reconfiguration of a Physical I/O Device

Adding or deleting a physical I/O device

The need for dynamic reconfiguration of each physical I/O device differs depending on the version of Oracle VM Server for SPARC. For details, see "[Table 1-2.](#)"

- Before adding or deleting a physical I/O device in the control domain, you need to enter delayed reconfiguration mode.
- Before adding or deleting a physical I/O device in the root domain, you need to enter delayed reconfiguration mode or shut down the root domain (ldm stop-domain).
- Before adding or deleting a physical I/O device in an I/O domain, you need to shut down the I/O domain (ldm stop-domain).

Device driver

Drivers that control physical I/O devices in a physical partition that will be dynamically reconfigured must support the suspend and resume functions of Oracle Solaris.

Supporting SR-IOV

When combined with Oracle VM Server for SPARC, the SPARC M12/M10 can support the PCIe SR-IOV function (single root I/O virtualization function). With the SR-IOV function, you can create multiple virtual functions for one physical function by using a PCI Express card that supports SR-IOV. Assigning created virtual functions to I/O domains allows logical domains to share PCIe card bands more efficiently, resulting in reduced overhead for the service domain. For details on the SR-IOV function, see the *Oracle VM Server for SPARC Administration Guide* and the *PCI Card Installation Guide* for your server. In addition, for the available Oracle Solaris versions, see the *Oracle VM Server for SPARC Release Notes* and the *Oracle VM Server for SPARC Administration Guide*.

1.5 What is Physical Partition Dynamic Reconfiguration?

This section provides an overview of physical partition dynamic reconfiguration (PPAR DR).

1.5.1 Overview of Physical Partition Dynamic Reconfiguration

You can configure physical partitions using the building block method for SPARC M12-2S or SPARC M10-4S units. The physical partition dynamic reconfiguration (PPAR DR) function is a technology that enables the addition and removal of hardware resources (CPU, memory, and I/O) to and from a physical partition. This function works without the need to stop any task on the physical partition. The unit of addition or removal is one SPARC M12-2S or SPARC M10-4S (building block <BB>). The firmware handles this unit as a system board (PSB). The hardware resources installed on the PSB (BB) can be reconfigured logically and dynamically.

Note - For details on the support information for the hardware resources (CPU, memory, and I/O) for which the physical partition dynamic reconfiguration function can be used, see the latest *Product Notes* of the server used.

1.5.2 Using Physical Partition Dynamic Reconfiguration

The physical partition dynamic reconfiguration function (PPAR DR function) is used as follows:

- Increasing hardware resources
In response to business expansion and system load increases, you can add a system board (PSB<BB>) to a physical partition without stopping any task on the physical partition. If necessary, additional CPU Activations must be procured and configured.
- Reducing hardware resources
To efficiently use hardware resources according to changes in the size of your business, you can delete a PSB (BB) from a physical partition without interrupting the ongoing business.
- Moving hardware resources
To handle temporary business expansion and load increases, you can temporarily delete the PSB (BB) from another physical partition. The deleted PSB (BB) can then be added to a physical partition that requires an increase in hardware resources. This enables hardware resources to be moved while tasks on the two physical partitions are running. The system can be configured to flexibly respond to

variations in the system load.

- Active replacement at part failure
If the hardware resources on a PSB (BB) are degraded as a result of a failure of the PSB (BB), you can temporarily delete that PSB (BB) from the physical partition and replace the failed parts without stopping any task on the physical partition. After the parts are replaced, you can then restore the original configuration by adding the PSB (BB) to the physical partition without stopping any task.

1.5.3 Dynamic Reconfiguration of CPUs and Memory

Adding or deleting a CPU

Oracle VM Server for SPARC automatically recognizes and makes available any CPUs added through physical partition dynamic reconfiguration.

One of the following operations is performed for any system board (PSB<BB>) incorporated or CPU added through physical partition dynamic reconfiguration.

- If the PSB (BB) is added to a physical partition with logical domain configuration information set to factory-default (configuration at factory shipment), all mounted CPUs on the added PSB (BB) are automatically added to the control domain.
- If a new PSB (BB) is added to a physical partition with logical domains configured or logical domain configuration information saved, the mounted CPUs on the PSB (BB) are not assigned to any of the logical domains. To use these CPUs, you need to add them to a logical domain by using the `ldm` command.
- If you release a PSB (BB) from a physical partition with logical domains configured, the CPU assignment of the logical domains is automatically deleted as a result when you replace the PSB (BB). In that case, adding a PSB (BB) automatically adds as many CPUs as were deleted to the logical domains.

To delete a CPU through dynamic reconfiguration, the following conditions must be met. If the conditions are not met, dynamic reconfiguration processing of the physical partition stops and a message appears.

- No running processes are bound to the CPU to be deleted.
If any process is bound, you need to release the binding of the process or stop the process.
- The CPU to be deleted does not belong to any processor set.
If the CPU belongs to any processor set, you need to delete it from the processor set in advance by using the `psrset` command.
- A CPU core is assigned to the logical domains without the core ID (CID) specified in the `ldm add-core` command or the `ldm set-core` command.

If the above conditions are not met when logical domain resources are deleted and a system board is released during physical partition dynamic reconfiguration, physical partition dynamic reconfiguration processing stops and a message appears.

Adding or deleting memory

One of the following operations is performed for any system board (PSB<BB>)

incorporated or memory added through physical partition dynamic reconfiguration.

- If the PSB (BB) is added to a physical partition with logical domain configuration information set to factory-default (configuration at factory shipment), all mounted memory on the added PSB (BB) is automatically added to the control domain.
- If a new PSB (BB) is added to a physical partition with logical domains configured or logical domain configuration information saved, the mounted memory on the PSB (BB) is not assigned to any of the logical domains. To use this memory, you need to add it to the logical domains by using the `ldm` command.
- If you release a PSB (BB) from a physical partition with logical domains configured, when you replace the PSB (BB), the memory assignment of the logical domains is automatically deleted as a result. In that case, adding a PSB (BB) automatically adds the same size memory as was deleted to the logical domains. Before deleting the PSB (BB) through physical partition dynamic reconfiguration, note the following:
 - If the memory with the specified physical address has been assigned to a logical domain by the `mblock` option of the `ldm add-memory` command or `ldm set-memory` command, do not delete the PSB (BB) through physical partition dynamic reconfiguration.

1.5.4 Software Conditions and Confirmation Methods

Software conditions needed to enable the use of physical partition dynamic reconfiguration

[Table 1-3](#) and [Table 1-4](#) list the XCP, Oracle Solaris, and required SRU/patches needed to perform physical partition dynamic reconfiguration. To use the physical partition dynamic reconfiguration function, an appropriate combination of versions of the XCP firmware and Oracle VM Server for SPARC software must be set. For the latest software requirements, see the latest *Product Notes* of the server used.

Table 1-3 List of Oracle Solaris and Required SRU/Patches Needed for Physical Partition Dynamic Reconfiguration on SPARC M12-2S

OS Version	Domain type			
	Control domain Non-virtual environment	Root domain	I/O domain	Guest domain
Oracle Solaris 11	Oracle Solaris 11.4 (*1)	Oracle Solaris 11.4 or later (*2)	Oracle Solaris 11.4 or later (*2)	Oracle Solaris 11.4 or later (*2)
	Oracle Solaris 11.3 (*1) SRU11.3.17.5.0 or later	Oracle Solaris 11.3 or later (*2)	Oracle Solaris 11.3 or later (*2)	Oracle Solaris 11.3 or later (*2)
		Oracle Solaris 11.2 or later (*2)	Oracle Solaris 11.2 or later (*2)	Oracle Solaris 11.2 or later (*2)
			Oracle Solaris 11.1 (*2) SRU1.4 or later	Oracle Solaris 11.1 or later (*2)

Table 1-3 List of Oracle Solaris and Required SRU/Patches Needed for Physical Partition Dynamic Reconfiguration on SPARC M12-2S (*continued*)

OS Version	Domain type			
	Control domain Non-virtual environment	Root domain	I/O domain	Guest domain
Oracle Solaris 10	-	-	-	Oracle Solaris 10 1/13 (*3) 150310-03 or later Oracle Solaris 10 8/11 (*3)(*4) Oracle Solaris 10 1/13 SPARC bundle Oracle Solaris 10 9/10 (*3)(*4) Oracle Solaris 10 1/13 SPARC bundle

*1 system/ldoms and system/ldoms/ldomsmanager packages are required. These packages are included in group/system/solaris-large-server and group/system/solaris-small-server.

*2 system/ldoms package is required. This packages are included in group/system/solaris-large-server and group/system/solaris-small-server.

*3 There are no limitations on the LSB number that can be assigned to the CPU in the guest domain. However, the maximum number of CPU (vcpu) that can be assigned in one guest domain is 1024 when Oracle Solaris 10 is running on the guest domain.

*4 You need to perform the additional procedures before installing Oracle Solaris 10 9/10 or Oracle Solaris 10 8/11. For details, see procedures described in "Notes on Installing Oracle Solaris 10 on a Guest Domain" in the latest *Product Notes*.

Table 1-4 List of XCP/Oracle Solaris and Required SRU/Patches Needed for Physical Partition Dynamic Reconfiguration on SPARC M10-4S

Server	XCP	Oracle Solaris	Required Package Required Product	Required SRU Required Patch
SPARC M10-4S	2220 or later	Oracle Solaris 11.2 or later	system/ldoms(*1) system/ldoms/ldomsmanager(*2)	None
		Oracle Solaris 11.1	system/ldoms(*1) system/ldoms/ldomsmanager(*2)	SRU11.1.14.5.0 or later (*3)
		Oracle Solaris 10 1/13	Oracle VM for SPARC 3.1 (*4)	150817-02 or later (*4)(*5)

*1 Required for control and guest domains. Included in group/system/solaris-large-server and group/system/solaris-small-server.

*2 Required only for the control domain. Included in group/system/solaris-large-server and group/system/solaris-small-server.

*3 Required for control and guest domains. SRU11.1.14.5.0 includes Oracle VM Server for SPARC 3.1.0.1. To ensure the stable operation of physical partition dynamic reconfiguration, however, the Oracle Solaris 11.1 patch (CR:17709858) is required. This issue has been resolved in SRU11.1.15.4.0 and later.

*4 Required only for the control domain.

*5 Do not use patches between 150400-01 and 150400-06.

How to check the software version:

■ How to check the version of the XCP firmware

1. **Log in to the XSCF shell.**
2. **Execute the version -c xcp command to check the firmware version information.**

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2220
XCP1 (Reserve): 2220
BB#01-XSCF#0 (Standby)
XCP0 (Current): 2220
XCP1 (Reserve): 2220
```

■ **How to check the version of the Oracle VM Server for SPARC software**

1. **Log in to the control domain console of the physical partition.**
For details on how to log in to the control domain console, see "8.3 Switching to the Control Domain Console From the XSCF Shell" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.
2. **For Oracle Solaris 11, execute the pkg info entire command to check the version of the SRU of Oracle Solaris.**

```
# pkg info entire
Name: entire
Summary: entire incorporation including Support Repository Update (Oracle
Solaris 11.1.14.5.0). <-- Version of SRU
```

3. **For Oracle Solaris 10, execute the ldm -V command to check the version of Oracle VM Server for SPARC.**

```
# ldm -V
Logical Domains Manager (v 3.1.0.1) <-- Version of Oracle VM Server for SPARC

Hypervisor control protocol v 1.9
Using Hypervisor MD v 1.3
```

1.6 Mixed Use of the SPARC64 X+ Processor and SPARC64 X Processor

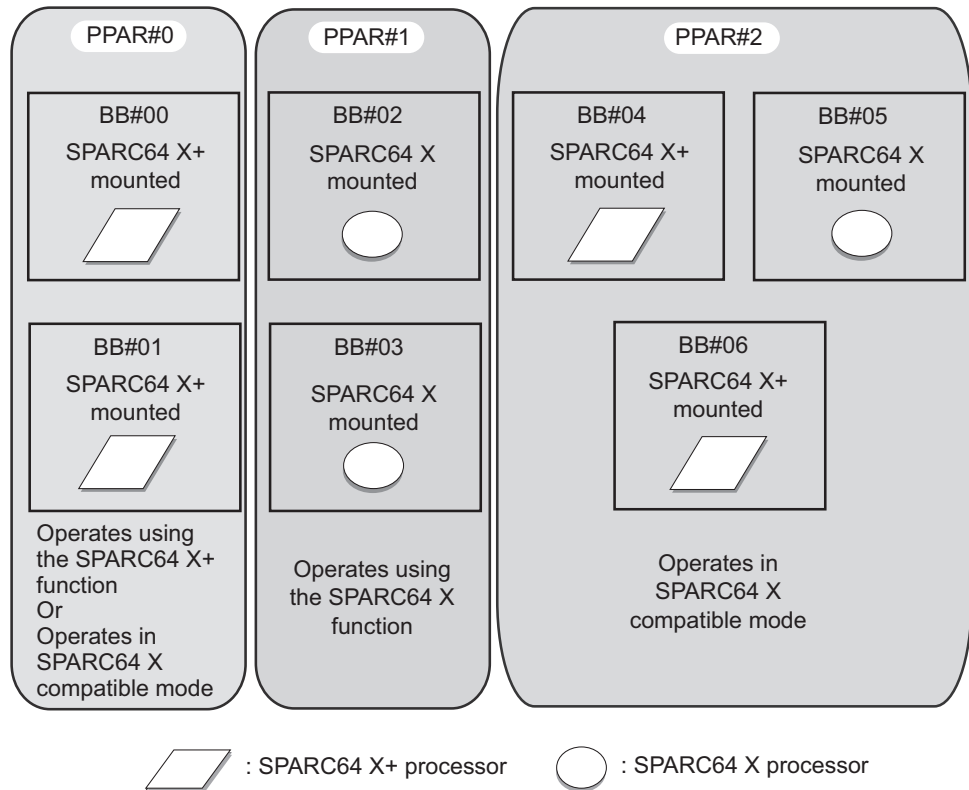
For the SPARC M10-4S system, a SPARC M10-4S with the SPARC64 X+ processor and a SPARC M10-4S with the SPARC64 X processor can be used together in a mixed configuration as one physical partition.

When you start a physical partition with a mixed configuration consisting of the

SPARC64 X+ processor and the SPARC64 X processor, all of the processors operate in the SPARC64 X compatible mode.

For details on the combined use of the SPARC64 X+ processor and the SPARC64 X processor, see "2.6.3 Conditions for a Mixed Configuration With the SPARC64 X+ Processor and the SPARC64 X Processor Within a PPAR."

Figure 1-19 Example of a Configuration Consisting of Processors and Physical Partitions in a SPARC M10 System



1.6.1 CPU Operation Type and CPU Operational Mode

In SPARC M10-4S with the CPU operational mode set by using the `setpparmode` command of the XSCF firmware, you can specify the CPU operation type for each physical partition. This automatically determines whether the CPU operates using the SPARC64 X+ processor function or the SPARC64 X processor function when Oracle Solaris next starts.

The CPU operations include the following types.

- Operates with the SPARC64 X+ function
All of the CPUs in the physical partitions operate using the extended SPARC64 X+ processor function.
- Operates with the SPARC64 X function

All of the CPUs in the physical partitions operate using the SPARC64 X processor function. This is used when operating all of the CPUs with the SPARC64 X processor function regardless of the CPU type installed on the system boards configuring the physical partitions. The SPARC64 X processor always operates using the SPARC64 X processor function.

Note - For details on the CPU operation types and supported versions of the XCP firmware, see the latest *SPARC M10 System Product Notes* of the latest XCP version.

Note - For details on the CPU operational mode settings, see "[2.6.2 CPU Operational Mode Settings](#)."

Note - For details on the CPU operational mode and physical partition dynamic reconfiguration, see "[2.6.4 Relationship Between the CPU Operational Mode and Physical Partition Dynamic Reconfiguration](#)."

Chapter 2

Conditions and Settings for System Operation

This chapter describes what you should know before starting domain configuration.

- [XSCF Status Management](#)
- [XSCF Conditions and Settings](#)
- [Oracle Solaris Conditions and Settings](#)
- [Logical Domain Conditions and Settings](#)
- [Dynamic Reconfiguration Conditions and Settings](#)
- [Considerations When Using the SPARC64 X+ Processor](#)

2.1 XSCF Status Management

To perform physical partition reconfiguration properly, you need to perform operations according to the status of the physical partition and system board (PSB<BB>) that appear as one building block. This section describes the status information for the physical partitions and PSB (BB) managed by the XSCF and provides notes on each status. You can thus understand the conditions for operating the functions for physical partition reconfiguration.

2.1.1 Physical Partition Status

The XSCF manages the physical partition status. You can use the user interface provided by the XSCF to display and refer to the physical partition status. The following example uses the `showpparstatus` command to check the physical partition status from the command line-based XSCF shell.

```
XSCF> showpparstatus -p 0
PPAR-ID PPAR Status
00      Initialization Complete
```

For details of the user interface, see "[Chapter 3 Operations for Domain Configuration.](#)"

The XSCF manages the following physical partition-related statuses.

Table 2-1 Physical Partition Status

Status	Meaning
Powered Off	Power-off state
Initialization Phase	State where the power-on self-test (POST) is running
Initialization Complete	POST completed state
Running	Running state after POST processing is completed
Hypervisor Aborted	Aborted state of Hypervisor until it is reset
-	Other than the above (state where the physical partition is not defined)

To perform physical partition dynamic reconfiguration, you need to determine how to work with system boards (PSB<BB>) according to the status of the relevant physical partition.

2.1.2 System Board Status

The XSCF manages the status for one building block in system board (PSB) units. You can use the user interface provided by the XSCF to display and refer to the PSB status. The following example uses the showboards command to confirm the PSB status from the command line-based XSCF shell.

XSCF> showboards -av									
PSB	R	PPAR-ID (LSB)	Assignment	Pwr	Conn	Conf	Test	Fault	
----	-	-----	-----	----	----	----	-----	-----	
00-0	*	00 (00)	Assigned	n	n	n	Unmount	Normal	
01-0		00 (01)	Assigned	y	y	n	Passed	Normal	
02-0		00 (02)	Assigned	y	y	n	Passed	Normal	
04-0		00 (04)	Assigned	y	y	n	Passed	Normal	
05-0		01 (00)	Assigned	y	y	n	Passed	Normal	
06-0		01 (01)	Assigned	y	y	n	Passed	Normal	
09-0		00 (09)	Assigned	y	y	n	Passed	Normal	
13-0		13 (00)	Assigned	y	y	n	Passed	Normal	
15-0		13 (01)	Assigned	y	y	n	Passed	Normal	

For details of the user interface, see "[Chapter 3 Operations for Domain Configuration.](#)"

The XSCF manages the following system board-related statuses.

Table 2-2 System Board Status

Display Item	Description	Meaning
PSB	xx-y (xx: Integer from 00 to 15; y: Fixed at 0)	Physical system board (PSB) number that represents one building block (BB) (where xx is the BB-ID)
R (*1)	*	The state where the PSB (BB) is incorporated into the current physical partition configuration or released and the configuration changes when the physical partition restarts
PPAR-ID	Integer from 0 to 15	Number of the physical partition to which the PSB (BB) is assigned
	SP	PSB (BB) in the system board pool state
	Other	State where physical partition configuration information is set for a physical partition having user privileges and the system board belongs to a physical partition having no user privilege
LSB	Integer from 00 to 15	Logical system board number used by the physical partition
Assignment (State of being assigned to PPAR)	Unavailable	PSB (BB) that is not assigned to a physical partition and is in any of following states (including cases where the PSB cannot be recognized): system board pool, undiagnosed, diagnosis in progress, abnormal diagnosis
	Available	PSB (BB) in the system board pool state, with a diagnosis ended normally
	Assigned	PSB (BB) assigned
Pwr (Power-on state of system board)	n	Power-off state
	y	Power-on state
Conn (Incorporated state in PPAR configuration)	n	The state where the PSB (BB) is released from the physical partition configuration
	y	The state where the PSB (BB) is incorporated into the physical partition configuration
Conf (Running state in Oracle Solaris)	n	State where Oracle Solaris is not operating on PSB (BB)
	y	State where Oracle Solaris is operating on PSB (BB)
Test (Diagnosis state)	Unmount	Unmounted or undefined state
	Unknown	Undiagnosed state
	Testing	Diagnosis in progress
	Passed	State of normal end of diagnosis
	Failed	The state where a diagnosis has detected an abnormality, and the PSB (BB) is not operating

Table 2-2 System Board Status (*continued*)

Display Item	Description	Meaning
Fault (Degraded state)	Normal	Normal state
	Degraded	The state where there is a degraded part (the PSB (BB) can operate)
	Faulted	State where the PSB (BB) cannot operate because of an abnormality or cannot be managed because of a communication failure

*1 This item is displayed only if the -v option is specified.

The XSCF changes/sets the PSB (BB) status according to the situation. Such situations include installing/removing a PSB (BB), registering the physical partition configuration information, and starting/stopping a physical partition. The status similarly changes as a PSB (BB) is added or deleted during physical partition dynamic reconfiguration.

To perform physical partition dynamic reconfiguration, you need to determine how to work with the PSB (BB) according to the status of the relevant PSB (BB).

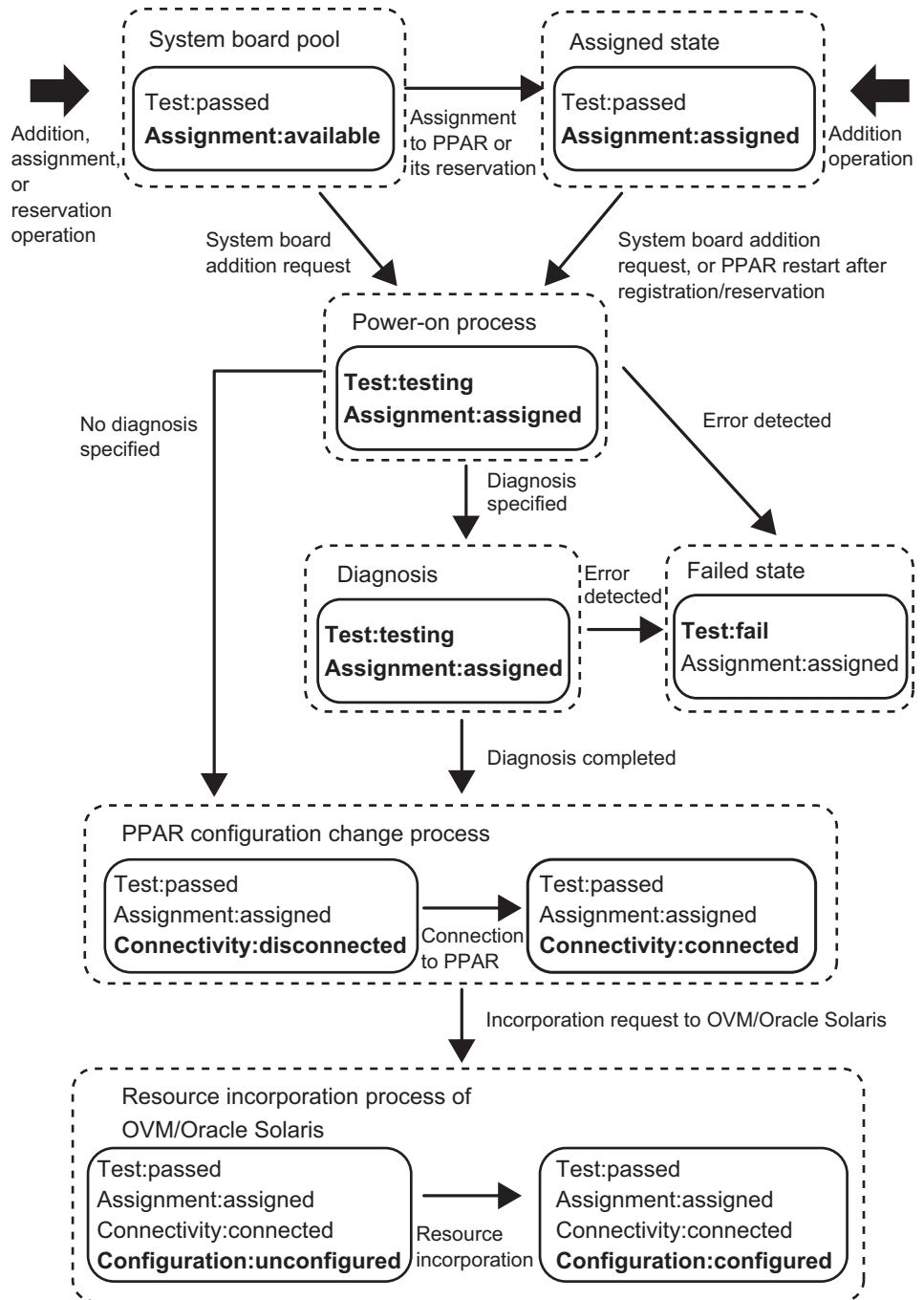
2.1.3 Status Change

This section describes the process flow and system board (PSB<BB>) status change that accompany each physical partition configuration change.

System Board Addition Process

This section uses a schematic diagram to describe the process flow and system board (PSB<BB>) status change at the time of the PSB (BB) addition or PSB (BB) addition reservation operation. [Figure 2-1](#) shows the major PSB (BB) status changes.

Figure 2-1 System Board Addition Process

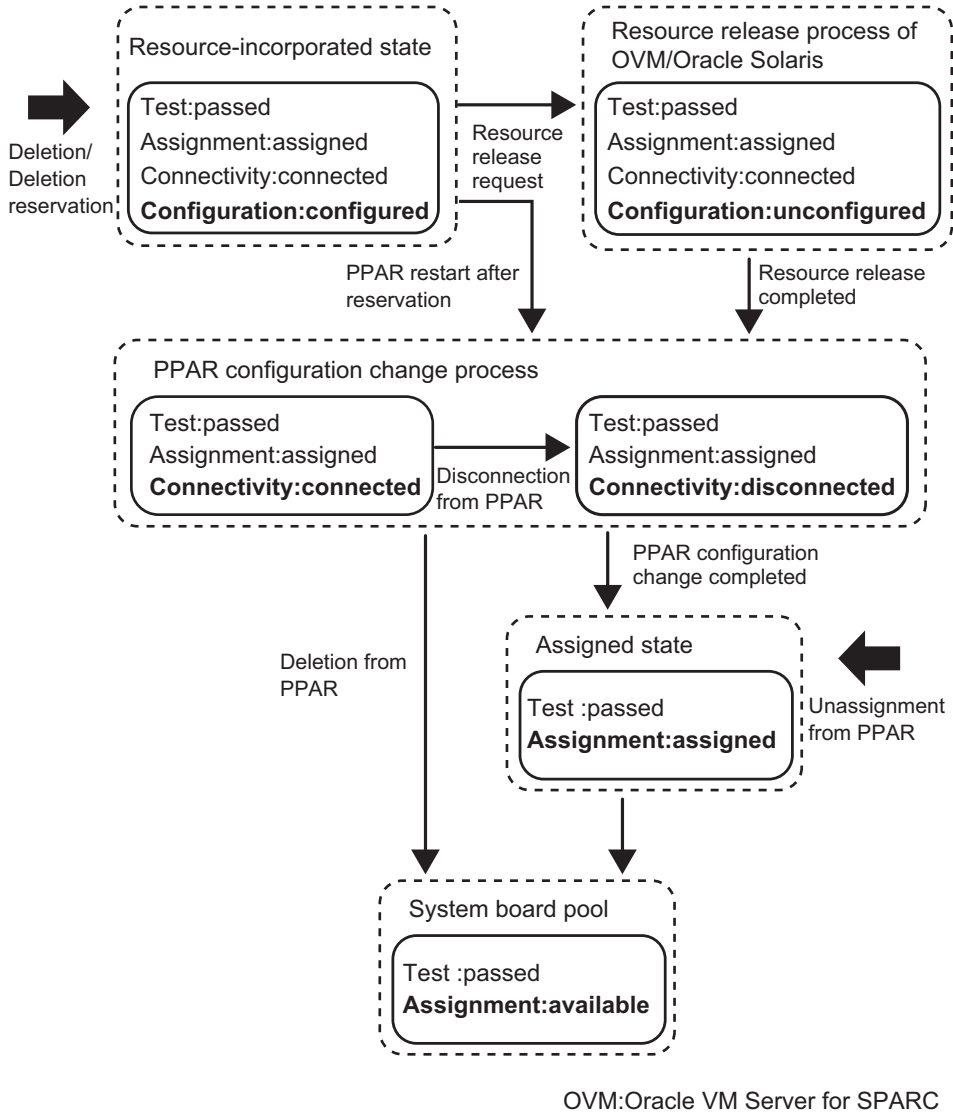


OVM:Oracle VM Server for SPARC

System board deletion process

This section uses a schematic diagram to describe the process flow and system board (PSB<BB>) status change at the time of the PSB (BB) deletion or PSB (BB) deletion reservation operation. Figure 2-2 shows the major PSB (BB) status changes.

Figure 2-2 System Board Deletion Process

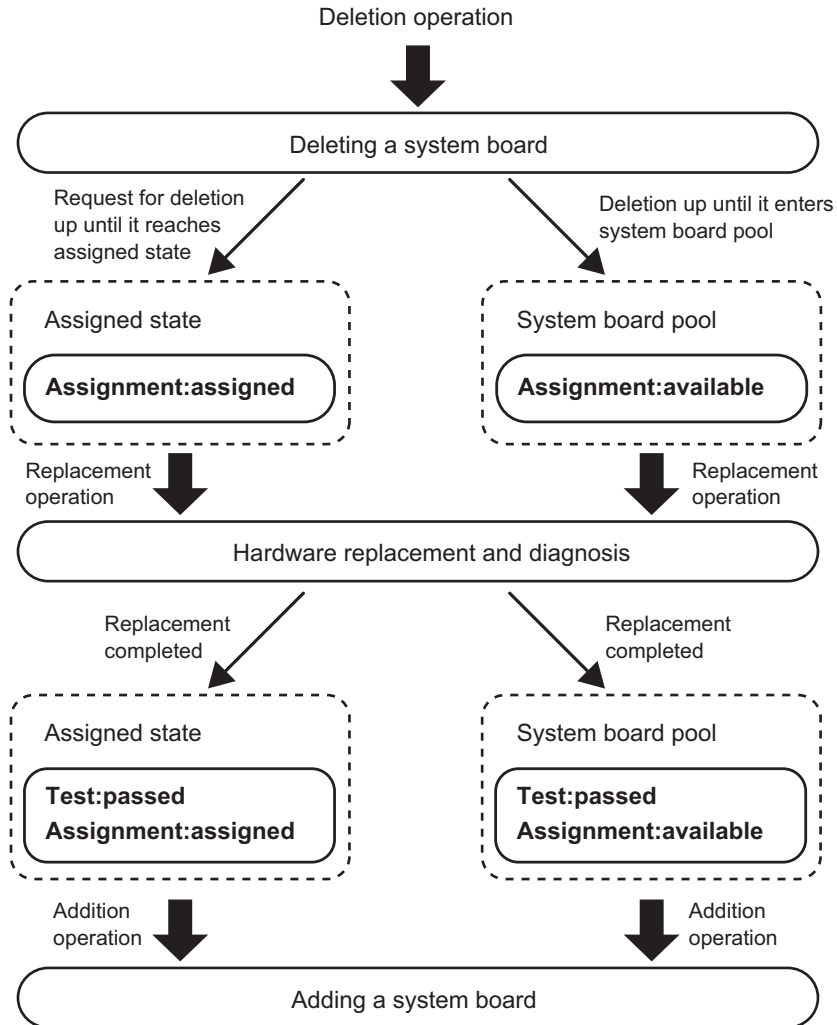


System board replacement process

This section uses a schematic diagram to describe the process flow and system board

(PSB<BB>) status change at the time of the PSB (BB) replacement operation. [Figure 2-3](#) shows the major PSB (BB) status changes. The statuses are shown as an example of the statuses before and after hardware replacement. When you replace hardware, its status may not match the corresponding status shown here. For details of the PSB (BB) addition and deletion process flows and statuses during the processes, see "[System Board Addition Process](#)" and "[System Board Deletion Process](#)." For details on the hardware replacement operations, see the *Service Manual* for your server.

Figure 2-3 System Board Replacement Process



2.2 XSCF Conditions and Settings

This section describes the conditions required for the XSCF and the settings made using the XSCF.

2.2.1 Configuration Conditions Required for the XSCF

When adding a system board (PSB<BB>), you cannot insert the system board into a physical partition simply by installing and connecting the SPARC M12-2S/M10-4S. Use the XSCF shell or XSCF Web to register the PSB (BB) in the physical partition configuration information (PPAR configuration information). This enables you to perform operations on the physical partition. When working with a physical partition, you need to confirm that the added PSB (BB) is registered in the PPAR configuration information before incorporating the PSB (BB).

When deleting or replacing a PSB (BB), the PSB (BB) should already be registered in the PPAR configuration information. You do not have to confirm that it is registered in the PPAR configuration information.

2.2.2 Settings From the XSCF

Physical partition configuration-related functions include several optional functions for reconfiguring a physical partition more smoothly and continuing operation. The functions prevent complexities regarding Oracle Solaris reconfiguration, memory allocation, etc. Use the XSCF shell or XSCF Web to configure these optional functions. This section describes the following optional functions:

- Configuration policy option
- Memory nullification option
- I/O nullification option

Use the `setpctl` command to set these options.

Setting the configuration policy option

When a physical partition is powered on or a system board (PSB<BB>) is added with physical partition dynamic reconfiguration, the hardware is subject to a power-on self-test to ensure safe insertion of the PSB (BB). The configuration policy is an option that specifies the degradation range that is applied if an error is detected by these power-on self-tests.

You can set the unit of degradation as the following: the part where the hardware error is detected, the system board (PSB) unit (BB unit) where that part is mounted, or the physical partition. [Table 2-3](#) lists the values and the units of degradation that can be set for this option. The default for this option is "fru."

Table 2-3 Unit of Degradation

Setting Value	Unit of Degradation
fru	Degradation is performed in units of parts, such as a CPU or memory.
psb	Degradation is performed in system board (PSB) by units (BB units).
system	The unit of degradation is the physical partition. The physical partition is stopped without degradation.

Note - Stop the relevant physical partition before setting the configuration policy option.

Memory nullification option

This option prevents the memory on a system board (PSB<BB>) to be incorporated into a physical partition from being used in logical domains. You can use this option in combination with the I/O nullification option to incorporate only the CPUs on a system board added to the target physical partition, into logical domains without incorporating the memory into the logical domains. You can select either "true" or "false" for this option. The former enables memory nullification, and the latter disables it. The default is "false."

Note - Even with the memory nullification option enabled, you need to actually load memory to diagnose and manage the PSB (BB). The memory available in the physical partition decreases and performance may deteriorate when the memory nullification option is enabled. You need to consider the impact on business before setting this option.

Note - Set the memory nullification option when the relevant PSB (BB) is in the system board pool state or when the physical partition power is off.

I/O nullification option

This option prevents the PCI card, disk drive, and basic local area network (LAN) port on a system board (PSB<BB>) and PCI expansion unit from being inserted into the target physical partition. If only the CPUs and memory on the PSB (BB) need to be used in the physical partition, specify "true" to enable the I/O nullification option. If the PCI card and I/O device of the PSB (BB) need to be used in the physical partition, set this option to "false." In this case, you need to fully understand the restrictions on using these I/O devices. You also need to stop any software (e.g., application program or daemon) and I/O domains that use the I/O devices before deleting the PSB (BB) with dynamic reconfiguration.

If the physical partition is started with the logical domain configuration information configured to the factory-default settings, it is started according to the setting of the I/O nullification option. Suppose that the physical partition is started with logical domain configuration information other than the factory-default settings, and that the root complex and the I/O devices of the relevant system board belong to any one

of the logical domains. The system determines that the physical partition configuration has been changed and starts the physical partition by switching to the factory-default settings. If the above does not belong to any of the logical domains, switching to the factory-default settings is not performed. If the PSB (BB) is added by means of dynamic reconfiguration, insertion processing is performed for the physical partition according to the settings made for the I/O nullification.

The default setting for the I/O nullification option is "false."

Note - Set the I/O nullification option when the relevant PSB (BB) is in the system board pool state or when the physical partition power is off.

2.2.3 Memory Mirror Mode Considerations

Memory mirror mode is a function that duplicates memory to ensure the hardware reliability of memory. You can enable memory mirror mode to continue operation of a physical partition even if some part of memory fails, as long as recovery from the failure is possible. When physical partition dynamic reconfiguration is used, enabling memory mirror mode does not limit the function of physical partition dynamic reconfiguration.

However, you have to be careful about the configuration and operation of a physical partition. For example, if you perform an operation on a system board (PSB<BB>) with memory mirror mode enabled in dynamic reconfiguration, when deleting the PSB (BB), the memory resources used by Oracle VM Server for SPARC and Oracle Solaris will be moved to a different PSB (BB). At this time, if memory mirror mode is disabled for the destination PSB (BB), the memory resources used by Oracle VM Server for SPARC and Oracle Solaris are moved properly, but operation will continue with decreased memory reliability as a result. Set memory mirror mode by considering requirements for the domain configuration, operation, etc. and developing a plan in advance.

For details on memory mirroring settings, see "Chapter 14 Configuring a Highly Reliable System" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

2.2.4 Considerations When Adding a CPU Module

If you add a CPU module (CMUU) when I/O bus reconfiguration is enabled, the device path of the I/O device that has been in use changes.

Normally, the I/O bus reconfiguration (ioreconfigure) function, which can be set using the -m function option of the setpparmode command, should be disabled (false).

Enabling the I/O bus reconfiguration function (true) may make it necessary to reinstall Oracle Solaris.

For details on the setpparmode (8) command, see the *Fujitsu SPARC M12 and Fujitsu*

2.3 Oracle Solaris Conditions and Settings

This section describes the required conditions and settings for starting operation after physical partition reconfiguration.

2.3.1 I/O and Software Requirements

I/O devices in a redundant configuration

If multiple chassis are used to configure the physical partition or if a business process is hindered due to an I/O device failure, it is recommended that software be used that enables you to configure the I/O device redundant configuration and switch I/O devices.

2.3.2 Swap Space-related Considerations

The available virtual memory capacity is the total of the capacity of memory mounted in the system and the swap space on the disk. You need to ensure that the system has sufficient available memory capacity to perform all necessary operations.

Considerations when adding a BB (PSB)

By default, Oracle Solaris also uses swap space for the purpose of storing system crash dumps. You should use a dump-dedicated device instead. For details, see the man page of `dumpadm` for Oracle Solaris. The default size of the swap space used to store crash dumps changes according to the size of the mounted memory capacity.

The size of the dump device that stores crash dumps must be larger than that of the mounted memory. If a system board is added and the mounted memory increases, reconfigure the dump device as required. For details, see the man page of `dumpadm` for Oracle Solaris.

Considerations when deleting a BB (PSB)

If deleting a BB (PSB) also deletes memory in a domain, memory on the BB (PSB) is swapped with the swap space on the disk. Then, memory decreases as much as the memory capacity deleted. Before executing a command to delete a BB (PSB), check the entire swap space to confirm that there is enough swap space for storing the physical memory data of the BB (PSB). Also, note that there are cases where a disk connected to the BB (PSB) to be deleted is used as a part of the swap space. That is, if deleting the BB (PSB) decreases the swap space, you also need to calculate the swap space that will be lost.

- Suppose (available swap space: 1.5 GB) > (deleted memory capacity: 1 GB). After the deletion, the available memory capacity becomes 0.5 GB.
- Suppose (available swap space: 1.5 GB) < (deleted capacity: 2 GB). Then, the BB (PSB) deletion process fails.

To find out the size of the currently available swap space, execute the `swap -s` command on Oracle Solaris and look at the size indicated as "available." For details, see the man page of `swap` for Oracle Solaris.

2.3.3 Real-Time Process Considerations

When physical partition dynamic reconfiguration deletes a system board, Oracle Solaris is suspended on the logical domain to which memory on the system board being deleted is assigned. Note that if the system has real-time requirements (which are displayed for any real-time processes), this processing may have a considerable impact on their processes.

2.4 Logical Domain Conditions and Settings

Note - Logical Domains Manager, which is management software for Oracle VM Server for SPARC, is used to configure a logical domain. When configuring a logical domain, see the Oracle VM Server for SPARC manual of the version used and find out the required software.

Note - This section only provides as much information as necessary to configure a logical domain. For details on the functions available in Oracle VM Server for SPARC, see the Oracle VM Server for SPARC manual of the version used and find out the required software.

Set the following contents when configuring a logical domain.

Initial configuration of the control domain

In the initial state, all the hardware resources are assigned to the control domain (factory-default configuration). Therefore, you should review the assignment of hardware resources on the control domain to prepare the hardware resources for guest domains. To do so, perform the following work:

- Default service setting
- Delayed reconfiguration mode setting
- Releasing virtual CPU resources
- Releasing virtual memory resources
- Releasing I/O resources or virtual I/O resources
- Saving control domain settings

Configuration of a guest domain

After reviewing the assignment of hardware resources to the control domain, assign hardware resources to each guest domain. To do so, perform the following work:

- Virtual CPU assignment
- Virtual memory assignment
- Virtual network settings
- Virtual disk settings
- Startup device settings
- Automatic restart settings
- Saving guest domain settings

2.4.1 Consideration of Logical Domain Configuration

This section describes what should be considered when configuring a logical domain.

Consideration of the number of logical domains

The maximum number of logical domains that can be configured is the same as the number of virtual CPUs (threads) that can be assigned to logical domains. Also, the maximum number of logical domains in each physical partition is limited by software. [Table 2-4](#) lists, by model, the maximum number of logical domains that can be configured.

Table 2-4 Maximum Number of Logical Domains by Model

Model	Maximum Number of Logical Domains
SPARC M12-1	48
SPARC M12-2	192
SPARC M12-2S	Number of threads per physical partition or 256, whichever the smaller
SPARC M10-1	32
SPARC M10-1 (3.7 GHz)	16
SPARC M10-4	128
SPARC M10-4 (3.7 GHz)	64
SPARC M10-4S	Number of threads per physical partition or 256, whichever the smaller

Normally, assign virtual CPUs to a logical domain in units of cores.

Before creating more logical domains than the number of cores in a physical partition, perform verification such as an operation test.

Considerations when using the automatic replacement function for CPUs

You need to consider the CPU configuration when using the automatic replacement function for CPUs, which is supported by Oracle VM Server for SPARC 3.0 and later. For details, see "10.7 Setting Automatic Replacement of Failed CPU Cores" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

Operation of the Logical Domains DR daemon

To assign hardware resources to a logical domain, the Logical Domains DR daemon (drd) must be operating on the control domain that manages the logical domain. For details of the drd daemon, see the *Oracle VM Server for SPARC Reference Manual*.

Registration of a CPU Activation key

Before a CPU core is assigned to a logical domain, it must be made available by assigning the CPU Activation key registered in the XSCF to the physical partition.

CPU Activation key is provided on the CD-ROM labeled "<model name> CPU Activation" supplied for server deployment. Have the CD-ROM on hand before registering a CPU Activation key.

The CPU Activation keys are contained in text files in the "ACTIVATION_KEY" folder on the CD-ROM. A file for registering the keys as a batch (XXXXX_XX.TXT) and another for registering them one at a time (XXXXX_XX_001.TXT, etc.) are provided. Use either file as needed for the situation.

Note - The XSCF for system deployment may already contain CPU Activation keys. The registered keys would be the same as those included on the CD-ROM that comes with the system. In this case, you do not have to register the keys to the XSCF.

Note - If the number of CPU cores made available by the CPU Activation keys decreases, reassign CPU cores so that the number of CPU cores assigned to physical partitions is the number of available CPU cores or less.

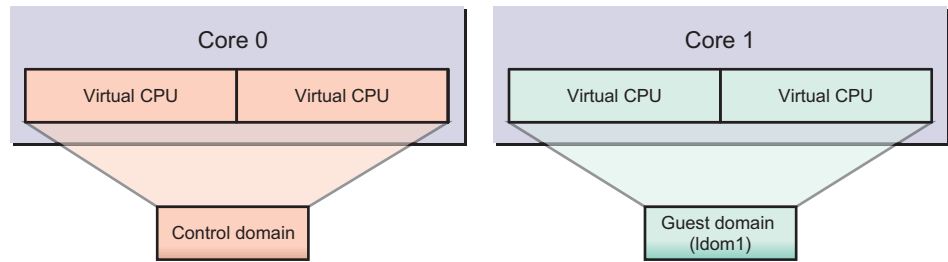
Notes on initial settings of the control domain

Do not use the memory dynamic reconfiguration (DR) function of the ldm command when making the initial settings of the control domain. Use delayed reconfiguration mode for the initial settings of the control domain. The mode reflects the configuration information through a logical domain restart.

Unit of virtual CPU assignment

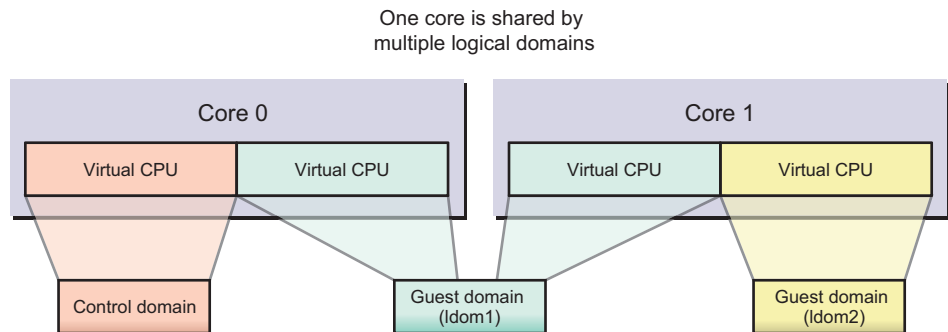
Although virtual CPUs can be assigned in units of cores or threads, we recommend you normally assign them in units of cores. [Figure 2-4](#) shows an example of normal assignment.

Figure 2-4 Example of Normal Assignment



You should note that performance may deteriorate in cases where virtual CPUs are not assigned in units of cores. Figure 2-5 shows an example of performance deterioration.

Figure 2-5 Example of Undesirable Assignment



CPU core ID specification

To assign or delete a named core by specifying a CPU core ID (CID) for a logical domain using the `ldm` command, you need to set the CID while understanding its relationship with the physical location of the CPU core.

Note - If you specify a CID to assign a CPU core to a logical domain, you cannot use the following functions. Make the settings after sufficiently understanding the content.

- Dynamic reconfiguration of CPUs
 - Automatic replacement function for CPUs
 - CPU Dynamic Resource Management (DRM)
-

In Oracle VM Server for SPARC, a core ID (CID) is determined by the physical CPU number (PID) and the number of threads.

The following figures show the physical locations of the CPUs in each model of the SPARC M12/M10.

Figure 2-6 SPARC M12-1 CPU Locations

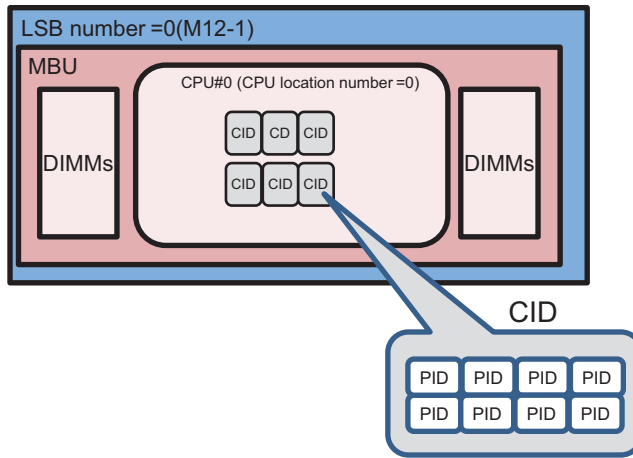


Figure 2-7 CPU Locations of the SPARC M12-2 or SPARC M12-2S (1 CPU Installed)

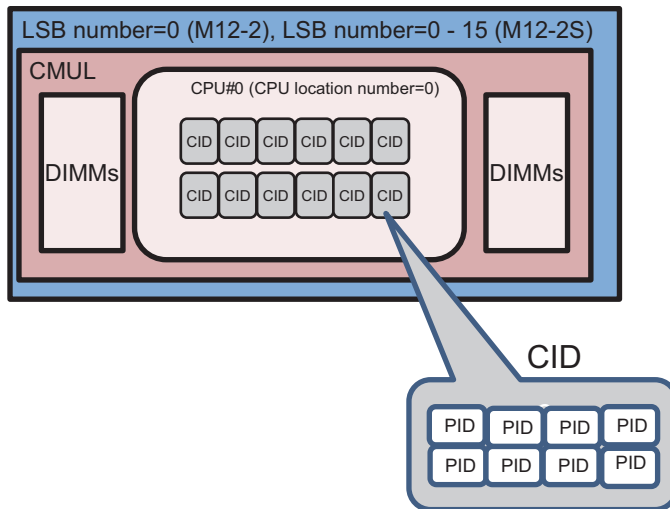


Figure 2-8 CPU Locations of the SPARC M12-2 or SPARC M12-2S (2 CPU Installed)

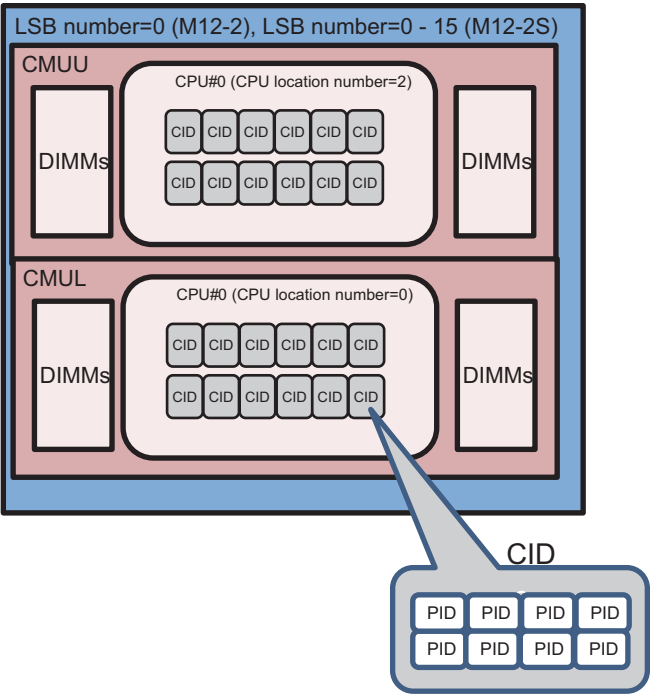


Figure 2-9 SPARC M10-1 CPU Locations

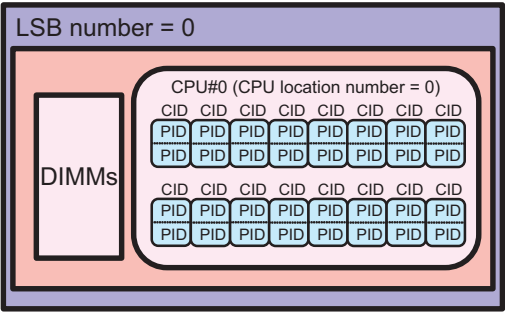


Figure 2-10 CPU Locations of the SPARC M10-4 or SPARC M10-4S (2 CPUs Installed)

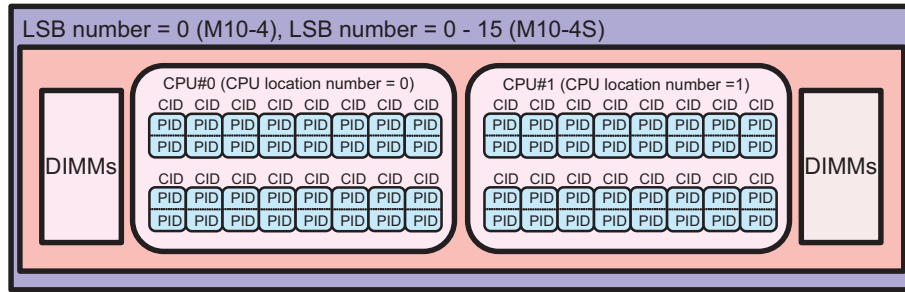
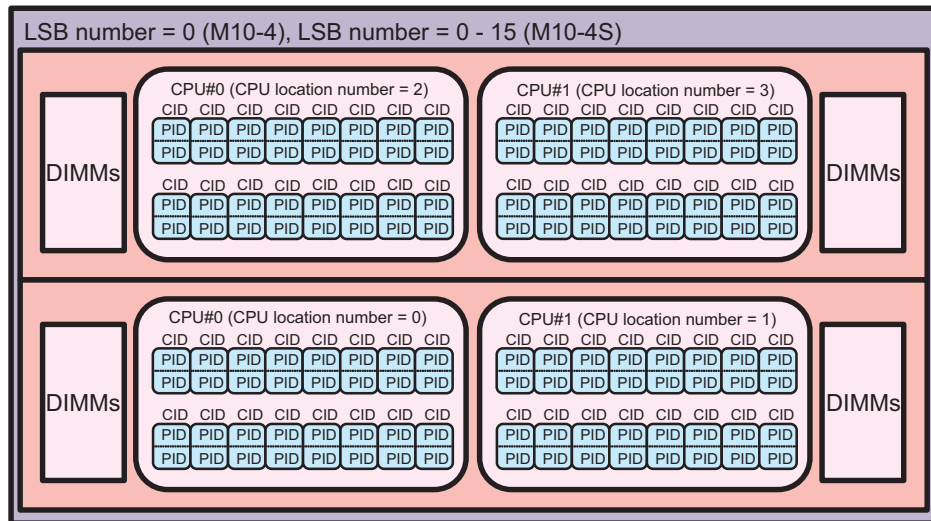


Figure 2-11 CPU Locations of the SPARC M10-4 or SPARC M10-4S (4 CPUs Installed)



For the SPARC M12/M10 system, the relationship between the physical location of a CPU and a CID is as follows.

In this section, as a parameter that indicates the location of the CPU, a CPU serial number on the logical system board (LSB) is called a CPU location number.

$$\begin{aligned} \text{PID} &= \text{LSB number} \times 1024 + \text{CPU location number} \times 256 + \text{CPU core number} \times 8 + \text{thread number} \\ \text{CID} &= \text{PID} / \text{number of threads} \end{aligned}$$

The range of each number used in the above formula is shown below.

Table 2-5 Range of Each Number Indicating a CPU Physical Location (SPARC M12)

Number Type	Range
LSB number	0 to 15 (*1)
CPU location number	0, 2 (*2)
CPU core number	0, 1, 2, 4, 5, 6, 8, 9, 10, 12, 13, 14 (0, 1, 2, 4, 5, 6) (*3)
Thread number	0 to 7
Number of threads	8

*1 For SPARC M12-1/M12-2, the LSB number is fixed to 0. For SPARC M12-2S, the LSB number can be set with the setpcl command of the XSCF firmware.

*2 When one CPU is installed in SPARC M12-1/M12-2 or SPARC M12-2S, the CPU location number is 0. When two CPUs are installed, the CPU#0 for CMUL is 0, and the CPU#0 for CMUU is 2.

*3 The figures in parentheses are the CPU core numbers of the SPARC M12-1.

Table 2-6 Range of Each Number Indicating a CPU Physical Location (SPARC M10)

Number Type	Range
LSB number	0 to 15 (*1)
CPU location number	0 to 3 (*2)
CPU core number	0 to 15 (0 to 7) (*3)
Thread number	0 to 1
Number of threads	2

*1 For SPARC M10-1 and M10-4, the LSB number is fixed to 0. For SPARC M10-4S, the LSB number can be set with the setpcl command of the XSCF firmware.

*2 When two CPUs are installed in SPARC M10-4 or M10-4S, the CPU location numbers are 0 to 1. When four CPUs are installed, the CPU location numbers are as follows: CMUL CPU#0 and #1 are 0 and 1, respectively, and CMUU CPU#0 and #1 are 2 and 3, respectively.

*3 The figures in parentheses are the CPU core numbers when SPARC64 X+ 8-core 3.7 GHz processors are mounted in SPARC M10-1/M10-4.

Accordingly, the relationship between the CPU core ID and physical location of each model of the SPARC M12 system is as shown in [Table 2-7](#).

The relationship between the CPU core ID and physical location of each model of the SPARC M10 system is as shown in [Table 2-8](#).

Table 2-7 List of CPU Core IDs of Each Model (SPARC M12)

Model	LSB Number	CPU Location Number	PID	CID
SPARC M12-1	0	0	0 to 55	0 to 6
SPARC M12-2	0	0	0 to 119	0 to 14
		2	512 to 631	64 to 78
SPARC M12-2S	0	0	0 to 119	0 to 14
		2	512 to 631	64 to 78
	1	0	1024 to 1143	128 to 142

Table 2-7 List of CPU Core IDs of Each Model (SPARC M12) (*continued*)

Model	LSB Number	CPU Location Number	PID	CID
		2	1536 to 1655	192 to 206
	2	0	2048 to 2167	256 to 270
		2	2560 to 2679	320 to 334
	3	0	3072 to 3191	384 to 398
		2	3584 to 3703	448 to 462
	4	0	4096 to 4215	512 to 526
		2	4608 to 4727	576 to 590
	5	0	5120 to 5239	640 to 654
		2	5632 to 5751	704 to 718
	6	0	6144 to 6263	768 to 782
		2	6656 to 6775	832 to 846
	7	0	7168 to 7287	896 to 910
		2	7680 to 7799	960 to 974
	8	0	8192 to 8311	1024 to 1038
		2	8704 to 8823	1088 to 1102
	9	0	9216 to 9335	1152 to 1166
		2	9728 to 9847	1216 to 1230
	10	0	10240 to 10359	1280 to 1294
		2	10752 to 10871	1344 to 1358
	11	0	11264 to 11383	1408 to 1422
		2	11776 to 11895	1472 to 1486
	12	0	12288 to 12407	1536 to 1550
		2	12800 to 12919	1600 to 1614
	13	0	13312 to 13431	1664 to 1678
		2	13824 to 13943	1728 to 1742
	14	0	14336 to 14455	1792 to 1806
		2	14848 to 14967	1856 to 1870
	15	0	15360 to 15479	1920 to 1934
		2	15872 to 15991	1984 to 1998

Table 2-8 List of CPU Core IDs of Each Model (SPARC M10)

Model	LSB Number	CPU Location Number	PID	CID
SPARC M10-1	0	0	0 to 121 (0 to 57) (*1)	0 to 60 (0 to 28) (*1)
SPARC M10-4	0	0	0 to 121 (0 to 57) (*1)	0 to 60 (0 to 28) (*1)
		1	256 to 377 (256 to 313) (*1)	128 to 188 (128 to 156) (*1)
		2	512 to 633 (512 to 569) (*1)	256 to 316 (256 to 284) (*1)
		3	768 to 889 (768 to 825) (*1)	384 to 444 (384 to 412) (*1)
SPARC M10-4S	0	0	0 to 121	0 to 60
		1	256 to 377	128 to 188
		2	512 to 633	256 to 316
		3	768 to 889	384 to 444
SPARC M10-4S	1	0	1024 to 1145	512 to 572
		1	1280 to 1401	640 to 700
		2	1536 to 1657	768 to 828
		3	1792 to 1913	896 to 956
SPARC M10-4S	2	0	2048 to 2169	1024 to 1084
		1	2304 to 2425	1152 to 1212
		2	2560 to 2681	1280 to 1340
		3	2816 to 2937	1408 to 1468
SPARC M10-4S	3	0	3072 to 3193	1536 to 1596
		1	3328 to 3449	1664 to 1724
		2	3584 to 3705	1792 to 1852
		3	3840 to 3961	1920 to 1980
SPARC M10-4S	4	0	4096 to 4217	2048 to 2108
		1	4352 to 4473	2176 to 2236
		2	4608 to 4729	2304 to 2364
		3	4864 to 4985	2432 to 2492
SPARC M10-4S	5	0	5120 to 5241	2560 to 2620
		1	5376 to 5497	2688 to 2748
		2	5632 to 5753	2816 to 2876
		3	5888 to 6009	2944 to 3004
SPARC M10-4S	6	0	6144 to 6265	3072 to 3132
		1	6400 to 6521	3200 to 3260
		2	6656 to 6777	3328 to 3388
		3	6912 to 7033	3456 to 3516
SPARC M10-4S	7	0	7168 to 7289	3584 to 3644
		1	7424 to 7545	3712 to 3772
		2	7680 to 7801	3840 to 3900
		3	7936 to 8057	3968 to 4028
SPARC M10-4S	8	0	8192 to 8313	4096 to 4156
		1	8448 to 8569	4224 to 4284
		2	8704 to 8825	4352 to 4412
		3	8960 to 9081	4480 to 4540

Table 2-8 List of CPU Core IDs of Each Model (SPARC M10) (*continued*)

Model	LSB Number	CPU Location Number	PID	CID
SPARC M10-4S	9	0	9216 to 9337	4608 to 4668
		1	9472 to 9593	4736 to 4796
		2	9728 to 9849	4864 to 4924
		3	9984 to 10105	4992 to 5052
SPARC M10-4S	10	0	10240 to 10361	5120 to 5180
		1	10496 to 10617	5248 to 5308
		2	10752 to 10873	5376 to 5436
		3	11008 to 11129	5504 to 5564
SPARC M10-4S	11	0	11264 to 11385	5632 to 5692
		1	11520 to 11641	5760 to 5820
		2	11776 to 11897	5888 to 5948
		3	12032 to 12153	6016 to 6076
SPARC M10-4S	12	0	12288 to 12409	6144 to 6204
		1	12544 to 12665	6272 to 6332
		2	12800 to 12921	6400 to 6460
		3	13056 to 13177	6528 to 6588
SPARC M10-4S	13	0	13312 to 13433	6656 to 6716
		1	13568 to 13689	6784 to 6844
		2	13824 to 13945	6912 to 6972
		3	14080 to 14201	7040 to 7100
SPARC M10-4S	14	0	14336 to 14457	7168 to 7228
		1	14592 to 14713	7296 to 7356
		2	14848 to 14969	7424 to 7484
		3	15104 to 15225	7552 to 7612
SPARC M10-4S	15	0	15360 to 15481	7680 to 7740
		1	15616 to 15737	7808 to 7868
		2	15872 to 15993	7936 to 7996
		3	16128 to 16249	8064 to 8124

*1 The figures in parentheses are the pid and cid when SPARC64 X+ 8-core 3.7 GHz processors are mounted in the SPARC M10-1/M10-4.

To check the CID and PID of the CPU not assigned to the logical domain, use the `ldm list-devices core` command.

The following example executes the `ldm list-devices core` command.

[ID] indicates the CPU core ID (CID), and [CPUSET] indicates the physical CPU number (PID) of the CPU associated with the CPU core ID.

# <code>ldm list-devices core</code>		
CORE		
ID	%FREE	CPUSET
920	100	(1840, 1841)
924	100	(1848, 1849)
936	100	(1872, 1873)
940	100	(1880, 1881)
944	100	(1888, 1889)

<u>948</u>	100	(1896, 1897)
CID		PID

To check the CID and PID of the CPU assigned to the logical domain, use the `ldm list-domain -o core` command.

The following example shows the CID and PID of the CPU core assigned to the control domain.

```
# ldm list-domain -o core primary
NAME
primary

CORE
  CID      CPUSSET
  0        (0, 1)
  4        (8, 9)
  8        (16, 17)
--- Omitted ---
  896      (1792, 1793)
  900      (1800, 1801)
  904      (1808, 1809)
  908      (1816, 1817)
  912      (1824, 1825)
  916      (1832, 1833)
  CID      PID
```

Memory assignment guideline for the logical domain

Assign 4 GB or more of virtual memory to the logical domain.

Memory size of a logical domain

If dynamic reconfiguration is not used to change the memory of the logical domain, the size of memory to be assigned to the logical domain can be set in units of 4 MB. If you need to use dynamic reconfiguration to change the memory of the logical domain, the size of memory to be assigned to the logical domain must be set in units of 256 MB.

Saving of logical domain configuration information

After the logical domain configuration is complete, save the logical domain configuration information to the XSCF. Also save the logical domain configuration information before performing work related to configuring the logical domain. Saving the configuration information before the work is performed ensures that the logical domain can be restored without fail to its pre-configuration state.

A hardware failure or the like may cause a loss of logical domain configuration information. Therefore, after changing guest domain configuration information, use the `ldm` command of Logical Domains Manager, which is management software for Oracle VM Server for SPARC, to acquire the guest domain configuration information, and keep it in a safe location. Replicate the acquired configuration information to a tape device or file server to protect against data loss from the disk. Using the `ldm`

command, you can also reconfigure a guest domain based on the acquired configuration information. For details, see the *Oracle VM Server for SPARC Administration Guide* and *Oracle VM Server for SPARC Reference Manual* of the version used.

You can back up the configuration information saved on the XSCF. To back up the configuration information to a file, use the `dumpconfig` command of the XSCF firmware. To restore the configuration information from the backup file, use the `restoreconfig` command.

Specified items when configuring a logical domain

To configure a logical domain, you need to define various names and specify various resources and their numbers. The following list shows items that you should define or specify when configuring a logical domain. Determine these items before starting work.

- Name of the virtual console terminal collection and distribution unit
- Virtual disk server name
- Virtual switch service name
- Virtual network device interface name
- Virtual device name
- Virtual disk name
- Guest domain name
- Name for logical domain configuration information
- Range for virtual console terminal port numbers
- Device used with the virtual switch service
- Device used with the virtual disk service
- Number of CPUs assigned to a guest domain
- Size of memory assigned to a guest domain
- Port number of the virtual console terminal assigned to a guest domain

Order of shutdown of logical domains

To use the ordered shutdown function and specify the shutdown order of each logical domain with the `poweroff` command of the XSCF, set a shutdown group for the created guest domain. Shutdown groups specify the order of shutdown of logical domains.

The shutdown groups range from 0 to 15. The group with a larger number shuts down earlier. Guest domains belong to group 15 by default. The control domain always belongs to group 0, and this cannot be changed. Set the shutdown order by considering the dependency of logical domains.

2.4.2 Consideration of Logical Domain Reconfiguration

You need to note the following points when reconfiguring hardware resources assigned to a logical domain.

Preparation on the device side

Before reconfiguring a virtual disk and virtual network among virtual I/O devices, perform the following work:

- Unmounting the virtual disk
- Executing unplumb for the virtual network interface

Delayed reconfiguration for the control and root domains

To statically add a physical I/O to or delete it from the control and root domains, operate it through delayed reconfiguration. Once you reconfigure physical I/Os assigned to the target control and root domains through delayed reconfiguration, any other reconfiguration requests for the control and root domains are postponed until the control and root domains are restarted.

Limit on delayed reconfiguration

You cannot reconfigure the hardware resources for a guest domain when delayed reconfiguration mode is set for the control or root domain to reconfigure hardware resources. Perform reconfiguration work for the guest domain after finishing the reconfiguration work on the control or root domain.

Cancellation of delayed reconfiguration

Use the `ldm cancel-operation reconf` command to cancel a delayed reconfiguration operation on the control or root domain. For details of the `ldm` command, see the *Oracle VM Server for SPARC Reference Manual*.

Logical domain requirements

- When dynamically reconfiguring a physical partition, ensure that Oracle Solaris on the control domain has already started.
- The disk that configures the system volume on the control domain must be in a redundant configuration. The redundant configuration is required with the disk on both the system board deleted through dynamic reconfiguration and the system board not deleted through dynamic reconfiguration. If the disk is not in a redundant configuration, the control domain cannot be started when the I/O devices have been released in advance to delete the system board where the disk is mounted.

Memory size assigned to the hypervisor

The amount of memory that can be used with the logical domain is less than the amount of memory actually installed in the chassis. This is because the memory used for hypervisor operation is reserved on each chassis. At system configuration, design

the domain configuration by subtracting the memory used by this hypervisor. The amount of memory assigned to the hypervisor depends on the model, building block configuration, and PPAR-DR mode setting. For details on PPAR DR mode, see ["2.5.3 How to Check/Set PPAR DR Mode."](#)

The hypervisor memory assigned to each physical partition is as follows:

Table 2-9 Amount of Memory Assigned to the Hypervisor

Model	PPAR DR Mode	
	Enabled	Disabled
SPARC M12-1/M12-2	2.5 GB	
SPARC M12-2S	2.5 GB + (1.5 GB x (number of chassis comprising the physical partition - 1))	
SPARC M10-1/M10-4	-	2 GB
SPARC M10-4S	2 GB + (1.25 GB x (number of chassis - 1))	2 GB + (1.0 GB x (number of chassis - 1))

Placement of CPU cores and memory

To enable disconnection of the system board with physical partition dynamic reconfiguration, be careful when determining the number of CPU cores to be assigned, as well as the memory placement. For details, see ["Placement of CPU cores and memory"](#) in ["2.5.2 Considerations in System Operation for Dynamic Reconfiguration."](#)

2.5 Dynamic Reconfiguration Conditions and Settings

2.5.1 Considerations When Configuring the System for Dynamic Reconfiguration

This section describes the points that must be considered when configuring the system, using the physical partition dynamic reconfiguration (PPAR DR) function. The dynamic reconfiguration function is supported only by the SPARC M12-2S/M10-4S.

Updating the XCP firmware

The physical partition dynamic reconfiguration (PPAR DR) function is supported by XCP 2220 and later. For details on how to update the XCP firmware, see ["Chapter 16 Updating the XCP Firmware"](#) in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC*

Checking PPAR DR mode

To use the physical partition dynamic reconfiguration (PPAR DR) function, PPAR DR mode must be enabled. This is set using the `setpparmode` command of the XSCF firmware. For details on how to check the setting of PPAR DR mode, see ["2.5.3 How to Check/Set PPAR DR Mode."](#)

Requirements for device drivers and I/O devices

To use the physical partition dynamic reconfiguration (PPAR DR) function, all the PCIe cards and I/O device interfaces mounted on the system board to be subject to physical partition dynamic reconfiguration must support PPAR DR. Moreover, all the installed I/O device drivers and duplicated software must support physical partition dynamic reconfiguration. For details on the PCIe card, driver, and duplicated software that support PPAR DR, see "Cards That Support PCI Hot Plug and Dynamic Reconfiguration" in the *PCI Card Installation Guide* for your server.

If all the PCIe cards and I/O devices installed on the system board to be subject to PPAR DR do not support PPAR DR, physical partition dynamic reconfiguration cannot be performed for that system board.

Software requirements

If the physical partition dynamic reconfiguration (PPAR DR) function is used, the logical domain may temporarily stop (suspend). Therefore, the software needs to take actions such as changing the timeout detection time and temporary stopping timeout monitoring. These software products include the software that detects an error with a timer or the like, such as the cluster software and system monitoring software.

Memory size assigned to the hypervisor

The amount of memory that can be used with the logical domain is less than the amount of memory actually installed in the chassis. This is because the memory used for hypervisor operation is reserved on each chassis. At system configuration, design the domain configuration by subtracting the memory used by this hypervisor. The amount of memory assigned to the hypervisor depends on the model, building block configuration, and PPAR-DR mode setting. For details on PPAR DR mode, see ["2.5.3 How to Check/Set PPAR DR Mode."](#)

For details on the hypervisor memory assigned to each physical partition, see [Table 2-9](#).

Setting unit of memory size

To enable physical partition dynamic reconfiguration when the version of Oracle VM Server for SPARC is earlier than 3.2, set the memory size to be assigned to the logical domain to a multiple of "number of cores assigned to the logical domain x 256 MB". For details on examples of the setting procedure, see ["Appendix A Environment Configuration Examples and Procedures Using Physical Partition Dynamic Reconfiguration."](#)

Placement of CPU cores and memory

To enable the disconnection of the SPARC M12-2S/M10-4S with physical partition dynamic reconfiguration, be careful when determining the number of CPU cores to be assigned, as well as the memory placement. For details, see "2.5.2 Considerations in System Operation for Dynamic Reconfiguration," "Placement of CPU cores and memory."

Time synchronization

When performing physical partition dynamic reconfiguration, the system time of a logical domain may be delayed. Before using the physical partition dynamic reconfiguration (PPAR DR) function, synchronize the time by using NTP (Network Time Protocol).

For details on the setting of the Oracle Solaris NTP server, see the *Oracle Solaris System Management (Network Service)* (Oracle Solaris 10) or *Oracle Solaris 11 Network Service Introduction* (Oracle Solaris 11).

Dynamic addition/deletion of PCIe root complex

In the case of the SPARC M10, you can add or delete a PCIe root complex without stopping Oracle Solaris on the root domain running XCP 2240 or later along with Oracle VM Server for SPARC 3.2 or later. In this case, you can add or delete only the root domain with Oracle Solaris 11.2 SRU11.2.8.4.0 or later installed.

Therefore, to add/delete a PCIe root complex when it is configured with only the control domains of Oracle Solaris 10, it is necessary to restart Oracle Solaris after delayed reconfiguration. So, dynamic reconfiguration of the physical partition is not available.

For details on this function, see "Assigning a Dynamic PCIe Bus" in the *Oracle VM Server for SPARC Administration Guide* for Oracle VM Server for SPARC 3.2 or later.

To configure a system using dynamic reconfiguration of physical partitions, the document including further detailed considerations and best practices has been posted to the Fujitsu SPARC servers Documentations site.

<https://www.fujitsu.com/global/products/computing/servers/unix/sparc/downloads/documents/index.html>

Refer to "1.2 Overview of PPAR DR" and "Appendix.A PPAR DR deleteboard Best Practice" in the *Building High Availability System on Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Servers (Maintenance procedure)*.

2.5.2 Considerations in System Operation for Dynamic Reconfiguration

This section describes points of consideration in system operation for dynamic reconfiguration.

About a logical domain and a kernel zone that have Oracle Solaris 11.3 or later installed

- If Oracle Solaris 11.3 is installed on the control domain, reboot the SPARC M10-4S

system after adding the following line to the /etc/system file in the control domain in advance. However, this procedure is unnecessary if the XCP firmware is XCP 2260 or later.

```
set enable_user_tick_stick_emulation = 0
```

- If a logical domain and an Oracle Solaris kernel zone (excluding the control domain) on which Oracle Solaris 11.3 or later is installed exist in the physical partition configured for the SPARC M12-2S/M10-4S, reboot the logical domain and Oracle kernel zone after adding the following line to the /etc/system files in the applicable logical domain and the Oracle Solaris kernel zone.

```
set uhrt_enable = 0x0
```

Suspend time of Oracle Solaris during deleteboard command execution

When the deleteboard command is executed, Oracle Solaris on the logical domains may temporarily stop (suspend). During this time, physical device communication and I/Os may stop, and the operation of the applications running on Oracle Solaris may also stop, which may have an impact on the business processes, such as shutting down the network with a remote unit. It is therefore necessary to determine the operations and other matters by determining, in advance, the time at which each logical domain may stop (suspend time).

The maximum suspend time is that for the suspend/resume processing caused by memory movement and the time required for suspend/resume processing for each I/O device.

An approximate value can be calculated based on the memory mounted in the chassis on which the deleteboard command will be executed, as well as the types and number of physical I/O devices mounted in the physical partition.

This is calculated using the following formula.

Suspend time = Memory movement time + Sum of suspend/resume times for on-board devices + Sum of suspend/resume times for PCI cards

The suspend time due to memory movement is 168 seconds per TB. For the calculation, check the amount of memory mounted in the chassis in which the deleteboard command is to be executed, using the showpparinfo command.

In the output example shown below, the suspend time is 21 seconds because the install size value for PSB 01-0 is 128 GB, which is displayed under "Memory:", if the system board (BB#01) is disconnected.

$$128 \text{ (GB)} \times 168 \text{ (seconds)/1024 (GB)} = 21 \text{ (seconds)}$$

```
XSCF> showpparinfo -p 0
PPAR#00 Information:
-----
....
Memory:
-----
install
```

```
PID PSB size GB
00 00-0 128
00 01-0 128
IO Devices:
-----
PID PSB device
....
```

Calculate the suspend time for physical I/Os from the types and number of all physical I/O devices mounted in the partition. The suspend/resume processing time for an on-board device is as indicated below.

- For SPARC M12,
5.2 seconds per chassis
Example) 10.4 seconds for 2BB configuration
- For SPARC M10,
21.4 seconds per chassis
Example) 42.8 seconds for 2BB configuration

For details on the suspend/resume processing times for PCI cards, see "Cards That Support PCI Hot Plug and Dynamic Reconfiguration" in the *PCI Card Installation Guide* for your server.

Combination with live migration

Do not perform live migration and physical partition dynamic reconfiguration at the same time.

Placement of CPU cores and memory

If the CPU or memory configuration does not satisfy the following conditions, executing the deleteboard command may cause the following message to be displayed. Take action while referring to each condition.

```
Some domain will be reduced with DR operation. But reducing
resource is not allowed.
```

- When configured with only the control domain:
If you use the deleteboard command to delete the system board through physical partition dynamic reconfiguration and both of the following cases are applicable, the above message will be produced and this operation will end abnormally in error:- When the logical domain configuration information is configured with only the control domain, such as factory-default, and- When the CPU cores and memory region on multiple system boards are assigned to the control domain.

To prevent this, perform the following:

- Specify the -m unbind=resource option in the deleteboard command to dynamically delete the CPU cores and memory region.
- Delete the CPU and memory in advance with the ldm remove-core or ldm remove-memory command to obtain the free resource space for the BB (PSB) to be removed.

To execute PPAR DR without reducing the number of CPU cores or amount of memory used with the control domain, it is necessary to set the configuration so that the free space is obtained in advance for the number of CPU cores and the memory region size assigned to the domain.

The `-m unbind=resource` option of the `deleteboard` command is supported in Oracle VM Server for SPARC 3.2 or later. The PPAR DR policy is supported in Oracle VM Server for SPARC 3.4 or later. For details on the policy, see "8.15 Setting the Physical Partition Dynamic Reconfiguration Policy" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

- When configured with the control domain as well as the logical domain:

[CPU cores]

If the number of CPU cores assigned to a logical domain would exceed the number of CPU cores remaining after deletion by the `deleteboard` command, the `deleteboard` command will fail. This is because the number of unassigned CPU cores, which would be the destination for the assigned CPU cores, will be insufficient.

To prevent this, perform the following:

- Configure the logical domain in advance while leaving as many unassigned CPU cores as the number of CPU cores to be deleted with the `deleteboard` command.
- Specify the `-m unbind=resource` option in the `deleteboard` command to dynamically delete the CPU cores and memory region.
- Delete the CPU cores and memory in advance with the `ldm remove-core` or `ldm remove-memory` command to obtain the resource free space for the system board to be removed.

The `-m unbind=resource` option of the `deleteboard` command is supported in Oracle VM Server for SPARC 3.2 or later. The PPAR DR policy is supported in Oracle VM Server for SPARC 3.4 or later. For details on the policy, see "8.15 Setting the Physical Partition Dynamic Reconfiguration Policy" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

Check the assignment status of the number of CPU cores, as follows:

1. Check the total number of CPU cores that are assigned to each logical domain.

This is the total number of CPU cores whose "%FREE" field contains a value other than "100". You can check this number by executing the `ldm list-devices -a core` command on the control domain. The following example executes the `ldm list-devices -a -p core` command to display the total number of CPU cores assigned to a logical domain.

```
# ldm list-devices -a core
CORE
  ID      %FREE  CPUSSET
  0        0      (0, 1)
  4        0      (8, 9)
  8        0     (16, 17)
  12       0     (24, 25)
(Omitted)
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
112
```

2. Check the total number of CPU cores on the system board that have not been removed.

This is the total number of "cores" for the PSB numbers that have not been deleted. You can check this number by executing the showpparinfo command on the XSCF. The following example executes the showpparinfo command for the SPARC M10-4S.

```
XSCF> showpparinfo -p 0
PPAR#00 Information:
-----
CPU(s) : 8
CPU Cores : 128
CPU Threads : 256
Memory size (GB) : 256
CoD Assigned (Cores) : 256
CPU(s) :
-----
PID PSB CPU# Cores Threads
00 00-0 0 16 32
00 00-0 1 16 32
00 00-0 2 16 32
00 00-0 3 16 32
00 01-0 0 16 32
00 01-0 1 16 32
00 01-0 2 16 32
00 01-0 3 16 32
(Omitted)
```

Using the formula below, calculate the CPU core shortfall that will occur after releasing the SPARC M12-2S/M10-4S chassis.

CPU core shortfall = Number of cores used in logical domain (1) - Number of physical cores after release (2)

If there are not enough cores, you must reduce the number of cores by deleting those CPU cores assigned to a logical domain with the ldm remove-core command.

[Memory]

Suppose you executed the deleteboard command through physical partition dynamic reconfiguration when the memory region on the system board to be deleted is assigned to a logical domain. In this case, the contents of the memory region assigned to the logical domain are reassigned to the memory region on the system board that has not been deleted, in order to move the contents.

Therefore, if there is no available space that is larger than the amount of memory that has been moved and which exists as an unused memory region on the destination (on the undeleted system board), the deleteboard command will end abnormally with an error.

To prevent this, perform the following:

- Specify the -m unbind=resource option in the deleteboard command to dynamically delete the CPU cores and memory region.
- Delete the memory in advance with the ldm remove-memory command to obtain sufficient resource free space for the system board to be removed.

The -m unbind=resource option of the deleteboard command is supported in Oracle VM Server for SPARC 3.2 or later. The PPAR DR policy is supported in Oracle VM Server for SPARC 3.4 or later. For details on the policy, see "8.15 Setting the Physical Partition Dynamic Reconfiguration Policy" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

Check the memory region use status, as follows:

1. Check the use status of the continuous region of the memory (memory block).

Execute the prtdiag command to check the correspondence between the physical addresses of the memory and the SPARC M10-4S having a building block configuration.

```
# prtdiag
(Omitted)
===== Physical Memory Configuration =====
Segment Table:
-----
Base Segment Interleave Bank Contains
Address Size Factor Size Modules
-----
0x7e0000000000 32 GB 4 8 GB /BB0/CMUL/CMP0/MEM00A
(Omitted)
0x7c0000000000 32 GB 4 8 GB /BB0/CMUL/CMP1/MEM10A
(Omitted)
0x7a0000000000 32 GB 4 8 GB /BB0/CMUU/CMP0/MEM00A
(Omitted)
0x780000000000 32 GB 4 8 GB /BB0/CMUU/CMP1/MEM10A
(Omitted)
0x760000000000 32 GB 4 8 GB /BB1/CMUL/CMP0/MEM00A
```

```

(Omitted)
0x740000000000 32 GB 4 8 GB /BB1/CMUL/CMP1/MEM10A
(Omitted)
0x720000000000 32 GB 4 8 GB /BB1/CMUU/CMP0/MEM00A
(Omitted)
0x700000000000 32 GB 4 8 GB /BB1/CMUU/CMP1/MEM10A
(Omitted)

```

The result of this example is rearranged in ascending order of physical addresses in memory. The following table lists the correspondence between the physical addresses and the SPARC M10-4S.

Table 2-10 Example of Correspondence Between Physical Addresses and the SPARC M10-4S

Base Address (Physical Address)	SPARC M10-4S
0x700000000000...	Building block BB-ID#01
0x720000000000...	Building block BB-ID#01
0x740000000000...	Building block BB-ID#01
0x760000000000...	Building block BB-ID#01
0x780000000000...	Building block BB-ID#00
0x7a0000000000...	Building block BB-ID#00
0x7c0000000000...	Building block BB-ID#00
0x7e0000000000...	Building block BB-ID#00

Next, execute the `ldm list-devices -a memory` command on the control domain to display the memory regions assigned to each logical domain, as well as any unused memory regions.

```

# ldm list-devices -a memory
MEMORY
PA SIZE BOUND
0x700000000000 24G root-dom1
0x700600000000 8G
0x720000000000 32G guest0
0x740000000000 32G guest1
0x760000800000 1272M _sys_
0x760050000000 24G root-dom0
0x760650000000 6912M
0x780000000000 32G
0x7a0000000000 32G
0x7c0000000000 32G
0x7e0000800000 1272M _sys_
0x7e0050000000 512M _sys_
0x7e0070000000 256M _sys_
0x7e0080000000 14G primary
0x7e0400000000 16G

```

From the above results and physical locations in "[Table 2-10 Example of Correspondence Between Physical Addresses and the SPARC M10-4S](#)," you can determine the memory block usage status as shown below.

Table 2-11 Example of Memory Block Use Statuses

SPARC M10-4S	Physical Address	Size	Logical Domain
Building block BB-ID#01 (to be replaced)	0x700000000000	24 GB	root-dom1
	0x700600000000	8 GB	Unassigned
	0x720000000000	32 GB	guest0
	0x740000000000	32 GB	guest1
	0x760050000000	24 GB	root-dom0
	0x760650000000	6912 MB	Unassigned
Building block BB-ID#00	0x780000000000	32 GB	Unassigned
	0x7a0000000000	32 GB	Unassigned
	0x7c0000000000	32 GB	Unassigned
	0x7e0080000000	14 GB	primary
	0x7e0400000000	16 GB	Unassigned

2. Check the size and quantity of the moved source memory blocks.

While referring to the check results for the memory block use status, check the memory blocks (hereinafter referred to as "source memory blocks") assigned to the SPARC M10-4S to be replaced.

In "[Table 2-11 Example of Memory Block Use Statuses](#)," you can determine that the number of memory blocks assigned to the logical domain is 32 GB x 2 (assignment to guest0 and guest1) and 24 GB x 1 (root-dom0) on the building block BB-ID#01 side.

Note - If root domain root-dom1 to which I/O of building block BB-ID#01 is assigned is unbound and placed in the inactive state before that building block is disconnected, you can exclude root-dom1 from movement.

3. Check the empty memory blocks.

Next, based on the check results obtained in step 1, check the memory blocks (hereinafter referred to as "empty memory blocks") not assigned to the logical domain on the SPARC M10-4S that is not disconnected.

For the example given in "[Table 2-11 Example of Memory Block Use Statuses](#)," it is possible to determine that the number of empty memory blocks is 32 GB x 3 and 16 GB x 1.

4. Check whether the memory block can be moved.

Using the check results obtained in steps 2 and 3, check whether the source memory block can be moved to the empty memory block.

This is possible if the size of the empty memory block is equal to or greater than that of the source memory block.

When the destination contains empty resources

For example, in "Table 2-11 Example of Memory Block Use Statuses," there are 32 GB x 3 empty memory blocks that are the destinations for guest0 (32 GB), guest1 (32 GB), and root-dom0 (24 GB). So, you can determine that the memory is placed such that building block BB-ID#01 can be disconnected. This is summarized in "Table 2-12 Destination Candidate Memory Blocks."

Table 2-12 Destination Candidate Memory Blocks

SPARC M10-4S	Size	Logical Domain	Destination Candidate
Building block BB-ID#01 (to be replaced)	24 GB	root-dom1	-
	8 GB	Unassigned	-
	32 GB	guest0	32 GB of building block BB-ID#00
	32 GB	guest1	32 GB of building block BB-ID#00
	24 GB	root-dom0	32 GB of building block BB-ID#00
	6912 MB	Unassigned	-
Building block BB-ID#00	32 GB	Unassigned	Moved here
	32 GB	Unassigned	Moved here
	32 GB	Unassigned	Moved here
	14 GB	primary	-
	16 GB	Unassigned	Excluded from destination candidates due to size insufficiency

When there are no empty resources at the destination

For example, in the configuration in "Table 2-13 Example of Placement of Memory Blocks With No Destinations," the number of source memory blocks is 32 GB x 2 and 24 GB x 2. Meanwhile, the free memory blocks at the destination are 32 GB x 3 and 16 GB x 1.

As a result, one memory block of 32 GB (guest0) and two memory blocks of 24 GB (two of guest1, guest2, and root-dom0) can be moved.

However, the number of remaining empty destination memory blocks is 16 GB x 1 and 8 GB x 2, obtained after the 24-GB memory block has been moved to a 32-GB memory block. So, any of guest1, guest2, and root-dom0, to which a 24-GB memory block is assigned, cannot be moved. In this case, you need to reduce the size of a possibly unmovable memory block in the logical domain to be equal to or less than the memory block size at the destination.

In the above example, you must execute the `ldm remove-memory` command to change any one of guest1, guest2, and root-dom0 from 24 GB to 16 GB or less.

Table 2-13 Example of Placement of Memory Blocks With No Destinations

SPARC M10-4S	Size	Logical Domain	Destination Candidate
Building block BB-ID#01 (to be replaced)	24 GB	guest2	May not be movable
	8 GB	root-dom1	-
	32 GB	guest0	32 GB of building block BB-ID#00
	32 GB	guest1	May not be movable
	24 GB	root-dom0	May not be movable
	6912 MB	Unassigned	-
Building block BB-ID#00	32 GB	Unassigned	Moved here
	32 GB	Unassigned	Any one of guest1, guest2, and root-dom0 (24 GB) moves here, with 8 GB remaining.
	32 GB	Unassigned	Any one of guest1, guest2, and root-dom0 (24 GB) moves here, with 8 GB remaining.
	14 GB	primary	-
	16 GB	Unassigned	Excluded from destination candidates due to size insufficiency

Dynamic reconfiguration operations when recovery mode is enabled

- Suppose that you add a system board using dynamic reconfiguration of physical partitions in the condition where the domain configuration has been recovered in the degraded configuration. The added resource is not allocated automatically to any logical domain. Allocate the added resource manually. Alternatively, execute the `ldm set-spcnfig` command to select the original domain configuration and then reboot the physical partition using the `poweron` and `poweroff` commands of XSCF firmware.
- Suppose that you delete a system board (PSB) using the `deleteboard` command while the physical partition (PPAR) is powered on after the domain configuration is recovered in a degraded configuration. This `deleteboard` command may fail if the version of Oracle VM Server for SPARC is earlier than 3.2. After a domain configuration is recovered in a degraded configuration, do not delete a system board using dynamic reconfiguration of physical partitions.

Combination with Oracle Solaris kernel zones

When an Oracle Solaris kernel zone is running on any logical domain in the physical

partition (PPAR), physical partition dynamic reconfiguration cannot be performed. Stop the Oracle Solaris kernel zone and then perform physical partition dynamic reconfiguration.

For the logical domain to which the virtual service is assigned

If the logical domain to which the virtual service is assigned is any of the following, it is necessary to remove the virtual disk server device (vdsdev) and the virtual network switch (vsw) assigned to the physical I/O, and the virtual disks (vdisk) and the virtual network (vnet) assigned to them by using the `ldm remove-vdsdev` command, the `ldm remove-vsw` command, the `ldm remove-vdisk` command, and the `ldm remove-vnet` command respectively beforehand.

- When a PCIe endpoint is deleted dynamically with the `ldm remove-io` command
- When the physical I/O of the logical domains is deleted dynamically by the dynamic reconfiguration of the physical partition

Changing the USB device path after a PPAR DR operation

If you delete a system board (PSB<BB>) and then add one using the physical partition dynamic reconfiguration function, the "usb@4,1" part of the USB device path for the deleted/added PSB (BB) changes to "usb@4", as shown in the following example.

- Example for an external DVD drive (front)

Before deletion: `/pci@8000/pci@4/pci@0/pci@1/pci@0/usb@4,1/hub@2/cdrom@1/disk@0,0:a`

After addition: `/pci@8000/pci@4/pci@0/pci@1/pci@0/usb@4/hub@2/cdrom@1/disk@0,0:a`

- Example for a DVD drive for remote storage

Before deletion: `/pci@8000/pci@4/pci@0/pci@1/pci@0/usb@4,1/storage@3/disk@0,0:a`

After addition: `/pci@8000/pci@4/pci@0/pci@1/pci@0/usb@4/storage@3/disk@0,0:a`

However, the device path returns to the state before deletion after the restart of the logical domain assigned the USB device whose device path was changed.

This phenomenon does not occur for USB devices connected to PSBs (BBs) that are not deleted/added using the physical partition dynamic reconfiguration function.

To configure a system using the dynamic reconfiguration of physical partitions, the document including more detailed considerations and best practices has been posted to the Fujitsu SPARC servers Documentations site.

<https://www.fujitsu.com/global/products/computing/servers/unix/sparc/downloads/documents/index.html>

Refer to "1.2 Overview of PPAR DR" and "Appendix.A PPAR DR deleteboard Best Practice" in the *Building High Availability System on Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 Servers (Maintenance procedure)*.

2.5.3 How to Check/Set PPAR DR Mode

This section describes how to check PPAR DR mode, and also how to set it.

Important - Before changing the PPAR DR mode setting, first save the logical domain configuration information.

Suppose that you change the PPAR DR mode setting from disabled to enabled, or enabled to

disabled, using the `setpparmode` command. To apply the setting, you must reset the physical partition. This reset returns the logical domain configuration information to the factory-default settings. Reconfiguring the logical domain requires the logical domain configuration information that has been saved previously. So, always save the data before changing the PPAR DR mode setting. For details on how to save the logical domain configuration information and reconfigure the logical domain, see the configuration examples given in "[A.1 Updating a System That Does Not Support Physical Partition Dynamic Reconfiguration to One That Does Support It.](#)"

The procedure for setting PPAR DR mode is as follows:

1. **Save the logical domain configuration information.**

In the following example, the logical domain configuration information is saved to an XML file called `file.xml`.

Note - This procedure is unnecessary if the logical domain has not yet been configured.

```
# ldm list-constraints -x > file.xml
```

2. **Execute the `showpparmode` command to check the PPAR DR mode setting.**

The display of PPAR DR (Current) indicates the current PPAR DR mode value. PPAR DR (Next) indicates the PPAR DR mode value set after the physical partition is next started.

In the following example, the power to the physical partition is turned off and the PPAR DR function is disabled when the physical partition is next started.

```
XSCF> showpparmode -p PPAR_ID
Host-ID                :9006000a
Diagnostic Level        :min
Message Level           :normal
Alive Check             :on
Watchdog Reaction       :reset
Break Signal            :off
Autoboot (Guest Domain) :on
Elastic Mode            :off
IOreconfigure           :false
PPAR DR (Current)       :-
PPAR DR (Next)          :off
```

3. **Execute the `setpparmode` command to set PPAR DR mode.**

The following example enables the PPAR DR function.

```
XSCF> setpparmode -p PPAR_ID -m ppar_dr=on
Diagnostic Level        :max      -> -
Message Level           :normal   -> -
Alive Check             :on       -> -
Watchdog Reaction       :reset    -> -
Break Signal            :on       -> -
Autoboot (Guest Domain) :on       -> -
```

```

Elastic Mode           :off          -> -
IOreconfigure          :true         -> -
PPAR DR                :off          -> on
The specified modes will be changed.
Continue? [y|n] :y
configured.
Diagnostic Level        :max
Message Level           :normal
Alive Check             :on (alive check:available)
Watchdog Reaction       :reset (watchdog reaction:reset)
Break Signal            :on (break signal:non-send)
Autoboot(Guest Domain) :on
Elastic Mode           :on
IOreconfigure          :false
PPAR DR                :on

```

4. Execute the **poweron** command to power on the physical partitions.

```
XSCF> poweron -p PPAR_ID
```

Note - The logical domain configuration information returns to the factory-default settings. After restarting the physical partitions, reconfigure the logical domains by using the logical domain configuration information saved in step 1. For details on how to reconfigure the logical domains with the configuration information, see the configuration examples given in ["A.1 Updating a System That Does Not Support Physical Partition Dynamic Reconfiguration to One That Does Support It."](#)

5. Execute the **showpparmode** command to check the current and next setting values.

The following example shows that the PPAR DR function is currently enabled and that it will remain enabled when the physical partition is next started.

```

XSCF> showpparmode -p PPAR_ID
Host-ID                :9006000a
Diagnostic Level        :min
Message Level           :normal
Alive Check            :on
Watchdog Reaction       :reset
Break Signal           :off
Autoboot(Guest Domain) :on
Elastic Mode           :off
IOreconfigure          :false
PPAR DR(Current)       :on
PPAR DR(Next)          :on

```

2.6 Considerations When Using the SPARC64 X+ Processor

This section describes the points that should be considered when configuring a system that includes the SPARC64 X+ processor.

Note - This applies to the SPARC M10. They are not applicable to the SPARC M12.

2.6.1 Updating the XCP firmware

For details on the versions of the XCP firmware and Oracle Solaris that support systems incorporating the SPARC64 X+ processor in SPARC M10, see "[1.5.4 Software Conditions and Confirmation Methods](#)" and the latest *SPARC M10 Systems Product Notes*.

2.6.2 CPU Operational Mode Settings

The SPARC64 X+ processor can provide the same functions as the SPARC64 X processor in SPARC M10. This section describes the settings and points to consider regarding the CPU operational mode.

CPU operational mode setting procedures

For details on the CPU operational mode, see "[1.6.1 CPU Operation Type and CPU Operational Mode](#)."

The CPU operational mode can be configured for each physical partition by executing the `setpparmode` command of the XSCF firmware.

There are two CPU operational modes (`cpumode`) that can be set with the `setpparmode` command: "auto" and "compatible". The CPU operational mode is set to "auto" by default.

Each mode operates as follows.

- auto mode
This mode enables the database acceleration instruction, an extension provided by the SPARC64 X+ processor. However, in a mixed configuration with both the SPARC64 X+ processor and the SPARC64 X processor, the SPARC64 X+ processor operates in SPARC64 X compatible mode. If you perform maintenance through dynamic reconfiguration in a mixed configuration with both the SPARC64 X+ processor and the SPARC64 X processor, set compatible mode.
- compatible mode
The SPARC64 X+ processor operates in SPARC64 X compatible mode. To configure a physical partition for the mixed use of the SPARC64 X+ processor and the SPARC64 X processor, set this mode.

Table 2-14 shows the relationship between the physical partition configuration, CPU operational mode settings, and CPU operation.

Table 2-14 CPU Configuration and CPU Operational Mode in a Physical Partition

Physical Partition Configuration	CPU Operational Mode Setting	CPU Operation
SPARC64 X+	auto	Operates with the SPARC64 X+ functions
SPARC64 X+	compatible	Operates in SPARC64 X compatible mode
SPARC64 X+/X	auto or compatible	SPARC64 X+ operates in SPARC64 X compatible mode
SPARC64 X	auto or compatible	Operates with the SPARC64 X functions

Change the CPU operational mode settings while the target physical partition is powered off.

Important - Before changing the CPU operational mode setting, first save (backup) the logical domain configuration information.

The logical domain configuration information in the Oracle VM Server for SPARC includes information related to the CPU operational mode used in the logical domain. If you apply the logical domain configuration information for a logical domain operating with the SPARC64 X+ function to a physical partition operating with the SPARC64 X function, a configuration information mismatch will occur. Then, XSCF switches the logical domain configuration information to the factory-default and starts the physical partition. To reconfigure the logical domain configuration information switched to the factory-default back to the original configuration settings, the logical domain configuration information that has been previously saved (backed up) is required. For these reasons, you must save (back up) the logical domain configuration information before changing the CPU operational mode with the `setpparmode` command. For details on how to save (back up) logical domain configuration information and reconfigure a logical domain, see the configuration examples given in "A.1 Updating a System That Does Not Support Physical Partition Dynamic Reconfiguration to One That Does Support It." For details on those cases in which a logical domain needs to be reconfigured, see "B.2 Supplementary Information on CPU Operational Mode."

To change the CPU operational mode, use the `setpparmode` command. The following shows an example of the procedures for changing the CPU operational mode to "compatible".

1. **Power off the physical partition.**

```
XSCF> poweroff -p PPAR_ID
```

2. **Execute the `showpparmode` command to check the current CPU operational mode (CPU mode) settings.**

```
XSCF> showpparmode -p PPAR_ID
Host-ID                :9007002b
Diagnostic Level        :min
Message Level           :normal
Alive Check             :on
Watchdog Reaction       :reset
Break Signal           :on
Autoboot (Guest Domain) :on
Elastic Mode            :off
IOreconfigure           :false
CPU Mode                :auto
PPAR DR (Current)       :-
PPAR DR (Next)         :on
```

3. **Execute the setpparmode command to change the CPU operational mode (CPU mode) from "auto" to "compatible".**

```
XSCF> setpparmode -p PPAR_ID -m cpumode=compatible
```

4. **Execute the showpparmode command to check that the CPU operational mode (CPU mode) has been set to "compatible".**

```
XSCF> showpparmode -p PPAR_ID
Host-ID                :9007002b
Diagnostic Level        :min
Message Level           :normal
Alive Check             :on
Watchdog Reaction       :reset
Break Signal           :on
Autoboot (Guest Domain) :on
Elastic Mode            :off
IOreconfigure           :false
CPU Mode                :compatible
PPAR DR (Current)       :-
PPAR DR (Next)         :on
```

5. **Power on the physical partition.**

```
XSCF> poweron -p PPAR_ID
```

Check the CPU operation type

To check the CPU operation type, execute the prtdiag or psrinfo command on Oracle Solaris.

When the CPU operates with the SPARC64 X+ function, the prtdiag command outputs "SPARC64-X+" in [Virtual CPUs]. When the CPU operates with the SPARC64 X function, the command outputs "SPARC64-X" in [Virtual CPUs].

Execute the psrinfo command with the -pv option specified. When the CPU operates

with the SPARC64 X+ function, "SPARC64-X+" is output in the provided physical processor information. Meanwhile, when the CPU operates with the SPARC64 X function, "SPARC64-X" is output.

The following shows display examples of the prtdiag and psrinfo commands.

- **When operating with the SPARC64 X+ function**

[Display example of prtdiag command]

```
# prtdiag
System Configuration:   Oracle Corporation   sun4v   SPARC   M10-4S
Memory size:  391168 Megabytes

===== Virtual CPUs =====

CPU ID Frequency Implementation      Status
-----
0      3700 MHz SPARC64-X+             on-line
:
```

[Display example for psrinfo command]

```
# psrinfo -pv
The physical processor has 16 cores and 32 virtual processors
(0-31)
  The core has 2 virtual processors (0 1)
  The core has 2 virtual processors (2 3)
  The core has 2 virtual processors (4 5)
  The core has 2 virtual processors (6 7)
  The core has 2 virtual processors (8 9)
  The core has 2 virtual processors (10 11)
  The core has 2 virtual processors (12 13)
  The core has 2 virtual processors (14 15)
  The core has 2 virtual processors (16 17)
  The core has 2 virtual processors (18 19)
  The core has 2 virtual processors (20 21)
  The core has 2 virtual processors (22 23)
  The core has 2 virtual processors (24 25)
  The core has 2 virtual processors (26 27)
  The core has 2 virtual processors (28 29)
  The core has 2 virtual processors (30 31)
    SPARC64-X+ (chipid 0, clock 3700 MHz)
  :
  :
```

- **When operating with the SPARC64 X function**

[Display example of prtdiag command]

```
# prtdiag
System Configuration:   Oracle Corporation   sun4v   SPARC   M10-4S
Memory size:  391168 Megabytes
```

```
===== Virtual CPUs =====
CPU ID Frequency Implementation      Status
-----
0      3700 MHz SPARC64-X             on-line
:
```

[Display example of psrinfo command]

```
# psrinfo -pv
The physical processor has 16 cores and 32 virtual processors
(0-31)
  The core has 2 virtual processors (0 1)
  The core has 2 virtual processors (2 3)
  The core has 2 virtual processors (4 5)
  The core has 2 virtual processors (6 7)
  The core has 2 virtual processors (8 9)
  The core has 2 virtual processors (10 11)
  The core has 2 virtual processors (12 13)
  The core has 2 virtual processors (14 15)
  The core has 2 virtual processors (16 17)
  The core has 2 virtual processors (18 19)
  The core has 2 virtual processors (20 21)
  The core has 2 virtual processors (22 23)
  The core has 2 virtual processors (24 25)
  The core has 2 virtual processors (26 27)
  The core has 2 virtual processors (28 29)
  The core has 2 virtual processors (30 31)
    SPARC64-X (chipid 0, clock 3700 MHz)
The physical processor has 16 cores and 32 virtual processors
(32-63)
  The core has 2 virtual processors (32 33)
```

2.6.3 Conditions for a Mixed Configuration With the SPARC64 X+ Processor and the SPARC64 X Processor Within a PPAR

With SPARC M10-4S, you can use the SPARC64 X+ processor mounted on one SPARC M10-4S and the SPARC64 X processor mounted on another SPARC M10-4S together in one physical partition. For a mixed configuration, you need to apply the XCP firmware that supports the SPARC64 X+ processor to all the SPARC M10-4S units comprising the building block. Depending on your system configuration, the XCP firmware that supports the SPARC64 X+ processor is automatically applied.

If you have a physical partition consisting of a SPARC M10-4S unit with a SPARC64 X processor and want to add a SPARC M10-4S with a SPARC64 X+ processor to this physical partition, you must update the XCP firmware on the SPARC M10-4S with the SPARC64 X processor beforehand.

If you have a physical partition consisting of a SPARC M10-4S unit with a SPARC64

X+ processor and want to add a SPARC M10-4S with a SPARC64 X processor to this physical partition, the XCP firmware on the SPARC M10-4S with the SPARC64 X processor will be automatically updated to the version of the XCP firmware applied to the master XSCF.

Note - For details on the versions of the XCP firmware that support systems incorporating the SPARC64 X+ processor, see the *Fujitsu SPARC M10 Systems Product Notes* for the latest XCP version.

2.6.4 Relationship Between the CPU Operational Mode and Physical Partition Dynamic Reconfiguration

This section describes the relationship between the CPU operational mode and the addition or deletion of a SPARC M10-4S through physical partition dynamic reconfiguration.

Relationship between the CPU operational mode and the addition through physical partition dynamic reconfiguration

A system board with the SPARC SPARC64 X+ processor can only be added to a physical partition operating with the SPARC SPARC64 X+ function in SPARC M10 through dynamic reconfiguration.
You cannot dynamically incorporate the SPARC64 X processor, which does not support the SPARC64 X+ function.
If you try to add a system board with the SPARC64 X processor through dynamic reconfiguration, the addition process fails with the following message.

The current configuration does not support this operation.

To a physical partition operating with the SPARC64 X function, you can dynamically add a system board either with the SPARC64 X or SPARC64 X+ processor through dynamic reconfiguration.

Table 2-15 lists the relationship between the physical partition configuration, CPU operation type, and CPU configuration for a system board that can be added through dynamic reconfiguration.

Table 2-15 Relationship Between the CPU Operational Mode and Addition Process Through Physical Partition Dynamic Reconfiguration (SPARC M10)

Physical Partition Configuration	CPU Operation Type	Dynamic Reconfiguration for System Board With SPARC64 X+ Processor	Dynamic Reconfiguration for System Board With SPARC64 X Processor
SPARC64 X+	Operates with the SPARC64 X+ function	Yes	No
SPARC64 X+	Operates in SPARC64 X compatible mode	Yes	Yes

Table 2-15 Relationship Between the CPU Operational Mode and Addition Process Through Physical Partition Dynamic Reconfiguration (SPARC M10) (*continued*)

Physical Partition Configuration	CPU Operation Type	Dynamic Reconfiguration for System Board With SPARC64 X+ Processor	Dynamic Reconfiguration for System Board With SPARC64 X Processor
Mixed use of SPARC64 X+/SPARC64 X	SPARC64 X+ operates in SPARC64 X compatible mode	Yes	Yes
SPARC64 X	Operates with the SPARC64 X function	Yes	Yes

Relationship between the CPU operational mode and deletion through physical partition dynamic reconfiguration

When the CPU operates with either the SPARC64 X+ or SPARC64 X function in SPARC M10, the system board can be deleted through dynamic reconfiguration.

[Table 2-16](#) lists the relationships between the physical partition configuration, CPU operation type, and CPU configuration for a system board that can be deleted through dynamic reconfiguration.

Table 2-16 Relationship Between the CPU Operational Mode and Deletion Process Through Physical Partition Dynamic Reconfiguration (SPARC M10)

Physical Partition Configuration	CPU Operation Type	Dynamic Reconfiguration for System Board With SPARC64 X+ Processor	Dynamic Reconfiguration for System Board With SPARC64 X Processor
SPARC64 X+	Operates with the SPARC64 X+ function	Yes	- (*1)
SPARC64 X+	Operates in SPARC64 X compatible mode	Yes	- (*1)
Mixed use of SPARC64 X+/SPARC64 X	Operates in SPARC64 X compatible mode	Yes	Yes (*2)
SPARC64 X	Operates with the SPARC64 X function	- (*1)	Yes

*1 This operation is not available in this configuration.

*2 For a physical partition with a mixed configuration incorporating the SPARC64 X+ processor and the SPARC64 X processor, execute the setpparmode command to set the CPU operational mode to "compatible". Suppose that you have a system that is configured only with a system board with the SPARC64 X+ processor, after deleting a system board with the SPARC64 X processor through dynamic reconfiguration and with the CPU operational mode set to "auto". In such cases, if you power off and then back on or reset the physical partition, the CPU is restarted with the SPARC64 X+ function. Then, even if you try to add a system board with the SPARC64 X processor through dynamic reconfiguration, it is not possible to do so because the system is operating with the SPARC64 X+ function. Therefore, if you consider an operation that involves active replacement of a system board of a physical partition with a mixed configuration combining the SPARC64 X+ processor and the SPARC64 X processor, set the CPU operational mode to "compatible".

Operations for Domain Configuration

This chapter describes operations and commands for domain configuration.

- [Operations and Commands Related to Physical Partition Configurations](#)
- [Operations and Commands Related to Logical Domain Configurations](#)

3.1 Operations and Commands Related to Physical Partition Configurations

The XSCF provides two user interfaces for physical partition configuration: the command line-based XSCF shell and Web browser-based XSCF Web. This section describes the XSCF shell commands that are mainly used for physical partition configuration-related operations.

There are two types of XSCF shell commands used to configure a physical partition: display-related commands and operation-related commands.

Table 3-1 Display-related Commands

Command Name	Functional Outline
showpcl	Displays physical partition configuration information.
showpparstatus	Displays the physical partition status.
showboards	Displays the system board status.
showfru	Displays device setting information.
showdomainstatus	Displays the logical domain status.
showcodactivation	Displays the CPU Activation key information.
showcodusage	Displays the usage status of the CPU core resources.
showcod	Displays the CPU Activation registration and setting information.
showpparmode	Displays the physical partition operation mode.

Table 3-1 Display-related Commands (*continued*)

Command Name	Functional Outline
showdomainconfig	Displays the logical domain configuration information for a physical partition.

Table 3-2 Operation-related Commands

Command Name	Functional Outline
setupfru	Sets memory mirroring.
setpcl	Sets physical partition configuration information.
addcodactivation	Adds a CPU Activation key.
setcod	Assigns CPU core resources used in a physical partition (PPAR).
addboard	Adds a system board (PSB) to a physical partition.
deleteboard	Deletes a PSB from a physical partition.
setpparmode	Sets the physical partition operation mode.
setdomainconfig	Specifies the logical domain configuration information for a physical partition.
poweron	Starts a physical partition.
poweroff	Stops a physical partition.
console	Connects to the control domain console.

This section describes the display-related commands and operation-related commands used for physical partition configuration in detail by using examples. For details of the options, operands, usage, etc. of each command, see the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual*.

Note - The XSCF shell and XSCF Web user interfaces are provided for some administrators, and they require administrator privileges to configure a physical partition. In cases where multiple physical partition administrators share a system board, preparation and deliberate operational planning are required for safer system operation.

3.1.1 Checking Physical Partition Configuration Information

Use the showpcl command to display physical partition configuration information (PPAR configuration information). The command displays PPAR configuration information, such as a PPAR-ID, a PSB number registered in the PPAR configuration information, and the physical partition status, in the form of a list.

Before performing operations related to physical partition dynamic reconfiguration, use the showpcl command to determine whether a physical partition can be dynamically reconfigured and to check the registration status of the target system

board in physical partition configuration information. Also use the command to check the status and configuration of a physical partition after a dynamic reconfiguration operation.

Note - A user account that has any of the following privileges can execute the showpcl command:

platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> showpcl [-v] -a
XSCF> showpcl [-v] -p ppar_id
```

- **-v option**
In addition to the default display, this option displays the [Cfg-policy], [No-Mem], and [No-IO] columns.
- **-a option**
This option displays PPAR configuration information for all the physical partitions.
- **-p ppar_id option**
Specify a physical partition number. This option displays the PPAR configuration information for the specified physical partition.

[Table 3-3](#) lists the display items of the showpcl command.

Table 3-3 Display Items of showpcl

Display Item	Description	Meaning
PPAR-ID	Integer from 0 to 15	Physical partition number
LSB	Integer from 0 to 15	Logical system board number
PSB	xx-y (xx: Integer from 00 to 15; y: Fixed at 0)	Physical system board number Indicates one building block (BB), supports LSB.
Status (PPAR operation status)	Powered Off	Power-off state
	Initialization Phase	State of POST operation in progress
	Initialization Complete	POST completed state
	Running	Running state after POST processing completed
Cfg-policy (*1) (Configuration policy of PPAR configuration information)	Hypervisor Aborted	Aborted state of Hypervisor until it is reset
	FRU	Degradation in units of parts, such as CPU or memory
	PSB	Degradation in units of PSBs
	System	Degradation in units of PPARs
No-Mem (*1) (no-mem flag of PPAR configuration information)	True	Do not allow the memory installed in the LSB to be used by a logical domain.
	False	Allow the memory installed in the LSB to be used by a logical domain.
No-IO (*1) (no-io flag of PPAR configuration information)	True	Do not allow the I/O device installed in the LSB to be used by a logical domain.
	False	Allow the I/O device installed in the LSB to be used by a logical domain.

*1 This item is displayed only if the -v option is specified.

The following example displays the PPAR configuration information for physical partition 0.

XSCF> showpcl -p 0			
PPAR-ID	LSB	PSB	Status
00			Running
	00	00-0	
	04	01-0	
	08	02-0	
	12	03-0	

To display the details of the configuration policy, no-mem, no-io, etc., specify the -v option. The following example displays the details of the PPAR configuration information for physical partition 0.

XSCF> showpcl -v -p 0						
PPAR-ID	LSB	PSB	Status	No-Mem	No-IO	Cfg-policy
00			Running			
						System
	00	-				
	01	-				
	02	-				
	03	-				
	04	01-0		False	False	
	05	-				
	06	-				
	07	-				
	08	02-0		True	False	
	09	-				
	10	-				
	11	-				
	12	03-0		False	True	
	13	-				
	14	-				
	15	-				

3.1.2 Checking the Physical Partition Status

Use the showpparstatus command to display the physical partition status.

Note - A user account that has any of the following privileges can execute the showpparstatus command:
 useradm, platadm, platop, fieldeng, pparadm, pparmgr, pparop

```

XSCF> showpparstatus -a
XSCF> showpparstatus -p ppar_id

```

- **-a option**
 This option displays the status of all physical partitions.
- **-p ppar_id option**
 Specify a physical partition number. This option displays the status of the specified physical partition.

Table 3-4 lists the display items of the showpparstatus command.

Table 3-4 Physical Partition Status

Display Item	Description	Meaning
PPAR-ID	Integer from 0 to 15	Physical partition number
PPAR Status (PPAR status)	Powered Off	Power-off state
	Initialization Phase	State of POST operation in progress
	Initialization Complete	POST completed state
	Running	Running state after POST processing completed
	Hypervisor Aborted	Aborted state of Hypervisor until it is reset
	-	Other than above (state where PSB is not assigned to PPAR)

The following example displays the status of operating physical partition 0.

```
XSCF> showpparstatus -p 0
PPAR-ID      PPAR Status
00           Running
```

3.1.3 Checking the System Board Status

Use the showboards command to display the status of a system board's assignment (PSB<BB>) to a physical partition.

Before performing physical partition dynamic reconfiguration operations, use the showboards command as follows: check the status of a PSB (BB) to confirm that a dynamic reconfiguration operation can be performed on it, check the number of the physical partition to which the PSB (BB) belongs, etc. Also use this command to check the PSB (BB) status after the dynamic reconfiguration operation.

Note - A user account that has any of the following privileges can execute the showboards command:
platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> showboards [-v] -a [-c sp]
XSCF> showboards [-v] -p ppar_id [-c sp]
XSCF> showboards [-v] psb
```

Note - The showboards command is effective only for a BB configuration.

- -v option
This option displays detailed information. Specify this option to check the status of a reservation to add or delete a PSB (BB).
- -a option
This option displays information on all the physical partitions.
- -c sp option
This option displays only the PSBs (BBs) in the system board pool state. The system board pool state is a state where the system board does not belong to any physical partition.
- -p ppar_id option
Specify a physical partition number. This option displays information on the specified physical partition.
- psb
Specify a PSB (BB) number.

Table 3-5 lists the display items of the showboards command.

Table 3-5 System Board Status

Display Item	Description	Meaning
PSB	xx-y (xx: Integer from 00 to 15; y: Fixed at 0)	Physical system board (PSB) number that represents one building block (BB) (where xx is the BB-ID)
R (*1)	*	The state where the PSB (BB) is either incorporated into the current physical partition configuration or it is released and the configuration changes when the physical partition restarts
PPAR-ID	Integer from 0 to 15	Number of the physical partition to which the PSB (BB) is assigned
	SP	PSB (BB) in the system board pool state
	Other	State where physical partition configuration information is set for a physical partition having user privileges and the system board belongs to a physical partition having no user privilege
LSB	Integer from 00 to 15	Logical system board number used by the physical partition
Assignment (State of being assigned to PPAR)	Unavailable	PSB (BB) that is not assigned to a physical partition and is in any of following states (including the state where the PSB (BB) is not mounted): system board pool, undiagnosed, diagnosis in progress, abnormal diagnosis
	Available	PSB (BB) in the system board pool state, with a diagnosis ended normally
	Assigned	PSB (BB) assigned

Table 3-5 System Board Status (*continued*)

Display Item	Description	Meaning
Pwr (Power-on state of system board)	n	Power-off state
	y	Power-on state
Conn (Incorporated state in PPAR configuration)	n	The state where the PSB (BB) is released from the physical partition configuration
	y	The state where the PSB (BB) is incorporated into the physical partition configuration
Conf (Running state in Oracle Solaris)	n	The state where the PSB (BB) is not operating with Oracle Solaris
	y	The state where the PSB (BB) is operating with Oracle Solaris
Test (Diagnosis state)	Unmount	Unmounted or undefined state
	Unknown	Undiagnosed state
	Testing	Diagnosis in progress
	Passed	State of normal end of diagnosis
	Failed	The state where a diagnosis has detected an abnormality, and the PSB (BB) is not operating
Fault (Degraded state)	Normal	Normal state
	Degraded	The state where there is a degraded part (the PSB (BB) can operate)
	Faulted	State where the PSB (BB) cannot operate because of an abnormality or cannot be managed because of a communication failure

*1 This item is displayed only if the -v option is specified.

In the following example, the `showboards` command is executed to display the assignment states of all PSBs (BBs) in all the physical partitions. PSBs displayed with "SP" under the [PPAR-ID] column are in the system board pool state. The system board pool state is a state where the PSB does not belong to any physical partition.

XSCF> showboards -a							
PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	00(01)	Assigned	y	y	y	Passed	Normal
02-0	SP	Available	n	n	n	Passed	Normal
03-0	SP	Available	n	n	n	Passed	Normal

To display the status of a reservation to add or delete a PSB, specify the -v option. The following example displays detailed information on all the mounted PSBs.

XSCF> showboards -v -a								
PSB	R	PPAR-ID (LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	*	00 (00)	Assigned	y	y	y	Passed	Normal
01-0		SP	Unavailable	n	n	n	Testing	Normal
02-0		Other	Assigned	y	y	n	Passed	Degraded
03-0		SP	Unavailable	n	n	n	Failed	Faulted

"*" displayed under the [R] column represents the reserved state. In the above example, you can see that PSB 00-0 is reserved for deletion, since it is in the assigned state.

3.1.4 Checking Device Setting Information

Use the showfru command to display device setting information.

Note - A user account that has the platadm or fieldeng privilege can execute the showfru command.

```
XSCF> showfru device location
```

- device
Specify sb (system board (PSB<BB>)) or cpu (CPU on a PSB (BB)) as a device name.
- location
If the device name is sb, specify BB-ID-0. If it is cpu, specify BB-ID-0-CPU. Here, the BB-ID range is 0 to 15, and the CPU range is 0 to 3.

Table 3-6 lists the display items of the showfru command.

Table 3-6 Display Items of the showfru Command

Display Item	Description	Meaning
Device (Device)	sb	System board (PSB)
	cpu	CPU
Location (Location of mounted device)	If Device is sb BB-ID-0 (BB-ID: Integer from 0 to 15)	PSB number
	If Device is cpu BB-ID-0-x (BB-ID: Integer from 0 to 15; x: Integer from 0 to 3)	CPU number
Memory Mirror Mode (Memory mirror mode)	yes	Memory mirror mode
	no	Not memory mirror mode

The following example displays device setting information for a PSB in the building

block BB-ID#01.

```
XSCF> showfru sb 01-0
Device Location Memory Mirror Mode
sb      01-0
  cpu 01-0-0    yes
  cpu 01-0-1    yes
  cpu 01-0-2    yes
  cpu 01-0-3    yes
```

The following example displays setting information for CPU#03 on a PSB in building block BB-ID#01.

```
XSCF> showfru cpu 01-0-3
Device Location Memory Mirror Mode
sb      01-0
  cpu 01-0-3    yes
```

3.1.5 Checking the Logical Domain Status

Use the showdomainstatus command to display the logical domain status.

Note - A user account that has any of the following privileges can execute the showdomainstatus command:
useradm, platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> showdomainstatus -p ppar_id [-g domainname]
```

- -p ppar_id option
Specify the number of the physical partition to be displayed.
- -g domainname option
Specify the name of the logical domain to be displayed.

The displayed items are shown in [Table 3-7](#).

Table 3-7 Logical Domain Status

Display Item	Description	Meaning
Logical Domain Name	-	Logical domain name
Status (current logical domain operation status)	Host stopped	Logical domain stopped state
	Solaris booting	State where logical domain Oracle Solaris is starting
	Solaris running	State where logical domain Oracle Solaris is running
	Solaris halting	State where logical domain Oracle Solaris is in stop process

Table 3-7 Logical Domain Status (*continued*)

Display Item	Description	Meaning
	Solaris suspended	State where logical domain Oracle Solaris is suspended
	Solaris powering down	State where logical domain Oracle Solaris is in power-off process
	Solaris rebooting	The state where logical domain Oracle Solaris is restarting
	Solaris panicking	State where panic has occurred in logical domain Oracle Solaris
	Solaris debugging	Stopped state with kmdb prompt of logical domain (Kernel being debugged)
	OpenBoot initializing	Initializing state of OpenBoot PROM of logical domain
	OpenBoot Running	State where initialization by OpenBoot PROM of logical domain has completed or stopped with ok prompt
	OpenBoot Primary Boot Loader	State of Oracle Solaris load in progress on logical domain
	OpenBoot Running OS Boot	State of transition to Oracle Solaris in progress on logical domain
	OS Started. No state support.	State where SUNW,soft-state-supported CIF has not been implemented and SUNW,set-trap-table CIF is running
	OpenBoot Running Host Halted	State where init 0 is being executed from logical domain Oracle Solaris
	OpenBoot Exited	State where reset-all is being executed from ok prompt of logical domain
	OpenBoot Host Received Break	State where enter service was called from logical domain Oracle Solaris
	OpenBoot Failed	Error occurred in initialization by OpenBoot PROM of logical domain
	Unknown	Unknown state where no logical domain was found to match logical domain name specified by user option (including state where add-spconfig command in Logical Domains Manager has not been executed)
	-	Other than above (state where PSB is not assigned to PPAR)

The following example displays the status of all the logical domains on physical partition 0.

```
XSCF> showdomainstatus -p 0
Logical Domain Name      Status
primary                  Solaris running
guest00                  Solaris running
guest01                  Solaris booting
guest02                  Solaris powering down
guest03                  Solaris panicking
guest04                  Shutdown Started
guest05                  OpenBoot initializing
guest06                  OpenBoot Primary Boot Loader
```

3.1.6 Checking the CPU Activation Key Information

Use the showcodactivation command to reference the information of the CPU Activation keys registered in the XSCF.

Note - A user account that has the platadm or platop privilege can execute the showcodactivation command.

```
XSCF> showcodactivation
```

The following example displays the CPU Activations for four CPU core resources in SPARC M10 (2 sets).

```
XSCF> showcodactivation
Index   Description Count
-----
0 PROC          2
1 PROC          2
```

Note - In SPARC M10, you can register one set of 2 CPU Activations in XSCF and enable 2 CPU core resources. In SPARC M12, you can register one set of 1 CPU Activation in the XSCF and enable 1 CPU core resource.

3.1.7 Checking the Usage Status of CPU Core Resources

Use the showcodusage command to display the usage status of CPU core resources.

Note - A user account that has any of the following privileges can execute the showcodusage command:
platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> showcodusage [-v] [-p {resource|ppar|all}]
```

- -v option
This option displays detailed information.
- -p {resource|ppar|all} option
Specify "all" to display the usage status of all the CPU core resources. Specify "resource" to display that information by resource. Specify "ppar" to display that information by physical partition.

The following example specifies the -p resource option in the showcodusage command to display the usage status of the CPU core resources by resource. As shown here, the system has 16 mounted CPU core resources and 4 registered CPU Activations, 4 of the CPU core resources are in use, and the number of CPU Activations currently not in use is 0.

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-----
PROC          4         16          4 OK: 0 cores available

Note:
  Please confirm the value of the "In Use" by the ldm command of
  Oracle VM Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of
  logical domains.
```

Table 3-8 lists the meanings of the display items of this command.

Table 3-8 Details Displayed by the showcodusage -p resource Command

Display Item	Meaning
Resource	Available CPU core resources (PROC: CPU, currently only for CPUs)
In Use	Number of CPU core resources currently used by the system (0 is returned when communication with Hypervisor cannot be established.)
Installed	Number of mounted CPU core resources in the system
CoD Permitted	Number of CPU Activations registered in the system
Status	CPU Activation availability for the CPU core resources in use (OK: Available, VIOLATION: Not available)

The following example specifies the -p ppar option in the showcodusage command in SPARC M10 to display the usage status of the CPU core resources in the physical partition. This example shows that 32 CPU core resources are mounted in physical partition 0, that the number of CPU Activations assigned to the partition is 4, and that 3 CPU core resources are currently in use there. Also, 32 CPU core resources are mounted in physical partition 1, the number of CPU Activations assigned to the partition is 4, and 4 CPU core resources are currently in use there. You can see from

this example that the number of CPU Activations currently not in use in the system (registered in the XSCF but not assigned to PPAR) is 12.

```
XSCF> showcodusage -p ppar
PPAR-ID/Resource In Use Installed Assigned
-----
0 - PROC          3          32          4 cores
1 - PROC          4          32          4 cores
2 - PROC          0           0          0 cores
3 - PROC          0           0          0 cores
4 - PROC          0           0          0 cores
5 - PROC          0           0          0 cores
6 - PROC          0           0          0 cores
7 - PROC          0           0          0 cores
8 - PROC          0           0          0 cores
9 - PROC          0           0          0 cores
10 - PROC         0           0          0 cores
11 - PROC         0           0          0 cores
12 - PROC         0           0          0 cores
13 - PROC         0           0          0 cores
14 - PROC         0           0          0 cores
15 - PROC         0           0          0 cores
Unused - PROC     0           0         12 cores

Note:
  Please confirm the value of the "In Use" by the ldm command of
  Oracle VM Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of
  logical domains.
```

If the -p ppar option is specified in the showcodusage command, you can confirm the items shown in [Table 3-9](#).

Table 3-9 Items Displayed by the showcodusage -p ppar Command

Display Item	Meaning
PPAR-ID/Resource	Physical partition number and CPU core resource type. A CPU core resource indicated as Unused is not used in the physical partition.
In Use	Number of CPU core resources currently used in the physical partition
Installed	Number of mounted CPU core resources in the physical partition
Assigned	Number of CPU Activations assigned to the physical partition

3.1.8 Checking CPU Activation Registration and Setting Information

Use the showcod command to check the CPU Activation registration and setting

information.

Note - A user account that has any of the following privileges can execute the showcod command:

platadm, platop, pparadm, pparmgr, pparop

```
XSCF> showcod [-v]
XSCF> showcod [-v] -s cpu
XSCF> showcod [-v] -p ppar_id
```

- -v option
This option displays detailed information. Displays a breakdown of CPU Activation.
- -s cpu option
Specify a CPU number.
- -p ppar_id option
Specify a physical partition number.

The following example uses the showcod command to display all the CPU Activation registration and setting information.

```
XSCF> showcod
PROC Permits installed: 6 cores
PROC Permits assigned for PPAR 0: 4
```

3.1.9 Checking the Physical Partition Operation Mode

Use the showpparmode command to display the physical partition operation mode.

Note - A user account that has any of the following privileges can execute the showpparmode command:

platadm, fieldeng, pparadm

```
XSCF> showpparmode -p ppar_id [-v]
```

- -p ppar_id option
Specify a physical partition.
- -v option
This option displays detailed information. The Ethernet (MAC) address of the physical partition is also displayed.

[Table 3-10](#) lists the display items of the showpparmode command.

Table 3-10 Display Items of the showpparmode Command

Display Item	Meaning
Host-ID	Host ID (hyphen "-" if no host ID is assigned) (*1)
Diagnostic Level	POST diagnosis level (none/normal/maximum)
Message Level	Detail level of POST diagnosis console messages (none/extract/normal/maximum/debug)
Alive Check	Enable/Disable for Alive Check
Watchdog Reaction	PPAR reaction when HOST Watchdog times out (reset/Panic process/disable (none))
Break Signal	Enable/Disable for Break signals (STOP-A)
Autoboot (Guest Domain)	Enable/Disable for the Auto boot function for guest domains at the PPAR start time
Elastic Mode	Enable/Disable for low-power CPU and memory operation
IOreconfigure	Enable/Disable for the I/O bus reconfiguration function according to the bus configuration when the PPAR is powered on or reboots (enable/disable/enable only at next start and automatically disable afterward)
PPAR DR(Current)	Current setting of the PPAR DR function (hyphen "-" if the PPAR power is not on) (*2)
PPAR DR(Next)	PPAR DR function settings after the next start time (*2)
Ethernet Address	Ethernet (MAC) address of the physical partition (*1)

*1 Different HOST-ID and Ethernet Address values are automatically assigned to each physical partition.

*2 The PPAR DR function is enabled by default in XCP 2210 and later. Before performing physical partition dynamic reconfiguration, confirm that the PPAR DR function is enabled, using the showpparmode command. If the function is disabled, execute the setpparmode command with -m ppar_dr=on specified to enable the PPAR DR function for the physical partition that will be dynamically reconfigured.

3.1.10 Displaying the Logical Domain Configuration Information for a Physical Partition

Use the showdomainconfig command to display logical domain configuration information.

```
XSCF> showdomainconfig -p ppar_id
```

- -p ppar_id option
Specify a physical partition.

Table 3-11 lists the display items of the showdomainconfig command.

Table 3-11 Display Items of the showdomainconfig Command

Display Item	Meaning
Index	Configuration information control number on the XSCF
PPAR-ID	Physical partition number
Booting config (Current)	Name of the configuration used for the physical partition that is currently running
Booting config (Next)	Name of the configuration used for the physical partition at the next startup
config_name	Configuration name
domains	Number of logical domains included in the logical domain configuration (*1)
date_created	Date and time when the configuration information was created

*1 When you saved logical domain configuration information to the XSCF with the ldm add-spconfig command, the number of logical domains in the bound or active state appears.

The following example displays the set configuration information for physical partition 0.

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) :ldm-set1
(Next) :ldm-set2
-----
Index :1
config_name :factory-default
domains :1
date_created:-
-----
Index :2
config_name :ldm-set1
domains :8
date_created:'2012-08-08 11:34:56'
-----
Index :3
config_name :ldm-set2
domains :20
date_created:'2012-08-09 12:43:56'
```

3.1.11 Setting Memory Mirroring

Use the setupfru command to set memory mirroring.

Note - A user account that has the platadm or fieldeng privilege can execute the setupfru

command.

- For SPARC M12

Specify `-c mirror=yes` to create a mirroring configuration for the memory in SPARC M12.

```
XSCF> setupfru [[-q] -{y|n}] -c function=mode device location
```

- For SPARC M10

Specify `-m y` to create a mirroring configuration for the memory in SPARC M10. The `-c mirror` option cannot be used.

```
XSCF> setupfru [-m {y|n}] device location
```

- `-c function=mode`
Specify whether to set the memory to mirror mode. To set it to mirror mode, specify `-c mirror=yes`. Otherwise, specify `-c mirror=no`. Can only be used in SPARC M12.
- `-m {y|n}`
Specify whether to set the memory to mirror mode. To set it to mirror mode, specify `y`. Otherwise, specify `n`. If the `-m` option is omitted, the previous setting is inherited.
- `device`
Specify the range to set for mirror mode. If `sb` is specified, the command reflects the setting on all the CPUs under the specified PSB. If `cpu` is specified, the command reflects the setting on only the specified CPU.
- `location`
If the device name is `sb`, specify a PSB number. The format is `BB-ID-0`. If the device name is `cpu`, specify a CPU. The format is `BB-ID-0-CPU`. Here, `BB-ID` is an integer ranging from 0 to 15, and `CPU` is an integer ranging from 0 to 3.

The following example sets all the CPUs under BB#01 (PSB#01) to memory mirror mode in SPARC M12.

```
XSCF> setupfru -c mirror=yes sb 01-0

Notice:
- Logical domain config_name will be set to "factory-default".

Memory mirror mode setting will be changed, Continue? [y|n] :y
```

The following example sets CPU1 under BB#02 to memory mirror mode in the SPARC M12.

```
XSCF> setupfru -c mirror=yes cpu 02-0-1
```

Notice:

- Logical domain config_name will be set to "factory-default".

Memory mirror mode setting will be changed, Continue? [y|n] :**y**

Note - If you use SPARC M10, specify the following:

XSCF> **setupfru -m y** *device location*

3.1.12 Setting Physical Partition Configuration Information

Use the setpcl command to register a system board in physical partition configuration information (PPAR configuration information) and to set the configuration policy, memory nullification option, and I/O nullification option.

Note - A user account that has the platadm privilege can execute the setpcl command.

```
XSCF> setpcl -p ppar_id -a lsb=psb [lsb=psb...]  
XSCF> setpcl -p ppar_id -s policy=value  
XSCF> setpcl -p ppar_id -s no-mem=value lsb [lsb...]  
XSCF> setpcl -p ppar_id -s no-io=value lsb [lsb...]
```

- **-p ppar_id option**

Specify the number to assign to the physical partition. Afterward, the number specified here is used for reference as the physical partition number.

Note - Assign a physical partition number that is the same as the BB-ID of any one of the chassis mounted on the system. If the BB-ID of an unmounted chassis is specified as a physical partition number, the power-on of the physical partition will fail.

- **-a lsb=psb [lsb=psb...] option**

This option maps a physical system board (PSB) to a logical system board (LSB). Specify a logical system board (LSB) number for lsb, and a physical system board (PSB) number for psb.

- **-s policy=value option**

This option sets the configuration policy for an entire physical partition. Specify any of the following for value as the unit of degradation: fru (each part), psb (system board unit (BB unit)), or system (entire physical partition).

Note - Set the configuration policy option while the power of the relevant physical partition is off.

- `-s {no-io|no-mem}=value`
Specify either true or false. If true is specified, nullification is set to make memory and I/O unavailable.
- `lsb`
Specify a logical system board number (LSB number).

Note - Set the memory nullification option and I/O nullification option when the relevant PSB (BB) is in the system board pool state or while the physical partition power is off. The system board pool state is a state where the system board does not belong to any physical partition.

Note - The memory nullification option and I/O nullification option can be set with the SPARC M12-2S/M10-4S.

The following example maps physical system boards (PSB) 00-0 and 01-0 to logical system boards (LSB) 0 and 1, respectively, of physical partition 0.

```
XSCF> setpctl -p 0 -a 0=00-0 1=01-0
```

The following example sets the configuration policy to "All physical partitions" for physical partitions 0 and 1.

```
XSCF> setpctl -p 0 -s policy=system
```

The following example sets the memory and I/O devices such that they are not used by logical system board (LSB) 1 in physical partition 0.

```
XSCF> setpctl -p 0 -s no-mem=true 1
XSCF> setpctl -p 0 -s no-io=true 1
```

3.1.13 Adding a CPU Activation Key

Use the `addcodactivation` command to add a CPU Activation key.

Note - A user account that has the `platadm` privilege can execute the `addcodactivation` command.

```
XSCF> addcodactivation {key-signature | -F url}
```

- `key-signature`
Specify the CPU Activation key by enclosing it in double quotation marks ("). Copy and paste the contents of the text file containing the CPU Activation key (XXXXX_XX_XXX.TXT).

- -F url

Specify the URL of the save destination for the CPU Activation key.

The following example adds the CPU Activation key for two CPU cores.

```
XSCF> addcodactivation "Product: SPARC M10-1
SequenceNumber: 10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxUddPQHKbtXgvZPsrtYguqiNUieB+mTDC
:
:
b1GCKFx1RH27FdVHiB2H0A=="
Above Key will be added, Continue?[y|n]:y
```

When the CPU Activation key is added to the system, CPU core resources are not yet ready for use on Oracle Solaris.

To make CPU core resources ready for use, see ["3.1.14 Assigning CPU Core Resources Used in a Physical Partition"](#) to assign them to a physical partition.

3.1.14 Assigning CPU Core Resources Used in a Physical Partition

Execute the setcod command interactively to assign CPU core resources to a physical partition based on the CPU Activations registered in the system.

Note - A user account that has the platadm privilege can execute the setcod command.

```
XSCF> setcod [-p ppar_id] -s cpu
PROC Permits installed: XX cores
PROC Permits assigned for PPAR 0 (X MAX)
[Permanent Xcores]
Permanent [X]: permits
PROC Permits assigned for PPAR 1 (X MAX)
[Permanent Xcores]
Permanent [X]: permits
: Omitted
```

If XSCF firmware of XCP 2260 or later has been applied, the following command can also be executed for this purpose.

```
XSCF> setcod [[-q] [-y|n]] -p ppar_id -s cpu -c {set|add|del} permits
```

- -p ppar_id option

Specify the physical partition ID to which to assign the CPU core resources. If you do not specify the permits operand, an interactive session to assign CPU core resources will start.

- `-s cpu`
Specify a CPU.
- `permits`
Specify the number of CPU Activations. A CPU Activation can be assigned on a core-by-core basis. The set number cannot exceed the number of available CPU Activations. If you do not specify the permits operand, an interactive session to assign CPU core resources will start.
- `-c {set|add|del}`
Specify `-c set` to assign a CPU Activation to a physical partition, `-c add` to increase the number of assigned CPU Activations, or `-c del` to delete an assigned CPU Activation.

Note - We do not recommend using the `setcod` command with the following specification:

XSCF> `setcod -p ppar_id -s cpu permits`

To execute the `setcod` command using XCP firmware of XCP 2260 or later, specify the `-c` option or use an interactive mode. Use an interactive mode for XCP 2250 or earlier.

The reasons for this are as follows:

- When the command is executed, the confirmation message that asks you whether to make a change using the set information (`[y|n]`) is not output.
 - When you reduce the number of CPU Activations assigned to a physical partition in operation, a warning message is not output.
This may cause system shutdown if there are not enough CPU Activations because of an incorrect permits specification or some other mistake.
-

The upper limit is the number of CPU Activations registered with the `addcodactivation` command.

Note - To specify the number of CPU Activations by using `-c set` or without using the `-c` option, specify in the permits operand either the currently set number plus the number of added CPU Activations or the currently set number minus the number of deleted CPU Activations. Do not specify the number of CPU Activations to be added or deleted alone. If you inadvertently specify the number of CPU Activations to be added or deleted alone, the number of CPU Activations decreases, which may cause the system to stop.

The following example assigns four CPU core resources to physical partition 1.

```
XSCF> setcod -p 1 -s cpu -c set 4
PROC Permits assigned for PPAR 1 : 0 -> 4
PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y
Completed.
```

The following example interactively assigns CPU core resources to the physical partition.

```
XSCF> setcod -s cpu
PROC Permits installed: 5 cores
PROC Permits assigned for PPAR 0 (5 MAX) [Permanent 2cores]
Permanent [2]:1
PROC Permits assigned for PPAR 1 (4 MAX) [Permanent 0cores]
```



```

Permanent [0]:4
PROC Permits assigned for PPAR 2 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 3 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 4 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 5 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 6 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 7 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 8 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 9 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 10 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 11 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 12 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 13 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 14 (0 MAX) [Permanent 0cores]
Permanent [0]:
PROC Permits assigned for PPAR 15 (0 MAX) [Permanent 0cores]
Permanent [0]:

```

The following example adds two CPU core resources to physical partition 0.

```

XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 10
XSCF> setcod -p 0 -s cpu -c add 2
PROC Permits assigned for PPAR 0 : 10 -> 12
PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y
Completed.
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 12

```

Note - With setcod -p 0 -s cpu -c set 12 specified, the command yields the same results.

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

```

XSCF> setcod -s cpu

```

3.1.15 Adding a System Board to a Building Block Configuration

Use the `addboard` command to add a system board (PSB<BB>) that represents a building block (BB) or reserve one for addition to a physical partition. Before executing this command, check the physical partition status (`showpparstatus` command) and PSB (BB) status (`showboards` command) by using the respective commands.

Note - A user account that has the `platadm` or `pparadm` privilege can execute the `addboard` command.

Note - The PSB (BB) to be added must be registered in advance in the PPAR configuration information for the relevant physical partition.

```
XSCF> addboard [-c configure] [-m bind={resource|none}]
               [-m diag={off|min}] -p ppar-id psb [psb...]
XSCF> addboard -c assign -p ppar_id psb [psb...]
XSCF> addboard -c reserve -p ppar_id psb [psb...]
```

- **-c configure option**
This option incorporates a PSB (BB) into the specified physical partition. If the physical partition power is off or Oracle Solaris is not running on the control domain, the system board is not incorporated, which results in an error. If the `-c assign` or `-c reserve` option is omitted, the `-c configure` option is assumed specified.
- **-c assign option**
This option assigns a PSB (BB) to the specified physical partition. The assigned PSB (BB) is reserved for the specified physical partition and cannot be assigned from another physical partition. After the assignment is done, restart the physical partition or execute the `addboard` command with the `-c configure` option specified to incorporate the PSB (BB) into the physical partition.
- **-c reserve option**
This option reserves the assignment of a PSB (BB) to the specified physical partition. Its behavior is the same as "assign."
- **-m bind={resource|none} option**
This option sets the automatic assignment function for additional resources from the PSB (BB) to be incorporated. If `resource` is specified, the automatic assignment function for additional resources is enabled. If `none` is specified, the automatic assignment function for additional resources is disabled, and resources from the added PSB (BB) become free resources in the physical partition. The default is `resource`.

If resources have been deleted with the `deleteboard` command before execution of the `addboard` command, you can recover the resource assignment that existed

before the execution of the deleteboard command by enabling the automatic assignment function. However, if you changed logical domain configuration information before executing the addboard command, the assignment of resources follows the logical domain configuration information.

If the logical domain configuration information for the physical partition is factory-default, the added resources are assigned to the control domain regardless of the setting of this option.

- **-m diag={off|min} option**
This option sets the level of hardware diagnosis performed when a PSB (BB) is incorporated into a physical partition. If off is specified, no hardware diagnosis is performed. If min is specified, normal is set as the hardware diagnosis level. The default is min.
- **-p ppar_id option**
Specify a physical partition number.
- **psb**
Specify a PSB (BB) number.

Note - In PSB (BB) incorporation using the -c configure option, the target PSB (BB) is incorporated into the physical partition after the PSB hardware diagnosis is completed. Therefore, it may take some time before execution of the addboard command finishes.

Note - When using the addboard command with the -c configure option specified to incorporate a PSB (BB) into a physical partition, you need to enable the PPAR DR function for the target physical partition by using the setpparmode command and restart the physical partition. The command with the -c configure option specified cannot incorporate a PSB (BB) into a physical partition where the PPAR DR function is disabled.

Note - If a CPU Activation violation has occurred in a physical partition, the addboard command cannot be executed for dynamic reconfiguration. Eliminate the CPU Activation violation, and then execute the addboard command to perform dynamic reconfiguration.

Note - If a dynamic reconfiguration operation by the addboard command fails, the target PSB (BB) cannot be returned to its state before this operation. Identify the cause from the error message output by the addboard command and the Oracle Solaris message, and take appropriate action. However, you may need to power off or restart the physical partition, depending on the error that occurred.

Note - If you omit the -m option or specify bind=resource for the -m option when you execute the addboard command for PSB (BB) replacement, the resource assignment may not be returned to its state before the deleteboard command is executed. If the amount of the resources of the PSB (BB) such as CPU and memory before replacement is different from that after replacement, you cannot return the resource assignment to its original state. Resources that cannot not be returned to their original assignment become free. In such cases, use the ldm command to assign resources to the logical domain again.

The following example assigns physical system boards (PSB) 00-0, 01-0, 02-0, and 03-0 to physical partition 0. The -y option is used to answer "y" to all the confirmation questions of the command.

```
XSCF> addboard -y -c assign -p 0 00-0 01-0 02-0 03-0
```

3.1.16 Deleting a System Board From a Building Block Configuration

Use the deleteboard command to delete a system board (PSB<BB>) that represents a building block (BB) or reserve the deletion of one from a physical partition. The deleted PSB (BB) is assigned to the system board pool, which does not belong to any physical partition.

Before executing this command, check the physical partition status (showpparstatus command), PSB (BB) status (showboards command), and status of device usage on the PSB (BB) (showfru command) by using the respective commands.

Note - A user account that has the platadm or pparadm privilege can execute the deleteboard command.

```
XSCF> deleteboard [-f] [-c disconnect]
                        [-m unbind={none|resource|shutdown}]
                        psb [psb...]
XSCF> deleteboard [-f] -c unassign psb [psb...]
XSCF> deleteboard [-f] -c reserve psb [psb...]
```

- -c disconnect option
This option releases a PSB (BB) operating in a physical partition from the physical partition configuration and puts it in the assigned state. To incorporate the PSB (BB) into the physical partition again, restart the physical partition or execute the addboard command while the system board remains assigned to the physical partition configuration. If the -c unassign or -c reserve option is omitted, the -c disconnect option is assumed specified.
- -c unassign option
This option unassigns a PSB (BB) from a physical partition and puts it in the system board pool state. The PSB (BB) in the system board pool state can be incorporated into or assigned to another physical partition.
- -c reserve option
This option reserves the unassignment of a PSB (BB) from a physical partition. After being reserved for unassignment, the PSB (BB) is unassigned based on PPAR configuration information at the power-off time of the specified physical partition, and put in the system board pool state.
- -m unbind={none|resource|shutdown} option

This option sets the action to take when a logical domain that uses resources on a PSB (BB) to be released is moved to a PSB (BB) that is not released and the destination has insufficient resources.

Setting	Description
none	Secures no resources at the destination. If the destination has insufficient resources, execution of the deleteboard command fails. If the logical domain configuration information for the physical partition is factory-default, the -m unbind=none option cannot be specified. If an error occurs, you must remove the CPU core or memory from the logical domain by using the virtual DR function of Oracle VM Server for SPARC.
resource	Secures resources at the destination by reducing either the resources of the logical domain that needs resources moved or the resources of another logical domain. However, it does not shut down a logical domain to secure resources at the destination. In addition, on XCP 2240 or later with Oracle VM Server for SPARC 3.2 or later, it automatically deletes the physical I/O.
shutdown	Secures resources at the destination by reducing either the resources of the logical domain that needs resources moved or the resources of another logical domain. In addition, it shuts down a logical domain to secure resources at the destination.

- psb
Specify a system board number.
- -f option
This option forcibly releases a PSB (BB).

Note - If the deleteboard command has the -f option specified to forcibly delete a PSB (BB) from a physical partition, a serious problem may occur in a cpu-bound process or a process accessing a device. Therefore, we recommend not using the -f option in normal operation. If you specify the -f option, be sure to check the status of the physical partition and business processes before and after executing the deleteboard command.

Note - Before using the deleteboard command with the -c disconnect or -c unassign option specified in order to delete a PSB (BB) while the physical partition is operating, do the following: use the setpparmode command to enable the PPAR DR function for the physical partition from which to delete the PSB (BB), and restart the physical partition. You cannot delete a PSB (BB) from a physical partition with the PPAR DR function disabled by specifying the -c disconnect or -c unassign option while the physical partition is operating.

Note - If a CPU Activation violation has occurred in a physical partition, the deleteboard command cannot be executed for dynamic reconfiguration. Eliminate the CPU Activation violation, and then execute the deleteboard command to perform dynamic reconfiguration.

Note - If dynamic reconfiguration processing by the deleteboard command fails, the target PSB (BB) cannot be returned back to its state before this processing. Identify the cause from the error message output by the deleteboard command and the Oracle Solaris message, and take appropriate action. However, you may need to power off or restart the physical partition, depending on the error that occurred.

The following example sets physical system boards (PSB) 00-0, 01-0, 02-0, and 03-0 to the system board pool state.

```
XSCF> deleteboard -c unassign 00-0 01-0 02-0 03-0
```

The following example reserves the release of PSBs 00-0, 01-0, 02-0, and 03-0.

```
XSCF> deleteboard -c reserve 00-0 01-0 02-0 03-0
```

3.1.17 Setting the Physical Partition Operation Mode

Use the setpparmode command to set the physical partition operation mode.

Note - To set the diagnosis level, message level, and Auto boot function for a guest domain, the setpparmode command must be executed with a user account that has the fieldeng privilege.
For Active Check, Break signals, the reaction when HOST Watchdog times out, the Auto boot function for guest domains, the power save function, the I/O bus reconfiguration function, and the PPAR DR function, this command must be executed with a user account that has the platadm or pparadm privilege.

```
XSCF> setpparmode -p ppar_id -m function=mode
```

- -p ppar_id option
Specify a physical partition.
- -m function=mode option
Specify a function and operation mode. For details, see the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual*.

Table 3-12 lists the operation modes that can be set.

Table 3-12 Modes That Can be Set for a PPAR

Function Name	Explanation	Settable Mode	Reflected
Diagnosis level	Sets the diagnosis level for the self diagnosis test (POST) at the power-on time.	None/Normal (default value)/Maximum	Immediately (*1)
Message level	Sets the detail level of POST diagnosis console messages.	None/Extract/Normal (default value)/Maximum/Debug	Immediately (*1)

Table 3-12 Modes That Can be Set for a PPAR (*continued*)

Function Name	Explanation	Settable Mode	Reflected
Alive Check	Sets enable/disable for Alive Check.	Enable (default value)/ Disable	At the next start time (*2)
Reaction when HOST Watchdog times out	Sets how each logical domain reacts when Watchdog times out.	None/Panic process/PPAR reset (default value)	Immediately (*1)
Break signal (STOP-A) suppression	Sets enable/disable for Break signal suppression.	Enable (default value)/ Disable	Immediately (*1)
Auto boot function for guest domains	Sets enable/disable for automatic startup of guest domains at the physical partition power-on time.	Enable (default value)/ Disable	At next start time (*2)
Power save function	Sets enable/disable for low-power CPU and memory operation.	Enable/Disable (default value)	Immediately (*1)
I/O bus reconfiguration (ioreconfigure) function	Selects the operation mode of the reconfiguration function for the I/O bus according to the bus configuration when the physical partition is powered on and then powered on again (reset).	Enable/Disable (default value)/Enable only at next start and automatically disable afterward	Immediately (*1)
PPAR DR function	Sets the function for incorporating and releasing system boards (PSB<BB>) in the configuration of a currently operating physical partition.	Enable/Disable (*3)	At next start time (*2)

*1 The setting is reflected immediately.

*2 To reflect the setting, you need to power on the physical partition, or power it off and then power it on again.

*3 The PPAR DR function is disabled by default in versions earlier than XCP 2210, and enabled by default in XCP 2210 and later.

Note - The I/O bus reconfiguration function is not supported by the SPARC M12-1/M10-1.

Note - The PPAR DR function is not supported by the SPARC M12-1/M12-2/M10-1/M10-4.

Note - Using the `ldm set-variable` command, set whether to automatically start Oracle Solaris together with a guest domain. By default, Oracle Solaris automatically starts together with a guest domain. To automatically start a domain, you need to make the domain active and save the logical domain configuration information. For details, see the *Oracle VM Server for SPARC Administration Guide* of the version used.

Mode switch operations

The physical partition operates as follows based on the setting of the mode switch on the operation panel.

- Locked
Operation is in the operation mode specified by the setpparmode command.
- Service
[Table 3-13](#) lists the operations at times when the mode switch setting is Service.

Table 3-13 Operations at Times When the Mode Switch Setting is Service

Function Name	Operation Mode
Diagnosis level	Operation is in the operation mode specified by the setpparmode command.
Message level	Operation is in the operation mode specified by the setpparmode command.
Alive Check	This is disabled regardless of the operation mode status.
Reaction when HOST Watchdog times out	Operation is in the operation mode specified by the setpparmode command.
Break signal (STOP-A) suppression	Break signals are transmitted regardless of the operation mode status.
Auto boot function for guest domains	Operation is in the operation mode specified by the setpparmode command.
Power save function	Operation is in the operation mode specified by the setpparmode command.
I/O bus reconfiguration function	Operation is in the operation mode specified by the setpparmode command.
PPAR DR function	Operation is in the operation mode specified by the setpparmode command.

3.1.18 Specifying the Logical Domain Configuration Information for a Physical Partition

Use the setdomainconfig command to specify physical partition configuration information.

```
XSCF> setdomainconfig -p ppar_id -i index
```

- -p ppar_id option
Specify a physical partition.
- -i index option
Specify a configuration information control number. You can check the control

number by using the `showdomainconfig` command. You can specify an integer from 1 to 8.

Note - The logical domain configuration saved on the XSCF does not become effective simply through a control domain restart with the shutdown command of Oracle Solaris. Be sure to restart the control domain by using the `poweroff` and `poweron` commands of the XSCF.

3.1.19 Starting a Physical Partition

Use the `poweron` command to power on a physical partition.

Note - A user account that has any of the following privileges can execute the `poweron` command:
`platadm`, `fieldeng`, `pparadm`, `pparmgr`

```
XSCF> poweron -p ppar_id
```

- `-p ppar_id` option
Specify a physical partition number.

3.1.20 Stopping a Physical Partition

Use the `poweroff` command to stop a physical partition.

```
XSCF> poweroff -p ppar_id
```

- `-p ppar_id` option
Specify a physical partition.

Note - For a logical domain configuration, execute the `ldm add-spconfig` command in Oracle VM Server for SPARC on the control domain to save the latest configuration information on the XSCF before the physical partition is powered off. For details, see "6.2.2 Saving the Logical Domain Configuration Information before System Stop" and "10.11.1 Saving/Displaying Logical Domain Configuration Information" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

3.1.21 Connecting to the Control Domain Console

Use the `console` command to connect to the control domain console of a physical partition.

Note - A user account that has any of the following privileges can execute the console command:
platadm, platop, fieldeng, pparadm, pparmgr, pparop

```
XSCF> console -p ppar_id
```

- -p ppar_id option
Specify a physical partition number.

3.2 Operations and Commands Related to Logical Domain Configurations

After configuring the physical partition, configure each logical domain. Use the `ldm` command of Logical Domains Manager, which is management software for Oracle VM Server for SPARC, to configure the logical domain. This section describes the display-related commands and operation-related commands used for logical domain configuration by using examples. For details of the options, operands, usage, etc. of each command, see the manuals of Oracle VM Server for SPARC and Oracle Solaris of the version used.

There are two types of commands used to configure a logical domain: display-related commands and operation-related commands.

Table 3-14 Display-related Commands

Command Name	Functional Outline
<code>svcs</code>	Checks whether Logical Domains Manager has started.
<code>ldm list-services</code>	Displays services.
<code>ldm list-permits</code>	Displays the number of virtual CPUs that can be assigned.
<code>ldm list-bindings</code>	Displays the assignment status of resources.
<code>ldm list-devices</code> <code>ldm list-rsrc-group</code> <code>ldm list-socket</code>	Displays the usage status of resources.
<code>ldm list-domain</code>	Displays the logical domain status.
<code>ldm list-spconfig</code>	Displays logical domain configuration information.
<code>ldm list-io</code>	Displays the usage status of I/O devices.

Table 3-15 Operation-related Commands

Command Name	Functional Outline
ldm start-reconf	Starts delayed reconfiguration mode.
ldm add-vconscon	Sets the default services.
ldm add-vdiskserver	
ldm add-vswitch	
ldm add-vcpu	Configures virtual CPUs.
ldm set-vcpu	
ldm remove-vcpu	
ldm add-core	
ldm set-core	
ldm remove-core	
ldm grow-socket	
ldm shrink-socket	
ldm add-memory	Configures virtual memory.
ldm set-memory	
ldm remove-memory	
ldm grow-socket	
ldm shrink-socket	
ldm add-spconfig	Sets logical domain configuration information.
ldm set-spconfig	
ldm remove-spconfig	
ldm add-domain	Creates a logical domain.
ldm add-io	Configures an I/O device.
ldm remove-io	
ldm create-vf	Creates or destroys a virtual function of the SR-IOV.
ldm destroy-vf	
ldm add-vnet	Configures a virtual network device.
ldm remove-vnet	
ldm add-vdiskserverdevice	Configures a virtual disk server.
ldm remove-vdiskserverdevice	
ldm add-vdisk	Configures a virtual disk.
ldm remove-vdisk	
ldm set-socket	Sets CPU socket constraints.
ldm set-vconsole	Configures a virtual console.
ldm set-variable	Configures a startup device.
ldm bind-domain	Binds resources.
ldm start-domain	Starts a guest domain.
ldm set-domain	Configures a shutdown group.
devfsadm	Reconfigures a device.
svccfg	Configures recovery mode.

3.2.1

Confirming That Logical Domains Manager is Running

Use the `svcs` command of Oracle Solaris to confirm that Logical Domains Manager is running.

```
# svcs service
```

- `service`
Specify a service name.

The following example checks whether Logical Domains Manager is running. The service name of Logical Domains Manager is `ldmd`. If the `[STATE]` column shows "online", it is running.

```
# svcs ldmd
STATE STIME FMRI
online 16:25:31 svc:/ldoms/ldmd:default
```

Note - Logical Domains Manager is software included in Oracle VM Server for SPARC. For details on the `svcs` command, see the reference manual of Oracle Solaris.

Note - If Logical Domains Manager has not started, execute the `svcadm` command to start it. For details of the `svcadm` command, see the reference manual of Oracle Solaris.

3.2.2

Checking Services

Use the `ldm list-services` command to confirm the set services.

```
# ldm list-services ldom
```

- `ldom`
Specify the logical domain name. The control domain name is "primary".

The following example displays "primary-vds0" as the virtual disk server, "primary-vcc0" as the virtual console terminal collection and distribution unit service, and "primary-vsw0" as the virtual switch service.

```
# ldm list-services primary
VDS
  NAME          VOLUME          OPTIONS          DEVICE
  primary-vds0
VCC
  NAME          PORT-RANGE
```

```
primary-vcc0 5000-5100
VSW
NAME          MAC          NET-DEV  DEVICE  MODE
primary-vsw0  02:04:4f:fb:9f:0d  net0    switch@0  prog,promisc
```

3.2.3 Checking the Number of Virtual CPUs That Can be Assigned Based on CPU Activations

Use the `ldm list-permits` command to check the number of CPU cores that can be assigned from the control domain.

```
# ldm list-permits
```

3.2.4 Checking the Assignment Status of Resources

Use the `ldm list-bindings` command to check an assignment.

```
# ldm list-bindings ldom
```

- `ldom`
Specify the logical domain. If it is the control domain, "primary" is the fixed setting.

3.2.5 Checking the Usage Status of Resources

Use the `ldm list-devices` command to check the usage status of the resources in a physical partition.

```
# ldm list-devices [-S] [-a] [cpu]
```

- `-S` option
This option displays the device status.
- `-a` option
This option displays the usage status of all the resources that are bound or not bound.
- `cpu`
CPU threads and physical CPU resources are displayed in a list.

[Table 3-16](#) lists the meanings of the display items of this command.

Table 3-16 Display Items of the ldm list-devices Command

Display Item	Description	Meaning
PID	Integer of 0 or greater	Physical CPU number
%FREE	Numeric value from 0 to 100	Percentage not used
PM	yes	CPU power supply managed
	no	CPU power on
	---	CPU not assigned to logical domain
STATUS (*1)	ok	CPU in normal status
	fail	CPU failed

*1 STATUS is displayed when the -S option is specified.

The following example displays the status of eight virtual CPUs.

```
# ldm list-devices -S -a cpu
VCPU
  PID %FREE PM  STATUS
    0   0   no  ok
    1   0  yes  ok
    2   0  yes  ok
    3   0  yes  ok
    4  100  ---  ok
    5  100  ---  ok
    6  100  ---  ok
    7  100  ---  ok
```

In the following example, the -a option is specified to display all resources bound to the logical domain and all resources that are not bound.

```
# ldm list-devices -a
CORE
  ID      %FREE  CPUSSET
    0       0    (0, 1)
    4       0    (8, 9)
    8       0   (16, 17)
(Omitted)
VCPU
  PID      %FREE  PM
    0       0    no
    1       0    no
    8       0    no
    9       0    no
(Omitted)
MEMORY
  PA              SIZE          BOUND
  0x7c0000000000    8G        guest2
  0x7c0200000000   24G
```

0x7e0000800000	1G	_sys_	
0x7e0040800000	384M	_sys_	
(Omitted)			
IO			
DEVICE		PSEUDONYM	BOUND
OPTIONS			
pci@8000		PCIE0	yes
pci@8100		PCIE1	yes
pci@8200		PCIE2	yes
pci@8300		PCIE3	yes
PCIEDEV			
pci@8000/pci@4/pci@0/pci@9		/BB0/CMUL/NET0	yes
pci@8000/pci@4/pci@0/pci@a		/BB0/CMUL/NET2	yes
pci@8000/pci@4/pci@0/pci@0		/BB0/CMUL/SASHBA	yes
pci@8100/pci@4/pci@0/pci@0		/BB0/PCI0	yes
(Omitted)			

3.2.6 Checking the Usage Status of Each Resource Group

Use the `ldm list-rsrc-group` command to check the resource usage status of each resource group in a physical partition. A resource group is configured based on the underlying physical relationship between processor cores, memory, and I/O buses. The SPARC M12/M10 system supports this subcommand for Oracle VM Server for SPARC 3.3 and later, and displays a chassis or BB-ID as a resource group. For details on a resource group, see "Managing Resources" in the *Oracle VM Server for SPARC Administration Guide*.

```
# ldm list-rsrc-group [-a] [-d ldom] [-l] [-o core|memory|io] [-p] [resource-group]
```

- **-a**
This option displays all the resources in each resource group. Resources that are not bound to any of the logical domains will also be included.
- **-d**
This option displays information only on the specified logical domain.
- **-l**
This option displays detailed information on each resource group.
- **-o**
This option displays information only on the specified resources.
- **-p**
This option outputs results in a parseable format.
- **resource-group**
Specify a resource group to list information.

The following example lists all the resources in each resource group in a physical

partition with two system boards.

# ldm list-rsrc-group			
NAME	CORE	MEMORY	IO
/BB0	64	262136M	8
/BB1	64	262136M	8

3.2.7 Checking the Resource Usage Status of Each CPU Socket

Use the ldm list-socket command to check the resource usage status of each CPU socket in a physical partition. This subcommand is supported by Oracle VM Server for SPARC 3.2 and later.

```
# ldm list-socket [-l] [-p] [--free] [-o format] [socket_id=id[,id[,...]]]
[ldom...]
```

- -l
This option displays a physical CPU set and core IDs.
- -p
This option outputs results in a parseable format.
- --free
This option displays only free resources.
- -o
This option limits the output results to one or more subsets. For details on subsets, see the *Oracle VM Server for SPARC Reference Manual* of the version used.
- socket_id
Specify one or more CPU sockets whose information you want to list.
- ldom
Specify one or more logical domains whose information you want to list.

The following example lists the usage statuses of all the resources of each CPU socket.

# ldm list-socket				
CONSTRAINTS				
SOCKET				
TENANT	VCPUS	CORES	SOCKET_ID	GROUP
primary	32	16	0	/BB0
primary	32	16	1	/BB0
primary	32	16	2	/BB0
primary	32	16	3	/BB0
primary	32	16	4	/BB1
primary	32	16	5	/BB1
primary	32	16	6	/BB1
primary	32	16	7	/BB1

MEMORY				
PA	SIZE	SOCKET_ID	BOUND	
0x700000000000	64G	7	primary	
0x720000000000	64G	6	primary	
0x740000000000	64G	5	primary	
0x760050000000	31488M	4	primary	
0x780000000000	64G	3	primary	
0x7a0000000000	64G	2	primary	
0x7c0000000000	64G	1	primary	
0x7e0080000000	62G	0	primary	
IO				
NAME	TYPE	BUS	SOCKET_ID	BOUND
PCIE0	BUS	PCIE0	0	primary
PCIE1	BUS	PCIE1	0	primary
PCIE2	BUS	PCIE2	1	primary
PCIE3	BUS	PCIE3	1	primary
PCIE4	BUS	PCIE4	2	primary
PCIE5	BUS	PCIE5	2	primary
PCIE6	BUS	PCIE6	3	primary
PCIE7	BUS	PCIE7	3	primary
PCIE8	BUS	PCIE8	4	primary
PCIE9	BUS	PCIE9	4	primary
PCIE10	BUS	PCIE10	5	primary
PCIE11	BUS	PCIE11	5	primary
PCIE12	BUS	PCIE12	6	primary
PCIE13	BUS	PCIE13	6	primary
PCIE14	BUS	PCIE14	7	primary
PCIE15	BUS	PCIE15	7	primary

3.2.8 Checking the Logical Domain Status

Use the `ldm list-domain` command to check the status of a logical domain operating in a physical partition and shutdown group settings.

```
# ldm list-domain [-o format] [ldom ...]
```

- **format**
Specify the command format. Specify domain to check shutdown group settings.
- **ldom**
Specify the name of the target logical domain.

The following example shows the control domain and three guest domains in operation.

# ldm list-domain							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	14	8G	64%	2h 54m
guest0	active	-n----	5000	16	8G	42%	2h 54m
guest1	active	-n----	5001	16	8G	11%	2h 54m
guest2	active	-n----	5002	16	8G	7.3%	2h 54m

The following example sets shutdown group 7 for the logical domain ldom1 and checks the settings. The last line shows that the shutdown group is 7.

```
# ldm set-domain shutdown-group=7 ldom1
# ldm list-domain -o domain ldom1
NAME                STATE                FLAGS                UTIL
ldom1               active              -n----              0.1%

CONTROL
    failure-policy=ignore
    extended-mapin-space=on
    shutdown-group=7
```

The shutdown group settings are included in the logical domain configuration information. Therefore, when changing the shutdown group settings, you need to save their contents to the XSCF by using the `ldm add-spconfig` command.

3.2.9 Displaying the Logical Domain Configuration Information

Use the `ldm list-spconfig` command from the control domain to display the logical domain configuration information saved to the XSCF.

```
# ldm list-spconfig
```

The following example executes the `ldm list-spconfig` command from the control domain to display logical domain configuration information.

```
# ldm list-spconfig
ldm-set1 [current]
ldm-set2
factory-default
```

3.2.10 Checking the Usage Status of I/O Devices

Use the `ldm list-io` command to check the I/O devices configured in logical domains in a physical partition.

```
# ldm list-io [-l] [-p] [bus | device | pf-name]
# ldm list-io -d pf-name
```

- -l
Displays the details such as the relationship between the root complex and the PCIe direct I/O device.
- -p

Displays information in parseable format, such as in the script.

- -d pf-name
Displays the list of information regarding the specified physical functions.
- bus, device, and pf-name
These are, respectively, the PCIe bus, the direct I/O-assignable device, and the PCIe SR-IOV physical function.

The following example displays the usage status of I/O devices in logical domains. For PCIe devices that support the SR-IOV function, "PF" is indicated in the [TYPE] column.

For devices that create or use a virtual function of the SR-IOV, "VF" is indicated in the [TYPE] column.

```
# ldm list-io
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
PCIE0                              BUS     PCIE0    primary   IOV
PCIE1                              BUS     PCIE1    primary   IOV
PCIE2                              BUS     PCIE2    primary   IOV
PCIE3                              BUS     PCIE3    primary   IOV
PCIE4                              BUS     PCIE4    primary   IOV
PCIE5                              BUS     PCIE5    primary   IOV
PCIE6                              BUS     PCIE6    primary   IOV
PCIE7                              BUS     PCIE7    primary   IOV
(Omitted)
/BB0/PCI6                          PCIE    PCIE6    primary   EMP
/BB0/PCI9                          PCIE    PCIE7    primary   EMP
/BB0/PCI10                         PCIE    PCIE7    primary   EMP
/BB0/PCI0/IOVNET.PF0              PF      PCIE1    primary
/BB0/PCI0/IOVNET.PF1              PF      PCIE1    primary
/BB0/CMUL/NET0/IOVNET.PF0         PF      PCIE0    primary
/BB0/CMUL/NET0/IOVNET.PF1         PF      PCIE0    primary
/BB0/CMUL/NET2/IOVNET.PF1         PF      PCIE4    primary
/BB0/CMUL/NET2/IOVNET.PF0         PF      PCIE4    primary
/BB0/PCI0/IOVNET.PF0.VF0          VF      PCIE1    iodom00
/BB0/PCI0/IOVNET.PF0.VF1          VF      PCIE1    iodom01
/BB0/PCI0/IOVNET.PF0.VF2          VF      PCIE1    iodom02
```

To confirm the details, such as the relationship between the root complex and the PCIe direct I/O device, execute the command with the -l option specified.

The following example displays the usage status of I/O devices in logical domains.

- Logical domain to which the PCIe card is assigned
The name displayed in the [DOMAIN] column of the slot (the line whose [NAME] column displays "/BBx/PCIx") of the maintenance target PCIe card indicates the logical domain to which the PCIe card is assigned.
- Logical domain to which the PCIe root complex that the PCIe card belongs to is assigned
"PCIEx," which is displayed in the [BUS] column that was confirmed above, is the name of the PCIe root complex to which the PCIe card belongs. The [DOMAIN] column displays a name for the same PCIe root complex (on the line whose [NAME] column displays "PCIEx"). The name is a logical domain name. That PCIe

root complex to which the PCIe card belongs is assigned to the logical domain.

```
# ldm list-io -l
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
[pci@8000]				
PCIE1	BUS	PCIE1	primary	IOV
[pci@8100]				
PCIE2	BUS	PCIE2	primary	IOV
[pci@8200]				
PCIE3	BUS	PCIE3	rootdom	IOV
[pci@8300]				
PCIE4	BUS	PCIE4	primary	IOV
[pci@8400]				
PCIE5	BUS	PCIE5	primary	IOV
[pci@8500]				
PCIE6	BUS	PCIE6	rootdom	IOV
[pci@8600]				
PCIE7	BUS	PCIE7	primary	IOV
[pci@8700]				
/BB0/CMUL/NET0	PCIE	PCIE0	primary	OCC
[pci@8000/pci@4/pci@0/pci@9]				
network@0				
network@0,1				
/BB0/CMUL/SASHBA	PCIE	PCIE0	primary	OCC
[pci@8000/pci@4/pci@0/pci@0]				
scsi@0/iprota@f/disk@w50000393a82368b2,0				
scsi@0/iprota@f/smp@w500000e0e06d03bf				
scsi@0/iprota@f/enclosure@w500000e0e06d03bd,0				
scsi@0/iprota@v0				
/BB0/PCI0	PCIE	PCIE1	primary	OCC
[pci@8100/pci@4/pci@0/pci@0]				
/BB0/PCI3	PCIE	PCIE2	primary	EMP
[pci@8200/pci@4/pci@0/pci@0]				
/BB0/PCI4	PCIE	PCIE2	iodom1	OCC
[pci@8200/pci@4/pci@0/pci@8]				
/BB0/PCI7	PCIE	PCIE3	rootdom	EMP
[pci@8300/pci@4/pci@0/pci@0]				
/BB0/PCI8	PCIE	PCIE3	rootdom	EMP
[pci@8300/pci@4/pci@0/pci@8]				
/BB0/CMUL/NET2	PCIE	PCIE4	primary	OCC
[pci@8400/pci@4/pci@0/pci@a]				
network@0				
network@0,1				

3.2.11 Starting Delayed Reconfiguration Mode

Switch to delayed reconfiguration mode before performing the work for reconfiguring hardware resources assigned to the control or root domain. This is because reconfiguration of virtual memory may take some time during

reconfiguration by dynamic reconfiguration. Use the `ldm start-reconf` command to start delayed reconfiguration mode.

```
# ldm start-reconf ld0m
```

The following example switches primary, the control domain, to delayed reconfiguration mode.

```
# ldm start-reconf primary
```

Use the `ldm list-bindings` command to check whether the mode is delayed reconfiguration mode. The following example shows the execution results of the `ldm list-bindings` command. In the results, the third character from the beginning of "-ndcv-" in the [FLAGS] column is "d," which represents delayed reconfiguration mode.

# ldm list-bindings primary							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-ndcv-	SP	8	4G	0.1%	2h 16m

After switching to delayed reconfiguration mode, you cannot operate a guest domain with the `ldm` command. The following example shows a failure to stop a guest domain.

```
# ldm stop-domain -a
-----
Notice: LDom primary is in the process of a delayed
reconfiguration.
Any changes made to primary will only take effect after it
reboots.
-----
Delayed reconfiguration of LDom ld0m1 is not permitted
```

3.2.12 Setting the Default Services

Use the following commands to set the default services for a logical domain.

Table 3-17 Commands for Setting the Default Services

Command	Functional Outline
<code>ldm add-vconscn</code>	Adds the virtual console terminal collection and distribution service.
<code>ldm add-vdiskserver</code>	Adds the virtual disk server.
<code>ldm add-vswitch</code>	Adds the virtual switch service.

ldm add-vconscon command

Use the `ldm add-vconscon` command to add the virtual console terminal collection and distribution unit service to the control domain.

```
# ldm add-vconscon port-range=x-y vcc-name ldom
```

- **x**
Specify the lower limit value for the port number of the virtual console terminal collection and distribution unit.
- **y**
Specify the upper limit value for the port number of the virtual console terminal collection and distribution unit.
- **vcc-name**
Specify the name of the virtual console terminal collection and distribution unit.
- **ldom**
Specify the logical domain name. The control domain name is "primary". If the specified logical domain is not the control domain, the logical domain becomes the console output destination.

If the specified logical domain is not the control domain, execute the `ldm add-vcc` command, and then enable the virtual network terminal server daemon (`vntsd`) for the logical domain by using the `svcadm enable` command.

```
# svcadm enable svc:/ldoms/vntsd
```

ldm add-vdiskserver command

Use the `ldm add-vdiskserver` command to add a virtual disk server to the control domain.

```
# ldm add-vdiskserver service-name ldom
```

- **service-name**
Specify a virtual disk server name. This name must be unique among all the virtual disk server names within the system.
- **ldom**
Specify the logical domain name. The control domain name is "primary".

ldm add-vswitch command

Use the `ldm add-vswitch` command to add the virtual switch service to the control domain.

```
# ldm add-vswitch net-dev=device vswitch-name ldom
```

- net-dev=device
Specify a network device name.
- vswitch-name
Specify a virtual switch service name.
- ldom
Specify the logical domain name. The control domain name is "primary".

3.2.13 Configuring Virtual CPUs

Use the following commands to configure virtual CPUs.

Table 3-18 Commands for Configuring Virtual CPUs

Command	Functional Outline
ldm add-vcpu	Adds virtual CPUs in units of threads.
ldm set-vcpu	Sets virtual CPUs in units of threads.
ldm remove-vcpu	Removes virtual CPUs in units of threads.
ldm add-core	Adds virtual CPUs in units of cores.
ldm set-core	Sets virtual CPUs in units of cores.
ldm remove-core	Removes virtual CPUs in units of cores.
ldm grow-socket	Adds virtual CPUs associated with a specific CPU socket.
ldm shrink-socket	Removes virtual CPUs associated with a specific CPU socket.

Note - CPUs cannot be assigned to a logical domain in units of cores and in units of threads simultaneously. Choose either of these units of assignment for each logical domain.

ldm add-vcpu command

Use the ldm add-vcpu command to add a virtual CPU to a logical domain.

ldm add-vcpu *number ldom*

- number
Specify the number of virtual CPUs (threads) to add.
- ldom
Specify the logical domain. If it is the control domain, "primary" is the fixed setting.

The following example adds one virtual CPU to the control domain.

```
# ldm add-vcpu 1 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconf  
iguration.  
Any changes made to the primary domain will only take effect aft  
er it reboots.  
-----
```

ldm set-vcpu command

Use the ldm set-vcpu command to assign a virtual CPU to a logical domain.

```
# ldm set-vcpu number ldom
```

- **number**
Specify the number of virtual CPUs (threads) to assign.
- **ldom**
Specify the logical domain. If it is the control domain, "primary" is the fixed setting.

ldm remove-vcpu command

Use the ldm remove-vcpu command to remove a virtual CPU from a logical domain.

```
# ldm remove-vcpu number ldom
```

- **number**
Specify the number of virtual CPUs (threads) to remove.
- **ldom**
Specify the logical domain. If it is the control domain, "primary" is the fixed setting.

The following example deletes one virtual CPU from the control domain.

```
# ldm remove-vcpu 1 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconf  
iguration.  
Any changes made to the primary domain will only take effect aft  
er it reboots.  
-----
```

ldm add-core command

Use the ldm add-core command to add a specific core to a logical domain.

```
# ldm add-core num ldom  
# ldm add-core cid=core-ID [,core-ID [,...]] ldom
```


- num
Specify the number of CPU cores to add.
- ldom
Specify the logical domain. If it is the control domain, "primary" is the fixed setting.
- cid=core-ID [,core-ID [...]]
Specify the ID of each CPU core to add.

Note - If you add a CPU core by specifying CID, the configuration cannot be changed through dynamic reconfiguration.

Note - For the relationship between the CPU core ID (cid) and the physical location of the CPU, see "CPU core ID specification" in ["2.4.1 Consideration of Logical Domain Configuration."](#)

Note - The add-core subcommand is not applicable to a domain that has a virtual CPU assigned with the whole-core constraint disabled. For information about the whole-core constraint, see "How to Apply the whole-core constraint" in the *Oracle VM Server for SPARC Administration Guide*.

The following example adds a 1-core CPU to the control domain.

```
# ldm add-core 1 primary
```

ldm set-core command

Use the ldm set-core command to set a specific core on a logical domain.

```
# ldm set-core num ldom
# ldm set-core cid=core-ID [,core-ID [...]] ldom
```

- num
Specify the number of CPU cores to set.
- ldom
Specify the logical domain. If it is the control domain, "primary" is the fixed setting.
- cid=core-ID [,core-ID [...]]
Specify an ID for the set CPU core.

Note - If you set a CPU core by specifying CID, the configuration cannot be changed through dynamic reconfiguration.

Note - For the relationship between the CPU core ID (cid) and the physical location of the CPU, see "CPU core ID specification" in ["2.4.1 Consideration of Logical Domain Configuration."](#)

ldm remove-core command

Use the ldm remove-core command to remove a specific core from a logical domain.

```
# ldm remove-core num ldom
# ldm remove-core cid=core-ID [,core-ID [...]] ldom
# ldm remove-core -g resource-group [-n num] ldom
```

- num
Specify the number of CPU cores to set.
- ldom
Specify the logical domain. If it is the control domain, "primary" is the fixed setting.
- cid=core-ID [,core-ID [...]]
Specify an ID for the set CPU core.
- -g resource-group
This option removes CPU cores by specifying a resource group. The SPARC M12/M10 system supports this option for Oracle VM Server for SPARC 3.3 and later.

Note - The remove-core subcommand is not applicable to a domain that has a virtual CPU assigned with the whole-core constraint disabled. For information about the whole-core constraint, see "How to Apply the whole-core constraint" in the *Oracle VM Server for SPARC Administration Guide*.

In the following example, one CPU core is deleted from the control domain.

```
# ldm remove-core 1 primary
```

ldm grow-socket command

Use the ldm grow-socket command to add a virtual CPU associated with a CPU socket to a logical domain. This subcommand is supported by Oracle VM for SPARC 3.2 and later.

```
# ldm grow-socket vcpus=num socket_id=id ldom
# ldm grow-socket cores=num socket_id=id ldom
```

- num
Specify the number of CPU threads or CPU cores to add.
- id
Specify the CPU socket ID to add.
- ldom
Specify the logical domain.

The following example adds a CPU core to the control domain from CPU socket#2.

```
# ldm grow-socket cores=1 socket_id=2 primary
```

ldm shrink-socket command

Use the ldm shrink-socket command to remove a virtual CPU associated with a CPU socket. This subcommand is supported by Oracle VM for SPARC 3.2 and later.

```
# ldm shrink-socket vcpus=num socket_id=id ldom
# ldm shrink-socket cores=num socket_id=id ldom
```

- num
Specify the number of CPU threads or CPU cores to remove.
- id
Specify the CPU socket ID to remove.
- ldom
Specify the logical domain.

The following example removes a CPU core associated with CPU socket#2 from the control domain.

```
# ldm shrink-socket cores=1 socket_id=2 primary
```

3.2.14 Configuring Virtual Memory

Use the following commands to configure virtual memory.

Table 3-19 Commands for Configuring Virtual Memory

Command	Functional Outline
ldm add-memory	Adds virtual memory.
ldm set-memory	Sets virtual memory.
ldm remove-memory	Removes virtual memory.
ldm grow-socket	Adds virtual memory associated with a CPU socket.
ldm shrink-socket	Removes virtual memory associated with a CPU socket.

ldm add-memory command

Use the ldm add-memory command to add virtual memory to a logical domain.

```
# ldm add-memory [--auto-adj] size[unit] ldom
```

- --auto-adj option
This option adjusts the memory size to the boundary of 256 MB.

- **size**
Specify the size of memory to add. The minimum assignment unit is 256 MB.
- **unit**
Specify the unit of size. You can specify the following units: G (gigabyte), M (megabyte), and K (kilobyte). The specification is case-insensitive.
- **ldom**
Specify the logical domain. If it is the control domain, "primary" is the fixed setting.

The following example adds 1 GB of virtual memory.

```
# ldm add-memory 1g primary
-----
Notice: The primary domain is in the process of a delayed
reconfiguration.
Any changes made to the primary domain will only take effect
after it reboots.
-----
```

ldm set-memory command

Use the ldm set-memory command to assign virtual memory to a logical domain.

```
# ldm set-memory [--auto-adj] size[unit] ldom
```

- **--auto-adj option**
This option adjusts the memory size to the boundary of 256 MB.
- **size**
Specify the size of memory to assign. The minimum assignment unit is 256 MB.
- **unit**
Specify the unit of size. You can specify the following units: G (gigabyte), M (megabyte), and K (kilobyte). The specification is case-insensitive.
- **ldom**
Specify the logical domain. If it is the control domain, "primary" is the fixed setting.

ldm remove-memory command

Use the ldm remove-memory command to remove virtual memory from a logical domain.

```
# ldm remove-memory [--auto-adj] size[unit] ldom
# ldm remove-memory -g resource-group [-s size[unit]] ldom
```

- **--auto-adj option**
This option adjusts the memory size to the boundary of 256 MB.
- **size**

Specify a memory size.

- **unit**
Specify the unit of size. You can specify the following units: G (gigabyte), M (megabyte), and K (kilobyte). The specification is case-insensitive.
- **-g resource-group**
This option removes memory by specifying a resource group. The SPARC M12/M10 system supports this option for Oracle VM Server for SPARC 3.3 and later.
- **ldom**
Specify the logical domain name.

The following example deletes 1 GB of virtual memory.

```
# ldm remove-memory 1g primary
```

```
-----  
Notice: The primary domain is in the process of a delayed  
reconfiguration.  
Any changes made to the primary domain will only take effect  
after it reboots.  
-----
```

ldm grow-socket command

Use the `ldm grow-socket` command to add virtual memory associated with a CPU socket to a logical domain. This subcommand is supported by Oracle VM for SPARC 3.2 and later.

```
# ldm grow-socket memory=size[unit] socket_id=id ldom
```

- **size**
Specify the size of memory to add. The smallest unit is 256 MB.
- **unit**
Specify the unit of size. You can specify G (gigabyte), M (megabyte), or K (kilobyte). The specification is case-insensitive.
- **id**
Specify the CPU socket ID.
- **ldom**
Specify the logical domain.

The following example adds 1 GB of virtual memory to the control domain from CPU socket#2.

```
# ldm grow-socket memory=1G socket_id=2 primary
```

ldm shrink-socket command

Use the `ldm shrink-socket` command to remove virtual memory associated with a

CPU socket from the logical domain. This subcommand is supported by Oracle VM for SPARC 3.2 and later.

```
# ldm shrink-socket memory=size[unit] socket_id=id ldom
```

- size
Specify the size of memory to remove. The smallest unit is 256 MB.
- unit
Specify the unit of size. You can specify G (gigabyte), M (megabyte), or K (kilobyte). The specification is case-insensitive.
- id
Specify the CPU socket ID to remove.
- ldom
Specify the logical domain.

The following example removes 1 GB of virtual memory associated with CPU socket#2 from the control domain.

```
# ldm shrink-socket memory=1G socket_id=2 primary
```

3.2.15 Setting CPU Socket Restrictions

Use the `ldm set-socket` command to set CPU socket constraints. CPU socket constraints are used to assign virtual CPUs, cores, and memory only from the specified CPU socket. This subcommand is supported by Oracle VM for SPARC 3.3 and later.

```
# ldm set-socket [-f] [--remap] socket_id=[id[,id[,...]]] ldom
```

- -f
This option clears constraints on all the existing virtual CPUs, cores, and memory for inactive logical domains. If the specified CPU socket for a bound or active domain has fewer resources than those defined by constraints, the constraints on resources and/or the degraded CPU socket may be modified.
- --remap
This option moves virtual resources assigned to an active physical resource to another physical resource while the domain that has the virtual resources is operating.
- id
Specify the CPU socket ID.

The following example sets CPU socket#2 constraints on logical domain `ldom1`.

```
# ldm set-socket socket_id=2 ldom1
```

3.2.16 Setting Logical Domain Configuration Information

Use the following commands to set logical domain configuration information.

Table 3-20 Commands for Setting Logical Domain Configuration Information

Command	Functional Outline
ldm add-spconfig	Saves logical domain configuration information.
ldm set-spconfig	Switches logical domain configuration information.
ldm remove-spconfig	Deletes logical domain configuration information.

ldm add-spconfig command

Use the ldm add-spconfig command to save logical domain configuration information.

```
# ldm add-spconfig config-name
```

- config-name
Specify a configuration name.

Excluding the default values at factory shipment (configuration name: factory-default), you can save up to seven sets of configuration information on the XSCF. The last saved configuration information becomes the default, which is used at the next and subsequent startups.

The following example saves configuration information with the name config1.

```
# ldm add-spconfig config1
```

ldm set-spconfig command

Use the ldm set-spconfig command to switch logical domain configuration information.

```
# ldm set-spconfig config-name
```

- config-name
Specify a configuration name. At the next start time, the logical domain starts based on the corresponding logical domain configuration information.

ldm remove-spconfig command

Use the ldm remove-spconfig command to delete logical domain configuration information from the XSCF.

```
# ldm remove-spconfig config-name
```

- **config-name**
Specify a configuration name. The corresponding logical domain configuration information is deleted.

3.2.17 Creating a Logical Domain

Use the `ldm add-domain` command to create a guest domain.

```
# ldm add-domain ldom
```

- **ldom**
Specify the logical domain name.

3.2.18 Configuring an I/O Device

Use the following commands to configure an I/O device.

Table 3-21 Commands for Configuring an I/O Device

Command	Functional Outline
<code>ldm add-io</code>	Adds an I/O device.
<code>ldm remove-io</code>	Deletes an I/O device.

Note that I/O devices can be reconfigured dynamically depending on the version of Oracle VM Server for SPARC. For details, see [Table 1-2](#), the latest *Product Notes* of the server used, the *Oracle VM Server for SPARC Release Notes* of the version used, and the *Oracle VM Server for SPARC Administration Guide*.

ldm add-io command

Use the `ldm add-io` command to add an I/O device to a logical domain.

```
# ldm add-io (bus | device | vf-name) ldom
```

- **bus**
Specify the PCIe bus (root complex).
- **device**
Specify a name (PCIe endpoint device name) for a direct I/O-assignable device.
- **vf-name**
Specify a virtual function name of the PCIe SR-IOV.
- **ldom**
Specify a logical domain to which a bus, device, or SR-IOV virtual function is added.

ldm remove-io command

Use the ldm remove-io command to delete an I/O device from a logical domain.

```
# ldm remove-io [-n] (bus | device | vf-name) ldom
```

- -n
Checks in advance whether the command is successfully executed.
- bus
Specify the PCIe bus (root complex).
- device
Specify a name (PCIe endpoint device name) for a direct I/O-assignable device.
- vf-name
Specify a virtual function name of the PCIe SR-IOV.
- ldom
Specify a logical domain from which a bus, device, or SR-IOV virtual function is removed.

The following example shows the procedure for deleting a PCIe bus (PCIE1) from the control domain (primary) and assigning it to an I/O domain. Also, the procedure is the same for the PCIe endpoint.

Note - With the default values at factory shipment (factory-default), all the I/O devices are assigned to the primary domain.

1. **Execute the fmadm faulty command of Oracle Solaris and the showstatus command of the XSCF firmware to confirm that the device has no error information.**

```
# fmadm faulty
```

```
XSCF> showstatus
```

If it has error information, execute the fmadm repair command to clear the error information after performing hardware maintenance work.

```
# fmadm repaired fmri | lable
```

2. **Set the control domain to delayed reconfiguration state.**
If the I/O device is reconfigured dynamically, you can omit this procedure.

```
# ldm start-reconf primary
```

3. **Release the assigned PCIe endpoint (PCIE1) from the control domain.**

```
# ldm remove-io PCIE1 primary
```

4. **Restart the control domain.**

If the I/O device is reconfigured dynamically, you can omit this procedure.

```
# shutdown -y -g0 -i6
```

5. **If Oracle Solaris at the assignment destination I/O domain is running, stop it.**

If the I/O device is reconfigured dynamically, you can omit this procedure.

```
# ldm stop-domain ldom1
```

6. **Add the assignment of the PCIe endpoint (PCIE1) to the I/O domain.**

```
# ldm add-io PCIE1 ldom1
```

7. **Start Oracle Solaris of the assignment destination I/O domain.**

If the I/O device is reconfigured dynamically, you can omit this procedure.

```
# ldm start-domain ldom1
```

3.2.19 Creating or Destroying the SR-IOV Virtual Function

Use the following commands to create or destroy the SR-IOV virtual function.

Table 3-22 Commands for Creating or Destroying the SR-IOV Function

Command	Functional Outline
ldm create-vf	Creates a virtual function (VF).
ldm destroy-vf	Destroys a virtual function (VF).

Note that I/O devices can be reconfigured dynamically depending on the version of Oracle VM Server for SPARC. For details, see [Table 1-2](#), the *Product Notes* of the server used, the *Oracle VM Server for SPARC Release Notes* of the version used, and the *Oracle VM Server for SPARC Administration Guide*.

Creating the SR-IOV virtual function

Use the `ldm create-vf` command to create a virtual function (VF).

ldm create-vf command

```
Creating multiple virtual functions (VFs) together
# ldm create-vf -n number | max pf-name
```

```
Creating one virtual function (VF) at a time
# ldm create-vf pf-name
```

- -n
Specify the number of virtual functions (VFs).
- pf-name
The PCIe device name that supports the SR-IOV function

The following example shows the procedure for creating the SR-IOV virtual function (VF) and adding a virtual device to the I/O domain.

1. **Check the PCIe device that supports the SR-IOV function.**

In the following execution example, the PCIe device names "/BB0/PCI0/IOVNET.PF0" and "/BB0/PCI0/IOVNET.PF1" out of the mounted PCIe cards are applicable and assigned to the control domain.

# ldm list-io				
NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	primary	IOV
PCIE2	BUS	PCIE2	primary	IOV
PCIE3	BUS	PCIE3	primary	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	primary	IOV
PCIE6	BUS	PCIE6	primary	IOV
PCIE7	BUS	PCIE7	primary	IOV
(Omitted)				
/BB0/PCI6	PCIE	PCIE6	primary	EMP
/BB0/PCI9	PCIE	PCIE7	primary	EMP
/BB0/PCI10	PCIE	PCIE7	primary	EMP
/BB0/PCI0/IOVNET.PF0	PF	PCIE1	primary	
/BB0/PCI0/IOVNET.PF1	PF	PCIE1	primary	
/BB0/CMUL/NET0/IOVNET.PF0	PF	PCIE0	primary	
/BB0/CMUL/NET0/IOVNET.PF1	PF	PCIE0	primary	
/BB0/CMUL/NET2/IOVNET.PF1	PF	PCIE4	primary	
/BB0/CMUL/NET2/IOVNET.PF0	PF	PCIE4	primary	

2. **Set the control domain to delayed reconfiguration state.**

If the I/O device is reconfigured dynamically, this is not necessary. Go to step 3.

```
# ldm start-reconf primary
```

3. Create a virtual function (VF).

```
# ldm create-vf /BB0/PCI0/IOVNET.PF0
# ldm create-vf /BB0/PCI0/IOVNET.PF1
```

4. Restart the control domain.

If the I/O device is reconfigured dynamically, this is not necessary. Go to step 6.

```
# shutdown -y -g0 -i6
```

5. Stop any running Oracle Solaris in the assignment destination I/O domains (ldom00 and ldom01).

If the I/O device is reconfigured dynamically, this is not necessary. Go to step 6.

```
# ldm stop-domain ldom00
# ldm stop-domain ldom01
```

6. Add the virtual functions (/BB0/PCI0/IOVNET.PF0.VF0 and /BB0/PCI0/IOVNET.PF1.VF1) to the I/O domain.

```
# ldm add-io /BB0/PCI0/IOVNET.PF0.VF0 ldom00
# ldm add-io /BB0/PCI0/IOVNET.PF1.VF0 ldom01
```

7. Start Oracle Solaris of the I/O domain.

If the I/O device is reconfigured dynamically, this is not necessary.

```
# ldm start-domain ldom00
# ldm start-domain ldom01
```

Destroying the SR-IOV virtual function

Use the `ldm destroy-vf` command to destroy a virtual function (VF).

`ldm destroy-vf` command

- Destroying multiple virtual functions (VFs) at a time

```
# ldm destroy-vf -n number | max pf-name
```

- Destroying a specified virtual function (VF)

```
# ldm destroy-vf vf-name
```

- `-n`
Specify the number of virtual functions (VFs).

- pf-name
The PCIe device name that supports the SR-IOV function
- vf-name
Name of a virtual function (VF)

The following example shows the procedure for disconnecting a virtual device from the I/O domain and destroying the SR-IOV virtual function (VF).

1. **Check the usage status of I/O devices in logical domains.**

In the following execution example, the virtual devices "/BB0/PCI0/IOVNET.PF0.VF0" and "/BB0/PCI0/IOVNET.PF0.VF1" are assigned to the I/O domains (iodom00 and iodom01).

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	----	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	primary	IOV
PCIE2	BUS	PCIE2	primary	IOV
PCIE3	BUS	PCIE3	primary	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	primary	IOV
PCIE6	BUS	PCIE6	primary	IOV
PCIE7	BUS	PCIE7	primary	IOV
(Omitted)				
/BB0/PCI6	PCIE	PCIE6	primary	EMP
/BB0/PCI9	PCIE	PCIE7	primary	EMP
/BB0/PCI10	PCIE	PCIE7	primary	EMP
/BB0/PCI0/IOVNET.PF0	PF	PCIE1	primary	
/BB0/PCI0/IOVNET.PF1	PF	PCIE1	primary	
/BB0/CMUL/NET0/IOVNET.PF0	PF	PCIE0	primary	
/BB0/CMUL/NET0/IOVNET.PF1	PF	PCIE0	primary	
/BB0/CMUL/NET2/IOVNET.PF1	PF	PCIE4	primary	
/BB0/CMUL/NET2/IOVNET.PF0	PF	PCIE4	primary	
/BB0/PCI0/IOVNET.PF0.VF0	VF	PCIE1	iodom00	
/BB0/PCI0/IOVNET.PF0.VF1	VF	PCIE1	iodom01	

2. **Stop any started Oracle Solaris in the I/O domain to which the virtual function is assigned.**

If the I/O device is reconfigured dynamically, this is not necessary. Go to step 3.

```
# ldm stop-domain iodom00
# ldm stop-domain iodom01
```

3. **From the I/O domain, release the assignment of the virtual functions (/BB0/PCI0/IOVNET.PF0.VF0 and /BB0/PCI0/IOVNET.PF0.VF1).**

```
# ldm remove-io /BB0/PCI0/IOVNET.PF0.VF0 iodom00
# ldm remove-io /BB0/PCI0/IOVNET.PF0.VF1 iodom01
```

4. **Start Oracle Solaris of the I/O domain.**

If the I/O device is reconfigured dynamically, this is not necessary. Go to step 6.

```
# ldm start-domain ldom00
# ldm start-domain ldom01
```

5. **Set the control domain to delayed reconfiguration state.**
If the I/O device is reconfigured dynamically, this is not necessary. Go to step 6.

```
# ldm start-reconf primary
```

6. **Destroy a virtual function (VF).**

```
# ldm destroy-vf /BB0/PCI0/IOVNET.PF0
# ldm destroy-vf /BB0/PCI0/IOVNET.PF1
```

7. **Restart the control domain.**
If the I/O device is reconfigured dynamically, this is not necessary.

```
# shutdown -y -g0 -i6
```

3.2.20 Configuring a Virtual Network Device

Use the following commands to configure a virtual network device.

Table 3-23 Commands for Configuring a Virtual Network Device

Command	Functional Outline
ldm add-vnet	Adds a virtual network device.
ldm remove-vnet	Removes a virtual network device.

ldm add-vnet command

Use the ldm add-vnet command to add a virtual network device to a logical domain.

```
# ldm add-vnet if-name vswitch-name ldom
```

- if-name
Specify the name of the interface assigned to the instance of the virtual network device.
- vswitch-name
Specify a virtual switch service name.
- ldom

Specify the logical domain name.

ldm remove-vnet command

Use the ldm remove-vnet command to remove a virtual network device.

```
# ldm remove-vnet if-name ldom
```

- if-name
Specify an interface name.
- ldom
Specify the logical domain name.

3.2.21 Configuring a Virtual Disk Server

Use the following commands to configure a virtual disk server.

Table 3-24 Commands for Configuring a Virtual Disk Server

Command	Functional Outline
ldm add-vdiskserverdevice	Adds a device volume to a virtual disk server.
ldm remove-vdiskserverdevice	Removes a device volume from a virtual disk server.

ldm add-vdiskserverdevice command

Use the ldm add-vdiskserverdevice command to add a volume to a virtual disk server.

```
# ldm add-vdiskserverdevice backend volume-name@service-name
```

- backend
Specify a physical disk path. In addition to a file, you can specify the path of a disk, disk slice, etc. for backend.
- volume-name
Specify a volume name that is unique within the virtual disk server.
- service-name
Specify a virtual disk service name.

The following example creates and adds a file of 100 GB as a volume of the disk server primary-vds0.

```
# mkfile 100g /ldoms/domain/test/fdisk1
# ldm add-vdiskserverdevice /ldoms/domain/test/fdisk1 vol2@primary-vds0
```

ldm remove-vdiskserverdevice command

Use the ldm remove-vdiskserverdevice command to delete a volume from a virtual disk server.

```
# ldm remove-vdiskserverdevice volume-name@service-name
```

- volume-name
Specify a volume name.
- service-name
Specify a virtual disk server name.

The following example deletes the volume vol2 from the virtual disk server primary-vds0.

```
# ldm remove-vdiskserverdevice vol2@primary-vds0
```

3.2.22 Configuring a Virtual Disk

Use the following commands to configure a virtual disk.

Table 3-25 Commands for Configuring a Virtual Disk

Command	Functional Outline
ldm add-vdisk	Adds a virtual disk.
ldm remove-vdisk	Removes a virtual disk.

ldm add-vdisk command

Use the ldm add-vdisk command to add a virtual device as a virtual disk to a guest domain.

```
# ldm add-vdisk disk-name volume-name@service-name ldom
```

- disk-name
Specify a virtual disk name.
- volume-name
Specify the device name of the existing virtual disk server to be connected.
- service-name
Specify the name of the existing virtual disk server to be connected.
- ldom
Specify the logical domain.

The following example adds the added device vol2 of the virtual disk server

primary-vds0 as the virtual disk vdisk1 to the guest domain ldom1.

```
# ldm add-vdisk vdisk1 vol2@primary-vds0 ldom1
```

ldm remove-vdisk command

Use the ldm remove-vdisk command to remove a virtual disk from a guest domain.

```
# ldm remove-vdisk disk-name ldom
```

- **disk-name**
Specify a virtual disk name.
- **ldom**
Specify the logical domain name.

The following example removes the virtual disk vdisk2 from the guest domain ldom1.

```
# ldm remove-vdisk vdisk2 ldom1
```

3.2.23 Configuring a Virtual Console

Use the ldm set-vconsole command to assign a virtual console to a guest domain and to set the output of the guest domain console log to a file.

```
# ldm set-vconsole port=port-num ldom  
# ldm set-vconsole [service=vcc-server] log=on ldom
```

- **port-num**
Specify the port number of the virtual console.
- **ldom**
Specify the logical domain name.
- **vcc-server**
If the virtual console terminal collection and distribution unit service is specified for a logical domain other than the control domain, specify the name of that logical domain. As a result, the console output and log of the guest domain specified for *ldom* are output to the specified logical domain.

Note - Logical domain specification by `service=vcc-server` is supported when the virtual console terminal distribution unit service is provided on Oracle Solaris 11.

Note - The log contents are output as a file named `/var/log/vntsd/(guest domain name)/console-log` in the service domain specified by `ldm add-vconscon`.

3.2.24 Configuring a Startup Device

Use the `ldm set-variable` command to set the startup device of a guest domain. Also use the `ldm set-variable` command to set whether to automatically start Oracle Solaris at the same time that a guest domain starts. By default, Oracle Solaris automatically starts together with a guest domain.

```
# ldm set-variable boot-device=disk-name ldom
# ldm set-variable auto-boot?={true|false} ldom
```

- `disk-name`
Specify a virtual disk name.
- `ldom`
Specify the logical domain name.
- `true|false`
Specify "true" for automatic startup. Specify "false" for no automatic startup.

Note - Uppercase characters cannot be used for the device alias of OpenBoot PROM. If the name of a virtual disk added by the `ldm add-vdisk` command contains an uppercase character, change all the letters of the virtual disk name to lowercase. Then, set it in the environment variable `boot-device` of OpenBoot PROM.

3.2.25 Binding Resources

Use the `ldm bind-domain` command to bind a resource to a guest domain.

```
# ldm bind-domain ldom
```

- `ldom`
Specify the logical domain name.

3.2.26 Starting a Guest Domain

Use the `ldm start-domain` command to start a guest domain.

```
# ldm start-domain ldom
```

- `ldom`
Specify the logical domain name.

3.2.27 Specifying a Shutdown Group

Use the `ldm set-domain` command to specify a shutdown group.

```
# ldm set-domain shutdown-group=num ldom
```

- `num`
Specify a shutdown group. The number is a value from 1 to 15.
- `ldom`
Specify the name of the target logical domain. You cannot specify "primary" (control domain).

Note - You can change the shutdown group only while the guest domain is in the inactive or bound state.

3.2.28 Reconfiguring a Device

Use the `devfsadm` command when reconfiguring a device. Log in to the guest domain to which the virtual disk is assigned, and execute the `devfsadm` command.

```
guest# devfsadm
```

The `devfsadm` command loads all drivers into the system to connect to as many devices as possible. Then, it creates a device file in the `/devices` directory and a logical link in the `/dev` directory. For details of the `devfsadm` command, see the reference manual of Oracle Solaris.

3.2.29 Enabling Recovery Mode

To enable recovery mode, execute the `svccfg` command to set the `ldmd/recovery_mode` property to `auto`. Then, execute the `svcadm` command to enable the setting value.

Note that recovery mode is enabled by default in Oracle VM Server for SPARC 3.3 and later.

```
# svccfg -s ldmd setprop ldmd/recovery_mode = astring: auto
# svcadm refresh ldmd
```

From XCP 2230, the recovery mode provided from the Oracle VM Server for SPARC 3.1 is supported.

For details on recovery mode, see "Handling Hardware Errors" in the *Oracle VM Server for SPARC Administration Guide* of the version used. In addition, for any

correction required for recovery mode, see the *Oracle VM Server for SPARC Release Notes* of the version used.

Configuring a Physical Partition

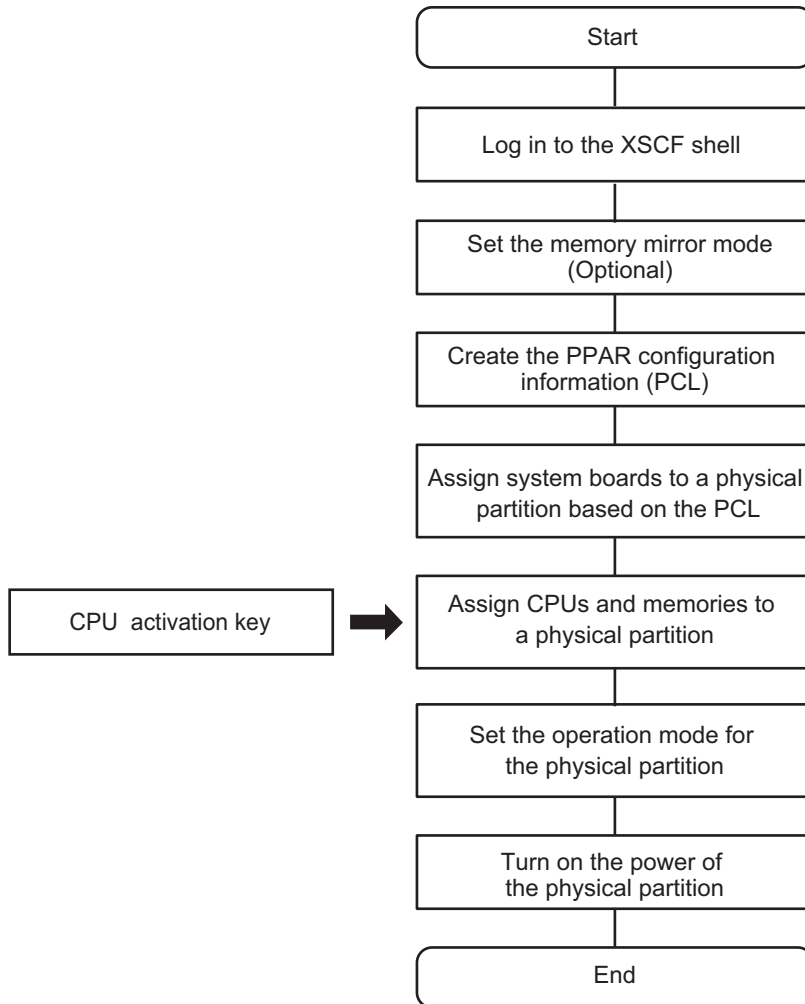
This chapter describes the configuring of a physical partition.

- [Physical Partition Configuration Flow](#)
- [Configuring and Operating a Physical Partition](#)

4.1 Physical Partition Configuration Flow

To configure a domain, first configure each physical partition. Use the XSCF firmware to configure the physical partition. [Figure 4-1](#) shows the physical partition configuration flow in a building block configuration.

Figure 4-1 Configuring a Physical Partition



4.2 Configuring and Operating a Physical Partition

This section describes points to consider when configuring a physical partition, and a configuration example.

- How to determine the recommended physical partition number (PPAR-ID)
You can specify any number from 0 to 15, depending on the system configuration, as a PPAR-ID. However, the BB-ID of a chassis mounted in the SPARC M12/M10 system must be specified for the PPAR-ID. If the BB-ID of an unmounted chassis

in the SPARC M12/M10 system is specified for the PPAR-ID, the physical partition will fail to power on and cannot be used.

For example, when the BB-IDs of the SPARC M12-2S/M10-4S units mounted on the system are BB#00, BB#01, BB#02, and BB#03, you can specify 0, 1, 2, or 3 as the PPAR-ID when configuring a physical partition.

For efficient use of resources after the start of operation, you may remove a SPARC M12-2S/M10-4S unit. If the BB-ID of the SPARC M12-2S/M10-4S unit to be removed is the same number as a PPAR-ID in the system, you need to stop the physical partition. In that case, physical partition reconfiguration may be required. Therefore, we recommend considering the following when configuring physical partitions.

[For a 2BB configuration]

- When configuring a system with only one physical partition, set PPAR-ID 0. You can remove BB#01, which is a different number from the PPAR-ID.
- When configuring a system with two physical partitions, assign BB#00 to PPAR-ID 0 and BB#01 to PPAR-ID 1.

In this way, you can remove a SPARC M12-2S/M10-4S unit without stopping the physical partition of the SPARC M12-2S/M10-4S unit that should be kept in the system.

Example: Configuring two physical partitions

[Recommended pattern]

Assign BB#00 to PPAR-ID 0 and BB#01 to PPAR-ID 1.

When a SPARC M12-2S/M10-4S unit is removed, the physical partition with a different number from the BB-ID of the removed unit is not affected.

PPAR-ID 0 [BB#00]

PPAR-ID 1 [BB#01]

[Non-recommended pattern]

Assign BB#01 to PPAR-ID 0 and BB#00 to PPAR-ID 1. When a SPARC M12-2S/M10-4S unit is removed, the physical partition of the unremoved unit also must be stopped.

As a result, physical partition reconfiguration is required.

PPAR-ID 0 [BB#01]

PPAR-ID 1 [BB#00]

[For a 3BB or larger configuration with directly interconnected chassis]

- When configuring a system with only one physical partition, set PPAR-ID 0. You can remove a SPARC M12-2S/M10-4S unit with a BB-ID different from the PPAR-ID.
- When configuring a system with two or more physical partitions, do not set, as a PPAR-ID, the BB-ID of any SPARC M12-2S/M10-4S unit that may be removed (*1). So even when that SPARC M12-2S/M10-4S unit is removed, each physical partition with a different number from the BB-ID of the removed unit is not affected.

*1 The master and standby SPARC M12-2S/M10-4S units must be kept in the system after reduction. Therefore, the SPARC M12-2S/M10-4S unit that may be removed should have a BB-ID other than BB#00 or BB#01.

In this way, you can remove the SPARC M12-2S/M10-4S unit without stopping the physical partition of the SPARC M12-2S/M10-4S unit that should be kept in the system.

Example: Configuring two physical partitions in a 4BB configuration with directly interconnected chassis

[Recommended pattern 1: Supposing the reduction of resources of each physical partition]

Assign BB#00 and BB#02 to PPAR-ID 0 and BB#01 and BB#03 to PPAR-ID 1. You can remove the SPARC M12-2S/M10-4S unit of BB#02 or BB#03, each of which is a different number from the PPAR-ID, without stopping either physical partition.

PPAR-ID 0 [BB#00 and BB#02]
PPAR-ID 1 [BB#01 and BB#03]

[Recommended pattern 2: Supposing the reduction of the number of physical partitions]

Assign BB#00 and BB#01 to PPAR-ID 0 and BB#02 and BB#03 to PPAR-ID 2. You can remove the SPARC M12-2S/M10-4S unit of BB#02 or BB#03 without stopping the physical partition of the SPARC M12-2S/M10-4S unit that should be kept in the system.

PPAR-ID 0 [BB#00 and BB#01]
PPAR-ID 2 [BB#02 and BB#03]

[Non-recommended pattern]

Assign BB#00 and BB#01 to PPAR-ID 0 and BB#02 and BB#03 to PPAR-ID 1. The SPARC M12-2S/M10-4S units at PPAR-ID 0 cannot be removed because they are the master and standby.

PPAR-ID 0 [BB#00 and BB#01]
PPAR-ID 1 [BB#02 and BB#03]

[For a configuration with connections via a crossbar box]

When configuring a physical partition in a 2BB configuration or larger, do not set, as a PPAR-ID, the BB-ID of a SPARC M12-2S/M10-4S unit that may be removed.

In this way, you can remove a SPARC M12-2S/M10-4S unit without stopping the physical partition of the SPARC M12-2S/M10-4S unit that should be kept in the system.

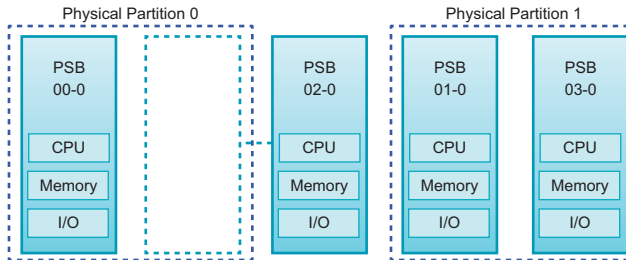
The remainder of this section uses examples to describe a specific procedure for configuring two physical partitions in a configuration with four SPARC M10-4S units.

- Assignment of PSB (BB)
Assign system boards (PSB 00-0 and PSB 02-0) to physical partition 0, and assign system boards (PSB 01-0 and PSB 03-0) to physical partition 1. However, keep PSB 02-0 in an assignable state instead of actually assigning it.

- Mirror mode setting (option)
Set memory mirroring only on PSB 00-0.
- Configuration policy / memory nullification / I/O nullification
In this example, set the configuration policy to "system" for physical partitions 0 and 1, and do not use memory nullification and I/O nullification.
- Physical partition operation mode
Set the message level of diagnosis messages to "normal."

Figure 4-2 is a conceptual diagram of configured physical partitions.

Figure 4-2 Conceptual Diagram of Configured Physical Partitions



1. **Log in to the XSCF shell with a user account that has the platadm privilege.**
2. **Execute the setupfru command to set memory mirroring.**

```
XSCF> setupfru -m y sb 00-0
```

3. **Execute the showfru command as required to check the status of memory mirroring.**
Check the status of memory mirroring on the system board (PSB 00-0). If "yes" appears in all of [Memory Mirror Mode], mirror mode is properly set.

```
XSCF> showfru sb 00-0
Device      Location      Memory Mirror Mode
sb          00-0
  cpu       00-0-0      yes
  cpu       00-0-1      yes
  cpu       00-0-2      yes
  cpu       00-0-3      yes
XSCF>
```

4. **Execute the setpcl command to create physical partition configuration information (PPAR configuration information).**
Map physical system boards PSB 00-0 and PSB 02-0 to logical system boards LSB 0 and LSB 1, respectively, of physical partition 0. Map physical system boards PSB 01-0 and PSB 03-0 to logical system boards LSB 0 and LSB 1, respectively, of physical partition 1. In this example, set the configuration policy to "system," and do not use memory nullification and I/O nullification.

```

XSCF> setpcl -p 0 -a 0=00-0 1=02-0
XSCF> setpcl -p 0 -s policy=system
XSCF> setpcl -p 0 -s no-mem=false
XSCF> setpcl -p 0 -s no-io=false
XSCF> setpcl -p 1 -a 0=01-0 1=03-0
XSCF> setpcl -p 1 -s policy=system
XSCF> setpcl -p 1 -s no-mem=false
XSCF> setpcl -p 1 -s no-io=false

```

5. **Execute the showpcl command to check the contents of the PPAR configuration information.**

[PPAR-ID] shows the physical partition number, [LSB] shows the logical system board number, and [PSB] shows the PSB number.

```

XSCF> showpcl -v -a
PPAR-ID  LSB   PSB   Status   No-Mem   No-IO   Cfg-policy
00
          00    00-0           Powered Off           System
          01    02-0           False    False
01
          01    02-0           False    False           System
          00    01-0           False    False
          01    03-0           False    False

```

6. **Execute the showboards command to check the PSB.**

The PSB displayed with "SP" under [PPAR-ID] and "Available" under [Assignment] can be assigned. "SP" represents the system board pool state, which means that the board does not belong to any physical partition.

```

XSCF> showboards -a
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0 SP          Available   n    n    n    Passed  Normal
01-0 SP          Available   n    n    n    Passed  Normal
02-0 SP          Available   n    n    n    Passed  Normal
03-0 SP          Available   n    n    n    Passed  Normal

```

7. **Execute the addboard command to assign the PSBs.**

```

XSCF> addboard -c assign -p 0 00-0
XSCF> addboard -c assign -p 1 01-0 03-0

```

8. **Execute the showboards command to check the PSB assigned status.**

When the [R] column displays "*" (asterisk) for an assigned PSB, and the physical partitions and logical system boards (LSBs) are set as shown by the setting in the [PPAR-ID] column, the PSB is correctly assigned.

```
XSCF> showboards -v -a
```

PSB	R	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	*	00(00)	Assigned	n	n	n	Passed	Normal
01-0	*	01(00)	Assigned	n	n	n	Passed	Normal
02-0		SP	Available	n	n	n	Passed	Normal
03-0	*	01(01)	Assigned	n	n	n	Passed	Normal

9. **Execute the showcodactivation command to check whether the CPU Activation key is registered in the system.**

If only the header is displayed, the CPU Activation key is not registered in the XSCF. To add the CPU Activation key, perform the procedure from step 10. If an index number is displayed, the CPU Activation key is registered in the XSCF. To continue the procedure, go to step 13.

```
XSCF> showcodactivation
```

Index	Description	Count
-----	-----	-----

10. **Execute the addcodactivation command to register the CPU Activation key.**

```
XSCF> addcodactivation "Product: SPARC M10-4S
SequenceNumber: 10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxeUDdPQHKbtXgVZPrtYguqiNUieB+mTDC
:
:
b1GCKFx1RH27FdVHiB2H0A=="
Above Key will be added, Continue?[y|n]:y
```

11. **Execute the showcodactivation command to check the contents of the added CPU Activation key.**

If executing the command displays the number of CPU Activations for the purchased resources, the CPU Activation key has been correctly registered.

```
XSCF> showcodactivation
```

Index	Description	Count
-----	-----	-----
0	PROC	2
1	PROC	2
2	PROC	2
3	PROC	2

12. **Execute the setcod command to assign the CPU core resources to the physical partitions.**

```

XSCF> setcod -p 0 -s cpu -c set 4
PROC Permits assigned for PPAR 0 : 0 -> 4

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.

XSCF> setcod -p 1 -s cpu -c set 4
PROC Permits assigned for PPAR 1 : 0 -> 4

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.

```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

XSCF> **setcod -s cpu**

13. **Execute the resetdateoffset command to reset the difference between the time managed by the XSCF and the time managed by physical partitions.**

```

XSCF> resetdateoffset -p 0
XSCF> resetdateoffset -p 1

```

14. **Execute the setpparmode command and set the detail level of diagnosis messages to "normal."**

```

XSCF> setpparmode -p 0 -m message=normal
XSCF> setpparmode -p 1 -m message=normal

```

15. **Execute the poweron command to power on the physical partitions.**

The following example powers on physical partitions 0 and 1.

```

XSCF> poweron -p 0
PPAR-IDs to power on:00
Continue? [y|n] :y
00 :Powering on

*Note*
This command only issues the instruction to power-on.
The result of the instruction can be checked by the "showlogs power".

XSCF> poweron -p 1
PPAR-IDs to power on:01
Continue? [y|n] :y
01 :Powering on

```

Note

This command only issues the instruction to power-on.

The result of the instruction can be checked by the "showlogs power".

16. **Execute the showboards command to check the PSB status.**

If "y" appears in all the [Pwr], [Conn], and [Conf] columns, the physical partition is operating correctly.

```
XSCF> showboards -a
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
----	-----	-----	----	----	----	-----	-----
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	01(00)	Assigned	y	y	y	Passed	Normal
02-0	SP	Available	n	n	n	Passed	Normal
03-0	01(01)	Assigned	y	y	y	Passed	Normal

Chapter 5

Logical Domain Configuration Example

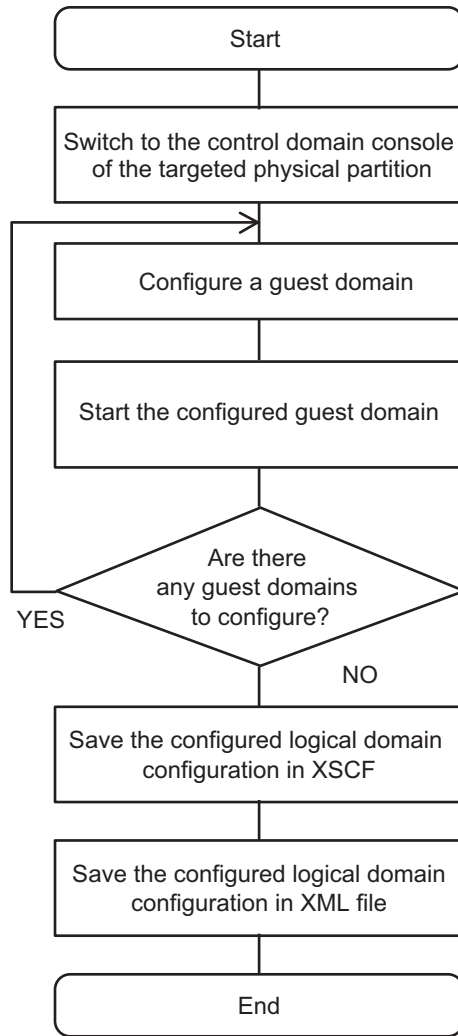
This chapter shows an example of configuring logical domains.

- [Logical Domain Configuration Flow](#)
- [Example of Logical Domain Configuration Operations](#)

5.1 Logical Domain Configuration Flow

After configuring and powering on the physical partition, configure each logical domain. Use the `ldm` command of Logical Domains Manager, which is management software for Oracle VM Server for SPARC, to configure the logical domain. [Figure 5-1](#) shows the logical domain configuration flow.

Figure 5-1 Configuring a Logical Domain



All the hardware resources on system boards are assigned to the control domain during physical partition configuration. When configuring the logical domains, assign the hardware resources to guest domains or various service domains as required.

After logical domain configuration is complete, save the logical domain configuration information to the XSCF. When you power on the physical partition, all the logical domains in the physical partition are started according to this configuration information.

5.2 Example of Logical Domain Configuration Operations

This section describes a specific example of logical domain configuration.

To configure a logical domain, you need to define various names and specify various resources and their numbers. Prepare and organize that information before starting work.

The following tables show the information used as examples in this section. [Table 5-1](#) lists examples of definitions of names. [Table 5-2](#) lists examples of resources assigned to the control domain. [Table 5-3](#) lists examples of resources assigned to a guest domain.

Table 5-1 Examples of Definitions of Names

Item	Name
Name of the virtual console terminal collection and distribution unit	primary-vcc0
Virtual disk server name	primary-vds0
Virtual switch service name	primary-vsw0
Virtual network device interface name	vnet1
Virtual device name	vol1
Virtual disk name	vdisk1
Guest domain name	ldom1
Name for logical domain configuration information	config1

Table 5-2 Examples of Resources Assigned to the Control Domain

Item	Value
Range for virtual console terminal port numbers	5000 to 5100
Device used with the virtual switch service	net0
Device used with the virtual disk service	/ldoms/domain/test/fdisk0
Number of assigned CPUs	4 cores
Size of assigned memory	8 GB

Table 5-3 Examples of Resources Assigned to a Guest Domain

Item	Value
Port number of the assigned virtual console terminal	5000
Number of assigned CPUs	2 cores
Size of assigned memory	4 GB

5.2.1 Logging In to the Control Domain

The descriptions in this section assume that the physical partition number of the target is 0.

1. **Log in to the XSCF shell with a user account that has the platadm privilege.**
2. **Execute the poweron command to power on physical partition 0.**

```
XSCF> poweron -p 0
PPAR-IDs to power on:00
Continue? [y|n] :y
00 :Powering on
*Note*
This command only issues the instruction to power-on.
The result of the instruction can be checked by the "showlogs power".
XSCF>
```

3. **Execute the console command to switch to the control domain console of physical partition 0.**

```
XSCF> console -p 0
Console contents may be logged. Connect to PPAR-ID 0?[y|n] :y
```

Note - If automatic startup is not set for Oracle Solaris, perform startup operation, and wait until the startup is completed.

4. **Log in to the control domain as a root user.**
5. **Execute the svcs ldmd command, and confirm that Logical Domains Manager is running.**

The service name of Logical Domains Manager is ldmd. If the [STATE] column displays "online", it is running.

```
# svcs ldmd
STATE  TIME      FMRI
online 16:25:31 svc:/ldoms/ldmd:default
```

Note - If Logical Domains Manager has not started, execute the `svcadm` command to start it. For details of the `svcadm` command, see the reference manual of Oracle Solaris.

5.2.2 Setting the Default Services

1. **Execute the `ldm add-vconscon` command to add the virtual console terminal collection and distribution unit service to the control domain.**
The following example adds the virtual console terminal collection and distribution unit with the port numbers 5000 to 5100 and the name `primary-vcc0` to the control domain.

```
# ldm add-vconscon port-range=5000-5100 primary-vcc0 primary
```

2. **Execute the `ldm add-vdiskserver` command to add a virtual disk server to the control domain.**
The following example adds the virtual disk server named `primary-vds0` to the control domain.

```
# ldm add-vdiskserver primary-vds0 primary
```

3. **Execute the `ldm add-vswitch` command to add the virtual switch service to the control domain.**
The following example adds the virtual switch service named `primary-vsw0` to the control domain, and uses `net0` as a device.

```
# ldm add-vswitch net-dev=net0 primary-vsw0 primary
```

4. **Execute the `ldm list-services` command to confirm the set services.**
The following example displays the services of the control domain.

```
# ldm list-services primary
```

5.2.3 Making Initial Settings for the Control Domain

1. **Execute the `ldm start-reconf` command to start the delayed reconfiguration mode.**
The control domain name is fixed at "primary".

```
# ldm start-reconf primary
```

2. **Execute the `ldm list-permits` command to check the number of CPU cores that can be assigned.**

```
# ldm list-permits
```

3. **Execute the `ldm set-core` command to assign a virtual CPU to the control domain.**

The following example assigns CPUs of 4 cores to the control domain.

```
# ldm set-core 4 primary
```

4. **Execute the `ldm set-memory` command to assign virtual memory.**

The following example assigns 8 GB of memory to the control domain.

```
# ldm set-memory 8g primary
```

5. **Execute the `ldm list-bindings` command to check the assignment status of resources.**

```
# ldm list-bindings primary
```

6. **Execute the `ldm add-spconfig` command to save the logical domain configuration to the XSCF.**

The following example saves logical domain configuration information with the name "config1".

```
# ldm add-spconfig config1
```

7. **Execute the `poweroff` and `poweron` commands of the XSCF firmware to power off the physical partition and power it on again.**

```
XSCF> poweroff -p 0  
XSCF> poweron -p 0
```

8. **After restarting the control domain, execute the `list-bindings` command to check the assignment status of resources.**

If the following command displays the same the assignment status as described in step 4, the settings were made correctly.

```
# ldm list-bindings primary
```

5.2.4 Configuring a Guest Domain

1. **Execute the `ldm list-devices` command to check the resources that can be assigned to a guest domain.**

The command displays resources not used by any domain.

```
# ldm list-devices
```

2. **Execute the `ldm add-domain` command to create the guest domain `ldom1`.**

```
# ldm add-domain ldom1
```

3. **Execute the `ldm list-permits` command to check the number of CPU cores that can be assigned.**

```
# ldm list-permits
```

4. **Execute the `ldm set-core` command to add the virtual CPUs of two cores.**

```
# ldm set-core 2 ldom1
```

5. **Execute the `ldm add-memory` command to add 4 GB of virtual memory.**

```
# ldm add-memory 4g ldom1
```

6. **Execute the `ldm add-vnet` command to add a virtual network device.**

```
# ldm add-vnet vnet1 primary-vsw0 ldom1
```

7. **Execute the `ldm add-vdiskserverdevice` command to add a device to the virtual disk server.**

The following example uses the `mkfile` command to create a normal file of 100 GB for use as a data storage location in the control domain. The file is added as a device.

```
# mkfile 100g /ldoms/domain/test/fdisk0
# ldm add-vdiskserverdevice /ldoms/domain/test/fdisk0 vol1@primary-vds0
```

8. **Execute the `ldm add-vdisk` command to add a virtual disk.**

Add the virtual disk added in step 6 to the guest domain.

```
# ldm add-vdisk vdisk1 vol1@primary-vds0 ldom1
```

9. **Execute the `ldm set-vconsole` command to assign a virtual console with the port number 5000.**

```
# ldm set-vconsole port=5000 ldom1
```

10. **Execute the `ldm set-variable` command to set the startup device of the guest domain.**

```
# ldm set-variable boot-device=vdisk1 ldom1
```

Note - Uppercase characters cannot be used for the device alias of OpenBoot PROM. If the name of a virtual disk added by the `ldm add-vdisk` command contains an uppercase character, change all the letters of the virtual disk name to lowercase. Then, set it in the environment variable `boot-device` of OpenBoot PROM.

11. **Execute the `ldm list-bindings` command to check the configuration of the guest domain.**

```
# ldm list-bindings ldom1
```

12. **Execute the `ldm bind-domain` and `ldm start-domain` commands to start the guest domain.**

```
# ldm bind-domain ldom1  
# ldm start-domain ldom1
```

5.2.5 Configuring Recovery Mode

Use the `svccfg` command and the `svcadm` command to configure recovery mode.

```
# svccfg -s ldmd setprop ldmd/recovery_mode = astring: auto  
# svcadm refresh ldmd
```

From XCP 2230, the recovery mode provided from the Oracle VM Server for SPARC 3.1 is supported.

For details on recovery mode, see "Handling Hardware Errors" in the *Oracle VM Server for SPARC Administration Guide* of the version used. In addition, for any correction required for recovery mode, see the *Oracle VM Server for SPARC Release Notes* of the version used.

5.2.6 Saving Logical Domain Configuration Information

Once all the guest domain configurations are complete, save the logical domain configuration information.

1. **Execute the `ldm add-spconfig` command to add the logical domain configuration to the XSCF.**

The following example saves the logical domain configuration information with the name "config2".

```
# ldm add-spconfig config2
```

2. **Execute the `ldm list-constraints` command to save the logical domain configuration information to the XML file.**

The following example saves the logical domain configuration information to the "config2.xml" file.

```
# ldm list-constraints -x > config2.xml
```


Physical Partition Reconfiguration

This chapter describes an example of reconfiguring a physical partition in a building block configuration.

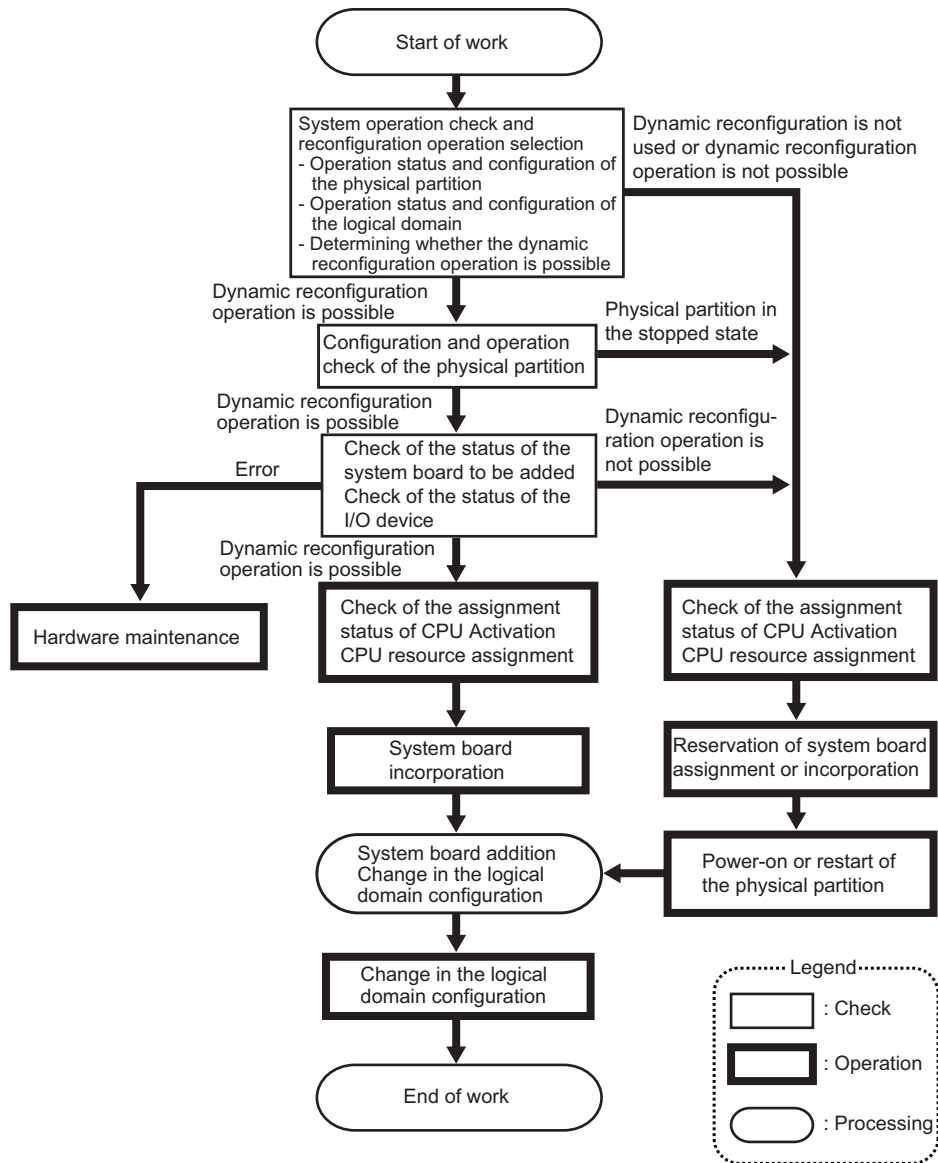
- [Flow for Physical Partition Reconfiguration](#)
- [Example of System Board Addition](#)
- [Example of Operations for Deleting a System Board](#)
- [Example of Operations for Moving a System Board](#)
- [Example of Operations for Replacing a System Board](#)
- [Considerations When Changing a Physical Partition Number](#)

6.1 Flow for Physical Partition Reconfiguration

This section describes the flow for physical partition reconfiguration operations such as system board (PSB) addition, deletion, move, and replacement, using flow diagrams.

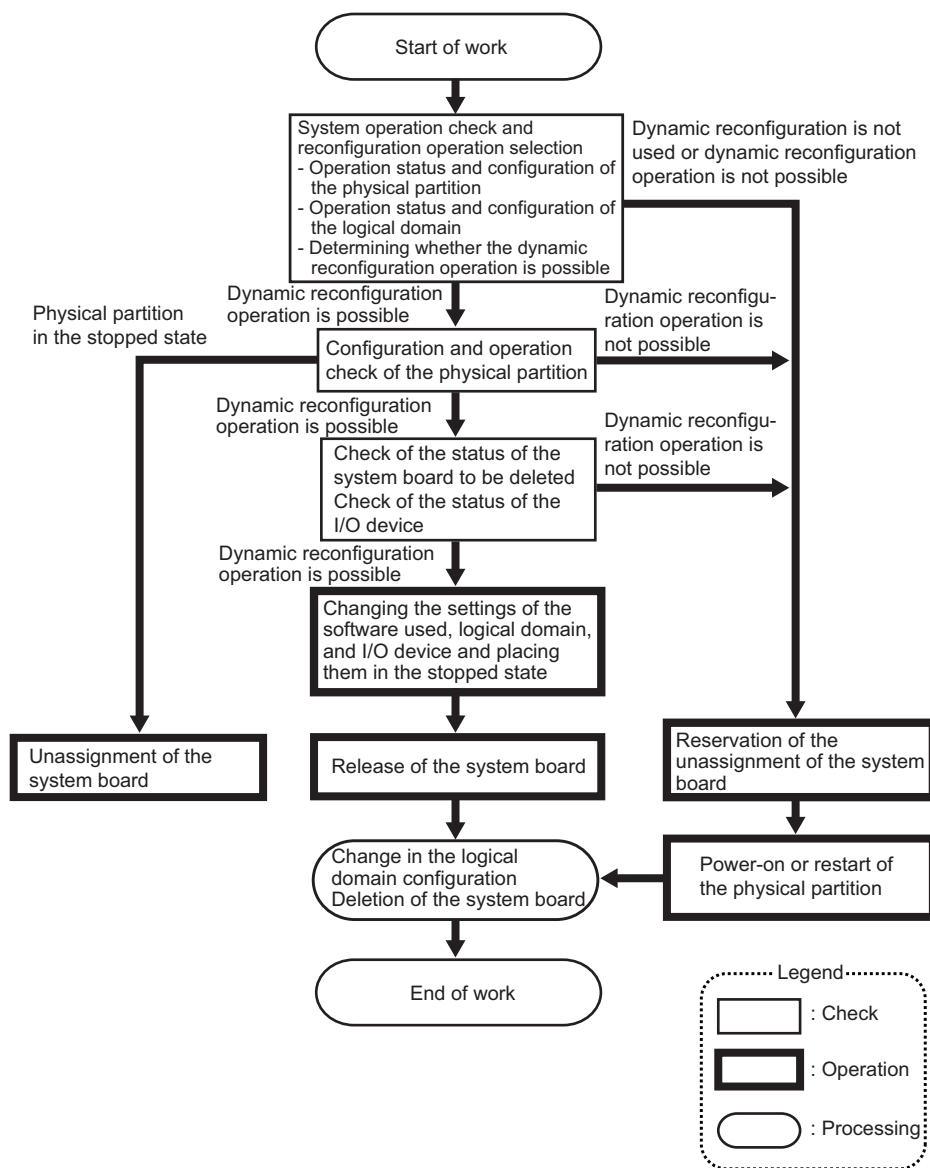
6.1.1 Flow for System Board Addition

Figure 6-1 Flow for System Board Addition



6.1.2 Flow for System Board Deletion

Figure 6-2 Flow for System Board Deletion



6.1.3 Flow for System Board Move

Figure 6-3 Flow for System Board Move (1)

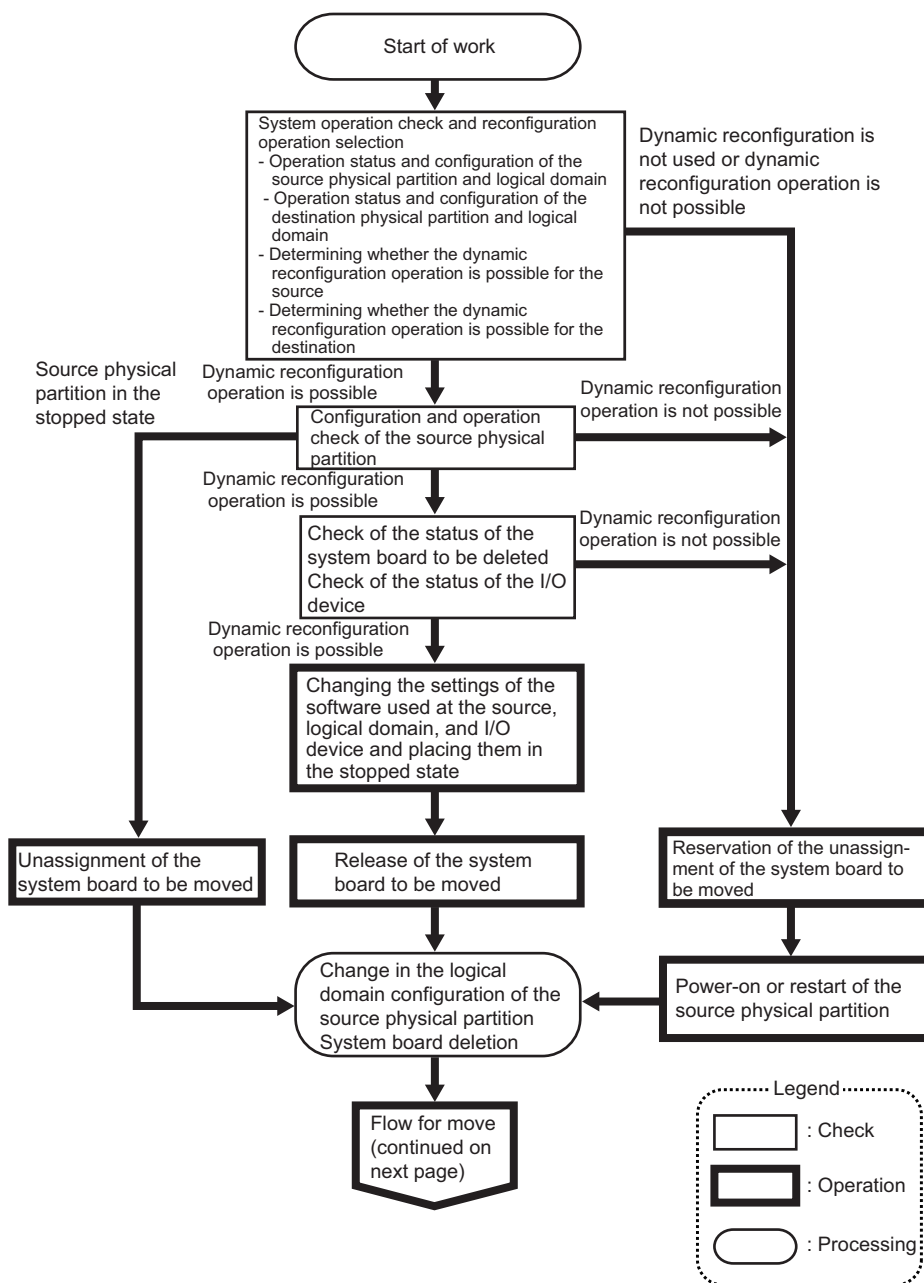
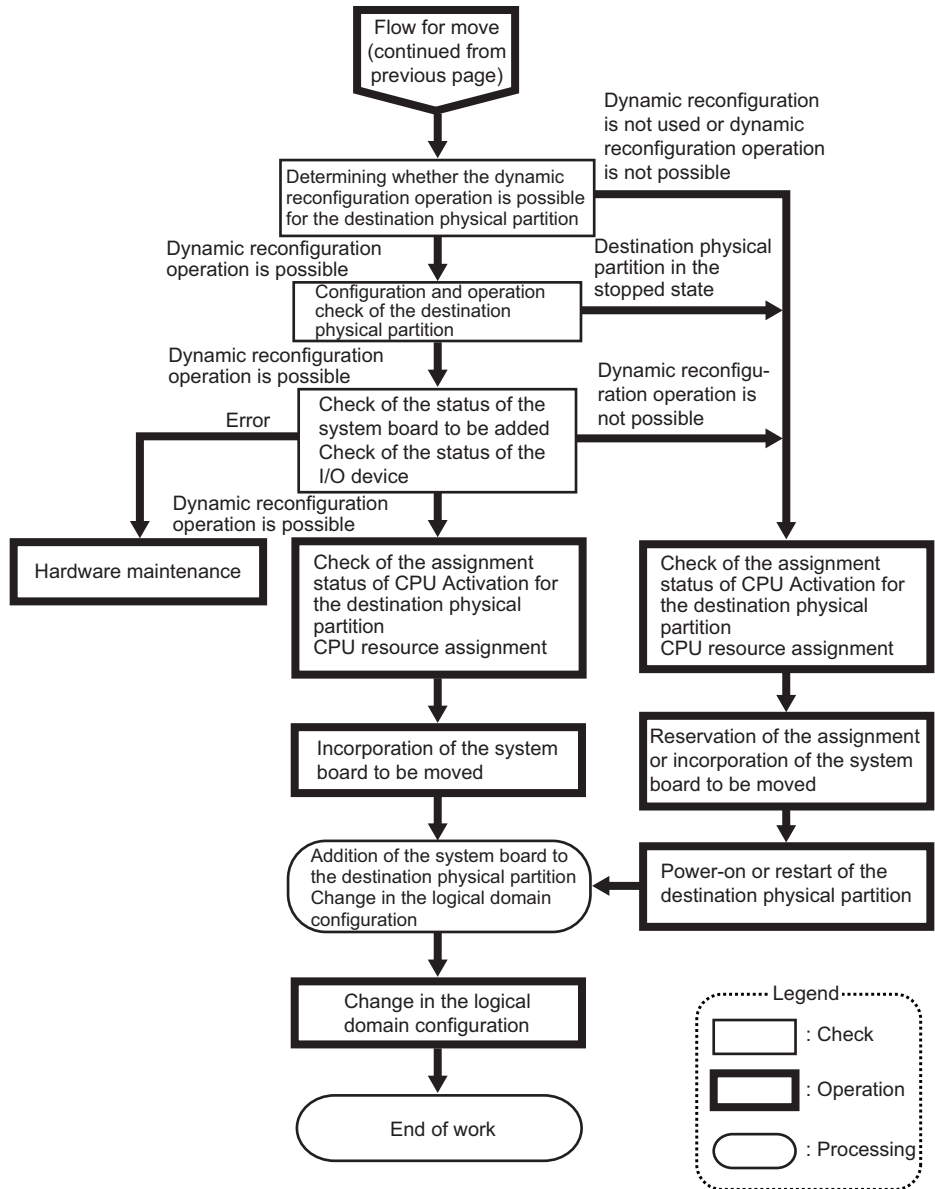


Figure 6-4 Flow for System Board Move (2)



6.1.4 Flow for System Board Replacement

Figure 6-5 Flow for System Board Replacement (1)

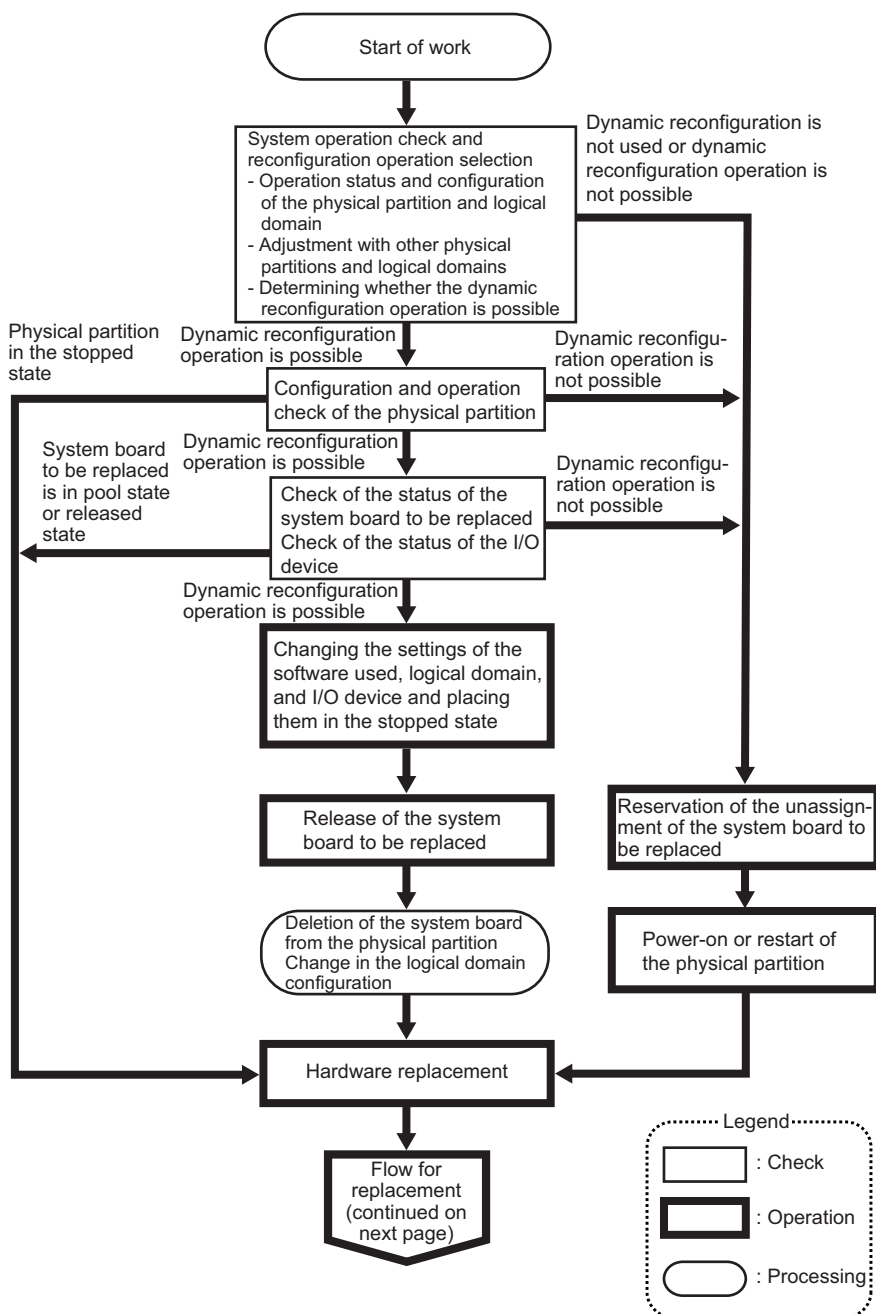
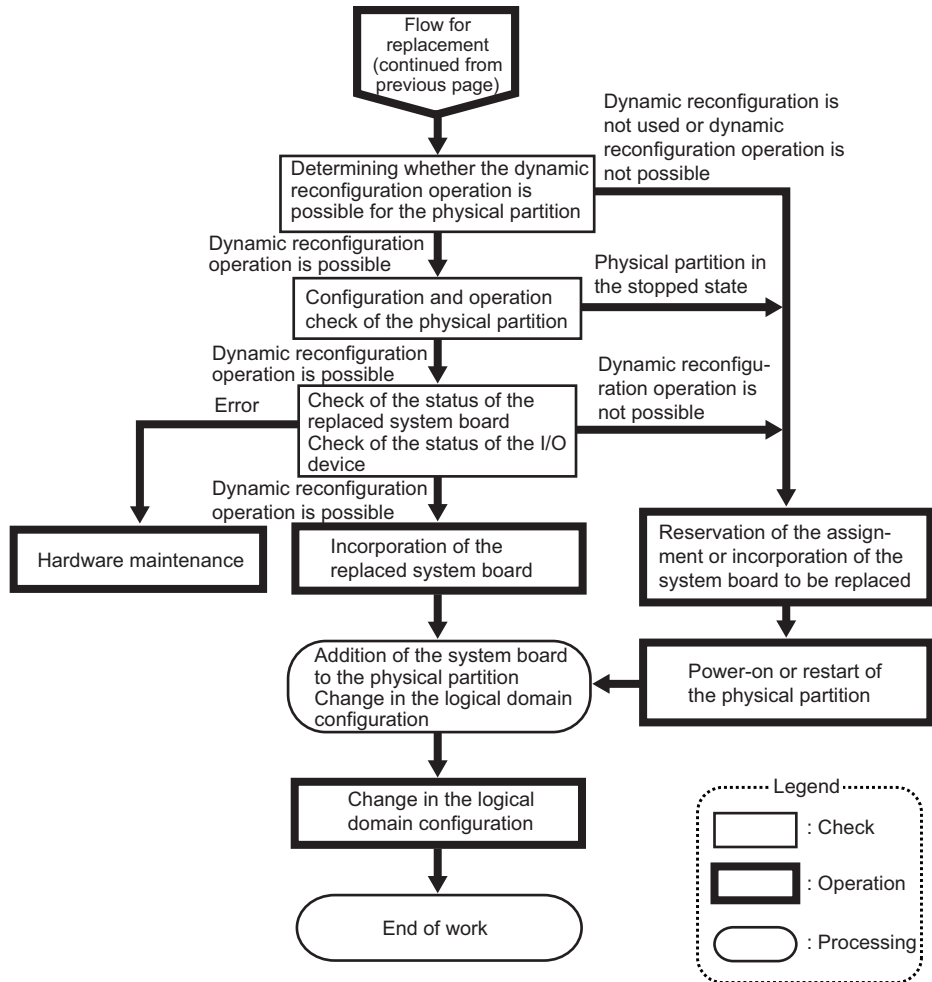


Figure 6-6 Flow for System Board Replacement (2)

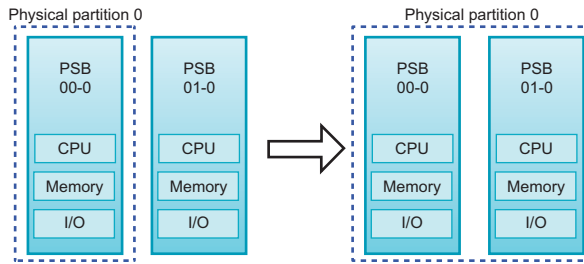


6.2

Example of System Board Addition

This section describes an example of the operations for adding a system board (PSB) as shown in [Figure 6-7](#).

Figure 6-7 Example of System Board Addition



6.2.1 Example of System Board Assignment

If the physical partition to which a system board (PSB) is added is in the stopped state, use the `addboard` command with the `-c` assign option specified to assign the PSB to the physical partition. Starting the physical partition after the assignment of the PSB incorporates the PSB into the physical partition.

The operation example of the SPARC M10-4S presented here uses XSCF shell commands. The same operations are also used for the SPARC M12-2S.

1. **Log in to the XSCF.**
2. **When you add a PSB for system expansion, use the `addfru` command to incorporate SPARC M10-4S into the system.**

Note - For details of system expansion, see "Chapter 8 Expanding a System With a Building Block Configuration" in the *SPARC M12-2S Installation Guide* or "Chapter 9 Expanding a System with a Building Block Configuration" in the *SPARC M10-4S Installation Guide*.

3. **Execute the `showpctl` command to check the current physical partition configuration information (PPAR configuration information).**

```
XSCF> showpctl -p 0
PPAR-ID   LSB   PSB   Status
00        00    00-0  Powered Off
```

4. **Execute the `setpctl` command to register the PSB in the PPAR configuration information.**

In the following example, PSB 01-0 is mapped to logical system board (LSB 1) of physical partition 0.

```
XSCF> setpctl -p 0 -a 1=01-0
```

5. **Execute the `showpctl` command to check the PPAR configuration information that has been set.**


```
XSCF> showpcl -p 0
PPAR-ID LSB PSB Status
00          Powered Off
          00 00-0
          01 01-0
```

6. **Execute the showcodusage command to display the CPU core resource information.**

The following example displays the CPU core resource information.

As shown here, the system has 128 mounted CPU core resources and 128 registered CPU Activations, none of the CPU core resources are in use, and the number of CPU Activations currently not in use is 128.

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-----
PROC          0          128          128 OK: 128 cores
available

Note:
  Please confirm the value of the "In Use" by the ldm command of
  Oracle VM Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of
  logical domains.
```

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and then add the CPU Activation keys. For details on how to add a CPU Activation key, see "Chapter 5 CPU Activation" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

7. **Execute the setcod command to assign the CPU core resources to the physical partitions.**

The following example assigns 128 CPU core resources to physical partition 0.

```
XSCF> setcod -p 0 -s cpu -c set 128
PROC Permits assigned for PPAR 0 : 0 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

```
XSCF> setcod -s cpu
```

8. **Execute the showboards command to check the PSB status.**

The following example shows that PSB 01-0 is in the system board pool state.

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	n	n	n	Passed	Normal
01-0	SP	Available	n	n	n	Passed	Normal

9. **Execute the addboard command with the -c assign option specified to add the PSB.**

```
XSCF> addboard -c assign -p 0 01-0
```

10. **Execute the showresult command to check the exit status of the addboard command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the addboard command.

```
XSCF> showresult
0
```

11. **Execute the showboards command to check the PSB status.**

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	n	n	n	Passed	Normal
01-0	00(01)	Assigned	n	n	n	Passed	Normal

12. **Power on the system to start the physical partition.**

13. **Check the physical partition operation status by using the showpparstatus command.**

The next example shows "Running" in the [PPAR Status] column, indicating that the physical partition is operating correctly.

```
XSCF> showpparstatus -p 0
```

PPAR-ID	PPAR Status
00	Running

14. **Use the showboards command to check the PSB status.**

The next example shows "y" in the [Conn] and [Conf] columns for PSB 01-0, indicating that the PSB has been added correctly.

XSCF> showboards -p 0							
PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	00(01)	Assigned	y	y	y	Passed	Normal

15. **Assign the physical partition resources in the logical domain to reconfigure the logical domain.**

6.2.2 Example of System Board Incorporation

If the physical partition to which a system board (PSB) is added is in the operation state, use the addboard command with the -c configure option specified to incorporate the PSB into the physical partition.

When the addboard command is executed, the PSB registered in the physical partition configuration information (PPAR configuration information) or the PSB assigned to the physical partition by the -c assign option is incorporated into the physical partition. At this time, the physical partition needs to be powered on, and the control domain needs to be operating.

When you incorporate a PSB into a physical partition, you can specify whether to enable the automatic assignment function of the added resources of the PSB to be incorporated.

- **Without the -m option specified or with -m bind=resource specified**
If resources have been deleted with the deleteboard command before execution of the addboard command, the resource assignment that existed before executing the deleteboard command can be recovered. However, if the logical domain configuration is changed before executing the addboard command, the resulting resource assignment is in accordance with the logical domain configuration information. Suppose that the logical domain configuration information of the physical partition is factory-default. Then the resources added with the PSB will be assigned to the control domain.
- **With -m bind=none specified**
The resources added with the PSB will be free resources of the physical partition.

Note - The command with the -c configure option specified cannot incorporate a PSB into a physical partition where the PPAR DR function is disabled.

Note - In PSB incorporation using the addboard command, the target PSB is incorporated into the physical partition after the hardware diagnosis of the PSB is completed. For this reason, it may take some time before the command finishes.

The operation example of the SPARC M10-4S presented here uses XSCF shell commands and the ldm command of Logical Domains Manager, which is management software of Oracle VM Server for SPARC. The same operations are also used for the SPARC M12-2S.

1. **Log in to the XSCF.**
2. **When you add a PSB for system expansion, use the addfru command to incorporate SPARC M10-4S into the system.**

Note - For details of system expansion, see "Chapter 9 Expanding a System with a Building Block Configuration" in the *Installation Guide* of your server.

3. **Execute the showpcl command to check the current physical partition configuration information (PPAR configuration information).**

```
XSCF> showpcl -p 0
PPAR-ID   LSB   PSB   Status
00                Running
          00   00-0
```

4. **Execute the setpcl command to register the PSB in the PPAR configuration information.**

In the following example, PSB 01-0 is mapped to logical system board (LSB 1) of physical partition 0.

```
XSCF> setpcl -p 0 -a 1=01-0
```

5. **Execute the showpcl command to check the PPAR configuration information that has been set.**

```
XSCF> showpcl -p 0
PPAR-ID   LSB   PSB   Status
00                Running
          00   00-0
          01   01-0
```

6. **Connect to the control domain console of the physical partition.**
7. **Execute the ldm list-domain command to check the operation status of the logical domain.**

The following example shows the control domain and three guest domains in operation.

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART     14    8G      64%   2h 54m
guest0        active    -n----   5000     16    8G      42%   2h 54m
guest1        active    -n----   5001     16    8G      11%   2h 54m
guest2        active    -n----   5002     16    8G      7.3%   2h 54m
```

8. **Return to the XSCF shell, and execute the showcodusage command to display information on the CPU core resources.**

The following example displays the CPU core resource information.

As shown here, the system has 128 mounted CPU core resources and 128 registered CPU Activations, 64 of the CPU core resources are in use, and the number of CPU Activations currently not in use is 64.

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-----
PROC          64          128          128 OK: 64 cores available

Note:
  Please confirm the value of the "In Use" by the ldm command of
  Oracle VM Server for SPARC.

  The XSCF may take up to 20 minutes to reflect the "In Use" of
  logical domains.
```

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and add the CPU Activation keys. For details on how to add a CPU Activation key, see "Chapter 5 CPU Activation" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

9. **Execute the setcod command to assign the CPU core resources to the physical partitions.**

The following example adds 64 CPU core resources to physical partition 0.

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 :64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

```
XSCF> setcod -s cpu
```

10. **Execute the showboards command to check the PSB status.**

The following example shows that PSB 01-0 is in the system board pool state.

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf  Test  Fault
-----
00-0 00(00)      Assigned  y    y    y    Passed Normal
01-0 SP          Available n    n    n    Passed Normal
```

11. **Execute the addboard command with the -c configure option specified to incorporate the PSB into the physical partition.**

In the following example, PSB 01-0 is incorporated into physical partition 0.

```
XSCF> addboard -c configure -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
 0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager. [1800sec]
0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

12. **Execute the showresult command to check the exit status of the addboard command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the addboard command.

```
XSCF> showresult
0
```

13. **Execute the showboards command to check the PSB status.**

The next example shows "y" in the [Conn] and [Conf] columns for PSB 01-0, indicating that the PSB has been added correctly.

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	00(01)	Assigned	y	y	y	Passed	Normal

14. **Execute the ldm list-domain command on the control domain console of the physical partition to confirm that the operation status of the logical domain has not changed after the addition of the PSB.**
15. **Assign the physical partition resources in the logical domain to reconfigure the logical domain.**

6.2.3 Example of Operations for Reserving the Incorporation of a System Board

If you cannot make operations related to the operation status or configuration of a logical domain, the operation status or configuration of a physical partition, or the

dynamic reconfiguration of a physical partition, reserve the incorporation of the system board (PSB) into the physical partition. Restarting the physical partition at an appropriate time during system operation will incorporate the PSB into the physical partition.

The operation example of the SPARC M10-4S presented here uses XSCF shell commands. The same operations are also used for the SPARC M12-2S.

1. **Log in to the XSCF.**
2. **When you add a PSB for system expansion, use the `addfru` command to incorporate SPARC M10-4S into the system.**

Note - For details of system expansion, see "Chapter 8 Expanding a System With a Building Block Configuration" in the *SPARC M12-2S Installation Guide* or "Chapter 9 Expanding a System with a Building Block Configuration" in the *SPARC M10-4S Installation Guide*.

3. **Execute the `showpcl` command to check the current physical partition configuration information (PPAR configuration information).**

```
XSCF> showpcl -p 0
PPAR-ID  LSB   PSB   Status
00                Running
           00    00-0
```

4. **Execute the `setpcl` command to register the PSB in the PPAR configuration information.**

In the following example, PSB 01-0 is mapped to logical system board (LSB 1) of physical partition 0.

```
XSCF> setpcl -p 0 -a 1=01-0
```

5. **Execute the `showpcl` command to check the PPAR configuration information that has been set.**

```
XSCF> showpcl -p 0
PPAR-ID LSB PSB Status
00                Running
           00    00-0
           01    01-0
```

6. **Execute the `showcodusage` command to display the CPU core resource information.**

The following example displays the CPU core resource information.

As shown here, the system has 128 mounted CPU core resources and 128 registered CPU Activations, 64 of the CPU core resources are in use, and the number of CPU Activations currently not in use is 64.

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-----
PROC          64          128          128 OK: 64 cores available
Note:
  Please confirm the value of the "In Use" by the ldm command of
  Oracle VM Server for SPARC.
  The XSCF may take up to 20 minutes to reflect the "In Use" of
  logical domains.
```

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and add the CPU Activation keys. For details on how to add a CPU Activation key, see "Chapter 5 CPU Activation" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

7. Execute the setcod command to assign the CPU core resources to the physical partitions.

The following example adds 64 CPU core resources to physical partition 0.

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 :64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

```
XSCF> setcod -s cpu
```

8. Execute the showboards command to check the PSB status.

The following example shows that PSB 01-0 is in the system board pool state.

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test  Fault
-----
00-0 00(00)      Assigned   y    y    y    Passed Normal
01-0 SP          Available  n    n    n    Passed Normal
```

9. Execute the addboard command with the -c reserve option specified to reserve the addition of the PSB.

```
XSCF> addboard -c reserve -p 0 01-0
```


10. **Execute the showresult command to check the exit status of the addboard command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the addboard command.

```
XSCF> showresult
0
```

11. **Execute the showboards command with the -v option specified to check the PSB status and confirm that the addition of the PSB is reserved.**

Because the following example shows "*" in the [R] column, which represents the reserved state, the PSB is correctly reserved for addition.

```
XSCF> showboards -v 01-0
PSB  R  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
01-0  *  00(01)        Assigned    n    n    n    Passed  Normal
```

12. **Restart the physical partition.**
13. **Check the physical partition operation status by using the showpparstatus command.**

The next example shows "Running" in the [PPAR Status] column, indicating that the physical partition is operating correctly.

```
XSCF> showpparstatus -p 0
PPAR-ID  PPAR Status
00        Running
```

14. **Use the showboards command to check the PSB status.**

The next example shows "y" in the [Conn] and [Conf] columns for PSB 01-0, indicating that the PSB has been added correctly.

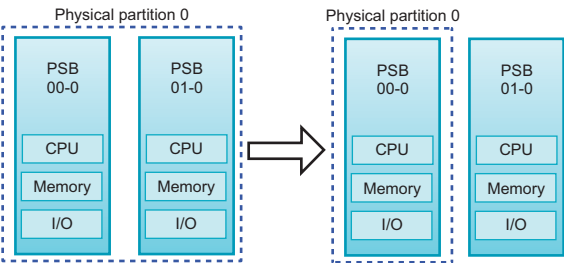
```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)        Assigned    y    y    y    Passed  Normal
01-0  00(01)        Assigned    y    y    y    Passed  Normal
```

15. **Assign the physical partition resources in the logical domain to reconfigure the logical domain.**

6.3 Example of Operations for Deleting a System Board

This section describes an example of the operations for deleting a system board (PSB) as shown in [Figure 6-8](#).

Figure 6-8 Example of Operations for Deleting a System Board



6.3.1 Example of Operations for Deleting a System Board Assignment

If the physical partition from which a system board (PSB) is deleted is in the stopped state, use the deleteboard command with the -c unassign option specified to unassign the PSB from the physical partition. The PSB is unassigned from the physical partition, and enters the system board pool state.

The operation example presented here uses XSCF shell commands.

- 1. **Log in to the XSCF.**
- 2. **Execute the showpparstatus command to confirm that the physical partition is powered off.**

```
XSCF> showpparstatus -p 0
PPAR-ID PPAR Status
00      Powered Off
```

- 3. **Execute the showboards command to check the status of the PSB to be deleted.**

```
XSCF> showboards 01-0
PSB  PPAR-ID(LSB)  Assignment Pwr  Conn Conf Test  Fault
-----
01-0 00(01)      Assigned  n    n    n    Passed Normal
```

4. **Execute the deleteboard command with the -c unassign option specified to delete the PSB.**

```
XSCF> deleteboard -c unassign 01-0
```

5. **Execute the showresult command to check the exit status of the deleteboard command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the deleteboard command.

```
XSCF> showresult
0
```

6. **Execute the showboards command to check the PSB status.**

The following example shows that PSB 01-0 is in the system board pool state.

```
XSCF> showboards 01-0
```

PSB	PPAR-ID (LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
----	-----	-----	----	-----	-----	-----	-----
01-0	SP	Available	n	n	n	Passed	Normal

6.3.2 Example of Operations for Releasing a System Board

If the physical partition from which the system board (PSB) is released is in the operation state, use the deleteboard command with the -c disconnect or -c unassign option specified to release the PSB from the physical partition.

If the -c disconnect option is specified, the PSB is released from the physical partition but remains assigned to the physical partition. Therefore, restarting the physical partition or executing the addboard command incorporates the released PSB into the physical partition again.

If you release a PSB from a physical partition by specifying the -c unassign option, the PSB enters the system pool state and can be incorporated and assigned to another physical partition. For a PSB that has been released by specifying the -c disconnect option, executing the deleteboard command with the -c unassign option specified changes the state of the PSB from the assigned state to the system board pool state.

When releasing a PSB from a physical partition, use free resources in the physical partition to move the logical domains that use the resources of the PSB to be released. You can specify how to obtain the resources when the logical domain is moved according to the free resource status. To do this, specify the -m option for the deleteboard command.

- If the -m option is not specified or -m unbind=none is specified

If free resources cannot be obtained, or if the physical I/O assignment is not released in advance, the release with the deleteboard command fails.

- If -m unbind=resource is specified

With Oracle VM Server for SPARC 3.4 or later, you can select a policy for unbinding resources through the PPAR DR policy set in the `ldmd/fj_ppar_dr_policy` property of the `ldmd` service. With Oracle VM Server for SPARC earlier than Oracle VM Server for SPARC 3.4, this policy is always "targeted".

For details on the policy, see "8.15 Setting the Physical Partition Dynamic Reconfiguration Policy" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

`ldmd/fj_ppar_dr_policy = auto`

This PPAR DR policy setting specifies to automatically use the latest policy. For a system that uses Oracle VM Server for SPARC 3.4, the "auto" setting works in the same way as the "ratio" setting. The "auto" setting is the default policy.

`ldmd/fj_ppar_dr_policy = ratio`

This policy is functional if all the logical domains in the physical partition run Oracle Solaris 11.3 or later and XCP 2271 or later is applied to the system. Otherwise, the system operates in the same way as when "targeted" is set.

If the policy is set to "ratio" and the remaining PSBs do not have enough free resources to move the resources from the PSB to be released, the resources are automatically released by reducing the resources from the existing domains. In this case, resources are automatically reduced from all the existing domains proportionally as much as possible. Because resources are busy or due to other reasons, exact proportional reduction does not always work. The proportional reduction is on a best effort basis.

`ldmd/fj_ppar_dr_policy = targeted`

If free resources cannot be obtained, resources for the move are obtained from the resources deleted from each logical domain that uses CPU/memory on the destination PSB. A policy defines the order of selecting the logical domain from which to delete resources. This policy defines the following order: the default logical domain (with nothing specified), logical domain with the master specified, I/O domains, root domain, and control domain.

If free resources cannot be obtained from the destination PSB, release with the deleteboard command fails.

- If -m unbind=shutdown is specified

The operation is the same as when `-m unbind=resource` is specified. If no more available resources can be obtained, the logical domain at the source shuts down. To use this option, `ldmd/fj_ppar_dr_policy` must be set to "targeted". Otherwise, the logical domain may not shut down.

Note - For a physical partition with the PPAR DR function disabled, you cannot release a PSB by specifying the `-c disconnect` or `-c unassign` option when the physical partition is in operation.

Note - When a PSB is released with the deleteboard command, the hardware resources on the

PSB are released from Oracle Solaris. Therefore, it may take some time before the command finishes.

The operation example presented here uses XSCF shell commands and the `ldm` command of Logical Domains Manager, which is management software of Oracle VM Server for SPARC.

1. **Log in to the XSCF.**
2. **Connect to the control domain console of the physical partition.**
3. **Execute the `ldm list-domain` command to check the operation status of the logical domain.**

The following example shows that the control domain, two guest domains, and one root domain are operating.

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART    14    8G      64%   2h 54m
guest0        active    -n----   5000    16    8G      42%   2h 54m
guest1        active    -n----   5001    16    8G      11%   2h 54m
domain01      active    -n----   5002    16    8G      7.3%  2h 54m
```

4. **Execute the `ldm list-devices` command with the `-a` option specified to check the resource usage status.**

In the following example, the `-a` option is specified to display all resources bound to the logical domain and all resources that are not bound.

```
# ldm list-devices -a
CORE
  ID      %FREE  CPUSSET
  0        0      (0, 1)
  4        0      (8, 9)
  8        0     (16, 17)
(Omitted)
 184       0     (368, 369)
 188      100     (376, 377)
 512      100    (1024, 1025)
 516      100    (1032, 1033)
 520      100    (1040, 1041)
 524      100    (1048, 1049)
(Omitted)
VCPU
  PID      %FREE  PM
  0        0      no
  1        0      no
  8        0      no
  9        0      no
(Omitted)
 369       0      no
 376      100     ---
 377      100     ---
1024      100     ---
1025      100     ---
```

1032	100	---
1033	100	---
(Omitted)		

5. **Execute the ldm list-io command to check the usage status of the I/O devices.**

In the following example, primary, which is a control domain, and domain01, which is a root domain, use I/O devices.

```
# ldm list-io
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
PCIE0                              BUS     PCIE0    primary   IOV
PCIE1                              BUS     PCIE1    primary   IOV
PCIE2                              BUS     PCIE2    primary   IOV
PCIE3                              BUS     PCIE3    primary   IOV
PCIE8                              BUS     PCIE8    domain01  IOV
PCIE9                              BUS     PCIE9    domain01  IOV
PCIE10                             BUS     PCIE10   domain01  IOV
PCIE11                             BUS     PCIE11   domain01  IOV
/BB0/CMUL/NET0                     PCIE    PCIE0    primary   OCC
/BB0/CMUL/NET2                     PCIE    PCIE0    primary   OCC
/BB0/CMUL/SASHBA                   PCIE    PCIE0    primary   OCC
/BB0/PCI0                           PCIE    PCIE1    primary   EMP
/BB0/PCI1                           PCIE    PCIE1    primary   EMP
(Omitted)
```

6. **Among the services provided by the root domain, release all I/O devices on the PSB to be deleted.**

In the following example, domain01, which is a root domain, is stopped, and of the PCIe root complexes owned by domain01, those related to the PSB to be deleted are released.

Note - To reconfigure dynamically, the ldm stop-domain command to stop the root domain is not required. Start operation from the I/O device release.

```
# ldm stop-domain domain01
# ldm remove-io PCIE11 domain01
# ldm remove-io PCIE10 domain01
# ldm remove-io PCIE9 domain01
# ldm remove-io PCIE8 domain01
# ldm unbind-domain domain01
```

7. **Execute the ldm list-io command again to check the usage status of the I/O devices.**

```
# ldm list-io
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
PCIE0                              BUS     PCIE0    primary   IOV
```

PCIE1	BUS	PCIE1	primary	IOV
PCIE2	BUS	PCIE2	primary	IOV
PCIE3	BUS	PCIE3	primary	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9		
PCIE10	BUS	PCIE10		
PCIE11	BUS	PCIE11		
/BB0/CMUL/NET0	PCIE	PCIE0	primary	OCC
/BB0/CMUL/NET2	PCIE	PCIE0	primary	OCC
/BB0/CMUL/SASHBA	PCIE	PCIE0	primary	OCC
/BB0/PCIO	PCIE	PCIE1	primary	EMP
/BB0/PCI1	PCIE	PCIE1	primary	EMP

(Omitted)

8. **Execute the deleteboard command with the -c unassign option specified to release the PSB from the physical partition.**

In the following example, PSB 01-0 is released from the physical partition, and is placed in the system board pool state.

```
XSCF> deleteboard -c unassign 01-0
PSB#01-0 will be unassigned from PPAR immediately. Continue?[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
  0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [43200sec]
  0..... 30.end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
  0..... 30..... 60..... 90.....120.....150.end
Operation has completed.
```

9. **Execute the showresult command to check the exit status of the deleteboard command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the deleteboard command.

```
XSCF> showresult
0
```

10. **Execute the showboards command to check the PSB status.**

The following example shows that PSB 01-0 is in the system board pool state.

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)         Assigned    y    y    y    Passed  Normal
01-0  SP             Available   n    n    n    Passed  Normal
```

11. **Execute the ldm list-domain command on the control domain console of the physical partition to confirm that the operation status of the logical domain**

has not changed after the deletion of the PSB.

12. **When you remove a building block, execute the `initbb` command on the master XSCF to release the target SPARC M12-2S/M10-4S from the system.**

Note - For details of system reduction, see "Chapter 9 Reducing a System With a Building Block Configuration" in the *SPARC M12-2S Installation Guide* or "Chapter 10 Removing a System with a Building Block Configuration" in the *SPARC M10-4S Installation Guide*.

6.3.3 Example of Operations for Reserving a System Board Unassignment

If you cannot make operations related to the operation status or configuration of a logical domain, the operation status or configuration of a physical partition, or the dynamic reconfiguration of a physical partition, reserve the unassignment of the system board (PSB) from the physical partition. By stopping a physical partition at an appropriate time during system operation, the PSB is unassigned from the physical partition. The PSB enters the system board pool state.

The operation example presented here uses XSCF shell commands.

1. **Execute the `showpparstatus` command to check the physical partition operation status.**

```
XSCF> showpparstatus -p 0
PPAR-ID PPAR Status
00      Running
```

2. **Execute the `showboards` command to check the status of the PSB to be deleted.**

```
XSCF> showboards 01-0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test      Fault
-----
01-0 00(01)      Assigned    y    y    y    Passed Normal
```

3. **Execute the `deleteboard` command with the `-c reserve` option specified to reserve the deletion of the PSB.**

```
XSCF> deleteboard -c reserve 01-0
```

4. **Execute the `showresult` command to check the exit status of the `deleteboard` command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the `deleteboard` command.


```
XSCF> showresult
0
```

5. **Execute the showboards command with the -v option specified to check the PSB status and confirm that the deletion of the PSB is reserved.**
Because the following example shows "*" in the [R] column, which represents the reserved state, the deletion of the PSB is correctly reserved.

```
XSCF> showboards -v 01-0
PSB  R  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
01-0  *  00(01)         Assigned    y    y    y    Passed Normal
```

6. **Restart the physical partition.**
7. **Check the physical partition operation status by using the showpparstatus command.**
The next example shows "Running" in the [PPAR Status] column, indicating that the physical partition is operating correctly.

```
XSCF> showpparstatus -p 0
PPAR-ID  PPAR Status
00        Running
```

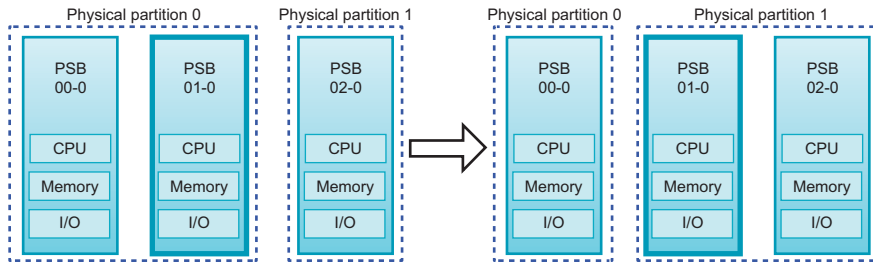
8. **Execute the showboards command to check the PSB status.**
The following example shows that PSB 01-0 is in the system board pool state.

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)         Assigned    y    y    y    Passed Normal
01-0  SP            Available    n    n    n    Passed Normal
```

6.4 Example of Operations for Moving a System Board

This section describes an example of the operations for moving a system board (PSB) between physical partitions as shown in [Figure 6-9](#).

Figure 6-9 Example of Operations for Moving a System Board



The operation for moving a PSB between physical partitions comprises the combined operations of PSB unassignment and incorporation.

If the dynamic reconfiguration operation is not possible due to the operation status of the source physical partition, delete the PSB from the source physical partition by unassigning the PSB or reserving the unassignment of the PSB. When the dynamic reconfiguration operation is not possible at the destination physical partition, add the PSB by assigning the PSB or reserving the assignment of the PSB to the destination physical partition.

Note - Suppose that the PPAR DR function of a physical partition is disabled, and the physical partition is operating. In such a case, you cannot release a PSB by specifying the `-c disconnect` or `-c unassign` option, or incorporate a PSB by specifying the `-c configure` option.

The operation example here assumes that the source and destination physical partitions allow the dynamic reconfiguration operation. The operation example of the SPARC M10-4S presented here uses XSCF shell commands and the `ldm` command of Logical Domains Manager, which is management software of Oracle VM Server for SPARC. The same operations are also used for the SPARC M12-2S.

1. **Log in to the XSCF.**
2. **Connect to the control domain console of the physical partition.**
3. **Execute the `ldm list-domain` command to check the operation status of the logical domain.**

The following example shows that the control domain, two guest domains, and one root domain are operating.

# <code>ldm list-domain</code>							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	14	8G	64%	2h 54m
guest0	active	-n----	5000	16	8G	42%	2h 54m
guest1	active	-n----	5001	16	8G	11%	2h 54m
domain01	active	-n----	5002	16	8G	7.3%	2h 54m

4. **Execute the `ldm list-devices` command with the `-a` option specified to check the resource usage status.**

In the following example, the `-a` option is specified to display all resources bound to the logical domain and all resources that are not bound.

```
# ldm list-devices -a
CORE
  ID      %FREE  CPUSSET
  0        0      (0, 1)
  4        0      (8, 9)
  8        0     (16, 17)
(Omitted)
 184       0     (368, 369)
 188      100    (376, 377)
 512      100   (1024, 1025)
 516      100   (1032, 1033)
 520      100   (1040, 1041)
 524      100   (1048, 1049)
(Omitted)
VCPU
  PID     %FREE  PM
  0        0     no
  1        0     no
  8        0     no
  9        0     no
(Omitted)
 369       0     no
 376      100    ---
 377      100    ---
 1024     100    ---
 1025     100    ---
 1032     100    ---
 1033     100    ---
(Omitted)
```

5. **Execute the ldm list-io command to check the usage status of the I/O devices.** In the following example, primary, which is a control domain, and domain01, which is a root domain, use I/O devices.

```
# ldm list-io
NAME                                TYPE  BUS      DOMAIN  STATUS
----                                -
PCIE0                              BUS   PCIE0    primary IOV
PCIE1                              BUS   PCIE1    primary IOV
PCIE2                              BUS   PCIE2    primary IOV
PCIE3                              BUS   PCIE3    primary IOV
PCIE8                              BUS   PCIE8    domain01 IOV
PCIE9                              BUS   PCIE9    domain01 IOV
PCIE10                             BUS   PCIE10   domain01 IOV
PCIE11                             BUS   PCIE11   domain01 IOV
/BB0/CMUL/NET0                     PCIE  PCIE0    primary OCC
/BB0/CMUL/NET2                     PCIE  PCIE0    primary OCC
/BB0/CMUL/SASHBA                   PCIE  PCIE0    primary OCC
/BB0/PCI0                          PCIE  PCIE1    primary EMP
/BB0/PCI1                          PCIE  PCIE1    primary EMP
(Omitted)
```

6. **Among the services provided by the root domain, release all I/O devices on**

the PSB to be deleted.

In the following example, domain01, which is a root domain, is stopped, and of the PCIe root complexes owned by domain01, those related to the PSB to be deleted are released.

Note - To reconfigure dynamically, the ldm stop-domain command to stop the root domain is not required. Start operation from the I/O device release.

```
# ldm stop-domain domain01
# ldm remove-io PCIE11 domain01
# ldm remove-io PCIE10 domain01
# ldm remove-io PCIE9 domain01
# ldm remove-io PCIE8 domain01
# ldm unbind-domain domain01
```

7. **Execute the ldm list-io command again to check the usage status of the I/O devices.**

```
# ldm list-io
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
PCIE0                               BUS     PCIE0    primary   IOV
PCIE1                               BUS     PCIE1    primary   IOV
PCIE2                               BUS     PCIE2    primary   IOV
PCIE3                               BUS     PCIE3    primary   IOV
PCIE8                               BUS     PCIE8
PCIE9                               BUS     PCIE9
PCIE10                              BUS     PCIE10
PCIE11                              BUS     PCIE11
/BB0/CMUL/NET0                      PCIE    PCIE0    primary   OCC
/BB0/CMUL/NET2                      PCIE    PCIE0    primary   OCC
/BB0/CMUL/SASHBA                    PCIE    PCIE0    primary   OCC
/BB0/PCI0                            PCIE    PCIE1    primary   EMP
/BB0/PCI1                            PCIE    PCIE1    primary   EMP
(Omitted)
```

8. **Execute the deleteboard command with the -c unassign option specified to release the PSB from the physical partition.**

In the following example, PSB 01-0 is released from the physical partition, and is placed in the system board pool state.

```
XSCF> deleteboard -c unassign 01-0
```

9. **Execute the showresult command to check the exit status of the deleteboard command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the deleteboard command.

```
XSCF> showresult
0
```

10. **Execute the showboards command to check the PSB status.**

The following example shows that PSB 01-0 is in the system board pool state.

```
XSCF> showboards -a
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0 00(00)       Assigned    y    y    y    Passed Normal
01-0 SP          Available   n    n    n    Passed Normal
02-0 01(00)       Assigned    y    y    y    Passed Normal
```

11. **Execute the ldm list-domain command on the control domain console of the source physical partition to confirm that the operation status of the logical domain has not changed after the deletion of the PSB.**
12. **Execute the showpcl command to check the current physical partition configuration information (PPAR configuration information) of the destination physical partition.**

```
XSCF> showpcl -p 1
PPAR-ID  LSB  PSB  Status
01
          00  00-0  Running
```

13. **Execute the setpcl command to register the PSB in the PPAR configuration information of the destination physical partition.**

In the following example, PSB 01-0 is mapped to logical system board (LSB 1) of physical partition 1.

```
XSCF> setpcl -p 1 -a 1=01-0
```

14. **Execute the showpcl command to check the PPAR configuration information that has been set.**

```
XSCF> showpcl -p 1
PPAR-ID  LSB  PSB  Status
01
          00  02-0  Running
          01  01-0
```

15. **Connect to the control domain console of the physical partition.**
16. **Execute the ldm list-domain command to check the operation status of the logical domain.**

The following example shows the control domain and three guest domains in operation.

# ldm list-domain							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	14	8G	64%	2h 54m
guest0	active	-n----	5000	16	8G	42%	2h 54m
guest1	active	-n----	5001	16	8G	11%	2h 54m
guest2	active	-n----	5002	16	8G	7.3%	2h 54m

17. Return to the XSCF shell, and execute the showcod command to check the CPU core resources assigned to the physical partition.

The following example displays the setting information for the CPU core resources.

As shown here, the system has 192 mounted CPU core resources, 128 CPU Activations registered in physical partition 0, and 64 CPU Activations registered in physical partition 1.

```
XSCF> showcod -v -s cpu
PROC Permits installed : 192 cores
PROC Permits assigned for PPAR 0: 128 [Permanent 128cores]
PROC Permits assigned for PPAR 1: 64 [Permanent 64cores]
PROC Permits assigned for PPAR 2: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 3: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 4: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 5: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 6: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 7: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 8: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 9: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 10: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 11: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 12: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 13: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 14: 0 [Permanent 0cores]
PROC Permits assigned for PPAR 15: 0 [Permanent 0cores]
```

18. Execute the showcodusage command to check the CPU core resource information.

The result shows that the mounting status is as follows.

[Physical partition 0]

- Number of CPU core resources: 64
- Number of registered CPU Activations: 128
- Number of CPU core resources in use: 64

[Physical partition 1]

- Number of CPU core resources: 64
- Number of registered CPU Activations: 64
- Number of CPU core resources in use: 64

[Entire system]

- Number of CPU Activations currently not in use (Unused): 64

```
XSCF> showcodusage -p ppar
```

PPAR-ID/Resource	In Use	Installed	Assigned
0 - PROC	64	64	128 cores
1 - PROC	64	64	64 cores
Unused - PROC	0	0	64 cores

Note:

Please confirm the value of the "In Use" by the ldm command of Oracle VM Server for SPARC.

The XSCF may take up to 20 minutes to reflect the "In Use" of logical domains.

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and add the CPU Activation keys. For details on how to add a CPU Activation key, see "Chapter 5 CPU Activation" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

19. **Execute the setcod command to assign the CPU core resources to the destination physical partitions.**

The following example reduces the number of assigned CPU Activations in physical partition 0 by removing 64 of them and adding them to physical partition 1.

```
XSCF> setcod -p 0 -s cpu -c del 64
PROC Permits assigned for PPAR 0 : 128 -> 64

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.

XSCF> setcod -p 1 -s cpu -c add 64
PROC Permits assigned for PPAR 1 : 64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

```
XSCF> setcod -s cpu
```

20. **Execute the showboards command to check the PSB status.**

```
XSCF> showboards -a
```

PSB	PPAR-ID (LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	SP	Available	n	n	n	Passed	Normal
02-0	01(00)	Assigned	y	y	y	Passed	Normal

21. **Execute the addboard command with the -c configure option specified to incorporate the PSB into the physical partition.**

In the following example, PSB 01-0 is incorporated into physical partition 1.

```
XSCF> addboard -c configure -p 1 01-0
```

22. **Execute the showresult command to check the exit status of the addboard command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the addboard command.

```
XSCF> showresult
0
```

23. **Execute the showboards command to check the PSB status.**

The next example shows "y" in the [Conn] and [Conf] columns for PSB 01-0, indicating that the PSB has been added correctly.

```
XSCF> showboards -a
```

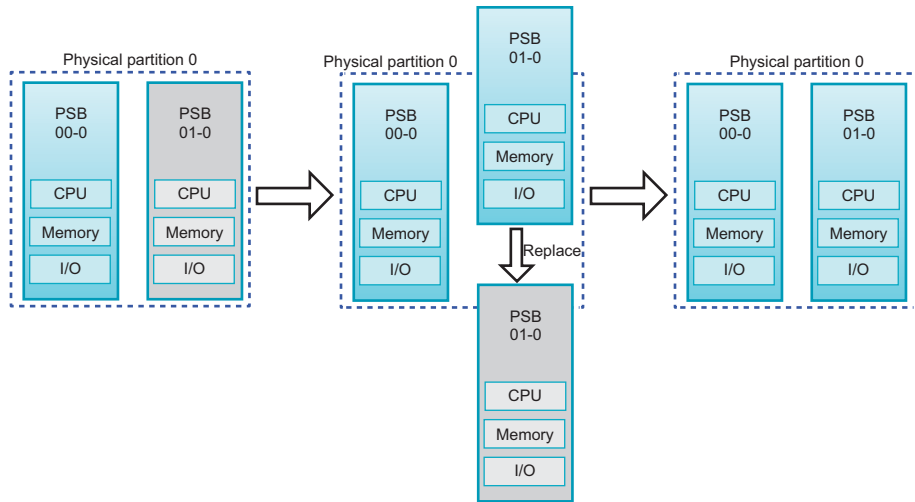
PSB	PPAR-ID (LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	01(01)	Assigned	y	y	y	Passed	Normal
02-0	01(00)	Assigned	y	y	y	Passed	Normal

24. **Execute the ldm list-domain command on the control domain console of the destination physical partition to confirm that the operation status of the logical domain has not changed after the addition of the PSB.**
25. **Assign the physical partition resources in the logical domain to reconfigure the logical domain.**

6.5 Example of Operations for Replacing a System Board

This section describes an example of the operations for replacing a system board (PSB) as shown in [Figure 6-10](#).

Figure 6-10 Example of Operations for Replacing a System Board



The operation for replacing a PSB comprises the combined operations of PSB release and incorporation. When the dynamic reconfiguration operation is not possible due to the operation status of the physical partition, replace the PSB after powering off the physical partition.

Note - Suppose that the PPAR DR function of a physical partition is disabled, and the physical partition is operating. In such a case, you cannot release a PSB by specifying the `-c disconnect` or `-c unassign` option, or incorporate a PSB by specifying the `-c configure` option.

The operation example here assumes that the physical partition allows dynamic reconfiguration. The operation example of the SPARC M10-4S presented here uses XSCF shell commands and the `ldm` command of Logical Domains Manager, which is management software of Oracle VM Server for SPARC. The same operations are also used for the SPARC M12-2S.

1. **Log in to the XSCF.**
2. **Execute the `showbbstatus` command to confirm that the XSCF in the PSB to be replaced is not the master XSCF.**

Note - If the XSCF of the PSB to be replaced is the master XSCF, execute the `switchscf` command to switch the XSCF from the active state to the standby state. Confirm that the XSCF has been switched and rebooted before you release the PSB.

3. **Connect to the control domain console of the physical partition.**
4. **Execute the `ldm list-domain` command to check the operation status of the logical domain.**

The following example shows that the control domain, two guest domains, and one root domain are operating.

# ldm list-domain							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	14	8G	64%	2h 54m
guest0	active	-n----	5000	16	8G	42%	2h 54m
guest1	active	-n----	5001	16	8G	11%	2h 54m
domain01	active	-n----	5002	16	8G	7.3%	2h 54m

5. **Execute the ldm list-devices command with the -a option specified to check the resource usage status.**

In the following example, the -a option is specified to display all resources bound to the logical domain and all resources that are not bound.

# ldm list-devices -a		
CORE		
ID	%FREE	CPUSET
0	0	(0, 1)
4	0	(8, 9)
8	0	(16, 17)
(Omitted)		
184	0	(368, 369)
188	100	(376, 377)
512	100	(1024, 1025)
516	100	(1032, 1033)
520	100	(1040, 1041)
524	100	(1048, 1049)
(Omitted)		
VCPU		
PID	%FREE	PM
0	0	no
1	0	no
8	0	no
9	0	no
(Omitted)		
369	0	no
376	100	---
377	100	---
1024	100	---
1025	100	---
1032	100	---
1033	100	---
(Omitted)		

6. **Execute the ldm list-io command to check the usage status of the I/O devices.**

In the following example, primary, which is a control domain, and domain01, which is a root domain, use I/O devices.

# ldm list-io				
NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	primary	IOV
PCIE2	BUS	PCIE2	primary	IOV

PCIE3	BUS	PCIE3	primary	IOV
PCIE8	BUS	PCIE8	domain01	IOV
PCIE9	BUS	PCIE9	domain01	IOV
PCIE10	BUS	PCIE10	domain01	IOV
PCIE11	BUS	PCIE11	domain01	IOV
/BB0/CMUL/NET0	PCIE	PCIE0	primary	OCC
/BB0/CMUL/NET2	PCIE	PCIE0	primary	OCC
/BB0/CMUL/SASHBA	PCIE	PCIE0	primary	OCC
/BB0/PCIO	PCIE	PCIE1	primary	EMP
/BB0/PCI1	PCIE	PCIE1	primary	EMP
(Omitted)				

7. **Among the services provided by the root domain, release all I/O devices on the PSB to be deleted.**

In the following example, domain01, which is a root domain, is stopped, and of the PCIe root complexes owned by domain01, those related to the PSB to be deleted are released.

Note - To reconfigure dynamically, the `ldm stop-domain` command to stop the root domain is not required. Start operation from the I/O device release.

```
# ldm stop-domain domain01
# ldm remove-io PCIE11 domain01
# ldm remove-io PCIE10 domain01
# ldm remove-io PCIE9 domain01
# ldm remove-io PCIE8 domain01
# ldm unbind-domain domain01
```

8. **Execute the `ldm list-io` command again to check the usage status of the I/O devices.**

```
# ldm list-io
NAME                TYPE    BUS      DOMAIN    STATUS
----                -
PCIE0                BUS     PCIE0    primary   IOV
PCIE1                BUS     PCIE1    primary   IOV
PCIE2                BUS     PCIE2    primary   IOV
PCIE3                BUS     PCIE3    primary   IOV
PCIE8                BUS     PCIE8
PCIE9                BUS     PCIE9
PCIE10               BUS     PCIE10
PCIE11               BUS     PCIE11
/BB0/CMUL/NET0       PCIE    PCIE0    primary   OCC
/BB0/CMUL/NET2       PCIE    PCIE0    primary   OCC
/BB0/CMUL/SASHBA     PCIE    PCIE0    primary   OCC
/BB0/PCIO             PCIE    PCIE1    primary   EMP
/BB0/PCI1             PCIE    PCIE1    primary   EMP
(Omitted)
```

9. **Execute the `deleteboard` command with the `-c disconnect` option specified to**

release the PSB from the physical partition.

In the following example, PSB 01-0 is released from the physical partition and assigned to physical partition 0.

```
XSCF> deleteboard -c disconnect 01-0
```

10. **Execute the showresult command to check the exit status of the deleteboard command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the deleteboard command.

```
XSCF> showresult
0
```

11. **Execute the showboards command to check the PSB status.**

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	00(01)	Assigned	n	n	n	Passed	Normal

12. **Execute the replacefru command to replace the PSB.**

```
XSCF> replacefru
```

Note - For details on PSB replacement using the replacefru command, see the *Service Manual* for your server.

13. **Execute the showboards command to check the PSB status.**

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	00(01)	Assigned	n	n	n	Passed	Normal

14. **Execute the addboard command with the -c configure option specified to incorporate the PSB into the physical partition.**

In the following example, PSB 01-0 is incorporated into physical partition 0.

```
XSCF> addboard -c configure -p 0 01-0
```

Note - If you omit the -m option or specify bind=resource for the -m option when you execute the addboard command, the resource assignment may not be returned to its state before the deleteboard command was executed.

If the amount of the resources of the PSB such as CPU and memory before replacement is different from that after replacement, you cannot return the resource assignment to its original state.

Resources that cannot not be returned to their original assignment become free. In such cases, use the ldm command to assign resources to the logical domain again.

15. **Execute the showresult command to check the exit status of the addboard command just executed.**

In the following example, exit status of 0 is returned, indicating normal termination of the addboard command.

```
XSCF> showresult
0
```

16. **Execute the showboards command to check the PSB status.**

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
----	-----	-----	----	-----	-----	-----	-----
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	00(01)	Assigned	y	y	y	Passed	Normal

17. **Execute the ldm list-domain command on the control domain console of the physical partition to confirm that the operation status of the logical domain has not changed after the addition of the PSB.**

18. **Start the root domain to resume its service.**

6.6 Considerations When Changing a Physical Partition Number

You can change the physical partition number of a configured physical partition to a different number. Consider the following point when changing the number:

- The physical partition configuration information must be re-set.
If you have changed a physical partition number, you need to set the configuration information of the new physical partition. For details on setting physical partition configuration information, see ["3.1.12 Setting Physical Partition Configuration Information"](#) and ["Chapter 4 Configuring a Physical Partition."](#)
- The CPU Activation must be reassigned.
After a physical partition number is changed, the CPU Activation is still assigned

to the original physical partition number. Therefore, you need to release the CPU Activation assignment and then reassign it by specifying the new physical partition number.

- The host ID and Ethernet (MAC) address are changed.

When the physical partition number is changed, the host ID and Ethernet (MAC) address specific to the physical partition are also changed. Consideration must also be given to the fact that the host ID and Ethernet (MAC) address may be used for license authentication and setting information for the Oracle Solaris ZFS file system and applications.

Migrating a Guest Domain

This chapter describes how to migrate a guest domain to a different physical partition by using the live migration function.

- [Overview](#)
- [Migrating a Guest Domain](#)
- [Guest Domain Migration Example](#)

7.1 Overview

The live migration function migrates a guest domain from one physical partition to another while the guest domain continues operating. The live migration function is used for the following purposes.

- **Adjusting the load between physical partitions**
The overall system load is adjusted by the migration of a guest domain from a physical partition that has a heavy load to a physical partition that has a lower load.
- **Maintaining hardware while a guest domain is operating**
To release a system board from a physical partition for maintenance purposes, a guest domain operating in the physical partition is migrated to a different physical partition.

Upon execution of the live migration function, Logical Domains Manager on the source physical partition communicates with that on the destination physical partition to execute the following processes:

- (1) In the destination physical partition, configure a guest domain with the same configuration as that in the source physical partition.
- (2) Copy information from the source guest domain to the destination guest domain. Execution on the source guest domain continues.
- (3) When the copying ends, delete the source logical domain, and start execution on the destination logical domain.

A logical domain can be migrated within the same system and also between different systems. [Figure 7-1](#) shows migration within the same system. [Figure 7-2](#) shows migration between different systems.

Figure 7-1 Migration Within the Same System

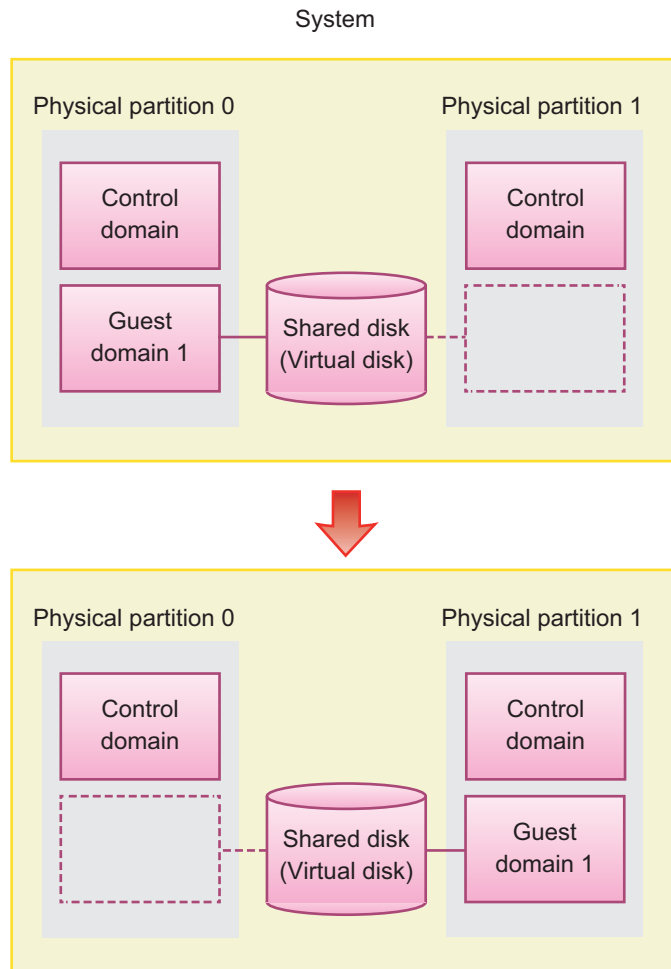
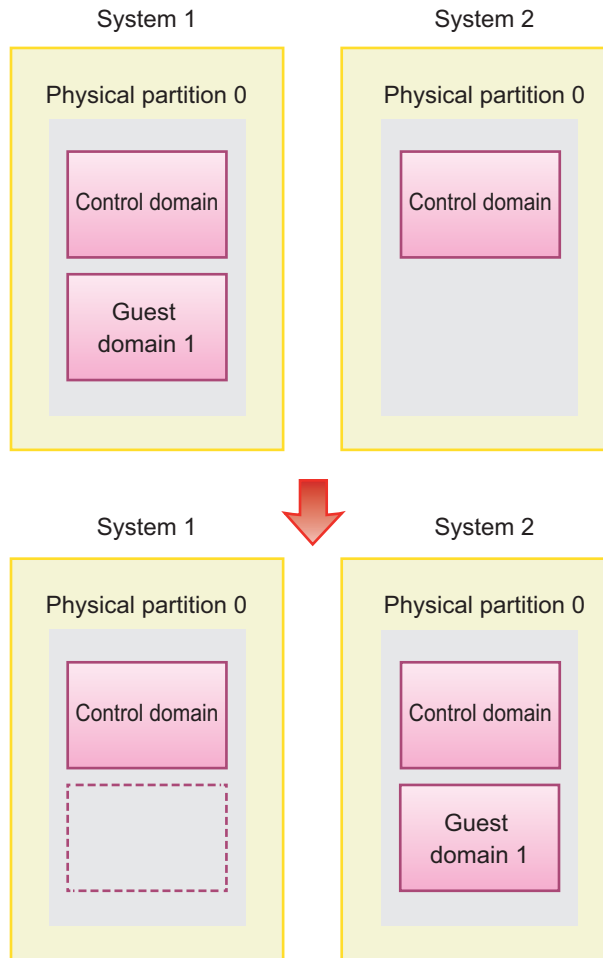


Figure 7-2 Migration Between Different Systems



7.1.1 Requirements for Live Migration

To use the live migration function, the following requirements must be met.

- Logical Domains Manager running at the source and destination
Oracle VM Server for SPARC version 2.1 or later must be installed at the source and destination, where Logical Domains Manager must be running.
- Destination with sufficient free resources
The destination physical partition must have enough available hardware resources for the source guest domain.

There are many other requirements in addition to the above for live migration. For details on those requirements, see "Migrating Domains" in the *Oracle VM Server for SPARC Administration Guide* of the version used, "Notes on live migration of Oracle VM Server for SPARC" in the *Product Notes* of the version used. You can check

whether your environment meets the requirements by executing the `ldm migrate-domain` command with the `-n` option specified. Any item that does not meet the requirements is reported as an error at this time.

7.2 Migrating a Guest Domain

To migrate a guest domain with the live migration function, log in to the control domain, and execute the `ldm migrate-domain` command.

`ldm migrate-domain` command

Note - Only the major options are shown here. For details, see the *Oracle VM Server for SPARC Reference Manual* of the version used.

```
# ldm migrate-domain [-n] source-ldom target-host[:target-ldom]
```

- `-n` option
This option only runs a test to determine if migration is possible. The actual migration is not performed.
- `source-ldom`
Specify the name of the source guest domain.
- `target-host`
Specify the host name of the control domain belonging to the destination physical partition. The host name is a valid host name on the network.
- `target-ldom`
Specify the name of the destination guest domain. If not specified, the name is the same as the source guest domain name.

7.3 Guest Domain Migration Example

This section shows a specific example of live migration. This example migrates the guest domain named `ldom1` to the physical partition of the control domain whose host name is `target1`.

1. **Log in to the control domain as a root user.**
2. **Execute the `ldm migrate-domain` command to migrate the guest domain `ldom1` to the physical partition of the control domain whose host name is `target1`.**

```
# ldm migrate-domain ldom1 target1
```

You are asked to enter the password of the root user of the control domain in the destination physical partition.

3. **Enter the password.**

```
Target Password:*****
```

4. **The guest domain is migrated.**

The guest domain named ldom1 is created on the destination physical partition, and ldom1 is deleted from the source.

Environment Configuration Examples and Procedures Using Physical Partition Dynamic Reconfiguration

This appendix provides examples of and the procedure for environment configuration using physical partition dynamic reconfiguration.

Examples of a system that uses a building block (BB) configuration with Oracle VM Server for SPARC 3.2 or later can be found as "Building a High Availability System" in Technical Information. For details, access the following URL.

<https://www.fujitsu.com/global/products/computing/servers/unix/sparc/downloads/documents/index.html>

The following environment is configured as an example.

- Updating a System That Does Not Support Physical Partition Dynamic Reconfiguration to One That Does Support It
- For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has No Free Space for Hardware Resources
- For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has Free Hardware Resources
- For Expansion of a Newly Installed System of Version XCP 2220 or Later from the 1BB to 2BB Configuration
- For Installation of the System Board Configured by the SPARC64 X+ Processor to the Physical Partition Configured Only by the SPARC64 X Processor
- For Active Replacement of a System Board in a Configuration with Only the Control Domain (for Oracle VM Server for SPARC 3.2 or Later)

For the above cases, the procedure for active replacement using physical partition dynamic reconfiguration is explained below.

- For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has No Free Space for Hardware Resources
- For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has Free Hardware Resources

The actual procedure for active addition using physical partition dynamic reconfiguration is also explained below.

- For Expansion of a Newly Installed System of Version XCP 2220 or Later from the 1BB to 2BB Configuration
- For Installation of the System Board Configured by the SPARC64 X+ Processor to the Physical Partition Configured Only by the SPARC64 X Processor

- [For Active Replacement of a System Board in a Configuration with Only the Control Domain \(for Oracle VM Server for SPARC 3.2 or Later\)](#)

A.1 Updating a System That Does Not Support Physical Partition Dynamic Reconfiguration to One That Does Support It

This section describes an environment configuration example and procedure for when the system is updated to one that can use physical partition dynamic reconfiguration in the SPARC M10. The system to be updated will be of a version prior to XCP 2210 that does not support physical partition dynamic reconfiguration. For details on the physical system board (PSB) and logical system board (LSB), see ["1.3.1 Understanding Physical Partition Components."](#)

A.1.1 Configuration Example

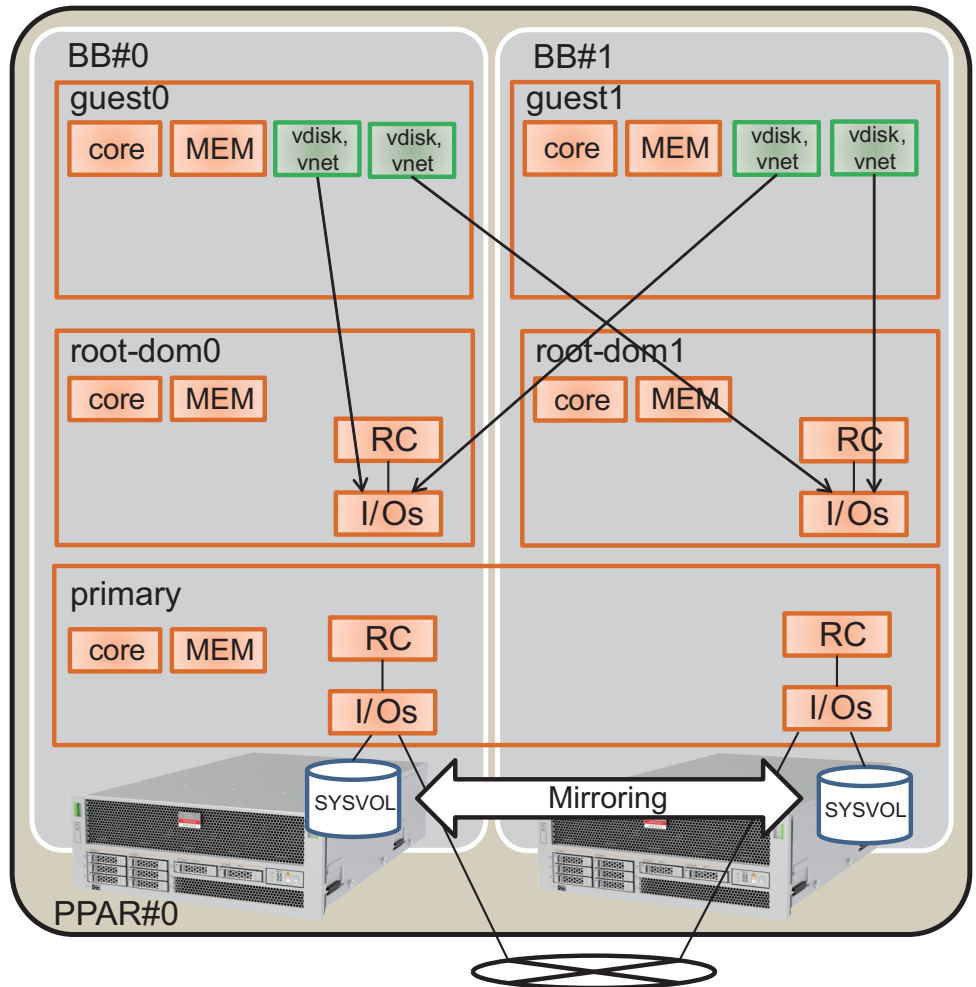
This section describes an example of configuring a logical domain consisting of two SPARC M10-4S, i.e. two system boards (PSB<BB>) (2BB configuration).

To enable the active replacement of the SPARC M10-4Ss with physical partition dynamic reconfiguration, the following conditions must be satisfied.

- Create a redundant configuration by connecting I/O devices under the root complex of each SPARC M10-4S to the system volume I/O devices and the network of the control domain.
This is done for the following reasons: When a SPARC M10-4S is removed, the I/O devices for the remaining SPARC M10-4Ss can be used to continue the operation even if an I/O device configuration change is done together with a restart of the control domain.
- Divide root domains for each SPARC M10-4S.
This is done for the following reasons: When a SPARC M10-4S is removed, the status whereby it is impossible to use services with the physical I/O devices of other SPARC M10-4Ss must be avoided.
- Operate the business with a guest domain consisting of virtual I/O devices (vdisk, vnet).
With this, the root domain is stopped at SPARC M10-4S removal and the physical I/O devices can be disconnected. Moreover, these operations will not influence the guest domain performing the business.

The following figure shows a configuration example in which physical partition dynamic reconfiguration is possible.

Figure A-1 Configuration Example in Which Physical Partition Dynamic Reconfiguration is Possible



A.1.2 Configuration Procedure

The following procedure updates a system that does not support physical partition dynamic reconfiguration to the one that does support it:

1. **Save logical domain configuration information to an XML file.**
If the PPAR DR function is enabled after the XCP firmware is updated, the logical domain configuration returns to the factory setting status (factory-default). So, the logical domain configuration information saved to XSCF cannot be used. For this reason, save the current logical domain configuration information to the XML file and then update the XCP firmware. After updating, restore the logical domain configuration information from the XML file to enable the easy reconfiguration of the logical domain.

The following explains the procedure for saving the logical domain configuration information saved in the XSCF to an XML file.

For details on the configuration information saved to the XML file, however, see the *Oracle VM Server for SPARC Administration Guide*.

- a. **Log in to the control domain.**
- b. **Switch to the configuration information to be saved.**

Execute the `ldm list-spconfig` command on the control domain to list the logical domain configuration information. The line with `[current]` indicates the currently applied configuration information. In the following example, `ldm-set3` is applied.

```
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3 [current]
```

If only `[next poweron]` is displayed, there will be a difference between the logical domain configuration information stored in the XSCF and that stored in the control domain. So, execute the `ldm add-spconfig` command to save the current logical domain configuration information with a different name.

```
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3 [next poweron]
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4 [current]
```

When the logical domain configuration information indicated in `[current]` matches the configuration information you want to save, proceed to step 1-c.

If they are different, execute the `ldm set-spconfig` command to switch to the configuration information you want to save.

In the following example, the configuration information is switched with the `ldm set-spconfig` command and then `ldm-set1` is saved.

```
# ldm set-spconfig ldm-set1
# ldm list-spconfig
factory-default
ldm-set1 [next poweron]
ldm-set2
ldm-set3 [current]
```


Execute the poweroff and poweron commands on XSCF to power off the physical partition (PPAR) and then power it back on. In this case, execute the poweroff command after changing the logical domain state to the state in which Oracle Solaris is running or the inactive state.

The following example powers off PPAR-ID 0 and then powers it back on.

```
XSCF> poweroff -p 0
XSCF> poweron -p 0
```

Execute the ldm list-spconfig command to check that the specified logical domain configuration information is set. The following example indicates that ldm-set1 is set as the current configuration information.

```
# ldm list-spconfig
factory-default
ldm-set1 [current]
ldm-set2
ldm-set3
```

c. Execute the ldm list-constraints command to save the current logical domain configuration information to the XML file.

Execute the ldm list-constraints command to save the current logical configuration information to the XML file.

The following example saves the current logical domain configuration information to ldm-set1.xml. To prevent the saved XML file from being lost, back it up to another media or the like.

```
# ldm list-constraints -x > /ldm-set1.xml
```

d. Check whether the configuration information is saved to the XML file.

Execute the more command of Oracle Solaris or the like to check that the information has been saved to the XML file.

```
# more /ldm-set1.xml
<?xml version="1.0"?>
<LDM_interfaceversion="1.3" xmlns:xsi=http://www.w3.org/2001/XMLSchema-
instance
```

e. To save any other logical domain configuration information, repeat steps 1.b to 1.e.

If there is any other configuration information that you want to save, repeat steps 1.b to 1.e.

2. Save the XSCF setting information.

XSCF setting information can be saved to a USB device or to an external server through a network. The following describes each method.

- **Saving the XSCF setting information to a USB device**

a. **Log in to the master XSCF.**

Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF. If the standby XSCF is being used, log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

b. **Save the setting information to the USB device of the master XSCF.**

i. **Connect the USB device to the USB port on the XSCF unit panel (rear panel) of the master XSCF.**

ii. **Specify the output file name on the local USB device on the XSCF and then execute the `dumpconfig` command.**

The setting information is saved with the specified file name in base64 encoding text format.

```
XSCF> dumpconfig file:///media/usb_msd/backup-file.txt
operation completed
```

When the "operation completed" message appears, data transfer has ended normally.

iii. **Once data transfer has ended, remove the USB device from the USB port.**

c. **Using an editor or the like on a PC, check the following information saved in the setting file.**

- User-Comments: Comments made when the `-c` option was specified in the `dumpconfig` command
- Created: Date and time when the information was saved
- Platform: Model name
- Serial-No: System serial number

```
XSCF Configuration File
User-Comments:
Encrypted: No
Created: Mon Jan 27 13:47:38 2014
Platform: M10-4S
Serial-No: 2111234001
Chassis-serial80:
Chassis-serial81:
Chassis-serial82:
Chassis-serial83:
Chassis-serial00:2111234001
Chassis-serial01:2111234003
Chassis-serial02:
Chassis-serial03:
```

```

Chassis-serial04:
Chassis-serial05:
Chassis-serial06:
Chassis-serial07:
Chassis-serial08:
Chassis-serial09:
Chassis-serial10:
Chassis-serial11:
Chassis-serial12:
Chassis-serial13:
Chassis-serial14:
Chassis-serial15:
Version: 0001
begin-base64_common
U1VOVyxTUEFSQy1FbnRlcnByeXNlAAAAAAAFMz6gAAPrfADhbdAAAAAIyMTEzMjM0MDAzAAAA
...

```

- **Saving the setting information to an external server through a network**

a. **Log in to the master XSCF.**

Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF. If the standby XSCF is being used, log in to the master XSCF again.

```

XSCF> showbbstatus
BB#00 (Master)

```

b. **Specify the target directory through a network to save the XSCF setting information.**

Specify the target directory and output file name and then execute the `dumpconfig` command. The setting information is saved with the specified file name in base64 encoding text format.

```

XSCF> dumpconfig -u "user-name" "ftp://server/backup/backup-sca-
ff2-16.txt"
operation completed

```

When the "operation completed" message appears, data transfer has ended normally.

c. **Using an editor or the like on a PC, check the saved setting file information.**

Check the following information.

- User-Comments: Comments made when the `-c` option was specified in the `dumpconfig` command
- Created: Date and time when the information was saved
- Platform: Model name
- Serial-No: System serial number

```

XSCF Configuration File
User-Comments:
Encrypted: No
Created: Mon Jan 27 13:47:38 2014
Platform: M10-4S
Serial-No: 2111234001
Chassis-serial80:
Chassis-serial81:
Chassis-serial82:
Chassis-serial83:
Chassis-serial00:2111234001
Chassis-serial01:2111234003
Chassis-serial02:
Chassis-serial03:
Chassis-serial04:
Chassis-serial05:
Chassis-serial06:
Chassis-serial07:
Chassis-serial08:
Chassis-serial09:
Chassis-serial10:
Chassis-serial11:
Chassis-serial12:
Chassis-serial13:
Chassis-serial14:
Chassis-serial15:
Version: 0001
begin-base64_common
U1VOVyxTUEFSQy1FbnRlcnByaXNlAAAAAAAAAFLmZ6gAAPrfADhbdAAAAAIyMTExMjMOMDAzAAAA
...

```

3. **Update Oracle Solaris and Oracle VM Server for SPARC.**

Obtain Oracle Solaris and Oracle VM Server for SPARC that support physical partition dynamic reconfiguration and apply them to the system.

a. **Update Oracle VM Server for SPARC of the control domain.**

See the latest *Product Notes* for your server to check the latest versions of Oracle Solaris and Oracle VM Server for SPARC. Then, update Oracle VM Server for SPARC using the following procedure.

i. **Obtain Oracle VM Server for SPARC for use with the control domain.**

- **If the control domain is Oracle Solaris 10**

Obtain Oracle VM Server for SPARC that supports physical partition dynamic reconfiguration. For details on how to obtain this, see the latest *Product Notes* for your server.

- **If the control domain is Oracle Solaris 11**

Obtain SRU11.1.14.0 or later.

ii. **When Oracle VM Server for SPARC or Oracle Solaris for the control domain has been updated, restart the control domain.**

Restarting the control domain may cause an I/O domain to panic or a guest domain I/O to stop. If such a logical domain exists, stop it in advance

with the shutdown command or stop it by executing the ldm stop-domain command from the control domain.

Using the following method, you can check whether the logical domain is to be stopped.

- For an I/O domain to which the PCIe endpoint is assigned from the control domain

Execute the ldm list-io command to determine whether the root complex (BUS) of the PCIe endpoint assigned to the logical domain is assigned to primary. The following example indicates that the bus "PCIE2" of PCIe endpoints "/BB0/PCI3" and "/BB0/PCI4" assigned to iodom0 is assigned to primary (control domain). Note that, since the configuration explained here differs from that in ["A.1.1 Configuration Example,"](#) changes have been made to the logical domains and some other information.

```
# ldm list-io
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
PCIE0                              BUS     PCIE0    primary   IOV
PCIE1                              BUS     PCIE1    primary   IOV
PCIE2                              BUS     PCIE2    primary   IOV
PCIE3                              BUS     PCIE3    primary   IOV
....
/BB0/CMUL/NET0                     PCIE    PCIE0    primary   OCC
/BB0/CMUL/SASHBA                   PCIE    PCIE0    primary   OCC
/BB0/PCI0                          PCIE    PCIE1    primary   OCC
/BB0/PCI3                          PCIE    PCIE2    iodom0    OCC
/BB0/PCI4                          PCIE    PCIE2    iodom0    OCC
/BB0/PCI7                          PCIE    PCIE3    primary   OCC
/BB0/PCI8                          PCIE    PCIE3    primary   EMP
....
```

- For the guest domain to which the virtual service of the control domain is assigned

Execute ldm list-bindings primary to check the correspondence between the virtual network switch (VSW) and the connection destination (PEER), as well as between the virtual disk service (VDS) and the logical domain (CLIENT) that uses a virtual disk. In the following example, you can check that "guestdom0" is set for VSW PEER and VDS CLIENT.

```
# ldm list-bindings primary
....
VSW
NAME                                MAC                                NET-DEV  ID  DEVICE      LINKPROP
DEFAULT-VLAN-ID PVID
VID                                MTU   MODE  INTER-VNET-LINK
vsw0                                00:14:4f:f9:88:ca net0      0   switch@0    1
1
PEER                                MAC                                PVID VID
```

MTU	MAXBW	LINKPROP			
INTERVNETLINK					
	vnet0@guestdom0		00:14:4f:fa:64:dd	1	
1500					
VDS					
	NAME	VOLUME	OPTIONS	MPGROUP	DEVICE
....					
	CLIENT	VOLUME			
	vdisk0@guestdom0	vol0			

Execute the ldm stop-domain command to stop the logical domain checked above. In the following example, the ldm stop-domain command is executed to stop iodom0 and guestdom0 and then the ldm list-domain command is executed to check that the bound state is set.

# ldm stop-domain guestdom0							
LDom guestdom0 stopped							
# ldm stop-domain iodom0							
LDom iodom0 stopped							
# ldm list-domain							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	58G	0.0%	6h 3m
guestdom0	bound	-----	5100	64	64G		
iodom0	bound	----v-	5000	32	32G		

iii. Update Oracle VM Server for SPARC of the control domain.

- For Oracle Solaris 10

Uninstall the old Oracle VM Server for SPARC and then install the new Oracle VM Server for SPARC. For details, see the README file provided with the obtained Oracle VM Server for SPARC.

- For Oracle Solaris 11

Apply the SRU. For details, see the installation manual of the obtained SRU.

iv. Execute the shutdown command to restart the control domain.

shutdown -i6 -g0 -y
....

v. Execute the ldm start-domain command to start the logical domain stopped in ii, above.

In the following example, the ldm start-domain command is executed to start "iodom0" and "guestdom0" and then the ldm list-domain command is executed to check that the active state is set.

# ldm start-domain guestdom0							
# ldm start-domain iodom0							
# ldm list-domain							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME

primary	active	-n-cv-	UART	64	58G	0.0%	6h 3m
guestdom0	active	-n----	5100	64	64G		
iodom0	active	-n--v-	5000	32	32G		

b. Update Oracle Solaris for logical domains other than the control domain.

See the latest *Product Notes* for your server, and update Oracle Solaris for logical domains other than the control domain. For details on the update procedure, see the information relating to each update.

4. Update the XCP firmware.

Install the PSB (BB) mounted on the SPARC64 X+ processor in the physical partition configured with the SPARC64 X processor. To do this, update the physical partition configured with the SPARC64 X processor to the XCP firmware of XCP 2220 or later.

a. Obtain the latest XCP firmware.

i. Download the program file for the XCP firmware.

Download, from the web site, the program file (XCPxxxx.tar.gz or XCPxxxx.exe) for the XCP firmware to any folder on a PC connected to this system.

Use one of the following methods to obtain the firmware for the server you are using.

- Japanese site

A customer who has entered into a contract with the SupportDesk can obtain firmware from SupportDesk-Web.

- Global site

For details on the method used to obtain the latest file of the firmware, contact our sales personnel.

The following files are provided.

- Firmware program file
(XSCF Control Package (XCP) file)
- XSCF extended MIB (XSCF-SP-MIB) definition file

ii. Check the XCP version of the downloaded program file.

Check the version of the program file of the downloaded XCP firmware. For the XCP version, refer to the 4-digit number in the file name of the firmware program (tar.gz format) to check that it is the XCP firmware version update to be applied. For example, if the name of the program file is "XCP2220.tar.gz," the XCP version is 2220.

iii. Decompress the downloaded program file.

Decompress the program file of the downloaded XCP firmware. The XCP image file to import into the system is expanded. For example, if "XCP2220.tar.gz" is decompressed, "BBXCP2220.tar.gz" is expanded.

b. Check the current XCP firmware version.

i. Log in to the master XSCF.

Execute the showbbstatus command to check that the XSCF to which you

have logged in is the master XSCF. If the standby XSCF is being used, log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

ii. Execute the version command to check the XCP version of the current system.

Before updating the firmware, check the XCP version of the current system. In the following example, the version command is executed with the -c xcp option added, to check if the XCP version is XCP 2041 that does not support physical partition dynamic reconfiguration.

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2041
XCP1 (Reserve): 2041
BB#01-XSCF#0 (Standby)
XCP0 (Current): 2041
XCP1 (Reserve): 2041
```

c. Update the XCP firmware.

i. Execute the poweroff command to power off all the physical partitions.

```
XSCF> poweroff -a
```

Execute the showpparstatus command to check that all the physical partitions are powered off.

```
XSCF> showpparstatus -a
PPAR-ID          PPAR Status
0                Powered Off
```

ii. Execute the showhardconf command to check that [Status] of the master and standby XSCFs is "Normal."

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Service;
+ System_Power:On; System_Phase:Cabinet Power On;
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
+ FRU-Part-Number: CA20393-B50X A2 ;
```



```

+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB;
:

```

iii. Execute the **getflashimage** command to import the XCP image file.

In the following example, the USB device is connected to the USB port (MAINTENANCE ONLY printed) on the XSCF unit panel (rear panel) of the master XSCF and the XCP image file is imported.

```

XSCF> getflashimage file:///mnt/share/scf-firm/xscf/user/scfadmin/BBXCP2220.tar.gz
0MB received
1MB received
2MB received
...
86MB received
87MB received
88MB received
Download successful: 90539 Kbytes in 58 secs (1562.668 Kbytes/sec) Checking
file...
MD5: 2b89c06548205ce35a8ecb6c2321d999

```

When the normal end messages "Download successful: ..." and "MD5: ..." appear, they indicate that the importing of the XCP image file has ended.

Note - The message "Warning: About to delete existing old versions." may appear when the XCP image file is imported. This message asks you to confirm whether to delete an old XCP image file that has already been imported. If "Continue?" appears, enter "y" to continue import processing.

Note - If the message "Error:File is invalid or corrupt" appears after the XCP image file is imported, this indicates that the imported XCP image file is inappropriate. The XCP image file may have been destroyed. So, obtain the correct XCP image file and then import it.

iv. Execute the **getflashimage -l** command to check the version of the imported XCP image file.

```

XSCF> getflashimage -l
Existing versions:
      Version              Size  Date
      BBXCP2220.tar.gz    92712351  Thu May 23 15:01:42 JST 2014

```

v. Execute the **flashupdate -c check** command to check whether the imported XCP image file can be used for update.

Execute the **showresult** command immediately after the execution of the **flashupdate** command. If the end value is 0, update is possible.

```
XSCF> flashupdate -c check -m xcp -s 2220
XCP update is started. [3600sec]
0XSCF>
XSCF> showresult
0
XSCF>
```

vi. **Execute the flashupdate command to update the firmware.**

```
XSCF> flashupdate -c update -m xcp -s 2220
The XSCF will be reset. Continue? [y|n]: y
XCP update is started. [3600sec]
0.....30.....60.....90.....120.....150.....180.....210.....240.....-
270.....300.....330.....360.....390.....420.....450.....480.....510.....|
540.....570.....600.....630.....660.....690.....720.....750.....780.....-
810.....840.....870.....900.....930
:
```

Here, the XSCF is rebooted and the XSCF session is disconnected. At this point, the XCP firmware update has not yet been completed.

Note - Regarding the work time for the firmware, the update takes about 60 minutes and the automatic switching of the XSCF after the completion of the update takes about 10 minutes.

Note - To perform the update safely, do not operate the power supply of the physical partition until the message "XCP update has been completed" appears, to indicate that the XCP firmware has been updated.

vii. **Connect to the master XSCF again.**

Immediately after an XSCF reboot, the master and standby XSCFs will be the exact reverse of the original states. For example, if the firmware update is executed with the master XSCF of BB-ID 0 and a connection is then made to the XSCF again, BB-ID 1 enters the master state and BB-ID 0 enters the standby state.

Note - If an inherited IP address is set and is used for connection, a connection is automatically made to the master XSCF.

viii. **Execute the showbbstatus command to check that you have logged in to the master XSCF. If the standby XSCF is being used, connect to the master XSCF again.**

```
XSCF> showbbstatus
BB#01 (Master)
```

ix. **Execute the showlogs monitor command to check that the XCP firmware update has been completed.**

If the message "XCP update has been completed" appears, XCP firmware update has been completed.

```
XSCF> showlogs monitor
May 29 09:38:05 M10-4S-0 Event: SCF:XCP update is started (XCP version=2220:
last version=2041)
May 29 09:40:31 M10-4S-0 Event: SCF:XSCF update is started (BBID=0, bank=0)
May 29 09:40:46 M10-4S-0 Event: SCF:XSCF update is started (BBID=1, bank=0)
May 29 09:41:03 M10-4S-0 Event: SCF:XSCF writing is performed (BBID=0, XSCF
version=02220000)
:
May 29 09:50:41 M10-4S-0 Event: SCF:XSCF bank apply has been completed
(BBID=1, bank=0, XCP version=2220: last version=2041)
May 29 10:01:50 M10-4S-0 Event: SCF:XSCF bank apply has been completed
(BBID=0, bank=0, XCP version=2220: last version=2041)
May 29 10:01:51 M10-4S-0 Event: SCF:Change Master Start (BB#01)May 29 10:03:26
M10-4S-1 Event: SCF:Change Master Complete (BB#01)
May 29 10:05:00 M10-4S-1 Event: SCF:Standby XSCF Ready (BBID#00)
:
May 29 10:32:38 M10-4S-1 Event: SCF:XCP update has been completed (XCP
version=2220: last version=2041)
May 29 10:32:39 M10-4S-1 Event: SCF:This XSCF will be switched back to standby
mode after completion of firmware update
May 29 10:32:39 M10-4S-1 Event: SCF:Change Master Start (BB#00)
May 29 10:33:29 M10-4S-1 Event: SCF:Change Master Complete (BB#00)
May 29 10:42:29 M10-4S-1 Event: SCF:Standby XSCF Ready (BBID#01)
```

Subsequently, the master and standby states of the XSCF are automatically switched.

Switching has been completed when the messages "This XSCF will be switched back to standby mode after completion of firmware update", "Change Master Complete", and "Stand by XSCF Ready" appear.

Note - If the message "XCP update has been completed" does not appear, update has not yet been completed. Execute the showlogs monitor command again to check that update has been completed.

Note - If firmware update is executed, the master and standby states are switched twice. As a result, the master XSCF returns to the XSCF for which the flashupdate command was executed. When the master and standby states are switched, the XSCF session may be disconnected. If the XSCF session has been disconnected, connect to it again.

To check that switching has been completed, execute the showhardconf command to check that [Status] of the master and standby XSCFs is "Normal."

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Service;
+ System_Power:On; System_Phase:Cabinet Power On;
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1;
```

```

+ Power_Supply_System: ;
+ Memory_Size:256 GB;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
+ FRU-Part-Number: CA20393-B50X A2 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB;
:

```

Note - Suppose that the showhardconf command is executed and the message "Cannot communicate with the other XSCF. Check the other XSCF's state." appears. This indicates that switching between the master and standby states of the XSCF has not yet been completed. Execute the showhardconf command again to check that switching has been completed.

x. To return the master and standby XSCFs to the states they had before the update, execute the switchscf command.

```

XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y

```

Since the XSCF session is disconnected by an XSCF reboot, connect to the master XSCF again.

To check that switching has been completed, execute the showhardconf command to check that [Status] of the master and standby XSCFs is "Normal."

Note - Suppose that the showhardconf command is executed and the message "Cannot communicate with the other XSCF. Check the other XSCF's state." appears. This indicates that switching between the master and standby states of the XSCF has not yet been completed. Execute the showhardconf command again to check that switching has been completed.

xi. Execute the version command to check that the firmware is the new version.

```

XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2220
XCP1 (Reserve): 2220
BB#01-XSCF#0 (Standby)
XCP0 (Current): 2220
XCP1 (Reserve): 2220

```

Note - If the physical partition is powered on and the firmware is updated, the current bank

of the CMU firmware will be new. If the physical partition is powered off and the firmware is updated, both the reserve bank and current bank of the CMU firmware will be new. For details on the versions of the CMU firmware associated with the XCP versions, see "Existing XCP Firmware Versions and Support Information" in the latest product notes.

5. **Specify the XSCF PPAR DR mode and logical domain to be started, and then start the physical partition.**

Note - To use physical partition dynamic reconfiguration, it is necessary to enable the PPAR DR mode using the `setpparmode` command. However, if the PPAR DR mode is enabled, the logical domain configuration information is changed to the factory-default. So, the existing logical domain configuration information cannot be used. In this case, you must create new logical domain configuration information.

To use the existing logical domain configuration information, disable the PPAR DR mode. Suppose that the PPAR DR mode is enabled and that the power to the physical partition is turned on while the existing logical domain configuration information is used. Problems such as a hypervisor abort and a failure to start Oracle VM Server for SPARC normally will occur.

a. **Enable the physical partition dynamic reconfiguration function (PPAR DR function).**

i. **Execute the `showpparmode` command to reference the PPAR DR mode setting.**

The following example specifies physical partition number 0 (PPAR ID 0).

```
XSCF> showpparmode -p 0
Host-ID                :9007002b
Diagnostic Level       :min
Message Level         :normal
Alive Check           :on
Watchdog Reaction     :reset
Break Signal          :on
Autoboot (Guest Domain) :on
Elastic Mode          :off
IOreconfigure         :false
PPAR DR (Current)     :-
PPAR DR (Next)        :off
```

ii. **If the PPAR DR mode is disabled, execute the `setpparmode` command to enable that mode.**

```
XSCF> setpparmode -p 0 -m ppar_dr=on
```

iii. **Execute the `showpparmode` command to check that the PPAR DR mode has been enabled.**

```
XSCF> showpparmode -p 0
Host-ID                :9007002b
Diagnostic Level       :min
```

Message Level	:normal
Alive Check	:on
Watchdog Reaction	:reset
Break Signal	:on
Autoboot (Guest Domain)	:on
Elastic Mode	:off
IOreconfigure	:false
PPAR DR (Current)	:-
PPAR DR (Next)	:on

b. Power on the physical partition.

- i. Execute the `showdomainconfig` command to check that the configuration information for the logical domains to be started is set to the factory-default.**

The following example specifies physical partition number 0 (PPAR ID 0).

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) : factory-default
(Next) : factory-default
-----
Index :1
config_name :factory-default
domains :1
date_created:-
-----
Index :2
config_name :ldm-set1
domains :8
date_created:'2012-08-0811:34:56'
-----
Index :3
config_name :ldm-set2
domains :20
date_created:'2012-08-0912:43:56':
```

- ii. Execute the `poweron` command to power on the physical partitions.**

The following example specifies physical partition number 0 (PPAR ID 0).

```
XSCF> poweron -p 0
```

6. Using the XML file, reconfigure the logical domain.

After completing the startup of the physical partition, reconfigure the logical domain from the XML file saved in step 1.

- a. Check that the system has started with the factory-default settings for the control domain.**

Execute the `ldm list-sponfig` command to check that [current] is displayed on the factory-default line.

```
# ldm list-spconfig
factory-default [current]
ldm-set1
ldm-set2
ldm-set3
```

b. Using the ldm init-system command, apply the logical domain configuration information from the XML file.

Execute the ldm init-system command to apply the saved XML file setting. Then, execute the shutdown command to restart the control domain.

```
# ldm init-system -i /ldm-set1.xml
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
# shutdown -y -g0 -i6
```

c. To enable physical partition dynamic reconfiguration, adjust the memory size of each logical domain.

Execute the ldm list-domain command or the like to check that the logical domains have been configured from the XML file.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	58G	0.0%	6m
guest0	inactive	-----		64	64G		
guest1	inactive	-----		64	64G		
root-dom0	inactive	-----		32	32G		
root-dom1	inactive	-----		32	32G		

To enable physical partition dynamic reconfiguration when the version of Oracle VM Server for SPARC is earlier than 3.2, set the memory size to be assigned to the logical domain to a multiple of "number of cores assigned to the logical domain x 256 MB". To re-set the memory size of a logical domain other than the control domain, use the ldm set-memory command.

To re-set the memory size of the control domain, do so as follows: Enter delayed reconfiguration mode with the ldm start-reconf command, set the same number of cores with the ldm set-core command, and then re-set the memory size with the ldm set-memory command. Then, restart Oracle Solaris.

When the version of Oracle VM Server for SPARC is earlier than 3.2, first, obtain a multiple of 8192 MB to make the value closer to the original setting value (58 GB = 59392 MB). The result is $59392/8192 = 7.25$. So, round it down to 7. As a result, the memory size to be reassigned to the control domain is $8192 \text{ MB} \times 7 = 56 \text{ GB}$.

Next, the following example shows the execution of the command for re-setting the control domain memory to 56 GB.

```
# ldm start-reconf primary
```

Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary domain reboots, at which time the new configuration for the primary domain will also take effect.

```
# ldm set-core 32 primary
```

Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.

```
# ldm set-memory 56G primary
```

Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.

```
# shutdown -i6 -g0 -y
```

```
....
```

Execute the `ldm list-domain` command to check that the memory size (MEMORY) has been set properly.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.0%	6m
guest0	inactive	-----		64	64G		
guest1	inactive	-----		64	64G		
root-dom0	inactive	-----		32	32G		
root-dom1	inactive	-----		32	32G		

Execute the `ldm bind-domain` command to bind each logical domain.

```
# ldm bind-domain root-dom0
```

```
# ldm bind-domain root-dom1
```

```
# ldm bind-domain guest0
```

```
# ldm bind-domain guest1
```

However, suppose that the logical domain configuration information is restored from an XML file that was saved when the PPAR DR function was disabled. In this case, the memory size of the hypervisor area increases when the PPAR DR function is enabled, such that the amount of allocated memory will be insufficient. If the `ldm bind-domain` command is executed while insufficient memory is allocated, the processing will end with an error together with the following error message, and the inactive state remains as is.

```
# ldm bind-domain guest1
```

```
Not enough free memory present to meet this request.  
Could not bind requested memory for LDom guest1.
```

In this case, see step 6.c to re-set the memory size allocated to the logical

domains. To adjust the memory allocation of a logical domain that has already been placed in the bound state, execute the `ldm unbind-domain` command to release the bound state and then make the setting. When re-setting has been completed, re-execute the `ldm bind-domain` command for re-binding.

d. **Check the memory resource allocation status.**

Execute the `ldm list-devices -a` command to check the status of the continuous area (memory block) of the memory allocated to each logical domain. Then, check that there are not a large number of small memory blocks. As a guide, check that there is not a large number of memory blocks with a SIZE of about 256 MB to 512 GB.

# ldm list-devices -a memory		
MEMORY		
PA	SIZE	BOUND
0x700000000000	32G	root-dom0
0x720000000000	32G	root-dom1
0x740000000000	32G	guest0
0x760000800000	1272M	_sys_
0x760050000000	2G	primary
0x7600d0000000	29440M	guest1
0x780000000000	32G	guest0
0x7a0000000000	32G	guest1
0x7c0000000000	32G	primary
0x7e0000800000	1272M	_sys_
0x7e0050000000	512M	_sys_
0x7e0070000000	256M	_sys_
0x7e0080000000	3328M	guest1
0x7e0150000000	4864M	
0x7e0280000000	22G	primary

If the memory block is broken up, execute the `ldm unbind-domain` command to restore each logical domain to the unbind status. Then, perform a re-set as described in step 6.c.

e. **Start all the logical domains.**

Execute the `ldm start-domain` command to start the logical domains other than the control domain. Suppose that there are dependency relationships such as a physical I/O allocation relationship (root domain, I/O domain) and relationships between virtual I/O services and clients. In this case, first start the supply source domain of the physical and virtual I/O services.

The following example shows that domains are started in the order assumed for the configuration example. The `ldm start-domain` command is used to start `root-dom0`, `root-dom1`, `guest0`, and `guest1`, in that order, and the `ldm list-devices` command is used to check that these domains are started. If [STATE] is "active" and the second character from the left of the FLAGS string is "n", this indicates that Oracle Solaris has started.

```
# ldm start-domain root-dom0
LDom root-dom0 started
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm start-domain guest0
LDom guest0 started
# ldm start-domain guest1
LDom guest1 started
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.2%	15m
guest0	active	-n----	5002	64	64G	0.4%	18s
guest1	active	-n----	5003	64	64G	7.9%	9s
root-dom0	active	-n--v-	5000	32	32G	0.5%	24s
root-dom1	active	-n--v-	5001	32	32G	1.6%	16s

f. Save the logical domain configuration information to the XSCF.

After restoring the logical domain configuration information, execute the ldm add-spconfig command to save the restored logical domain configuration information to the XSCF. The following example saves the ldm-set4 configuration information to the XSCF.

```
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4 [current]
```

g. If there are multiple items of logical domain configuration information to be restored, restore the factory-default configuration and then repeat steps 6.b to 6.g.

The following example shows the return to the factory-default configuration. Execute the showdomainconfig command from the XSCF to check the configuration information for the logical domain to be started.

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) :ldm-set1
(Next) :ldm-set1
-----
Index :1
config_name :factory-default
domains :1
date_created:-
-----
Index :2
config_name :ldm-set1
domains :8
```

```
date_created:'2012-08-0811:34:56'
```

```
-----  
....
```

Execute the `setdomainconfig` command to specify factory-default Index1 in physical partition number 0 (PPAR ID 0).

```
XSCF> setdomainconfig -p 0 -i 1
```

Execute the `poweroff` and then the `poweron` commands to restart the physical partition. Once startup is complete, repeat steps 6.b to 6.g.

```
XSCF> poweroff -p 0  
...  
XSCF> poweron -p 0  
...
```

A.2 For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has No Free Space for Hardware Resources

[“A.2 For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has No Free Space for Hardware Resources”](#) to [“A.4 For Expansion of a Newly Installed System of Version XCP 2220 or Later from the 1BB to 2BB Configuration”](#) describe an example of a system configuration that can use physical partition dynamic reconfiguration with a 2BB configuration of the SPARC M10-4S, and describe the procedure for constructing a new configuration.

However, a description is provided, assuming that the SPARC M10-4S uses XCP 2220 or later as its firmware. It also assumes that the XSCF has been set up, and the system has been installed with a 2BB configuration, as described in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

This section describes an example of the procedure for constructing a configuration with no free space for the hardware resources (CPU core and memory) required to move resources upon SPARC M10-4S disconnection in a physical partition consisting of two SPARC M10-4Ss, i.e. two system boards (PSB<BB>). This section also describes an example of the procedure for performing active replacement of the SPARC M10-4S. However, this case leads to a reduction in the number of CPU cores and the amount of memory temporarily assigned to the logical domain at the active replacement of the SPARC M10-4S.

This description also applies to SPARC M12.

A.2.1 Configuration Example

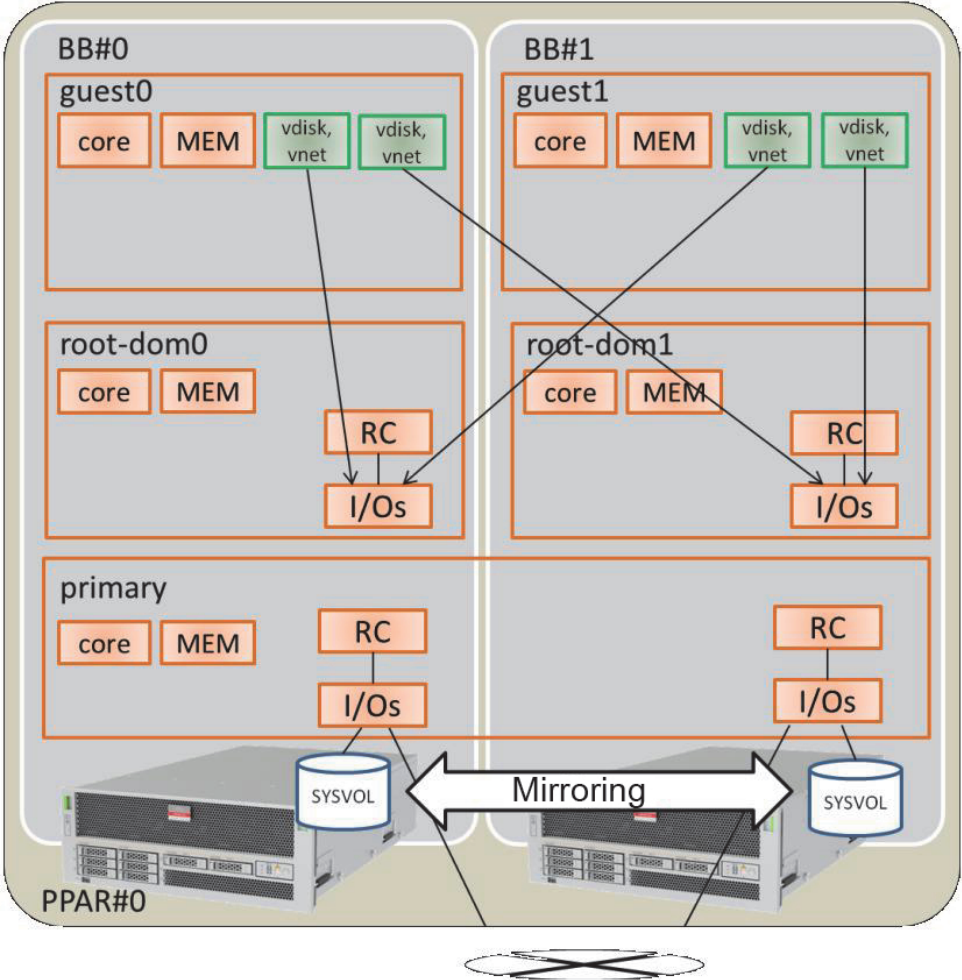
This section describes an example of configuring a logical domain consisting of two SPARC M10-4S, i.e. two system boards (PSB<BB>) (2BB configuration).

To enable physical partition dynamic reconfiguration in a physical partition, the I/O configuration must satisfy the following conditions.

- The system volume disk and network interface of the control domain must be made redundant by connecting the I/O of each SPARC M10-4S. This is done to make it possible to continue operation with the I/O of the remaining SPARC M10-4Ss when a SPARC M10-4S is removed.
- It is necessary to place a root domain for each SPARC M10-4S and to use that root domain as the dedicated one for providing a virtual I/O to the guest domain. The reason for this is as follows: If the root domain is being used for business, it must be stopped upon the disconnection of the SPARC M10-4S and so the business will stop.
- The business must be operated in the guest domain. The virtual I/O service of each root domain must be assigned to the virtual I/O devices (vdisk, vnet) of this guest domain to make those virtual I/O devices redundant. The reason for this is as follows: Even if the SPARC M10-4S is disconnected on one side, I/O access can be continued by using the virtual I/O service of the root domain on the remaining SPARC M10-4S. In this way, disconnecting the SPARC M10-4S does not influence the guest domain, allowing business operations to continue.

The following is a schematic diagram of a configuration that satisfies the above conditions. Moreover, the I/O configuration of each logical domain is simplified.

Figure A-2 Configuration Example in Which Operations Can Continue in the 2BB Configuration



The following table gives an example of the CPU cores, memory, and I/O configuration to be assigned to the control domain, root domain, and guest domain in the 2BB configuration. There is no empty resource which is large enough to enable movement of the CPU core and memory assigned to each logical domain upon SPARC M10-4S disconnection.

Table A-1 Example of the 2BB Configuration With No Empty Hardware Resources

Logical Domain Name	primary	root-dom0	root-dom1	guest0	guest1
CPU core (activation: 128)	32	16	16	32	32
Memory (Total: 256 GB)	56 GB (*1)	32 GB	32 GB	64 GB	64 GB

Table A-1 Example of the 2BB Configuration With No Empty Hardware Resources (continued)

Logical Domain Name	primary	root-dom0	root-dom1	guest0	guest1
Physical I/O (Root complex assignment)	On-board #0 (PCIE0, PCIE4) On-board #1 (PCIE8, PCIE12)	PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, PCIE7	PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, PCIE15	-	-
Virtual I/O device	-	vds0 vsw0, vsw1	vds1 vsw10, vsw11	vdisk0, vdisk10 vnet0, vnet10	vdisk1, vdisk11 vnet1, vnet11

*1 In the configuration example shown above, the size of memory to be assigned to the control domain is calculated as the remainder of the memory after assignment to the other logical domains.

First, when the remaining size, calculated by subtracting the sizes assigned to domains other than the control domain from the total capacity of the physical memory, is obtained, it is 64 GB.

$$256\text{ GB} - 32\text{ GB} - 32\text{ GB} - 64\text{ GB} - 64\text{ GB} = 64\text{ GB}$$

The maximum size to be actually assigned to the control domain is up to the size (60.75 GB) calculated by subtracting, from the above result, the memory size (2 GB + 1.25 GB) assigned to the hypervisor.

$$64\text{ GB} - 2\text{ GB} - 1.25\text{ GB} = 60.75\text{ GB}$$

In addition, when the version of Oracle VM Server for SPARC is earlier than 3.2, it is 56 GB when the calculation is made with a maximum size of less than 60.75 GB to make it a multiple of (number of cores x 256 MB) as a condition for performing physical partition dynamic reconfiguration.

$$60.75\text{ GB} / (32\text{ cores} \times 256\text{ MB}) = 7.59$$

To make it an integral multiple, round it down to "7."

$$(32\text{ cores} \times 256\text{ MB}) \times 7 = 56\text{ GB}$$

Note - You should design the size of memory to be assigned to logical domains by noting the following two points. For details, see "2.4.1 Consideration of Logical Domain Configuration."

- The size of memory to be assigned to logical domains becomes smaller than the physical memory size.
- When the version of Oracle VM Server for SPARC is earlier than 3.2, use a multiple of (number of cores x 256 MB) to enable physical partition dynamic reconfiguration.

A.2.2 Example of the Physical Partition Configuration Procedure

1. **Log in to the master XSCF.**
Execute the showbbstatus command to check that the XSCF to which you have logged in is the master XSCF.

If you have logged in to a standby XSCF, log out and then log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```
2. **Set mirror mode for memory.**
 - a. **Execute the showfru command to check mirror mode of the memory on the SPARC M10-4S system board (PSB<BB>).**

The following example displays the setting information for devices under

PSB 00-0.

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
  cpu 00-0-0 no
  cpu 00-0-1 no
  cpu 00-0-2 no
  cpu 00-0-3 no
```

- b. To use the memory mirror mode, execute the setupfru command to set that mode.**

If you are not using memory mirror mode, this step is not necessary.

For details on memory mirror mode, see "14.1 Configuring Memory Mirroring" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

The following example places all the CPUs under PSB 00-0 in memory mirror mode.

```
XSCF> setupfru -m y sb 00-0
```

Note - When SPARC M12 is used, specify -c mirror=yes to create a mirroring configuration for the memory.

Example: XSCF> **setupfru -c mirror=yes sb 00-0**

Execute the showfru command to check the setting of memory mirror mode.

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
  cpu 00-0-0 yes
  cpu 00-0-1 yes
  cpu 00-0-2 yes
  cpu 00-0-3 yes
```

- 3. Create the physical partition configuration information.**
 - a. Execute the showpcl command to check the physical partition configuration information.**

```
XSCF> showpcl -p 0
PPAR-ID LSB PSB Status
```

- b. Execute the setpcl command to register the system board in the physical partition configuration information.**

Execute the setpcl command to register the SPARC M10-4S system board in the physical partition configuration information for the built-in destination.

In the following example, physical system boards (PSB 00-0 and PSB 01-0) are mapped to logical system boards (LSB 00 and LSB 01) of physical partition 0.

```
XSCF> setpcl -p 0 -a 00=00-0 01=01-0
```

- c. **Execute the showpcl command to check the physical partition configuration information.**

Check the physical partition configuration information.

```
XSCF> showpcl -p 0
PPAR-ID    LSB      PSB      Status
00
           00      00-0     Running
           01      01-0
```

Specify the -v option to display details on the configuration policy, IO nullification option (No-IO), and memory nullification option (No-Mem).

```
XSCF> showpcl -v -p 0
PPAR-ID    LSB      PSB      Status    No-Mem    No-IO    Cfg-policy
00
           00      00-0     Running
           01      01-0
                                False     False    System
                                False     False
```

Use the setpcl command to change the setting of the configuration policy, IO nullification option (No-IO), and memory nullification option (No-Mem).

For details on the setpcl command, see the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual*.

4. **Assign a system board (PSB<BB>) to a physical partition.**
 - a. **Execute the showboards-a command to check the PSB status.**

Execute the showboards -a command to check that each PSB status is "SP" (system board pool).

```
XSCF> showboards -a
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test  Fault
----  -
00-0 SP          Available  n    n    n    Passed Normal
01-0 SP          Available  n    n    n    Passed Normal
```

- b. **Execute the addboard -c assign command to assign the PSBs.**

```
XSCF> addboard -c assign -p 0 00-0 01-0
```

- c. **Execute the showboards-p command to check the PSB status.**

Execute the showboards -p command to check the status of each PSB assigned to the physical partition.

This example checks that the [Assignment] field of each PSB becomes "Assigned" since each PSB has been normally assigned to physical partition 0.

```
XSCF> showboards -p 0
```

PSB	PPAR-ID (LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00 (00)	Assigned	n	n	n	Passed	Normal
01-0	00 (01)	Assigned	n	n	n	Passed	Normal

5. **Register the CPU Activation key to assign CPU core resources.**
 - a. **Execute the showcodactivation command to check the information on the CPU Activation key.**

Execute the showcodactivation command to check whether the physical partition contains an assignable CPU Activation key.

If only the header is displayed, the CPU Activation key is not registered in the XSCF.

```
XSCF> showcodactivation
```

Index	Description	Count
-----	-----	-----

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and add the CPU Activation keys.

- b. **Execute the addcodactivation command to register the CPU Activation key.**

For details on how to register a CPU Activation key, see "5.3 Adding CPU Core Resources" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

```
XSCF> addcodactivation "Product: SPARC M10-4S
SequenceNumber:10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxUDdPQHKbtXgvZPsrtYguqiNUieB+mTDC
:
:
b1GckFx1RH27FdVHiB2H0A=="
AboveKeywillbeadded,Continue?[y|n]:y
```

- c. **Execute the showcodactivation command to check the information on the CPU Activation keys.**

Execute the showcodactivation command to check whether the physical partition contains an assignable CPU Activation key.

```
XSCF> showcodactivation
Index      Description Count
-----
          0  PROC          2
          1  PROC          2
          2  PROC          2
          3  PROC          2

--- Omitted ---

        62  PROC          2
        63  PROC          2
```

d. **Execute the setcod command to assign CPU core resources.**

Execute the setcod command to assign the CPU core resources to the physical partition.

The following example assigns 128 CPU core resources to physical partition 0.

```
XSCF> setcod -p 0 -s cpu -c set 128
PROC Permits assigned for PPAR 0 : 0 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

XSCF> **setcod -s cpu**

Execute the showcod command to check information on the CPU core resources assigned to the physical partition.

The following example confirms that 128 CPU core resources have been assigned to physical partition 0 with the setcod command that was just executed.

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 128
```

6. **Execute the resetdateoffset command to reset the difference with the time managed by the XSCF.**

Execute the resetdateoffset command to reset the difference between the time managed by the XSCF and the time managed by the physical partitions.

```
XSCF> resetdateoffset -p 0
```

7. **Execute the showpparmode command to check the setting of the detail level of the diagnosis messages and that of the PPAR DR mode.**

Execute the showpparmode command to check that the detail level (Message Level) of the diagnosis message is "normal" (standard) and that Next of the PPAR DR mode is set to "on" (enabled).

```
XSCF> showpparmode -p 0
Host-ID                :9007002b
Diagnostic Level        :min
Message Level           :normal
Alive Check             :on
Watchdog Reaction       :reset
Break Signal            :on
Autoboot (Guest Domain) :on
Elastic Mode            :off
IOreconfigure           :false
PPAR DR (Current)       :-
PPAR DR (Next)          :on
```

If the detail level of the diagnosis message is other than "normal", execute the setpparmode command to set it to "normal".

For details on the setpparmode command, see the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual*.

```
XSCF> setpparmode -p 0 -m message=normal
```

If PPAR DR mode is set to "off" (disabled), execute the setpparmode command to set it to "on".

```
XSCF> setpparmode -p 0 -m ppar_dr=on
```

8. **Execute the poweron command to power on the physical partitions.**

```
XSCF> poweron -p 0
```

9. **Execute the console command to connect the console to the physical partition.**

```
XSCF> console -p 0
```

10. **Install Oracle Solaris and Oracle VM Server for SPARC.**

Install, on the control domain, Oracle Solaris and Oracle VM Server for SPARC.

For details on the versions and conditions of Oracle Solaris required for physical partition dynamic reconfiguration, see [Table 1-3](#) and [Table 1-4](#).

For details on the installation, see the following documents, presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

- Oracle Solaris 10

- Oracle Solaris 11

Installing Oracle Solaris 11.x Systems

- Oracle VM Server for SPARC

"Installing and Enabling Software" in the Oracle VM Server for SPARC Administration Guide

11. Configure a logical domain.

This item describes how to configure the logical domains defined in "[Table A-1 Example of the 2BB Configuration With No Empty Hardware Resources.](#)"

a. Reduce the number of control domain resources.

For the factory-default configuration, all the CPU cores, memory, and the PCIe root complexes are assigned to the control domain (primary). To allow these resources to be assigned to other logical domains, first delete some of the resources from the control domain and configure the console service.

The following shows command execution examples.

Execute the `ldm start-reconf` command to switch to delayed reconfiguration mode.

```
# ldm start-reconf primary
```

```
Initiating a delayed reconfiguration operation on the primary domain.  
All configuration changes for other domains are disabled until the primary  
domain reboots, at which time the new configuration for the primary domain  
will also take effect.
```

Remove the root complex with the `ldm remove-io` command.

The following example partially describes the command for removing PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, PCIE7, PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, and PCIE15 according to the configuration example.

```
# ldm remove-io PCIE1 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
....
```

```
# ldm remove-io PCIE15 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

Reduce the number of CPU cores and the size of memory assigned to the control domain by specifying a size smaller than the original size with the `ldm set-core` and `ldm set-memory` commands.

The following gives an example of setting the number of CPU cores of the control domain to 32 and the memory size to 56 GB, according to the

configuration example.

```
# ldm set-core 32 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
# ldm set-memory 56G primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

Note - We recommend that you first configure the CPU cores with the `ldm set-core` command and then the memory with the `ldm set-memory` command. This facilitates the assignment of a continuous area of collective memory. This also makes it easy to easily satisfy the placement conditions for the CPU cores for which physical partition dynamic reconfiguration is possible and the memory placement conditions. For details on the CPU core and memory placement conditions, see ["Placement of CPU cores and memory" in "2.5.2 Considerations in System Operation for Dynamic Reconfiguration."](#)

Create a service of the virtual console terminal concentrator, called `vcc0`, with the `ldm add-vconscon` command and then start the daemon of the virtual network terminal server (`vntsd`) with the `svcadm` command. Configure a console connection to each logical domain via this `vcc0`.

```
# ldm add-vconscon port-range=5000-5200 vcc0 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
# svcadm enable vntsd
```

Save the configuration information and then restart Oracle Solaris.

The following example checks the configuration information saved with the `ldm list-spconfig` command, and then saves the configuration as name `ldm-set1` with the `ldm add-spconfig` command. Then, it again checks that the configuration has been saved with the `ldm list-spconfig` command and finally gives an example of restarting Oracle Solaris.

```
# ldm list-spconfig  
factory-default [current]  
# ldm add-spconfig ldm-set1  
# ldm list-spconfig  
factory-default  
ldm-set1 [current]  
# shutdown -i6 -g0 -y
```

b. Establish a redundant configuration for the system volume of the control

domain.

Establish a redundant configuration for the system volume of the control domain. This item describes an example of the commands for establishing a mirroring configuration with ZFS. For details, see the following documents presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

- For Oracle Solaris 10

"How to Create a Mirrored Root Pool (Post Installation)" in the *Oracle Solaris ZFS Administration Guide*

- For Oracle Solaris 11

"How to Configure a Mirrored Root Pool (SPARC or x86/VTOC)" in the *Oracle Solaris 11.x Administration: ZFS File Systems*

To use other redundant configuration software, see the manual for that software.

Execute the `zpool status` command to check the status of the root pool.

The following example indicates that `c2t50000393E802CCE2d0s0` has been assigned to the default root pool (`rpool`).

```
# zpool status rpool
pool: rpool
state: ONLINE
  scan: none requested
config:
      NAME                                STATE      READ  WRITE CKSUM
      rpool                               ONLINE      0     0     0
          c2t50000393E802CCE2d0s0        ONLINE      0     0     0
errors: No known data errors
```

Execute the `format` command to check which disks can be added.

The following example indicates that `c3t50000393A803B13Ed0` exists as the other disk.

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
   0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
      /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
      /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
   1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
      /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
      /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
```

Execute the `zpool attach` command to add the second disk to the `rpool` to provide a mirror configuration.

The following example adds `c3t50000393A803B13Ed0s0` with the `zpool attach`

command and then checks the status of the synchronization processing (resilver) with the zpool status command. By referring to status and action, you can determine that the synchronization processing is in progress. Regularly execute the zpool status command to check the status of the synchronization processing until the processing ends.

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered. The pool will
       continue to function in a degraded state.
action: Wait for the resilver to complete.
       Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Wed Jan 29 21:35:39 2014
      3.93G scanned out of 70.6G at 71.9M/s, 0h15m to go
      3.90G resilvered, 5.56% done
config:
      NAME                                STATE      READ WRITE CKSUM
      rpool                               DEGRADED   0      0      0
        mirror-0                          DEGRADED   0      0      0
          c2t50000393E802CCE2d0s0         ONLINE    0      0      0
          c3t50000393A803B13Ed0s0         DEGRADED   0      0      0 (resilvering)
```

Upon the completion of the synchronization processing, [state] is set to "ONLINE", as shown below.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h10m with 0 errors on Wed Jan 29 21:45:42 2014
config:
      NAME                                STATE      READ WRITE CKSUM
      rpool                               ONLINE     0      0      0
        mirror-0                          ONLINE     0      0      0
          c2t50000393E802CCE2d0s0         ONLINE     0      0      0
          c3t50000393A803B13Ed0s0         ONLINE     0      0      0
errors: No known data errors
```

c. Configure a root domain.

This item describes the procedure for configuring a root domain.

Execute the ldm add-domain command to add a logical domain named root-dom0.

```
# ldm add-domain root-dom0
```

Execute the ldm set-variable command to change the OpenBoot PROM environment variable "auto-boot?", which is designed to automatically boot Oracle Solaris, to "false" (disabled). By default, this setting is "true" (enabled). So, OpenBoot PROM tries to start Oracle Solaris automatically when Oracle

Solaris is not installed. Changing this setting to disabled facilitates the work to be performed before Oracle Solaris installation.

```
# ldm set-variable auto-boot\?=false root-dom0
```

First, assign the CPU cores with the `ldm set-core` command and then assign the memory with the `ldm set-memory` command.

The following example assigns 16 CPU cores with the `ldm set-core` command and 32 GB of memory with the `ldm set-memory` command, according to the configuration example.

```
# ldm set-core 16 root-dom0
# ldm set-memory 32G root-dom0
```

Note - We recommend that you first configure the CPU cores with the `ldm set-core` command and then the memory with the `ldm set-memory` command. This facilitates the assignment of a continuous area of collective memory. This also makes it easy to easily satisfy the placement conditions for the CPU cores for which physical partition dynamic reconfiguration is possible and the memory placement conditions. For details on the CPU core and memory placement conditions, see ["Placement of CPU cores and memory" in "2.5.2 Considerations in System Operation for Dynamic Reconfiguration."](#)

Execute the `ldm set-vconsole` command to assign the virtual console (vcons).

The following example executes the `ldm set-vconsole` command to assign port number 5000 of the service (vcc0) of the virtual console terminal concentrator in the control domain to the virtual console.

```
# ldm set-vconsole service=vcc0 port=5000 root-dom0
```

The following example executes the `ldm list-io -l` command to display the PCI assignment status.

The line where NAME begins with `/BB0` and the [TYPE] column has `"PCIE"` indicates the PCIe endpoint on the SPARC M10-4S0 (BB0). The line where the [DOMAIN] column is empty indicates an unassigned PCIe endpoint and the related root complex is displayed in the [BUS] column.

Therefore, you can quickly understand that PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, and PCIE7 are unassigned root complexes on BB0.

```
# ldm list-io -l
NAME                                     TYPE   BUS      DOMAIN   STATUS
----
(Omitted)
/BB0/CMUL/NET0                         PCIE   PCIE0    primary OCC
[pci@8000/pci@4/pci@0/pci@9]
    network@0
    network@0,1
```


/BB0/CMUL/SASHBA	PCIE	PCIE0	primary	OCC
[pci@8000/pci@4/pci@0/pci@0]				
scsi@0/iport@f/disk@w500000393e802cce2,0				
scsi@0/iport@f/disk@w500000393d8285226,0				
scsi@0/iport@f/smp@w500000e0e06d027f				
scsi@0/iport@f/enclosure@w500000e0e06d027d,0				
scsi@0/iport@v0				
/BB0/PCI0	PCIE	PCIE1		EMP
[pci@8100/pci@4/pci@0/pci@0]				
/BB0/PCI3	PCIE	PCIE2		EMP
[pci@8200/pci@4/pci@0/pci@0]				
/BB0/PCI4	PCIE	PCIE2		EMP
[pci@8200/pci@4/pci@0/pci@8]				
/BB0/PCI7	PCIE	PCIE3		EMP
[pci@8300/pci@4/pci@0/pci@0]				
/BB0/PCI8	PCIE	PCIE3		EMP
[pci@8300/pci@4/pci@0/pci@8]				
/BB0/CMUL/NET2	PCIE	PCIE4	primary	OCC
[pci@8400/pci@4/pci@0/pci@a]				
network@0				
network@0,1				
/BB0/PCI1	PCIE	PCIE5		EMP
[pci@8500/pci@4/pci@0/pci@8]				
/BB0/PCI2	PCIE	PCIE5		EMP
[pci@8500/pci@4/pci@0/pci@9]				
/BB0/PCI5	PCIE	PCIE6		EMP
[pci@8600/pci@4/pci@0/pci@9]				
/BB0/PCI6	PCIE	PCIE6		EMP
[pci@8600/pci@4/pci@0/pci@11]				
/BB0/PCI9	PCIE	PCIE7		EMP
[pci@8700/pci@4/pci@0/pci@9]				
/BB0/PCI10	PCIE	PCIE7		EMP
[pci@8700/pci@4/pci@0/pci@11]				
(Omitted)				

See the device path (string displayed as [pci@....]) displayed in the above result and "A.6 SPARC M10-4S Device Paths" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide* to determine which root complexes are to be assigned to the root domain.

In the configuration example, all the unassigned root complexes (PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, and PCIE7) on BB0 checked above are to be assigned. So, execute the `ldm add-io` command to assign them to root-dom0.

The following example shows command execution.

```
# ldm add-io PCIE1 root-dom0
# ldm add-io PCIE2 root-dom0
# ldm add-io PCIE3 root-dom0
# ldm add-io PCIE5 root-dom0
# ldm add-io PCIE6 root-dom0
# ldm add-io PCIE7 root-dom0
```

Place the root domain in the bound status with the `ldm bind-domain`

command, and then execute the ldm list-io command to check that the root complexes have been assigned.

The following example checks that root-dom0 is bound with the ldm bind-domain command to check with the ldm list-io command that the root complexes have been assigned.

The line where the [TYPE] column is "BUS" and the [DOMAIN] column is "root-dom0" indicates the root complex assigned to root-dom0. BUS in that line is the name of the assigned root complex.

In the following example, you can check that PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, and PCIE7 have been assigned to root-dom0.

```
# ldm bind-domain root-dom0
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8	primary	IOV

(Omitted)

According to step 11. c, configure the root domains (root-dom1 in the example) of the other SPARC M10-4Ss.

d. Install Oracle Solaris in the root domain.

This item describes the procedure for installing Oracle Solaris in the root domain.

Execute the ldm start-domain command to start root domain root-dom0.

```
# ldm start-domain root-dom0
LDom root-dom0 started
```

Execute the ldm set-variable command to change the OpenBoot PROM environment variable "auto-boot?", which is designed to automatically boot Oracle Solaris, to "true" (enabled). After Oracle Solaris is installed, execution of the ldm start-domain command also starts Oracle Solaris.

```
# ldm set-variable auto-boot?=true root-dom0
```

Execute the telnet command to connect to the console of the root domain.

The following example checks that the port number of root-dom0 is 5000 by executing the ldm list-domain command. It can also check that root-dom0 is

stopped in the OpenBoot PROM (OBP) status by connecting to localhost port number 5000 with the telnet command.

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART    64    56G     0.0%  7h 7m
root-dom0     active    -t----   5000    32    32G     0.0%  20s
root-dom1     bound     ------  5001    32    32G
# telnet localhost 5000
....
{0} ok
```

Install Oracle Solaris in the root domain.

For details on the versions and conditions of Oracle Solaris required for physical partition dynamic reconfiguration, see [Table 1-3](#) and [Table 1-4](#).

The following example executes the command to start Oracle Solaris 11 installation through the network.

```
{0} ok boot net:dhcp
....
```

For details on the installation, see the following documents, presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

- Oracle Solaris 10

Oracle Solaris 10 1/13 Installation Guide

- Oracle Solaris 11

Installing Oracle Solaris 11.x Systems

According to step 11. d, install the other root domains (root-dom1 in the example) in the same way.

e. Assign the virtual I/O service to the root domain.

This item describes an example of the procedure for setting the virtual I/O service to the root domain.

The example shows that the entire physical disk of the root domain is lent as the virtual disk service (vds).

This example shows that the Ethernet card port is assigned to the virtual switch network device (vsw).

For details on the virtual I/O service, see the *Oracle VM Server for SPARC Administration Guide* published by Oracle Corporation.

Log in to the root domain with the root privilege.

```
root-dom0 console login: root
Password:
....
```

To specify the back-end device (physical disk) to be assigned to the virtual

disk service (vds), execute the format command to display the disk and its disk path.

A string in the format of cXtXXXXXXXXXXXXXXXXdX indicates the disk, and a string beginning with "/pci@" indicates the device path.

```
root-dom0# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  0. c2t50000393A802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
    /pci@8100/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
  1. c3t50000393D8285226d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
    /pci@8200/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393d8285226,0
  2. c4t50000393A804B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
    /pci@3200/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
```

See the device path displayed in the above result and "A.6 SPARC M10-4S Device Paths" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*, and check the physical location of the disk. Then, determine the back-end disk (cXtXXXXXXXXXXXXXXXXdX) to be assigned to the virtual disk service.

The configuration example assigns two disks (c3t50000393D8285226d0 and c4t50000393A804B13Ed0) as back-ends of the virtual disks of the guest domains (guest0, guest1).

Next, to specify the network interface to be assigned to the virtual switch service (vsw), execute the dladm show-phys command to display the network interface name (LINK) associated with the location (LOC) of the PCIe slot.

From the displayed results, check the location of the network interface and determine the network interface to be assigned to the virtual switch service.

The configuration example assigns two network interfaces (net1 and net2) to the virtual switch associated with the virtual network of each guest domain.

```
root-dom0# dladm show-phys -L
LINK      DEVICE      LOC
net0      igb0          BB#0-PCI#0
net1      igb1          BB#0-PCI#1
net2      igb2          BB#0-PCI#2
net3      igb3          BB#0-PCI#3
...
```

Execute the ldm add-vdiskserver command to add the virtual disk service to the root domain.

The following example adds the virtual disk service (vds0) to the root domain (root-dom0) with the ldm add-vdiskserver command.

```
# ldm add-vdiskserver vds0 root-dom0
```

Execute the `ldm add-vdiskserverdevice` command to export the back-end of the virtual disk from the root domain.

The following example executes the `ldm add-vdiskserverdevice` command to assign two back-end disks to `vds0` to export them.

```
# ldm add-vdiskserverdevice /dev/dsk/c3t50000393D8285226d0s2 vol10@  
vds0  
# ldm add-vdiskserverdevice /dev/dsk/c4t50000393A804B13Ed0s2 vol11@  
vds0
```

Execute the `ldm add-vswitch` command to add the virtual switch to the root domain.

The following example adds `vsw0` and `vsw1` to add a virtual switch for each guest domain and assigns a different physical network interface (`net1`, `net2`) to each.

```
# ldm add-vswitch net-dev=net1 vsw0 root-dom0  
# ldm add-vswitch net-dev=net2 vsw1 root-dom0
```

Similarly, add the virtual I/O service to the root domain (`root-dom1`) to which the I/O of SPARC M10-4S on the BB1 side was assigned.

As an example, assign the following virtual disk service name and virtual switch names to `root-dom1`. For details on the step, see 11. e.

Virtual disk service name: `vds1` (assign `vol10` and `vol11` as two back-end volume names.)

Virtual switch name: `vsw10`, `vsw11`

f. Configure a guest domain.

This item describes the procedure for configuring a guest domain.

Execute the `ldm add-domain` command to add the logical domain named `guest0`.

```
# ldm add-domain guest0
```

Execute the `dm set-variable` command to change OpenBoot PROM environment variable "auto-boot?", which is designed to automatically boot Oracle Solaris, to "false" (disabled). By default, this setting is "true" (enabled). So, OpenBoot PROM tries to start Oracle Solaris automatically when Oracle Solaris is not installed. Changing this setting to disabled facilitates the work to be performed before Oracle Solaris installation.

```
# ldm set-variable auto-boot\?=false guest0
```

First, assign the CPU cores with the `ldm set-core` command and then assign

the memory with the `ldm set-memory` command.

The following example assigns 32 CPU cores with the `ldm set-core` command and 64 GB of memory with the `ldm set-memory` command, according to the configuration example.

```
# ldm set-core 32 root-dom0
# ldm set-memory 64G root-dom0
```

Note - We recommend that you first configure the CPU cores with the `ldm set-core` command and then the memory with the `ldm set-memory` command. This facilitates the assignment of a continuous area of collective memory. This also makes it easy to easily satisfy the placement conditions for the CPU cores for which physical partition dynamic reconfiguration is possible and the memory placement conditions. For details on the CPU core and memory placement conditions, see ["Placement of CPU cores and memory" in "2.5.2 Considerations in System Operation for Dynamic Reconfiguration."](#)

Execute the `ldm set-vconsole` command to assign the virtual console (vcons).

The following example executes the `ldm set-vconsole` command to assign port number 5100 of the service (vcc0) of the virtual console terminal concentrator in the control domain to the virtual console.

```
# ldm set-vconsole service=vcc0 port=5100 guest0
```

Execute the `ldm add-vdisk` command to assign the virtual disk (vdisk).

The following example assigns two virtual disks (vdisk0, vdisk10) according to the configuration example. The back-end of each virtual disk specifies the back-ends (vds0 vol0, vds1 vol10) added to two root domains (root-dom0, root-dom1).

```
# ldm add-vdisk vdisk0 vol0@vds0 guest0
# ldm add-vdisk vdisk10 vol10@vds1 guest0
```

Execute the `ldm add-vnet` command to assign the virtual network device (vnet).

The following example assigns two virtual network devices (vnet0, vnet10) according to the configuration example. The virtual switch connected to each virtual network device specifies virtual switches (vsw0, vsw10) added to two root domains (root-dom0, root-dom1).

```
# ldm add-vnet vnet0 vsw0 guest0
# ldm add-vnet vnet10 vsw10 guest0
```

Note - If multiple virtual disks or virtual network devices are assigned, record the value of the ID assigned to the virtual device, from the `ldm list -l` command execution result. To add a

virtual device again after being dynamically deleted, this ID is required so that the virtual device path remains unchanged.

g. **Install Oracle Solaris in the guest domain.**

This item describes the procedure for installing Oracle Solaris in the guest domain.

Execute the `ldm bind-domain` command to bind the guest domain and then start it by executing the `start-domain` command.

The following example shows the execution of the commands for starting `guest0`.

```
# ldm bind-domain guest0
# ldm start-domain guest0
LDom guest0 started
```

Execute the `ldm set-variable` command to change the OpenBoot PROM environment variable "auto-boot?", which is designed to automatically boot Oracle Solaris, to "true" (enabled). After Oracle Solaris is installed, execution of the `ldm start-domain` command also starts Oracle Solaris.

```
# ldm set-variable auto-boot\?=true guest0
```

Execute the `telnet` command to connect to the console of the guest domain.

The following example checks that the port number of the `guest0` console is "5100" by executing the `ldm list-domain` command. It can also check that `guest0` is stopped in the OpenBoot PROM status by connecting to localhost port number "5100" with the `telnet` command.

```
# ldm list-domain
NAME                STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  UPTIME
primary             active     -n-cv-   UART    64    56G     0.0%  8h 7m
guest0              active     -t----   5100    64    64G     0.0%  20s
root-dom0           active     -n--v-   5000    32    32G     0.0%  43s
root-dom1           active     -n--v-   5001    32    32G     0.0%  20s
guest1              inactive   ------ 64      64G
# telnet localhost 5100
....
{0} ok
```

For details on the versions and conditions of Oracle Solaris required for physical partition dynamic reconfiguration, see [Table 1-3](#) and [Table 1-4](#).

The following example executes the command to start Oracle Solaris 11 installation through the network.

```
{0} ok boot net:dhcp
....
```

For details on the installation, see the following documents, presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

- Oracle Solaris 10

Oracle Solaris 10 1/13 Installation Guide

- Oracle Solaris 11

Installing Oracle Solaris 11.x Systems

According to step 11. g, install the other guest domains (guest1 in the example) in the same way.

h. **Establish a redundant configuration for the virtual I/Os of a guest domain.**

The following describes an example of the procedure for establishing a redundant configuration for two virtual network interfaces (vnets) assigned to guest domain guest0, using IPMP. For details on the procedures for other redundant configurations, see the documentation about the software for the respective redundant configurations.

Log in to guest domain guest0.

In the example below, the `ldm list-domain` command is used to check the port number of the console of guest0, and the `telnet` command is used to connect to port number "5100."

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART     16    14G     0.0%  8h 7m
guest0        active    -n----   5100     32    32G     0.0%  20s
guest1        active    -n----   5101     32    32G     0.0%  19s
root-dom0     active    -n--v-   5000     24    24G     0.0%  43s
root-dom1     active    -n--v-   5001     24    24G     0.0%  20s
# telnet localhost 5100
....
guest0 console login: root
Password:
...
guest0#
```

Execute the `dladm` command to check that the virtual network devices are visible.

In the example below, it is possible to refer to virtual network devices as network interfaces `net0` and `net1`.

```
guest0# dladm show-phys
LINK          MEDIA          STATE    SPEED  DUPLEX    DEVICE
net0          Ethernet      up        0      unknown  vnet0
net1          Ethernet      up        0      unknown  vnet1
```

This section describes the procedure on Oracle Solaris 11. Execute the `ipadm show-if` command to check that `net0` and `net1` are not displayed.


```

guest0# ipadm show-if
IFNAME      CLASS      STATE      ACTIVE OVER
lo0         loopback ok         yes    --

```

Execute the `ipadm create-ip` command to create IP interfaces `net0` and `net1`, and then use the `ipadm show-if` command to check that they have been created normally.

```

guest0# ipadm create-ip net0
guest0# ipadm create-ip net1
guest0# ipadm show-if
IFNAME      CLASS      STATE      ACTIVE OVER
lo0         loopback ok         yes    --
net0        ip         down       no     --
net1        ip         down       no     --

```

Execute the `ipadm create-ipmp` command to create IPMP interface `ipmp0`, and then execute the `ipadm add-ipmp` command to add IP interfaces `net0` and `net1` to the IPMP group.

```

guest0# ipadm create-ipmp ipmp0
guest0# ipadm add-ipmp -i net0 -i net1 ipmp0

```

Execute the `ipadm create-addr` command to assign an IP address to IPMP interface `ipmp0`, and then use the `ipadm show-addr` command to check the setting. In the example below, a fixed IP address is assigned.

```

guest0# ipadm create-addr -T static -a local=xx.xx.xx.xx/24 ipmp0/v4
guest0# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
ipmp0/v4     static    ok         xx.xx.xx.xx/24
lo0/v6       static    ok         ::1/128

```

Execute the `ipadm set-ifprop` command to set a standby interface, and use the `ipmpstat -i` command to check the IPMP configuration.

```

guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS      LINK      PROBE      STATE
net1       no     ipmp0      is-----  up        disabled   ok
net0       yes    ipmp0      --mbM--    up        disabled   ok

```

Perform the same procedure for the other guest domains (guest1 in the example).

i. Check the memory resource allocation status.

This item describes the procedure for checking and adjusting the location of

the memory assigned to each logical domain.

To enable the SPARC M10-4S to be disconnected using the physical partition dynamic reconfiguration, the memory placement conditions described in ["Placement of CPU cores and memory" in "2.5.2 Considerations in System Operation for Dynamic Reconfiguration"](#) need to be met.

On the other hand, the configuration example shows a configuration in which no empty resources are allocated. These empty resources are used in advance as movement destinations for the memory block. Therefore, it is important to adjust the memory block placement in advance according to the procedure described in ["A.2.3 Example of the Active Replacement Procedure \(for Oracle VM Server for SPARC 3.1.x\)"](#) to make it easy to take memory placement into consideration.

The following describes the example procedure.

Execute the `ldm list-devices -a memory` command to check the status of the continuous region (memory block) of the memory assigned to each logical domain. Then, check that there are not a large number of small memory blocks.

As a guide, check that there is not a large number of memory blocks with a SIZE of about 256 MB to 512 GB.

The following example shows the execution of the `ldm list-devices -a memory` command.

```
# ldm list-devices -a memory
MEMORY
  PA          SIZE          BOUND
  0x700000000000 4G        root-dom0
  0x700100000000 4G        root-dom1
  0x700200000000 8G        root-dom0
  0x700400000000 4G        root-dom1
  0x700500000000 4G        root-dom0
  0x700600000000 8G        root-dom1
  0x720000000000 8G        root-dom0
  0x720200000000 8G        root-dom0
  0x720400000000 8G        root-dom1
  0x720600000000 8G        root-dom1
  0x740000000000 32G       guest0
  0x760000800000 1272M     _sys_
  0x760050000000 31488M   guest1
  0x780000000000 16G       guest0
  0x780400000000 16G       guest1
  0x7a0000000000 16G       guest0
  0x7a0400000000 16G       guest1
  0x7c0000000000 28G       primary
  0x7c0700000000 256M     guest1
  0x7c0710000000 3840M
  0x7e0000800000 1272M     _sys_
  0x7e0050000000 512M     _sys_
  0x7e0070000000 256M     _sys_
  0x7e0080000000 8G       primary
  0x7e0280000000 20G      primary
  0x7e0780000000 2G       guest1
```

In the above example, the memory blocks of root-dom0 and root-dom1 are broken up into relatively small blocks. So, reassign the memory blocks as described in step 11. j.

As shown below, if the memory blocks of each logical domain is not broken up, you need not perform reconfiguration. Go to step 11. k.

```
# ldm list-devices -a memory
MEMORY
      PA                               SIZE                               BOUND
0x7000000000000000                    32G                               root-dom0
0x7200000000000000                    32G                               root-dom1
0x7400000000000000                    32G                               guest0
0x7600008000000000                   1272M                               _sys_
0x7600500000000000                    2G                               primary
0x7600d00000000000                   29440M                              guest1
0x7800000000000000                    32G                               guest0
0x7a00000000000000                    32G                               guest1
0x7c00000000000000                    32G                               primary
0x7e00008000000000                   1272M                               _sys_
0x7e00500000000000                   512M                               _sys_
0x7e00700000000000                   256M                               _sys_
0x7e00800000000000                   3328M                              guest1
0x7e01500000000000                   4864M
0x7e02800000000000                   22G                               primary
```

j. Adjust the memory block placement of the logical domains.

As a result of the checks performed in step 11. i, if the memory block of a logical domain is broken up, reallocate the resources assigned to the logical domains. The following is an example of the procedure.

Stop each logical domain with the `ldm stop-domain` command and make it inactive with the `ldm unbind-domain` command.

Note - Suppose that there are dependencies between the physical I/O assignment (root complex, PCIe endpoint, SR-IOV Virtual Function) or that there are virtual I/O dependencies. Stop and unbind the guest domain to which virtual I/O was assigned, the I/O domain to which the PCIe endpoint and SR-IOV VF were assigned, and the root domain, in that order.

The following example shows the commands for making guest0, guest1, root-dom0, and root-dom1 inactive.

```
# ldm stop-domain guest0
LDom guest0 stopped
# ldm stop-domain guest1
LDom guest1 stopped
# ldm stop-domain root-dom0
LDom root-dom0 stopped
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain guest0
# ldm unbind-domain guest1
```

```
# ldm unbind-domain root-dom0
# ldm unbind-domain root-dom1
```

Using the ldm list-domain command, check that they are inactive.

The following example shows that guest0, guest1, root-dom0, and root-dom1 have been made inactive.

```
# ldm list-domain
NAME                STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary             active     -n-cv-   UART    64      56G      0.2%    5h 28m
guest0              inactive   -----          64      64G
guest1              inactive   -----          64      64G
root-dom0           inactive   -----          32      32G
root-dom1           inactive   -----          32      32G
```

Using the ldm start-reconf command, switch from the control domain to delay reconfiguration mode.

```
# ldm start-reconf primary
```

Initiating a delayed reconfiguration operation on the primary domain. All configuration changes for other domains are disabled until the primary domain reboots, at which time the new configuration for the primary domain will also take effect.

Again set the same values as the number of cores and memory size already assigned, in the order in which the ldm set-core and ldm set-memory commands are used. Then, restart Oracle Solaris of the control domain.

The following example again sets the number of CPU cores to 32 with the ldm set-core command and again sets the memory size to 56 GB with the ldm set-memory command to restart Oracle Solaris. This is done with the shutdown command.

```
# ldm set-core 32 primary
```

```
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
```

```
# ldm set-memory 56G primary
```

```
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
```

```
# shutdown -i6 -g0 -y
```

Bind and start each logical domain by executing the ldm bind-domain and ldm start-domain commands.

Bind and start are performed in the reverse order to that applied for a stop.

The following example shows that root-dom0, root-dom1, guest0, and guest1

are started (in that order) by executing the ldm bind-domain and ldm start-domain commands.

```
# ldm bind-domain root-dom0
# ldm bind-domain root-dom1
# ldm bind-domain guest0
# ldm bind-domain guest1
# ldm start-domain root-dom0
LDom root-dom0 started
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm start-domain guest0
LDom guest0 started
# ldm start-domain guest1
LDom guest1 started
```

Execute the ldm list-domain command to check that all the logical domains are active.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.4%	5h 55m
guest0	active	-n----	5100	64	64G	1.7%	12s
guest1	active	-n----	5101	64	64G	10%	9s
root-dom0	active	-n--v-	5000	32	32G	3.6%	15s
root-dom1	active	-n--v-	5001	32	32G	2.2%	11s

Execute the ldm list-devices command to check the memory assignment status.

```
# ldm list-devices -a memory
```

MEMORY	PA	SIZE	BOUND
	0x700000000000	32G	root-dom0
	0x720000000000	32G	root-dom1
	0x740000000000	32G	guest0
	0x760000800000	1272M	_sys_
	0x760050000000	31488M	guest1
	0x780000000000	32G	guest0
	0x7a0000000000	32G	guest1
	0x7c0000000000	28G	primary
	0x7c0700000000	4G	
	0x7e0000800000	1272M	_sys_
	0x7e0050000000	512M	_sys_
	0x7e0070000000	256M	_sys_
	0x7e0080000000	28G	primary
	0x7e0780000000	1280M	guest1
	0x7e07d0000000	768M	

12. Save the configured local domain configuration information in the XSCF.

Execute the ldm set-spcfg command to save the configured information.

The following example checks the saved configuration information and then

saves it with the same name as that of the existing configuration information.
Execute the `ldm list-spconfig` command to check the current configuration information.

```
# ldm list-spconfig
factory-default
ldm-set1 [next poweron]
```

Execute the `ldm remove-spconfig` command to delete the configuration information to be overwritten.

```
# ldm remove-spconfig ldm-set1
```

Execute the `ldm add-spconfig` command to re-save the configuration information.

```
# ldm add-spconfig ldm-set1
```

Execute the `ldm list-spconfig` command to check that the saved configuration information has become `[current]`.

```
# ldm list-spconfig
factory-default
ldm-set1 [current]
```

13. **Save the configured logical domain configuration information to an XML file.**

To guard against the configuration information saved to the XSCF being unusable, save the configuration information to an XML file. It is recommended that the XML file be saved to a different media.

The following describes the example procedure.

Execute the `ldm list-domain` command to check that all the logical domains are active.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.0%	6h 9m
guest0	active	-n----	5100	64	64G	0.0%	15m
guest1	active	-n----	5101	64	64G	0.0%	15m
root-dom0	active	-n--v-	5000	32	32G	0.0%	15m
root-dom1	active	-n--v-	5001	32	32G	0.0%	15m

Execute the `ldm list-constraints` command to save the configuration information to an XML file.

```
# ldm list-constraints -x > /ldm-set1.xml
```

A.2.3 Example of the Active Replacement Procedure (for Oracle VM Server for SPARC 3.1.x)

This section describes an example of the procedure for actively replacing the BB#01 SPARC M10-4S using PPAR DR for the 2BB configuration system described in "Figure A-2."

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.

You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

1. Log in to the master XSCF.

Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF.

If you have logged in to a standby XSCF, log out and then log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

2. Execute the `showhardconf` command to check that [Status] of the XSCF in the SPARC M10-4S to be replaced is "Normal."

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Locked;
+ System_Power:On; System_Phase:Cabinet Power On;
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP1236052K ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
+ Freq:3.000 GHz; Type:0x10;
+ Core:16; Strand:2;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
+ FRU-Part-Number: CA20393-B50X A2 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
```

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.

You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

3. **Execute the `showbbstatus` command to confirm that the XSCF in the SPARC M10-4S to be replaced is not the master XSCF.**

```
XSCF> showbbstatus
BB#00 (Master)
```

If the SPARC M10-4S to be replaced is the master XSCF, execute the `switchscf` command to switch the XSCF.

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

Note - Confirm that the XSCF has been switched and rebooted before you release the PSB.

4. **Execute the console command to connect to the console of the control domain and then log in to it.**

```
XSCF> console -p 0
```

5. **Check the operation status and resource usage status of the logical domain.**
 - a. **Execute the `ldm list-domain` command to check the operation status of the logical domain.**

To check the logical domain operation status, check the [STATE] and [FLAGS] combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

The following example shows that the control domain, two root domains, and two guest domains are operating.

Check whether all domains are in "active", which indicates that Oracle Solaris is in operating state, or "inactive" state. If there is a domain with OpenBoot PROM status or bound status, the dynamic reconfiguration of the physical partition may fail.


```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.0%	1h 33m
guest0	active	-n----	5100	64	64G	3.1%	2s
guest1	active	-n----	5101	64	64G	1.6%	18m
root-dom0	active	-n--v-	5000	32	32G	3.1%	17m
root-dom1	active	-n--v-	5001	32	32G	3.1%	17m

- b. **Execute the ldm list-devices command with the -a option specified to check the resource usage status.**

In the following example, the -a option is specified to display all resources bound to the logical domain and all resources that are not bound.

```
# ldm list-devices -a
```

CORE		
ID	%FREE	CPUSET
0	0	(0, 1)
4	0	(8, 9)
8	0	(16, 17)
(Omitted)		
944	0	(1888, 1889)
948	0	(1896, 1897)
952	0	(1904, 1905)
956	0	(1912, 1913)
VCPU		
PID	%FREE	PM
0	0	no
1	0	no
8	0	no
9	0	no
(Omitted)		
1904	0	no
1905	0	no
1912	0	no
1913	0	no
(Omitted)		

6. **Release the redundant configuration of the system volume and I/O devices in the control domain.**

To enable the release of BB#01, release the I/O devices of the SPARC M10-4S to be replaced and which are used in the control domain. For details on the procedure for canceling a redundant configuration, see the document for the software for that redundant configuration.

- a. **Cancel the redundant configuration of the system volume in the control domain.**

The following example describes how to cancel the ZFS mirroring function for the system volume in the control domain.

Execute the zpool status command in the control domain to check the mirroring configuration status.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                               ONLINE     0     0     0
        mirror-0
            c2t50000393E802CCE2d0s0    ONLINE     0     0     0
            c3t50000393A803B13Ed0s0    ONLINE     0     0     0
errors: No known data errors
```

Execute the `zpool detach` command to release the disk from the mirroring configuration.

```
# zpool detach rpool c3t50000393A803B13Ed0
```

Execute the `zpool status` command to confirm that the mirroring configuration has been canceled.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                               ONLINE     0     0     0
        mirror-0
            c2t50000393E802CCE2d0s0    ONLINE     0     0     0
errors: No known data errors
```

If you are using other devices in BB#01, remove the redundant configuration or stop using those devices. For details on how to cancel a redundant configuration or stop using the devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

b. Delete the I/O configuration of the control domain.

From among the physical I/O devices assigned to the control domain, delete the root complexes in BB#01 through delayed reconfiguration.

First, place the control domain in delayed reconfiguration mode.

```
# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
```

Execute the `ldm list-io` command to check the root complexes assigned to the primary.

The following example shows that the root complexes with BB1 devices are PCIE8 and PCIE12.

```
# ldm list-io | grep primary
PCIE0          BUS      PCIE0      primary  IOV
PCIE4          BUS      PCIE4      primary  IOV
PCIE8          BUS      PCIE8      primary  IOV
PCIE12         BUS      PCIE12     primary  IOV
/BB0/CMUL/NET0 PCIE      PCIE0      primary  OCC
/BB0/CMUL/SASHBA PCIE      PCIE0      primary  OCC
/BB0/CMUL/NET2 PCIE      PCIE4      primary  OCC
/BB1/CMUL/NET0 PCIE      PCIE8      primary  OCC
/BB1/CMUL/SASHBA PCIE      PCIE8      primary  OCC
/BB1/CMUL/NET2 PCIE      PCIE12     primary  OCC
```

Execute the `ldm remove-io` command to delete PCIE8 and PCIE12 from the primary, and then restart Oracle Solaris.

```
# ldm remove-io PCIE8 primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----

# ldm remove-io PCIE12 primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----

# shutdown -i6 -g0 -y
....
```

Once Oracle Solaris has started, execute the `ldm list-io` command to confirm that the root complexes in BB#01 have been deleted from the control domain.

```
# ldm list-io | grep primary
PCIE0          BUS      PCIE0      primary  IOV
PCIE4          BUS      PCIE4      primary  IOV
/BB0/CMUL/NET0 PCIE      PCIE0      primary  OCC
/BB0/CMUL/SASHBA PCIE      PCIE0      primary  OCC
/BB0/CMUL/NET2 PCIE      PCIE4      primary  OCC
```

c. Cancel the redundant configuration of the virtual I/O devices assigned to a guest domain.

To first shut down the root domain (root-dom1) to which the root complexes in BB#01 are assigned, log in to each guest domain, and then cancel the redundant configuration of the virtual I/O device from root-dom1.

For details on how to use the redundant configuration software, see the documentation about the software for that redundant configuration.

In the following example, a virtual network device (vnet1) is canceled from the IPMP configuration. For details on the command, see the manual for

Oracle Solaris.

Log in to the guest domain (guest0).

```
# ldm list-domain
NAME                STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary             active    -n-cv-   UART     64      56G       0.0%    4h 17m
guest0              active    -n----   5100     64      64G       0.0%    1h 13m
guest1              active    -n----   5101     64      64G       0.0%    1h 4m
root-dom0            active    -n--v-   5000     32      32G       0.0%    1h 47m
root-dom1            active    -n--v-   5001     32      32G       0.0%    1h 19m
# telnet localhost 5100
....
guest0#
```

Execute the `dladm show-phys` command to check the mapping between the virtual network interface (vnet1) and the network interface name (net1).

```
guest0# dladm show-phys
LINK                MEDIA                STATE    SPEED  DUPLEX    DEVICE
net0                 Ethernet             up        0      unknown  vnet0
net1                 Ethernet             up        0      unknown  vnet1
```

Execute the `ipmpstat -i` command to check the configuration information for the network interfaces configuring IPMP.

```
guest0# ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS    LINK    PROBE    STATE
net0       yes    ipmp0      -smbM--  up      disabled ok
net1       no     ipmp0      is----- up      disabled ok
```

Execute the `if_mpadm -d net1` command to release net1 from the IPMP group, and then execute the `ipmpstat -i` command to confirm that it has been released. The following example confirms that STATE is offline.

```
guest0# if_mpadm -d net1
guest0# ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS    LINK    PROBE    STATE
net0       yes    ipmp0      -smbM--  up      disabled ok
net1       no     ipmp0      -s---d-  up      disabled offline
```

Execute the `ipadm delete-ip` command to delete net1.

```
guest0# ipadm delete-ip net1
```

Similarly, perform the same release processing for the guest domain (guest1).

d. **Remove the virtual I/O devices assigned from the root domain to be stopped.**

Execute the `ldm remove-vdisk` and `ldm remove-vnet` commands to remove

the assigned virtual disk (vdisk) and virtual network device (vnet) from the root domain to be stopped according to the following step.

The following example shows the execution of the commands for removing the virtual disk (vdisk10) and virtual network device (vnet10) that use the virtual I/O service of the BB#01 root domain (root-dom1).

```
# ldm remove-vdisk vdisk10 guest0
# ldm remove-vnet vnet10 guest0
```

Perform the same deletion for the guest domain (guest1).

7. **Check the resource usage status of the I/O devices, and then cancel all the I/O devices in the SPARC M10-4S to be replaced.**

a. **Check the logical domain to which the root complexes in the SPARC M10-4S to be released are assigned.**

Execute the `ldm list-io` command to check the logical domain to which the root complexes in BB#01 are assigned.

The following example shows that only root-dom1 has PCIe endpoints starting with "/BB1/." You can see that the PCIe endpoint root complexes (BUS) PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, and PCIE15 are assigned to root-dom1.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV
....				
/BB1/CMUL/NET0	PCIE	PCIE8		UNK
/BB1/CMUL/SASHBA	PCIE	PCIE8		UNK
/BB1/PCI0	PCIE	PCIE9	root-dom1OCC	
/BB1/PCI3	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI4	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI7	PCIE	PCIE11	root-dom1OCC	
/BB1/PCI8	PCIE	PCIE11	root-dom1OCC	
/BB1/CMUL/NET2	PCIE	PCIE12		UNK
/BB1/PCI1	PCIE	PCIE13	root-dom1OCC	

/BB1/PCI2	PCIE	PCIE13	root-dom1OCC
/BB1/PCI5	PCIE	PCIE14	root-dom1OCC
/BB1/PCI6	PCIE	PCIE14	root-dom1OCC
/BB1/PCI9	PCIE	PCIE15	root-dom1OCC
/BB1/PCI10	PCIE	PCIE15	root-dom1OCC

- b. **Stop the root domain to which the root complexes in SPARC M10-4S to be released are assigned and then release the SPARC M10-4S.**

The following example executes the `ldm stop-domain` and `ldm unbind-domain` commands to release the root domain (root-dom1) and shows that the root domain is in the inactive state.

```
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain root-dom1
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.2%	4h 59m
guest0	active	-n----	5100	64	64G	0.0%	1h 55m
guest1	active	-n----	5101	64	64G	0.0%	1h 46m
root-dom0	active	-n--v-	5000	32	32G	0.0%	2h 29m
root-dom1	inactive	-----		32	32G		

- c. **Confirm that all of the I/O devices in the SPARC M10-4S to be replaced have been canceled.**

Execute the `ldm list-io` command to confirm that all the I/O devices have been released.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9		
PCIE10	BUS	PCIE10		
PCIE11	BUS	PCIE11		
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13		
PCIE14	BUS	PCIE14		
PCIE15	BUS	PCIE15		
(Omitted)				

8. **Manually reduce the number of CPU cores and the memory resources assigned to the logical domains.**

The following explains the procedure for manually reducing the number of CPU cores and the memory resources assigned to the logical domains, instead of specifying the `-m unbind=resource` option, to satisfy the conditions for "Placement of CPU cores and memory" in "2.5.2 Considerations in System Operation for Dynamic Reconfiguration."

If, for example, the `deleteboard` command fails due to an error, it may be possible in some cases to execute the `deleteboard` command by following this procedure.

When specifying the `-m unbind=resource` option of the `deleteboard` command, go to step 9.

Note - In versions of Oracle VM Server for SPARC earlier than Oracle VM Server for SPARC 3.2, the `-m unbind=resource` option of the `deleteboard -c disconnect` command is not supported. Perform this procedure and then release the SPARC M10-4S with the `deleteboard` command without specifying `-m unbind=resource`.

a. Check the number of CPU cores and then delete them.

The number of CPU cores that can be used will be reduced as a result of releasing the SPARC M10-4S, so first apply the following procedure to reduce the number of CPU cores assigned to the logical domains beforehand.

i. Check the number of CPU cores after SPARC M10-4S has been released.

From the XSCF, execute the `showpparinfo` command to check the number of CPU cores, excluding those of SPARC M10-4S to be released.

In the following example, if a SPARC M10-4S with a physical system board (PSB) number of 01-0 is to be released, the sum of the CPU cores with another PSB number of 00-0 must be calculated. Thus, $16 + 16 + 16 + 16 = 64$ cores.

```
XSCF> showpparinfo -p 0
PPAR#00 Information:
-----
CPU(s)                :      8
CPU Cores              :     128
CPU Threads            :     256
Memory size (GB)       :     256
CoD Assigned (Cores)   :     256
CPU(s) :
-----
PID  PSB   CPU#  Cores  Threads
00   00-0   0      16      32
00   00-0   1      16      32
00   00-0   2      16      32
00   00-0   3      16      32
00   01-0   0      16      32
00   01-0   1      16      32
00   01-0   2      16      32
00   01-0   3      16      32
(Omitted)
```

ii. **Check the total number of the CPU cores that are assigned to each logical domain.**

Execute `ldm list-devices -a core`. The number of rows with a value other than 100 in the %FREE column is the total number of CPU cores assigned to the logical domain.

In the following example, the `ldm list-devices -a core` command is executed, and a check is performed using the `-p` option. As a result, it can be seen that 112 cores are bound to the entire logical domains.

```
# ldm list-devices -a core
CORE
  ID      %FREE  CPUSSET
  0        0    (0, 1)
  4        0    (8, 9)
  8        0   (16, 17)
 12        0   (24, 25)
(Omitted)
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
112
```

iii. **Calculate the core shortfall that results from the release of the SPARC M10-4S.**

Using the formula below, calculate the CPU core shortfall that will result after releasing SPARC M10-4S.

CPU core shortfall = Number of cores used in logical domain (step ii) -
Number of physical cores after release (step i)

For the example in steps i and ii, it is found that the shortfall will be 112 cores (in use) - 64 cores (remaining) = 48 cores.

iv. **Consider which logical domains to delete if a CPU core shortfall occurs.**

If it is found as a result of step iii that a CPU core shortfall will occur, it is necessary to delete the CPU cores from the logical domains.

Execute the `ldm list-domain` command to check the number of CPU cores assigned to each of the logical domains in an active or bound state, and the check the logical domains from which the CPU cores will be deleted.

In the following example, it is found that 32 cores (64vcpu) are assigned to primary, 32 cores (64vcpu) to guest0, 32 cores (64vcpu) to guest1, 16 cores (32vcpu) to root-dom0, and 16 cores (32vcpu) to root-dom1. In the example, 48 cores need to be deleted. Thus, 16 cores will be deleted from each of primary, guest0, and guest1.

```
# ldm list-domain
NAME          STATE    FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
primary       active  -n-cv-  UART   64    56G     0.1%  18h 17m
guest0        active  -n----  5100   64    64G     0.0%  15h 13m
guest1        active  -n----  5101   64    64G     0.0%  15h  4m
root-dom0     active  -n--v-  5000   32    32G     0.0%  15h 47m
root-dom1     inactive  -----  32     32    32G
```


v. **Execute the ldm remove-core command to delete the CPU cores from the target logical domains.**

In the following example, 16 cores are deleted from each of primary, guest0, and guest1, and a check is made to determine if they have actually been deleted.

```
# ldm remove-core 16 primary
# ldm remove-core 16 guest0
# ldm remove-core 16 guest1
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active    -n-cv-   UART    32      56G       0.0%    18h 19m
guest0        active    -n----   5100    32      64G       0.0%    15h 15m
guest1        active    -n----   5101    32      64G       0.0%    15h 5m
root-dom0     active    -n--v-   5000    32      32G       0.0%    15h 49m
root-dom1     inactive  -----   5001    32      32G
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
64
```

b. **Check and delete memory resources.**

The memory area that can be used will be reduced as a result of releasing the SPARC M10-4S, so use the procedure below to delete the memory resources assigned to the logical domains.

i. **Check the use status of the continuous region of the memory (memory block).**

Execute the prtdiag command and the ldm list-devices -a memory command to check the memory block assigned to each logical domain, and check the SPARC M10-4Ss to which unassigned memory blocks are assigned.

First, execute the prtdiag command to check the correspondence between the physical addresses in memory and the SPARC M10-4Ss.

```
# prtdiag
(Omitted)
===== Physical Memory Configuration =====
Segment Table:
-----
Base          Segment  Interleave  Bank    Contains
Address       Size      Factor      Size    Modules
-----
0x7e0000000000 32 GB    4           8 GB    /BB0/CMUL/CMP0/MEM00A
(Omitted)
0x7c0000000000 32 GB    4           8 GB    /BB0/CMUL/CMP1/MEM10A
(Omitted)
0x7a0000000000 32 GB    4           8 GB    /BB0/CMUU/CMP0/MEM00A
(Omitted)
0x780000000000 32 GB    4           8 GB    /BB0/CMUU/CMP1/MEM10A
(Omitted)
0x760000000000 32 GB    4           8 GB    /BB1/CMUL/CMP0/MEM00A
(Omitted)
0x740000000000 32 GB    4           8 GB    /BB1/CMUL/CMP1/MEM10A
```

(Omitted)				
0x720000000000	32 GB	4	8 GB	/BB1/CMUU/CMP0/MEM00A
(Omitted)				
0x700000000000	32 GB	4	8 GB	/BB1/CMUU/CMP1/MEM10A
(Omitted)				

The result of this example is rearranged in ascending order of physical addresses in memory. The following table lists the correspondence between the physical addresses and the SPARC M10-4S.

Table A-2 Example of Correspondence Between Physical Addresses and the SPARC M10-4S

Base Address (Physical Address)	Building Block Configuration of SPARC M10-4S
0x700000000000 and after	BB1
0x720000000000 and after	BB1
0x740000000000 and after	BB1
0x760000000000 and after	BB1
0x780000000000 and after	BB0
0x7a0000000000 and after	BB0
0x7c0000000000 and after	BB0
0x7e0000000000 and after	BB0

Then, execute the `ldm list-devices -a memory` command to check the continuous areas (called memory blocks in the remainder of this document) of memory assigned to the logical domains and any unassigned memory blocks.

In the following example, the `ldm list-devices -a memory` command is executed.

The meaning of each parameter is as follows.

- PA: Start physical address of the memory block
- SIZE: Size of the memory block
- BOUND: Name of the logical domain to which the memory block is assigned A blank is an unassigned area, and `_sys_` is a control area that is not assigned to a logical domain.

# ldm list-devices -a memory		
MEMORY		
PA	SIZE	BOUND
0x700000000000	32G	root-dom0
0x720000000000	32G	
0x740000000000	32G	guest0
0x760000800000	1272M	_sys_
0x760050000000	31488M	guest1
0x780000000000	32G	guest0

0x7a0000000000	32G	guest1
0x7c0000000000	28G	primary
0x7c0700000000	4G	
0x7e0000800000	1272M	_sys_
0x7e0050000000	512M	_sys_
0x7e0070000000	256M	_sys_
0x7e0080000000	28G	primary
0x7e0780000000	1280M	guest1
0x7e07d0000000	768M	

By combining the results shown above with the physical positions checked with the prtdiag command, it can be seen that the memory block usage status is as below.

Table A-3 Example of Memory Block Use Statuses

SPARC M10-4S	Physical Address	Size	Logical Domain
BB1 (target for replacement)	0x700000000000	32 GB	root-dom0
	0x720000000000	32 GB	Unassigned
	0x740000000000	32 GB	guest0
	0x760050000000	31,488 MB	guest1
BB0	0x780000000000	32 GB	guest0
	0x7a0000000000	32 GB	guest1
	0x7c0000000000	28 GB	primary
	0x7c0700000000	4 GB	Unassigned
	0x7e0080000000	28 GB	primary
	0x7e0780000000	1,280 MB	guest1
	0x7e07d0000000	768 MB	Unassigned

ii. Check the size and quantity of the source memory blocks.

While referring to the check results of the memory block use status, check the memory block (hereafter called "source memory block") assigned to the SPARC M10-4S to be replaced.

For "Table A-3 Example of Memory Block Use Statuses," memory blocks of 32 GB x 2 and 31,488 MB x 1 are being used in BB1 as the blocks assigned to the logical domain.

iii. Determine the logical domains from which to delete memory, as well as the amounts.

Then, check the locations of the memory blocks assigned to each logical domain, and make sure that the source memory blocks can be moved to unassigned memory blocks (called "destination memory blocks" in the remainder of this document) of the SPARC M10-4Ss that will not be released, by deleting the memory in memory block units and by reducing the memory block size.

As a result, make a final decision as to how much memory to delete and

from which logical domains to delete it.

The following methods are supported.

- Reduce the number of memory blocks that need to be moved by deleting, as a whole, the source memory blocks.
- Increase the number of possible destinations by deleting, as a whole, the memory blocks assigned to the logical domains on the SPARC M10-4Ss that are not to be released.
- Reduce the size of the source memory blocks so that they will fit into the available free area at the destinations.
- Increase the number of free memory blocks at the destinations to make the movement possible, by reducing the size of the source memory blocks and by reducing the size of the memory blocks of the destinations that are in use.

Note - After the size is reduced, the free memory blocks will not be continuous (they will be fragmented). Even if multiple small memory blocks are deleted to increase the number of free areas, the free areas will not be continuous areas. If the continuous areas of the source memory blocks are large, movement will be impossible. In such a case, delete the source memory blocks to adjust their size.

Note - When considering this, the possibility of continuous areas being fragmented after deletion can be reduced by selecting deletable ones that have the same sizes as the existing memory blocks whenever possible. This increases the possibility of the success of the movement of memory blocks.

Note - If too much memory is deleted, this will place a burden on the memory in the logical domains, possibly causing problems such as Oracle Solaris hanging. It is necessary to be careful not to delete too much by using the vmstat command and checking the "free" size as a rough guide.

Investigate the deletion plan based on "[Table A-3 Example of Memory Block Use Statuses](#)."

As a result of this consideration, make a plan to delete 4 GB from root-dom0, 32 GB from guest0, 31,488 MB from guest1, and 28 GB from the primary as shown in "[Table A-4 Memory Block Deletion Plan](#)" as an example.

Table A-4 Memory Block Deletion Plan

SPARC M10-4S	Size	Logical Domain	Deletion Plan
BB1 (target for replacement)	32 GB	root-dom0	Reduce this area by 4 GB to 28 GB, delete 28 GB from the primary on BB0, and then implement the movement.
	32 GB	Unassigned	-
	32 GB	guest0	Delete this because there is a 32-GB guest0 memory on BB0.

Table A-4 Memory Block Deletion Plan (*continued*)

SPARC M10-4S	Size	Logical Domain	Deletion Plan
BB0	31,488 MB	guest1	Delete this because there is a 32-GB guest1 memory on BB0.
	32 GB	guest0	Leave it.
	32 GB	guest1	Leave it.
	28 GB	primary	Leave it.
	4 GB	Unassigned	-
	28 GB	primary	Delete it to move 28 GB of root-dom0.
	1,280 MB	guest1	Leave it.
	768 MB	Unassigned	-

iv. Manually delete memory from the logical domains.

Delete memory from the logical domains, using the `ldm remove-memory` command, according to the memory deletion plan devised in step iii.

The following example indicates the execution of the command for deleting memory according to "Table A-4 Memory Block Deletion Plan."

```
# ldm remove-memory 4G root-dom0
# ldm remove-memory 32G guest0
# ldm remove-memory 31488M guest1
# ldm remove-memory 28G primary
```

v. Check the states of the deleted memory blocks.

Execute the `ldm list-devices -a memory` command to check if the layout is such that movement is possible, by referring to the results of the deletion. If there are any memory blocks that cannot be moved, consider which additional memory blocks to delete based on the results, and then delete them.

In the following example, it is easier to check if movement is possible by comparing, side by side, the large sizes assigned to BB1 with the large sizes in the free areas on BB0.

```
# ldm list-devices -a memory
MEMORY
  PA          SIZE          BOUND
(BB1)
  0x700000000000 256M      root-dom0 →Can be moved by dividing
                                     a 4-GB destination
  0x700010000000 4G
  0x700110000000 28416M    root-dom0 →Can be moved to
                                     0x780000000000 (28GB)
```

0x720000000000	32G		
0x740000000000	256M	guest0	→Can be moved by dividing a 4-GB destination
0x740010000000	4G		
0x740110000000	28416M	guest0	→Must be deleted again because there is no destination(*)
0x760000800000	1272M	_sys_	
0x760050000000	256M	guest1	→Can be moved by dividing a 4-GB area
0x760060000000	1792M		
0x7600d0000000	29440M	guest1	→Can be moved to 0x7a0000000000 (29GB)
(BB0)			
0x780000000000	28G		←Can be moved from rootdom0 (0x700110000000)
0x780700000000	4G	guest0	
0x7a0000000000	29G		←Can be moved from guest1 (0x7600d0000000)
0x7a0740000000	3G	guest1	
0x7c0000000000	256M	primary	
0x7c0010000000	4G		←Can be moved from 256 MB of root-dom0, guest0, or guest1
0x7c0110000000	24320M	primary	
0x7c0700000000	4G		
0x7e0000800000	1272M	_sys_	
0x7e0050000000	512M	_sys_	
0x7e0070000000	256M	_sys_	
0x7e0080000000	24G		←Not sufficient for guest0 (0x740110000000) to move(*)
0x7e0680000000	4G	primary	
0x7e0780000000	1280M	guest1	
0x7e07d0000000	768M		

In the above example, the destination marked with (*) has only 24 GB (24,576 MB) of free space, so delete 3,840 MB from the 28416-MB area (guest0) on the source (BB1), and then repeat the check. In the following example, it can be seen that all of the memory blocks can now be moved.

# ldm remove-memory 3840M guest0			
# ldm list-devices -a memory			
MEMORY			
PA	SIZE	BOUND	
(BB1)			
0x700000000000	256M	root-dom0	
0x700010000000	4G		
0x700110000000	28416M	root-dom0	
0x720000000000	32G		
0x740000000000	256M	guest0	
0x740010000000	7936M		
0x740200000000	24G	guest0	→Can be moved to

			0x7e0080000000 (24G)
0x760000800000	1272M	_sys_	
0x760050000000	256M	guest1	
0x760060000000	1792M		
0x7600d0000000	29440M	guest1	
(BB0)			
0x780000000000	28G		
0x780700000000	4G	guest0	
0x7a0000000000	29G		
0x7a0740000000	3G	guest1	
0x7c0000000000	256M	primary	
0x7c0010000000	4G		
0x7c0110000000	24320M	primary	
0x7c0700000000	4G		
0x7e0000800000	1272M	_sys_	
0x7e0050000000	512M	_sys_	
0x7e0070000000	256M	_sys_	
0x7e0080000000	24G		←Can be moved from guest0 (0x740200000000)
0x7e0680000000	4G	primary	
0x7e0780000000	1280M	guest1	
0x7e07d0000000	768M		

9. **Release the system board (PSB<BB>) that corresponds to the SPARC M10-4S from the physical partition.**

- a. **Execute the deleteboard -c disconnect command to release the PSB from the physical partition.**

```
XSCF> deleteboard -c disconnect 01-0
PSB#01-0 will be unconfigured from PPAR immediately. Continue?
[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.
```

- b. **Execute the showresult command to check the exit status of the deleteboard command that was just executed.**

An end value of 0 indicates the normal termination of the deleteboard command.

If the end value is other than 0 or if an error message is displayed upon executing the deleteboard command, it indicates abnormal termination of the deleteboard command. By referring to "C.1.2 deleteboard" based on the error message, identify the error and then take corrective action.

```
XSCF> showresult
0
```

c. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the SPARC M10-4S to be replaced is in the "Assigned" state and that the [Pwr], [Conn], and [Conf] columns all show "n."

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)         Assigned    y    y    y    Passed Normal
01-0  00(01)         Assigned    n    n    n    Passed Normal
```

10. **Execute the replacefru command to replace the SPARC M10-4S.**

```
XSCF> replacefru
```

Note - For details on the replacement of SPARC M10-4Ss by using the replacefru command, see "5.8 Releasing a SPARC M10-4/M10-4S FRU from the System with the replacefru Command" and "6.2 Incorporating a SPARC M10-4/M10-4S FRU into the System with the replacefru Command" in the *Fujitsu M10-4/Fujitsu M10-4S/SPARC M10-4/SPARC M10-4S Service Manual*.

11. **Incorporate the system board (PSB<BB>) of the replaced SPARC M10-4 into the physical partition.**

a. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the SPARC M10-4S to be replaced is in the Assigned state and that the [Pwr], [Conn], and [Conf] columns all show "n."

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)         Assigned    y    y    y    Passed Normal
01-0  00(01)         Assigned    n    n    n    Passed Normal
```

b. **Execute the addboard -c configure command to incorporate the PSB into the physical partition.**

To recover the original logical domain configuration, execute the addboard -c configure command with the -m bind=resource option specified.

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0.....30.....60.....90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
```



```

810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.

```

Note - If an error message appears during execution of the `addboard` command, see "[C.1.1 addboard](#)," and then identify the error and take corrective action.

- c. **Execute the `showresult` command to check the exit status of the `addboard` command that was just executed.**

An end value of 0 indicates the normal termination of the `addboard` command. If the end value is other than 0 or if an error message is displayed upon executing the `addboard` command, it indicates abnormal termination of the `addboard` command. By referring to "[C.1.1 addboard](#)" based on the error message, identify the error and then take corrective action.

```

XSCF> showresult
0

```

- d. **Execute the `showboards` command to check the PSB status.**

Confirm that both of the [Conn] and [Conf] columns show "y" after the PSB in the replaced SPARC M10-4S has been successfully incorporated.

```

XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
----  -
00-0  00(00)         Assigned    y    y    y    Passed  Normal
01-0  00(01)         Assigned    y    y    y    Passed  Normal

```

12. Check the logical domain operation status.

- a. **Execute the console command to connect to the console of the control domain and then log in to it.**

```

XSCF> console -p 0

```

- b. **Execute the `ldm list-domain` command to confirm that the logical domain operation status has not changed after the addition of the PSB.**

To check the logical domain operation status, check the [STATE] and [FLAGS] combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	28G	64%	2h 54m
guest0	active	-n----	5100	32	28928M	42%	2h 54m
guest1	active	-n----	5101	32	34048M	11%	2h 54m
root-dom0	active	-n--v-	5000	32	28G	10%	2h 54m
root-dom1	inactive	-----		32	32G		

13. Restore the deleted resources.

If the resources assigned to the logical domains have been deleted with the deleteboard command, add the resources with the ldm add-core and ldm add-memory commands to restore them.

In the following example, deleted CPU cores and memory are added to restore the resources before the replacement of the SPARC M10-4S.

```
# ldm add-core 16 primary
# ldm add-core 16 guest0
# ldm add-core 16 guest1
# ldm add-memory 28G primary
# ldm add-memory 4G root-dom0
# ldm add-memory 36608M guest0
# ldm add-memory 31488M guest1
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.2%	4h 59m
guest0	active	-n----	5100	64	64G	0.0%	1h 55m
guest1	active	-n----	5101	64	64G	0.0%	1h 46m
root-dom0	active	-n--v-	5000	32	32G	0.0%	2h 29m
root-dom1	inactive	-----		32	32G		

14. Restart the use of the I/O devices.

a. Reassign root complexes.

Execute the ldm bind-domain and ldm start-domain commands to start the root domain in the unbind state to which root complexes in the replaced SPARC M10-4S were assigned.

The following example starts the root domain (root-dom1) in the unbind state, and confirms that it has started.

```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.2%	3h 8m
guest0	active	-n----	5100	64	64G	0.0%	3h 8m
guest1	active	-n----	5101	64	64G	0.0%	3h 8m
root-dom0	active	-n--v-	5000	32	32G	0.0%	3h 8m
root-dom1	active	-n--v-	5001	32	32G	7.3%	8s

Execute the ldm list-io command to confirm that the physical I/O devices are

assigned to the root domain that has just started.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV
....				
/BB1/CMUL/NET0	PCIE	PCIE8		UNK
/BB1/CMUL/SASHBA	PCIE	PCIE8		UNK
/BB1/PCI0	PCIE	PCIE9	root-dom1OCC	
/BB1/PCI3	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI4	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI7	PCIE	PCIE11	root-dom1OCC	
/BB1/PCI8	PCIE	PCIE11	root-dom1OCC	
/BB1/CMUL/NET2	PCIE	PCIE12		UNK
/BB1/PCI1	PCIE	PCIE13	root-dom1OCC	
/BB1/PCI2	PCIE	PCIE13	root-dom1OCC	
/BB1/PCI5	PCIE	PCIE14	root-dom1OCC	
/BB1/PCI6	PCIE	PCIE14	root-dom1OCC	
/BB1/PCI9	PCIE	PCIE15	root-dom1OCC	
/BB1/PCI10	PCIE	PCIE15	root-dom1OCC	

b. Add the virtual I/O device from the root domain to the guest domain.

Execute the `ldm add-vdisk` and `ldm add-vnet` commands to add, to each guest domain, the virtual disk (vdisk) and virtual network device (vnet) supported for the virtual I/O service of the started root domain.

The following example indicates the execution of the command for adding the virtual disk (vdisk10) and virtual network device (vnet10) that use the virtual I/O service of BB#01 root domain (root-dom1).

Perform the same addition for the guest domain (guest1).

```
# ldm add-vdisk id=1 vdisk10 vol10@vds1 guest0
# ldm add-vnet id=1 vnet10 vsw10 guest0
```

Note - To add the virtual I/O device again, it is necessary to specify the ID assigned

beforehand. You can check the ID from the result of execution of the `ldm list -l` command in the status used before the virtual I/O device is deleted.

c. **Incorporate the virtual I/O devices assigned to a guest domain into the redundant configuration.**

Once the root domain (root-dom1) to which root complexes in BB#1 have been assigned has started, the virtual I/O device services corresponding to each guest domain are also started.

Log in to each guest domain, and then incorporate the virtual I/O devices from root-dom1 that were previously canceled into the redundant configuration. For details on how to use the redundant configuration software, see the documentation for the software for that redundant configuration.

The following describes an example of incorporating a virtual network device (vnet1) into the IPMP configuration. For details on the commands, see the manual for Oracle Solaris.

First, log in to the guest domain (guest0).

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART    64     56G     0.0%  4h 17m
guest0        active    -n----   5100    64     64G     0.0%  1h 13m
guest1        active    -n----   5101    64     64G     0.0%  1h  4m
root-dom0     active    -n----   5000    32     32G     0.0%  1h 47m
root-dom1     active    -n----   5001    32     32G     0.0%  1h 19m
# telnet localhost 5100
....
guest0#
```

Execute the `dladm show-phys` command to check the mapping between the virtual network interface (vnet1) and the network interface name (net1).

```
guest0# dladm show-phys
LINK          MEDIA          STATE    SPEED  DUPLEX    DEVICE
net0          Ethernet      up        0      unknown  vnet0
net1          Ethernet      up        0      unknown  vnet1
```

Execute the `ipadm create-ip` command, the `ipadm set-ifprop` command, and the `ipadm add-ipmp` command to register net1 as a standby device of ipmp0.

```
guest0# ipadm create-ip net1
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipadm add-ipmp -i net1 ipmp0
```

Execute the `ipmpstat -i` command to confirm that STATE of the network interface configuring IPMP indicates ok.

Perform the same step for the other guest domain (guest1).

```

guest0# ipmpstat -i
INTERFACE ACTIVE GROUP FLAGS LINK PROBE STATE
net0 yes ipmp0 -smbM-- up disabled ok
net1 no ipmp0 -s---d- up disabled ok

```

15. **Restore the system volume and I/O devices on the control domain to a redundant configuration.**

a. **Add the root complex configuration for the control domain.**

By means of delayed configuration, add the root complexes in BB#01 that were previously removed from the control domain.

First, place the control domain in delayed reconfiguration mode.

```

# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.

```

Execute the ldm list-io command to check the unassigned root complexes.

The following example shows that root complexes with BB1 devices, PCIE8 and PCIE12 are not assigned.

```

# ldm list-io

```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
(Omitted)				

Execute the ldm add-io command to add PCIE8 and PCIE12 to primary, and then restart Oracle Solaris.

```

# ldm add-io PCIE8 primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----

```

```
# ldm add-io PCIE12 primary
```

```
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
```

```
# shutdown -i6 -g0 -y
....
```

Once Oracle Solaris has started, execute the `ldm list-io` command to confirm that the root complexes in BB#01 have been added to the control domain.

```
# ldm list-io | grep primary
```

PCIE0	BUS	PCIE0	primary	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE8	BUS	PCIE8	primary	IOV
PCIE12	BUS	PCIE12	primary	IOV
/BB0/CMUL/NET0	PCIE	PCIE0	primary	OCC
/BB0/CMUL/SASHBA	PCIE	PCIE0	primary	OCC
/BB0/CMUL/NET2	PCIE	PCIE4	primary	OCC

b. Place the system volume in the control domain in a redundant configuration.

Execute the `zpool status` command in the control domain to check the mirroring configuration status.

The following example describes how to configure the ZFS mirroring function for the system volume in the control domain.

```
# zpool status rpool
```

```
pool: rpool
state: ONLINE
scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                                ONLINE     0     0     0
        c2t50000393E802CCE2d0s0        ONLINE     0     0     0
errors: No known data errors
```

Execute the `zpool attach` command to incorporate the disks into a mirroring configuration.

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

Execute the `zpool status` command, and then confirm that the mirroring configuration has been established.

Use the `zpool status` command to confirm whether synchronization processing (resilver) is completed.

The following shows an example of the display during synchronization processing.

```
# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Mon Jan 27 15:55:47 2014
      21.1G scanned out of 70.6G at 120M/s, 0h7m to go
      21.0G resilvered, 29.84% done
config:
      NAME                                STATE      READ WRITE CKSUM
      rpool                                DEGRADED   0     0     0
        mirror-0                          DEGRADED   0     0     0
          c2t50000393E802CCE2d0s0        ONLINE    0     0     0
          c3t50000393A803B13Ed0s0        DEGRADED   0     0     0 (resilvering)
errors: No known data errors
```

Once synchronization processing is complete, the displayed screen will be as follows:

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
      NAME                                STATE      READ WRITE CKSUM
      rpool                                ONLINE    0     0     0
        mirror-0                          ONLINE    0     0     0
          c2t50000393E802CCE2d0s0        ONLINE    0     0     0
          c3t50000393A803B13Ed0s0        ONLINE    0     0     0
errors: No known data errors
```

If you are using other devices in BB#01, establish a redundant configuration or resume the use of the devices. For details on how to establish a redundant configuration or resume the use of devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

A.2.4 Example of the Active Replacement Procedure (When Dynamic Assignment of the PCIe Bus is Available)

This section describes an example of the procedure for actively replacing BB#01 using PPAR DR for the 2BB configuration system described in "[Figure A-2 Configuration Example in Which Operations Can Continue in the 2BB Configuration](#)." The example is for an environment where the PCIe bus is assignable dynamically (XCP 2240 or later with Oracle VM Server for SPARC 3.2 or later and the root domain with Oracle Solaris 11.2 SRU11.2.8.4.0 or later).

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.
You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

1. **Log in to the master XSCF.**

Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF.

If you have logged in to a standby XSCF, log out and then log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

2. **Execute the `showhardconf` command to check that [Status] of the XSCF in the SPARC M10-4S to be replaced is "Normal."**

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Locked;
+ System_Power:On; System_Phase:Cabinet Power On;
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP1236052K ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
+ Freq:3.000 GHz; Type:0x10;
+ Core:16; Strand:2;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
+ FRU-Part-Number: CA20393-B50X A2 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
:
```

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.
You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

3. **Execute the showbbstatus command to confirm that the XSCF in the SPARC M10-4S to be replaced is not the master XSCF.**

```
XSCF> showbbstatus
BB#00 (Master)
```

If the SPARC M10-4S to be replaced is the master XSCF, execute the switchscf command to switch the XSCF.

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

Note - Confirm that the XSCF has been switched and rebooted before you release the system board.

4. **Execute the console command to connect to the console of the control domain and then log in to it.**

```
XSCF> console -p 0
```

5. **Check the operation status and resource usage status of the logical domain.**
 - a. **Execute the ldm list-domain command to check the operation status of the logical domain.**

To check the logical domain operation status, check the [STATE] and [FLAGS] combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

The following example shows that the control domain, two root domains, and two guest domains are operating.

Check whether all domains are in "active", which indicates that Oracle Solaris is in operating state, or "inactive" state. If there is a domain with OpenBoot PROM status or bound status, the dynamic reconfiguration of the physical partition may fail.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.0%	1h 33m
guest0	active	-n----	5100	64	64G	3.1%	2s
guest1	active	-n----	5101	64	64G	1.6%	18m
root-dom0	active	-n--v-	5000	32	32G	3.1%	17m
root-dom1	active	-n--v-	5001	32	32G	3.1%	17m

- b. **Execute the `ldm list-devices` command with the `-a` option specified to check the resource usage status.**

In the following example, the `-a` option is specified to display all resources bound to the logical domain and all resources that are not bound.

```
# ldm list-devices -a
CORE
  ID      %FREE    CPUSSET
  0        0      (0, 1)
  4        0      (8, 9)
  8        0     (16, 17)
(Omitted)
  944      0     (1888, 1889)
  948      0     (1896, 1897)
  952      0     (1904, 1905)
  956      0     (1912, 1913)
VCPU
  PID      %FREE    PM
  0        0      no
  1        0      no
  8        0      no
  9        0      no
(Omitted)
  1904     0      no
  1905     0      no
  1912     0      no
  1913     0      no
(Omitted)
```

6. **Release the redundant configuration of the system volume and I/O devices in the control domain.**

To enable the release of BB#01, release the I/O devices of the SPARC M10-4S to be replaced and which are used in the control domain. For details on the procedure for canceling a redundant configuration, see the document for the software for that redundant configuration.

- a. **Cancel the redundant configuration of the system volume in the control domain.**

The following example describes how to cancel the ZFS mirroring function for the system volume in the control domain.

Execute the `zpool status` command in the control domain to check the mirroring configuration status.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
  NAME                                STATE    READ WRITE CKSUM
  rpool                                ONLINE    0     0     0
    mirror-0                           ONLINE    0     0     0
```

```

c2t50000393E802CCE2d0s0  ONLINE      0      0      0
c3t50000393A803B13Ed0s0  ONLINE      0      0      0
errors: No known data errors

```

Execute the `zpool detach rpool` command to release the disk from the mirroring configuration.

```
# zpool detach rpool c3t50000393A803B13Ed0
```

Execute the `zpool status` command to confirm that the mirroring configuration has been canceled.

```

# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                               ONLINE     0     0     0
        mirror-0                       ONLINE     0     0     0
            c2t50000393E802CCE2d0s0    ONLINE     0     0     0
errors: No known data errors

```

If you are using other devices in BB#01, remove the redundant configuration or stop using those devices. For details on how to cancel a redundant configuration or stop using the devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

b. Delete the I/O configuration of the control domain.

From among the physical I/O devices assigned to the control domain, delete the root complexes in BB#01 through dynamic reconfiguration.

Execute the `ldm list-io` command to check the root complexes assigned to the primary.

The following example shows that the root complexes with BB1 devices are PCIE8 and PCIE12.

```

# ldm list-io | grep primary
PCIE0                                BUS      PCIE0      primary  IOV
PCIE4                                BUS      PCIE4      primary  IOV
PCIE8                                BUS      PCIE8      primary  IOV
PCIE12                               BUS      PCIE12     primary  IOV
/BB0/CMUL/NET0                       PCIE     PCIE0      primary  OCC
/BB0/CMUL/SASHBA                     PCIE     PCIE0      primary  OCC
/BB0/CMUL/NET2                       PCIE     PCIE4      primary  OCC
/BB1/CMUL/NET0                       PCIE     PCIE8      primary  OCC
/BB1/CMUL/SASHBA                     PCIE     PCIE8      primary  OCC
/BB1/CMUL/NET2                       PCIE     PCIE12     primary  OCC

```

Execute the `ldm remove-io` command to delete PCIE8 and PCIE12 from the

primary.

```
# ldm remove-io PCIE8 primary
# ldm remove-io PCIE12 primary
```

Execute the `ldm list-io` command to confirm that the root complexes in BB#01 have been deleted from the control domain.

```
# ldm list-io | grep primary
PCIE0          BUS      PCIE0      primary  IOV
PCIE4          BUS      PCIE4      primary  IOV
/BB0/CMUL/NET0 PCIE     PCIE0      primary  OCC
/BB0/CMUL/SASHBA PCIE     PCIE0      primary  OCC
/BB0/CMUL/NET2 PCIE     PCIE4      primary  OCC
```

c. **Cancel the redundant configuration of the virtual I/O devices assigned to a guest domain.**

To first shut down the root domain (root-dom1) to which the root complexes in BB#01 are assigned, log in to each guest domain, and then cancel the redundant configuration of the virtual I/O device from root-dom1.

For details on how to use the redundant configuration software, see the documentation about the software for that redundant configuration.

In the following example, a virtual network device (vnet1) is canceled from the IPMP configuration. For details on the command, see the manual for Oracle Solaris.

Log in to the guest domain (guest0).

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active    -n-cv-   UART    64      56G       0.0%    4h 17m
guest0        active    -n----   5100    64      64G       0.0%    1h 13m
guest1        active    -n----   5101    64      64G       0.0%    1h 4m
root-dom0     active    -n--v-   5000    32      32G       0.0%    1h 47m
root-dom1     active    -n--v-   5001    32      32G       0.0%    1h 19m
# telnet localhost 5100
....
guest0#
```

Execute the `dladm show-phys` command to check the mapping between the virtual network interface (vnet1) and the network interface name (net1).

```
guest0# dladm show-phys
LINK          MEDIA          STATE    SPEED    DUPLEX    DEVICE
net0          Ethernet       up        0        unknown   vnet0
net1          Ethernet       up        0        unknown   vnet1
```

Execute the `ipmpstat -i` command to check the configuration information for the network interfaces configuring IPMP.

```

guest0# ipmpstat -i

```

INTERFACE	ACTIVE	GROUP	FLAGS	LINK	PROBE	STATE
net0	yes	ipmp0	-smbM--	up	disabled	ok
net1	no	ipmp0	is-----	up	disabled	ok

Execute the `if_mpadm -d` command to release `net1` from the IPMP group, and then execute the `ipmpstat -i` command to confirm that it has been released. The following example confirms that STATE is offline.

```

guest1# if_mpadm -d net1
guest1# ipmpstat -i

```

INTERFACE	ACTIVE	GROUP	FLAGS	LINK	PROBE	STATE
net0	yes	ipmp0	-smbM--	up	disabled	ok
net1	no	ipmp0	-s---d-	up	disabled	offline

Execute the `ipadm delete-ip` command to delete `net1`.

```

guest0# ipadm delete-ip net1

```

Similarly, perform the same release processing for the guest domain (`guest1`).

d. Remove the virtual I/O devices assigned from the root domain to be stopped.

Execute the `ldm remove-vdisk` and `ldm remove-vnet` commands to remove the assigned virtual disk (vdisk) and virtual network device (vnet) from the root domain to be stopped according to the following step.

The following example shows the execution of the commands for removing the virtual disk (vdisk10) and virtual network device (vnet10) that use the virtual I/O service of the BB#01 root domain (root-dom1).

```

# ldm remove-vdisk vdisk10 guest0
# ldm remove-vnet vnet10 guest0

```

Perform the same deletion for the guest domain (`guest1`).

7. Check the resource usage status of the I/O devices, and then cancel all the I/O devices in the SPARC M10-4S to be replaced.

a. Check the logical domain to which the root complexes in the SPARC M10-4S to be released are assigned.

Execute the `ldm list-io` command to check the logical domain to which the root complexes of SPARC M10-4S in BB#01 are assigned.

The following example shows that only `root-dom1` has PCIe endpoints starting with `"/BB1/."` You can see that the PCIe endpoint root complexes (BUS) PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, and PCIE15 are assigned to `root-dom1`.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV
....				
/BB1/CMUL/NET0	PCIE	PCIE8		UNK
/BB1/CMUL/SASHBA	PCIE	PCIE8		UNK
/BB1/PCI0	PCIE	PCIE9	root-dom1OCC	
/BB1/PCI3	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI4	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI7	PCIE	PCIE11	root-dom1OCC	
/BB1/PCI8	PCIE	PCIE11	root-dom1OCC	
/BB1/CMUL/NET2	PCIE	PCIE12		UNK
/BB1/PCI1	PCIE	PCIE13	root-dom1OCC	
/BB1/PCI2	PCIE	PCIE13	root-dom1OCC	
/BB1/PCI5	PCIE	PCIE14	root-dom1OCC	
/BB1/PCI6	PCIE	PCIE14	root-dom1OCC	
/BB1/PCI9	PCIE	PCIE15	root-dom1OCC	
/BB1/PCI10	PCIE	PCIE15	root-dom1OCC	

- b. **Stop the root domain to which the root complexes in SPARC M10-4S to be released are assigned and then release the SPARC M10-4S.**

The following example executes the ldm stop-domain and ldm unbind-domain commands to release the root domain (root-dom1) and shows that the root domain is in the inactive state.

```
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain root-dom1
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.2%	4h 59m
guest0	active	-n----	5100	64	64G	0.0%	1h 55m
guest1	active	-n----	5101	64	64G	0.0%	1h 46m
root-dom0	active	-n--v-	5000	32	32G	0.0%	2h 29m
root-dom1	inactive	-----		32	32G		

- c. **Confirm that all of the I/O devices in the SPARC M10-4S to be replaced have**

been canceled.

Execute the `ldm list-io` command to confirm that all the I/O devices have been released.

```
# ldm list-io
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
PCIE0                              BUS     PCIE0    primary   IOV
PCIE1                              BUS     PCIE1    root-dom0 IOV
PCIE2                              BUS     PCIE2    root-dom0 IOV
PCIE3                              BUS     PCIE3    root-dom0 IOV
PCIE4                              BUS     PCIE4    primary   IOV
PCIE5                              BUS     PCIE5    root-dom0 IOV
PCIE6                              BUS     PCIE6    root-dom0 IOV
PCIE7                              BUS     PCIE7    root-dom0 IOV
PCIE8                              BUS     PCIE8
PCIE9                              BUS     PCIE9
PCIE10                             BUS     PCIE10
PCIE11                             BUS     PCIE11
PCIE12                             BUS     PCIE12
PCIE13                             BUS     PCIE13
PCIE14                             BUS     PCIE14
PCIE15                             BUS     PCIE15
(Omitted)
```

8. **Manually reduce the number of CPU cores and the memory resources assigned to the logical domains.**

The following explains the procedure for manually reducing the number of CPU cores and the memory resources assigned to the logical domains, instead of specifying the `-m unbind=resource` option, to satisfy the conditions for ["Placement of CPU cores and memory" in "2.5.2 Considerations in System Operation for Dynamic Reconfiguration."](#)

If, for example, the `deleteboard` command fails due to an error, it may be possible in some cases to execute the `deleteboard` command by following this procedure.

When specifying the `-m unbind=resource` option of the `deleteboard` command, go to step 9.

a. **Check the number of CPU cores and then delete them.**

The number of CPU cores that can be used will be reduced as a result of releasing the SPARC M10-4S, so first apply the following procedure to reduce the number of CPU cores assigned to the logical domains beforehand.

i. **Check the number of CPU cores after SPARC M10-4S has been released.**

From the XSCF, execute the `showpparinfo` command to check the number of CPU cores, excluding those of SPARC M10-4S to be released.

In the following example, if a physical system board (PSB) number of 01-0 is to be released, the sum of the CPU cores with another PSB number of 00-0 must be calculated. Thus, $16 + 16 + 16 + 16 = 64$ cores.

```

XSCF> showpparinfo -p 0
PPAR#00 Information:
-----
CPU(s)                :      8
CPU Cores              :     128
CPU Threads           :     256
Memory size (GB)      :     256
CoD Assigned (Cores)  :     256
CPU(s) :
-----
PID  PSB   CPU#  Cores  Threads
00   00-0  0       16      32
00   00-0  1       16      32
00   00-0  2       16      32
00   00-0  3       16      32
00   01-0  0       16      32
00   01-0  1       16      32
00   01-0  2       16      32
00   01-0  3       16      32
(Omitted)

```

ii. Check the total number of the CPU cores that are assigned to each logical domain.

Execute `ldm list-devices -a core`. The number of rows with a value other than 100 in the %FREE column is the total number of CPU cores assigned to the logical domain.

In the following example, the `ldm list-devices -a core` command is executed, and a check is performed using the `-p` option. As a result, it can be seen that 112 cores are bound to the entire logical domain.

```

# ldm list-devices -a core
CORE
  ID      %FREE  CPUSSET
  0        0    (0, 1)
  4        0    (8, 9)
  8        0   (16, 17)
 12        0   (24, 25)
(Omitted)
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
112

```

iii. Calculate the core shortfall that results from the release of the SPARC M10-4S.

Using the formula below, calculate the CPU core shortfall that will result after releasing SPARC M10-4S.

CPU core shortfall = Number of cores used in logical domain (step ii) -
Number of physical cores after release (step i)

For the example in steps i and ii, it is found that the shortfall will be 112 cores (in use) - 64 cores (remaining) = 48 cores.

iv. Consider which logical domains to delete if a CPU core shortfall occurs.

If it is found as a result of step iii that a CPU core shortfall will occur, it is necessary to delete the CPU cores from the logical domains.

Execute the `ldm list-domain` command to check the number of CPU cores assigned to each of the logical domains in an active or bound state, and then check the logical domains from which the CPU cores will be deleted.

In the following example, it is found that 32 cores (64vcpu) are assigned to primary, 32 cores (64vcpu) to guest0, 32 cores (64vcpu) to guest1, and 16 cores (32vcpu) to root-dom0. In the example, 48 cores need to be deleted. Thus, 16 cores will be deleted from each of primary, guest0, and guest1.

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active    -n-cv-   UART     64     56G      0.1%   18h 17m
guest0        active    -n----   5100     64     64G      0.0%   15h 13m
guest1        active    -n----   5101     64     64G      0.0%   15h 4m
root-dom0     active    -n--v-   5000     32     32G      0.0%   15h 47m
root-dom1     inactive  ------  5001     32     32G
```

v. Execute the `ldm remove-core` command to delete the CPU cores from the target logical domains.

In the following example, 16 cores are deleted from each of primary, guest0, and guest1, and a check is made to determine if they have actually been deleted.

```
# ldm remove-core 16 primary
# ldm remove-core 16 guest0
# ldm remove-core 16 guest1
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active    -n-cv-   UART     32     56G      0.0%   18h 19m
guest0        active    -n----   5100     32     64G      0.0%   15h 15m
guest1        active    -n----   5101     32     64G      0.0%   15h 5m
root-dom0     active    -n--v-   5000     32     32G      0.0%   15h 49m
root-dom1     inactive  ------  5001     32     32G
# ldm list-devices -a -p core | egrep -v "CORE|VERSION|free=100" | wc -l
64
```

b. Check and delete memory resources.

The memory area that can be used will be reduced as a result of releasing the SPARC M10-4S, so use the procedure below to delete the memory resources assigned to the logical domains.

i. Check the use status of the continuous region of the memory (memory block).

Execute the `prtdiag` command and the `ldm list-devices -a memory` command to check the memory block assigned to each logical domain, and check the SPARC M10-4Ss to which unassigned memory blocks are assigned.

First, execute the `prtdiag` command to check the correspondence between the physical addresses in memory and the SPARC M10-4Ss.

```

# prtdiag
(Omitted)
===== Physical Memory Configuration =====
Segment Table:
-----
Base          Segment  Interleave  Bank      Contains
Address       Size      Factor      Size      Modules
-----
0x7e0000000000 32 GB    4           8 GB     /BB0/CMUL/CMP0/MEM00A
(Omitted)
0x7c0000000000 32 GB    4           8 GB     /BB0/CMUL/CMP1/MEM10A
(Omitted)
0x7a0000000000 32 GB    4           8 GB     /BB0/CMUU/CMP0/MEM00A
(Omitted)
0x780000000000 32 GB    4           8 GB     /BB0/CMUU/CMP1/MEM10A
(Omitted)
0x760000000000 32 GB    4           8 GB     /BB1/CMUL/CMP0/MEM00A
(Omitted)
0x740000000000 32 GB    4           8 GB     /BB1/CMUL/CMP1/MEM10A
(Omitted)
0x720000000000 32 GB    4           8 GB     /BB1/CMUU/CMP0/MEM00A
(Omitted)
0x700000000000 32 GB    4           8 GB     /BB1/CMUU/CMP1/MEM10A
(Omitted)

```

The result of this example is rearranged in ascending order of physical addresses in memory. The following table lists the correspondence between the physical addresses and the SPARC M10-4S.

Table A-5 Example of Correspondence Between Physical Addresses and the SPARC M10-4S

Base Address (Physical Address)	Building Block Configuration of SPARC M10-4S
0x700000000000 and after	BB1
0x720000000000 and after	BB1
0x740000000000 and after	BB1
0x760000000000 and after	BB1
0x780000000000 and after	BB0
0x7a0000000000 and after	BB0
0x7c0000000000 and after	BB0
0x7e0000000000 and after	BB0

Then, execute the `ldm list-devices -a memory` command to check the continuous areas (called memory blocks in the remainder of this document) of memory assigned to the logical domains and any unassigned memory blocks.

In the following example, the `ldm list-devices -a memory` command is executed.

The meaning of each parameter is as follows.

PA: Start physical address of the memory block

SIZE: Size of the memory block

BOUND: Name of the logical domain to which the memory block is assigned A blank is an unassigned area, and `_sys_` is a control area that is not assigned to a logical domain.

# ldm list-devices -a memory		
MEMORY		
PA	SIZE	BOUND
0x700000000000	32G	root-dom0
0x720000000000	32G	
0x740000000000	32G	guest0
0x760000800000	1272M	_sys_
0x760050000000	31488M	guest1
0x780000000000	32G	guest0
0x7a0000000000	32G	guest1
0x7c0000000000	28G	primary
0x7c0700000000	4G	
0x7e0000800000	1272M	_sys_
0x7e0050000000	512M	_sys_
0x7e0070000000	256M	_sys_
0x7e0080000000	28G	primary
0x7e0780000000	1280M	guest1
0x7e07d0000000	768M	

By combining the results shown above with the physical positions checked with the `prtdiag` command, it can be seen that the memory block usage status is as below.

Table A-6 Example of Memory Block Use Statuses

SPARC M10-4S	Physical Address	Size	Logical Domain
BB1 (target for replacement)	0x700000000000	32 GB	root-dom0
	0x720000000000	32 GB	Unassigned
	0x740000000000	32 GB	guest0
	0x760050000000	31,488 MB	guest1
BB0	0x780000000000	32 GB	guest0
	0x7a0000000000	32 GB	guest1
	0x7c0000000000	28 GB	primary
	0x7c0700000000	4 GB	Unassigned
	0x7e0080000000	28 GB	primary
	0x7e0780000000	1,280 MB	guest1
	0x7e07d0000000	768 MB	Unassigned

ii. Check the size and quantity of the movement source memory blocks.

While referring to the check results of the memory block use status, check

the memory block (hereafter called "source memory block") assigned to the SPARC M10-4S to be replaced.

For "Table A-6 Example of Memory Block Use Statuses," memory blocks of 32 GB x 2 and 31,488 MB x 1 are being used in BB1 as the blocks assigned to the logical domain.

iii. **Determine the logical domains from which to delete memory, as well as the amounts.**

Then, check the locations of the memory blocks assigned to each logical domain, and make sure that the source memory blocks can be moved to unassigned memory blocks (called "destination memory blocks" in the remainder of this document) of the SPARC M10-4Ss that will not be released, by deleting the memory in memory block units and by reducing the memory block size.

As a result, make a final decision as to how much memory to delete and from which logical domains to delete it.

The following methods are supported.

- Reduce the number of memory blocks that need to be moved by deleting, as a whole, the source memory blocks.
- Increase the number of possible destinations by deleting, as a whole, the memory blocks assigned to the logical domains on the SPARC M10-4Ss that are not to be released.
- Reduce the size of the source memory blocks so that they will fit into the available free area at the destinations.
- Increase the number of free memory blocks at the destinations to make the movement possible, by reducing the size of the source memory blocks and by reducing the size of the memory blocks of the destinations that are in use.

Note - After the size is reduced, the free memory blocks will not be continuous (they will be fragmented). Even if multiple small memory blocks are deleted to increase the number of free areas, the free areas will not be continuous areas. If the continuous areas of the source memory blocks are large, movement will be impossible. In such a case, delete the source memory blocks to adjust their size.

Note - When considering this, the possibility of continuous areas being fragmented after deletion can be reduced by selecting deletable ones that have the same sizes as the existing memory blocks whenever possible. This increases the possibility of the success of the movement of memory blocks.

Note - If too much memory is deleted, this will place a burden on the memory in the logical domains, possibly causing problems such as Oracle Solaris hanging. It is necessary to be careful not to delete too much by using the vmstat command and checking the "free" size as a rough guide.

Investigate the deletion plan based on "Table A-6 Example of Memory Block Use Statuses."

As a result of this consideration, make a plan to delete 4 GB from root-dom0, 32 GB from guest0, 31,488 MB from guest1, and 28 GB from the primary as shown in "Table A-7 Memory Block Deletion Plan" as an example.

Table A-7 Memory Block Deletion Plan

SPARC M10-4S	Size	Logical Domain	Deletion Plan
BB1 (target for replacement)	32 GB	root-dom0	Reduce this area by 4 GB to 28 GB, delete 28 GB from the primary on BB0, and then implement the movement.
	32 GB	Unassigned	-
	32 GB	guest0	Delete this because there is a 32-GB guest0 memory on BB0.
	31,488 MB	guest1	Delete this because there is a 32-GB guest1 memory on BB0.
BB0	32 GB	guest0	Leave it.
	32 GB	guest1	Leave it.
	28 GB	primary	Leave it.
	4 GB	Unassigned	-
	28 GB	primary	Delete it to move 28 GB of root-dom0.
	1,280 MB	guest1	Leave it.
	768 MB	Unassigned	-

iv. Manually delete memory from the logical domains.

Delete memory from the logical domains, using the ldm remove-memory command, according to the memory deletion plan devised in step iii.

The following example indicates the execution of the command for deleting memory according to "Table A-7 Memory Block Deletion Plan."

```
# ldm remove-memory 4G root-dom0
# ldm remove-memory 32G guest0
# ldm remove-memory 31488M guest1
# ldm remove-memory 28G primary
```

v. Check the states of the deleted memory blocks.

Execute the ldm list-devices -a memory command to check if the layout is such that movement is possible, by referring to the results of the deletion. If there are any memory blocks that cannot be moved, consider which additional memory blocks to delete based on the results, and then delete them.

In the following example, it is easier to check if movement is possible by comparing, side by side, the large sizes assigned to BB1 with the large sizes

in the free areas on BB0.

# ldm list-devices -a memory				
MEMORY				
PA	SIZE	BOUND		
(BB1)				
0x700000000000	256M	root-dom0	→Can be moved by dividing a 4-GB destination	
0x700010000000	4G			
0x700110000000	28416M	root-dom0	→Can be moved to 0x780000000000 (28GB)	
0x720000000000	32G			
0x740000000000	256M	guest0	→Can be moved by dividing a 4-GB destination	
0x740010000000	4G			
0x740110000000	28416M	guest0	→Must be deleted again because there is no destination(*)	
0x760000800000	1272M	_sys_		
0x760050000000	256M	guest1	→Can be moved by dividing a 4-GB area	
0x760060000000	1792M			
0x7600d0000000	29440M	guest1	→Can be moved to 0x7a0000000000 (29GB)	
(BB0)				
0x780000000000	28G		←Can be moved from rootdom0 (0x700110000000)	
0x780700000000	4G	guest0		
0x7a0000000000	29G		←Can be moved from guest1 (0x7600d0000000)	
0x7a0740000000	3G	guest1		
0x7c0000000000	256M	primary		
0x7c0010000000	4G		←Can be moved from 256 MB of root-dom0, guest0, or guest1	
0x7c0110000000	24320M	primary		
0x7c0700000000	4G			
0x7e0000800000	1272M	_sys_		
0x7e0050000000	512M	_sys_		
0x7e0070000000	256M	_sys_		
0x7e0080000000	24G		←Not sufficient for guest0 (0x740110000000) to move(*)	
0x7e0680000000	4G	primary		
0x7e0780000000	1280M	guest1		
0x7e07d0000000	768M			

In the above example, the destination marked with (*) has only 24 GB (24,576 MB) of free space, so delete 3,840 MB from the 28416-MB area (guest0) on the source (BB1), and then repeat the check. In the following example, it can be seen that all of the memory blocks can now be moved.

```
# ldm remove-memory 3840M guest0
# ldm list-devices -a memory
MEMORY
  PA                SIZE                BOUND
(BB1)
  0x700000000000    256M                root-dom0
  0x700010000000     4G                  root-dom0
  0x700110000000    28416M               root-dom0
  0x720000000000    32G                  guest0
  0x740000000000    256M                guest0
  0x740010000000    7936M                guest0
  0x740200000000    24G                  guest0 →Can be moved to
                                                0x7e0080000000 (24G)
  0x760000800000    1272M                _sys_
  0x760050000000    256M                guest1
  0x760060000000    1792M                guest1
  0x7600d0000000    29440M               guest1
(BB0)
  0x780000000000    28G                  guest0
  0x780700000000     4G                  guest0
  0x7a0000000000    29G                  guest1
  0x7a0740000000     3G                  guest1
  0x7c0000000000    256M                primary
  0x7c0010000000     4G                  primary
  0x7c0110000000    24320M               primary
  0x7c0700000000     4G                  primary
  0x7e0000800000    1272M                _sys_
  0x7e0050000000    512M                _sys_
  0x7e0070000000    256M                _sys_
  0x7e0080000000    24G                  ←Can be moved from guest0
                                                (0x740200000000)
  0x7e0680000000     4G                  primary
  0x7e0780000000    1280M                guest1
  0x7e07d0000000    768M
```

9. Release the system board (PSB<BB>) of the SPARC M10-4S from the physical partition.
 - a. Execute the deleteboard -c disconnect command to release the PSB from the physical partition.

```
XSCF> deleteboard -c disconnect 01-0
PSB#01-0 will be unconfigured from PPAR immediately. Continue?
[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.
```

- b. **Execute the `showresult` command to check the exit status of the `deleteboard` command that was just executed.**

An end value of 0 indicates the normal termination of the `deleteboard` command.

If the end value is other than 0 or if an error message is displayed upon executing the `deleteboard` command, it indicates abnormal termination of the `deleteboard` command. By referring to "C.1.2 `deleteboard`" based on the error message, identify the error and then take corrective action.

```
XSCF> showresult
0
```

- c. **Execute the `showboards` command to check the PSB status.**

Confirm that the PSB in the SPARC M10-4S to be replaced is in the "Assigned" state and that the [Pwr], [Conn], and [Conf] columns all show "n."

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)         Assigned    y    y    y    Passed Normal
01-0  00(01)         Assigned    n    n    n    Passed Normal
```

10. **Execute the `replacefru` command to replace the SPARC M10-4S.**

```
XSCF> replacefru
```

Note - For details on the replacement of SPARC M10-4Ss by using the `replacefru` command, see "5.8 Releasing a SPARC M10-4/M10-4S FRU from the System with the `replacefru` Command" and "6.2 Incorporating a SPARC M10-4/M10-4S FRU into the System with the `replacefru` Command" in the *Fujitsu M10-4/Fujitsu M10-4S/SPARC M10-4/SPARC M10-4S Service Manual*.

11. **Incorporate the PSB into the physical partition.**

- a. **Execute the `showboards` command to check the PSB status.**

Confirm that the PSB in the SPARC M10-4S to be replaced is in the Assigned state and that the [Pwr], [Conn], and [Conf] columns all show "n."

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)         Assigned    y    y    y    Passed Normal
01-0  00(01)         Assigned    n    n    n    Passed Normal
```

- b. **Execute the `addboard -c` configure command to incorporate the PSB into the physical partition.**

To recover the original logical domain configuration, execute the addboard -c configure command with the -m bind=resource option specified.

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

Note - If an error message appears during execution of the addboard command, see "[C.1.1 addboard](#)," and then identify the error and take corrective action.

- c. **Execute the showresult command to check the exit status of the addboard command that was just executed.**

An end value of 0 indicates the normal termination of the addboard command. If the end value is other than 0 or if an error message is displayed upon executing the addboard command, it indicates abnormal termination of the addboard command. By referring to "[C.1.1 addboard](#)" based on the error message, identify the error and then take corrective action.

```
XSCF> showresult
0
```

- d. **Execute the showboards command to check the PSB status.**

Confirm that both of the [Conn] and [Conf] columns show "y" after the PSB in the replaced SPARC M10-4S has been successfully incorporated.

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)         Assigned    y    y    y    Passed  Normal
01-0  00(01)         Assigned    y    y    y    Passed  Normal
```

- 12. **Check the logical domain operation status.**
 - a. **Execute the console command to connect to the console of the control domain and then log in to it.**

```
XSCF> console -p 0
```

- b. **Execute the `ldm list-domain` command to confirm that the logical domain operation status has not changed after the addition of the SPARC M10-4S PSB (BB).**

To check the logical domain operation status, check the [STATE] and [FLAGS] combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

- "n": Oracle Solaris is operating
- "t": OpenBoot PROM status
- "-": In another state (including [STATE] other than "active")

# <code>ldm list-domain</code>							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	28G	64%	2h 54m
guest0	active	-n----	5100	32	61876M	42%	2h 54m
guest1	active	-n----	5101	32	62388M	11%	2h 54m
root-dom0	active	-n--v-	5000	32	28G	0.0%	2h 54m
root-dom1	inactive	-----		32	32G	0.0%	2h 54m

13. Restore the deleted resources.

If the resources assigned to the logical domains have been deleted with the `deleteboard` command, add the resources with the `ldm add-core` and `ldm add-memory` commands to restore them.

In the following example, deleted CPU cores and memory are added to restore the resources before the replacement of the SPARC M10-4S.

# <code>ldm add-core 16 primary</code>							
# <code>ldm add-core 16 guest0</code>							
# <code>ldm add-core 16 guest1</code>							
# <code>ldm add-memory 28G primary</code>							
# <code>ldm add-memory 4G root-dom0</code>							
# <code>ldm add-memory 36608M guest0</code>							
# <code>ldm add-memory 31488M guest1</code>							
# <code>ldm list-domain</code>							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.2%	4h 59m
guest0	active	-n----	5100	64	64G	0.0%	1h 55m
guest1	active	-n----	5101	64	64G	0.0%	1h 46m
root-dom0	active	-n--v-	5000	32	32G	0.0%	2h 29m
root-dom1	inactive	-----		32	32G		

14. Restart the use of the I/O devices.

a. Reassign root complexes.

Execute the `ldm bind-domain` and `ldm start-domain` commands to start the root domain in the unbind state to which root complexes in the replaced SPARC M10-4S were assigned.

The following example starts the root domain (root-dom1) in the unbind state, and confirms that it has started.

```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.2%	3h 8m
guest0	active	-n----	5100	64	64G	0.0%	3h 8m
guest1	active	-n----	5101	64	64G	0.0%	3h 8m
root-dom0	active	-n--v-	5000	32	32G	0.0%	3h 8m
root-dom1	active	-n--v-	5001	32	32G	7.3%	8s

Execute the ldm list-io command to confirm that the physical I/O devices are assigned to the root domain that has just started.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV
....				
/BB1/CMUL/NET0	PCIE	PCIE8		UNK
/BB1/CMUL/SASHBA	PCIE	PCIE8		UNK
/BB1/PCI0	PCIE	PCIE9	root-dom1OCC	
/BB1/PCI3	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI4	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI7	PCIE	PCIE11	root-dom1OCC	
/BB1/PCI8	PCIE	PCIE11	root-dom1OCC	
/BB1/CMUL/NET2	PCIE	PCIE12		UNK
/BB1/PCI1	PCIE	PCIE13	root-dom1OCC	
/BB1/PCI2	PCIE	PCIE13	root-dom1OCC	
/BB1/PCI5	PCIE	PCIE14	root-dom1OCC	
/BB1/PCI6	PCIE	PCIE14	root-dom1OCC	
/BB1/PCI9	PCIE	PCIE15	root-dom1OCC	
/BB1/PCI10	PCIE	PCIE15	root-dom1OCC	

b. Add the virtual I/O device from the root domain to the guest domain.

Execute the ldm add-vdisk and ldm add-vnet commands to add, to each guest domain, the virtual disk (vdisk) and virtual network device (vnet) supported for the virtual I/O service of the started root domain.

The following example indicates the execution of the command for adding the virtual disk (vdisk10) and virtual network device (vnet10) that use the virtual I/O service of BB#01 root domain (root-dom1).

```
# ldm add-vdisk id=1 vdisk10 vol10@vds1 guest0
# ldm add-vnet id=1 vnet10 vsw10 guest0
```

Perform the same addition for the guest domain (guest1).

Note - To add the virtual I/O device again, it is necessary to specify the ID assigned beforehand. You can check the ID from the result of execution of the `ldm list -l` command in the status used before the virtual I/O device is deleted.

c. Incorporate the virtual I/O devices assigned to a guest domain into the redundant configuration.

Once the root domain (root-dom1) to which root complexes in BB#1 have been assigned has started, the virtual I/O device services corresponding to each guest domain are also started.

Log in to each guest domain, and then incorporate the virtual I/O devices from root-dom1 that were previously canceled into the redundant configuration. For details on how to use the redundant configuration software, see the documentation for the software for that redundant configuration.

The following describes an example of incorporating a virtual network device (vnet1) into the IPMP configuration. For details on the commands, see the manual for Oracle Solaris.

First, log in to the guest domain (guest0).

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active    -n-cv-   UART     64      56G       0.0%    4h 17m
guest0        active    -n----   5100     64      64G       0.0%    1h 13m
guest1        active    -n----   5101     64      64G       0.0%    1h 4m
root-dom0     active    -n--v-   5000     32      32G       0.0%    1h 47m
root-dom1     active    -n--v-   5001     32      32G       0.0%    1h 19m
# telnet localhost 5100
....
guest0#
```

Execute the `dladm show-phys` command to check the mapping between the virtual network interface (vnet1) and the network interface name (net1).

```
guest0# dladm show-phys
LINK          MEDIA          STATE    SPEED    DUPLEX    DEVICE
net0          Ethernet       up        0        unknown   vnet0
net1          Ethernet       up        0        unknown   vnet1
```

Execute the `ipadm create-ip net1` command, the `ipadm set-ifprop -p standby=on -m ip net1` command, and the `ipadm add-ipmp -i net1 ipmp0` command to register `net1` as a standby device of `ipmp0`.

```
guest0# ipadm create-ip net1
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipadm add-ipmp -i net1 ipmp0
```

Execute the `ipmpstat -i` command to confirm that STATE of the network interface configuring IPMP indicates ok.

```
guest0# ipmpstat -i
INTERFACE ACTIVE GROUP FLAGS LINK PROBE STATE
net0 yes ipmp0 -smbM-- up disabled ok
net1 no ipmp0 -s---d- up disabled ok
```

Perform the same step for the other guest domain (`guest1`).

15. **Restore the system volume and I/O devices on the control domain to a redundant configuration.**

a. **Add the root complex configuration for the control domain.**

Add the root complexes in BB#01 that were previously removed from the control domain.

Execute the `ldm list-io` command to check the unassigned root complexes.

The following example shows that root complexes with BB1 devices, PCIE8 and PCIE12 are not assigned.

```
# ldm list-io
NAME                                     TYPE    BUS      DOMAIN    STATUS
----                                     -
PCIE0                                   BUS     PCIE0    primary   IOV
PCIE1                                   BUS     PCIE1    root-dom0 IOV
PCIE2                                   BUS     PCIE2    root-dom0 IOV
PCIE3                                   BUS     PCIE3    root-dom0 IOV
PCIE4                                   BUS     PCIE4    primary   IOV
PCIE5                                   BUS     PCIE5    root-dom0 IOV
PCIE6                                   BUS     PCIE6    root-dom0 IOV
PCIE7                                   BUS     PCIE7    root-dom0 IOV
PCIE8                                   BUS     PCIE8
PCIE9                                   BUS     PCIE9    root-dom1 IOV
PCIE10                                  BUS     PCIE10    root-dom1 IOV
PCIE11                                  BUS     PCIE11    root-dom1 IOV
PCIE12                                  BUS     PCIE12
PCIE13                                  BUS     PCIE13    root-dom1 IOV
(Omitted)
```

Execute the `ldm add-io` command to add PCIE8 and PCIE12 to primary.

```
# ldm add-io PCIE8 primary
# ldm add-io PCIE12 primary
```

Execute the `ldm list-io` command to confirm that the root complexes in BB#01 have been added to the control domain.

```
# ldm list-io | grep primary
PCIE0          BUS      PCIE0      primary    IOV
PCIE4          BUS      PCIE4      primary    IOV
PCIE8          BUS      PCIE8      primary    IOV
PCIE12         BUS      PCIE12     primary    IOV
/BB1/CMUL/NET0 PCIE      PCIE0      primary    OCC
/BB1/CMUL/SASHBA PCIE      PCIE0      primary    OCC
/BB1/CMUL/NET2 PCIE      PCIE4      primary    OCC
```

b. Place the system volume in the control domain in a redundant configuration.

Execute the `zpool status` command in the control domain to check the mirroring configuration status.

The following example describes how to configure the ZFS mirroring function for the system volume in the control domain.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
  NAME                                STATE     READ WRITE CKSUM
  rpool                               ONLINE    0     0     0
    c2t50000393E802CCE2d0s0          ONLINE    0     0     0
errors: No known data errors
```

Execute the `zpool attach` command to incorporate the disks into a mirroring configuration.

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

Execute the `zpool status` command, and then confirm that the mirroring configuration has been established.

Use the `zpool status` command to confirm whether synchronization processing (resilver) is completed.

The following shows an example of the display during synchronization processing.

```
# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
       continue to function in a degraded state.
action: Wait for the resilver to complete.
```

```

Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                               DEGRADED   0      0      0
        mirror-0                       DEGRADED   0      0      0
            c2t50000393E802CCE2d0s0    ONLINE    0      0      0
            c3t50000393A803B13Ed0s0    DEGRADED   0      0      0 (resilvering)
errors: No known data errors

```

Once synchronization processing is complete, the displayed screen will be as follows:

```

# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                               ONLINE     0      0      0
        mirror-0                       ONLINE     0      0      0
            c2t50000393E802CCE2d0s0    ONLINE    0      0      0
            c3t50000393A803B13Ed0s0    ONLINE    0      0      0
errors: No known data errors

```

If you are using other devices in BB#01, establish a redundant configuration or resume the use of the devices. For details on how to establish a redundant configuration or resume the use of devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

A.3 For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has Free Hardware Resources

This section describes how to establish a configuration with free hardware resources (CPU core and memory) required to move resources during physical partition dynamic reconfiguration with two SPARC M10-4Ss, i.e. two system boards (PSB<BB>). That is, this section describes an example of how to establish a configuration whereby sufficient hardware resources for a single system board are assigned to a logical domain. It also describes an example of how to perform the active replacement of a SPARC M10-4S through physical partition dynamic

reconfiguration.

In this case, you can actively replace the system board without reducing the CPU cores and memory assigned to the logical domains. This description also applies to SPARC M12.

A.3.1 Configuration Example

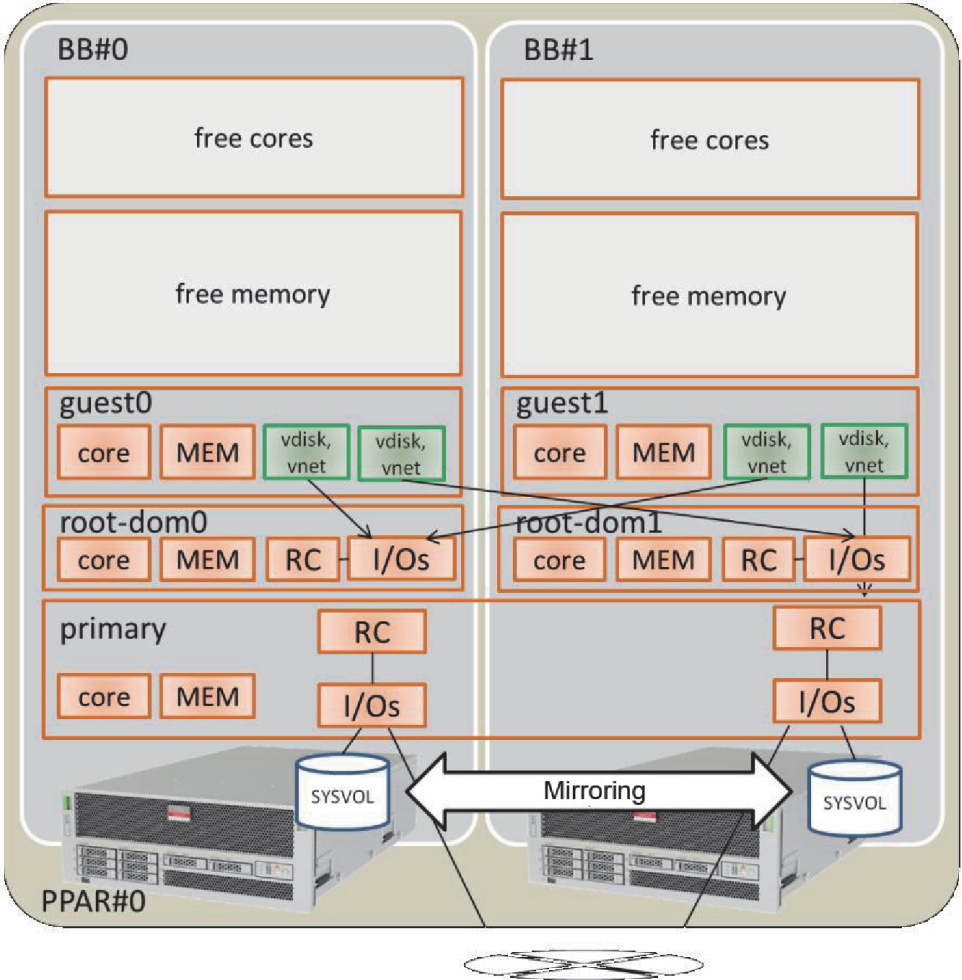
This section describes an example of configuring a logical domain consisting of two SPARC M10-4S that have the same number of CPUs and memory size, i.e. two system boards (PSB<BB>) (2BB configuration).

To enable physical partition dynamic reconfiguration in a physical partition, the I/O configuration must satisfy the following conditions.

- The system volume disk and network interface of the control domain must be made redundant by connecting the I/O of each SPARC M10-4S. This is done to make it possible to continue operation with the I/O of the remaining SPARC M10-4Ss when a SPARC M10-4S is removed.
- It is necessary to place a root domain for each SPARC M10-4S and to use that root domain as the dedicated one for providing a virtual I/O to the guest domain. The reason for this is as follows: If the root domain is being used for business, it must be stopped upon the disconnection of the SPARC M10-4S and so the business will stop.
- The business must be operated in the guest domain. The virtual I/O service of each root domain must be assigned to the virtual I/O devices (vdisk, vnet) of this guest domain to make those virtual I/O devices redundant. The reason for this is as follows: Even if the SPARC M10-4S is disconnected on one side, I/O access can be continued by using the virtual I/O service of the root domain on the remaining SPARC M10-4S. In this way, disconnecting the SPARC M10-4S does not influence the guest domain, allowing business operations to continue.

The following is a schematic diagram of a configuration that satisfies the above conditions. Moreover, the I/O configuration of each logical domain is simplified.

Figure A-3 Example of the 2BB Configuration for Operational Continuity (With Free Resources)



The following table gives an example of the CPU cores, memory, and I/O configuration to be assigned to the control domain, root domain, and guest domain in the 2BB configuration.

The total number of CPU cores and the total amount of memory assigned to each logical domain must be configured for a single SPARC M10-4S to cover it when releasing one of the SPARC M10-4Ss.

Table A-8 Example of the 2BB Configuration With Free Hardware Resources

Logical Domain Name	primary	root-dom0	root-dom1	guest0	guest1	Free space	Total
CPU core	8	12	12	16	16	64	128 (activation: 64)
Memory	14 GB (*1)	24 GB	24 GB	32 GB	32 GB	128 GB	256 GB
Physical I/O (Root complex assignment)	On-board #0 (PCIE0, PCIE4) On-board #1 (PCIE8, PCIE12)	PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, PCIE7	PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, PCIE15	-	-	-	-
Virtual I/O device	-	vds0 vsw0, vsw1	vds1 vsw10, vsw11	vdisk0, vdisk10 vnet0, vnet10	vdisk1, vdisk11 vnet1, vnet11	-	-

*1 In the configuration example shown above, the size of memory to be assigned to the control domain is calculated as the remainder of the memory after assignment to the other logical domains.
When the remaining size, obtained by subtracting the sizes assigned to domains other than the control domain from the capacity (128 GB) of the physical memory of a single SPARC M10-4S, is obtained, it is 16 GB.
128 GB - 32 GB - 32 GB - 24 GB - 24 GB = 16 GB
The maximum size to be actually assigned to the control domain is up to the size (14 GB) obtained by subtracting, from the above result, the maximum memory size (2 GB) assigned to the hypervisor in a single SPARC M10-4S.
16 GB - 2 GB = 14 GB
In addition, whether it is a multiple of (number of cores x 256 MB) is checked when the version of Oracle VM Server for SPARC is earlier than 3.2. This is the condition for performing physical partition dynamic reconfiguration. Since it turns out that it satisfies the condition, the memory size becomes 14 GB.
14 GB/(8 cores x 256 MB) = 7
Dividable. So there is no problem with 14 GB.

Note - You should design the size of memory to be assigned to logical domains by noting the following two points. For details, see "2.4.1 Consideration of Logical Domain Configuration."
- The size of memory to be assigned to logical domains becomes smaller than the physical memory size.
- When the version of Oracle VM Server for SPARC is earlier than 3.2, use a multiple of (number of cores x 256 MB) to enable physical partition dynamic reconfiguration.

A.3.2 Example of the Physical Partition Configuration Procedure

1. **Log in to the master XSCF.**
Execute the showbbstatus command to check that the XSCF to which you have logged in is the master XSCF.
If you have logged in to a standby XSCF, log out and then log in to the master

XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

2. **Set mirror mode for memory.**

- a. **Execute the showfru command to check mirror mode of the memory on the SPARC M10-4S system board (PSB<BB>).**

The following example displays the setting information for devices under the physical system board (PSB 00-0).

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
  cpu 00-0-0 no
  cpu 00-0-1 no
  cpu 00-0-2 no
  cpu 00-0-3 no
```

- b. **To use the memory mirror mode, execute the setupfru command to set that mode.**

If you are not using memory mirror mode, this step is not necessary.

For details on memory mirror mode, see "14.1 Configuring Memory Mirroring" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

The following example places all the CPUs under PSB 00-0 in memory mirror mode.

```
XSCF> setupfru -m y sb 00-0
```

Execute the showfru command to check the setting of memory mirror mode.

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
  cpu 00-0-0 yes
  cpu 00-0-1 yes
  cpu 00-0-2 yes
  cpu 00-0-3 yes
```

3. **Create the physical partition configuration information.**

- a. **Execute the showpctl command to check the physical partition configuration information.**

```
XSCF> showpctl -p 0
PPAR-ID LSB PSB Status
```

b. **Execute the setpcl command to register the system board in the physical partition configuration information.**

Execute the setpcl command to register the SPARC M10-4S system board in the physical partition configuration information for the built-in destination.

In the following example, physical system boards (PSBs) 00-0 and 01-0 are mapped to logical system boards (LSBs) 00 and 01 of physical partition 0.

```
XSCF> setpcl -p 0 -a 00=00-0 01=01-0
```

c. **Execute the showpcl command to check the physical partition configuration information.**

Check the physical partition configuration information.

```
XSCF> showpcl -p 0
PPAR-ID      LSB      PSB      Status
00
              00      00-0
              01      01-0
              Running
```

Specify the -v option to display details on the configuration policy, IO nullification option (No-IO), and memory nullification option (No-Mem).

```
XSCF> showpcl -v -p 0
PPAR-ID      LSB      PSB      Status      No-Mem      No-IO      Cfg-policy
00
              00      00-0      Running
              01      01-0
              False      False      System
              False      False
```

Use the setpcl command to change the setting of the configuration policy, IO nullification option (No-IO), and memory nullification option (No-Mem).

For details on the setpcl command, see the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual*.

4. **Assign a system board (PSB<BB>) to a physical partition.**

a. **Execute the showboards-a command to check the PSB status.**

Execute the showboards -a command to check that each PSB status is "SP" (system board pool).

```
XSCF> showboards -a
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  SP           Available  n    n    n    Passed  Normal
01-0  SP           Available  n    n    n    Passed  Normal
```

b. **Execute the addboard -c assign command to assign the PSBs.**

```
XSCF> addboard -c assign -p 0 00-0 01-0
```

c. **Execute the showboards-p command to check the PSB status.**

Execute the showboards -p command to check the status of each PSB assigned to the physical partition.

This example checks that the [Assignment] field of each PSB becomes "Assigned" since each PSB has been normally assigned to physical partition 0.

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	n	n	n	Passed	Normal
01-0	00(01)	Assigned	n	n	n	Passed	Normal

5. **Register the CPU Activation key to assign CPU core resources.**

a. **Execute the showcodactivation command to check the information on the CPU Activation key.**

Execute the showcodactivation command to check whether the physical partition contains an assignable CPU Activation key.

If only the header is displayed, the CPU Activation key is not registered in the XSCF.

```
XSCF> showcodactivation
Index Description Count
-----
```

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and add the CPU Activation keys.

b. **Execute the addcodactivation command to register the CPU Activation key.**

For details on how to register a CPU Activation key, see "5.3 Adding CPU Core Resources" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

```
XSCF> addcodactivation "Product: SPARC M10-4S
SequenceNumber:10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxeUDdPQHKbtxgvgZPsrtYguqiNUieB+mTDC
:
:
b1GckFx1RH27FdVHiB2H0A=="
AboveKeywillbeadded,Continue?[y|n]:y
```

c. **Execute the showcodactivation command to check the information on the CPU Activation keys.**

Execute the `showcodactivation` command to check whether the physical partition contains an assignable CPU Activation key.

```
XSCF> showcodactivation
Index      Description  Count
-----
          0  PROC           2
          1  PROC           2
          2  PROC           2
          3  PROC           2

--- Omitted ---

        30  PROC           2
        31  PROC           2
```

d. **Execute the `setcod` command to assign CPU core resources.**

Execute the `setcod` command to assign the CPU core resources to the physical partition.

The following example assigns 64 CPU core resources to physical partition 0.

```
XSCF> setcod -p 0 -s cpu -c set 64
PROC Permits assigned for PPAR 0 : 0 -> 64

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the `-c add`, `-c delete`, and `-c set` options. Specify the options of the `setcod` command as shown below to interactively add and delete CPU core resources.

XSCF> **setcod -s cpu**

Execute the `showcod` command to check information on the CPU core resources assigned to the physical partition.

The following example confirms that 64 CPU core resources have been assigned to physical partition 0 with the `setcod` command that was just executed.

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 64
```

6. **Execute the `resetdateoffset` command to reset the difference with the time managed by the XSCF.**

Execute the `resetdateoffset` command to reset the difference between the time managed by the XSCF and the time managed by the physical partitions.

```
XSCF> resetdateoffset -p 0
```

7. **Execute the showpparmode command to check the setting of the detail level of the diagnosis messages and that of the PPAR DR mode.**

Execute the showpparmode command to check that the detail level (Message Level) of the diagnosis message is "normal" (standard) and that Next of the PPAR DR mode is set to "on" (enabled).

```
XSCF> showpparmode -p 0
Host-ID                :9007002b
Diagnostic Level        :min
Message Level          :normal
Alive Check            :on
Watchdog Reaction      :reset
Break Signal           :on
Autoboot (Guest Domain) :on
Elastic Mode           :off
IOreconfigure          :false
PPAR DR (Current)      :-
PPAR DR (Next)         :on
```

If the detail level of the diagnosis message is other than "normal", execute the setpparmode command to set it to "normal".

For details on the setpparmode command, see the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual*.

```
XSCF> setpparmode -p 0 -m message=normal
```

If PPAR DR mode is set to "off" (disabled), execute the setpparmode command to set it to "on".

```
XSCF> setpparmode -p 0 -m ppar_dr=on
```

8. **Execute the poweron command to power on the physical partitions.**

```
XSCF> poweron -p 0
```

9. **Execute the console command to connect the console to the physical partition.**

```
XSCF> console -p 0
```

10. **Install Oracle Solaris and Oracle VM Server for SPARC.**

Install, on the control domain, Oracle Solaris and Oracle VM Server for SPARC.

For details on the versions and conditions of Oracle Solaris required for physical partition dynamic reconfiguration, see [Table 1-3](#) and [Table 1-4](#).

For details on the installation, see the following documents, presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

- Oracle Solaris 10

Oracle Solaris 10 1/13 Installation Guide

- Oracle Solaris 11

Installing Oracle Solaris 11.x Systems

- Oracle VM Server for SPARC

"Installing and Enabling Software" in the *Oracle VM Server for SPARC Administration Guide*

11. Configure a logical domain.

This item describes how to configure the logical domains defined in "Figure A-3 Example of the 2BB Configuration for Operational Continuity (With Free Resources)."

a. Reduce the number of control domain resources.

For the factory-default configuration, all the CPU cores, memory, and the PCIe root complexes are assigned to the control domain (primary). To allow these resources to be assigned to other logical domains, first delete some of the resources from the control domain and configure the console service.

The following shows command execution examples.

Execute the `ldm start-reconf` command to switch to delayed reconfiguration mode.

```
# ldm start-reconf primary
```

```
Initiating a delayed reconfiguration operation on the primary domain.  
All configuration changes for other domains are disabled until the primary  
domain reboots, at which time the new configuration for the primary domain  
will also take effect.
```

Remove the root complex with the `ldm remove-io` command.

The following example partially describes the command for removing PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, PCIE7, PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, and PCIE15 according to the configuration example.

```
# ldm remove-io PCIE1 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
....
```

```
# ldm remove-io PCIE15 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

Reduce the number of CPU cores and the size of memory assigned to the

control domain by specifying a size smaller than the original size with the `ldm set-core` and `ldm set-memory` commands.

The following gives an example of setting the number of CPU cores of the control domain to 8 and the memory size to 14 GB, according to the configuration example.

```
# ldm set-core 8 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
# ldm set-memory 14G primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

Note - We recommend that you first configure the CPU cores with the `ldm set-core` command and then the memory with the `ldm set-memory` command. This facilitates the assignment of a continuous area of collective memory. This also makes it easy to easily satisfy the placement conditions for the CPU cores for which physical partition dynamic reconfiguration is possible and the memory placement conditions. For details on the CPU core and memory placement conditions, see "[Placement of CPU cores and memory](#)" in "[2.5.2 Considerations in System Operation for Dynamic Reconfiguration](#)."

Create a service of the virtual console terminal concentrator, called `vcc0`, with the `ldm add-vconscon` command and then start the daemon of the virtual network terminal server (`vntsd`) with the `svcadm` command. Configure a console connection to each logical domain via this `vcc0`.

```
# ldm add-vconscon port-range=5000-5200 vcc0 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
# svcadm enable vntsd
```

Save the configuration information and then restart Oracle Solaris.

The following example checks the configuration information saved with the `ldm list-spconfig` command, and then saves the configuration as name `ldm-set1` with the `ldm add-spconfig` command. Then, it again checks that the configuration has been saved with the `ldm list-spconfig` command and finally gives an example of restarting Oracle Solaris.

```
# ldm list-spconfig  
factory-default [current]  
# ldm add-spconfig ldm-set1  
# ldm list-spconfig
```

```
factory-default
ldm-set1 [current]
# shutdown -i6 -g0 -y
```

b. Establish a redundant configuration for the system volume of the control domain.

Establish a redundant configuration for the system volume of the control domain. This item describes an example of the commands for establishing a mirroring configuration with ZFS. For details, see the following documents presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

- For Oracle Solaris 10

"How to Create a Mirrored Root Pool (Post Installation)" in the *Oracle Solaris ZFS Administration Guide*

- For Oracle Solaris 11

"How to Configure a Mirrored Root Pool (SPARC or x86/VTOC)" in the *Oracle Solaris 11.1 Administration: ZFS File Systems*

To use other redundant configuration software, see the manual for that software.

Execute the `zpool status` command to check the status of the root pool.

The following example indicates that `c2t50000393E802CCE2d0s0` has been assigned to the default root pool (`rpool`).

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: none requested
config:
      NAME                                STATE      READ WRITE CKSUM
      rpool                                ONLINE      0     0     0
        c2t50000393E802CCE2d0s0          ONLINE      0     0     0
errors: No known data errors
```

Execute the `format` command to check which disks can be added.

The following example indicates that `c3t50000393A803B13Ed0` exists as the other disk.

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
    0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
        /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
        /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
    1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
        /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
        /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
```

```
Specify disk (enter its number): ^C
```

Execute the `zpool attach` command to add the second disk to the `rpool` to provide a mirror configuration.

The following example adds `c3t50000393A803B13Ed0s0` with the `zpool attach` command and then checks the status of the synchronization processing (resilver) with the `zpool status` command. By referring to status and action, you can determine that the synchronization processing is in progress. Regularly execute the `zpool status` command to check the status of the synchronization processing until the processing ends.

```
# zpool attach rpools c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
# zpool status rpools
pool: rpools
state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
       continue to function in a degraded state.
action: Wait for the resilver to complete.
       Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Wed Jan 29 21:35:39 2014
      3.93G scanned out of 70.6G at 71.9M/s, 0h15m to go
      3.90G resilvered, 5.56% done
config:
          NAME                                STATE      READ WRITE CKSUM
          rpools                              DEGRADED    0     0     0
            mirror-0                          DEGRADED    0     0     0
              c2t50000393E802CCE2d0s0        ONLINE     0     0     0
              c3t50000393A803B13Ed0s0        DEGRADED    0     0     0 (resilvering)
```

Upon the completion of the synchronization processing, `[state]` is set to "ONLINE", as shown below.

```
# zpool status rpools
pool: rpools
state: ONLINE
scan: resilvered 70.6G in 0h10m with 0 errors on Wed Jan 29 21:45:42 2014
config:
          NAME                                STATE      READ WRITE CKSUM
          rpools                              ONLINE     0     0     0
            mirror-0                          ONLINE     0     0     0
              c2t50000393E802CCE2d0s0        ONLINE     0     0     0
              c3t50000393A803B13Ed0s0        ONLINE     0     0     0
errors: No known data errors
```

c. Configure a root domain.

This item describes the procedure for configuring a root domain.

Execute the `ldm add-domain` command to add a logical domain named `root-dom0`.

```
# ldm add-domain root-dom0
```

Execute the ldm set-variable command to change the OpenBoot PROM environment variable "auto-boot?", which is designed to automatically boot Oracle Solaris, to "false" (disabled). By default, this setting is "true" (enabled). So, OpenBoot PROM tries to start Oracle Solaris automatically when Oracle Solaris is not installed. Changing this setting to disabled facilitates the work to be performed before Oracle Solaris installation.

```
# ldm set-variable auto-boot\?=false root-dom0
```

First, assign the CPU cores with the ldm set-core command and then assign the memory with the ldm set-memory command.

The following example assigns 12 CPU cores with the ldm set-core command and 24 GB of memory with the ldm set-memory command, according to the configuration example.

```
# ldm set-core 12 root-dom0
# ldm set-memory 24G root-dom0
```

Note - We recommend that you first configure the CPU cores with the ldm set-core command and then the memory with the ldm set-memory command. This facilitates the assignment of a continuous area of collective memory. This also makes it easy to easily satisfy the placement conditions for the CPU cores for which physical partition dynamic reconfiguration is possible and the memory placement conditions. For details on the CPU core and memory placement conditions, see "[Placement of CPU cores and memory](#)" in "[2.5.2 Considerations in System Operation for Dynamic Reconfiguration](#)."

Execute the ldm set-vconsole command to assign the virtual console (vcons).

The following example executes the ldm set-vconsole command to assign port number 5000 of the service (vcc0) of the virtual console terminal concentrator in the control domain to the virtual console.

```
# ldm set-vconsole service=vcc0 port=5000 root-dom0
```

The following example executes the ldm list-io -l command to display the PCI assignment status.

The line where NAME begins with "/BB0" and the [TYPE] column has "PCIE" means the PCIe endpoint on the SPARC M10-4S (BB0). The line where the [DOMAIN] column is empty indicates an unassigned PCIe endpoint and the related root complex is displayed in the [BUS] column.

Therefore, you can quickly understand that PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, and PCIE7 are unassigned root complexes on BB0.

```
# ldm list-io -l
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
(Omitted)
/BB0/CMUL/NET0                      PCIE    PCIE0    primary  OCC
[pci@8000/pci@4/pci@0/pci@9]
network@0
network@0,1
/BB0/CMUL/SASHBA                    PCIE    PCIE0    primary  OCC
[pci@8000/pci@4/pci@0/pci@0]
scsi@0/iport@f/disk@w50000393e802cce2,0
scsi@0/iport@f/disk@w50000393d8285226,0
scsi@0/iport@f/smp@w500000e0e06d027f
scsi@0/iport@f/enclosure@w500000e0e06d027d,0
scsi@0/iport@v0
/BB0/PCI0                           PCIE    PCIE1                                EMP
[pci@8100/pci@4/pci@0/pci@0]
/BB0/PCI3                           PCIE    PCIE2                                EMP
[pci@8200/pci@4/pci@0/pci@0]
/BB0/PCI4                           PCIE    PCIE2                                EMP
[pci@8200/pci@4/pci@0/pci@8]
/BB0/PCI7                           PCIE    PCIE3                                EMP
[pci@8300/pci@4/pci@0/pci@0]
/BB0/PCI8                           PCIE    PCIE3                                EMP
[pci@8300/pci@4/pci@0/pci@8]
/BB0/CMUL/NET2                      PCIE    PCIE4    primary  OCC
[pci@8400/pci@4/pci@0/pci@a]
network@0
network@0,1
/BB0/PCI1                           PCIE    PCIE5                                EMP
[pci@8500/pci@4/pci@0/pci@8]
/BB0/PCI2                           PCIE    PCIE5                                EMP
[pci@8500/pci@4/pci@0/pci@9]
/BB0/PCI5                           PCIE    PCIE6                                EMP
[pci@8600/pci@4/pci@0/pci@9]
/BB0/PCI6                           PCIE    PCIE6                                EMP
[pci@8600/pci@4/pci@0/pci@11]
/BB0/PCI9                           PCIE    PCIE7                                EMP
[pci@8700/pci@4/pci@0/pci@9]
/BB0/PCI10                          PCIE    PCIE7                                EMP
[pci@8700/pci@4/pci@0/pci@11]
(Omitted)
```

See the device path (string displayed as [pci@....]) displayed in the above result and "A.6 SPARC M10-4S Device Paths" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide* to determine which root complexes are to be assigned to the root domain.

In the configuration example, all the unassigned root complexes (PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, and PCIE7) on BB0 checked above are to be assigned. So, execute the ldm add-io command to assign them to root-dom0.

The following example shows command execution.

```
# ldm add-io PCIE1 root-dom0
# ldm add-io PCIE2 root-dom0
# ldm add-io PCIE3 root-dom0
# ldm add-io PCIE5 root-dom0
# ldm add-io PCIE6 root-dom0
# ldm add-io PCIE7 root-dom0
```

Place the root domain in the bound status with the ldm bind-domain command, and then execute the ldm list-io command to check that the root complexes have been assigned.

The following example checks that root-dom0 is bound with the ldm bind-domain command to check with the ldm list-io command that the root complexes have been assigned.

The line where the [TYPE] column is "BUS" and the [DOMAIN] column is "root-dom0" indicates the root complex assigned to root-dom0. BUS in that line is the name of the assigned root complex.

In the following example, you can check that PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, and PCIE7 have been assigned to root-dom0.

```
# ldm bind-domain root-dom0
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8	primary	IOV
(Omitted)				

According to step 11. c, configure the root domains (root-dom1 in the example) of the other SPARC M10-4Ss.

d. Install Oracle Solaris in the root domain.

This item describes the procedure for installing Oracle Solaris in the root domain.

Execute the ldm start-domain command to start root domain root-dom0.

```
# ldm start-domain root-dom0
LDom root-dom0 started
```

Execute the ldm set-variable command to change the OpenBoot PROM environment variable "auto-boot?", which is designed to automatically boot Oracle Solaris, to "true" (enabled). After Oracle Solaris is installed, execution

of the ldm start-domain command also starts Oracle Solaris.

```
# ldm set-variable auto-boot\?=true root-dom0
```

Execute the telnet command to connect to the console of the root domain.

The following example checks that the port number of root-dom0 is 5000 by executing the ldm list-domain command. It can also check that root-dom0 is stopped in the OpenBoot PROM (OBP) status by connecting to localhost port number 5000 with the telnet command.

```
# ldm list-domain
NAME                STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary             active     -n-cv-   UART    16      14G       0.0%    7h 7m
root-dom0           active     -t----- 5000    24      24G       0.0%    20s
root-dom1           bound     ------ 5001    24      24G
# telnet localhost 5000
....
{0} ok
```

Install Oracle Solaris in the root domain.

For details on the versions and conditions of Oracle Solaris required for physical partition dynamic reconfiguration, see [Table 1-3](#) and [Table 1-4](#).

The following example executes the command to start Oracle Solaris 11 installation through the network.

```
{0} ok boot net:dhcp
....
```

For details on the installation, see the following documents, presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

- Oracle Solaris 10

Oracle Solaris 10 1/13 Installation Guide

- Oracle Solaris 11

Installing Oracle Solaris 11.x Systems

According to step 11. d, install the other root domains (root-dom1 in the example) in the same way.

e. Assign the virtual I/O service to the root domain.

This item describes an example of the procedure for setting the virtual I/O service to the root domain.

The example shows that the entire physical disk of the root domain is lent as the virtual disk service (vds).

This example shows that the Ethernet card port is assigned to the virtual switch network device (vsw).

For details on the virtual I/O service, see the *Oracle VM Server for SPARC*

Log in to the root domain with the root privilege.

```
root-dom0 console login: root
Password:
....
```

To specify the back-end device (physical disk) to be assigned to the virtual disk service (vds), execute the format command to display the disk and its disk path.

A string in the format of cXtXXXXXXXXXXXXXXXXXdx indicates the disk, and a string beginning with "/pci@" indicates the device path.

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  0. c2t50000393A802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
    /pci@8100/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
  1. c3t50000393D8285226d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
    /pci@8200/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393d8285226,0
  2. c4t50000393A804B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
    /pci@3200/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
```

See the device path displayed in the above result and "A.6 SPARC M10-4S Device Paths" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*, and check the physical location of the disk. Then, determine the back-end disk (cXtXXXXXXXXXXXXXXXXdx) to be assigned to the virtual disk service.

The configuration example assigns two disks (c3t50000393D8285226d0 and c4t50000393A804B13Ed0) as back-ends of the virtual disks of the guest domains (guest0, guest1).

Next, to specify the network interface to be assigned to the virtual switch service (vsw), execute the dladm show-phys command to display the network interface name (LINK) associated with the location (LOC) of the PCIe slot.

From the displayed results, check the location of the network interface and determine the network interface to be assigned to the virtual switch service.

The configuration example assigns two network interfaces (net1 and net2) to the virtual switch associated with the virtual network of each guest domain.

```
# dladm show-phys -L
LINK      DEVICE      LOC
net0      igb0        BB#0-PCI#0
net1      igb1        BB#0-PCI#1
net2      igb2        BB#0-PCI#2
```



```
net3          igb3          BB#0-PCI#3
...
```

Execute the `ldm add-vdiskserver` command to add the virtual disk service to the root domain.

The following example adds the virtual disk service (`vds0`) to the root domain (`root-dom0`) with the `ldm add-vdiskserver` command.

```
# ldm add-vdiskserver vds0 root-dom0
```

Execute the `ldm add-vdiskserverdevice` command to export the back-end of the virtual disk from the root domain.

The following example executes the `ldm add-vdiskserverdevice` command to assign two back-end disks to `vds0` to export them.

```
# ldm add-vdiskserver vds0 root-dom0
# ldm add-vdiskserverdevice /dev/dsk/c3t50000393D8285226d0s2 vol10@
vds0
# ldm add-vdiskserverdevice /dev/dsk/c4t50000393A804B13Ed0s2 vol11@
vds0
```

Execute the `ldm add-vswitch` command to add the virtual switch to the root domain.

The following example adds `vsw0` and `vsw1` to add a virtual switch for each guest domain and assigns a different physical network interface (`net1`, `net2`) to each.

```
# ldm add-vswitch net-dev=net1 vsw0 root-dom0
# ldm add-vswitch net-dev=net2 vsw1 root-dom0
```

Similarly, add the virtual I/O service to the root domain (`root-dom1`) to which the I/O of SPARC M10-4S on the BB1 side was assigned.

As an example, assign the following virtual disk service name and virtual switch names to `root-dom1`. For details on the step, see 11. e.

Virtual disk service name: `vds1` (assign `vol10` and `vol11` as two back-end volume names.)

Virtual switch name: `vsw10`, `vsw11`

f. Configure a guest domain.

This item describes the procedure for configuring a guest domain.

Execute the `ldm add-domain` command to add the logical domain named `guest0`.

```
# ldm add-domain guest0
```

Execute the `ldm set-variable` command to change the OpenBoot PROM

environment variable "auto-boot?", which is designed to automatically boot Oracle Solaris, to "false" (disabled). By default, this setting is "true" (enabled). So, OpenBoot PROM tries to start Oracle Solaris automatically when Oracle Solaris is not installed. Changing this setting to disabled facilitates the work to be performed before Oracle Solaris installation.

```
# ldm set-variable auto-boot\?=false guest0
```

First, assign the CPU cores with the `ldm set-core` command and then assign the memory with the `ldm set-memory` command.

The following example assigns 16 CPU cores with the `ldm set-core` command and 32 GB of memory with the `ldm set-memory` command, according to the configuration example.

```
# ldm set-core 16 root-dom0
# ldm set-memory 32G root-dom0
```

Note - We recommend that you first configure the CPU cores with the `ldm set-core` command and then the memory with the `ldm set-memory` command. This facilitates the assignment of a continuous area of collective memory. This also makes it easy to easily satisfy the placement conditions for the CPU cores for which physical partition dynamic reconfiguration is possible and the memory placement conditions. For details on the CPU core and memory placement conditions, see ["Placement of CPU cores and memory" in "2.5.2 Considerations in System Operation for Dynamic Reconfiguration."](#)

Execute the `ldm set-vconsole` command to assign the virtual console (vcons).

The following example executes the `ldm set-vconsole` command to assign port number 5100 of the service (vcc0) of the virtual console terminal concentrator in the control domain to the virtual console.

```
# ldm set-vconsole service=vcc0 port=5100 guest0
```

Execute the `ldm add-vdisk` command to assign the virtual disk (vdisk).

The following example assigns two virtual disks (vdisk0, vdisk10) according to the configuration example. The back-end of each virtual disk specifies the back-ends (vds0 vol0, vds1 vol10) added to two root domains (root-dom0, root-dom1).

```
# ldm add-vdisk vdisk0 vol10@vds0 guest0
# ldm add-vdisk vdisk10 vol110@vds1 guest0
```

Execute the `ldm add-vnet` command to assign the virtual network device (vnet).

The following example assigns two virtual network devices (vnet0, vnet10) according to the configuration example. The virtual switch connected to each

virtual network device specifies virtual switches (vsw0, vsw10) added to two root domains (root-dom0, root-dom1).

```
# ldm add-vnet vnet0 vsw0 guest0
# ldm add-vnet vnet10 vsw10 guest0
```

Similarly, configure the other guest domain (in this example, guest1) as described in step 11. f.

The following devices are to be assigned based on the configuration example:

Virtual disk: vdisk1 (vol1@vds0 for the back-end), vdisk11 (vol11@vds1 for the back-end)

Virtual network: vnet1 (vsw1 for virtual switch), vnet11 (vsw11 for virtual switch)

Note - If multiple virtual disks or virtual network devices are assigned, record the value of the IDs assigned to the virtual device, from the `ldm list -l` command execution result. If a virtual device is added again and the ID is specified after being dynamically deleted, the ID is required so that the virtual device path remains unchanged.

g. Install Oracle Solaris in the guest domain.

This item describes the procedure for installing Oracle Solaris in the guest domain.

Execute the `ldm bind-domain` command to bind the guest domain and then start it by executing the `start-domain` command.

The following example shows the execution of the commands for starting guest0.

```
# ldm bind-domain guest0
# ldm start-domain guest0
LDom guest0 started
```

Execute the `ldm set-variable` command to change the OpenBoot PROM environment variable "auto-boot?", which is designed to automatically boot Oracle Solaris, to "true" (enabled). After Oracle Solaris is installed, execution of the `ldm start-domain` command also starts Oracle Solaris.

```
# ldm set-variable auto-boot\?=true guest0
```

Execute the `telnet` command to connect to the console of the guest domain.

The following example checks that the port number of the guest0 console is "5100" by executing the `ldm list-domain` command. It can also check that guest0 is stopped in the OpenBoot PROM status by connecting to localhost port number "5100" with the `telnet` command.

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active    -n-cv-   UART     16     14G       0.0%    8h 7m
guest0        active    -t----- 5100     32     32G       0.0%    20s
root-dom0     active    -n--v-   5000     24     24G       0.0%    43s
root-dom1     active    -n--v-   5001     24     24G       0.0%    20s
guest1        inactive  -----             32     32G
# telnet localhost 5100
....
{0} ok
```

For details on the versions and conditions of Oracle Solaris required for physical partition dynamic reconfiguration, see [Table 1-3](#) and [Table 1-4](#).

The following example executes the command to start Oracle Solaris 11 installation through the network.

```
{0} ok boot net:dhcp
....
```

For details on the installation, see the following documents, presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

- Oracle Solaris 10

Oracle Solaris 10 1/13 Installation Guide

- Oracle Solaris 11

Installing Oracle Solaris 11.x Systems

According to step 11. g, install the other guest domains (guest1 in the example) in the same way.

h. **Establish a redundant configuration for the virtual I/Os of a guest domain.**

The following describes an example of the procedure for establishing a redundant configuration for two virtual network interfaces (vnets) assigned to guest domain guest0, using IPMP. For details on the procedures for other redundant configurations, see the documentation about the software for the respective redundant configurations.

Log in to guest domain guest0.

In the example below, the ldm list-domain command is used to check the port number of the console of guest0, and the telnet command is used to connect to port number "5100."

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active    -n-cv-   UART     16     14G       0.0%    8h 7m
guest0        active    -n----- 5100     32     32G       0.0%    20s
guest1        active    -n----- 5101     32     32G       0.0%    19s
root-dom0     active    -n--v-   5000     24     24G       0.0%    43s
root-dom1     active    -n--v-   5001     24     24G       0.0%    20s
# telnet localhost 5100
....
```

```

guest0 console login: root
Password:
...
guest0#

```

Execute the `dladm` command to check that the virtual network devices are visible.

In the example below, it is possible to refer to virtual network devices as network interfaces `net0` and `net1`.

```

guest0# dladm show-phys

```

LINK	MEDIA	STATE	SPEED	DUPLEX	DEVICE
net0	Ethernet	up	0	unknown	vnet0
net1	Ethernet	up	0	unknown	vnet1

Execute the `ipadm show-if` command to check that `net0` and `net1` are not displayed.

```

guest0# ipadm show-if

```

IFNAME	CLASS	STATE	ACTIVE	OVER
lo0	loopback	ok	yes	--

Execute the `ipadm create-ip` command to create IP interfaces `net0` and `net1`, and then use the `ipadm show-if` command to check that they have been created normally.

```

guest0# ipadm create-ip net0
guest0# ipadm create-ip net1
guest0# ipadm show-if

```

IFNAME	CLASS	STATE	ACTIVE	OVER
lo0	loopback	ok	yes	--
net0	ip	down	no	--
net1	ip	down	no	--

Execute the `ipadm create-ipmp` command to create IPMP interface `ipmp0`, and then execute the `ipadm add-ipmp` command to add IP interfaces `net0` and `net1` to the IPMP group.

```

guest0# ipadm create-ipmp ipmp0
guest0# ipadm add-ipmp -i net0 -i net1 ipmp0

```

Execute the `ipadm create-addr` command to assign an IP address to IPMP interface `ipmp0`, and then use the `ipadm show-addr` command to check the setting. In the example below, a fixed IP address is assigned.

```

guest0# ipadm create-addr -T static -a local=xx.xx.xx.xx/24 ipmp0/v4
guest0# ipadm show-addr

```

ADDROBJ	TYPE	STATE	ADDR
lo0/v4	static	ok	127.0.0.1/8
ipmp0/v4	static	ok	xx.xx.xx.xx/24
lo0/v6	static	ok	:::1/128

Execute the `ipadm set-ifprop -p standby=on -m ip net1` command to set a standby interface, and use the `ipmpstat -i` command to check the IPMP configuration.

```

guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS      LINK      PROBE      STATE
net1       no      ipmp0      is----- up        disabled   ok
net0       yes     ipmp0      --mbM--   up        disabled   ok

```

Perform the same procedure for the other guest domains (guest1 in the example).

i. **Check the memory resource assignment status, and group the distributed memory blocks.**

First, see steps 11. h and 11. i in "[A.2.2 Example of the Physical Partition Configuration Procedure](#)," and group the distributed memory blocks.

The following example shows the result of executing the `ldm list-devices` command with memory distribution based on the configuration example.

You can confirm that there is only one memory block per logical domain and that there are not many distributed memory blocks of 256 to 512 MB.

```

# ldm list-devices -a memory
MEMORY
  PA                SIZE                BOUND
  0x700000000000    24G                root-dom1
  0x700600000000    8G
  0x720000000000    32G                guest0
  0x740000000000    32G                guest1
  0x760000800000    1272M             _sys_
  0x760050000000    24G                root-dom0
  0x760650000000    6912M
  0x780000000000    32G
  0x7a0000000000    32G
  0x7c0000000000    32G
  0x7e0000800000    1272M             _sys_
  0x7e0050000000    512M             _sys_
  0x7e0070000000    256M             _sys_
  0x7e0080000000    14G                primary
  0x7e0400000000    16G

```

j. **Check whether there are enough free memory blocks, and make an adjustment accordingly.**

This step describes how to check whether logical domain memory blocks can be moved to free memory blocks after releasing the SPARC M10-4Ss, after which an adjustment is made.

- i. **Check the use status of the continuous region of the memory (memory block).**
- Execute the prtdiag command to check the mapping between the memory physical addresses and SPARC M10-4Ss.

# prtdiag				
(Omitted)				
===== Physical Memory Configuration =====				
Segment Table:				

Base	Segment	Interleave	Bank	Contains
Address	Size	Factor	Size	Modules

0x7e0000000000	32 GB	4	8 GB	/BB0/CMUL/CMP0/MEM00A
(Omitted)				
0x7c0000000000	32 GB	4	8 GB	/BB0/CMUL/CMP1/MEM10A
(Omitted)				
0x7a0000000000	32 GB	4	8 GB	/BB0/CMUU/CMP0/MEM00A
(Omitted)				
0x780000000000	32 GB	4	8 GB	/BB0/CMUU/CMP1/MEM10A
(Omitted)				
0x760000000000	32 GB	4	8 GB	/BB1/CMUL/CMP0/MEM00A
(Omitted)				
0x740000000000	32 GB	4	8 GB	/BB1/CMUL/CMP1/MEM10A
(Omitted)				
0x720000000000	32 GB	4	8 GB	/BB1/CMUU/CMP0/MEM00A
(Omitted)				
0x700000000000	32 GB	4	8 GB	/BB1/CMUU/CMP1/MEM10A
(Omitted)				

The result of this example is rearranged in ascending order of physical addresses in memory. The following table lists the correspondence between the physical addresses and the SPARC M10-4S.

Table A-9 Example of Correspondence Between Physical Addresses and the SPARC M10-4S	
Base Address (Physical Address)	Building Block Configuration of SPARC M10-4S
0x700000000000 and after	BB1
0x720000000000 and after	BB1
0x740000000000 and after	BB1
0x760000000000 and after	BB1
0x780000000000 and after	BB0
0x7a0000000000 and after	BB0
0x7c0000000000 and after	BB0
0x7e0000000000 and after	BB0

The usage status of the memory blocks will be as follows after matching the result in step 11. i to the physical addresses confirmed with the prtdiag

command.

Table A-10 Example of Memory Block Use Statuses

SPARC M10-4S	Physical Address	Size	Logical Domain
BB1 (target for replacement)	0x700000000000	24 GB	root-dom1
	0x700600000000	8 GB	Unassigned
	0x720000000000	32 GB	guest0
	0x740000000000	32 GB	guest1
	0x760050000000	24 GB	root-dom0
	0x760650000000	6,912 MB	Unassigned
BB0	0x780000000000	32 GB	Unassigned
	0x7a0000000000	32 GB	Unassigned
	0x7c0000000000	32 GB	Unassigned
	0x7e0080000000	14 GB	primary
	0x7e0400000000	16 GB	Unassigned

ii. **Check the size and quantity of the movement source memory blocks.**

While referring to the check results of the memory block use status, check the memory block (hereafter called "source memory block") assigned to the SPARC M10-4S to be replaced.

In "Table A-10 Example of Memory Block Use Statuses," you can see that the number of memory blocks assigned to the logical domain on the BB1 side is 32 GB x 2 (assignment to guest0 and guest1) and 24 GB x 1 (root-dom0).

Note - A root domain (for example, root-dom1 if the BB1 is released) to which I/Os in SPARC M10-4S to be released are assigned is not considered as a target for movement because it is unbound and placed in the inactive state when SPARC M10-4S is released.

iii. **Check the empty memory blocks.**

Next, based on the check results obtained in step i, check the memory blocks (hereinafter referred to as "empty memory blocks") not assigned to the logical domain on the SPARC M10-4S that is not disconnected.

In "Table A-10 Example of Memory Block Use Statuses," you can see that the number of empty memory blocks is 32 GB x 3 and 16 GB x 1.

iv. **Check whether the memory block can be moved.**

Using the check results obtained in steps ii and iii, check whether the source memory block can be moved to the empty memory block.

This is possible if the size of the empty memory block is equal to or greater than that of the source memory block.

For example, in "[Table A-10 Example of Memory Block Use Statuses](#)," there are 32 GB x 3 empty memory blocks as the destinations for guest0 (32 GB), guest1 (32 GB), and root-dom0 (24 GB). So, you can see that the memory is placed such that BB1 can be disconnected. This is summarized in "[Table A-11 Destination Candidate Memory Blocks](#)."

If this is the case, skip to step 12.

Table A-11 Destination Candidate Memory Blocks

SPARC M10-4S	Size	Logical Domain	Destination Candidate
BB1 (target for replacement)	24 GB	root-dom0	-
	8 GB	Unassigned	-
	32 GB	guest0	32 GB in BB0
	32 GB	guest1	32 GB in BB0
	24 GB	root-dom0	32 GB in BB0
	6,912 MB	Unassigned	Delete this because there is a 32-GB guest1 memory on BB0.
BB0	32 GB	Unassigned	Moved here
	32 GB	Unassigned	Moved here
	32 GB	Unassigned	Moved here
	14 GB	primary	-
	16 GB	Unassigned	Excluded from destination candidates due to size insufficiency

v. If there is no destination available, reduce the memory assignment in the logical domain.

This step describes how to proceed when there are not enough free memory blocks as destination candidates.

If the conditions described in "[Placement of CPU cores and memory](#)" in "[2.5.2 Considerations in System Operation for Dynamic Reconfiguration](#)" are not satisfied, memory block movement will fail.

For example, in the configuration in "[Table A-12 Example of Placement of Memory Blocks With No Destinations](#)," the number of source memory blocks is 32 GB x 1 and 24 GB x 3. Meanwhile, the free memory blocks at the destination are 32 GB x 3 and 16 GB x 1.

Therefore, you can move one 32-GB memory block (guest0) and two 24-GB memory blocks (either two of guest1, guest2, and root-dom0) in the BB1.

However, the number of remaining empty destination memory blocks is 16 GB x 1 and 8 GB x 2, obtained after the 24-GB memory block has been moved to a 32-GB memory block. So, any of guest1, guest2, and root-dom0, to which a 24-GB memory block is assigned, cannot be moved.

Table A-12 Example of Placement of Memory Blocks With No Destinations

SPARC M10-4S	Size	Logical Domain	Destination Candidate
BB1 (target for replacement)	24 GB	guest2	May not be movable
	8 GB	root-dom1	-
	32 GB	guest0	32 GB in BB0
	32 GB	guest1	May not be movable
	24 GB	root-dom0	May not be movable
	6,912 MB	Unassigned	-
BB0	32 GB	Unassigned	Moved here
	32 GB	Unassigned	Any one of guest1, guest2, and root-dom0 (24 GB) moves here, with 8 GB remaining.
	32 GB	Unassigned	Any one of guest1, guest2, and root-dom0 (24 GB) moves here, with 8 GB remaining.
	14 GB	primary	-
	16 GB	Unassigned	Excluded from destination candidates due to size insufficiency

In this case, you need to reduce the size of a possibly unmovable memory block in the logical domain to be equal to or less than the memory block size at the destination.

In the above example, you must change the memory size for either one of logical domains guest1, guest2, or root-dom0 from 24 GB to 16 GB or less.

The following example shows how to change the memory size for guest domain guest2 to 16 GB or less.

Execute the `ldm list-domain` command to check the logical domain status.

# <code>ldm list-domain</code>							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	16	14G	0.5%	40m
guest0	active	-n----	5100	32	32G	0.0%	26m
guest1	active	-n----	5101	32	24G	0.0%	26m
guest2	active	-n----	5102	12	24G	0.1%	1m
root-dom0	active	-n--v-	5000	24	24G	0.0%	26m
root-dom1	active	-n--v-	5001	16	8G	0.0%	26m

Execute the `ldm stop-domain` command to stop the domain, and then execute the `ldm unbind-domain` command to place it in the inactive state.

The following example executes the command to place guest domain guest2 in the inactive state.

```
# ldm stop-domain guest2
ldom guest2 stopped
# ldm unbind-domain guest2
```

Note - Before stopping a domain with a virtual I/O service configuration, you must first stop the guest domain to which the corresponding virtual I/O devices are assigned by using the ldm stop-domain command.

If the version of Oracle VM Server for SPARC is earlier than 3.2, execute the ldm set-memory command to reconfigure the target logical domain memory.

Set up the configuration such that the size of guest domain guest2 is 16 GB (16,384 MB) or less and a multiple of (the number of CPU cores x 256 MB) because 6 cores (12vcpu) are assigned to the domain.

$16384 / (6 \times 256) = 10.66$, rounded down to 10

Therefore, reconfigure the memory size to $(6 \times 256 \text{ MB}) \times 10 = 15,360 \text{ MB}$ (15 GB) for guest domain guest2.

The following example executes the ldm set-memory command to reconfigure the memory size to 15 GB.

```
# ldm set-memory 15G guest2
```

Execute the ldm list-domain command to confirm that the memory size has been changed as reconfigured.

The following example executes the ldm list-domain command to show that the memory size of guest domain guest2 is reconfigured to 15 GB.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	16	14G	0.5%	40m
guest0	active	-n----	5100	32	32G	0.0%	26m
guest1	active	-n----	5101	32	24G	0.0%	26m
root-dom0	active	-n--v-	5000	24	24G	0.0%	26m
root-dom1	active	-n--v-	5001	16	8G	0.0%	26m
guest2	inactive	-----		12	15G		

Execute the ldm bind-domain and ldm start-domain commands to bind and start the logical domain.

The following example executes the ldm bind-domain and ldm start-domain commands to start guest domain guest2.

```
# ldm bind-domain guest2
# ldm start-domain guest2
LDom guest2 started
```

Execute the `ldm list-domain` command, and then confirm that the logical domain has started.

The following example executes the `ldm list-domain` command to show that guest domain `guest2` has started with the reconfigured memory size.

Confirm that `[STATE]` shows "active" and that the second character from the left in the string of `[FLAGS]` is "n".

```
# ldm bind-domain guest2 root-dom0
-domain
NAME                STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary             active    -n-cv-   UART     16     14G       0.5%    44m
guest0              active    -n----   5100     32     32G       0.0%    30m
guest1              active    -n----   5101     32     24G       0.0%    30m
guest2              active    -n----   5102     12     15G       0.1%    1m
root-dom0           active    -n----   5000     24     24G       0.0%    30m
root-dom1           active    -n----   5001     16     8G        0.0%    30m
```

Execute the `ldm list-devices -a memory` command to check whether the memory block placement satisfies the conditions described in "[Placement of CPU cores and memory](#)" in "[2.5.2 Considerations in System Operation for Dynamic Reconfiguration](#)."

The following example executes the `ldm list-devices -a memory` command.

```
# ldm ls-devices -a mem
MEMORY
PA                SIZE                BOUND
(BB1)
0x700000000000    32G                guest0 →Move to 0x780000000000
(32GB)
0x720000000000    24G                root-dom0 →Move to 0x7a0000000000
(32GB)
0x720600000000    8G
0x740000000000    8G                root-dom1 (not to be moved)
0x740200000000    24G
0x760000800000    1272M              _sys_
0x760050000000    24G                guest1 →Move to 0x7c0000000000
(32GB)
0x760650000000    6912M
(BB0)
0x780000000000    32G
0x7a0000000000    32G
0x7c0000000000    32G
0x7e0000800000    1272M              _sys_
0x7e0050000000    512M              _sys_
0x7e0070000000    256M              _sys_
0x7e0080000000    14G                primary
0x7e0400000000    15G                guest2
0x7e07c0000000    1G
```

12. Save the configured local domain configuration information in the XSCF.

Execute the `ldm set-spcfg` command to save the configured information.

The following example checks the saved configuration information and then saves it with the same name as that of the existing configuration information. Execute the `ldm list-spconfig` command to check the current configuration information.

```
# ldm list-spconfig
factory-default
ldm-set1 [next poweron]
```

Execute the `ldm remove-spconfig` command to delete the configuration information to be overwritten.

```
# ldm remove-spconfig ldm-set1
```

Execute the `ldm add-spconfig` command to re-save the configuration information.

```
# ldm add-spconfig ldm-set1
```

Execute the `ldm list-spconfig` command to check that the saved configuration information has become `[current]`.

```
# ldm list-spconfig
factory-default
ldm-set1 [current]
```

13. Save the configured logical domain configuration information to an XML file.

To guard against the configuration information saved to the XSCF being unusable, save the configuration information to an XML file. It is recommended that the XML file be saved to a different media.

The following describes the example procedure.

Execute the `ldm list-domain` command to check that all the logical domains are active.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	16	14G	0.0%	10h 7m
guest0	active	-n----	5100	32	32G	0.0%	2h 20s
guest1	active	-n----	5101	32	32G	0.0%	2h 5s
root-dom0	active	-n----	5000	24	24G	0.0%	2h 43s
root-dom1	active	-n----	5001	24	24G	0.0%	2h 20s

Execute the `ldm list-constraints` command to save the configuration information to an XML file.

```
# ldm list-constraints -x > /ldm-set1.xml
```

A.3.3

Example of the Active Replacement Procedure (for Oracle VM Server for SPARC 3.1.x)

This section describes an example of the procedure for actively replacing the BB#01 SPARC M10-4S using PPAR DR for the 2BB configuration system described in [Figure A-3 Example of the 2BB Configuration for Operational Continuity \(With Free Resources\)](#).

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.

You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

1. **Log in to the master XSCF.**

Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF.

If you have logged in to a standby XSCF, log out and then log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

2. **Execute the `showhardconf` command to check that [Status] of the XSCF in the SPARC M10-4S to be replaced is "Normal."**

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Locked;
+ System_Power:On; System_Phase:Cabinet Power On;
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP1236052K ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
+ Freq:3.000 GHz; Type:0x10;
+ Core:16; Strand:2;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
+ FRU-Part-Number: CA20393-B50X A2 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
```

```
+ Memory_Size:128 GB; Type: A ;
```

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.

You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

3. **Execute the `showbbstatus` command to confirm that the XSCF in the SPARC M10-4S to be replaced is not the master XSCF.**

```
XSCF> showbbstatus
BB#00 (Master)
```

If the SPARC M10-4S to be replaced is the master XSCF, execute the `switchscf` command to switch the XSCF.

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

Note - Confirm that the XSCF has been switched and rebooted before you release the SPARC M10-4S.

4. **Execute the `console` command to connect to the console of the control domain and then log in to it.**

```
XSCF> console -p 0
```

5. **Check the operation status and resource usage status of the logical domain.**
 - a. **Execute the `ldm list-domain` command to check the operation status of the logical domain.**

To check the logical domain operation status, check the [STATE] and [FLAGS] combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

The following example shows that the control domain, two root domains, and two guest domains are operating.

Check whether all domains are in "active", which indicates that Oracle Solaris is in operating state, or "inactive" state. If there is a domain with OpenBoot

PROM status or bound status, the dynamic reconfiguration of the physical partition may fail.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	16	14G	0.0%	10h 7m
guest0	active	-n----	5100	32	32G	0.0%	2h 20s
guest1	active	-n----	5101	32	32G	0.0%	2h 5s
root-dom0	active	-n--v-	5000	24	24G	0.0%	2h 43s
root-dom1	active	-n--v-	5001	24	24G	0.0%	2h 20s

- b. **Execute the `ldm list-devices` command with the `-a` option specified to check the resource usage status.**

In the following example, the `-a` option is specified to display all resources bound to the logical domain and all resources that are not bound.

```
# ldm list-devices -a
```

CORE		
ID	%FREE	CPUSET
0	0	(0, 1)
4	0	(8, 9)
8	0	(16, 17)
(Omitted)		
944	0	(1888, 1889)
948	0	(1896, 1897)
952	0	(1904, 1905)
956	0	(1912, 1913)
VCPU		
PID	%FREE	PM
0	0	no
1	0	no
8	0	no
9	0	no
(Omitted)		
1904	0	no
1905	0	no
1912	0	no
1913	0	no
(Omitted)		

6. **Release the redundant configuration of the system volume and I/O devices in the control domain.**

This step describes how to release the I/O devices of the SPARC M10-4S to be replaced and which are used in the control domain in order enable the release of BB-ID#01 SPARC M10-4S, If you are using other redundant configuration software, see the documentation for the software for that redundant configuration for details on how to cancel the configuration.

- a. **Cancel the redundant configuration of the system volume in the control domain.**

The following example describes how to cancel the ZFS mirroring function

for the system volume in the control domain.

Execute the `zpool status` command in the control domain to check the mirroring configuration status.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                                ONLINE     0     0     0
        mirror-0
            c2t50000393E802CCE2d0s0    ONLINE     0     0     0
            c3t50000393A803B13Ed0s0    ONLINE     0     0     0
errors: No known data errors
```

Execute the `zpool detach` command to release the disk from the mirroring configuration.

```
# zpool detach rpool c3t50000393A803B13Ed0
```

Execute the `zpool status` command to confirm that the mirroring configuration has been canceled.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                                ONLINE     0     0     0
        mirror-0
            c2t50000393E802CCE2d0s0    ONLINE     0     0     0
errors: No known data errors
```

If you are using other devices in BB#01, remove the redundant configuration or stop using those devices. For details on how to cancel a redundant configuration or stop using the devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

b. Delete the I/O configuration of the control domain.

From among the physical I/O devices assigned to the control domain, delete the root complexes in BB#01 through delayed reconfiguration.

First, place the control domain in delayed reconfiguration mode.

```
# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
```

will also take effect.

Execute the `ldm list-io` command to check the root complexes assigned to the primary.

The following example shows that the root complexes with BB1 devices are PCIE8 and PCIE12.

```
# ldm list-io | grep primary
PCIE0          BUS      PCIE0      primary  IOV
PCIE4          BUS      PCIE4      primary  IOV
PCIE8          BUS      PCIE8      primary  IOV
PCIE12         BUS      PCIE12     primary  IOV
/BB0/CMUL/NET0 PCIE    PCIE0      primary  OCC
/BB0/CMUL/SASHBA PCIE    PCIE0      primary  OCC
/BB0/CMUL/NET2  PCIE    PCIE4      primary  OCC
/BB1/CMUL/NET0  PCIE    PCIE8      primary  OCC
/BB1/CMUL/SASHBA PCIE    PCIE8      primary  OCC
/BB1/CMUL/NET2  PCIE    PCIE12     primary  OCC
```

Execute the `ldm remove-io` command to delete PCIE8 and PCIE12 from the primary, and then restart Oracle Solaris.

```
# ldm remove-io PCIE8 primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----

# ldm remove-io PCIE12 primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----

# shutdown -i6 -g0 -y
....
```

Once Oracle Solaris has started, execute the `ldm list-io` command to confirm that the root complexes in BB#01 have been deleted from the control domain.

```
# ldm list-io | grep primary
PCIE0          BUS      PCIE0      primary  IOV
PCIE4          BUS      PCIE4      primary  IOV
/BB0/CMUL/NET0 PCIE    PCIE0      primary  OCC
/BB0/CMUL/SASHBA PCIE    PCIE0      primary  OCC
/BB0/CMUL/NET2  PCIE    PCIE4      primary  OCC
```

c. **Cancel the redundant configuration of the virtual I/O devices assigned to a guest domain.**

To first shut down the root domain (root-dom1) to which the root complexes in BB#01 are assigned, log in to each guest domain, and then cancel the redundant configuration of the virtual I/O device from root-dom1.

For details on how to use the redundant configuration software, see the documentation about the software for that redundant configuration.

In the following example, a virtual network device (vnet1) is canceled from the IPMP configuration. For details on the command, see the manual for Oracle Solaris.

Log in to the guest domain (guest0).

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active     -n-cv-   UART    64      56G       0.0%    4h 17m
guest0        active     -n----   5100    64      64G       0.0%    1h 13m
guest1        active     -n----   5101    64      64G       0.0%    1h 4m
root-dom0     active     -n--v-   5000    32      32G       0.0%    1h 47m
root-dom1     active     -n--v-   5001    32      32G       0.0%    1h 19m
# telnet localhost 5100
....
guest0#
```

Execute the `dladm show-phys` command to check the mapping between the virtual network interface (vnet1) and the network interface name (net1).

```
guest0# dladm show-phys
LINK          MEDIA          STATE    SPEED    DUPLEX    DEVICE
net0          Ethernet      up        0        unknown  vnet0
net1          Ethernet      up        0        unknown  vnet1
```

Execute the `ipmpstat -i` command to check the configuration information for the network interfaces configuring IPMP.

```
guest0# ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS    LINK    PROBE    STATE
net0       yes    ipmp0      -smbM--  up      disabled ok
net1       no     ipmp0      is----- up      disabled ok
```

Execute the `if_mpadm -d net1` command to release net1 from the IPMP group, and then execute the `ipmpstat -i` command to confirm that it has been released. The following example confirms that STATE is offline.

```
guest0# if_mpadm -d net1
guest0# ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS    LINK    PROBE    STATE
net0       yes    ipmp0      -smbM--  up      disabled ok
net1       no     ipmp0      -s---d-  up      disabled offline
```

Execute the `ipadm delete-ip` command to delete net1.

```
guest0# ipadm delete-ip net1
```

Similarly, perform the same release processing for the guest domain (guest1).

d. **Remove the virtual I/O devices assigned from the root domain to be stopped.**

Execute the ldm remove-vdisk and ldm remove-vnet commands to remove the assigned virtual disk (vdisk) and virtual network device (vnet) from the root domain to be stopped according to the following step.

The following example shows the execution of the commands for removing the virtual disk (vdisk10) and virtual network device (vnet10) that use the virtual I/O service of the BB#01 root domain (root-dom1).

Perform the same deletion for the guest domain (guest1).

```
# ldm remove-vdisk vdisk10 guest0
# ldm remove-vnet vnet10 guest0
```

7. **Check the resource usage status of the I/O devices, and then cancel all the I/O devices in the SPARC M10-4S to be replaced.**

a. **Check the logical domain to which the root complexes in the SPARC M10-4S to be released are assigned.**

Execute the ldm list-io command to check the logical domain to which the root complexes in BB#01 are assigned.

The following example shows that only root-dom1 has PCIe endpoints starting with "/BB1/." You can see that the PCIe endpoint root complexes (BUS) PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, and PCIE15 are assigned to root-dom1.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV
....				
/BB1/CMUL/NET0	PCIE	PCIE8		UNK
/BB1/CMUL/SASHBA	PCIE	PCIE8		UNK
/BB1/PCI0	PCIE	PCIE9	root-dom1	OCC
/BB1/PCI3	PCIE	PCIE10	root-dom1	OCC
/BB1/PCI4	PCIE	PCIE10	root-dom1	OCC

/BB1/PCI7	PCIE	PCIE11	root-dom1OCC
/BB1/PCI8	PCIE	PCIE11	root-dom1OCC
/BB1/CMUL/NET2	PCIE	PCIE12	UNK
/BB1/PCI1	PCIE	PCIE13	root-dom1OCC
/BB1/PCI2	PCIE	PCIE13	root-dom1OCC
/BB1/PCI5	PCIE	PCIE14	root-dom1OCC
/BB1/PCI6	PCIE	PCIE14	root-dom1OCC
/BB1/PCI9	PCIE	PCIE15	root-dom1OCC
/BB1/PCI10	PCIE	PCIE15	root-dom1OCC

- b. **Stop the root domain to which the root complexes in SPARC M10-4S to be released are assigned and then release the SPARC M10-4S.**

The following example executes the ldm stop-domain and ldm unbind-domain commands to release the root domain (root-dom1) and shows that the root domain is in the inactive state.

```
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain root-dom1
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	16	14G	0.2%	4h 59m
guest0	active	-n----	5100	32	32G	0.0%	1h 55m
guest1	active	-n----	5101	32	32G	0.0%	1h 46m
root-dom0	active	-n--v-	5000	24	24G	0.0%	2h 29m
root-dom1	inactive	-----		24	24G		

- c. **Confirm that all of the I/O devices in the SPARC M10-4S to be replaced have been canceled.**

Execute the ldm list-io command to confirm that all the I/O devices have been released.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9		
PCIE10	BUS	PCIE10		
PCIE11	BUS	PCIE11		
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13		
PCIE14	BUS	PCIE14		
PCIE15	BUS	PCIE15		

8. **Release the system board (PSB<BB>) of the SPARC M10-4S from the physical partition.**
 - a. **Execute the deleteboard -c disconnect command to release the PSB from the physical partition.**

```
XSCF> deleteboard -c disconnect 01-0
PSB#01-0 will be unconfigured from PPAR immediately. Continue?
[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.
```

- b. **Execute the showresult command to check the exit status of the deleteboard command that was just executed.**

An end value of 0 indicates the normal termination of the deleteboard command.

If the end value is other than 0 or if an error message is displayed upon executing the deleteboard command, it indicates abnormal termination of the deleteboard command. By referring to "C.1.2 deleteboard" based on the error message, identify the error and then take corrective action.

```
XSCF> showresult
0
```

- c. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the SPARC M10-4S to be replaced is in the "Assigned" state and that the [Pwr], [Conn], and [Conf] columns all show "n."

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test  Fault
-----
00-0  00(00)      Assigned    y    y    y    Passed Normal
01-0  00(01)      Assigned    n    n    n    Passed Normal
```

9. **Execute the replacefru command to replace the SPARC M10-4S.**

```
XSCF> replacefru
```

Note - For details on the replacement of SPARC M10-4Ss by using the replacefru command, see "5.8 Releasing a SPARC M10-4/M10-4S FRU from the System with the replacefru Command" and "6.2 Incorporating a SPARC M10-4/M10-4S FRU into the System with the replacefru Command" in the *Fujitsu M10-4/Fujitsu M10-4S/SPARC M10-4/SPARC M10-4S Service Manual*.

10. **Incorporate the PSB into the physical partition.**

a. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the SPARC M10-4S to be replaced is in the Assigned state and that the [Pwr], [Conn], and [Conf] columns all show "n."

XSCF> showboards -p 0							
PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	00(01)	Assigned	n	n	n	Passed	Normal

b. **Execute the addboard -c configure command to incorporate the PSB into the physical partition.**

To recover the original logical domain configuration, execute the addboard -c configure command with the -m bind=resource option specified.

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

Note - If an error message appears during execution of the addboard command, see "[C.1.1 addboard](#)," and then identify the error and take corrective action.

c. **Execute the showresult command to check the exit status of the addboard command that was just executed.**

An end value of 0 indicates the normal termination of the addboard command.

If the end value is other than 0 or if an error message is displayed upon executing the addboard command, it indicates abnormal termination of the addboard command. By referring to "[C.1.1 addboard](#)" based on the error message, identify the error and then take corrective action.

```
XSCF> showresult
0
```

d. **Execute the showboards command to check the PSB status.**

Confirm that both of the [Conn] and [Conf] columns show "y" after the PSB in the replaced SPARC M10-4S has been successfully incorporated.

```
XSCF> showboards -p 0
```

PSB	PPAR-ID (LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	00(01)	Assigned	y	y	y	Passed	Normal

11. **Check the logical domain operation status.**

a. **Execute the console command to connect to the console of the control domain and then log in to it.**

```
XSCF> console -p 0
```

b. **Execute the ldm list-domain command to confirm that the logical domain operation status has not changed after the addition of the SPARC M10-4S PSB (BB).**

To check the logical domain operation status, check the [STATE] and [FLAGS] combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	16	14G	0.0%	10h 30m
guest0	active	-n----	5100	32	32G	0.0%	2h 50s
guest1	active	-n----	5101	32	32G	0.0%	2h 30s
root-dom0	active	-n--v-	5000	24	24G	0.0%	3h 20s
root-dom1	inactive	-----		24	24G		

12. **Restart the use of the I/O devices.**

a. **Reassign root complexes.**

Execute the ldm bind-domain and ldm start-domain commands to start the root domain in the unbind state to which root complexes in the replaced SPARC M10-4S were assigned.

The following example starts the root domain (root-dom1) in the unbind state, and confirms that it has started.


```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	16	16G	0.2%	3h 8m
guest0	active	-n----	5100	32	32G	0.0%	3h 8m
guest1	active	-n----	5101	32	32G	0.0%	3h 8m
root-dom0	active	-n--v-	5000	24	24G	0.0%	3h 8m
root-dom1	active	-n--v-	5001	24	24G	7.3%	8s

Execute the `ldm list-io` command to confirm that the physical I/O devices are assigned to the root domain that has just started.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV
....				
/BB1/CMUL/NET0	PCIE	PCIE8		UNK
/BB1/CMUL/SASHBA	PCIE	PCIE8		UNK
/BB1/PCI0	PCIE	PCIE9	root-dom1OCC	
/BB1/PCI3	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI4	PCIE	PCIE10	root-dom1OCC	
/BB1/PCI7	PCIE	PCIE11	root-dom1OCC	
/BB1/PCI8	PCIE	PCIE11	root-dom1OCC	
/BB1/CMUL/NET2	PCIE	PCIE12		UNK
/BB1/PCI1	PCIE	PCIE13	root-dom1OCC	
/BB1/PCI2	PCIE	PCIE13	root-dom1OCC	
/BB1/PCI5	PCIE	PCIE14	root-dom1OCC	
/BB1/PCI6	PCIE	PCIE14	root-dom1OCC	
/BB1/PCI9	PCIE	PCIE15	root-dom1OCC	
/BB1/PCI10	PCIE	PCIE15	root-dom1OCC	

b. Add the virtual I/O device from the root domain to the guest domain.

Execute the `ldm add-vdisk` and `ldm add-vnet` commands to add, to each guest domain, the virtual disk (vdisk) and virtual network device (vnet) supported for the virtual I/O service of the started root domain.

The following example indicates the execution of the command for adding the virtual disk (vdisk10) and virtual network device (vnet10) that use the virtual I/O service of BB#01 root domain (root-dom1).

Perform the same addition for the guest domain (guest1).

```
# ldm add-vdisk id=1 vdisk10 vol10@vds1 guest0
# ldm add-vnet id=1 vnet10 vsw10 guest0
```

Note - To add the virtual I/O device again, it is necessary to specify the ID assigned beforehand. You can check the ID from the result of execution of the `ldm list -l` command in the status used before the virtual I/O device is removed.

c. Incorporate the virtual I/O devices assigned to a guest domain into the redundant configuration.

Once the root domain (root-dom1) to which root complexes in BB#1 have been assigned has started, the virtual I/O device services corresponding to each guest domain are also started.

Log in to each guest domain, and then incorporate the virtual I/O devices from root-dom1 that were previously canceled into the redundant configuration. For details on how to use the redundant configuration software, see the documentation for the software for that redundant configuration.

The following describes an example of incorporating a virtual network device (vnet1) into the IPMP configuration. For details on the commands, see the manual for Oracle Solaris.

First, log in to the guest domain (guest0).

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active    -n-cv-   UART     16     14G       0.0%    4h 17m
guest0        active    -n----   5100     32     32G       0.0%    1h 13m
guest1        active    -n----   5101     32     32G       0.0%    1h 4m
root-dom0     active    -n----   5000     24     24G       0.0%    1h 47m
root-dom1     active    -n----   5001     24     24G       0.0%    1h 19m
# telnet localhost 5100
....
guest0#
```

Execute the `dladm show-phys` command to check the mapping between the virtual network interface (vnet1) and the network interface name (net1).

```
guest0# dladm show-phys
LINK          MEDIA          STATE    SPEED    DUPLEX    DEVICE
net0          Ethernet       up        0        unknown   vnet0
net1          Ethernet       up        0        unknown   vnet1
```

Execute the `ipadm create-ip net1` command, the `ipadm set-ifprop -p standby=on -m ip net1` command, and the `ipadm add-ipmp -i net1 ipmp0` command to register `net1` as a standby device of `ipmp0`.

```
guest0# ipadm create-ip net1
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipadm add-ipmp -i net1 ipmp0
```

Execute the `ipmpstat -i` command to confirm that STATE of the network interface configuring IPMP indicates ok.

Perform the same step for the other guest domain (`guest1`).

```
guest0# ipmpstat -i
INTERFACE ACTIVE GROUP FLAGS LINK PROBE STATE
net0 yes ipmp0 -smbM-- up disabled ok
net1 no ipmp0 -s---d- up disabled ok
```

13. **Restore the system volume and I/O devices on the control domain to a redundant configuration.**

a. **Add the root complex configuration for the control domain.**

By means of delayed configuration, add the root complexes in BB#01 that were previously removed from the control domain.

First, place the control domain in delayed reconfiguration mode.

```
# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
```

Execute the `ldm list-io` command to check the unassigned root complexes.

The following example shows that root complexes with BB1 devices, PCIE8 and PCIE12 are not assigned.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV

PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
(Omitted)				

Execute the `ldm add-io` command to add PCIE8 and PCIE12 to primary, and then restart Oracle Solaris.

```
# ldm add-io PCIE8 primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
# ldm add-io PCIE12 primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
# shutdown -i6 -g0 -y
....
```

Once Oracle Solaris has started, execute the `ldm list-io` command to confirm that the root complexes in BB#01 have been added to the control domain.

```
# ldm list-io | grep primary
PCIE0          BUS      PCIE0      primary    IOV
PCIE4          BUS      PCIE4      primary    IOV
PCIE8          BUS      PCIE8      primary
PCIE12         BUS      PCIE12     root-dom1
/BB0/CMUL/NET0 PCIE     PCIE0      primary    OCC
/BB0/CMUL/SASHBA PCIE     PCIE0      primary    OCC
/BB0/CMUL/NET2 PCIE     PCIE4      primary    OCC
```

b. Place the system volume in the control domain in a redundant configuration.

Execute the `zpool status` command in the control domain to check the mirroring configuration status.

The following example describes how to configure the ZFS mirroring function for the system volume in the control domain.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
   NAME                                STATE     READ WRITE CKSUM
   rpool                               ONLINE      0     0     0
       c2t50000393E802CCE2d0s0        ONLINE      0     0     0
errors: No known data errors
```

Execute the `zpool attach` command to incorporate the disks into a mirroring configuration.

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

Execute the `zpool status` command, and then confirm that the mirroring configuration has been established.

Use the `zpool status` command to confirm whether synchronization processing (resilver) is completed.

The following shows an example of the display during synchronization processing.

```
# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Mon Jan 27 15:55:47 2014
      21.1G scanned out of 70.6G at 120M/s, 0h7m to go
      21.0G resilvered, 29.84% done
config:
      NAME                                STATE      READ WRITE CKSUM
      rpool                                DEGRADED   0      0      0
        mirror-0                          DEGRADED   0      0      0
          c2t50000393E802CCE2d0s0         ONLINE     0      0      0
          c3t50000393A803B13Ed0s0         DEGRADED   0      0      0 (resilvering)
errors: No known data errors
```

Once synchronization processing is complete, the displayed screen will be as follows:

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
      NAME                                STATE      READ WRITE CKSUM
      rpool                                ONLINE     0      0      0
        mirror-0                          ONLINE     0      0      0
          c2t50000393E802CCE2d0s0         ONLINE     0      0      0
          c3t50000393A803B13Ed0s0         ONLINE     0      0      0
errors: No known data errors
```

If you are using other devices in BB#01, establish a redundant configuration or resume the use of the devices. For details on how to establish a redundant configuration or resume the use of devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

A.3.4 Example of the Active Replacement Procedure (When Dynamic Assignment of the PCIe Bus is Available)

This section describes an example of the procedure for actively replacing BB#01 using PPAR DR for the 2BB configuration system described in "[Figure A-3 Example of the 2BB Configuration for Operational Continuity \(With Free Resources\)](#)." The example is for an environment where the PCIe bus is assignable dynamically (XCP 2240 or later with Oracle VM Server for SPARC 3.2 or later and the root domain with Oracle Solaris 11.2 SRU11.2.8.4.0 or later).

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.
You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

1. Log in to the master XSCF.

Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF.

If you have logged in to a standby XSCF, log out and then log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

2. Execute the `showhardconf` command to check that [Status] of the XSCF in the SPARC M10-4S to be replaced is "Normal."

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Locked;
+ System_Power:On; System_Phase:Cabinet Power On;
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP1236052K ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
+ Freq:3.000 GHz; Type:0x10;
+ Core:16; Strand:2;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
+ FRU-Part-Number: CA20393-B50X A2 ;
```

```
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
:
```

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.

You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

3. **Execute the `showbbstatus` command to confirm that the XSCF in the SPARC M10-4S to be replaced is not the master XSCF.**

```
XSCF> showbbstatus
BB#00 (Master)
```

If the SPARC M10-4S to be replaced is the master XSCF, execute the `switchscf` command to switch the XSCF.

```
XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y
```

Note - Confirm that the XSCF has been switched and rebooted before you release the system board.

4. **Execute the console command to connect to the console of the control domain and then log in to it.**

```
XSCF> console -p 0
```

5. **Check the operation status and resource usage status of the logical domain.**
 - a. **Execute the `ldm list-domain` command to check the operation status of the logical domain.**

To check the logical domain operation status, confirm that [STATE] indicates "active" and check the second character from the left of the string in FLAGS. The displayed characters and their meanings are as follows:

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

The following example shows that the control domain, two root domains, and

two guest domains are operating.

Check whether all domains are in "active", which indicates that Oracle Solaris is in operating state, or "inactive" state. If there is a domain with OpenBoot PROM status or bound status, the dynamic reconfiguration of the physical partition may fail.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	16	14G	0.0%	10h 7m
guest0	active	-n----	5100	32	32G	0.0%	2h 20s
guest1	active	-n----	5101	32	32G	0.0%	2h 5s
root-dom0	active	-n--v-	5000	24	24G	0.0%	2h 43s
root-dom1	active	-n--v-	5001	24	24G	0.0%	2h 20s

b. Execute the ldm list-devices command with the -a option specified to check the resource usage status.

In the following example, the -a option is specified to display all resources bound to the logical domain and all resources that are not bound.

```
# ldm list-devices -a
```

CORE		
ID	%FREE	CPUSET
0	0	(0, 1)
4	0	(8, 9)
8	0	(16, 17)
(Omitted)		
944	0	(1888, 1889)
948	0	(1896, 1897)
952	0	(1904, 1905)
956	0	(1912, 1913)
VCPU		
PID	%FREE	PM
0	0	no
1	0	no
8	0	no
9	0	no
(Omitted)		
1904	0	no
1905	0	no
1912	0	no
1913	0	no
(Omitted)		

6. Release the redundant configuration of the system volume and I/O devices in the control domain.

To enable the release of building block BB-ID#01, this step describes how to release the I/O devices of the SPARC M10-4S to be replaced and which are used in the control domain. If you are using other redundant configuration software, see the documentation for the software for that redundant configuration for details on how to cancel the configuration.

a. **Cancel the redundant configuration of the system volume in the control domain.**

The following example describes how to cancel the ZFS mirroring function for the system volume in the control domain.

Execute the `zpool status` command in the control domain to check the mirroring configuration status.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
  NAME                                STATE      READ WRITE CKSUM
  rpool                               ONLINE    0     0     0
    mirror-0                          ONLINE    0     0     0
      c2t50000393E802CCE2d0s0        ONLINE    0     0     0
      c3t50000393A803B13Ed0s0        ONLINE    0     0     0
errors: No known data errors
```

Execute the `zpool detach` command to release the disk from the mirroring configuration.

```
# zpool detach rpool c3t50000393A803B13Ed0
```

Execute the `zpool status` command to confirm that the mirroring configuration has been canceled.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
  NAME                                STATE      READ WRITE CKSUM
  rpool                               ONLINE    0     0     0
    mirror-0                          ONLINE    0     0     0
      c2t50000393E802CCE2d0s0        ONLINE    0     0     0
errors: No known data errors
```

If you are using other devices in BB#01, remove the redundant configuration or stop using those devices. For details on how to cancel a redundant configuration or stop using the devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

b. **Delete the I/O configuration of the control domain.**

From among the physical I/O devices assigned to the control domain, delete the root complexes in BB#01.

Execute the `ldm list-io` command to check the root complexes assigned to the primary.

The following example shows that the root complexes with BB1 devices are

PCIE8 and PCIE12.

```
# ldm list-io | grep primary
```

PCIE0	BUS	PCIE0	primary	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE8	BUS	PCIE8	primary	IOV
PCIE12	BUS	PCIE12	primary	IOV
/BB0/CMUL/NET0	PCIE	PCIE0	primary	OCC
/BB0/CMUL/SASHBA	PCIE	PCIE0	primary	OCC
/BB0/CMUL/NET2	PCIE	PCIE4	primary	OCC
/BB1/CMUL/NET0	PCIE	PCIE8	primary	OCC
/BB1/CMUL/SASHBA	PCIE	PCIE8	primary	OCC
/BB1/CMUL/NET2	PCIE	PCIE12	primary	OCC

Execute the ldm remove-io command to delete PCIE8 and PCIE12 from the primary.

```
# ldm remove-io PCIE8 primary
# ldm remove-io PCIE12 primary
```

Execute the ldm list-io command to confirm that the root complexes in BB#01 have been deleted from the control domain.

```
# ldm list-io | grep primary
```

PCIE0	BUS	PCIE0	primary	IOV
PCIE4	BUS	PCIE4	primary	IOV
/BB0/CMUL/NET0	PCIE	PCIE0	primary	OCC
/BB0/CMUL/SASHBA	PCIE	PCIE0	primary	OCC
/BB0/CMUL/NET2	PCIE	PCIE4	primary	OCC

c. Cancel the redundant configuration of the virtual I/O devices assigned to a guest domain.

To first shut down the root domain (root-dom1) to which the root complexes in BB#01 are assigned, log in to each guest domain, and then cancel the redundant configuration of the virtual I/O device from root-dom1.

For details on how to use the redundant configuration software, see the documentation about the software for that redundant configuration.

In the following example, a virtual network device (vnet1) is canceled from the IPMP configuration. For details on the command, see the manual for Oracle Solaris.

Log in to the guest domain (guest0).

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.0%	4h 17m
guest0	active	-n----	5100	64	64G	0.0%	1h 13m
guest1	active	-n----	5101	64	64G	0.0%	1h 4m
root-dom0	active	-n--v-	5000	32	32G	0.0%	1h 47m

```

root-dom1      active      -n--v-  5001    32      32G      0.0%  1h 19m
# telnet localhost 5100
....
guest0#

```

Execute the `dladm show-phys` command to check the mapping between the virtual network interface (`vnet1`) and the network interface name (`net1`).

```

guest0# dladm show-phys
LINK          MEDIA          STATE    SPEED  DUPLEX    DEVICE
net0          Ethernet      up       0      unknown  vnet0
net1          Ethernet      up       0      unknown  vnet1

```

Execute the `ipmpstat -i` command to check the configuration information for the network interfaces configuring IPMP.

```

guest0# ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS      LINK      PROBE      STATE
net0       yes    ipmp0      -smbM--    up        disabled   ok
net1       no     ipmp0      is-----  up        disabled   ok

```

Execute the `if_mpadm -d net1` command to release `net1` from the IPMP group, and then execute the `ipmpstat -i` command to confirm that it has been released. The following example confirms that `STATE` is offline.

```

guest0# if_mpadm -d net1
guest0# ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS      LINK      PROBE      STATE
net0       yes    ipmp0      -smbM--    up        disabled   ok
net1       no     ipmp0      -s---d-    up        disabled   offline

```

Execute the `ipadm delete-ip net1` command to delete `net1`.

```

guest0# ipadm delete-ip net1

```

Similarly, perform the same release processing for the guest domain (`guest1`).

d. Remove the virtual I/O devices assigned from the root domain to be stopped.

Execute the `ldm remove-vdisk` and `ldm remove-vnet` commands to remove the assigned virtual disk (`vdisk`) and virtual network device (`vnet`) from the root domain to be stopped according to the following step.

The following example shows the execution of the commands for removing the virtual disk (`vdisk10`) and virtual network device (`vnet10`) that use the virtual I/O service of the `BB#01` root domain (`root-dom1`).

```

# ldm remove-vdisk vdisk10 guest0
# ldm remove-vnet vnet10 guest0

```

Perform the same deletion for the guest domain (guest1).

7. **Check the resource usage status of the I/O devices, and then cancel all the I/O devices in the SPARC M10-4S to be replaced.**

a. **Check the logical domain to which the root complexes in the SPARC M10-4S to be released are assigned.**

Execute the ldm list-io command to check the logical domain to which the root complexes in BB#01 are assigned.

The following example shows that only root-dom1 has PCIe endpoints starting with "/BB1/." You can see that the PCIe endpoint root complexes (BUS) PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, and PCIE15 are assigned to root-dom1.

# ldm list-io					
NAME	TYPE	BUS	DOMAIN	STATUS	
----	----	---	-----	-----	
PCIE0	BUS	PCIE0	primary	IOV	
PCIE1	BUS	PCIE1	root-dom0	IOV	
PCIE2	BUS	PCIE2	root-dom0	IOV	
PCIE3	BUS	PCIE3	root-dom0	IOV	
PCIE4	BUS	PCIE4	primary	IOV	
PCIE5	BUS	PCIE5	root-dom0	IOV	
PCIE6	BUS	PCIE6	root-dom0	IOV	
PCIE7	BUS	PCIE7	root-dom0	IOV	
PCIE8	BUS	PCIE8			
PCIE9	BUS	PCIE9	root-dom1	IOV	
PCIE10	BUS	PCIE10	root-dom1	IOV	
PCIE11	BUS	PCIE11	root-dom1	IOV	
PCIE12	BUS	PCIE12			
PCIE13	BUS	PCIE13	root-dom1	IOV	
PCIE14	BUS	PCIE14	root-dom1	IOV	
PCIE15	BUS	PCIE15	root-dom1	IOV	
....					
/BB1/CMUL/NET0	PCIE	PCIE8		UNK	
/BB1/CMUL/SASHBA	PCIE	PCIE8		UNK	
/BB1/PCI0	PCIE	PCIE9	root-dom1	OCC	
/BB1/PCI3	PCIE	PCIE10	root-dom1	OCC	
/BB1/PCI4	PCIE	PCIE10	root-dom1	OCC	
/BB1/PCI7	PCIE	PCIE11	root-dom1	OCC	
/BB1/PCI8	PCIE	PCIE11	root-dom1	OCC	
/BB1/CMUL/NET2	PCIE	PCIE12		UNK	
/BB1/PCI1	PCIE	PCIE13	root-dom1	OCC	
/BB1/PCI2	PCIE	PCIE13	root-dom1	OCC	
/BB1/PCI5	PCIE	PCIE14	root-dom1	OCC	
/BB1/PCI6	PCIE	PCIE14	root-dom1	OCC	
/BB1/PCI9	PCIE	PCIE15	root-dom1	OCC	
/BB1/PCI10	PCIE	PCIE15	root-dom1	OCC	

b. **Stop the root domain to which the root complexes in SPARC M10-4S to be released are assigned and then release the SPARC M10-4S.**

The following example executes the ldm stop-domain and ldm unbind-domain commands to release the root domain (root-dom1) and shows that

the root domain is in the inactive state.

```
# ldm stop-domain root-dom1
LDom root-dom1 stopped
# ldm unbind-domain root-dom1
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	16	14G	0.2%	4h 59m
guest0	active	-n----	5100	32	32G	0.0%	1h 55m
guest1	active	-n----	5101	32	32G	0.0%	1h 46m
root-dom0	active	-n--v-	5000	24	24G	0.0%	2h 29m
root-dom1	inactive	-----		24	24G		

- c. **Confirm that all of the I/O devices in the SPARC M10-4S to be replaced have been canceled.**

Execute the ldm list-io command to confirm that all the I/O devices have been released.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9		
PCIE10	BUS	PCIE10		
PCIE11	BUS	PCIE11		
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13		
PCIE14	BUS	PCIE14		
PCIE15	BUS	PCIE15		
(Omitted)				

8. **Release the system board (PSB<BB>) of the SPARC M10-4S from the physical partition.**
- Execute the deleteboard -c disconnect command to release the PSB from the physical partition.**

```
XSCF> deleteboard -c disconnect 01-0
PSB#01-0 will be unconfigured from PPAR immediately. Continue?
[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
```

```

0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.

```

- b. **Execute the showresult command to check the exit status of the deleteboard command that was just executed.**

An end value of 0 indicates the normal termination of the deleteboard command.

If the end value is other than 0 or if an error message is displayed upon executing the deleteboard command, it indicates abnormal termination of the deleteboard command. By referring to "C.1.2 deleteboard" based on the error message, identify the error and then take corrective action.

```

XSCF> showresult
0

```

- c. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the SPARC M10-4S to be replaced is in the "Assigned" state and that the [Pwr], [Conn], and [Conf] columns all show "n."

```

XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)        Assigned    y    y    y    Passed Normal
01-0  00(01)        Assigned    n    n    n    Passed Normal

```

9. **Execute the replacefru command to replace the SPARC M10-4S.**

```

XSCF> replacefru

```

Note - For details on the replacement of SPARC M10-4Ss by using the replacefru command, see "5.8 Releasing a SPARC M10-4/M10-4S FRU from the System with the replacefru Command" and "6.2 Incorporating a SPARC M10-4/M10-4S FRU into the System with the replacefru Command" in the *Fujitsu M10-4/Fujitsu M10-4S/SPARC M10-4/SPARC M10-4S Service Manual*.

10. **Incorporate the PSB into the physical partition.**

- a. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the replaced SPARC M10-4S is in the Assigned state and that the [Pwr], [Conn], and [Conf] columns all show "n."

```

XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----

```

00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	00(01)	Assigned	n	n	n	Passed	Normal

- b. **Execute the addboard -c configure command to incorporate the PSB into the physical partition.**
- To recover the original logical domain configuration, execute the addboard -c configure command with the -m bind=resource option specified.

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0.....30.....60.....90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

Note - If an error message appears during execution of the addboard command, see "[C.1.1 addboard](#)," and then identify the error and take corrective action.

- c. **Execute the showresult command to check the exit status of the addboard command that was just executed.**
- An end value of 0 indicates the normal termination of the addboard command. If the end value is other than 0 or if an error message is displayed upon executing the addboard command, it indicates abnormal termination of the addboard command. By referring to "[C.1.1 addboard](#)" based on the error message, identify the error and then take corrective action.

```
XSCF> showresult
0
```

- d. **Execute the showboards command to check the PSB status.**
- Confirm that both of the [Conn] and [Conf] columns show "y" after the PSB in the replaced SPARC M10-4S has been successfully incorporated.

```
XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)        Assigned    y    y    y    Passed  Normal
01-0  00(01)        Assigned    y    y    y    Passed  Normal
```

11. **Check the logical domain operation status.**
- a. **Execute the console command to connect to the console of the control**

domain and then log in to it.

```
XSCF> console -p 0
```

- b. **Execute the `ldm list-domain` command to confirm that the logical domain operation status has not changed after the incorporation of the SPARC M10-4S PSB (BB).**

To check the logical domain operation status, confirm that [STATE] indicates "active" and check the second character from the left of the string in [FLAGS]. The displayed characters and their meanings are as follows:

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including when [STATE] is other than "active")

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART    16     14G     0.0%  10h 30m
guest0        active    -n----   5100    32     32G     0.0%  2h 50s
guest1        active    -n----   5101    32     32G     0.0%  2h 30s
root-dom0     active    -n--v-   5000    24     24G     0.0%  3h 20s
root-dom1     inactive  ------  24      24G
```

12. Restart the use of the I/O devices.

a. Reassign root complexes.

Execute the `ldm bind-domain` and `ldm start-domain` commands to start the root domain in the unbind state to which root complexes in the replaced SPARC M10-4S were assigned.

The following example starts the root domain (root-dom1) in the unbind state, and confirms that it has started.

```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART    16     16G     0.2%  3h 8m
guest0        active    -n----   5100    32     32G     0.0%  3h 8m
guest1        active    -n----   5101    32     32G     0.0%  3h 8m
root-dom0     active    -n--v-   5000    24     24G     0.0%  3h 8m
root-dom1     active    -n--v-   5001    24     24G     7.3%  8s
```

Execute the `ldm list-io` command to confirm that the physical I/O devices are assigned to the root domain that has just started.

```
# ldm list-io
NAME          TYPE    BUS    DOMAIN    STATUS
----
```


PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV
....				
/BB1/CMUL/NET0	PCIE	PCIE8		UNK
/BB1/CMUL/SASHBA	PCIE	PCIE8		UNK
/BB1/PCI0	PCIE	PCIE9	root-dom1	OCC
/BB1/PCI3	PCIE	PCIE10	root-dom1	OCC
/BB1/PCI4	PCIE	PCIE10	root-dom1	OCC
/BB1/PCI7	PCIE	PCIE11	root-dom1	OCC
/BB1/PCI8	PCIE	PCIE11	root-dom1	OCC
/BB1/CMUL/NET2	PCIE	PCIE12		UNK
/BB1/PCI1	PCIE	PCIE13	root-dom1	OCC
/BB1/PCI2	PCIE	PCIE13	root-dom1	OCC
/BB1/PCI5	PCIE	PCIE14	root-dom1	OCC
/BB1/PCI6	PCIE	PCIE14	root-dom1	OCC
/BB1/PCI9	PCIE	PCIE15	root-dom1	OCC
/BB1/PCI10	PCIE	PCIE15	root-dom1	OCC

b. Add the virtual I/O device from the root domain to the guest domain.

Execute the `ldm add-vdisk` and `ldm add-vnet` commands to add, to each guest domain, the virtual disk (vdisk) and virtual network device (vnet) supported for the virtual I/O service of the started root domain.

The following example indicates the execution of the command for adding the virtual disk (vdisk10) and virtual network device (vnet10) that use the virtual I/O service of BB#01 root domain (root-dom1).

```
# ldm add-vdisk id=1 vdisk10 vol110@vds1 guest0
# ldm add-vnet id=1 vnet10 vsw10 guest0
```

Note - To add the virtual I/O device again, it is necessary to specify the ID assigned beforehand. You can check the ID from the result of execution of the `ldm list -l` command in the status used before the virtual I/O device is deleted.

Perform the same addition for the guest domain (guest1).

c. Incorporate the virtual I/O devices assigned to a guest domain into the redundant configuration.

Once the root domain (root-dom1) to which root complexes in BB#1 have been assigned has started, the virtual I/O device services corresponding to each guest domain are also started.

Log in to each guest domain, and then incorporate the virtual I/O devices from root-dom1 that were previously canceled into the redundant configuration. For details on how to use the redundant configuration software, see the documentation for the software for that redundant configuration.

The following describes an example of incorporating a virtual network device (vnet1) into the IPMP configuration. For details on the commands, see the manual for Oracle Solaris.

First, log in to the guest domain (guest0).

```
# ldm list-domain
NAME                STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary             active     -n-cv-   UART    16      14G       0.0%    4h 17m
guest0              active     -n----   5100    32      32G       0.0%    1h 13m
guest1              active     -n----   5101    32      32G       0.0%    1h 4m
root-dom0           active     -n--v-   5000    24      24G       0.0%    1h 47m
root-dom1           active     -n--v-   5001    24      24G       0.0%    1h 19m
# telnet localhost 5100
....
guest0#
```

Execute the `dladm show-phys` command to check the mapping between the virtual network interface (vnet1) and the network interface name (net1).

```
guest0# dladm show-phys
LINK      MEDIA      STATE      SPEED    DUPLEX    DEVICE
net0      Ethernet   up         0        unknown   vnet0
net1      Ethernet   up         0        unknown   vnet1
```

Execute the `ipadm create-ip` command, the `ipadm set-ifprop` command, and the `ipadm add-ipmp` command to register net1 as a standby device of ipmp0.

```
guest0# ipadm create-ip net1
guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipadm add-ipmp -i net1 ipmp0
```

Execute the `ipmpstat -i` command to confirm that STATE of the network interface configuring IPMP indicates ok.

```
guest0# ipmpstat -i
INTERFACE ACTIVE GROUP  FLAGS    LINK PROBE    STATE
net0      yes   ipmp0 -smbM-- up   disabled ok
net1      no    ipmp0 -s---d- up   disabled ok
```

Perform the same step for the other guest domain (guest1).

13. **Restore the system volume and I/O devices on the control domain to a redundant configuration.**

a. **Add the root complex configuration for the control domain.**

Add the root complexes in BB#01 that were previously removed from the control domain.

Execute the `ldm list-io` command to check the unassigned root complexes.

The following example shows that PCIE8 and PCIE12 are not assigned to root complexes with BB1 devices.

```
# ldm list-io
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
PCIE0                              BUS     PCIE0    primary   IOV
PCIE1                              BUS     PCIE1    root-dom0 IOV
PCIE2                              BUS     PCIE2    root-dom0 IOV
PCIE3                              BUS     PCIE3    root-dom0 IOV
PCIE4                              BUS     PCIE4    primary   IOV
PCIE5                              BUS     PCIE5    root-dom0 IOV
PCIE6                              BUS     PCIE6    root-dom0 IOV
PCIE7                              BUS     PCIE7    root-dom0 IOV
PCIE8                              BUS     PCIE8
PCIE9                              BUS     PCIE9    root-dom1 IOV
PCIE10                             BUS     PCIE10   root-dom1 IOV
PCIE11                             BUS     PCIE11   root-dom1 IOV
PCIE12                             BUS     PCIE12
PCIE13                             BUS     PCIE13   root-dom1 IOV
(Omitted)
```

Execute the `ldm add-io` command to add PCIE8 and PCIE12 to primary.

```
# ldm add-io PCIE8 primary
# ldm add-io PCIE12 primary
```

Execute the `ldm list-io` command to confirm that the root complexes in BB#01 have been added to the control domain.

```
# ldm list-io | grep primary
PCIE0                              BUS     PCIE0    primary   IOV
PCIE4                              BUS     PCIE4    primary   IOV
PCIE8                              BUS     PCIE8    primary   IOV
PCIE12                             BUS     PCIE12   primary   IOV
/BB0/CMUL/NET0                    PCIE    PCIE0    primary   OCC
/BB0/CMUL/SASHBA                  PCIE    PCIE0    primary   OCC
/BB0/CMUL/NET2                    PCIE    PCIE4    primary   OCC
(Omitted)
```

b. **Place the system volume in the control domain in a redundant configuration.**

Execute the `zpool status` command in the control domain to check the mirroring configuration status.

The following example describes how to configure the ZFS mirroring function for the system volume in the control domain.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                                ONLINE     0     0     0
        c2t50000393E802CCE2d0s0        ONLINE     0     0     0
errors: No known data errors
```

Execute the `zpool attach` command to incorporate the disks into a mirroring configuration.

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

Execute the `zpool status` command, and then confirm that the mirroring configuration has been established.

Use the `zpool status` command to confirm whether synchronization processing (resilver) is completed.

The following shows an example of the display during synchronization processing.

```
# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Mon Jan 27 15:55:47 2014
      21.1G scanned out of 70.6G at 120M/s, 0h7m to go
      21.0G resilvered, 29.84% done
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                                DEGRADED   0     0     0
        mirror-0
            c2t50000393E802CCE2d0s0        ONLINE     0     0     0
            c3t50000393A803B13Ed0s0        DEGRADED   0     0     0 (resilvering)
errors: No known data errors
```

Once synchronization processing is complete, the displayed screen will be as follows:

```
# zpool status rpool
pool: rpool
state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
    NAME                                STATE      READ  WRITE CKSUM
    rpool                                ONLINE      0     0     0
      mirror-0
        c2t50000393E802CCE2d0s0        ONLINE      0     0     0
        c3t50000393A803B13Ed0s0        ONLINE      0     0     0
errors: No known data errors
```

If you are using other devices in BB#01, establish a redundant configuration or resume the use of the devices. For details on how to establish a redundant configuration or resume the use of devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

A.4 For Expansion of a Newly Installed System of Version XCP 2220 or Later from the 1BB to 2BB Configuration

This section describes an example of constructing an environment in which a SPARC M10-4S with a 1BB configuration is expanded to a 2BB configuration with physical partition dynamic reconfiguration while the physical partition is being operated. It also describes the related procedure for environment construction. An example is used for explanation.

However, the SPARC M10-4S with the 1BB configuration assumes that the system configuration has been completed, including the XCP firmware and Oracle Solaris-related software conditions and XSCF setting. System configuration is required to use the physical partition dynamic reconfiguration.

This description also applies to SPARC M12.

A.4.1 Configuration Example

This section expands a SPARC M10-4S with a 1BB configuration to a 2BB configuration through physical partition dynamic reconfiguration and adds guest domain guest1 and root domain root-dom1 to which the root complexes of an expanded SPARC M10-4S (BB#1) have been assigned. This section also gives a configuration example in which the virtual I/O of each guest domain is made redundant and the system volume of the control domain is mirrored with the internal hard disk.

Figure A-4 Example of the 1BB Configuration (Before Expansion)

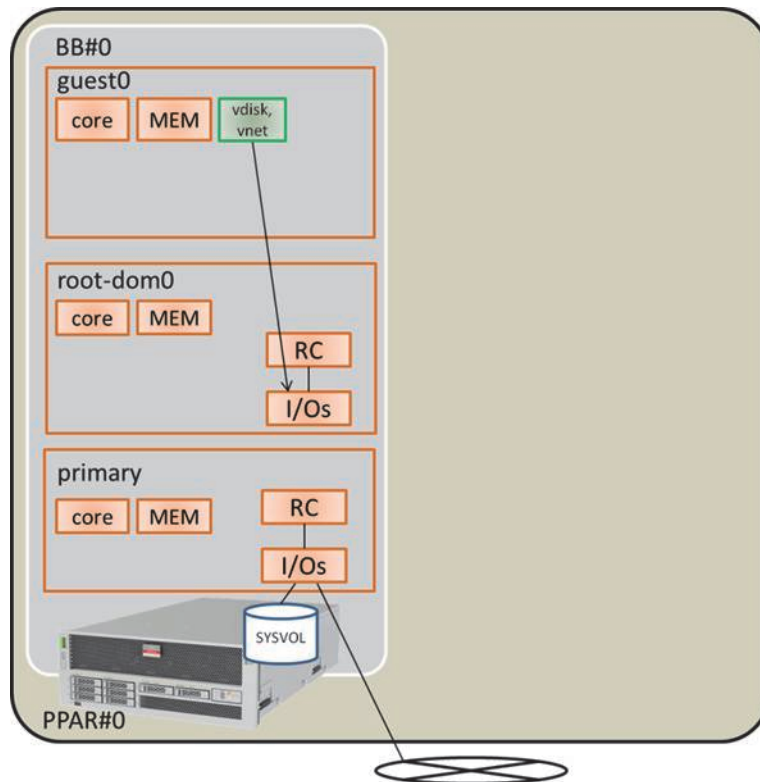


Figure A-5 Example of the Configuration After Expansion From 1BB to 2BB

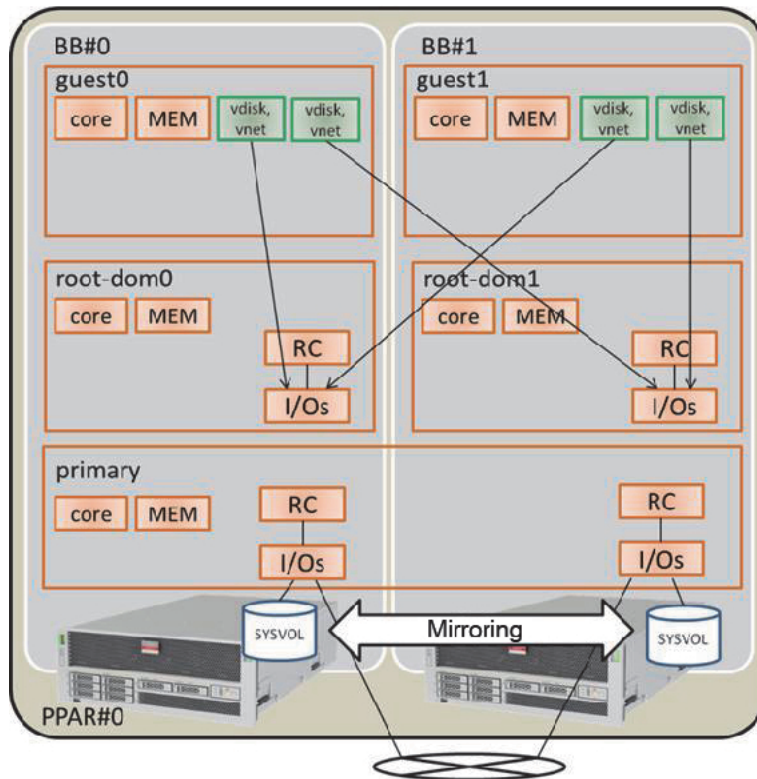


Table A-13 Example of a Configuration Expanded From 1BB to 2BB

Logical Domain	Before Expansion			After Expansion		
	CPU Core	Memory	I/O Configuration	CPU Core	Memory	I/O Configuration
Control domain (primary)	16	28 GB	On-board #0 (PCIE0, PCIE4)	32	56 GB	On-board #0 (PCIE0, PCIE4) On-board #1 (PCIE8, PCIE12)
guest0	32	64 GB	vdisk0, vnet0	32	64 GB	vdisk0, vdisk10 vnet0, vnet10
guest1	-	-	-	32	64 GB	vdisk1, vdisk11 vnet1, vnet11
root-dom0	16	32 GB	PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, PCIE7, vds0, vsw0	16	32 GB	PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, PCIE7, vds0, vsw0
root-dom1	-	-	-	16	32 GB	PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, PCIE15, vds1, vsw1

Table A-13 Example of a Configuration Expanded From 1BB to 2BB (*continued*)

Logical Domain	Before Expansion			After Expansion		
	CPU Core	Memory	I/O Configuration	CPU Core	Memory	I/O Configuration
Unassigned resource	0	Approx. 2 GB(*1)	-	0	Approx. 4.75 GB(*1)	-
Total	64	128 GB	-	128	256 GB	-

*1 A memory region of approx. 2 GB or 1.25 GB on each SPARC M10-4S is assigned for the hypervisor. So, the memory resource that can be assigned to the logical domain is smaller than the physically mounted memory capacity.

A.4.2 Procedure for Expansion (for Oracle VM Server for SPARC 3.1.x)

The following shows the procedure for expanding a SPARC M10-4S with a 1BB configuration to a 2BB configuration through dynamic reconfiguration according to the configuration example given in "[Figure A-5 Example of the Configuration After Expansion From 1BB to 2BB.](#)"

1. **Log in to the master XSCF.**
Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF.
If the standby XSCF is being used, log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

2. **Execute the `showsscp` command to check whether the IP address of the SP to SP communication protocol (SSCP) is the default value or a user-specified value.**

```
XSCF> showsscp
```

When the IP address is the default value and you are using the default value of the IP address of the SPARC M10-4S to be expanded, go to the next step.

To set a user value, set the IP address with the `setsscp` command. Then, use the `applynetwork` command to apply and check the IP address of the SSCP of the SPARC M10-4S to be expanded. Afterwards, execute the `rebootxscf` command to complete the setting and then go to the next step. For details on the procedure, see "7.5.6 Applying network settings" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

3. **Expand SPARC M10-4S.**
 - a. **Execute the `addfru` command to expand SPARC M10-4S according to the message.**

Note - If the IP address of the SSCP has not yet been set, executing the addfru command results in an error.

Note - If the addfru command is executed, the firmware version of the SPARC M10-4S to be expanded is automatically set to that of the SPARC M10-4S on which the master XSCF is running. In this example, the firmware version of the SPARC M10-4S (BB-ID 1) to be expanded is automatically set to that of the SPARC M10-4S (BB-ID 0) on which the master XSCF is running.

The following example expands BB#1.

```
XSCF> addfru
-----
Maintenance/Addition Menu
Please select the chassis including added FRU.
No.  FRU                      Status
-----
  1  /BB#0                     Normal
  2  /BB#1                     Unmount
  3  /BB#2                     Unmount
  4  /BB#3                     Unmount
-----
Select [1-16|c:cancel] :2
Maintenance/Addition Menu
Please select the BB or a type of FRU to be added.
1. BB itself
2. PSU (Power Supply Unit)
-----
Select [1,2|c:cancel] :1
Maintenance/Addition Menu
Please select a FRU to be added.
No.  FRU                      Status
-----
  1  /BB#1                     Unmount
Select [1|b:back] :1
```

b. Mount SPARC M10-4S to be expanded, to the rack.

While the previously mentioned addfru command is executed, if the message "After the added device is connected with the system, please turn on the breaker of the BB#1." appears, mount the SPARC M10-4S to be expanded in the rack.

For details on the mounting procedure, see "3.4.1 Mounting the SPARC M10-4S in a rack" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

c. Set the identification ID (BB-ID) of the SPARC M10-4S to be expanded.

See "4.1 Setting the ID (BB-ID) Identifying a Chassis" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

d. Connect the crossbar cable.

Connect the crossbar cable between the existing SPARC M10-4S and the expanded BB. For the connection cable routing figure and cable list, see "Appendix B Cable Connection Information on Building Block Configurations" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*. For details on direct connections between chassis, see "4.2 Connecting Cables (for Direct Connections between Chassis)" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

e. **Connect the XSCF BB control cable.**

Connect the existing SPARC M10-4S and expanded SPARC M10-4S through the XSCF BB control cable. For the routing figure and cable list of the XSCF BB control cables, see "Appendix B Cable Connection Information on Building Block Configurations" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*. For details on direct connections between chassis, see "4.2 Connecting Cables (for Direct Connections between Chassis)" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

f. **Connect the XSCF DUAL control cable.**

Connect the existing SPARC M10-4S and expanded SPARC M10-4S through the XSCF DUAL control cable. For details, see "4.2 Connecting Cables (for Direct Connections between Chassis)" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

g. **Connect the serial cable or LAN cable.**

Connect the serial cable to the XSCF serial port of the SPARC M10-4S to be expanded. Moreover, connect the cable to each LAN port of the XSCF-LAN, GbE port, and PCIe card. For details, see "5.1 Connecting Cables to the SPARC M10-4S" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

h. **Connect the input power to the SPARC M10-4S to be expanded.**

Connect the power cord of the SPARC M10-4S to be expanded to the input power.

i. **Enter "f" from the input screen of the addfru command.**

Enter "f" on the input screen for the addfru command executed in step a. to perform SPARC M10-4S expansion.

```
2) Please select[f:finish] :f
```

```
Waiting for BB#1 to enter install state.
```

```
[This operation may take up to 20 minute(s)] (progress scale  
reported in seconds)
```

```
0..... 30.... done
```

```
Waiting for BB#1 to enter ready state.
```

```
[This operation may take up to 45 minute(s)] (progress scale  
reported in seconds)
```

```
0..... 30..... 60... done
```

```
Do you want to start to diagnose BB#1? [s:start|c:cancel] :
```

- j. **Skip the diagnosis of the SPARC M10-4S to be expanded to exit the addfru command.**

Input "c" to the input screen of the addfru command on the master XSCF, and then skip the diagnosis processing for the SPARC M10-4S to be expanded. If "The addition of BB#1 has completed." appears, input "f" and then input "c" to exit the addfru command.

```
Do you want to start to diagnose BB#1? [s:start|c:cancel] :c

Diagnostic tests are about to be skipped.
Running diagnostic tests are strongly recommended before using BB#1.
Are you sure you want to skip testing? [y:yes|n:no] :y

-----
Maintenance/Addition Menu Status of the added FRU.

FRU                      Status
-----
/BB#1                    Normal
-----

[Warning:007]
Running diagnostic tests on BB#1 is strongly recommended after addfru has
completed.
The addition of BB#1 has completed. [f:finish] :f

-----
Maintenance/Addition Menu
Please select the chassis including added FRU.

No. FRU                  Status
-----
1  /BB#0                Normal
2  /BB#1                Normal
-----

Select [1,2|c:cancel] :c
```

4. **Diagnose the expanded SPARC M10-4S.**
a. **Execute the testsb command to perform a diagnosis test for the expanded SPARC M10-4S.**

Execute the testsb command to perform the diagnosis test. Specify the physical system board (PSB) number of the expanded SPARC M10-4S and check the initial diagnosis and connection I/O.

```
XSCF> testsb -v -p -s -y 01-0
Initial diagnosis is about to start, Continue?[y|n] :y
PSB#01-0 power on sequence started.
```

If an error is displayed, see "A.2.4 Checking diagnosis results" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

b. **Execute the diagxbu command to perform the crossbar cable diagnosis test.**

Execute the diagxbu command to perform the crossbar cable diagnosis test for the physical partition of the destination and for the expanded SPARC M10-4S. Specify the identification ID (BB-ID) of the expanded SPARC M10-4S and the physical partition ID (PPAR-ID) of the destination.

```
XSCF> diagxbu -y -b 01 -p 00
XBU diagnosis is about to start, Continue?[y|n] :y
Power on sequence started. [7200sec]
0..... 30..... 60..end
XBU diagnosis started. [7200sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240...../
270.....300.....330.....360.....390.....420.....450.....480.....510.....\
540.....570.....600.....630.....660.....690.....720.....750.....780.....\
810.....840.....870.....900.....930..end
completed.
Power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....180.end
completed.
```

c. **Execute the showlogs error command to check that no error is displayed.**

```
XSCF> showlogs error
```

If an error is displayed, see "A.2.4 Checking diagnosis results" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

d. **Execute the showhardconf command to check the configuration and status of the expanded SPARC M10-4S.**

Execute the showhardconf command to check the hardware configuration (CPU, memory, or the like) of the expanded SPARC M10-4S, and then check that Status of each unit is Normal.

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Service;
+ System_Power:On; System_Phase:Cabinet Power On;
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP1236052K ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
+ Freq:3.000 GHz; Type:0x10;
+ Core:16; Strand:2;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
```

```
+ FRU-Part-Number: CA20393-B50X A2 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
:
```

5. **Set the XSCF network of the expanded SPARC M10-4S.**

Set the XSCF network for the expanded SPARC M10-4S. For details, see "7.5.2 Setting an Ethernet (XSCF-LAN) IP address" and "7.5.3 Setting a takeover IP address" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*. After making this setting, execute the `applynetwork` command to apply and check the setting. Afterwards, execute the `rebootxscf` command to complete the setting and then go to the next step. For details on the procedure, see "7.5.6 Applying network settings" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

6. **Set mirror mode for the memory of the expanded SPARC M10-4S.**

If you are not using memory mirror mode, this step is not necessary.

For details on memory mirror mode, see "14.1 Configuring Memory Mirroring" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

a. **If you are making a mirror configuration of the memory in the expanded SPARC M10-4S, use the `setupfru` command to set memory mirror mode.**

The following example places all the CPUs under the SPARC M10-4S physical system board (PSB 01-0) in memory mirror mode.

```
XSCF> setupfru -m y sb 01-0
```

b. **Execute the `showfru` command to check the mirror mode setting of the memory.**

```
XSCF> showfru sb 01-0
Device Location Memory Mirror Mode
sb      01-0
  cpu 01-0-0   yes
  cpu 01-0-1   yes
  cpu 01-0-2   yes
  cpu 01-0-3   yes
```

7. **Register the system board of the expanded SPARC M10-4S in the physical partition configuration information.**

a. **Execute the `showpctl` command to check the physical partition configuration information.**

```
XSCF> showpctl -p 0
PPAR-ID      LSB      PSB      Status
00           Running
```

- b. **Execute the setpcl command to register the system board in the physical partition configuration information.**

Execute the setpcl command to register the expanded SPARC M10-4S system board in the physical partition configuration information for the built-in destination.

In the following example, physical system board (PSB) 01-0 is mapped to logical system board (LSB) 01 of physical partition 0.

```
XSCF> setpcl -p 0 -a 01=01-0
```

- c. **Execute the showpcl command to check the physical partition configuration information.**

```
XSCF> showpcl -p 0
PPAR-ID      LSB      PSB      Status
00
              00      00-0     Running
              01      01-0
```

8. **Register the CPU Activation key to assign CPU core resources.**

- a. **Execute the showcodusage command to check the CPU Activation information.**

Execute the showcodusage command to check whether the physical partition contains an assignable CPU core resource.

As shown here, the system has 128 mounted CPU core resources and 128 registered CPU Activations, 64 of the CPU core resources are in use, and the number of CPU Activations currently not in use is 64.

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-----
PROC      64      128      128 OK: 64 cores available
```

Note:

Please confirm the value of the "In Use" by the ldm command of Oracle VM Server for SPARC.

The XSCF may take up to 20 minutes to reflect the "In Use" of logical domains.

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and add the CPU Activation keys. For details on how to add a CPU Activation key, see "5.3 Adding CPU Core Resources" in the *Fujitsu SPARC M12*

b. Execute the showcod command to check the CPU core resource information for the physical partition.

Execute the showcod command to check information on the CPU core resources assigned to the physical partition.

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 64
```

If the assigned resources are insufficient, execute the setcod command to assign CPU core resources to the physical partition.

The following example adds 64 CPU core resources to physical partition 0.

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 : 64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

```
XSCF> setcod -s cpu
```

When the CPU core resources have been assigned, execute the showcod command again to check the information for the CPU core resources assigned to the physical partition.

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 128
```

9. Check the logical domain operation status.

a. Execute the console command to connect to the console of the control domain and then log in to it.

```
XSCF> console -p 0
```

b. Execute the ldm list-domain command to check the operation status of the logical domain.

To check the logical domain operation status, check the [STATE] and [FLAGS] combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

The following example shows that the control domain, one root domain, and one guest domain are operating.

Check whether all domains are in "active", which indicates that Oracle Solaris is in operating state, or "inactive" state. If there is a domain with OpenBoot PROM status or bound status, the dynamic reconfiguration of the physical partition may fail.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	28G	0.0%	1h 33m
guest0	active	-n----	5100	64	64G	3.1%	2s
root-dom0	active	-n--v-	5000	32	32G	3.1%	17m

10. **Incorporate the system board (PSB<BB>) of the expanded SPARC M10-4 into the physical partition.**

- a. **Return to the XSCF shell, and then execute the showboards command to check the PSB status.**

Execute the showboards command to check that the PSB (BB) status of the expanded SPARCM10-4S is "SP" (system board pool).

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	SP	Available	n	n	n	Passed	Normal

- b. **Execute the addboard command to incorporate the PSB into the physical partition.**

Execute the addboard -c configure command to incorporate the PSB into the physical partition.

```
XSCF> addboard -c configure -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.
```

Note - If an error message appears during execution of the addboard command, see "[C.1.1 addboard](#)," and then identify the error and take corrective action.

- c. **Execute the showresult command to check the exit status of the addboard command that was just executed.**

An end value of 0 indicates the normal termination of the addboard command.

If the end value is other than 0 or if an error message is displayed upon executing the addboard command, it indicates abnormal termination of the addboard command. By referring to "[C.1.1 addboard](#)" based on the error message, identify the error and then take corrective action.

```
XSCF> showresult
0
```

- d. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the added SPARC M10-4S is in the Assigned state and that the Pwr, Conn, and Conf columns all show "y."

```
XSCF> showboards -p 0
PSB   PPAR-ID(LSB)  Assignment  Pwr   Conn  Conf  Test   Fault
-----
00-0  00(00)         Assigned    y     y     y     Passed Normal
01-0  00(01)         Assigned    y     y     y     Passed Normal
```

11. **Check the logical domain operation status.**

- a. **Execute the console command to connect to the console of the control domain and then log in to it.**

```
XSCF> console -p 0
```

- b. **Execute the ldm list-domain command to check the operation status of the logical domain.**

To check the logical domain operation status, check the [STATE] and [FLAGS] combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

The following example shows that the control domain, one root domain, and one guest domain are operating.

```
# ldm list-domain
NAME          STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active  -n-cv-   UART    32      28G      0.0%    2h 3m
guest0        active  -n----   5100    64      64G      3.1%    33m
```

root-dom0	active	-n--v-	5000	32	32G	3.1%	47m
-----------	--------	--------	------	----	-----	------	-----

12. **Change the configuration of the existing logical domain and then add a new logical domain.**

The resources of the added SPARC M10-4S are assigned to the existing logical domain or to a newly configured logical domain. The following example performs the procedure for creating a mirroring configuration according to the configuration example. In other words, add the CPU cores and memory of the control domain and then add guest domain guest1 and root domain root-dom1 to create a redundant configuration of the virtual I/O for each guest domain. Furthermore, create a mirroring configuration of the system volume of the control domain with an internal hard disk.

- a. **Execute the ldm list-devices command to check the state of the added hardware resources.**

```
# ldm list-devices
CORE
  ID      %FREE    CPUSSET
  128     100     (256, 257)
  132     100     (264, 265)
  136     100     (272, 273)
  140     100     (280, 281)
....
MEMORY
  PA      SIZE      BOUND
  0x7000000000000000 32G
  0x7200000000000000 32G
  0x7400000000000000 32G
  0x7600500000000000 31488M
....
```

- b. **Execute the ldm add-core command and the ldm add-memory command to add the CPU cores and memory resources of the control domain.**

Execute the ldm add-core and ldm add-memory commands to add the CPU cores and memory of the control domain. Check that the CPU cores and memory have been added by executing the ldm list-domain command.

```
# ldm add-core 16 primary
# ldm add-memory 38G primary
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.0%	2h 13m
guest0	active	-n----	5100	64	64G	0.1%	43m
root-dom0	active	-n--v-	5000	32	32G	0.1%	57m

- c. **Create a new logical domain to which to assign the CPU cores and memory resources.**

The example here adds the root domain (root-dom1) and guest domain (guest1) to which CPU cores and memory can be assigned. However, suppose

that when a domain is configured, the number of CPU cores is first set with the `ldm set-core` command and then the memory size to be assigned is set with the `ldm set-memory` command. In this case, the memory blocks assigned to the logical domain become difficult to distribute (fragmentation) and memory placement becomes easy to consider at building block release.

The following shows command execution examples.

```
# ldm add-domain root-dom1
# ldm add-domain guest1
# ldm set-core 16 root-dom1
# ldm set-core 32 guest1
# ldm set-memory 32G root-dom1
# ldm set-memory 64G guest1
```

Note - To enable physical partition dynamic reconfiguration when the version of Oracle VM Server for SPARC is earlier than 3.2, set a multiple of (number of CPU cores x 256 MB) for the memory size to be assigned to each logical domain.

d. Assign the root complex to the root domain.

The following example assigns the root complexes of the added SPARC M10-4S to the root domain (root-dom1).

Execute the `ldm list-io` command to check the root complex to be added.

A device whose NAME begins with "/BB1" is on BB-ID 1 and the string shown in BUS of that device indicates the root complex.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9		
PCIE10	BUS	PCIE10		
PCIE11	BUS	PCIE11		
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13		
PCIE14	BUS	PCIE14		
PCIE15	BUS	PCIE15		
....				
/BB1/CMUL/NET0	PCIE	PCIE8		
/BB1/CMUL/SASHBA	PCIE	PCIE8		
/BB1/PCIO	PCIE	PCIE9		
/BB1/PCI3	PCIE	PCIE10		

/BB1/PCI4	PCIE	PCIE10
/BB1/PCI7	PCIE	PCIE11
/BB1/PCI8	PCIE	PCIE11
/BB1/CMUL/NET2	PCIE	PCIE12
/BB1/PCI1	PCIE	PCIE13
/BB1/PCI2	PCIE	PCIE13
/BB1/PCI5	PCIE	PCIE14
/BB1/PCI6	PCIE	PCIE14
/BB1/PCI9	PCIE	PCIE15
/BB1/PCI10	PCIE	PCIE15

Execute the `ldm add-io` command to add the root complexes to the root domain (root-dom1). Then, execute the `ldm list-io` command to perform a check.

The following example adds BB1 root complexes (PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, and PCIE15) other than root complexes (PCIE8 and PCIE12) of the above-mentioned on-board devices (/BB1/CMUL/NET0, /BB1/CMUL/SASHBA, and /BB1/CMUL/NET2), to root-dom1.

```
# ldm add-io PCIE9 root-dom1
# ldm add-io PCIE10 root-dom1
....
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8	root-dom1	IOV
PCIE9	BUS	PCIE9		
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV

e. Assign the console device of the root domain and then start it.

Assign the console device to the root domain (root-dom1) with the `ldm set-vconsole` command and then start it with the `ldm bind-domain` and `ldm start-domain` commands.

```
# ldm set-vconsole port=5001 vcc0 root-dom1
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
```

```
# telnet localhost 5001
....
```

f. **Install Oracle Solaris in the root domain.**

Install Oracle Solaris in the started root domain. For details on how to install Oracle Solaris, see documentation related to Oracle Solaris.

g. **Check the physical device to be assigned to the virtual service of the root domain.**

After installing Oracle Solaris, log in to the root domain and then execute the format command to check the name of the device to be assigned to the virtual disk service. In this case, executing the ldm list-io -l command displays the device path associated with the PCIe endpoint. So, you can check the path through which the disk is connected, by comparing the name of that device path with the name of the device path of the disk displayed with the format command.

```
# ldm list-io -l
NAME                                     TYPE    BUS      DOMAIN    STATUS
----                                     -
....
/BB1/PCIO                               PCIE    PCIE9    root-dom1OCC
[pci@8900/pci@4/pci@0/pci@0]
/BB1/PCI3                               PCIE    PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@0]
/BB1/PCI4                               PCIE    PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@8]....
# telnet localhost 5000
....
root-dom1# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
    0. c3t500000E01BDA70B2d0 <FUJITSU-MBB2147RC-3703 cyl 14087 alt 2 hd 24
sec 848>
        / pci@8900/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w500000e01bda70b2,0
    1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
        /pci@8a00/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
Specify disk (enter its number): ^C
```

Subsequently, executing the dladm show-phys -L command enables you to check the Ethernet interface name and physical location assigned to the root domain.

```
root-dom0# dladm show-phys -L
LINK          DEVICE          LOC
net0          igb0            /BB1/PCIO
net1          igb1            /BB1/PCIO
net2          igb2            /BB1/PCIO
```

h. Assign the virtual I/O service to the root domain.

Return to the control domain to add the virtual I/O service to the root domain (root-dom1). For details on the I/O device setting, see the *Oracle VM Server for SPARC Administration Guide* provided by Oracle Corporation.

The following example adds the virtual disk service (vds1) and virtual switch network interfaces (vsw10 and vsw11) to the added root domain (root-dom1).

```
# ldm add-vdiskserver vds1 root-dom1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol10@
vds1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol11@
vds1
# ldm add-vswitch net-dev=net1 vsw10 root-dom1
# ldm add-vswitch net-dev=net2 vsw11 root-dom1
```

i. Assign the virtual I/O device to the guest domain.

Add the virtual I/O device that uses the added virtual I/O service, to each guest domain.

The following example adds the virtual I/O devices (vdisk10, vnet10) that use the virtual I/O service of root-dom1 to the existing guest domain (guest0). It also adds the virtual I/O devices (vdisk1, vdisk11, vnet1, and vnet11) that use the virtual I/O services of root-dom0 and root-dom1 and the virtual console (vcons), to a guest domain (guest1) to be newly added.

```
# ldm add-vdisk vdisk10 vol10@vds1 guest0
# ldm add-vnet vnet10 vsw10 guest0
# ldm add-vdisk vdisk1 vol11@vds0 guest1
# ldm add-vdisk vdisk11 vol11@vds1 guest1
# ldm add-vnet vnet1 vsw1 guest1
# ldm add-vnet vnet11 vsw11 guest1
# ldm set-vconsole port=5101 guest1
```

j. Start the added guest domain.

Execute the ldm bind-domain and ldm start-domain commands to start the added guest domain (guest1).

```
# ldm bind-domain guest1
# ldm start-domain guest1
LDom guest1 started
```

k. Install Oracle Solaris in the added guest domain.

Install Oracle Solaris in the added guest domain. For details on how to install Oracle Solaris, see the documentation related to Oracle Solaris.

l. Establish a redundant configuration for the I/O of each guest domain.

Change the virtual I/O device added to each guest domain to establish a redundant configuration. For details on how to set the redundant configuration, see the documentation related to the software for each redundant configuration.

m. **Establish a redundant configuration for the system volume of the control domain.**

The example here assigns the on-board device of the added SPARC M10-4S to the control domain to create a redundant configuration. Switch the control domain to delayed reconfiguration mode.

```
# ldm start-reconf primary
```

```
Initiating a delayed reconfiguration operation on the primary domain.  
All configuration changes for other domains are disabled until the primary  
domain reboots, at which time the new configuration for the primary domain  
will also take effect.
```

Add the root complexes (PCIE8 and PCIE12) of the added SPARC M10-4S to the control domain and then restart Oracle Solaris.

```
# ldm add-io PCIE8 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
# ldm add-io PCIE12 primary
```

```
-----  
Notice: The primary domain is in the process of a delayed reconfiguration.  
Any changes made to the primary domain will only take effect after it reboots.  
-----
```

```
# shutdown -i6 -g0 -y
```

```
....
```

After restarting Oracle Solaris, establish a redundant configuration for the system volume. The following example checks the disk with the format command to establish a mirror configuration with ZFS. For details on how to set the redundant configuration, see the documentation related to the software for each redundant configuration.

```
# format
```

```
Searching for disks...done
```

```
AVAILABLE DISK SELECTIONS:
```

```
0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20  
sec 625>
```

```
    /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0  
    /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
```

```
1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20  
sec 625>
```

```
    /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
```

```

/dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

        NAME                                STATE      READ WRITE CKSUM
        rpool                                ONLINE     0     0     0
            c2t50000393E802CCE2d0s0        ONLINE     0     0     0

errors: No known data errors
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting

```

Execute the zpool status command, and then confirm that the mirroring configuration has been established. At this time, execute the zpool status command to check whether the synchronization processing (resilver) has been completed. The following example indicates that synchronization processing is in progress.

```

# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered. The pool will
continue to function in a degraded state.
action: Wait for the resilver to complete.
    Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Mon Jan 27 15:55:47 2014
    21.1G scanned out of 70.6G at 120M/s, 0h7m to go
    21.0G resilvered, 29.84% done
config:

        NAME                                STATE      READ WRITE CKSUM
        rpool                                DEGRADED   0     0     0
            mirror-0
                c2t50000393E802CCE2d0s0        ONLINE     0     0     0
                c3t50000393A803B13Ed0s0        DEGRADED   0     0     0 (resilvering)

errors: No known data errors

```

Once synchronization processing is complete, the displayed screen will be as follows:

```

# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

        NAME                                STATE      READ WRITE CKSUM
        rpool                                ONLINE     0     0     0

```



```

mirror-0          ONLINE      0      0      0
c2t50000393E802CCE2d0s0  ONLINE      0      0      0
c3t50000393A803B13Ed0s0  ONLINE      0      0      0

errors: No known data errors

```

n. Check and adjust the memory placement of each logical domain.

Execute the `ldm list-devices -a memory` command to check the memory block placement.

```

# ldm list-devices -a memory
MEMORY
  PA          SIZE          BOUND
0x700000000000 32G          root-dom1
0x720000000000 32G          guest1
0x740000000000 32G          guest1
0x760000800000 1272M         _sys_
0x760050000000 16G          primary
0x760450000000 10G          primary
0x7606d0000000 4864M
0x780000000000 32G          guest0
0x7a0000000000 32G          guest0
0x7c0000000000 256M         primary
0x7c0010000000 3584M         root-dom0
0x7c00f0000000 24832M        primary
0x7c0700000000 4G          root-dom0
0x7e0000800000 1272M         _sys_
0x7e0050000000 512M         _sys_
0x7e0070000000 256M         _sys_
0x7e0080000000 25088M        root-dom0
0x7e06a0000000 3584M         primary
0x7e0780000000 2G          primary

```

Suppose that memory blocks assigned to a logical domain or unassigned memory blocks are divided into small fragments. You can collect the fragmented memory blocks into a large continuous region by unbinding the logical domain and then binding it again. This enables the SPARC M10-4S to be easily added or deleted.

For details on the procedure, see step 11 in "[A.2.2 Example of the Physical Partition Configuration Procedure](#)" of "[A.2 For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has No Free Space for Hardware Resources.](#)"

o. Save logical domain configuration information.

Execute the `ldm list-domain` command to check that the configured logical domain has started.

```

# ldm list-domain
NAME          STATE      FLAGS    CONS  VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART   64    56G     64%   2h 54m
guest0        active    -n----   5100   64    64G     42%   2h 54m

```

guest1	active	-n----	5101	64	64G	11%	2h 54m
root-dom0	active	-n--v-	5000	32	32G	20%	2h 54m
root-dom1	active	-n--v-	5001	32	32G	18%	2h 54m

Execute the `ldm add-spconfig` command to save the logical domain configuration information to the XSCF.

```
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4 [current]
```

A.4.3 Procedure for Expansion (When Dynamic Assignment of the PCIe Bus is Available)

The following shows the procedure for expanding a SPARC M10-4S with a 1BB configuration to a 2 BB configuration through dynamic reconfiguration according to the configuration example given in ["Figure A-5 Example of the Configuration After Expansion From 1BB to 2BB."](#) The procedure is for an environment where the PCIe bus dynamic assignment is available (XCP 2240 or later with Oracle VM Server for SPARC 3.2 or later and the root domain with Oracle Solaris 11.2 SRU11.2.8.4.0 or later).

This description also applies to SPARC M12.

1. **Log in to the master XSCF.**
 Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF.
 If the standby XSCF is being used, log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

2. **Execute the `showsscp` command to check whether the IP address of the SP to SP communication protocol (SSCP) is the default value or a user-specified value.**

```
XSCF> showsscp
```

When the IP address is the default value and you are using the default value of the IP address of the SPARC M10-4S to be expanded, go to the next step.
 To set a user value, set the IP address with the `setsscp` command. Then, use the `applynetwork` command to apply and check the IP address of the SSCP of the

SPARC M10-4S to be expanded. Afterwards, execute the `rebootxscf` command to complete the setting and then go to the next step. For details on the procedure, see "7.5.6 Applying network settings" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

3. **Expand SPARC M10-4S.**

- a. **Execute the `addfru` command to expand SPARC M10-4S according to the message.**

Note - If the IP address of the SSCP has not yet been set, executing the `addfru` command results in an error.

Note - If the `addfru` command is executed, the firmware version of the SPARC M10-4S to be expanded is automatically set to that of the SPARC M10-4S on which the master XSCF is running. In this example, the firmware version of the SPARC M10-4S (BB-ID 1) to be expanded is automatically set to that of the SPARC M10-4S (BB-ID 0) on which the master XSCF is running.

The following example expands BB#1.

```
XSCF> addfru
-----
Maintenance/Addition Menu
Please select the chassis including added FRU.
No. FRU              Status
---
1  /BB#0             Normal
2  /BB#1             Unmount
3  /BB#2             Unmount
4  /BB#3             Unmount
-----
Select [1-16|c:cancel] :2
Maintenance/Addition Menu
Please select the BB or a type of FRU to be added.
1. BB itself
2. PSU (Power Supply Unit)
-----
Select [1,2|c:cancel] :1
Maintenance/Addition Menu
Please select a FRU to be added.
No. FRU              Status
---
1  /BB#1             Unmount
Select [1|b:back] :1
```

b. **Mount SPARC M10-4S to be expanded, to the rack.**

While the previously mentioned `addfru` command is executed, if the message "After the added device is connected with the system, please turn on the breaker of the BB#1." appears, mount the SPARC M10-4S to be expanded in the rack.

For details on the mounting procedure, see "3.4.1 Mounting the SPARC

M10-4S in a rack" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

c. **Set the identification ID (BB-ID) of the SPARC M10-4S to be expanded.**

See "4.1 Setting the ID (BB-ID) Identifying a Chassis" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

d. **Connect the crossbar cable.**

Connect the crossbar cable between the existing SPARC M10-4S and the expanded SPARC M10-4S. For the connection cable routing figure and cable list, see "Appendix B Cable Connection Information on Building Block Configurations" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*. For details on direct connections between chassis, see "4.2 Connecting Cables (for Direct Connections between Chassis)" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

e. **Connect the XSCF BB control cable.**

Connect the existing SPARC M10-4S and expanded SPARC M10-4S through the XSCF BB control cable. For the routing figure and cable list of the XSCF BB control cables, see "Appendix B Cable Connection Information on Building Block Configurations" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*. For details on direct connections between chassis, see "4.2 Connecting Cables (for Direct Connections between Chassis)" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

f. **Connect the XSCF DUAL control cable.**

Connect the existing SPARC M10-4S and expanded SPARC M10-4S through the XSCF DUAL control cable. For details, see "4.2 Connecting Cables (for Direct Connections between Chassis)" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

g. **Connect the serial cable or LAN cable.**

Connect the serial cable to the XSCF serial port of the SPARC M10-4S to be expanded. Moreover, connect the cable to each LAN port of the XSCF-LAN, GbE port, and PCIe card. For details, see "5.1 Connecting Cables to the SPARC M10-4S" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

h. **Connect the input power to the SPARC M10-4S to be expanded.**

Connect the power cord of the SPARC M10-4S to be expanded to the input power.

i. **Enter "f" from the input screen of the addfru command.**

Enter "f" on the input screen for the addfru command executed in step a. to perform SPARC M10-4S expansion.

```
2) Please select[f:finish] :f
```

```
Waiting for BB#1 to enter install state.
```

```
[This operation may take up to 20 minute(s)] (progress scale  
reported in seconds)
```

```
0..... 30..... done
```

```
Waiting for BB#1 to enter ready state.
```

```
[This operation may take up to 45 minute(s)] (progress scale
reported in seconds)
0..... 30..... 60... done

Do you want to start to diagnose BB#1? [s:start|c:cancel] :
```

j. **Skip the diagnosis of the SPARC M10-4S to be expanded to exit the addfru command.**

Input "c" to the input screen of the addfru command on the master XSCF, and then skip the diagnosis processing for the SPARC M10-4S to be expanded. If "The addition of BB#1 has completed." appears, input "f" and then input "c" to exit the addfru command.

```
Do you want to start to diagnose BB#1? [s:start|c:cancel] :c

Diagnostic tests are about to be skipped.
Running diagnostic tests are strongly recommended before using BB#1.
Are you sure you want to skip testing? [y:yes|n:no] :y

-----
Maintenance/Addition Menu Status of the added FRU.

FRU                      Status
-----
/BB#1                    Normal
-----

[Warning:007]
Running diagnostic tests on BB#1 is strongly recommended after addfru has
completed.
The addition of BB#1 has completed.[f:finish] :f

-----
Maintenance/Addition Menu
Please select the chassis including added FRU.

No. FRU                      Status
-----
1   /BB#0                    Normal
2   /BB#1                    Normal
-----

Select [1,2|c:cancel] :c
```

4. **Diagnose the expanded SPARC M10-4S.**

a. **Execute the testsb command to perform a diagnosis test for the expanded SPARC M10-4S.**

Execute the testsb command to perform the diagnosis test. Specify the physical system board (PSB) number of the expanded SPARC M10-4S and check the initial diagnosis and connection I/O.

```
XSCF> testsb -v -p -s -y 01-0
Initial diagnosis is about to start, Continue?[y|n] :y
PSB#01-0 power on sequence started.
```

If an error is displayed, see "A.2.4 Checking diagnosis results" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

b. Execute the diagxbu command to perform the crossbar cable diagnosis test.

Execute the diagxbu command to perform the crossbar cable diagnosis test for the physical partition of the destination and for the expanded SPARC M10-4S. Specify the identification ID (BB-ID) of the expanded SPARC M10-4S and the physical partition ID (PPAR-ID) of the destination.

```
XSCF> diagxbu -y -b 01 -p 00
XBU diagnosis is about to start, Continue?[y|n] :y
Power on sequence started. [7200sec]
0.....30.....60..end
XBU diagnosis started. [7200sec]
0.....30.....60.....90.....120.....150.....180.....210.....240...../
270.....300.....330.....360.....390.....420.....450.....480.....510.....\
540.....570.....600.....630.....660.....690.....720.....750.....780.....\
810.....840.....870.....900.....930...end
completed.
Power off sequence started. [1200sec]
0.....30.....60.....90.....120.....150.....180.end
completed.
```

c. Execute the showlogs error command to check that no error is displayed.

```
XSCF> showlogs error
```

If an error is displayed, see "A.2.4 Checking diagnosis results" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

d. Execute the showhardconf command to check the configuration and status of the expanded SPARC M10-4S.

Execute the showhardconf command to check the hardware configuration (CPU, memory, or the like) of the expanded SPARC M10-4S, and then check that Status of each unit is Normal.

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Service;
+ System_Power:On; System_Phase:Cabinet Power On;
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
```

```

CMUL Status:Normal; Ver:0101h; Serial:PP1236052K ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
+ Freq:3.000 GHz; Type:0x10;
+ Core:16; Strand:2;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
+ FRU-Part-Number: CA20393-B50X A2 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
:

```

5. **Set the XSCF network of the expanded SPARC M10-4S.**

a. **Execute the testsb command to perform a diagnosis test for the expanded SPARC M10-4S.**

Set the XSCF network for the expanded SPARC M10-4S. For details, see "7.5.2 Setting an Ethernet (XSCF-LAN) IP address" and "7.5.3 Setting a takeover IP address" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*. After making this setting, execute the applynetwork command to apply and check the setting. Afterwards, execute the rebootxscf command to complete the setting and then go to the next step. For details on the procedure, see "7.5.6 Applying network settings" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

6. **Set mirror mode for the memory of the expanded SPARC M10-4S.**

If you are not using memory mirror mode, this step is not necessary.

For details on memory mirror mode, see "14.1 Configuring Memory Mirroring" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

a. **If you are making a mirror configuration of the memory in the expanded SPARC M10-4S, use the setupfru command to set memory mirror mode.**

The following example places all the memories for the CPUs under the SPARC M10-4S physical system board (PSB 01-0) in memory mirror mode.

```
XSCF> setupfru -m y sb 01-0
```

b. **Execute the showfru command to check the mirror mode setting of the memory.**

```

XSCF> showfru sb 01-0
Device Location Memory Mirror Mode
sb      01-0
  cpu  01-0-0   yes
  cpu  01-0-1   yes
  cpu  01-0-2   yes

```

```
cpu 01-0-3    yes
```

7. **Register the system board of the expanded SPARC M10-4S in the physical partition configuration information.**
 - a. **Execute the showpctl command to check the physical partition configuration information.**

```
XSCF> showpctl -p 0
PPAR-ID      LSB      PSB      Status
00
              00      00-0     Running
```

- b. **Execute the setpctl command to register the system board in the physical partition configuration information.**

Execute the setpctl command to register the expanded SPARC M10-4S system board in the physical partition configuration information for the built-in destination.

In the following example, physical system board (PSB) 01-0 is mapped to logical system board (LSB) 01 of physical partition 0.

```
XSCF> setpctl -p 0 -a 01=01-0
```

- c. **Execute the showpctl command to check the physical partition configuration information.**

```
XSCF> showpctl -p 0
PPAR-ID      LSB      PSB      Status
00
              00      00-0     Running
              01      01-0
```

8. **Register the CPU Activation key to assign CPU core resources.**
 - a. **Execute the showcodusage command to check the CPU core resource information.**

Execute the showcodusage command to check whether the physical partition contains an assignable CPU core resource.

As shown here, the system has 128 mounted CPU core resources and 128 registered CPU Activations, 64 of the CPU core resources are in use, and the number of CPU Activations currently not in use is 64.

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-----
PROC          64      128      128 OK: 64 cores available
```


Note:

Please confirm the value of the "In Use" by the ldm command of Oracle VM Server for SPARC.

The XSCF may take up to 20 minutes to reflect the "In Use" of logical domains.

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and add the CPU Activation keys. For details on how to add a CPU Activation key, see "5.3 Adding CPU Core Resources" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

b. Execute the showcod command to check the CPU core resource information for the physical partition.

Execute the showcod command to check information on the CPU core resources assigned to the physical partition.

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 64
```

If the assigned resources are insufficient, execute the setcod command to assign CPU core resources to the physical partition.

The following example adds 64 CPU core resources to the physical partition.

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 : 64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

```
XSCF> setcod -s cpu
```

When the CPU core resources have been assigned, execute the showcod command again to check the information for the CPU core resources assigned to the physical partition.

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 128
```

9. Check the logical domain operation status.

- a. **Execute the console command to connect to the console of the control domain and then log in to it.**

```
XSCF> console -p 0
```

- b. **Execute the `ldm list-domain` command to check the operation status of the logical domain.**

To check the logical domain operation status, check the [STATE] and [FLAGS] combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

The following example shows that the control domain, one root domain, and one guest domain are operating.

Check whether all domains are in "active", which indicates that Oracle Solaris is in operating state, or "inactive" state. If there is a domain with OpenBoot PROM status or bound status, the dynamic reconfiguration of the physical partition may fail.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	28G	0.0%	1h 33m
guest0	active	-n----	5100	64	64G	3.1%	2s
root-dom0	active	-n--v-	5000	32	32G	3.1%	17m

10. **Incorporate the system board (PSB<BB>) of the expanded SPARC M10-4 into the physical partition.**

- a. **Return to the XSCF shell, and then execute the `showboards` command to check the PSB status.**

Execute the `showboards` command to check that the PSB status of the expanded SPARCM10-4S is "SP" (system board pool).

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	y	y	y	Passed	Normal
01-0	SP	Available	n	n	n	Passed	Normal

- b. **Execute the `addboard` command to incorporate the PSB into the physical partition.**

Execute the `addboard -c` configure command to incorporate the PSB into the physical partition.

```

XSCF> addboard -c configure -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.

```

Note - If an error message appears during execution of the addboard command, see "[C.1.1 addboard](#)," and then identify the error and take corrective action.

- c. **Execute the showresult command to check the exit status of the addboard command that was just executed.**

An end value of 0 indicates the normal termination of the addboard command. If the end value is other than 0 or if an error message is displayed upon executing the addboard command, it indicates abnormal termination of the addboard command. By referring to "[C.1.1 addboard](#)" based on the error message, identify the error and then take corrective action.

```

XSCF> showresult
0

```

- d. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the added SPARC M10-4S is in the Assigned state and that the Pwr, Conn, and Conf columns all show "y."

```

XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)         Assigned    y    y    y    Passed Normal
01-0  00(01)         Assigned    y    y    y    Passed Normal

```

11. Check the logical domain operation status.

- a. **Execute the console command to connect to the console of the control domain and then log in to it.**

```

XSCF> console -p 0

```

- b. **Execute the ldm list-domain command to check the operation status of the logical domain.**

To check the logical domain operation status, check the [STATE] and [FLAGS]

combination. If [STATE] indicates "active", the second character from the left of the string in [FLAGS] has the following meaning.

"n": Oracle Solaris is operating

"t": OpenBoot PROM status

"-": In another state (including [STATE] other than "active")

The following example shows that the control domain, one root domain, and one guest domain are operating.

# ldm list-domain							
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	28G	0.0%	2h 3m
guest0	active	-n----	5100	64	64G	3.1%	33m
root-dom0	active	-n--v-	5000	32	32G	3.1%	47m

12. Change the configuration of the existing logical domain and then add a new logical domain.

The resources of the added SPARC M10-4S are assigned to the existing logical domain or to a newly configured logical domain. The following example performs the procedure for creating a mirroring configuration according to the configuration example. In other words, add the CPU cores and memory of the control domain and then add guest domain guest1 and root domain root-dom1 to create a redundant configuration of the virtual I/O for each guest domain. Furthermore, create a mirroring configuration of the system volume of the control domain with an internal hard disk.

a. Execute the **ldm list-devices** command to check the state of the added hardware resources.

```
# ldm list-devices
CORE
    ID          %FREE      CPUSET
    128         100        (256, 257)
    132         100        (264, 265)
    136         100        (272, 273)
    140         100        (280, 281)
    ....
MEMORY
    PA          SIZE          BOUND
    0x7000000000000000    32G
    0x7200000000000000    32G
    0x7400000000000000    32G
    0x7600500000000000    31488M
    ....
```

b. Execute the **ldm add-core** command and the **ldm add-memory** command to add the CPU cores and memory resources of the control domain.

Execute the **ldm add-core** and **ldm add-memory** commands to add the CPU cores and memory of the control domain. Check that the CPU cores and memory have been added by executing the **ldm list-domain** command.

```
# ldm add-core 16 primary
# ldm add-memory 56G primary
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.0%	2h 13m
guest0	active	-n----	5100	64	64G	0.1%	43m
root-dom0	active	-n--v-	5000	32	32G	0.1%	57m

c. Create a new logical domain to which to assign the CPU cores and memory resources.

The example here adds the root domain (root-dom1) and guest domain (guest1) to which CPU cores and memory can be assigned. However, suppose that when a domain is configured, the number of CPU cores is first set with the ldm set-core command and then the memory size to be assigned is set with the ldm set-memory command. In this case, the memory blocks assigned to the logical domain become difficult to distribute (fragmentation) and memory placement becomes easy to consider at building block release.

The following shows command execution examples.

```
# ldm add-domain root-dom1
# ldm add-domain guest1
# ldm set-core 16 root-dom1
# ldm set-core 32 guest1
# ldm set-memory 32G root-dom1
# ldm set-memory 64G guest1
```

Note - To enable physical partition dynamic reconfiguration when the version of Oracle VM Server for SPARC is earlier than 3.2, set a multiple of (number of CPU cores x 256 MB) for the memory size to be assigned to each logical domain.

d. Assign the root complex to the root domain.

The following example assigns the root complexes of the added SPARC M10-4S to the root domain (root-dom1).

Execute the ldm list-io command to check the root complexes to be added to the root domain (root-dom1).

A device whose NAME begins with "/BB1" is on BB-ID 1 and the string shown in BUS of that device indicates the root complex.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV

PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9		
PCIE10	BUS	PCIE10		
PCIE11	BUS	PCIE11		
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13		
PCIE14	BUS	PCIE14		
PCIE15	BUS	PCIE15		
....				
/BB1/CMUL/NET0	PCIE	PCIE8		
/BB1/CMUL/SASHBA	PCIE	PCIE8		
/BB1/PCI0	PCIE	PCIE9		
/BB1/PCI3	PCIE	PCIE10		
/BB1/PCI4	PCIE	PCIE10		
/BB1/PCI7	PCIE	PCIE11		
/BB1/PCI8	PCIE	PCIE11		
/BB1/CMUL/NET2	PCIE	PCIE12		
/BB1/PCI1	PCIE	PCIE13		
/BB1/PCI2	PCIE	PCIE13		
/BB1/PCI5	PCIE	PCIE14		
/BB1/PCI6	PCIE	PCIE14		
/BB1/PCI9	PCIE	PCIE15		
/BB1/PCI10	PCIE	PCIE15		

Execute the `ldm add-io` command to add the root complexes to the root domain (root-dom1). Then, execute the `ldm list-io` command to perform a check.

The following example adds BB1 root complexes (PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, and PCIE15) other than root complexes (PCIE8 and PCIE12) of the above-mentioned on-board devices (/BB1/CMUL/NET0, /BB1/CMUL/SASHBA, and /BB1/CMUL/NET2), to root-dom1.

# <code>ldm add-io PCIE9 root-dom1</code>				
# <code>ldm add-io PCIE10 root-dom1</code>				
....				
# <code>ldm list-io</code>				
NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9	root-dom1	IOV
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		

PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV

e. **Assign the console device of the root domain and then start it.**

Assign the console device to the root domain (root-dom1) with the ldm set-vconsole command and then start it with the ldm bind-domain and ldm start-domain commands.

```
# ldm set-vconsole port=5001 vcc0 root-dom1
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
# telnet localhost 5001
....
```

f. **Install Oracle Solaris in the root domain.**

Install Oracle Solaris in the started root domain. For details on how to install Oracle Solaris, see documentation related to Oracle Solaris.

g. **Check the physical device to be assigned to the virtual service of the root domain.**

After installing Oracle Solaris, log in to the root domain and then execute the format command to check the name of the device to be assigned to the virtual disk service.

Executing the ldm list-io -l command on the control domain before logging in to the root domain displays the device path associated with the PCIe endpoint. So, you can check the path through which the disk is connected, by comparing the name of that device path with the name of the device path of the disk displayed with the format command.

```
# ldm list-io -l
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
....
/BB1/PCI0                          PCIE    PCIE9    root-dom1OCC
[pci@8900/pci@4/pci@0/pci@0]
/BB1/PCI3                          PCIE    PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@0]
/BB1/PCI4                          PCIE    PCIE10   root-dom1OCC
[pci@8a00/pci@4/pci@0/pci@8]....
# telnet localhost 5001
....
root-dom1# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
    0. c3t500000E01BDA70B2d0 <FUJITSU-MBB2147RC-3703 cyl 14087 alt 2 hd 24
sec 848>
        / pci@8900/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w500000e01bda70b2,0
    1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
```

```
sec 625>
      /pci@8a00/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
Specify disk (enter its number): ^C
```

Subsequently, executing the `dladm show-phys -L` command enables you to check the Ethernet interface name and physical location assigned to the root domain.

```
root-dom1# dladm show-phys -L
LINK          DEVICE          LOC
net0          igb0            /BB1/PCI0
net1          igb1            /BB1/PCI0
net2          igb2            /BB1/PCI0
net3          igb3            /BB1/PCI0
```

h. Assign the virtual I/O service to the root domain.

Return to the control domain to add the virtual I/O service to the root domain (root-dom1). For details on the I/O device setting, see the *Oracle VM Server for SPARC Administration Guide* provided by Oracle Corporation.

The following example adds the virtual disk service (vds1) and virtual switch network interfaces (vsw10 and vsw11) to the added root domain (root-dom1).

```
# ldm add-vdiskserver vds1 root-dom1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol10@
vds1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol11@
vds1
# ldm add-vswitch net-dev=net1 vsw10 root-dom1
# ldm add-vswitch net-dev=net2 vsw11 root-dom1
```

i. Assign the virtual I/O device to the guest domain.

Add the virtual I/O device that uses the added virtual I/O service, to each guest domain.

The following example adds the virtual I/O devices (vdisk10, vnet10) that use the virtual I/O service of root-dom1 to the existing guest domain (guest0). It also adds the virtual I/O devices (vdisk1, vdisk11, vnet1, and vnet11) that use the virtual I/O services of root-dom0 and root-dom1 and the virtual console (vcons), to a guest domain (guest1) to be newly added.

```
# ldm add-vdisk vdisk10 vol10@vds1 guest0
# ldm add-vnet vnet10 vsw10 guest0
# ldm add-vdisk vdisk1 vol11@vds0 guest1
# ldm add-vdisk vdisk11 vol11@vds1 guest1
# ldm add-vnet vnet1 vsw1 guest1
# ldm add-vnet vnet11 vsw11 guest1
# ldm set-vconsole port=5101 guest1
```


j. **Start the added guest domain.**

Execute the `ldm bind-domain` and `ldm start-domain` commands to start the added guest domain (guest1).

```
# ldm bind-domain guest1
# ldm start-domain guest1
LDom guest1 started
```

k. **Install Oracle Solaris in the added guest domain.**

Install Oracle Solaris in the added guest domain. For details on how to install Oracle Solaris, see the documentation related to Oracle Solaris.

l. **Establish a redundant configuration for the I/O of each guest domain.**

Change the virtual I/O device added to each guest domain to establish a redundant configuration. For details on how to set the redundant configuration, see the documentation related to the software for each redundant configuration.

m. **Establish a redundant configuration for the system volume of the control domain.**

The example here assigns the on-board device of the added SPARC M10-4S to the control domain to create a redundant configuration.

Add the root complexes (PCIE8 and PCIE12) of the added SPARC M10-4S to the control domain.

```
# ldm add-io PCIE8 primary
# ldm add-io PCIE12 primary
```

Then, establish a redundant configuration for the system volume. The following example checks the disk with the `format` command to establish a mirror configuration with ZFS. For details on how to set the redundant configuration, see the documentation related to the software for each redundant configuration.

```
# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
    0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
    sec 625>
        /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
        /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
    1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
    sec 625>
        /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
        /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
# zpool status rpool
pool: rpool
```

```

state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

      NAME                                STATE      READ WRITE CKSUM
      rpool                                ONLINE      0     0     0
        c2t50000393E802CCE2d0s0          ONLINE      0     0     0

errors: No known data errors
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting

```

Execute the `zpool status` command, and then confirm that the mirroring configuration has been established. At this time, execute the `zpool status` command to check whether the synchronization processing (resilver) has been completed. The following example indicates that synchronization processing is in progress.

```

# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered. The pool will
continue to function in a degraded state.
action: Wait for the resilver to complete.
  Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Mon Jan 27 15:55:47 2014
        21.1G scanned out of 70.6G at 120M/s, 0h7m to go
        21.0G resilvered, 29.84% done
config:

      NAME                                STATE      READ WRITE CKSUM
      rpool                                DEGRADED      0     0     0
        mirror-0
          c2t50000393E802CCE2d0s0          ONLINE      0     0     0
          c3t50000393A803B13Ed0s0          DEGRADED      0     0     0 (resilvering)

errors: No known data errors

```

Once synchronization processing is complete, the displayed screen will be as follows:

```

# zpool status rpool
pool: rpool
state: ONLINE
  scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

      NAME                                STATE      READ WRITE CKSUM
      rpool                                ONLINE      0     0     0
        mirror-0
          c2t50000393E802CCE2d0s0          ONLINE      0     0     0
          c3t50000393A803B13Ed0s0          ONLINE      0     0     0

```

n. Check and adjust the memory placement of each logical domain.

Execute the `ldm list-devices -a memory` command to check the memory block placement.

```
# ldm list-devices -a memory
MEMORY
  PA                SIZE          BOUND
  0x700000000000    32G           root-dom1
  0x720000000000    32G           guest1
  0x740000000000    32G           guest1
  0x760000800000    1272M        _sys_
  0x760050000000    16G           primary
  0x760450000000    10G           primary
  0x7606d0000000    4864M
  0x780000000000    32G           guest0
  0x7a0000000000    32G           guest0
  0x7c0000000000    256M        primary
  0x7c0010000000    3584M        root-dom0
  0x7c00f0000000    24832M       primary
  0x7c0700000000    4G           root-dom0
  0x7e0000800000    1272M        _sys_
  0x7e0050000000    512M        _sys_
  0x7e0070000000    256M        _sys_
  0x7e0080000000    25088M       root-dom0
  0x7e06a0000000    3584M        primary
  0x7e0780000000    2G           primary
```

Suppose that memory blocks assigned to a logical domain or unassigned memory blocks are divided into small fragments. You can collect the fragmented memory blocks into a large continuous region by unbinding the logical domain and then binding it again. This enables the SPARC M10-4S to be easily added or deleted.

For details on the procedure, see step 11 in "[A.2.2 Example of the Physical Partition Configuration Procedure](#)" of "[A.2 For New Installations from XCP 2220 or Later, Where the Logical Domain Configuration Has No Free Space for Hardware Resources.](#)"

o. Save logical domain configuration information.

Execute the `ldm list-domain` command to check that the configured logical domain has started.

```
# ldm list-domain
NAME      STATE    FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
primary   active   -n-cv-  UART   64    56G     64%   2h 54m
guest0    active   -n----  5100   64    64G     42%   2h 54m
guest1    active   -n----  5101   64    64G     11%   2h 54m
root-dom0 active   -n--v-  5000   32    32G     20%   2h 54m
root-dom1 active   -n--v-  5001   32    32G     18%   2h 54m
```

Execute the `ldm add-spconfig ldm-set4` command to save the logical domain configuration information to the XSCF.

```
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4 [current]
```

A.5 For Installation of the System Board Configured by the SPARC64 X+ Processor to the Physical Partition Configured Only by the SPARC64 X Processor

This section gives an example of environment establishment that has a SPARC64 X+ processor mounted on a SPARC M10-4S added to the physical partition in SPARC M10. The physical partition has the mounted SPARC64 X processor to which the XCP firmware of XCP 2210 or earlier was applied. This section also gives a procedural example.

A.5.1 Configuration Example

This section gives a configuration example in which a SPARC M10-4S with a SPARC64 X+ processor mounted is added to the physical partition configured with one SPARC M10-4S with a SPARC64 X processor mounted. This enables active replacement of each SPARC M10-4S.

To enable active replacement of the SPARC M10-4S with PPAR DR, the configuration must satisfy the following conditions.

- It is necessary to create a redundant configuration by connecting I/O devices under the root complex on each SPARC M10-4S to the system volume I/O devices and the network of the control domain. This is done for the following reasons: When a SPARC M10-4S is removed, the I/O devices for the remaining SPARC M10-4Ss can be used to continue the operation even if an I/O configuration change is done by restarting the control domain.
- It is necessary to divide root domains for each SPARC M10-4S. This is done for the following reasons: When a SPARC M10-4S is removed, the status whereby it is impossible to use services provided by the physical I/O devices of other SPARC

M10-4Ss must be avoided.

- It is necessary to operate the business with a guest domain consisting of virtual I/O devices (vdisk, vnet). This is done for the following reasons: The root domain is stopped at SPARC M10-4S removal and the physical I/O devices can be disconnected. Moreover, these operations will not influence the guest domain that is performing the operation.

Figure A-6 System Configuration Example (1BB Configuration: SPARC64 X Processor Mounted) (Before Expansion)

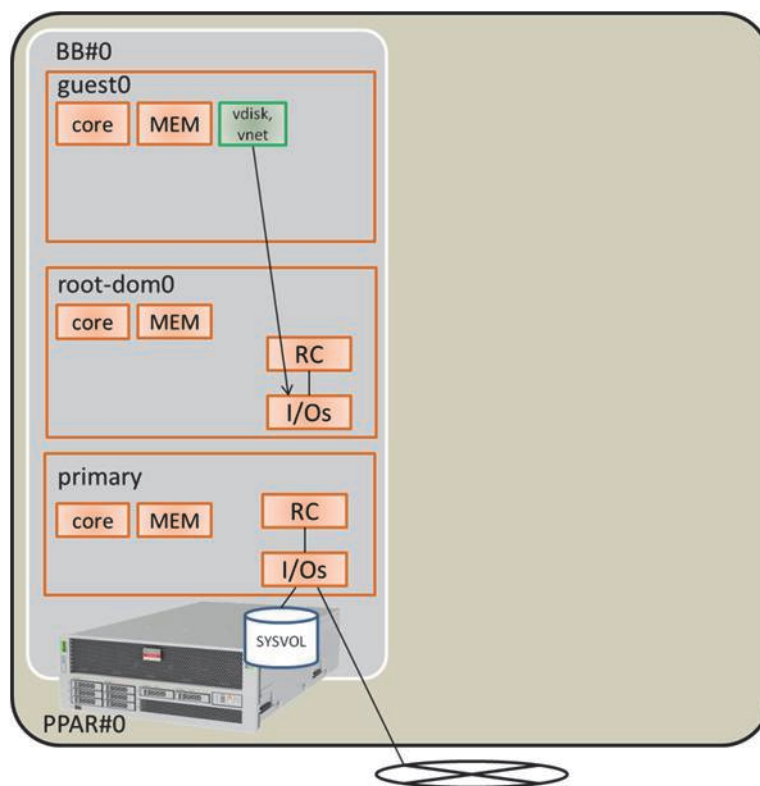


Figure A-7 Example of a System Configuration in Which PPAR DR is Possible
(2BB Configuration: After Installation of the System Board Mounted on the SPARC64 X+ Processor)

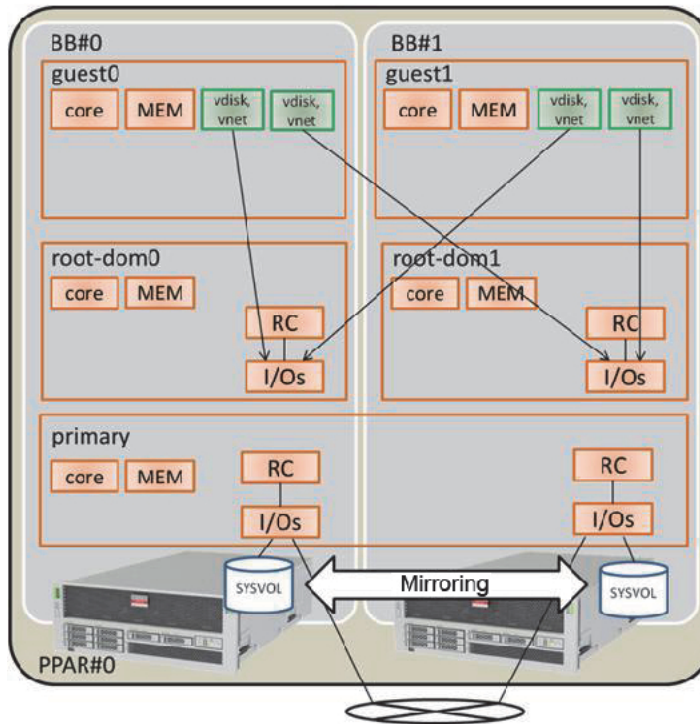


Table A-14 Example of a Configuration Expanded From 1BB to 2BB

Logical Domain	Before Expansion			After Expansion		
	CPU Core	Memory	I/O Configuration	CPU Core	Memory	I/O Configuration
Control domain (primary)	16	28 GB	On-board #0 (PCIE0, PCIE4)	32	56 GB	On-board #0 (PCIE0, PCIE4) On-board #1 (PCIE8, PCIE12)
guest0	32	64 GB	vdisk0, vnet0	32	64 GB	vdisk0, vdisk10 vnet0, vnet10
guest1	-	-	-	32	64 GB	vdisk1, vdisk11 vnet1, vnet11
root-dom0	16	32 GB	PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, PCIE7, vds0, vsw0	16	32 GB	PCIE1, PCIE2, PCIE3, PCIE5, PCIE6, PCIE7, vds0, vsw0
root-dom1	-	-	-	16	32 GB	PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, PCIE15, vds1, vsw1

Table A-14 Example of a Configuration Expanded From 1BB to 2BB (*continued*)

Logical Domain	Before Expansion			After Expansion		
	CPU Core	Memory	I/O Configuration	CPU Core	Memory	I/O Configuration
Unassigned resource	0	Approx. 2 GB(*1)	-	0	Approx. 4.75 GB(*1)	-
Total	64	128 GB	-	128	256 GB	-

*1 A memory region of approx. 2 GB or 1.25 GB on each SPARC M10-4S is assigned for the hypervisor. So, the memory resource that can be assigned to the logical domain is smaller than the physically mounted memory capacity.

A.5.2 Configuration Procedure

The following gives an example of the procedure for installing a SPARC M10-4S mounted on a SPARC64 X+ processor to the physical partition mounted on the SPARC64 X processor to which the XCP firmware of XCP 2210 or earlier was applied.

1. **Save logical domain configuration information to an XML file.**

If the PPAR DR function is enabled after the XCP firmware is updated, the logical domain configuration returns to the factory setting status (factory-default). So, the logical domain configuration information saved to XSCF cannot be used. For this reason, save the current logical domain configuration information to the XML file and then update the XCP firmware. After updating, restore the logical domain configuration information from the XML file to enable the easy reconfiguration of the logical domain.

The following explains the procedure for saving the logical domain configuration information saved in XSCF to the XML file.

For details on the configuration information saved to the XML file, however, see the *Oracle VM Server for SPARC Administration Guide*.

- a. **Log in to the control domain.**

- b. **Switch to the configuration information to be saved.**

Execute the `ldm list-spconfig` command on the control domain to list the logical domain configuration information. The line with `[current]` indicates the currently applied configuration information. In the following example, `ldm-set3` is applied.

```
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3 [current]
```

If only `[next poweron]` is displayed, there will be a difference between the logical domain configuration information stored in the XSCF and that stored in the control domain. So, execute the `ldm add-spconfig` command to save the current logical domain configuration information with a different name.

```
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3 [next poweron]
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4 [current]
```

When the logical domain configuration information indicated in [current] matches the configuration information you want to save, proceed to step 1-c.

If they are different, execute the `ldm set-spconfig` command to switch to the configuration information you want to save.

In the following example, the configuration information is switched with the `ldm set-spconfig` command and then `ldm-set1` is saved.

```
# ldm set-spconfig ldm-set1
# ldm list-spconfig
factory-default
ldm-set1 [next poweron]
ldm-set2
ldm-set3 [current]
```

Execute the `poweroff` and `poweron` commands on XSCF to power off the physical partition (PPAR) and then power it back on. In this case, execute the `poweroff` command after changing the logical domain state to the state in which Oracle Solaris is running or the inactive state.

The following example powers off PPAR-ID 0 and then powers it back on.

```
XSCF> poweroff -p 0
XSCF> poweron -p 0
```

Execute the `ldm list-spconfig` command to check that the specified logical domain configuration information is set. The following example indicates that `ldm-set1` is set as the current configuration information.

```
# ldm list-spconfig
factory-default
ldm-set1 [current]
ldm-set2
ldm-set3
```

- c. **Execute the `ldm list-constraints` command to save the current logical domain configuration information to the XML file.**

Execute the `ldm list-constraints` command to save the current logical configuration information to the XML file.

The following example saves the current logical domain configuration information to `ldm-set1.xml`. To prevent the saved XML file from being lost, back it up to another media or the like.

```
# ldm list-constraints -x > /ldm-set1.xml
```

d. Check whether the configuration information is saved to the XML file.

Execute the `more` command of Oracle Solaris or the like to check that the information has been saved to the XML file.

```
# more /ldm-set1.xml
<?xml version="1.0"?>
<LDM_interfaceversion="1.3" xmlns:xsi=http://www.w3.org/2001/
XMLSchema-instance
```

e. To save any other logical domain configuration information, repeat steps 1.b to 1.e.

If there is any other configuration information that you want to save, repeat steps 1.b to 1.e.

2. Save the XSCF setting information.

XSCF setting information can be saved to a USB device or to an external server through a network. The following describes each method.

- Saving the XSCF setting information to a USB device

a. Log in to the master XSCF.

Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF.

Note - On a system with a building block configuration, if you logged in to the XSCF in the standby state, log in again to the master XSCF.

```
XSCF> showbbstatus
BB#00 (Master)
```

b. Save the setting information to the USB device of the master XSCF.

i. Connect the USB device to the USB port on the XSCF unit panel (rear panel) of the master XSCF.

ii. Specify the output file name on the local USB device on the XSCF and then execute the `dumpconfig` command.

The setting information is saved with the specified file name in base64 encoding text format.

```
XSCF> dumpconfig file:///media/usb_msd/backup-file.txt
operation completed
```

When the "operation completed" message appears, data transfer has ended normally.

iii. **Once data transfer has ended, remove the USB device from the USB port.**

c. **Using an editor or the like on a PC, check the saved setting file information.**

Check the following information.

- User-Comments: Comments made when the -c option was specified in the dumpconfig command
- Created: Date and time when the information was saved
- Platform: Model name
- Serial-No: System serial number

Note - Any file saved to a USB device cannot be opened on the XSCF. It is necessary to download the file for checking.

```
XSCF Configuration File
User-Comments:
Encrypted: No
Created: Mon Jan 27 13:47:38 2014
Platform: M10-4S
Serial-No: 2111234001
Chassis-serial80:
Chassis-serial81:
Chassis-serial82:
Chassis-serial83:
Chassis-serial00:2111234001
Chassis-serial01:2111234003
Chassis-serial02:
Chassis-serial03:
Chassis-serial04:
Chassis-serial05:
Chassis-serial06:
Chassis-serial07:
Chassis-serial08:
Chassis-serial09:
Chassis-serial10:
Chassis-serial11:
Chassis-serial12:
Chassis-serial13:
Chassis-serial14:
Chassis-serial15:
Version: 0001
begin-base64_common
U1VOVyxTUEFSQy1FbnRlcnByaXNlAAAAAAAFmZ6gAAPrfADhbdAAAAAIyMTExMjMOMDAzAAAA
...
```

- **Saving the setting information to an external server through a network**

a. **Log in to the master XSCF.**

Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF.

Note - On a system with a building block configuration, if you logged in to the XSCF in the standby state, log in again to the master XSCF.

```
XSCF> showbbstatus
BB#00 (Master)
```

b. **Specify the target directory through a network to save the XSCF setting information.**

Specify the target directory and output file name and then execute the `dumpconfig` command. The setting information is saved with the specified file name in base64 encoding text format.

`user_name` specifies the user name of the save destination server.

```
XSCF> dumpconfig -u user-name ftp://server/backup/backup-sca-ff2-16.txt
operation completed
```

When the "operation completed" message appears, data transfer has ended normally.

c. **Using an editor or the like on a PC, check the saved setting file information.**

Check the following information.

- User-Comments: Comments made when the `-c` option was specified in the `dumpconfig` command
- Created: Date and time when the information was saved
- Platform: Model name
- Serial-No: System serial number

```
XSCF Configuration File
User-Comments:
Encrypted: No
Created: Mon Jan 27 13:47:38 2014
Platform: M10-4S
Serial-No: 2111234001
Chassis-serial80:
Chassis-serial81:
Chassis-serial82:
Chassis-serial83:
Chassis-serial00:2111234001
Chassis-serial01:2111234003
Chassis-serial02:
Chassis-serial03:
```

```

Chassis-serial04:
Chassis-serial05:
Chassis-serial06:
Chassis-serial07:
Chassis-serial08:
Chassis-serial09:
Chassis-serial10:
Chassis-serial11:
Chassis-serial12:
Chassis-serial13:
Chassis-serial14:
Chassis-serial15:
Version: 0001
begin-base64_common
U1VOVyxTUEFSQy1FbnRlcnByaXNlAAAAAAAFmZ6gAAPrfADhbdAAAAAIyMTExMjM0MDAzAAAA
...

```

3. **Update Oracle Solaris and Oracle VM Server for SPARC.**

Obtain Oracle Solaris and Oracle VM Server for SPARC that support physical partition dynamic reconfiguration and apply them to the system.

a. **Update Oracle VM Server for SPARC of the control domain.**

See the latest *Product Notes* for your server to check the latest versions of Oracle Solaris and Oracle VM Server for SPARC. Then, update Oracle VM Server for SPARC using the following procedure.

i. **Obtain Oracle VM Server for SPARC for use with the control domain.**

- **If the control domain is Oracle Solaris 10**

Obtain Oracle VM Server for SPARC that supports physical partition dynamic reconfiguration. For details on how to obtain this, see the latest *Product Notes* for your server.

- **If the control domain is Oracle Solaris 11**

Obtain SRU11.1.14.0 or later.

ii. **Whenever Oracle VM Server for SPARC or Oracle Solaris for the control domain has been updated, restart the control domain.**

Restarting the control domain may cause an I/O domain to panic or a guest domain I/O to stop. If such a logical domain exists, stop it in advance with the shutdown command or stop it by executing the `ldm stop-domain` command from the control domain.

Using the following method, you can check whether the logical domain is to be stopped.

- **For an I/O domain to which the PCIe endpoint is assigned from the control domain**

Execute the `ldm list-io` command to determine whether the root complex (BUS) of the PCIe endpoint assigned to the logical domain is assigned to primary. The following example indicates that the bus "PCIE2" of PCIe endpoints "/BB0/PCI3" and "/BB0/PCI4" assigned to `iodom0` is assigned to primary (control domain). Note that, since the configuration explained here differs from that in ["A.5.1 Configuration Example,"](#) changes have

been made to the logical domains and some other information.

```
# ldm list-io
NAME                                TYPE    BUS      DOMAIN    STATUS
----                                -
PCIE0                              BUS     PCIE0    primary   IOV
PCIE1                              BUS     PCIE1    primary   IOV
PCIE2                              BUS     PCIE2    primary   IOV
PCIE3                              BUS     PCIE3    primary   IOV
....
/BB0/CMUL/NET0                     PCIE    PCIE0    primary   OCC
/BB0/CMUL/SASHBA                    PCIE    PCIE0    primary   OCC
/BB0/PCI0                           PCIE    PCIE1    primary   OCC
/BB0/PCI3                           PCIE    PCIE2    iodom0    OCC
/BB0/PCI4                           PCIE    PCIE2    iodom0    OCC
/BB0/PCI7                           PCIE    PCIE3    primary   OCC
/BB0/PCI8                           PCIE    PCIE3    primary   EMP
....
```

- For the guest domain to which the virtual service of the control domain is assigned

Execute `ldm list-bindings primary` to check the correspondence between the virtual network switch (VSW) and the connection destination (PEER), as well as between the virtual disk service (VDS) and the logical domain (CLIENT) that uses a virtual disk. In the following example, you can check that "guestdom0" is set for VSW PEER and VDS CLIENT.

```
# ldm list-bindings primary
....
VSW
NAME                                MAC                                NET-DEV  ID  DEVICE      LINKPROP
DEFAULT-VLAN-ID PVID
VID
vsw0                                00:14:4f:f9:88:ca net0      0  switch@0    1
1
PEER                                MAC                                PVID VID
MTU  MAXBW  LINKPROP
INTERVNETLINK
vnet0@guestdom0                    00:14:4f:fa:64:dd 1
1500
VDS
NAME                                VOLUME  OPTIONS  MPGROUP  DEVICE
....
CLIENT
vdisk0@guestdom0                    vol0
```

Execute the `ldm stop-domain` command to stop the logical domain checked above. In the following example, the `ldm stop-domain` command is executed to stop `iodom0` and `guestdom0` and then the `ldm list-domain` command is executed to check that the bound state is set.

```
# ldm stop-domain guestdom0
LDom guestdom0 stopped
# ldm stop-domain iodom0
LDom iodom0 stopped
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	58G	0.0%	6h 3m
guestdom0	bound	-----	5100	64	64G		
iodom0	bound	-----	5000	32	32G		

iii. Update Oracle VM Server for SPARC of the control domain.

- For Oracle Solaris 10

Uninstall the old Oracle VM Server for SPARC and then install the new Oracle VM Server for SPARC. For details, see the README file provided with the obtained Oracle VM Server for SPARC.

- For Oracle Solaris 11

Apply the SRU. For details, see the installation manual of the obtained SRU.

iv. Execute the shutdown command to restart the control domain.

```
# shutdown -i6 -g0 -y
....
```

v. Execute the ldm start-domain command to start the logical domain stopped in ii, above.

In the following example, the ldm start-domain command is executed to start "iodom0" and "guestdom0" and then the ldm list-domain command is executed to check that the active state is set.

```
# ldm start-domain guestdom0
# ldm start-domain iodom0
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	58G	0.0%	6h 3m
guestdom0	active	-n----	5100	64	64G		
iodom0	active	-n----	5000	32	32G		

b. Update Oracle Solaris for logical domains other than the control domain.

See the latest *Product Notes* for your server, and update Oracle Solaris for logical domains other than the control domain. For details on the update procedure, see the information relating to each update.

4. Update the XCP firmware.

Install the SPARC M10-4S mounted on the SPARC64 X+ processor in the physical partition configured with the SPARC64 X processor. To do this, update the physical partition configured with the SPARC64 X processor to the XCP firmware of XCP 2220 or later.

a. **Obtain the latest XCP firmware.**

i. **Download the program file for the XCP firmware.**

Download, from the web site, the program file (XCPxxxx.tar.gz or XCPxxxx.exe) for the XCP firmware to any folder on a PC connected to this system.

Use one of the following methods to obtain the firmware for the server you are using.

- **Japanese site**

A customer who has entered into a contract with the SupportDesk can obtain firmware from SupportDesk-Web.

- **Global site**

For details on the method used to obtain the latest file of the firmware, contact our sales personnel. The following files are provided.

- Firmware program file
(XSCF Control Package (XCP) file)
- XSCF extended MIB (XSCF-SP-MIB) definition file

ii. **Check the XCP version of the downloaded program file.**

Check the version of the program file of the downloaded XCP firmware. For the XCP version, refer to the 4-digit number in the file name of the firmware program (tar.gz format) to check that it is the XCP firmware version update to be applied. For example, if the name of the program file is "XCP2220.tar.gz," the XCP version is 2220.

iii. **Decompress the downloaded program file.**

Decompress the program file of the downloaded XCP firmware. The XCP image file to import into the system is expanded. For example, if "XCP2220.tar.gz" is decompressed, "BBXCP2220.tar.gz" is expanded.

b. **Check the current XCP firmware version.**

i. **Log in to the master XSCF.**

Execute the showbbstatus command to check that the XSCF to which you have logged in is the master XSCF.

Note - On a system with a building block configuration, if you logged in to the XSCF in the standby state, log in again to the master XSCF.

```
XSCF> showbbstatus
BB#00 (Master)
```

ii. **Execute the version command to check the XCP version of the current system.**

Before updating the firmware, check the XCP version of the current system. In the following example, the version command is executed with the -c xcp option added, to check if the XCP version is XCP 2041 that does not support physical partition dynamic reconfiguration.

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2041
XCP1 (Reserve): 2041
```

c. Update the XCP firmware.

i. Execute the poweroff command to power off all the physical partitions.

```
XSCF> poweroff -a
```

Execute the showpparstatus command to check that all the physical partitions are powered off.

```
XSCF> showpparstatus -a
PPAR-ID          PPAR Status
0                Powered Off
```

ii. Execute the showhardconf command to check that [Status] of the master and standby XSCFs is "Normal."

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Service;
+ System_Power:Off; System_Phase:Cabinet Power Off;
Partition#0 PPAR_Status:Powered Off;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
:
```

iii. Execute the getflashimage command to import the XCP image file.

In the following example, the USB device is connected to the USB port (MAINTENANCE ONLY printed) on the XSCF unit panel (rear panel) of the master XSCF and the XCP image file is imported.

```
XSCF> getflashimage file:///mnt/share/scf-firm/xscf/user/scfadmin/BBXCP2220.tar.gz
 0MB received
 1MB received
 2MB received
...
86MB received
87MB received
88MB received
Download successful: 90539 Kbytes in 58 secs (1562.668 Kbytes/sec) Checking
file...
MD5: 2b89c06548205ce35a8ecb6c2321d999
```


When the normal end messages "Download successful: ..." and "MD5: ..." appear, they indicate that the importing of the XCP image file has ended.

Note - The message "Warning: About to delete existing old versions." may appear when the XCP image file is imported. This message asks you to confirm whether to delete an old XCP image file that has already been imported. If "Continue?" appears, enter "y" to continue import processing.

Note - If the message "Error:File is invalid or corrupt" appears after the XCP image file is imported, this indicates that the imported XCP image file is inappropriate. The XCP image file may have been destroyed. So, obtain the correct XCP image file and then import it.

iv. Execute the getflashimage -l command to check the version of the imported XCP image file.

```
XSCF> getflashimage -l
Existing versions:
      Version                Size   Date
BBXCP2220.tar.gz          92712351  Thu May 23 15:01:42 JST 2014
```

v. Execute the flashupdate -c check command to check whether the imported XCP image file can be used for update.

Execute the showresult command immediately after the execution of the flashupdate command. If the end value is 0, update is possible.

```
XSCF> flashupdate -c check -m xcp -s 2220
XCP update is started. [3600sec]
0XSCF>
XSCF> showresult
0
XSCF>
```

vi. Execute the flashupdate command to update the firmware.

```
XSCF> flashupdate -c update -m xcp -s 2220
The XSCF will be reset. Continue? [y|n]: y
XCP update is started. [3600sec]
0.....30.....60.....90.....120.....150.....180.....210.....240.....-
270.....300.....330.....360.....390.....420.....450.....480.....510.....|
540.....570.....600.....630.....660.....690.....720.....750.....780.....-
810.....840.....870.....900.....930
:
```

Here, the XSCF is rebooted and the XSCF session is disconnected. At this point, the XCP firmware update has not yet been completed.

Note - In the firmware work time, the update takes 45 minutes. On a system with a building block configuration, the update takes about 60 minutes, and automatic switching of the XSCF after the completion of the update takes about 10 minutes.

Note - To perform the update safely, do not operate the power supply of the physical partition until the message "XCP update has been completed" appears, to indicate that the XCP firmware has been updated.

vii. **Connect to the master XSCF again.**

Note - Immediately after an XSCF reboot on a system with a building block configuration, the master and standby XSCFs will be in the exact reverse of their original states. For example, if the firmware update is executed with the master XSCF of BB-ID 0 and a connection is then made to the XSCF again, BB-ID 1 enters the master state and BB-ID 0 enters the standby state.

Note - If an inherited IP address is set and is used for connection, a connection is automatically made to the master XSCF.

viii. **Execute the showbbstatus command to check that you have logged in to the master XSCF.**

Note - On a system with a building block configuration, if you logged in to the XSCF in the standby state, log in again to the master XSCF.

```
XSCF> showbbstatus
BB#00 (Master)
```

ix. **Execute the showlogs monitor command to check that XCP firmware update has been completed.**

If the message "XCP update has been completed" appears, XCP firmware update has been completed.

```
XSCF> showlogs monitor
Jan 23 16:42:57 5 M10-4S-0 Event: SCF:XCP update is started (XCP version=2220:
last version=2041)
Jan 23 16:44:52 M10-4S-0 Event: SCF:XSCF update is started (BBID=0, bank=0)
Jan 23 16:45:11 M10-4S-0 Event: SCF:XSCF writing is performed (BBID=0, XSCF
version=02220000)
:
Jan 23 16:54:05 01:50 M10-4S-0 Event: SCF:XSCF bank apply has been completed
(BBID=0, bank=0, XCP version=2220: last version=2041)
:
Jan 23 17:19:57 32:38 M10-4S-1 Event: SCF:CMU update has been completed
(BBID=0)
Jan 23 17:19:59 32:41 M10-4S-1 Even SCF:XCP update has been completed (XCP
version=2220: last version=2041)
```

Note - If the message "XCP update has been completed" does not appear, update has not yet been completed. Execute the showlogs monitor command again to check that update has

been completed.

x. Execute the showhardconf command to check that [Status] of the XSCF is "Normal."

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Service;
+ System_Power:Off; System_Phase:Cabinet Power Off;
Partition#0 PPAR_Status:Powered Off;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
:
```

xi. Execute the version command to check that the firmware is the new version.

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 2220
XCP1 (Reserve): 2220
```

Note - If the physical partition is powered on and the firmware is updated, the current bank of the CMU firmware will be new. If the physical partition is powered off and the firmware is updated, both the reserve bank and current bank of the CMU firmware will be new. For details on the versions of the CMU firmware associated with the XCP versions, see "Existing XCP Firmware Versions and Support Information" in the latest product notes.

5. Enable the physical partition dynamic reconfiguration function (PPAR DR function).

Note - To enable each SPARC M10-4S of the SPARC64 X processor and of the SPARC64 X+ processor to coexist in a single physical partition, execute the setpparmode command to set the CPU operational mode to compatible mode (SPARC64 X compatibility mode). For details, see ["2.6.3 Conditions for a Mixed Configuration With the SPARC64 X+ Processor and the SPARC64 X Processor Within a PPAR."](#)

Note - To use physical partition dynamic reconfiguration, it is necessary to enable the PPAR DR mode using the setpparmode command. However, if the PPAR DR mode is enabled, the logical domain configuration information is changed to the factory-default. So, the existing logical domain configuration information cannot be used. In this case, you must create new logical domain configuration information.

To use the existing logical domain configuration information, disable the PPAR DR mode. Suppose that the PPAR DR mode is enabled and that the power to the physical partition is turned on while the existing logical domain configuration information is used. Problems such as a hypervisor abort and a failure to start Oracle VM Server for SPARC normally will occur.

- a. **Execute the showpparmode command to reference the PPAR DR mode setting.**

The following example specifies physical partition number 0 (PPAR ID 0).

```
XSCF> showpparmode -p 0
Host-ID                :9007002b
Diagnostic Level       :min
Message Level          :normal
Alive Check            :on
Watchdog Reaction      :reset
Break Signal           :on
Autoboot(Guest Domain) :on
Elastic Mode           :off
IOreconfigure          :false
PPAR DR(Current)       :-
PPAR DR(Next)          :off
```

- b. **If the CPU operational mode is auto, execute the setpparmode command to set that mode to compatible.**

```
XSCF> setpparmode -p 0 -m cpumode=compatible
```

- c. **If the PPAR DR mode is disabled, execute the setpparmode command to enable that mode.**

```
XSCF> setpparmode -p 0 -m ppar_dr=on
```

- d. **Execute the showpparmode command to check that the PPAR DR mode has been enabled.**

```
XSCF> showpparmode -p 0
Host-ID                :9007002b
Diagnostic Level       :min
Message Level          :normal
Alive Check            :on
Watchdog Reaction      :reset
Break Signal           :on
Autoboot(Guest Domain) :on
Elastic Mode           :off
IOreconfigure          :false
PPAR DR(Current)       :-
PPAR DR(Next)          :on
```

6. **Execute the showsscp command to check whether the IP address of the SP to SP communication protocol (SSCP) is the default value or a user-specified value.**

```
XSCF> showsscp
```

When the IP address is the default value and you are using the default value of the IP address of the SPARC M10-4S to be expanded, go to the next step.

To set a user value, set the IP address with the `setsscp` command. Then, use the `applynetwork` command to apply and check the IP address of the SSCP of the SPARC M10-4S to be expanded. Afterwards, execute the `rebootxscf` command to complete the setting and then go to the next step. For details on the procedure, see "7.5.6 Applying network settings" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

7. Expand SPARC M10-4S.

- a. **Execute the `addfru` command to expand SPARC M10-4S according to the message.**

Note - If the IP address of the SSCP has not yet been set, executing the `addfru` command results in an error.

Note - If the `addfru` command is executed, the firmware version of the SPARC M10-4S to be expanded is automatically set to that of the SPARC M10-4S on which the master XSCF is running. In this example, the firmware version of the SPARC M10-4S (BB-ID 1) to be expanded is automatically set to that of the SPARC M10-4S (BB-ID 0) on which the master XSCF is running.

The following example expands BB#1.

```
XSCF> addfru
-----
Maintenance/Addition Menu
Please select the chassis including added FRU.
No.  FRU                Status
-----
 1  /BB#0                Normal
 2  /BB#1                Unmount
 3  /BB#2                Unmount
 4  /BB#3                Unmount
-----
Select [1-16|c:cancel] :2
Maintenance/Addition Menu
Please select the BB or a type of FRU to be added.
1. BB itself
2. PSU (Power Supply Unit)
-----
Select [1,2|c:cancel] :1
Maintenance/Addition Menu
Please select a FRU to be added.
No.  FRU                Status
-----
 1  /BB#1                Unmount
Select [1|b:back] :1
```

- b. **Mount SPARC M10-4S to be expanded, to the rack.**

While the previously mentioned `addfru` command is executed, if the message

"After the added device is connected with the system, please turn on the breaker of the BB#1." appears, mount the SPARC M10-4S to be expanded in the rack.

For details on the mounting procedure, see "3.4.1 Mounting the SPARC M10-4S in a rack" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

c. **Set the identification ID (BB-ID) of the SPARC M10-4S to be expanded.**

See "4.1 Setting the ID (BB-ID) Identifying a Chassis" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

d. **Connect the crossbar cable.**

Connect the crossbar cable between the existing SPARC M10-4S and the expanded SPARC M10-4S. For the connection cable routing figure and cable list, see "Appendix B Cable Connection Information on Building Block Configurations" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*. For details on direct connections between chassis, see "4.2 Connecting Cables (for Direct Connections between Chassis)" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

e. **Connect the XSCF BB control cable.**

Connect the existing SPARC M10-4S and expanded SPARC M10-4S through the XSCF BB control cable. For the routing figure and cable list of the XSCF BB control cables, see "Appendix B Cable Connection Information on Building Block Configurations" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*. For details on direct connections between chassis, see "4.2 Connecting Cables (for Direct Connections between Chassis)" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

f. **Connect the XSCF DUAL control cable.**

Connect the existing SPARC M10-4S and expanded SPARC M10-4S through the XSCF DUAL control cable. For details, see "4.2 Connecting Cables (for Direct Connections between Chassis)" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

g. **Connect the serial cable or LAN cable.**

Connect the serial cable to the XSCF serial port of the SPARC M10-4S to be expanded. Moreover, connect the cable to each LAN port of the XSCF-LAN, GbE port, and PCIe card. For details, see "5.1 Connecting Cables to the SPARC M10-4S" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

h. **Connect the input power to the SPARC M10-4S to be expanded.**

Connect the power cord of the SPARC M10-4S to be expanded to the input power.

i. **Enter "f" from the input screen of the addfru command.**

Enter "f" on the input screen for the addfru command executed in step a. to perform SPARC M10-4S expansion.

```
2) Please select[f:finish] :f
```

```
Waiting for BB#1 to enter install state.
```

```
[This operation may take up to 20 minute(s)] (progress scale
```

```

    reported in seconds)
0..... 30.... done

Waiting for BB#1 to enter ready state.
[This operation may take up to 45 minute(s)] (progress scale
reported in seconds)
0..... 30..... 60... done

Do you want to start to diagnose BB#1? [s:start|c:cancel] :

```

j. **Skip the diagnosis of the SPARC M10-4S to be expanded to exit the addfru command.**

Input "c" to the input screen of the addfru command on the master XSCF, and then skip the diagnosis processing for the SPARC M10-4S to be expanded. If "The addition of BB#1 has completed." appears, input "f" and then input "c" to exit the addfru command.

```

Do you want to start to diagnose BB#1? [s:start|c:cancel] :c

Diagnostic tests are about to be skipped.
Running diagnostic tests are strongly recommended before using BB#1.
Are you sure you want to skip testing? [y:yes|n:no] :y

-----
Maintenance/Addition Menu Status of the added FRU.

FRU                      Status
-----
/BB#1                    Normal
-----

[Warning:007]
Running diagnostic tests on BB#1 is strongly recommended after addfru has
completed.
The addition of BB#1 has completed.[f:finish] :f

-----
Maintenance/Addition Menu
Please select the chassis including added FRU.

No. FRU                      Status
---
1   /BB#0                    Normal
2   /BB#1                    Normal
-----

Select [1,2|c:cancel] :c

```

8. **Diagnose the expanded SPARC M10-4S.**

a. **Execute the testsb command to perform a diagnosis test for the expanded SPARC M10-4S.**

Execute the `testsb` command to perform the diagnosis test. Specify the physical system board (PSB) number of the expanded SPARC M10-4S and check the initial diagnosis and connection I/O.

```
XSCF> testsb -v -p -s -y 01-0
Initial diagnosis is about to start, Continue?[y|n] :y
PSB#01-0 power on sequence started.
```

If an error is displayed, see "A.2.4 Checking diagnosis results" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

- b. **Specify the `-t` option with the `diagxbu` command to perform the crossbar cable diagnosis test of the existing SPARC M10-4S and the expanded SPARC M10-4S.**

The following example specifies the identification ID "00" (BB-ID 00) of the existing SPARC M10-4S and the identification ID "01" (BB-ID 01) of the expanded SPARC M10-4S to diagnose the crossbar cable connection status.

```
XSCF> diagxbu -y -b 00 -t 01
XBU diagnosis is about to start, Continue?[y|n] :y
Power on sequence started. [7200sec]
0.....30.....60..end
XBU diagnosis started. [7200sec]
0.....30.....60.....90.....120.....150.....180.....210.....240...../
270.....300.....330.....360.....390.....420.....450.....480.....510.....\
540.....570.....600.....630.....660.....690.....720.....750.....780.....\
810.....840.....870.....900.....930..end
completed.
Power off sequence started. [1200sec]
0.....30.....60.....90.....120.....150.....180..end
completed.
```

- c. **Execute the `showlogs error` command to check that no error is displayed.**

```
XSCF> showlogs error
```

If an error is displayed, see "A.2.4 Checking diagnosis results" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

- d. **Execute the `showhardconf` command to check the configuration and status of the expanded SPARC M10-4S.**

Execute the `showhardconf` command to check the hardware configuration (CPU, memory, or the like) of the expanded SPARC M10-4S, and then check that Status of each unit is Normal.

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Service;
+ System_Power:Off; System_Phase:Cabinet Power Off;
Partition#0 PPAR_Status:Powered Off;
```



```

BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP1236052K ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
+ Freq:3.000 GHz; Type:0x10;
+ Core:16; Strand:2;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
+ FRU-Part-Number: CA20393-B50X A2 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
:

```

9. **Set the XSCF network of the expanded SPARC M10-4S.**

Set the XSCF network for the expanded SPARC M10-4S. For details, see "7.5.2 Setting an Ethernet (XSCF-LAN) IP address" and "7.5.3 Setting a takeover IP address" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*. After making this setting, execute the `applynetwork` command to apply and check the setting. Afterwards, execute the `rebootxscf` command to complete the setting and then go to the next step. For details on the procedure, see "7.5.6 Applying network settings" in the *Fujitsu M10-4S/SPARC M10-4S Installation Guide*.

10. **Set mirror mode for the memory of the expanded SPARC M10-4S.**

If you are not using memory mirror mode, this step is not necessary.

For details on memory mirror mode, see "14.1 Configuring Memory Mirroring" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

a. **If you are making a mirror configuration of the memory in the expanded SPARC M10-4S, use the `setupfru` command to set memory mirror mode.**

The following example places all the memories for the CPUs under the SPARC M10-4S physical system board (PSB 01-0) in memory mirror mode.

```
XSCF> setupfru -m y sb 01-0
```

b. **Execute the `showfru` command to check the mirror mode setting of the memory.**

```

XSCF> showfru sb 01-0
Device Location Memory Mirror Mode
sb      01-0
  cpu 01-0-0   yes
  cpu 01-0-1   yes
  cpu 01-0-2   yes

```

11. **Register the system board of the expanded SPARC M10-4S in the physical partition configuration information.**
 - a. **Execute the showpctl command to check the physical partition configuration information.**

```
XSCF> showpctl -p 0
PPAR-ID      LSB      PSB      Status
00
              00      00-0     Powered Off
```

- b. **Execute the setpctl command to register the system board in the physical partition configuration information.**

Execute the setpctl command to register the expanded SPARC M10-4S system board in the physical partition configuration information for the built-in destination.

In the following example, physical system board (PSB) 01-0 is mapped to logical system board (LSB) 01 of physical partition 0.

```
XSCF> setpctl -p 0 -a 01=01-0
```

- c. **Execute the showpctl command to check the physical partition configuration information.**

```
XSCF> showpctl -p 0
PPAR-ID      LSB      PSB      Status
00
              00      00-0     Powered Off
              01      01-0
```

12. **Register the CPU Activation key to assign CPU core resources.**
 - a. **Execute the showcodusage command to check the CPU core resource information.**

Execute the showcodusage command to check whether the physical partition contains an assignable CPU core resource.

As shown here, the system has 128 mounted CPU core resources and 128 registered CPU Activations, 64 of the CPU core resources are in use, and the number of CPU Activations currently not in use is 64.

```
XSCF> showcodusage -p resource
Resource In Use Installed CoD Permitted Status
-----
PROC          64      128      128 OK: 64 cores available
```

Note:

Please confirm the value of the "In Use" by the ldm command of Oracle VM Server for SPARC.

The XSCF may take up to 20 minutes to reflect the "In Use" of logical domains.

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and add the CPU Activation keys. For details on how to add a CPU Activation key, see "5.3 Adding CPU Core Resources" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

b. Execute the showcod command to check the CPU core resource information for the physical partition.

Execute the showcod command to check information on the CPU core resources assigned to the physical partition.

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 64
```

If the assigned resources are insufficient, execute the setcod command to assign CPU core resources to the physical partition.

The following example adds 64 CPU core resources to physical partition 0.

```
XSCF> setcod -p 0 -s cpu -c add 64
PROC Permits assigned for PPAR 0 : 64 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

```
XSCF> setcod -s cpu
```

When the CPU core resources have been assigned, execute the showcod command again to check the information for the CPU core resources assigned to the physical partition.

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 128
```

13. Incorporate the system board (PSB<BB>) of the expanded SPARC M10-4 into the physical partition.

- a. **Return to the XSCF shell, and then execute the showboards command to check the PSB status.**

Execute the showboards command to check that the PSB status of the expanded SPARCM10-4S is "SP" (system board pool).

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	n	n	n	Passed	Normal
01-0	SP	Available	n	n	n	Passed	Normal

- b. **Execute the addboard command to assign the PSB to the physical partition.**

Execute the addboard -c assign command to assign the PSB to the physical partition.

```
XSCF> addboard -c assign -p 0 01-0
PSB#01-0 will be assigned into PPAR-ID 0. Continue?[y|n] :y
XSCF>
```

Note - If an error message appears during execution of the addboard command, see "[C.1.1 addboard](#)," and then identify the error and take corrective action.

- c. **Execute the showresult command to check the exit status of the addboard command that was just executed.**

An end value of 0 indicates the normal termination of the addboard command.

If the end value is other than 0 or if an error message is displayed upon executing the addboard command, it indicates abnormal termination of the addboard command. By referring to "[C.1.1 addboard](#)" based on the error message, identify the error and then take corrective action.

```
XSCF> showresult
0
```

- d. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the added SPARC M10-4S is in the Assigned state and that the Pwr, Conn, and Conf columns all show "n."

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	n	n	n	Passed	Normal
01-0	00(01)	Assigned	n	n	n	Passed	Normal

14. **Power on the physical partition.**

Execute the showdomainconfig command to check that the configuration

information for the logical domains to be started is set to the factory-default. The following example specifies physical partition number 0 (PPAR ID 0).

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) : factory-default
(Next) : factory-default
-----
Index :1
config_name :factory-default
domains :1
date_created:-
-----
Index :2
config_name :ldm-set1
domains :8
date_created:'2012-08-0811:34:56'
-----
Index :3
config_name :ldm-set2
domains :20
date_created:'2012-08-0912:43:56'
:
```

Execute the poweron command to power on the physical partitions.

```
XSCF> poweron -p 0
```

15. Using the XML file, reconfigure the logical domain.

After completing the startup of the physical partition, reconfigure the logical domain from the XML file saved in step 1.

a. Check that the system has started with the factory-default settings for the control domain.

Execute the ldm list-spconfig command to check that [current] is displayed on the factory-default line.

The following shows an example of command execution.

```
# ldm list-spconfig
factory-default [current]
ldm-set1
ldm-set2
ldm-set3
```

b. Using the ldm init-system command, apply the logical domain configuration information from the XML file.

Execute the ldm init-system command to apply the saved XML file setting.

Then, execute the shutdown command to restart the control domain.

```
# ldm init-system -i /ldm-set1.xml
```

Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary domain reboots, at which time the new configuration for the primary domain will also take effect.

```
# shutdown -y -g0 -i6
```

- c. **To enable physical partition dynamic reconfiguration when the version of Oracle VM Server for SPARC is earlier than 3.2, adjust the memory size of the existing logical domain.**

Execute the `ldm list-domain` command or the like to check that the logical domains have been reconfigured from the XML file.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	28G	0.0%	6m
guest0	inactive	-----		64	64G		
root-dom0	inactive	-----		32	32G		

To enable the use of physical partition dynamic reconfiguration, set the memory size for each logical domain to "a multiple of the number of CPU cores of the logical domain x 256 MB."

To re-set the memory size of a logical domain other than the control domain, use the `ldm set-memory` command.

To re-set the memory size of the control domain, do so as follows: First enter delayed reconfiguration mode with the `ldm start-reconf` command, set the same number of cores with the `ldm set-core` command, and then re-set the memory size with the `ldm set-memory` command. Finally, restart Oracle Solaris.

In this example, since the number of CPU cores of the control domain is 32, set a multiple of $32 \times 256 \text{ MB} = 8192 \text{ MB}$.

First, obtain a multiple of 8192 MB to make the value closer to the original setting value (58 GB = 59392 MB). The result is $59392/8192 = 7.25$. So, round it down to 7. As a result, the memory size to be reassigned to the control domain is $8192 \text{ MB} \times 7 = 56 \text{ GB}$.

The following example shows the execution of the command for re-setting the control domain memory to 56 GB.

```
# ldm start-reconf primary
```

Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary domain reboots, at which time the new configuration for the primary domain will also take effect.

```
# ldm set-core 32 primary
```

Notice: The primary domain is in the process of a delayed reconfiguration.

Any changes made to the primary domain will only take effect after it reboots.

```
# ldm set-memory 56G primary
```

Notice: The primary domain is in the process of a delayed reconfiguration. Any changes made to the primary domain will only take effect after it reboots.

```
# shutdown -i6 -g0 -y
```

Execute the `ldm list-domain` command to check that the memory size (MEMORY) has been set properly.

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	64	56G	0.0%	6m
guest0	inactive	-----		64	64G		
root-dom0	inactive	-----		32	32G		

Execute the `ldm bind-domain` command to bind each logical domain, and then start each logical domain by executing the `ldm start-domain` command. The following shows command execution examples.

```
# ldm bind-domain root-dom0
# ldm bind-domain guest0
# ldm start-domain root-dom0
LDom root-dom0 started.
# ldm start-domain guest0
LDom guest0 started.
```

Execute the `ldm list-domain` command, and then confirm that each logical domain has started. Check that [STATE] is "active" and that the second character from the left of the [FLAGS] string is "n."

```
# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-n-cv-	UART	32	28G	0.0%	18m
guest0	active	-n----	5100	64	64G	0.1%	2m
root-dom0	active	-n-v-	5000	32	32G	0.0%	3m

16. Add a new logical domain.

Configure a new logical domain and assign the resources of the added SPARC M10-4S. The following example performs the procedure for creating a mirroring configuration according to the configuration example. In other words, add guest domain `guest1` and root domain `root-dom1` to create a redundant configuration of the virtual I/O for each guest domain. Furthermore, create a mirroring configuration of the system volume of the control domain with an internal hard disk.

a. Execute the `ldm list-devices` command to check the state of the added

hardware resources.

When the `ldm list-devices` command is executed, any unassigned hardware resources are listed. Check that the hardware resources have been added. The following example checks the number of unassigned CPU cores by counting the number of lines of "free = 100" from the output of the `ldm list-devices -a -p core` command.

```
# ldm list-devices -a -p core | grep "free=100" | wc -l
64
```

For memory, check the mapping between the mounted SPARC M10-4S location and physical address in memory from [Contains Modules and Base Address] of memory information output with the `prtdiag` command. Next, check the physical address and size of unassigned memory with the `ldm-list-devices memory` command.

The following example enables you to determine that unassigned memory has been added. This is clear from the fact that the head of the physical address of BB-ID 1 memory is 0x700000000000 to 0x760000000000 and that the addresses in this range are displayed with the `ldm list-devices memory` command.

```
# prtdiag
....
===== Physical Memory Configuration =====
Segment Table:
-----
Base      Segment  Interleave  Bank      Contains
Address   Size       Factor      Size      Modules
-----
....
0x760000000000 32 GB    4           8 GB     /BB1/CMUL/CMP0/MEM00A
....
0x740000000000 32 GB    4           8 GB     /BB1/CMUL/CMP1/MEM10A
....
0x720000000000 32 GB    4           8 GB     /BB1/CMUU/CMP0/MEM00A
....
0x700000000000 32 GB    4           8 GB     /BB1/CMUU/CMP1/MEM10A
....
# ldm list-devices memory
MEMORY
  PA              SIZE              BOUND
  0x700000000000  32G
  0x720000000000  32G
  0x740000000000  32G
  0x760050000000  31488M
....
```

b. Create a new logical domain to which to assign the CPU cores and memory resources.

However, when a domain is configured, if the number of CPU cores is first

set with the ldm set-core command and then the memory size to be assigned is set with the ldm set-memory command, the memory blocks assigned to the logical domain become difficult to distribute (fragmentation) and memory placement becomes easy to consider at SPARC M10-4S release.

The following example adds the root domain (root-dom1) and guest domain (guest1) to which CPU cores and memory are assigned.

```
# ldm add-domain root-dom1
# ldm add-domain guest1
# ldm set-core 16 root-dom1
# ldm set-core 32 guest1
# ldm set-memory 32G root-dom1
# ldm set-memory 64G guest1
```

Note - To perform physical partition dynamic reconfiguration when the version of Oracle VM Server for SPARC is earlier than 3.2, set "a multiple of number of CPU cores x 256 MB" for the memory size to be assigned to each logical domain.

c. **Assign the root complex to the root domain.**

Assign the root complexes of the added SPARC M10-4S to the added root domain (root-dom1). Execute the ldm list-io command to check the root complexes to be added to the root domain (root-dom1). A device whose NAME begins with "/BB1" is on BB-ID 1 and the string shown in BUS of that device indicates the root complex.

```
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8		
PCIE9	BUS	PCIE9		
PCIE10	BUS	PCIE10		
PCIE11	BUS	PCIE11		
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13		
PCIE14	BUS	PCIE14		
PCIE15	BUS	PCIE15		
....				
/BB1/CMUL/NET0	PCIE	PCIE8		
/BB1/CMUL/SASHBA	PCIE	PCIE8		
/BB1/PCI0	PCIE	PCIE9		
/BB1/PCI3	PCIE	PCIE10		
/BB1/PCI4	PCIE	PCIE10		

/BB1/PCI7	PCIE	PCIE11
/BB1/PCI8	PCIE	PCIE11
/BB1/CMUL/NET2	PCIE	PCIE12
/BB1/PCI1	PCIE	PCIE13
/BB1/PCI2	PCIE	PCIE13
/BB1/PCI5	PCIE	PCIE14
/BB1/PCI6	PCIE	PCIE14
/BB1/PCI9	PCIE	PCIE15
/BB1/PCI10	PCIE	PCIE15

Execute the `ldm add-io` command to add the root complexes to be added to the root domain (root-dom1). Then, execute the `ldm list-io` command to perform a check.

The following example adds BB1 root complexes (PCIE9, PCIE10, PCIE11, PCIE13, PCIE14, and PCIE15) other than root complexes (PCIE8 and PCIE12) of the above-mentioned on-board devices (/BB1/CMUL/NET0, /BB1/CMUL/SASHBA, and /BB1/CMUL/NET2), to root-dom1.

```
# ldm add-io PCIE9 root-dom1
# ldm add-io PCIE10 root-dom1
....
# ldm list-io
```

NAME	TYPE	BUS	DOMAIN	STATUS
PCIE0	BUS	PCIE0	primary	IOV
PCIE1	BUS	PCIE1	root-dom0	IOV
PCIE2	BUS	PCIE2	root-dom0	IOV
PCIE3	BUS	PCIE3	root-dom0	IOV
PCIE4	BUS	PCIE4	primary	IOV
PCIE5	BUS	PCIE5	root-dom0	IOV
PCIE6	BUS	PCIE6	root-dom0	IOV
PCIE7	BUS	PCIE7	root-dom0	IOV
PCIE8	BUS	PCIE8	root-dom1	IOV
PCIE9	BUS	PCIE9		
PCIE10	BUS	PCIE10	root-dom1	IOV
PCIE11	BUS	PCIE11	root-dom1	IOV
PCIE12	BUS	PCIE12		
PCIE13	BUS	PCIE13	root-dom1	IOV
PCIE14	BUS	PCIE14	root-dom1	IOV
PCIE15	BUS	PCIE15	root-dom1	IOV

d. Assign the root complex to the root domain.

Execute the `ldm set-vconsole` command to assign the console device to the root domain (root-dom1).

```
# ldm set-vconsole port=5001 vcc0 root-dom1
```

Execute the `ldm bind-domain` and `ldm start-domain` commands to bind again information, including that on configured root complexes, into the root domain (root-dom1) and to start the root domain.

```
# ldm bind-domain root-dom1
# ldm start-domain root-dom1
LDom root-dom1 started
```

Execute the telnet command to connect to the console of the root domain.

```
# telnet localhost 5001
....
```

e. Install Oracle Solaris in the root domain.

Install Oracle Solaris in the started root domain. For details on how to install Oracle Solaris, see the documentation related to Oracle Solaris.

f. Check the physical device to be assigned to the virtual service of the root domain.

After installing Oracle Solaris, log in to the root domain and then execute the format command to check the name of the device to be assigned to the virtual disk service. Executing the ldm list-io -l command on the control domain before logging in to the root domain displays the device path associated with the PCIe endpoint. So, you can check the path through which the disk is connected, by comparing the name of that device path with the name of the device path of the disk displayed with the format command.

```
# ldm list-io -l
```

NAME	TYPE	BUS	DOMAIN	STATUS
----	----	---	-----	-----
....				
/BB1/PCI0	PCIE	PCIE9	root-dom1OCC	
[pci@8900/pci@4/pci@0/pci@0]				
/BB1/PCI3	PCIE	PCIE10	root-dom1OCC	
[pci@8a00/pci@4/pci@0/pci@0]				
/BB1/PCI4	PCIE	PCIE10	root-dom1OCC	
[pci@8a00/pci@4/pci@0/pci@8]....				

```
# telnet localhost 5000
....
root-dom1# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
    0. c3t500000E01BDA70B2d0 <FUJITSU-MBB2147RC-3703 cyl 14087 alt 2 hd 24
sec 848>
        / pci@8900/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w500000e01bda70b2,0
    1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
        /pci@8a00/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
....
Specify disk (enter its number): ^C
```

Subsequently, executing the dladm show-phys -L command enables you to check the Ethernet interface name and physical location as viewed from the top of the root domain.

```

root-dom0# dladm show-phys -L
LINK          DEVICE          LOC
net0          igb0            /BB1/PCI0
net1          igb1            /BB1/PCI0
net2          igb2            /BB1/PCI0
net3          igb3            /BB1/PCI0

```

g. Assign the virtual I/O service to the root domain.

Return to the control domain to add the virtual I/O service to the root domain (root-dom1). For details on the I/O device setting, see the *Oracle VM Server for SPARC Administration Guide* provided by Oracle Corporation.

The following example adds the virtual disk service (vds1) and virtual switch network interfaces (vsw10 and vsw11) to the added root domain (root-dom1).

```

# ldm add-vdiskserver vds1 root-dom1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol10@vds1
# ldm add-vdiskserverdevice /dev/dsk/cXtXXXXXXXXXXXXXXXXdXsX vol11@vds1
# ldm add-vswitch net-dev=net1 vsw10 root-dom1
# ldm add-vswitch net-dev=net2 vsw11 root-dom1

```

h. Assign the virtual I/O device to the guest domain.

Add the virtual I/O device that uses the added virtual I/O service, to each guest domain.

The following example adds the virtual I/O devices (vdisk10, vnet10) that use the virtual I/O service of root-dom1 to the existing guest domain (guest0). It also adds the virtual I/O devices (vdisk1, vdisk11, vnet1, and vnet11) that use the virtual I/O services of root-dom0 and root-dom1 and the virtual console (vcons), to a guest domain (guest1) to be newly added.

```

# ldm add-vdisk vdisk10 vol10@vds1 guest0
# ldm add-vnet vnet10 vsw10 guest0
# ldm add-vdisk vdisk1 vol11@vds0 guest1
# ldm add-vdisk vdisk11 vol11@vds1 guest1
# ldm add-vnet vnet1 vsw1 guest1
# ldm add-vnet vnet11 vsw11 guest1
# ldm set-vconsole port=5101 guest1

```

i. Start the added guest domain.

Execute the `ldm bind-domain` and `ldm start-domain` commands to start the added guest domain (guest1).

```

# ldm bind-domain guest1
# ldm start-domain guest1
LDom guest1 started

```

j. Install Oracle Solaris in the added guest domain.

Install Oracle Solaris in the added guest domain. For details on how to install

Oracle Solaris, see the documentation related to Oracle Solaris.

k. **Establish a redundant configuration for the virtual I/Os of each guest domain.**

The following describes an example of a procedure for making two virtual network interfaces (vnets), assigned to the guest domain (guest0), into a redundant configuration, using IPMP. For details on the procedures for other redundant configurations, see the documentation for the software of each redundant configuration.

Log in to guest domain guest0. In the example below, the `ldm list-domain` command is used to check the port number of the console of guest0, and then the `telnet` command is used to connect to port number 5100.

```
# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART    16    14G     0.0%  8h 7m
guest0        active    -n----   5100    32    32G     0.0%  20s
guest1        active    -n----   5101    32    32G     0.0%  19s
root-dom0     active    -n--v-   5000    24    24G     0.0%  43s
root-dom1     active    -n--v-   5001    24    24G     0.0%  20s

# telnet localhost 5100
....
guest0 console login: root
Password:
...
guest0#
```

Execute the `dladm show-phys` command to check that the virtual network devices are visible. In the example below, it is possible to refer to virtual network devices as network interfaces (net0 and net1).

```
guest0# dladm show-phys
LINK          MEDIA          STATE    SPEED  DUPLEX    DEVICE
net0          Ethernet      up        0      unknown  vnet0
net1          Ethernet      up        0      unknown  vnet1
```

Execute the `ipadm show-if` command to check that net0 and net1 are not displayed.

```
guest0# ipadm show-if
IFNAME  CLASS    STATE    ACTIVE  OVER
lo0     loopback ok      yes     --
```

Execute the `ipadm create-ip` command to create IP interfaces net0 and net1, and then use the `ipadm show-if` command to check that they have been created normally.

```

guest0# ipadm create-ip net0
guest0# ipadm create-ip net1
guest0# ipadm show-if

```

IFNAME	CLASS	STATE	ACTIVE	OVER
lo0	loopback	ok	yes	--
net0	ip	down	no	--
net1	ip	down	no	--

Execute the `ipadm create-ipmp` command to create IPMP interface `ipmp0`, and then execute the `ipadm add-ipmp` command to add IP interfaces `net0` and `net1` to the IPMP group.

```

guest0# ipadm create-ipmp ipmp0
guest0# ipadm add-ipmp -i net0 -i net1 ipmp0

```

Execute the `ipadm create-addr` command to assign an IP address to IPMP interface `ipmp0`, and then execute the `ipadm show-addr` command to check the setting. The following example assigns a fixed IP address.

```

guest0# ipadm create-addr -T static -a local=xx.xx.xx.xx/24 ipmp0/v4
guest0# ipadm show-addr

```

ADDROBJ	TYPE	STATE	ADDR
lo0/v4	static	ok	127.0.0.1/8
ipmp0/v4	static	ok	xxx.xxx.xxx.xxx/24
lo0/v6	static	ok	:::1/128

Execute the `ipadm set-ifprop` command to set a standby interface, and then execute the `ipmpstat -i` command to check the IPMP configuration.

```

guest0# ipadm set-ifprop -p standby=on -m ip net1
guest0# ipmpstat -i

```

INTERFACE	ACTIVE	GROUP	FLAGS	LINK	PROBE	STATE
net1	no	ipmp0	is-----	up	disabled	ok
net0	yes	ipmp0	--mbM--	up	disable	

Perform the same procedure for the other guest domains (guest1 in the example).

1. Establish a redundant configuration for the system volume of the control domain.

The on-board device of the added SPARC M10-4S is assigned to the control domain to create a redundant configuration. Switch the control domain to delayed reconfiguration mode.

```

# ldm start-reconf primary

```

Initiating a delayed reconfiguration operation on the primary domain. All configuration changes for other domains are disabled until the primary domain reboots, at which time the new configuration for the primary domain will also take effect.

Add the root complexes (PCIE8 and PCIE12) of the added SPARC M10-4S to the control domain and then restart Oracle Solaris.

```
# ldm add-io PCIE8 primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
# ldm add-io PCIE12 primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
# shutdown -i6 -g0 -y
....
```

After restarting Oracle Solaris, establish a redundant configuration for the system volume. For details on how to set the redundant configuration, see the documentation related to the software for each redundant configuration. The following example checks the disk drive with the format command and then establishes a mirror configuration with ZFS.

```
# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
  0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
    sec 625>
      /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
      /dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
  1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
    sec 625>
      /pci@8800/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393a803b13e,0
      /dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

        NAME                                STATE      READ WRITE CKSUM
        rpool                                ONLINE     0     0     0
          c2t50000393E802CCE2d0s0            ONLINE     0     0     0

errors: No known data errors
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
```

Execute the `zpool status` command, and then confirm that the mirroring configuration has been established. Use the `zpool status` command to confirm whether synchronization processing (resilver) has been completed. The following example shows that synchronization processing is in progress.

```
# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
       continue to function in a degraded state.
action: Wait for the resilver to complete.
       Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Mon Jan 27 15:55:47 2014
      21.1G scanned out of 70.6G at 120M/s, 0h7m to go
      21.0G resilvered, 29.84% done
config:

          NAME                                STATE      READ  WRITE CKSUM
rpool
  mirror-0
    c2t50000393E802CCE2d0s0  ONLINE        0      0      0
    c3t50000393A803B13Ed0s0  DEGRADED      0      0      0 (resilvering)

errors: No known data errors
```

Once synchronization processing is complete, the displayed screen will be as follows:

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:

          NAME                                STATE      READ  WRITE CKSUM
rpool
  mirror-0
    c2t50000393E802CCE2d0s0  ONLINE        0      0      0
    c3t50000393A803B13Ed0s0  ONLINE        0      0      0

errors: No known data errors
```

m. Check and adjust the memory placement of each logical domain.

Execute the `ldm list-devices -a memory` command to check the memory block placement.

```
# ldm list-devices -a memory
MEMORY
  PA          SIZE          BOUND
  0x700000000000 32G          root-dom1
  0x720000000000 32G          guest1
```


0x740000000000	32G	guest1
0x760000800000	1272M	_sys_
0x760050000000	16G	primary
0x760450000000	10G	primary
0x7606d0000000	4864M	
0x780000000000	32G	guest0
0x7a0000000000	32G	guest0
0x7c0000000000	256M	primary
0x7c0010000000	3584M	root-dom0
0x7c00f0000000	24832M	primary
0x7c0700000000	4G	root-dom0
0x7e0000800000	1272M	_sys_
0x7e0050000000	512M	_sys_
0x7e0070000000	256M	_sys_
0x7e0080000000	25088M	root-dom0
0x7e06a0000000	3584M	primary
0x7e0780000000	2G	primary

Suppose that memory blocks assigned to a logical domain or unassigned memory blocks are divided into small fragments. You can collect the fragmented memory blocks into a large continuous region by unbinding the logical domain and then binding it again. This facilitates adding or deleting the SPARC M10-4S later. For details on the procedure, see step 11-j in "[A.2.2 Example of the Physical Partition Configuration Procedure](#)."

n. Save logical domain configuration information.

Execute the `ldm list-domain` command to check that the configured logical domain has started.

```
# ldm list-domain
NAME          STATE      FLAGS    CONS  VCPU  MEMORY  UTIL  UPTIME
primary       active    -n-cv-   UART   64    56G     64%   2h 54m
guest0        active    -n----   5100   64    64G     42%   2h 54m
guest1        active    -n----   5101   64    64G     11%   2h 54m
root-dom0     active    -n--v-   5000   32    32G     20%   2h 54m
root-dom1     active    -n--v-   5001   32    32G     18%   2h 54m
```

Execute the `ldm add-spconfig` command to save the logical domain configuration information to the XSCF.

```
# ldm add-spconfig ldm-set4
# ldm list-spconfig
factory-default
ldm-set1
ldm-set2
ldm-set3
ldm-set4 [current]
```

o. If there are multiple items of logical domain configuration information to be restored, return the current configuration to the factory-default configuration and then repeat steps 15 and later.

This item gives an example of the procedure for returning the current configuration to the factory-default configuration. Execute the `showdomainconfig` command from the XSCF to check the configuration information for the logical domain to be started.

```
XSCF> showdomainconfig -p 0
PPAR-ID :0
Booting config
(Current) :ldm-set1
(Next) :ldm-set1
-----
Index :1
config_name :factory-default
domains :1
date_created:-
-----
Index :2
config_name :ldm-set1
domains :8
date_created:'2012-08-0811:34:56'
-----
....
```

Execute the `setdomainconfig` command to specify factory-default Index1 in physical partition number 0 (PPAR ID 0).

```
XSCF> setdomainconfig -p 0 -i 1
```

Execute the `poweroff` and then the `poweron` commands to restart the physical partition. Once startup is complete, repeat steps 15 and later.

```
XSCF> poweroff -p 0
...
XSCF> poweron -p 0
...
```

A.6 For Active Replacement of a System Board in a Configuration with Only the Control Domain (for Oracle VM Server for SPARC 3.2 or Later)

Dynamic PCIe bus assignment becomes available in Oracle VM Server for SPARC 3.2 or later. So, you can add or delete a PCIe root complex without delayed

reconfiguration on the control domain and the root domain. Therefore, if it is configured with only the control domain (factory-default configuration), you can add or delete the system board without stopping or restarting Oracle Solaris.

This section describes the procedure example for releasing one of the SPARC M10-4Ss in a physical partition with two SPARC M10-4Ss set to the factory-default configuration. Note that all the CPU cores and memory are assigned to the control domain if it is set to the factory-default configuration. This description also applies to SPARC M12.

Note - If the control domain or the root domain is Oracle Solaris 10, this configuration is not available because dynamic PCIe bus addition and deletion is disabled.

A.6.1 Configuration Example

This section describes an example of configuring a control domain consisting of two SPARC M10-4S, i.e. two system boards (PSB<BB>) (2BB configuration).

To enable physical partition dynamic reconfiguration, it is necessary to establish a redundant configuration for the system volume disks and the network interfaces of the control domain by connecting the physical I/O of each SPARC M10-4S. This is done to make it possible to continue operation with the physical I/O of the remaining SPARC M10-4Ss when a SPARC M10-4S is removed.

The following is a schematic diagram of a configuration that satisfies the above conditions. Moreover, the I/O configuration of each logical domain is simplified.

Figure A-8 Configuration Example of a 2BB Configuration Where All the Resources are Assigned

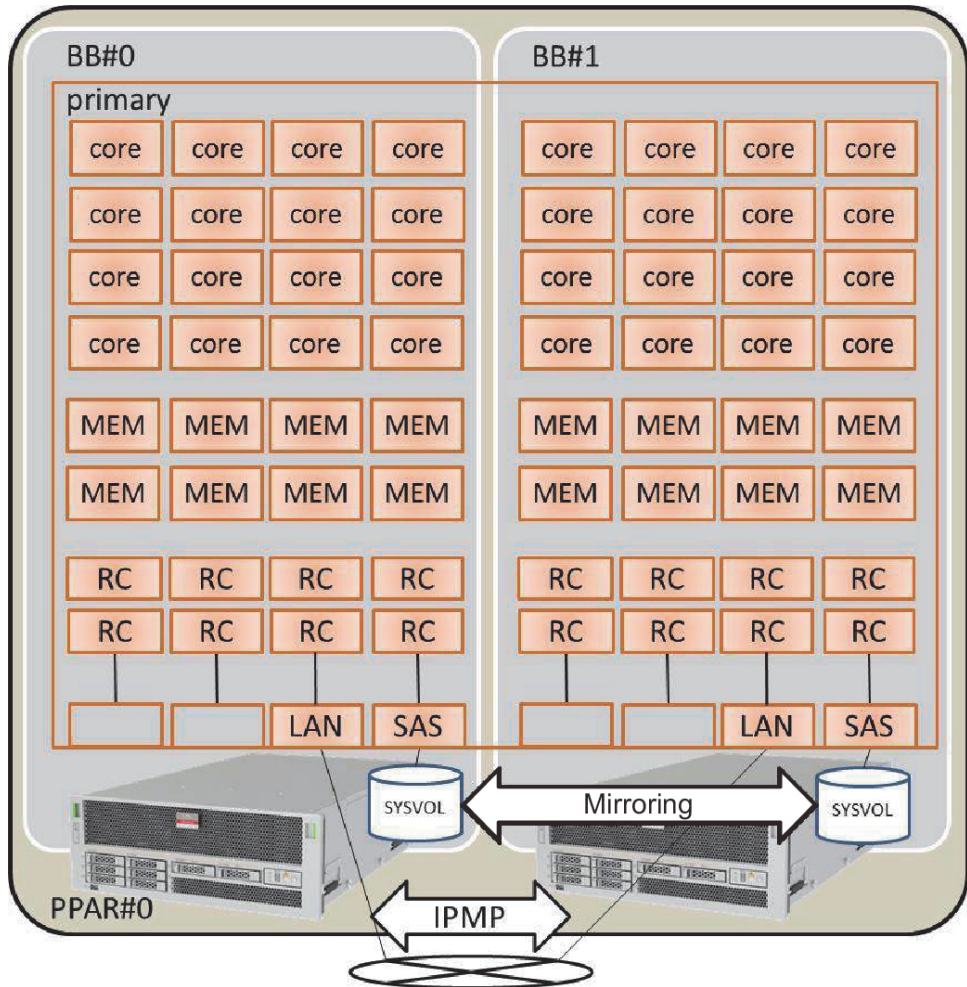


Figure A-8 shows a configuration example of a 2BB configuration where all the CPU cores, memory, and physical I/Os are assigned to the control domain. Therefore, the CPU cores and memory mounted on the SPARC M10-4S to be disconnected through physical partition dynamic reconfiguration are to be deleted, being unable to be moved at SPARC M10-4S release.

A.6.2 Example of the Physical Partition Configuration Procedure

1. **Log in to the master XSCF.**
Execute the `showbbstatus` command to check that the XSCF to which you have logged in is the master XSCF.

If you have logged in to a standby XSCF, log out and then log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

2. **Set mirror mode for memory.**

- a. **Execute the showfru command to check mirror mode of the memory on the SPARC M10-4S system board (PSB<BB>).**

The following example displays the setting information for devices under the physical system board (PSB 00-0).

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
  cpu 00-0-0 no
  cpu 00-0-1 no
  cpu 00-0-2 no
  cpu 00-0-3 no
```

- b. **To use the memory mirror mode, execute the setupfru command to set that mode.**

If you are not using memory mirror mode, this step is not necessary.

For details on memory mirror mode, see "14.1.1 Overview of Memory Mirroring" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

The following example places all the memories for the CPUs under the system board (PSB 00-0) in memory mirror mode.

```
XSCF> setupfru -m y sb 00-0
```

Execute the showfru command to check the setting of memory mirror mode.

```
XSCF> showfru sb 00-0
Device Location Memory Mirror Mode
sb 00-0
  cpu 00-0-0 yes
  cpu 00-0-1 yes
  cpu 00-0-2 yes
  cpu 00-0-3 yes
```

3. **Create the physical partition configuration information.**

- a. **Execute the showpctl command to check the physical partition configuration information.**

```
XSCF> showpctl -p 0
PPAR-ID LSB PSB Status
```

b. **Execute the setpctl command to register the system board in the physical partition configuration information.**

Execute the setpctl command to register the SPARC M10-4S system board in the physical partition configuration information for the built-in destination.

In the following example, physical system boards (PSBs) 00-0 and 01-0 are mapped to logical system boards (LSBs) 00 and 01 of physical partition 0.

```
XSCF> setpctl -p 0 -a 00=00-0 01=01-0
```

c. **Execute the showpctl command to check the physical partition configuration information.**

Check the physical partition configuration information.

```
XSCF> showpctl -p 0
PPAR-ID  LSB      PSB      Status
00
          00      00-0
          01      01-0
```

Specify the -v option to display details on the configuration policy, IO nullification option (No-IO), and memory nullification option (No-Mem).

```
XSCF> showpctl -v -p 0
PPAR-ID  LSB      PSB      Status  No-Mem  No-IO  Cfg-policy
00
          00      00-0      Running
          01      01-0
                                False   False   System
                                False   False
```

Use the setpctl command to change the setting of the configuration policy, I/O nullification option (No-IO), and memory nullification option (No-Mem).

For details on the setpctl command, see the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual*.

4. **Assign a system board (PSB<BB>) to a physical partition.**

a. **Execute the showboards-a command to check the PSB status.**

Execute the showboards -a command to check that each PSB status is "SP" (system board pool).

```
XSCF> showboards -a
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0 SP          Available  n    n    n    Passed  Normal
```

01-0	SP	Available	n	n	n	Passed	Normal
------	----	-----------	---	---	---	--------	--------

b. **Execute the addboard -c assign command to assign the PSBs.**

```
XSCF> addboard -c assign -p 0 00-0 01-0
```

c. **Execute the showboards-p command to check the PSB status.**

Execute the showboards -p command to check the status of each PSB assigned to the physical partition.

This example checks that the [Assignment] field of each PSB becomes "Assigned" since each PSB has been normally assigned to physical partition 0.

```
XSCF> showboards -p 0
```

PSB	PPAR-ID(LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	00(00)	Assigned	n	n	n	Passed	Normal
01-0	00(01)	Assigned	n	n	n	Passed	Normal

5. **Register the CPU Activation key to assign CPU core resources.**

a. **Execute the showcodactivation command to check the information on the CPU Activation key.**

Execute the showcodactivation command to check whether the physical partition contains an assignable CPU Activation key.

If only the header is displayed, the CPU Activation key is not registered in the XSCF.

```
XSCF> showcodactivation
```

Index	Description	Count
-----	-----	-----

Note - If the number of registered CPU Activations is not enough for the number of CPUs to be used, purchase CPU Activations and add the CPU Activation keys.

b. **Execute the addcodactivation command to add the CPU Activation key.**

For details on how to add a CPU Activation key, see "5.3 Adding CPU Core Resources" in the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 System Operation and Administration Guide*.

```
XSCF> addcodactivation "Product: SPARC M10-4S
SequenceNumber:10005
Cpu: noExpiration 2
Text-Signature-SHA256-RSA2048:
PSSrElBrse/r69AVSVFd38sT6AZm2bxUDdPQHKbtxgVZPsrtYguqiNUieB+mTDC
:
:
```

```
b1GCKFx1RH27FdVHiB2H0A=="
AboveKeywillbeadded,Continue?[y|n]:y
```

c. **Execute the showcodactivation command to check the information on the CPU Activation keys.**

Execute the showcodactivation command to check whether the physical partition contains an assignable CPU Activation key.

```
XSCF> showcodactivation
Index      Description Count
-----
          0  PROC           2
          1  PROC           2
          2  PROC           2
          3  PROC           2

--- Omitted ---

        62  PROC           2
        63  PROC           2
```

d. **Execute the setcod command to assign CPU core resources.**

Execute the setcod command to assign the CPU core resources to the physical partition.

The following example assigns 128 CPU core resources to physical partition 0.

```
XSCF> setcod -p 0 -s cpu -c set 128
PROC Permits assigned for PPAR 0 : 0 -> 128

PROC Permits assigned for PPAR will be changed.
Continue? [y|n] :y

Completed.
```

Note - XSCF firmware of versions XCP 2250 and earlier do not support the -c add, -c delete, and -c set options. Specify the options of the setcod command as shown below to interactively add and delete CPU core resources.

```
XSCF> setcod -s cpu
```

Execute the showcod command to check information on the CPU core resources assigned to the physical partition.

The following example confirms that 128 CPU core resources have been assigned to physical partition 0 with the setcod command that was just executed.

```
XSCF> showcod -p 0
PROC Permits assigned for PPAR 0: 128
```


6. **Execute the resetdateoffset command to reset the difference with the time managed by the XSCF.**

Execute the resetdateoffset command to reset the difference between the time managed by the XSCF and the time managed by the physical partitions.

```
XSCF> resetdateoffset -p 0
```

7. **Execute the showpparmode command to check the setting of the detail level of the diagnosis messages and that of the PPAR DR mode.**

Execute the showpparmode command to check that the detail level (Message Level) of the diagnosis message is "normal" (standard) and that Next of the PPAR DR mode is set to "on" (enabled).

```
XSCF> showpparmode -p 0
Host-ID                :9007002b
Diagnostic Level        :min
Message Level           :normal
Alive Check             :on
Watchdog Reaction       :reset
Break Signal            :on
Autoboot (Guest Domain) :on
Elastic Mode            :off
IOreconfigure           :false
PPAR DR (Current)       :-
PPAR DR (Next)          :on
```

If the detail level of the diagnosis message is other than "normal", execute the setpparmode command to set it to "normal".

For details on the setpparmode command, see the *Fujitsu SPARC M12 and Fujitsu M10/SPARC M10 XSCF Reference Manual*.

```
XSCF> setpparmode -p 0 -m message=normal
```

If PPAR DR mode is set to "off" (disabled), execute the setpparmode command to set it to "on".

```
XSCF> setpparmode -p 0 -m ppar_dr=on
```

8. **Execute the poweron command to power on the physical partitions.**

```
XSCF> poweron -p 0
```

9. **Execute the console command to connect the console to the physical partition.**

```
XSCF> console -p 0
```

10. **Install Oracle Solaris and Oracle VM Server for SPARC.**

Install, on the control domain, Oracle Solaris and Oracle VM Server for SPARC.

For details on the versions and conditions of Oracle Solaris required for physical partition dynamic reconfiguration, see [Table 1-3](#) and [Table 1-4](#).

For details on the installation, see the following documents, presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

- Oracle Solaris 11

Installing Oracle Solaris 11.2 Systems

- Oracle VM Server for SPARC

"Installing and Enabling Software" in the *Oracle VM Server for SPARC Administration Guide*

11. **Establish a redundant physical I/O configuration of the control domain.**

a. **Establish a redundant configuration for the system volume of the control domain.**

Establish a redundant configuration for the system volume of the control domain. This item describes an example of the commands for establishing a mirroring configuration with ZFS. For details, see the following documents presented on the Oracle Corporation web site (<https://docs.oracle.com/en/>).

"How to Configure a Mirrored Root Pool (SPARC or x86/VTOC)" in the *Oracle Solaris 11.2 Administration: ZFS File Systems*

To use other redundant configuration software, see the manual for that software.

Execute the `zpool status` command to check the status of the root pool.

The following example indicates that `c2t50000393E802CCE2d0s0` has been assigned to the default root pool (`rpool`).

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: none requested
config:
      NAME                                STATE      READ  WRITE CKSUM
      rpool                                ONLINE      0     0     0
      c2t50000393E802CCE2d0s0             ONLINE      0     0     0
errors: No known data errors
```

Execute the `format` command to check which disks can be added.

The following example indicates that `c3t50000393A803B13Ed0` exists as the other disk.

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  0. c2t50000393E802CCE2d0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
    sec 625>
     /pci@8000/pci@4/pci@0/pci@0/scsi@0/iport@f/disk@w50000393e802cce2,0
```

```

/dev/chassis/FUJITSU-BBEXP.500000e0e06d027f/0123_HDD00/disk
1. c3t50000393A803B13Ed0 <TOSHIBA-MBF2300RC-3706 cyl 46873 alt 2 hd 20
sec 625>
/dev/chassis/FUJITSU-BBEXP.500000e0e06d243f/022U_HDD01/disk
Specify disk (enter its number): ^C

```

Execute the `zpool attach` command to add the second disk to the `rpool` to provide a mirror configuration.

The following example adds `c3t50000393A803B13Ed0s0` with the `zpool attach` command and then checks the status of the synchronization processing (resilver) with the `zpool status` command. By referring to status and action, you can determine that the synchronization processing is in progress. Regularly execute the `zpool status` command to check the status of the synchronization processing until the processing ends.

```

# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered. The pool will
continue to function in a degraded state.
action: Wait for the resilver to complete.
       Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Wed Jan 29 21:35:39 2014
      3.93G scanned out of 70.6G at 71.9M/s, 0h15m to go 3.90G resilvered, 5.56%
done
config:

```

NAME	STATE	READ	WRITE	CKSUM
rpool	DEGRADED	0	0	0
mirror-0	DEGRADED	0	0	0
c2t50000393E802CCE2d0s0	ONLINE	0	0	0
c3t50000393A803B13Ed0s0	DEGRADED	0	0	0

```

(resilvering)

```

Upon the completion of the synchronization processing, [state] is set to "ONLINE", as shown below.

```

# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h10m with 0 errors on Wed Jan 29 21:45:42 2014
config:

```

NAME	STATE	READ	WRITE	CKSUM
rpool	ONLINE	0	0	0
mirror-0	ONLINE	0	0	0
c2t50000393E802CCE2d0s0	ONLINE	0	0	0
c3t50000393A803B13Ed0s0	ONLINE	0	0	0

```

errors: No known data errors

```

b. Establish a redundant configuration for the network of the control domain.

Execute the `dladm` command to check that the network devices are visible. The following example shows that it is possible to refer to on-board network interfaces as `net0` to `net7`.

```
# dladm show-phys -L
LINK          DEVICE          LOC
net0          igb0            /BB0/CMUL
net1          igb1            /BB0/CMUL
net2          igb6            /BB0/CMUU
net3          igb7            /BB0/CMUU
net4          igb14           /BB1/CMUL
net5          igb15           /BB1/CMUL
net6          igb12           /BB1/CMUU
net7          igb13           /BB1/CMUU
```

The following describes an example of establishing a redundant configuration for network interface `net0` on `BB0` and network interface `net4` on `BB1`, using IPMP. First, execute the `ipadm show-if` command to check that `net0` (on `BB0`) and `net4` (on `BB1`) are not displayed.

```
# ipadm show-if
IFNAME      CLASS      STATE      ACTIVE OVER
lo0         loopback  ok         yes    --
```

Execute the `ipadm create-ip` command to create IP interfaces `net0` and `net4`, and then execute the `ipadm show-if` command to check that they have been created normally.

```
# ipadm create-ip net0
# ipadm create-ip net4
# ipadm show-if
IFNAME      CLASS      STATE      ACTIVE OVER
lo0         loopback  ok         yes    --
net0        ip         down       no     --
net4        ip         down       no     --
```

Execute the `ipadm create-ipmp` command to create IPMP interface `ipmp0`, and then execute the `ipadm add-ipmp` command to add IP interfaces `net0` and `net4` to the IPMP group.

```
# ipadm create-ipmp ipmp0
# ipadm add-ipmp -i net0 -i net4 ipmp0
```

Execute the `ipadm create-addr` command to assign an IP address to IPMP interface `ipmp0`, and then execute the `ipadm show-addr` command to check the setting. In the example below, a fixed IP address is assigned.

```
# ipadm create-addr -T static -a local=xx.xx.xx.xx/24 ipmp0/v4
# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
```

ipmp0/v4	static	ok	xx.xx.xx.xx/24
lo0/v6	static	ok	:::1/128

Execute the ipadm set-ifprop command to set a standby interface, and execute the ipmpstat -i command to check the IPMP configuration.

```
# ipadm set-ifprop -p standby=on -m ip net4
# ipmpstat -i
```

INTERFACE	ACTIVE	GROUP	FLAGS	LINK	PROBE	STATE
net4	no	ipmp0	is-----	up	disabled	ok
net0	yes	ipmp0	--mbM--	up	disabled	ok

A.6.3 Example of the Active Replacement Procedure

This section describes an example of the procedure for actively replacing BB#01 using PPAR DR for the 2BB configuration system described in "[Figure A-8 Configuration Example of a 2BB Configuration Where All the Resources are Assigned.](#)" The example is for an environment where dynamic PCIe bus assignment is available (Oracle VM Server for SPARC 3.2 or later).

This description also applies to SPARC M12.

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.
 You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

1. **Log in to the master XSCF.**
 Execute the showbbstatus command to check that the XSCF to which you have logged in is the master XSCF.

 If you have logged in to a standby XSCF, log out and then log in to the master XSCF again.

```
XSCF> showbbstatus
BB#00 (Master)
```

2. **Execute the showhardconf command to check that [Status] of the XSCF in the SPARC M10-4S to be replaced is "Normal."**

```
XSCF> showhardconf
SPARC M10-4S;
+ Serial: 2081230011; Operator_Panel_Switch:Locked;
+ System_Power:On; System_Phase:Cabinet Power On;
Partition#0 PPAR_Status:Running;
BB#00 Status:Normal; Role:Master; Ver:2003h; Serial:2081231002;
+ FRU-Part-Number: CA07361-D202 A1;
```

```

+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP1236052K ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
CPU#0 Status:Normal; Ver:4142h; Serial:00322658;
+ Freq:3.000 GHz; Type:0x10;
+ Core:16; Strand:2;
:
BB#01 Status:Normal; Role:Standby; Ver:0101h;Serial:7867000297;
+ FRU-Part-Number: CA20393-B50X A2 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:0101h; Serial:PP123406CB ;
+ FRU-Part-Number:CA07361-D941 C4 /7060911 ;
+ Memory_Size:128 GB; Type: A ;
:

```

Note - If the XSCF in SPARC M10-4S to be actively replaced is defective, you cannot perform active replacement using PPAR DR.

You must stop the physical partition to which the SPARC M10-4S to be actively replaced belongs and then perform maintenance with the input power to the SPARC M10-4S to be replaced turned off.

3. Execute the `showbbstatus` command to confirm that the XSCF in the SPARC M10-4S to be replaced is not the master XSCF.

```

XSCF> showbbstatus
BB#00 (Master)

```

If the SPARC M10-4S to be replaced is the master XSCF, execute the `switchscf` command to switch the XSCF.

```

XSCF> switchscf -t Standby
The XSCF unit switch between the Active and Standby states.
Continue? [y|n] :y

```

Note - Confirm that the XSCF has been switched and rebooted before you release the SPARC M10-4S.

4. Execute the console command to connect to the console of the control domain and then log in to it.

```

XSCF> console -p 0

```

5. Release the redundant configuration of the system volume and physical I/O devices in the control domain.

Release the physical I/O devices of the SPARC M10-4S (on BB#01) to be replaced, that are used in the control domain. For details on the procedure for canceling a redundant configuration, see the document for the software for that redundant configuration.

a. Cancel the redundant configuration of the system volume in the control domain.

The following example describes how to cancel the ZFS mirroring function for the system volume in the control domain.

Execute the `zpool status` command in the control domain to check the mirroring configuration status.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                                ONLINE    0     0     0
        mirror-0                        ONLINE    0     0     0
            c2t50000393E802CCE2d0s0    ONLINE    0     0     0
            c3t50000393A803B13Ed0s0    ONLINE    0     0     0
errors: No known data errors
```

Execute the `zpool detach` command to release the disk from the mirroring configuration.

```
# zpool detach rpool c3t50000393A803B13Ed0
```

Execute the `zpool status` command to confirm that the mirroring configuration has been canceled.

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 28.7M in 0h0m with 0 errors on Tue Jan 21 10:10:01 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                                ONLINE    0     0     0
        mirror-0                        ONLINE    0     0     0
            c2t50000393E802CCE2d0s0    ONLINE    0     0     0
errors: No known data errors
```

If you are using other devices in BB#01, remove the redundant configuration or stop using those devices. For details on how to cancel a redundant configuration or stop using the devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

b. Cancel the redundant configuration of the network of the control domain.

Execute the `ipmpstat -i` command to check the configuration information for the network interfaces configuring IPMP.

```
# ipmpstat -i
```

INTERFACE	ACTIVE	GROUP	FLAGS	LINK	PROBE	STATE
net0	yes	ipmp0	-smbM--	up	disabled	ok
net4	no	ipmp0	is-----	up	disabled	ok

Execute the `if_mpadm -d net4` command to release net4 from the IPMP group, and then execute the `ipmpstat -i` command to confirm that it has been released. The following example confirms that STATE is offline.

```
# if_mpadm -d net4
# ipmpstat -i
```

INTERFACE	ACTIVE	GROUP	FLAGS	LINK	PROBE	STATE
net0	yes	ipmp0	-smbM--	up	disabled	ok
net4	no	ipmp0	-s---d-	up	disabled	offline

Execute the `ipadm delete-ip` command to delete net4.

```
# ipadm delete-ip net4
```

6. **Release the system board (PSB<BB>) of the SPARC M10-4S from the physical partition.**

a. **Execute the `deleteboard -c disconnect` command to release the PSB from the physical partition.**

If you have not manually released any of CPU core, memory, and PCIe root complex resources in advance, be sure to specify the `"-m unbind=resource"` option. When this option is specified, the resources are automatically deleted, and then the PSB is released.

```
XSCF> deleteboard -c disconnect -m unbind=resource 01-0
PSB#01-0 will be unconfigured from PPAR immediately.
Continue?[y|n] :y
Start unconfigure preparation of PSB. [1200sec]
0end
Unconfigure preparation of PSB has completed.
Start unconfiguring PSB from PPAR. [7200sec]
0..... 30..... 60....end
Unconfigured PSB from PPAR.
PSB power off sequence started. [1200sec]
0..... 30..... 60..... 90.....120.....150.....end
Operation has completed.
```

b. **Execute the `showresult` command to check the exit status of the `deleteboard` command that was just executed.**

An end value of 0 indicates the normal termination of the `deleteboard` command.

If the end value is other than 0 or if an error message is displayed upon executing the `deleteboard` command, it indicates abnormal termination of the `deleteboard` command. By referring to "[C.1.2 deleteboard](#)" based on the error message, identify the error and then take corrective action.


```
XSCF> showresult
0
```

c. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the SPARC M10-4S to be replaced is in the "Assigned" state and that the [Pwr], [Conn], and [Conf] columns all show "n."

```
XSCF> showboards -p 0
PSB   PPAR-ID(LSB)  Assignment  Pwr   Conn  Conf  Test   Fault
-----
00-0  00(00)        Assigned    y     y     y     Passed Normal
01-0  00(01)        Assigned    n     n     n     Passed Normal
```

7. **Execute the replacefru command to replace the SPARC M10-4S.**

```
XSCF> replacefru
```

Note - For details on the replacement of SPARC M10-4Ss by using the replacefru command, see "5.8 Releasing a SPARC M10-4/M10-4S FRU from the System with the replacefru Command" and "6.2 Incorporating a SPARC M10-4/M10-4S FRU into the System with the replacefru Command" in the *Fujitsu M10-4/Fujitsu M10-4S/SPARC M10-4/SPARC M10-4S Service Manual*.

8. **Incorporate the PSB into the physical partition.**

a. **Execute the showboards command to check the PSB status.**

Confirm that the PSB in the replaced SPARC M10-4S is in the Assigned state and that the [Pwr], [Conn], and [Conf] columns all show "n."

```
XSCF> showboards -p 0
PSB   PPAR-ID(LSB)  Assignment  Pwr   Conn  Conf  Test   Fault
-----
00-0  00(00)        Assigned    y     y     y     Passed Normal
01-0  00(01)        Assigned    n     n     n     Passed Normal
```

b. **Execute the addboard -c configure command to incorporate the PSB into the physical partition.**

To recover the original logical domain configuration, execute the addboard -c configure command with the -m bind=resource option specified.

```
XSCF> addboard -c configure -m bind=resource -p 0 01-0
PSB#01-0 will be configured into PPAR-ID 0. Continue?[y|n] :y
Start connecting PSB to PPAR. [3600sec]
0..... 30..... 60..... 90.....120.....150.....180.....210.....240.....
270.....300.....330.....360.....390.....420.....450.....480.....510.....
540.....570.....600.....630.....660.....690.....720.....750.....780.....
810.....840.....870.....900.....930.....960.....end
```

```

Connected PSB to PPAR.
Start configuring PSB to Logical Domains (LDoms) Manager.
[1800sec] 0.....end
Configured PSB to Logical Domains (LDoms) Manager.
Operation has completed.

```

Note - If an error message appears during execution of the addboard command, see "[C.1.1 addboard](#)," and then identify the error and take corrective action.

c. **Execute the showresult command to check the exit status of the addboard command that was just executed.**

An end value of 0 indicates the normal termination of the addboard command.

If the end value is other than 0 or if an error message is displayed upon executing the addboard command, it indicates abnormal termination of the addboard command. By referring to "[C.1.1 addboard](#)" based on the error message, identify the error and then take corrective action.

```

XSCF> showresult
0

```

d. **Execute the showboards command to check the PSB status.**

Confirm that both of the [Conn] and [Conf] columns show "y" after the PSB in the replaced SPARC M10-4S has been successfully incorporated.

```

XSCF> showboards -p 0
PSB  PPAR-ID(LSB)  Assignment  Pwr  Conn  Conf  Test  Fault
-----
00-0  00(00)         Assigned    y    y    y    Passed  Normal
01-0  00(01)         Assigned    y    y    y    Passed  Normal

```

9. **Restore the system volume and physical I/O devices on the control domain to a redundant configuration.**

a. **Place the system volume in the control domain in a redundant configuration.**

Execute the zpool status command in the control domain to check the mirroring configuration status.

The following example describes how to configure the ZFS mirroring function for the system volume in the control domain.

```

# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 29.1M in 0h0m with 0 errors on Thu Jan 23 17:27:59 2014
config:
    NAME                                STATE      READ WRITE CKSUM
    rpool                                ONLINE     0     0     0
        c2t50000393E802CCE2d0s0        ONLINE     0     0     0

```

```
errors: No known data errors
```

Execute the `zpool attach` command to incorporate the disks into a mirroring configuration.

```
# zpool attach rpool c2t50000393E802CCE2d0s0 c3t50000393A803B13Ed0s0
Make sure to wait until resilver is done before rebooting.
#
```

Execute the `zpool status` command, and then confirm that the mirroring configuration has been established.

Use the `zpool status` command to confirm whether synchronization processing (resilver) is completed.

The following shows an example of the display during synchronization processing.

```
# zpool status rpool
pool: rpool
state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
scan: resilver in progress since Mon Jan 27 15:55:47 2014
      21.1G scanned out of 70.6G at 120M/s, 0h7m to go
      21.0G resilvered, 29.84% done
config:
      NAME                                STATE      READ WRITE CKSUM
      rpool                                DEGRADED   0      0      0
        mirror-0                          DEGRADED   0      0      0
          c2t50000393E802CCE2d0s0         ONLINE    0      0      0
          c3t50000393A803B13Ed0s0         DEGRADED   0      0      0 (resilvering)
errors: No known data errors
```

Once synchronization processing is complete, the displayed screen will be as follows:

```
# zpool status rpool
pool: rpool
state: ONLINE
scan: resilvered 70.6G in 0h9m with 0 errors on Mon Jan 27 16:05:34 2014
config:
      NAME                                STATE      READ WRITE CKSUM
      rpool                                ONLINE    0      0      0
        mirror-0                          ONLINE    0      0      0
          c2t50000393E802CCE2d0s0         ONLINE    0      0      0
          c3t50000393A803B13Ed0s0         ONLINE    0      0      0
errors: No known data errors
```

If you are using other devices in BB#01, establish a redundant configuration

or resume the use of the devices. For details on how to establish a redundant configuration or resume the use of devices, see the documentation for the software for that redundant configuration and Oracle Solaris.

Supplementary Information for Using Dynamic Reconfiguration

This appendix provides supplementary information for using the physical partition dynamic reconfiguration function and how to deal with any problems.

- [Considerations for the XSCF Reboot or Failover Time](#)
- [Supplementary Information on CPU Operational Mode](#)
- [Other Considerations](#)

B.1 Considerations for the XSCF Reboot or Failover Time

If an XSCF restart or failover occurs when you perform physical partition dynamic reconfiguration by adding or deleting a system board (PSB<BB>), operations related to the physical partition dynamic reconfiguration may not complete. In this case, perform the following to check that the operation has been completed. If the operation has not yet been completed, add or remove the PSB (BB) by using dynamic reconfiguration again.

Checking the completion of addition

1. **Execute the showboards command on the XSCF to check the system board (PSB<BB>) state.**

The following example checks that the installed PSB (BB) are in the Assigned state and that the [Pwr], [Conn], and [Conf] columns all indicate "y."

XSCF> showboards -p PPAR_ID							
PSB	PPAR-ID (LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
----	-----	-----	----	----	----	-----	-----
00-0	00 (00)	Assigned	y	y	y	Passed	Normal
01-0	00 (01)	Assigned	y	y	y	Passed	Normal

2. **Execute the console command to connect to the console of the control domain and then log in to it.**

```
XSCF> console -p PPAR_ID
```

3. **Execute the ldm list-devices command to check the state of the added hardware resources.**
 When the ldm list-devices command is executed, the unassigned hardware resources are listed.
 Check that the hardware resources have been added.

```
# ldm list-devices
CORE
  ID      %FREE    CPUSET
  128     100     (256, 257)
  132     100     (264, 265)
  136     100     (272, 273)
  140     100     (280, 281)
  ....
MEMORY
  PA              SIZE      BOUND
  0x7000000000000  32G
  0x7200000000000  32G
  0x7400000000000  32G
  0x7600500000000  31488M
  ....
```

Checking the completion of removal

Execute the showboards command to check the system board (PSB<BB>) state. Check that the removed PSB (BB) are in the Assigned state and that the [Pwr], [Conn], and [Conf] columns all indicate "n." Alternatively, check that the removed system boards are in the "SP" (system board pool) and "Available" states and that the [Pwr], [Conn], and [Conf] columns all indicate "n."

The following shows examples of each.

```
XSCF> showboards -p PPAR_ID
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test  Fault
-----
00-0 00(00)      Assigned    y    y    y    Passed Normal
01-0 00(01)      Assigned    n    n    n    Passed Normal
```

```
XSCF> showboards -p PPAR_ID
PSB  PPAR-ID(LSB) Assignment  Pwr  Conn Conf Test  Fault
-----
00-0 00(00)      Assigned    y    y    y    Passed Normal
```

B.2 Supplementary Information on CPU Operational Mode

If the CPU operational mode set by the `setpparmode` command is not matched with the logical domain configuration information used to start the physical partition, XSCF switches the logical domain configuration information to the factory-default settings.

The following shows combinations of the specified physical partition configuration, CPU operational mode settings, and logical domain configuration information that can switch the information to the factory-default settings.

Note - The SPARC M12 system does not support the CPU operational mode.

Table B-1 Relationship Between the CPU Configuration, CPU Operational Mode, and Logical Domain Configuration Information That Switches the Information to the Factory-Default Settings

Physical Partition CPU Configuration	CPU Operational Mode Settings	Logical Domain Configuration Information
All SPARC64 X+ processors	compatible	Specifying logical domain configuration information created during operation using the SPARC64 X+ function with the <code>setdomainconfig</code> or <code>ldm set-spconfig</code> command
All SPARC64 X processors	Both auto and compatible	
Mixed use of SPARC64 X+ processor and SPARC64 X processor	Both auto and compatible	

The following shows operations that may switch the logical domain configuration information to the factory-default settings based on the above table.

- When the CPU operational mode for a physical partition configured with only the SPARC64 X+ processor is changed from "auto" to "compatible", and then the physical partition is started
- When a system board configured with the SPARC64 X processor is added to a physical partition configured with only the SPARC64 X+ processor using the `addboard -c assign` command, and then the physical partition is started

B.3 Other Considerations

- When a CPU memory unit (upper stage) (CMUU) is mounted on all the system boards constituting the physical partition (PPAR), if the PSB is temporarily

removed from the PPAR configuration to remove the CMUU by using the dynamic reconfiguration function (PPAR DR), some of the I/O for the CMUL will not be visible when PSB is returned. If necessary, set the configuration information for the logical domains again.

- If an abnormal end occurs when executing the deleteboard command, do not execute the addboard command. Otherwise, the addboard command will detect an error called "internal error," resulting in an abnormal end. This command may also affect any subsequent operations.
- If an abnormal end occurs while the dynamic reconfiguration function is being used and while the PSB (BB) is removed with the deleteboard command, perform the following procedure:
 1. Identify the cause of the error according to the error message.
 2. Remove the cause of the error.
 3. Use the dynamic reconfiguration function and execute the deleteboard command again to delete the PSB from the PPAR configuration.

If you fail to delete the PSB again, stop the physical partition (PPAR) and then perform the operation described either in ["6.3.1 Example of Operations for Deleting a System Board Assignment"](#) or in ["6.3.3 Example of Operations for Reserving a System Board Unassignment."](#)

- If an abnormal end occurs while using the dynamic reconfiguration function and executing the addboard command to add the PSB (BB), perform the following procedure or operations:
 1. Identify the cause of the error according to the error message.
 2. Remove the cause of the error.
 3. Use the dynamic reconfiguration function and execute the addboard command again to add the PSB to the PPAR configuration.

Or, use the dynamic reconfiguration function and execute the deleteboard command to delete the PSB from the PPAR configuration, and then execute the addboard command again to add the PSB to the PPAR configuration.

If you fail to add a PSB again with both of the above operations, stop the physical partition (PPAR) and then perform the operation described either in ["6.2.1 Example of System Board Assignment"](#) or in ["6.2.3 Example of Operations for Reserving the Incorporation of a System Board."](#)

- When you use the dynamic reconfiguration function to add the PSB (BB) with the addboard command or to delete the PSB with the deleteboard command, do not stop or restart Oracle VM Server for SPARC. Otherwise, the addboard command or the deleteboard command will abnormally end.

If you restart Oracle VM Server for SPARC and then rerun the addboard command or the deleteboard command, an abnormal end may occur again.

In this case, when the system board is added, stop the physical partition (PPAR) and then perform the operation described in ["6.2.1 Example of System Board Assignment"](#) or in ["6.2.3 Example of Operations for Reserving the Incorporation of a System Board."](#)

When the system board is removed, stop the physical partition and then perform

the operation described either in "6.3.1 Example of Operations for Deleting a System Board Assignment" or in "6.3.3 Example of Operations for Reserving a System Board Unassignment."

- When you use the dynamic reconfiguration function to add the PSB (BB) with the addboard command, or to delete the PSB with the deleteboard command, do not restart all the XSCFs using the rebootxscf -a command or XSCF Web. This operation may cause not only abnormal termination of the DR processing, but also abnormal stop of the physical partition due to an Oracle Solaris hangup or hardware failure with the operating physical partition, and parts replacement may therefore be required.
- If the assignment to the internal hard disk is deleted by using the dynamic reconfiguration function, the following message may appear and the deletion processing may fail.

```
# ldm remove-io /BB0/CMUL/SASHBA primary
The removal of /BB0/CMUL/SASHBA from the domain primary failed.
Error message from svc:/ldoms/agents in domain primary:
ERROR: devices or resources are busy.
```

The above error may occur when the fmd service uses the device to be deleted. Issue the stop command to the fmd service on the domain to which the target device is assigned, check that the service has stopped and then execute the deletion processing again. In addition, restart the fmd service after the assignment of the target device is deleted.

Use the following commands to stop/restart the fmd service.

- Stopping the fmd service

```
# svcadm disable svc:/system/fmd:default
```

- Checking the fmd service has stopped

It may take a few minutes for the fmd service to stop.

After issuing the stop command, check that the STATE is disabled and the service has stopped.

```
# svcs fmd
STATE  TIME  FMRI
disabled 00:00:00 svc:/system/fmd:default
```

- Restarting the fmd service

```
# svcadm enable svc:/system/fmd:default
```

- If Oracle Solaris suspends or resumes with the dynamic reconfiguration function in the logical domain to which the SPARC M12-2S on-board LAN or Dual 10Gigabit Ethernet card (SP1X7HF1F, SE1X7HE3G) is assigned, an "NIC-8000-1C" error may be detected. Ignore this error because it does not affect the operation.

- When deleting a PSB using deleteboard, either release the LAN or I/O devices of the PSB to be deleted from the redundant configuration or change to the unused status in advance.
- When adding a PSB using the addboard command, either incorporate the LAN or I/O device of the PSB to be added or change the settings to enable reuse.

Meanings of Messages and Corresponding Corrective Action

This appendix describes the meanings of the messages displayed during domain configuration work and the corresponding corrective action.

- [Command Messages](#)

C.1 Command Messages

This section describes the meanings of individual messages displayed during the configuration operations of physical partitions and the corresponding corrective action.

C.1.1 addboard

A hardware error occurred. Please check the error log for details.

[Meaning] The command terminated due to a hardware error.

[Action] Check the error log for the possible fault location, and replace the faulty component.

An internal error has occurred. Please contact your system administrator.

[Meaning] An internal error has occurred. Contact the system administrator.

[Action] An error or XSCF internal resource shortage may have occurred in a library in use in XSCF. Wait for a while, and then perform the operation again. If the error occurs again, contact a Fujitsu field engineer to have the error investigated.

Another DR operation is in progress. Try again later.

[Meaning] Cannot access the specified physical partition because it is undergoing dynamic reconfiguration in another session.

[Action] Dynamic reconfiguration is in progress in another session. Check the status of the physical partition, wait for a while, and then perform the operation again.

Cannot communicate with Hypervisor or Logical Domains Manager. Please

check the domain's state.

[Meaning] An error occurred in the communication with Hypervisor or Logical Domains Manager.

[Action] From the monitoring messages and console messages output, determine the cause of the failure of the dynamic reconfiguration in Oracle Solaris of the logical domain or Logical Domains Manager. Then, eliminate the cause, and perform the operation again.

Cannot communicate with XXX. Please check XXX's state.

[Meaning] The standby or slave XSCF has an error.

[Action] Resolve the error of the standby or slave XSCF. This error does not occur in the SPARC M12/M10 system with the one-unit configuration.

Configured PSB to Logical Domains (LDoms) Manager.

[Meaning] Incorporation into Logical Domains Manager has completed.

Connected PSB to PPAR.

[Meaning] Incorporation into the physical partition configuration has completed.

Control domain is stopped for CoD resource violation.

[Meaning] The specified system board could not be incorporated into the physical partition because the control domain stopped due to a violation of CPU Activation.

[Action] Resolve the CPU Activation violation, and then perform the operation again.

DR operation canceled by operator.

[Meaning] Dynamic reconfiguration was canceled because the user issued an instruction to stop it.

Initial diagnosis has completed.

[Meaning] Hardware diagnosis has completed.

Initial diagnosis started.

[Meaning] Hardware diagnosis has started.

Invalid parameter.

[Meaning] Parameter error

[Action] Check the arguments and the operand.

Operation has completed.

[Meaning] Incorporation has completed.

Operation is continued as "no".

[Meaning] The operation that applies when [n] is selected was performed because the correct operation was not executed in response to the [y|n] prompt.

[Action] Check the system board (PSB) information. If there is a PSB that needs to be incorporated, incorporate it again.

Operation not supported on this system.

[Meaning] The entered command is not supported by the system.

[Action] Check the model and the XCP version to find out whether the command can be executed.

Permission denied.

[Meaning] You do not have access privilege. Or, you attempted to execute other than an executable command on the standby XSCF, resulting in an execution privilege error.

[Action] Compare your user privileges and the execution privilege required for the command.

PPAR DR function setting is disabled.

[Meaning] Dynamic reconfiguration cannot be performed because the PPAR DR function is disabled.

[Action] Check the PPAR DR function setting, and perform the operation again.

PPAR is currently unavailable for DR, because XBU status has failed.

[Meaning] Dynamic reconfiguration cannot be performed because there is an error in either the crossbar unit (XBU) of a specified physical partition or system board (PSB), or the crossbar unit of a configured crossbar box.

[Action] Check the monitoring messages and the error log to confirm that the relevant crossbar unit does not have an error. If it has an error, identify the faulty component and replace it.

PPAR (PPAR-ID XX) is currently unavailable for DR, because another operation is in progress. Try again later.

[Meaning] Dynamic reconfiguration cannot be performed because the specified physical partition is being accessed by another operation.

[Action] The power may be off in another session, or an operation to change the configuration information of the connected PCI expansion unit may be underway in another session. Wait for a while checking the status of the physical partition, and then perform the operation again.

PPAR (PPAR-ID XX) is not currently running.

[Meaning] You specified the -c configure option, but the incorporation failed because the destination physical partition was not operating.

[Action] Check the operation status of the physical partition, or specify the -c assign option.

PSB#XX-X could not be configured into PPAR-ID XX due to operating system or Logical Domains Manager error.

[Meaning] Incorporation failed due to a dynamic reconfiguration error returned by Oracle Solaris or the logical domain or Logical Domains Manager.

[Action] From the monitoring messages and console messages output, determine the cause of the failure of the dynamic reconfiguration in Oracle Solaris or Logical Domains Manager. Eliminate the cause, and perform the operation again.

PSB#XX-X encountered a hardware error. See error log for details.

[Meaning] Hardware diagnosis of dynamic reconfiguration has detected an error. The hardware may have an error.

[Action] Check the monitoring messages and the error log to confirm that the hardware does not have an error. If it has an error, identify the faulty component and replace it.

PSB#XX-X has not been registered in PCL.

[Meaning] The operation could not be performed because the specified system board is not registered in the physical partition (PPAR) configuration information.

[Action] Register the physical partition (PPAR) configuration information, and then perform the operation.

PSB#XX-X is already assigned to another PPAR.

[Meaning] The specified PSB could not be incorporated into the specified physical partition because it is assigned to another physical partition.

[Action] The PSB is assigned to another physical partition. Check the status of the PSB.

PSB#XX-X is currently unavailable for DR. Try again later.

[Meaning] Dynamic reconfiguration cannot be performed because the specified PSB is being accessed by another operation.

[Action] It is possible that dynamic reconfiguration, start/stop of the physical partition, or maintenance work is being performed in another session. Check the status of the PSB, wait for a while, and then perform the operation again.

PSB#XX-X is not installed.

[Meaning] The operation could not be performed because the specified PSB is not mounted.

[Action] You may have specified the wrong PSB. Check the status of the system board.

PSB#XX-X status is FAILED.

[Meaning] The status of the specified PSB is failed.

[Action] Check the monitoring messages and the error log to confirm that the hardware does not have an error. If it has an error, identify the faulty component and replace it.

PSB#XX-X will be assigned to PPAR-ID XX. Continue?[y|n] :

[Meaning] PSB number XX-X will be assigned to physical partition YY. Do you want to continue? Enter "y" to continue, and enter "n" to cancel.

PSB#XX-X will be configured into PPAR-ID XX. Continue?[y|n] :

[Meaning] PSB number XX-X will be incorporated to physical partition YY. Do you want to continue? Enter "y" to continue, and enter "n" to cancel.

Start configuring PSB to Logical Domains (LDoms) Manager.

[Meaning] Incorporation into Logical Domains Manager has started.

Start connecting PSB to PPAR.

[Meaning] Incorporation into the physical partition has started.

The current configuration does not support this operation.

[Meaning] The command terminated due to a configuration error.

[Action] Check the hardware installation status and the functional settings required for the command execution.

The other XSCF is currently unavailable for DR. Try again later.

[Meaning] Dynamic reconfiguration cannot be performed because the standby or slave XSCF is being shut down.

[Action] The standby or slave XSCF may be shutting down. Perform dynamic

reconfiguration again checking the status of the standby or slave XSCF.

The system is currently unavailable for DR, because another operation is in progress. Try again later.

[Meaning] Dynamic reconfiguration cannot be performed because an operation for the entire SPARC M12/M10 system is being performed.

[Action] The firmware may be undergoing update in another session. After the firmware is completely updated, perform dynamic reconfiguration again.

Timeout detected during communicate with Logical Domains Manager.

[Meaning] A timeout occurred in the communication with Logical Domains Manager.

[Action] From the monitoring messages and console messages output, determine the cause of the failure of the dynamic reconfiguration in Oracle Solaris of the logical domain or Logical Domains Manager. Then, eliminate the cause, and perform the operation again.

Timeout detected during configuration into PPAR-ID XX.

[Meaning] Incorporation timed out before completion.

[Action] From the monitoring messages and console messages output, determine the cause of the failure of the dynamic reconfiguration in Oracle Solaris or Logical Domains Manager. Eliminate the cause, and perform the operation again.

Timeout detected during self-test of PSB#XX-X.

[Meaning] Hardware diagnosis of dynamic reconfiguration timed out before completion. The hardware may have an error.

[Action] Check the monitoring messages and the error log to confirm that the hardware does not have an error. If it has an error, identify the faulty component and replace it.

C.1.2 deleteboard

A hardware error occurred. Please check the error log for details.

[Meaning] The command terminated due to a hardware error.

[Action] Check the error log for the possible fault location, and replace the faulty component.

An internal error has occurred. Please contact your system administrator.

[Meaning] An internal error has occurred. Contact the system administrator.

[Action] An error or XSCF internal resource shortage may have occurred in a library in use in XSCF. Wait for a while, and then perform the operation again. If the error occurs again, contact a Fujitsu field engineer to have the error investigated.

Cannot communicate with Hypervisor or Logical Domains Manager. Please check the domain's state.

[Meaning] An error occurred in the communication with Hypervisor or Logical Domains Manager.

[Action] From the monitoring messages and console messages output, determine

the cause of the failure of the dynamic reconfiguration in Oracle Solaris of the logical domain or Logical Domains Manager. Then, eliminate the cause, and perform the operation again.

Cannot communicate with XXX. Please check XXX's state.

[Meaning] The standby or slave XSCF has an error.

[Action] Resolve the error of the standby or slave XSCF. This error does not occur in the SPARC M12/M10 system with the one-unit configuration.

Control domain is stopped for CoD resource violation.

[Meaning] Because the control domain stopped due to a violation of CPU Activation, the specified PSB (BB) could not be released.

[Action] Resolve the CPU Activation violation, and then perform the operation again.

DR operation canceled by operator.

[Meaning] Dynamic reconfiguration was canceled because the user issued an instruction to stop it.

Invalid parameter.

[Meaning] Parameter error

[Action] Check the arguments and the operand.

Operation has completed.

[Meaning] Release has completed.

Operation is continued as "no".

[Meaning] The operation that applies when [n] is selected was performed because the correct operation was not executed in response to the [y | n] prompt.

[Action] Check the system board (PSB) information. If there is a PSB that needs to be released, release it again.

Operation not supported on this system.

[Meaning] The entered command is not supported by the system.

[Action] Check the model and the XCP version to find out whether the command can be executed.

Permission denied.

[Meaning] You do not have access privilege. Or, you attempted to execute other than an executable command on the standby XSCF, resulting in an execution privilege error.

[Action] Compare your user privileges and the execution privilege required for the command.

PPAR DR function setting is disabled.

[Meaning] Dynamic reconfiguration cannot be performed because the PPAR DR function is disabled.

[Action] Check the PPAR DR function setting, and perform the operation again.

PPAR is currently unavailable for DR, because XBU status has failed.

[Meaning] Dynamic reconfiguration cannot be performed because there is an error in either the crossbar unit (XBU) of a specified physical partition or system board (PSB), or the crossbar unit of a configured crossbar box.

[Action] Check the monitoring messages and the error log to confirm that the relevant crossbar unit does not have an error. If it has an error, identify the faulty component and replace it.

PPAR (PPAR-ID XX) is currently unavailable for DR, because another operation is in progress. Try again later.

[Meaning] Dynamic reconfiguration cannot be performed because the specified physical partition is being accessed by another operation.

[Action] The power may be off in another session, or an operation to change the configuration information of the connected PCI expansion unit may be underway in another session. Wait for a while checking the status of the physical partition, and then perform the operation again.

PPAR (PPAR-ID XX) is not currently running.

[Meaning] You specified the -c unassign or -c disconnect option, but the release failed because the destination physical partition was not operating.

[Action] Check the operation status of the physical partition.

PSB power off sequence started.

[Meaning] Power off of the power supply of the PSB (BB) has started.

PSB#XX-X can not be unconfigured because there is Hypervisor global memory.

[Meaning] The specified PSB cannot be released because the hypervisor memory needs to be relocated.

PSB#XX-X could not be unconfigured from PPAR-ID XX due to Hypervisor error.

[Meaning] Release failed due to a dynamic reconfiguration error returned by the hypervisor.

[Action] From the monitoring messages and console messages output, determine the cause of the failure of the dynamic reconfiguration in Oracle Solaris of the logical domain or Logical Domains Manager. Then, eliminate the cause, and perform the operation again.

PSB#XX-X could not be unconfigured from PPAR-ID XX due to operating system or Logical Domains Manager error.

[Meaning] Release failed due to a dynamic reconfiguration error returned by Oracle Solaris of the logical domain or Logical Domains Manager.

[Action] From the monitoring messages and console messages output, determine the cause of the failure of the dynamic reconfiguration in Oracle Solaris of the logical domain or Logical Domains Manager. Then, eliminate the cause, and perform the operation again.

PSB#XX-X has not been registered in PCL.

[Meaning] The operation could not be performed because the specified PSB is not registered in the physical partition (PPAR) configuration information.

[Action] Register the physical partition (PPAR) configuration information, and then perform the operation.

PSB#XX-X is currently unavailable for DR. Try again later.

[Meaning] Dynamic reconfiguration cannot be performed because the specified PSB is being accessed by another operation.

[Action] It is possible that dynamic reconfiguration, start/stop of the physical partition, or maintenance work is being performed in another session. Check the status of the PSB, wait for a while, and then perform the operation again.

PSB#XX-X is not installed.

[Meaning] The operation could not be performed because the specified PSB is not mounted.

[Action] You may have specified the wrong PSB. Check the status of the PSB.

**PSB#XX-X is the last LSB for PPAR-ID XX, and this PPAR is still running.
Operation failed.**

[Meaning] The specified PSB (BB) cannot be removed from the configuration because it is the last PSB (BB) that constitutes the physical partition.

[Action] Perform the operation after stopping the physical partition, or stop the physical partition by specifying the -c reserve option.

PSB#XX-X status is FAILED.

[Meaning] The specified PSB (BB) could not be released because its status is failed.

[Action] Check the monitoring messages and the error log to confirm that the hardware does not have an error. If it has an error, identify the faulty component and replace it.

PSB#XX-X will be unassigned from PPAR after the PPAR restarts. Continue?

[y|n] :

[Meaning] Release of PSB number XX-X from the physical partition following the restart of the physical partition will be reserved. Do you want to continue?

PSB#XX-X will be unassigned from PPAR immediately. Continue?[y|n] :

[Meaning] PSB number XX-X will be released from the physical partition. Do you want to continue? Enter "y" to continue, and enter "n" to cancel.

PSB#XX-X will be unconfigured from PPAR immediately. Continue?[y|n] :

[Meaning] PSB number XX-X will be released from the physical partition, and will be in the assigned state. Do you want to continue? Enter "y" to continue, and enter "n" to cancel.

Start unconfigure preparation of PSB.

[Meaning] Preparation of PSB (BB) release has started.

Start unconfiguring PSB from PPAR.

[Meaning] Release process of PSB (BB) has started.

System may be temporarily suspended, proceed?[y|n] :

[Meaning] The system will be temporarily suspended. Do you want to continue? Enter "y" to continue, and enter "n" to cancel.

The current configuration does not support this operation.

[Meaning] The command terminated due to a configuration error.

[Action] Check the hardware installation status and the functional settings required for the command execution.

The other XSCF is currently unavailable for DR. Try again later.

[Meaning] Dynamic reconfiguration cannot be performed because the standby

or slave XSCF is being shut down.

[Action] The standby or slave XSCF may be shutting down. Perform dynamic reconfiguration again checking the status of the standby or slave XSCF.

The system is currently unavailable for DR, because another operation is in progress. Try again later.

[Meaning] Dynamic reconfiguration cannot be performed because an operation for the entire SPARC M12/M10 system is being performed.

[Action] The firmware may be undergoing update in another session. After the firmware is completely updated, perform dynamic reconfiguration again.

There is no available action for the following resources.

Or,

DR operation would require shutting down domains, use force to proceed

[Meaning] A physical I/O device of the system board (PSB) to be deleted may be assigned to a logical domain.

[Action] Execute the `ldm list-io` command from the control domain to check the assignment status of the relevant physical I/O device. If the relevant physical I/O device is assigned to a logical domain, use the `ldm remove-io` command to delete it.

Timeout detected during communicate with Logical Domains Manager.

[Meaning] A timeout occurred in the communication with Logical Domains Manager.

[Action] From the monitoring messages and console messages output, determine the cause of the failure of the dynamic reconfiguration in Oracle Solaris of the logical domain or Logical Domains Manager. Then, eliminate the cause, and perform the operation again.

Timeout detected during poweroff.

[Meaning] The system board (PSB) power-off processing of dynamic reconfiguration timed out before completion.

[Action] Confirm that the hardware has not failed. If it has an error, identify the faulty component and replace it.

Timeout detected during unconfiguration of PSB#XX-X.

[Meaning] Release of dynamic reconfiguration timed out before completion.

[Action] Check the monitoring messages and the error log to confirm that the hardware does not have an error. If it has an error, identify the faulty component and replace it.

Unconfigure preparation of PSB has completed.

[Meaning] Preparation of PSB (BB) release has completed.

Unconfigured PSB from PPAR.

[Meaning] PSB (BB) release has completed.

Index

A

addboard, 471

C

checking CPU Activation setting information, 92

command messages, 471

control domain console, 109

control domain, initial settings for, 159

control domain, logging in to, 158

CPU, 24

CPU Activation key information, checking, 90

CPU Activation registration, 92

CPU core resources, assigning, 99

CPU core resource usage status, checking, 90

CPU operational mode, 28, 76

CPU socket, resource usage status checking for each, 116

CPU socket restrictions, setting, 130

D

default services, setting, 121, 159

delayed reconfiguration mode, starting, 120

deleteboard, 475

device, reconfiguring, 143

device setting information, checking, 87

domain configuration, 1, 79

domain configuration, overview of, 1

dynamic reconfiguration, 465

dynamic reconfiguration conditions, 56

dynamic reconfiguration settings, 56

G

guest domain, migrating, 203, 206

guest domain migration example, 206

guest domain overview, 203

guest domain, starting, 142

guest domain, configuring, 161

H

hardware resources, 10

I

I/O device, configuring, 132

I/O device usage status, checking, 118

L

logical domain, 18

logical domain components, 18

logical domain conditions, 42

logical domain configuration, 43, 231, 307

logical domain configuration flow, 155

logical domain configuration information, 108

logical domain configuration information, displaying, 94, 118

logical domain configuration information, setting , 131

logical domain configuration information, saving , 163

logical domain configuration operation example,

157
logical domain configuration, working with, 20
logical domain configurations, 110
logical domain, creating, 132
logical domain reconfiguration, 55
logical domain settings, 42
logical domain status, checking, 88, 117
Logical Domains Manager is running,
confirming, 112

M

memory mirror mode, 40
memory mirroring, setting, 95

O

Oracle Solaris conditions, 41
Oracle Solaris settings, 41

P

physical I/O device, 22
physical partition, 13
physical partition components, 13
physical partition configuration flow, 145
physical partition configuration information,
checking, 80
physical partition configuration information,
setting, 97
physical partition configuration, working with,
15
physical partition configurations, 79
physical partition dynamic reconfiguration, 23,
210
physical partition dynamic reconfiguration,
environment configuration examples and
procedures using, 209
physical partition number, considerations when
changing, 201
physical partition operation mode, checking, 93
physical partition operation mode, setting, 106
physical partition reconfiguration, 165
physical partition reconfiguration, flow for, 165
physical partition, starting, 109
physical partition status, checking, 83

physical partition, stopping, 109
physical partition, configuring and operating, 146
physical partition, dynamic reconfiguration,
overview of, 23
physical partition, dynamic reconfiguration,
using, 23
physical partition, status, 31
PPAR DR mode, 68

R

real-time process, 42
recovery mode, 162
recovery mode, enabling, 143
recovery mode, configuring, 162
resource assignment status, checking, 113
resource group, usage status checking for each,
115
resource usage status, checking, 113
resources, binding, 142

S

services, checking, 112
shutdown group, specifying, 143
software conditions, 25
SPARC64 X processor, 27, 75
SPARC64 X+ processor, 27, 71, 75
SR-IOV virtual function, creating or destroying,
134
startup device, configuring, 142
status management, XSCF, 31
status, change, 34
swap space, 41
system board, example of operations for deleting
, 182
system board, example of operations for moving
, 189
system board, example of operations for
replacing, 196
system board, deleting, 104
system operation, conditions for, 31
system operation, settings for, 31

V

- virtual console, configuring, 141
- virtual CPUs, configuring, 123
- virtual CPUs that can be assigned, checking the number of, 113
- virtual disk, configuring, 140
- virtual disk server, configuring, 139
- virtual memory, configuring, 127
- virtual network device, configuring, 138

X

- XSCF conditions, 38
- XSCF settings, 38

