

SPARC M12 XSCF 設定手順書



2017年8月

第1.0版

富士通株式会社

■使用条件

- 著作権・商標権・その他の知的財産権について

コンテンツ(文書・画像・音声等)は、著作権・商標権・その他の知的財産権で保護されています。

本コンテンツは、個人的に使用する範囲でプリントアウトまたはダウンロードできます。ただし、これ以外の利用(ご自分のページへの再利用や他のサーバへのアップロード等)については、当社または権利者の許諾が必要となります。

- 保証の制限

本コンテンツについて、当社は、その正確性、商品性、ご利用目的への適合性等に関して保証するものではなく、そのご利用により生じた損害について、当社は法律上のいかなる責任も負いかねます。本コンテンツは、予告なく変更・廃止されることがあります。

- 輸出または提供

本製品を輸出または提供する場合は、外国為替及び外国貿易法及び米国輸出管理関連法規等の規制をご確認の上、必要な手続きをおとりください。

■商標について

- UNIX は、米国およびその他の国におけるオープン・グループの登録商標です。
- SPARC Enterprise、SPARC64、SPARC64 ロゴおよびすべての SPARC 商標は、米国 SPARC International, Inc.のライセンスを受けて使用している、同社の米国およびその他の国における商標または登録商標です。
- Oracle と Java は、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。
- その他各種製品名は、各社の製品名称、商標または登録商標です。

はじめに

本書の内容

- 本書は、SPARC M12 を設置したあと、システムの初期診断や XSCF 環境の初期セットアップを行う手順について記載しています。

留意事項

- 本書の内容は、SPARC M12 の設置（ラック搭載など）が完了し、電源ケーブルを接続して XSCF が起動していることを前提としています。
- 本書の内容は、SPARC M12 のすべてのモデルを対象としています。コマンド実行例は、SPARC M12-2S における実行例を示しています。ご使用のモデルによっては実行例が異なる場合があります。
- 本書の内容は、Oracle Solaris 11.3 を基に記載しています。
- 本書に記載されているコマンドや各設定項目などの詳細は、以下のマニュアルをご確認ください。
<http://www.fujitsu.com/jp/products/computing/servers/unix/sparc/lineup/m12-2s/documents/index.html>
 - 『システム運用・管理ガイド』
 - 『XSCF リファレンスマニュアル』
- 【任意設定】と記載されている項目は、必要に応じて設定してください。

本書での表記

- 本書では、Oracle Solaris を「Solaris」と記載することがあります。

目次

1. システムの初期診断	1
1.1. XSCF への接続と初期設定	1
1.2. 初期診断	5
2. XSCF 環境のセットアップ	11
2.1. パスワードポリシーの設定	11
2.2. ユーザーアカウントの作成	13
2.3. telnet の設定	14
2.4. ネットワークの設定	15
2.5. NTP サーバの設定【任意設定】	20
2.6. SNMP エージェントの設定【任意設定】	21
2.7. XSCF Web の設定【任意設定】	22
2.8. 消費電力制限(パワーキャッピング)の設定【任意設定】	25
2.9. メモリミラーの設定【任意設定】	27
2.10. CPU リソースの初期設定	28
2.11. 物理パーティションの起動／停止とコンソール接続【任意設定】	31
3. CPU コア アクティベーション	34
3.1. OS 上での CPU リソースの確認	34
3.2. CPU コア アクティベーション機能による CPU コアの追加【任意設定】	37
3.3. OS 環境での動的 CPU リソース追加【任意設定】	42
3.4. CPU コアの一時利用機能【任意設定】	44
3.4.1. CPU コア一時利用機能の有効化	44
3.4.2. 一時利用する CPU コアのドメインへの割り当て	45
3.4.3. 新規 CPU コア アクティベーションの追加	48
3.4.4. CPU コア一時利用機能の無効化	49
改版履歴	51

1. システムの初期診断

本章では、XSCF の初期設定からシステムの初期診断までの手順を説明します。

本章は、サーバの設置後に電源ケーブルを接続した状態を前提としています。ターミナルソフトウェアで、サーバにシリアル接続した状態からスタートします。

1.1. XSCF への接続と初期設定

1) サーバにシリアル接続します。

- i) サーバのシリアルポートに、システム管理用端末が接続されていることを確認します。
- ii) ターミナルソフトウェアが、次の設定値になっていることを確認し、サーバにシリアル接続します。




ターミナルソフトウェアの設定値

設定項目	値
ボーレート	9600
データ長	8 ビット
パリティ	なし
STOP ビット	1 ビット
フロー制御	なし
ディレイ	0 以外

2) XSCF のログイン ID を入力します。

- i) シリアル接続後、“SCF_READY”と表示されたら[Enter]キーを押します。
- ii) ログイン画面が表示されるので、デフォルトユーザー (default) でログインします。

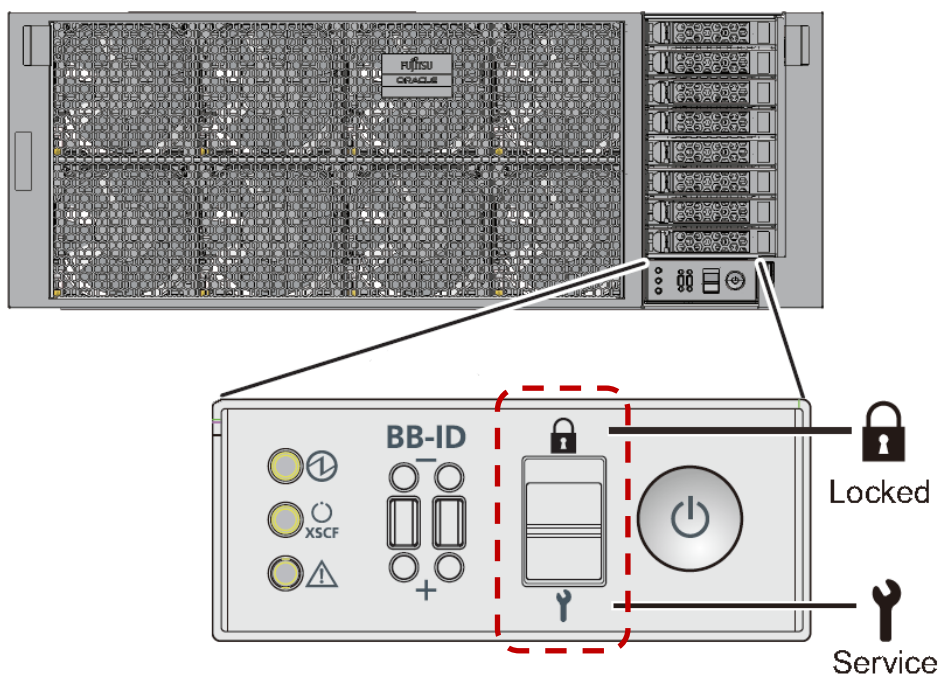
<[Enter]キーを押す>
localhost login: **default**

-  ユーザーアカウントの作成を行うまでは、デフォルトのユーザーを使用して XSCF にログインします。
デフォルトのユーザー権限は、useradm および platadm です。
-  platadm 権限を持つユーザーは、useradm 権限または auditadm 権限が必要な操作を除く、XSCF に関するすべての設定と操作ができます。
-  useradm 権限を持つユーザーは、ユーザー権限に関する操作ができます。

3) オペレーションパネルのモードスイッチを操作します。

オペレーションパネルのモードスイッチとは、サーバの保守に使用するスイッチで、筐体の前面にあります。Locked が通常運用時に使用するモードで、Service は保守作業時のモードです。

オペレーションパネルのモードスイッチ



オペレーションパネルのモードスイッチを使用して XSCF にログインします。以下に記載しているモードスイッチの切り替え操作は 1 分以内で行います。1 分を経過するとログイン認証がタイムアウトします。

- i) 以下のメッセージが表示されたら、オペレーションパネルのモードスイッチを Locked に切り替え、Enter キーを押します。

※ 元々 Locked になっていた場合は、Service に切り替えるよう指示されます。Service に切り替えてください。

Change the panel mode switch to Locked and press return...

- ii) 5 秒以上待ちます。

Leave it in that position for at least 5 seconds.

- iii) オペレーションパネルのモードスイッチを Service に戻し、Enter キーを押します。
手順 i) で Service に切り替えた場合は、Locked に戻し、Enter キーを押します。

Change the panel mode switch to Service and press return...

- iv) XSCF プロンプトが表示されることを確認します。

XSCF>

4) ファームウェア(XCP)の版数を確認します。

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Current): 3017
XCP1 (Reserve): 3017
```

☞ 実行例では旧版のファームウェアが表示されていますが、実際には出荷時の最新のファームウェアが適用されています。

☞ XCP 版数は「xyz」のように 4 桁で表示されます。

x : メジャーリリース番号

yy : マイナーリリース番号

z : マイクロリリース番号

☞ 最新の XCP は SupportDesk サービスを契約していただくと、以下のサイトからダウンロードできます。

<http://eservice.fujitsu.com/supportdesk/>

5) 高度設定を確認します。

```
XSCF> showaltitude
0m
```

《参考》

● 高度の設定

SPARC M12 では、設置された地点の高度と温度によって、システム内部を冷却するためのファンの回転数を制御しています。そのため、標高が高い場所にサーバを設定する場合、高度を設定します。

高度設定には、setaltitude コマンドを使用します。

単位は「m」(メートル)です。値は 100 単位で指定可能です。

ここでは、高度を 1000 m に設定します。

```
XSCF> setaltitude -s altitude=1000
1000m
```

☞ 設定後、rebootxscf コマンドで設定を反映する必要があります。手順 7) で時刻を設定する場合は、XSCF のリセットが行われるため、ここでは rebootxscf コマンドの実行をスキップしてもかまいません。

6) タイムゾーンを確認します。

デフォルトでは、タイムゾーンは協定世界時(UTC)に設定されています。

```
XSCF> showtimezone -c tz
UTC
```

《参考》

● タイムゾーンの設定

タイムゾーンの設定には、settimezone コマンドを使用します。

ここでは、タイムゾーンを「Asia/Tokyo」に設定します。

```
XSCF> settimezone -c settz -s Asia/Tokyo
Asia/Tokyo
```

7) 現在時刻を確認します。

```
XSCF> showdate
Wed Feb 22 11:41:15 JST 2017
```

8) 現在時刻を設定します。

時刻設定には、setdate コマンドを使用します。


既にシステムが起動している場合は、時刻を設定する前にパーティションの電源を切断(poweroff)する必要があります。

ここでは、2017 年 2 月 22 日 11 時 45 分に設定します。

時刻は、「mmddHHMMyyyy.SS」または「yyyy.MM.DD-HH:MM:SS」のどちらかの形式で指定します。

```
XSCF> setdate -s 022211452017.00
Wed Feb 22 11:45:00 JST 2017
The XSCF will be reset. Continue? [y|n] :y
Wed Feb 22 02:45:00 UTC 2017
```

← 「y」を入力

 時刻を設定すると XSCF のリセットが行われます。XSCF のセッションは切断されるため、再度 XSCF へログインする必要があります。

1.2. 初期診断

1) ハードウェアの状態を確認します。

以下の 3 点を確認します。

- 初期診断テストがまだ実行されていないこと : Test(初期診断の状態)が「Unkown」
- システムボード(SB)が正常な状態であること : Fault(縮退状態)が「Normal」
- サーバに電源が投入されていないこと : Pwr(電源の投入状態)が「n」

```
XSCF> showboards -a -v
```

PSB	R	PPAR-ID (LSB)	Assignment	Pwr	Conn	Conf	Test	Fault
00-0	*	00 (00)	Assigned	n	n	n	Unknown	Normal

2) サーバの初期診断を実施します。

サーバを起動(poweron)する前に、故障部品がないかどうかを確認します。

testsb コマンドのオプションは以下のとおりです。

オプション	説明
-v	情報を詳細表示
-p	診断処理の途中で OpenBoot PROM(OBP) の probe-scsi-all を実行し、結果を表示
-s	診断処理の途中で OpenBoot PROM(OBP) の show-devs を実行し、結果を表示
-a	システムに搭載されているすべての PSB を診断
-y	プロンプトに対して自動的に「y」を入力

```
XSCF> testsb -v -p -s -a -y
```

```
Initial diagnosis is about to start, Continue?[y|n] :y
```

← 「y」を入力

```
PSB power on sequence started.
```

(※ 2,3 分程度待ちます)

```
POST Sequence 01 Banner
```

```
LSB#00: POST 5.8.0 (2016/11/28 11:39)
```

```
POST Sequence 02 CPU Check
```

```
POST Sequence 03 CPU Register
```

```
:
```

```
-<省略>-
```

```
:
```

```
POST Sequence 1E System Status Check
```

```
POST Sequence 1F Prepare To Start Hypervisor
```

```
POST Sequence Complete.
```

(※ 2,3 分程度待ちます)

```
Hypervisor version: @(#)Hypervisor 1.5.13 2017/01/31 13:00 1.5.12+4
```

```
Configuring System Board....Completed.
```

```
:
```

```
-<省略>-
```

```
:
```

```
/packages/kbd-translator
```

```
/packages/SUNW,asr
```

```
/packages/dropins
```

```
/packages/terminal-emulator
```

```
/packages/disk-label
```

```
/packages/deblocker
```

```
/packages/SUNW,probe-error-handler
```

```
/packages/SUNW,builtin-drivers
```

(※ 1, 2 分程度待ちます)

```
:
```

```
-<省略>-
```

```
:
```

```
PSB Test Fault
```

```
-----
```

```
00-0 Passed Normal
```

☞ 実行に時間がかかる箇所があります。

☞ TEST(初期診断の状態)に「Passed」と表示されることを確認します。

3) 初期診断の結果を確認します。

以下の 2 点を確認します。

- 初期診断が実行されたこと : Test(初期診断の状態)が「Passed」
- 実行結果が正常な状態であること : Fault(縮退状態)が「Normal」

```
XSCF> showboards -a -v
```

```
PSB R PPAR-ID (LSB) Assignment Pwr Conn Conf Test Fault
```

```
-----
```

```
00-0 * 00(00) Assigned n n n Passed Normal
```

4) 故障部品を確認します。

初期診断時に異常が見つかった場合、showstatus コマンドで詳細を確認します。

```
XSCF> showstatus
```

```
XSCF>
```

☞ 故障(縮退)部品がなければ、何も表示されずプロンプトが戻ります。

5) CPU やメモリ、IO デバイスの状態を確認します。

各コンポーネントのステータスが「Normal」であることを確認します。

```
XSCF> showhardconf -M
SPARC M12-2S;
+ Serial:PZ51626007; 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:3017h; Serial:PZ51626007;
+ FRU-Part-Number:CA20369-B17X 003AB/9999999 ;
+ Power_Supply_System: ;
+ Memory_Size:256 GB;
CMUL Status:Normal; Ver:1101h; Serial:PP162201TV ;
+ FRU-Part-Number:CA07855-D201 A2 /9999999 ;
+ Memory_Size:128 GB; Type: C ;
CPU#0 Status:Normal; Ver:4242h; Serial:00070539;
+ Freq:4.250 GHz; Type:0x30;
+ Core:12; Strand:8;
MEM#00A Status:Normal;
+ Code:ce8002M393A2K40BB1-CRC 00-316D59BA;
+ Type:83; Size:16 GB;
:
-<省略>-
:
CMUU Status:Normal; Ver:1101h; Serial:PP1621003G ;
+ FRU-Part-Number:CA07855-D401 A2 /9999999 ;
+ Memory_Size:128 GB; Type: C ;
CPU#0 Status:Normal; Ver:4242h; Serial:00070455;
+ Freq:4.250 GHz; Type:0x30;
+ Core:12; Strand:8;
MEM#00A Status:Normal;
+ Code:2c800f18ASF2G72PZ-2G3B1 31-1221DE24;
+ Type:83; Size:16 GB;
:
-<省略>-
:
XBU#0 Status:Normal; Ver:1101h; Serial:PP161702XJ ;
+ FRU-Part-Number:CA20369-B18X 001AA/9999999 ;
+ Type: C ;
XBU#1 Status:Normal; Ver:1101h; Serial:PP161702XF ;
+ FRU-Part-Number:CA20369-B18X 001AA/9999999 ;
+ Type: C ;
XSCFU Status:Normal; Ver:0101h; Serial:PP161702WK ;
+ FRU-Part-Number:CA20369-B08X 002AB/9999999 ;
+ Type: A ;
OPNL Status:Normal; Ver:0101h; Serial:PP161800UX ;
```

```

+ FRU-Part-Number:CA20365-B35X 005AC/7060922      ;
+ Type: A ;
PSUBP Status:Normal; Ver:1101h; Serial:PP161800CA  ;
+ FRU-Part-Number:CA20369-B17X 003AB/9999999      ;
+ Type: C ;
PSU#0 Status:Normal; Ver:303141h; Serial:HWCD1614000297;
+ FRU-Part-Number:CA01022-0850/T. B. D            ;
+ Power_Status:ON; AC:200 V; Type: C ;
PSU#1 Status:Normal; Ver:303141h; Serial:HWCD1614000266;
+ FRU-Part-Number:CA01022-0850/T. B. D            ;
+ Power_Status:ON; AC:200 V; Type: C ;
PSU#2 Status:Normal; Ver:303141h; Serial:HWCD1614000189;
+ FRU-Part-Number:CA01022-0850/T. B. D            ;
+ Power_Status:ON; AC:200 V; Type: C ;
PSU#3 Status:Normal; Ver:303141h; Serial:HWCD1614000305;
+ FRU-Part-Number:CA01022-0850/T. B. D            ;
+ Power_Status:ON; AC:200 V; Type: C ;
FANU#0 Status:Normal; Type: C ;
FANU#1 Status:Normal; Type: C ;
FANU#2 Status:Normal; Type: C ;
FANU#3 Status:Normal; Type: C ;
FANU#4 Status:Normal; Type: C ;
FANU#5 Status:Normal; Type: C ;
FANU#6 Status:Normal; Type: C ;
FANU#7 Status:Normal; Type: C ;
HDDBP Status:Normal; Type: A ;

```

XSCF>

《参考》

● 異常が発生しているコンポーネントがある場合

コンポーネントの前にアスタリスク(*)が表示されます。

```

* PSU#1 Status:Faulted; Ver:303141h; Serial:GWSD1416000597;
+ FRU-Part-Number:CA01022-0751-M/7088116      ;
+ Power_Status:OFF; AC:100 V;

```

6) CPU やメモリ、IO デバイスなど、サーバに搭載されているコンポーネントの数を確認します。

XSCF> showhardconf -u

SPARC M12-2S; Memory_Size:256 GB;

FRU	Quantity
BB	1
CMUL	1
Type:C	(1)
CPU	1
Freq:4.250 GHz;	(1)
MEM	8
Type:83; Size:16 GB;	(8)
CMUU	1
Type:C	(1)
CPU	1
Freq:4.250 GHz;	(1)
MEM	8
Type:83; Size:16 GB;	(8)
PCICARD	0
LINKCARD	0
PCIBOX	0
IOB	0
LINKBOARD	0
PCI	0
FANBP	0
PSU	0
FAN	0
XBU	2
Type:C	(2)
XSCFU	1
Type:A	(1)
OPNL	1
Type:A	(1)
PSUBP	1
Type:C	(1)
PSU	4
Type:C	(4)
FANU	8
HDDBP	1
XBBOX	0
XBU	0
XSCFU	0
OPNL	0
XBBPU	0

	XSCF IFU		0	
	PSU		0	
	FANU		0	
+-----+				

7) エラーログを確認します。

エラーログに Warning や Alarm などのエラーがないかを確認します。

```
XSCF> showlogs error
```

 エラーがない場合は何も表示されません。

2. XSCF 環境のセットアップ

本章では、ユーザーの設定やネットワーク設定など、XSCF のセットアップ手順を説明します。

2.1. パスワードポリシーの設定

1) XSCF 用ユーザーアカウントのパスワードポリシーを確認します。

パスワードの有効期限や、パスワードの入カミス時のリトライ回数などを確認します。

```
XSCF> showpasswordpolicy
Mindays: 0
Maxdays: 99999
Warn: 7
Inactive: -1
Expiry: 0
Retry: 3
Difok: 3
Minlen: 9
Dcredit: 1
Ucredit: 1
Lcredit: 1
Ocredit: 1
Remember: 3
```

2) パスワードポリシーを設定します。

setpasswordpolicy コマンドのオプションは以下のとおりです。

オプション	説明
-y	リトライ回数を指定 (Retry)
-m	パスワードの最小文字数長を指定 (Minlen)
-M	パスワードの有効期限を指定 (Maxdays)
-W	パスワード期限切れ警告を開始する日を指定 (Warn)

その他のオプションについては、『XSCF リファレンスマニュアル』をご参照ください。

ここでは、リトライ回数 3 回、6 文字以上のパスワード、有効期限 60 日、期限切れ警告開始日を 15 日前に設定します。

```
XSCF> setpasswordpolicy -y 3 -m 6 -M 60 -w 15
```

3) 設定したポリシーが反映されていることを確認します。

```
XSCF> showpasswordpolicy
```

```
Mindays: 0
```

```
Maxdays: 60
```

```
Warn: 15
```

```
Inactive: -1
```

```
Expiry: 0
```

```
Retry: 3
```

```
Difok: 3
```

```
Minlen: 6
```

```
Dcredit: 1
```

```
Ucredit: 1
```

```
Lcredit: 1
```

```
Ocredit: 1
```

```
Remember: 3
```


2.2. ユーザーアカウントの作成

1) XSCF にログインするためのユーザーアカウントを作成します。

ここでは、「edu01」というユーザーを作成します。

```
XSCF> adduser edu01
```

2) ユーザー権限を追加します。

作成したユーザーに platadm 権限および useradm 権限を追加します。

```
XSCF> setprivileges edu01 platadm useradm
```

3) パスワードを設定します。

「2.1 パスワードポリシーの設定」で設定したパスワードポリシーに則り、パスワードを入力します。

```
XSCF> password edu01
Password:*****
Retype new password:*****
passwd: password updated successfully
```

 パスワードは画面には表示されません。

4) ユーザーアカウントを確認します。

作成したユーザー「edu01」が存在することを確認します。

```
XSCF> showuser -l
User Name:      edu01
UID:            102
Status:         Enabled
Minimum:        0
Maximum:        60
Warning:        15
Inactive:       -1
Last Change:    Mar 15, 2017
Password Expires: May 14, 2017
Password Inactive: Never
Account Expires: Never
Privileges:     useradm
                platadm
```

2.3. telnet の設定

1) telnet サービスの状態を確認します。

デフォルトでは、telnet サービスは無効(disabled)に設定されています。

```
XSCF> showtelnet  
Telnet status: disabled
```

2) telnet サービスを有効化します。

```
XSCF> settelnet -c enable  
Continue? [y|n] : y ← 「y」を入力
```

3) telnet サービスが有効(enabled)になっていることを確認します。

```
XSCF> showtelnet  
Telnet status: enabled
```

2.4. ネットワークの設定

1) XSCF ホスト名とドメイン名を確認します。

```
XSCF> showhostname -a  
bb#00:localhost.localdomain
```

🔊 SPARC M12-2S のビルディングブロック構成の場合は、接続されている複数の筐体の情報が表示されます。

2) XSCF にホスト名と DNS ドメイン名を設定します。

【書式】 sethostname 筐体名 ホスト名
sethostname -d DNS ドメイン名

OS に設定するホスト名とは異なる名前を設定してください。

筐体名はあらかじめ決められています。手順 1) で表示された筐体名を指定します。

```
XSCF> sethostname bb#00 xscf0-hostname  
XSCF> sethostname -d example.com
```

🔊 ホスト名および DNS ドメイン名の設定は XSCF の再起動後に反映されます。

3) XSCF のネットワークインターフェース名を確認します。

```
XSCF> shownetwork -a  
bb#00-lan#0  
Link encap:Ethernet HWaddr B0:99:28:9B:B8:60  
BROADCAST MULTICAST MTU:1500 Metric:1  
RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)  
  
bb#00-lan#1  
Link encap:Ethernet HWaddr B0:99:28:9B:B8:61  
BROADCAST MULTICAST MTU:1500 Metric:1  
RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)  
Base address:0x8000
```

🔊 SPARC M12-1, M12-2, M12-2S(1BB 構成)の場合、ネットワークインターフェース名は 2 つ表示されます。

4) IP アドレスとサブネットマスクを設定します。

【書式】 setnetwork -m サブネットマスク ネットワークインターフェース名 IP アドレス

ネットワークインターフェース名 (XSCF-LAN) は、手順 3) で確認したとおりです。

ここでは、XSCF-LAN#0 に IP アドレス「192.168.10.20」、サブネットマスク「255.255.255.0」を設定します。

```
XSCF> setnetwork -m 255.255.255.0 bb#00-lan#0 192.168.10.20
```

🔊 IP アドレスの設定は XSCF の再起動後に反映されます。

5) DNS サーバの情報を確認します。

デフォルトでは、DNS サーバは登録されていません。

```
XSCF> shownameserver
```

```
nameserver ---
search     ---
```

6) DNS サーバを設定します。

DNS サーバの IP アドレスを指定します。

```
XSCF> setnameserver 192.168.10.100
```

🔊 DNS サーバの IP アドレス設定は XSCF の再起動後に反映されます。

7) DNS サーバのサーチパスを指定します。

【書式】 setnameserver -c addsearch ドメイン名

```
XSCF> setnameserver -c addsearch nsserver.com
```

🔊 DNS サーバのドメイン名設定は XSCF の再起動後に反映されます。

8) ルーティング環境を確認します。

デフォルトでは、ルーティング情報は設定されていません。

```
XSCF> showroute -a
```

Destination	Gateway	Netmask	Flags	Interface
-------------	---------	---------	-------	-----------

9) デフォルトゲートウェイを設定します。

ここでは、bb#00 の XSCF-LAN#0 にデフォルトゲートウェイ「192.168.10.1」を指定します。

デフォルトのルーティング情報の宛先となる IP アドレスを設定する場合、「0.0.0.0」を入力します。

```
XSCF> setroute -c add -n 0.0.0.0 -g 192.168.10.1 bb#00-lan#0
```

🔊 ルーティングの設定は XSCF の再起動後に反映されます。

10) ネットワークの設定内容を XSCF に適用します。

ホスト名と DNS ドメイン名が設定されていない場合、エラーとなります。

```
XSCF> applynetwork
```

```
The following network settings will be applied:
```

```
bb#00 hostname :xscf0-hostname
```

```
bb#01 hostname :
```

```
DNS domain name :example.com
```

```
nameserver :192.168.10.100
```

```
search :nserver.com
```

```
interface :bb#00-lan#0
```

```
status :up
```

```
IP address :192.168.10.20
```

```
netmask :255.255.255.0
```

```
route :-n 0.0.0.0 -m 0.0.0.0 -g 192.168.10.1
```

```
interface :bb#00-lan#1
```

```
status :down
```

```
IP address :
```

```
netmask :
```

```
route :
```

```
:
```

```
-<省略>-
```

```
:
```

```
Continue? [y|n] :y ← 「y」を入力
```

```
Please reset the all XSCFs by rebootxscf to apply the network settings.
```

```
Please confirm that the settings have been applied by executing
```

```
showhostname, shownetwork, showroute, showsscp and shownameserver after  
rebooting the all XSCFs.
```

11) XSCF をリセットします。

XSCF を再起動して、設定を XSCF に反映させます。

```
XSCF> rebootxscf -a
The XSCF will be reset. Continue? [y|n] :y
XSCF> multi_set_system_scf_ready.sh -- complete
checkbrand.sh -- complete
snmpwatch.sh -- complete
snmpd.sh -- complete
settmpnetwork.sh -- complete
cli_scf_ready_after_setting.sh -- complete
setting_remcs_conf.sh -- complete
service syslog-ng -- stop
Stopping syslog-ng: [ OK ]
init_script2 -- complete
cli_ntp_setting.sh -- complete
cli_network_setting.sh -- complete
service iptables -- stop
iptables: Flushing firewall rules: [ OK ]
iptables: Setting chains to policy ACCEPT: filter [ OK ]
iptables: Unloading modules: [ OK ]
service network -- stop
Shutting down loopback interface: [ OK ]
:
:
start /sp/bin/coremgrd (pid=2315)
cli_scf_ready_after_setting.sh -- complete
settmpnetwork.sh -- complete
snmpd.sh -- complete
snmpwatch.sh -- complete
checkbrand.sh -- complete
multi_set_system_scf_ready.sh -- complete
-- initialize complete (SCF_READY)
<[Enter]キーを押す>
localhost login: edu01
Password: ****
XSCF>
```

← 「y」を入力

←メッセージを確認

←作成したユーザーでログイン

12) ホスト名が変更されていることを確認します。

```
XSCF> showhostname -a
bb#00:xscf0-hostname.example.com
```

13) IP アドレスの設定を確認します。

IP アドレス (inet addr) とサブネットマスク (Mask) が設定されていることを確認します。

```
XSCF> shownetwork bb#00-lan#0
bb#00-lan#0
    Link encap:Ethernet  HWaddr B0:99:28:A1:9B:F9
    inet addr:192.168.10.20  Bcast:192.168.10.255  Mask:255.255.255.0
    UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
    RX packets:164 errors:0 dropped:0 overruns:0 frame:0
    TX packets:45 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:1000
    RX bytes:13830 (13.5 KiB)  TX bytes:2598 (2.5 KiB)
    Base address:0x2000
```

14) DNS サーバの設定を確認します。

ネームサーバ (nameserver) の IP アドレスとドメイン名 (search) が設定されていることを確認します。

```
XSCF> shownameserver
nameserver 192.168.10.100
search nsserver.com
```

15) ルーティング環境を確認します。

bb#00 の XSCF-LAN#0 にネットワークアドレス (Destination)、サブネットマスク (Netmask)、デフォルトゲートウェイ (Gateway) が設定されていることを確認します。

```
XSCF> showroute -a
Destination      Gateway          Netmask          Flags  Interface
192.168.10.0     *               255.255.255.0    U      bb#00-lan#0
default         192.168.10.1    0.0.0.0          UG     bb#00-lan#0
```

ここでシリアル接続を切断します。

以降の作業は、新規で作成したユーザー「edu01」で XSCF に telnet 接続して実施します。

2.5. NTP サーバの設定【任意設定】

1) NTP サービスの状態を確認します。

XSCF が NTP サーバまたは NTP クライアントとして設定されているかどうかを確認します。

デフォルトでは、NTP サーバと NTP クライアントの両方の設定が無効 (disable) になっています。

```
XSCF> showntp -a
client : disable
server : disable
```

2) NTP サーバのサービスを有効化します。

XSCF に NTP サーバの機能を持たせるため、NTP の設定を有効 (enable) にします。

```
XSCF> setntp -s server -c enable
Please reset the XSCF by rebootxscf to apply the ntp settings.
```

🔊 NTP の設定後、XSCF の再起動が必要です。

3) XSCF をリセットします。

XSCF を再起動して、NTP サーバの設定を XSCF に反映させます。

```
XSCF> rebootxscf -a
```

XSCF をリセットする (rebootxscf コマンドを実行する) と、セッションは切断されます。

再度ユーザー「edu01」で XSCF に telnet 接続します。

4) NTP サーバの設定が有効(enable)になっていることを確認します。

```
XSCF> showntp -a
client : disable
server : enable
```


2.6. SNMP エージェントの設定【任意設定】

1) SNMP の設定内容を確認します。

デフォルトでは、SNMP エージェントの設定は無効(Disabled)になっています。

```
XSCF> showsnmp

Agent Status:      Disabled
Agent Port:        161
System Location:    Unknown
System Contact:     Unknown
System Description: Unknown

Trap Hosts: None
SNMP V1/V2c: None
Enabled MIB Modules: None
```

2) SNMP エージェントの設定をします。

ここでは、システムの設置場所に「NorthTower34F」、システム管理者のメールアドレスに「admin@example.com」、説明に「DataBaseServer」を設定します。

```
XSCF> setsnmp -l NorthTower34F -c admin@example.com -d DataBaseServer
```

3) SNMP エージェントを有効化します。

```
XSCF> setsnmp enable
setsnmp: Agent Enabled.
```

4) SNMP の設定が有効(Enabled)になっていることを確認します。

```
XSCF> showsnmp

Agent Status:      Enabled
Agent Port:        161
System Location:    NorthTower34F
System Contact:     admin@example.com
System Description: DataBaseServer

Trap Hosts: None
SNMP V1/V2c: None
Enabled MIB Modules:

SP MIB
```

2.7. XSCF Web の設定【任意設定】

XSCF Web には https でアクセスします。そのため、事前に認証局を設定する必要があります。
選択できる認証局には、以下の 3 種類があります。

- 外部認証局
- イン트라ネット内の認証局
- 自己認証局

本書では自己認証局を使用する手順を説明します。

1) https サービスの状態を確認します。

デフォルトでは、https サービスの設定は無効(disabled)になっています。

```
XSCF> showhttps
HTTPS status: disabled
```

2) https サービスを有効化します。

```
XSCF> sethttps -c enable
The web serverkey or web server certificate which has been signed by an external
certification authority does not exist.
Created self-signed certificate for HTTPS service.Continue? [y|n] :y ← 「y」を入力
```

👉 「enable」を指定し「y」を入力すると、次の 1 から 4 が自動的に設定されます。

1. XSCF の自己認証局を構築します。
2. XSCF のウェブサーバの秘密鍵を生成します。
3. XSCF で自己署名したウェブサーバ証明書を作成します。
4. HTTPS を有効にします。

3) https サービスが有効(enabled)になっていることを確認します。

ウェブサーバ証明書が作成されていることを確認します。

```
XSCF> showhttps
HTTPS status: enabled
Server key: installed in Feb 22 15:48:44 JST 2017
CA key: installed in Feb 22 15:48:39 JST 2017
CA cert: installed in Feb 22 15:48:39 JST 2017
CSR:
-----BEGIN CERTIFICATE REQUEST-----
MIICwTCCAaAqCAQAwFDELMAkGA1UEBhMCIAxETAPBgNVBAGMCEthbmFnYXdhMREw
DwYDVQQHDAhLYXdhc2FraTEYMBYGA1UECgwPRnVqaXRzdSBMaW1pdGVkMRgwFgYD
VQQLDA9GdWppdHN1IEVpbWl0ZWQxEzARBgNVBAMMCiBaNTE2MjYwMDcwggEiMAOG
CSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCXD84gUCsEteM7TVEfd1k0IUechd6N
2dzIY22IPL/ysONNCN/jzWe+Aq2HfqIu+ucx29I7uA6r1oxozoZ1E0dhmFgKPsqB
```

```
Q0Pk iOE1wTvsp869io9//D6kXa1csaQ7Q6y4T0daB08jT48n11vbFjkA0c82iym1
iClzCf1FjBvny1lCOMe2ksSvzIDJ3T4P0Yc1UqPihZ0ipTpGbNzXd/q3CKN7L8g7
Zzj2Np1bKaJpg5k7LzlwEVRAzQ3RxAJ7aXfIOdNIMor6Ix262tU6hdq+F1n607oi
KuHfgKmQhn3rXCP+IFJZcUnNzb/sdFSLEEWIa85dodhgxcNIXcb0f3GPAgMBAAGg
ADANBgkqhkiG9w0BAQsFAAOCAQEAG9xI8PaINZoIjlxIeuJbI6xurNUmlpkfrroN
YLxA1zE6EcmoIHBorZa40nPQd4DPJ1Y9t/5bCDbFt4TBISfM3Md5ak0R3QDbN4zF
OpDHXbkhwqVklA6pywFyg7IOFTs2kczbClCvkVhW550utvg/pHjLTszFgcIHw1gB
WlOCgj0x7HivoKCjnrFQECJicQcD0m3qrRVn3eRxW/GF8wy9jtAfUHqdf0zzHqVg
q/gmQZm2AJGS1UnxL1BYKMSejzIoePqDc5P5krIiwFVOC1vke07KfVgX264YpLEn
jMPIYq8CHTcZwE/hmKHsQNQTR0hozAHGIfcuTC3CcyiW1pSpYg==
```

-----END CERTIFICATE REQUEST-----

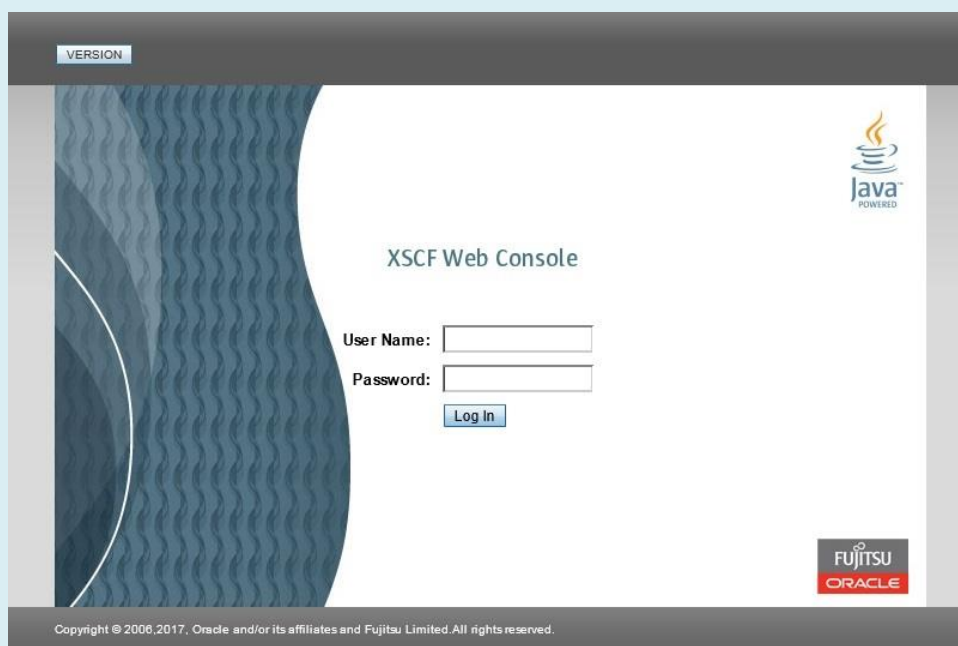
《参考》

● XSCF Web への接続

ブラウザに XSCF の IP アドレスまたはホスト名を指定して、XSCF へ接続します。

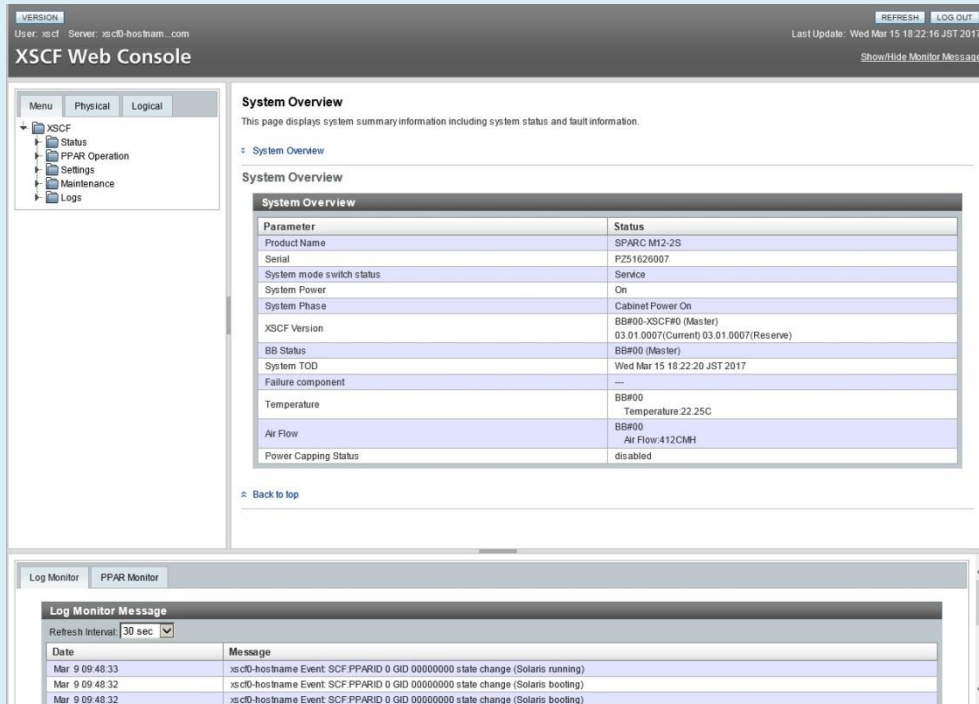
https://<XSCF の IP アドレス>/

● XSCF Web ログイン画面



👉 新規で作成したユーザー「edu01」でログインします。

- XSCF Web 画面




The screenshot displays the XSCF Web Console interface. At the top, it shows the user 'xscf' and server 'xscf0-hostnam.com'. The main content area is titled 'System Overview' and provides a summary of system status and fault information. A table lists various parameters and their current status.

Parameter	Status
Product Name	SPARC M12-2S
Serial	PZ51626007
System mode switch status	Service
System Power	On
System Phase	Cabinet Power On
XSCF Version	BB#00-XSCF#0 (Master) 03.01.0007(Current) 03.01.0007(Reserve)
BB Status	BB#00 (Master)
System TOD	Wed Mar 15 18:22:20 JST 2017
Failure component	---
Temperature	BB#00 Temperature 22.25C
Air Flow	BB#00 Air Flow 412CMH
Power Capping Status	disabled

Below the System Overview section, there is a 'Log Monitor' tab. It shows a 'Log Monitor Message' table with columns for Date and Message. The messages indicate state changes for Solaris running and booting.

Date	Message
Mar 9 09:48:33	xscf0-hostname Event: SCF PPARID 0 GUID 00000000 state change (Solaris running)
Mar 9 09:48:32	xscf0-hostname Event: SCF PPARID 0 GUID 00000000 state change (Solaris booting)
Mar 9 09:48:32	xscf0-hostname Event: SCF PPARID 0 GUID 00000000 state change (Solaris booting)

 XSCF Web を使用することで、GUI ベースで構成情報の設定や管理を行うことができます。

2.8. 消費電力制限(パワーキャッピング)の設定【任意設定】

1) パワーキャッピングの設定内容を確認します。

デフォルトでは、消費電力制限は無効(disabled)になっています。

```
XSCF> showpowercapping
activate_state      :disabled
powerlimit          :100%
timelimit           :30
violation_actions   :none
```

2) パワーキャッピングの設定をします。

ここでは、消費電力の上限値を 2000 W、消費電力の上限値を超えた場合の猶予時間を 100 秒に設定します。

```
XSCF> setpowercapping -s powerlimit_w=2000 -s timelimit=100
activate_state      :disabled -> -
powerlimit          :2798w      -> 2000w
timelimit           :30         -> 100
violation_actions   :none       -> -
The specified option will be changed.
Continue? [y|n]:y
Configuration complete.
activate_state      :disabled
powerlimit          :2000w
timelimit           :100
violation_actions   :none
```

← 「y」を入力

3) 設定内容が反映されていることを確認します。

消費電力の上限値と、消費電力の上限値を超えた場合の猶予時間が正しく設定されているかを確認します。

```
XSCF> showpowercapping
activate_state      :disabled
powerlimit          :2000w
timelimit           :100
violation_actions   :none
```

4) パワーキャッピングを有効化します。

```
XSCF> setpowercapping -s activate_state=enabled
activate_state      :disabled -> enabled
powerlimit          :2000w    -> -
timelimit           :100      -> -
violation_actions    :none     -> -
The specified option will be changed.
Continue? [y|n]:y      ← 「y」を入力
Configuration complete.
activate_state      :enabled
powerlimit          :2000w
timelimit           :100
violation_actions    :none
```

5) パワーキャッピングの設定が有効(enabled)になっていることを確認します。

```
XSCF> showpowercapping
activate_state      :enabled
powerlimit          :2000w
timelimit           :100
violation_actions    :none
```

2.9. メモリミラーの設定【任意設定】

1) デバイス情報を確認します。

デフォルトでは、メモリミラーの設定は無効(no)になっています。

```
XSCF> showfru -a
Device      Location      Memory Mirror Mode
sb          00-0
  cpu       00-0-0      no
  cpu       00-0-2      no
```

2) メモリミラーを有効化します。

setupfru コマンドの -m オプションの後ろに「y」(有効)を指定することで、メモリミラーを有効にします。

```
XSCF> setupfru -m y sb 00-0
```

Note:

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

3) メモリミラーモードが有効(yes)になっていることを確認します。

```
XSCF> showfru -a
Device      Location      Memory Mirror Mode
sb          00-0
  cpu       00-0-0      yes
  cpu       00-0-2      yes
```

2.10. CPU リソースの初期設定

1) CPU コアリソースの使用状況を確認します。

CPU コア アクティベーションにより使用を許可されている CPU コア数を確認します。

```
XSCF> showcodusage
Resource In Use Installed CoD Permitted Status
-----
PROC          0          24          6 OK: 6 cores available
PPAR-ID/Resource In Use Installed Assigned
-----
0 - PROC          0          24          0 cores
1 - PROC          0          0          0 cores
:
-<省略>-
:
15 - PROC          0          0          0 cores
Unused - PROC          0          0          6 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.

 showcodusage コマンドの出力結果には、次の情報が含まれます。

- ・上段 : サーバ全体の情報
- ・下段 : 物理パーティション (PPAR) ごとの情報
 - ・SPARC M12-2 は、単一の物理パーティションが表示されます。
 - ・SPARC M12-2S の場合、複数の物理パーティションを構築できます。
- ・In Use : 稼動している CPU コア数
- ・Installed : 物理的に搭載されている CPU コア数
- ・CoD Permitted : CPU コア アクティベーションにより使用を許可された CPU コア数
- ・Status : CPU コア アクティベーションの許諾違反の有無
- ・Assigned : 物理パーティション (PPAR) に割り当てられている CPU コア数

 Unused の Assigned に 6 コア、PPAR-ID 0 の Assigned に 0 コアが表示された場合、使用を許可されたコアは 6 コア存在するが、PPAR にはまだコアが割り当てられていない状態であることを意味します。

《参考》

● 物理パーティション(PPAR)に割り当てられている CPU コア数の確認

showcod コマンドでも、使用を許可された CPU コア数と、PPAR に割り当てられた CPU コア数を確認できます。

```
XSCF> showcod
PROC Permits installed: 6 cores
PROC Permits assigned for PPAR 0: 0
:
-<省略>-
```

2) 物理パーティション(PPAR)に CPU コアを割り当てます。

【書式】 setcod -p PPAR-ID -s cpu PPAR に割り当てる CPU コア数

PPAR に割り当てる CPU コアの総数を指定します。追加する CPU コア数ではないので注意してください。ここでは、PPAR 0 に 4 コア分の CPU を割り当てています。

```
XSCF> setcod -p 0 -s cpu 4
```

👉 使用を許可された CPU コア数を上限値として割り当てることができます。

3) CPU コアリソースを確認します。

```
XSCF> showcodusage
Resource In Use Installed CoD Permitted Status
-----
PROC          0          24          6 OK: 6 cores available
PPAR-ID/Resource In Use Installed Assigned
-----
0 - PROC          0          24          4 cores
:
-<省略>-
:
Unused - PROC          0          0          6 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.

👉 PPAR-ID 0 の Assigned を見ると、4 コアが割り当てられています。

4) 物理パーティション(PPAR)に割り当てられている CPU コア数を確認します。

```
XSCF> showcod  
PROC Permits installed: 6 cores  
PROC Permits assigned for PPAR 0: 4  
:  
-<省略>-  
:  
PROC Permits assigned for PPAR 15: 0
```

2.11. 物理パーティションの起動／停止とコンソール接続【任意設定】

1) 物理パーティション起動時に OS が自動起動(auto-boot)しないように設定します。

※ OS を再インストールする場合などで、物理パーティション起動後に OBP の状態に遷移させたいとき、本設定を実施します。

※ OS の自動起動設定は、運用ポリシーに従って設定してください。

XSCF 上で OpenBoot PROM (OBP) 環境変数である auto-boot の設定を「false」に変更します。

【書式】 setpparparam -p PPAR-ID -s bootscript "OBP 環境変数設定"

```
XSCF> setpparparam -p 0 -s bootscript "setenv auto-boot? false"
PPAR-ID of PPARs that will be affected:0
OpenBoot PROM variable bootscript will be changed.
Continue? [y|n] :y ← 「y」を入力
```

🔊 auto-boot の設定は次回起動時(次回 poweron 時)に OBP 環境に反映されます。

2) 物理パーティションを起動します。

```
XSCF> poweron -p 0
PPAR-IDs to power on :00
Continue? [y|n] :y ← 「y」を入力
00 : Powering on

*Note*
This command only issues the instruction to power-on.
The result of the instruction can be checked by the "showpparprogress".
```

🔊 物理パーティションが起動するまで 5 分ほどかかります。

3) 物理パーティションの状態を確認します。

```
XSCF> showpci -p 0
PPAR-ID  LSB  PSB  Status
00                Running
                00  00-0
```

🔊 物理パーティションの状態が「Running」であることを確認します。

4) OS 環境の状態を確認します。

```
XSCF> showdomainstatus -p 0
Logical Domain Name  Status
primary             OpenBoot Running
```

☞ ドメイン(OS)の状態が「OpenBoot Running」であることを確認します。

☞ auto-boot 設定を false にしているため、OS は自動起動しません。

5) 物理パーティションのコンソールに接続します。

ここでは、PPAR-ID 0 のコンソールに接続します。

```
XSCF> console -p 0

Console contents may be logged.
Connect to PPAR-ID 0?[y|n] :y      ← 「y」を入力
<[Enter]キーを押す>
{0} ok
```

☞ ok プロンプトが表示されることを確認します。

6) コンソール接続を切断します。

「#.(シャープ+ドット)」を入力してコンソール接続を切断します。

```
{0} ok #.
exit from console.
XSCF>
```

☞ 実際の実行環境では、「#」は画面上には表示されません。

☞ コンソール接続を切断すると、XSCF プロンプトが表示されます。

7) 物理パーティションを停止します。

```
XSCF> poweroff -p 0
PPAR-IDs to power off :00
Continue? [y|n] :y      ← 「y」を入力
00 : Powering off

*Note*
This command only issues the instruction to power-off.
The result of the instruction can be checked by the "showpparprogress".
```

8) 物理パーティションの状態を確認します。

```
XSCF> showpci -p 0
PPAR-ID  LSB  PSB  Status
00              Powered Off
          00    00-0
```

 物理パーティションの状態が「Powered Off」であることを確認します。

物理パーティションには Oracle Solaris 11 がプレインストールされています。プレインストール OS をそのまま使用するか、Solaris の再インストールを行うかを用途に合わせて選択してください。

Oracle Solaris 11 を再インストールする場合には、最新の SRU を適用してください。

Oracle Solaris 11 のインストールおよび SRU の適用方法は『Oracle Solaris 11 を使ってみよう』をご参照ください。

- 「技術情報 Technical Park」 > ドキュメント > オペレーティングシステム

<http://www.fujitsu.com/jp/sparc-technical/document/solaris/index.html#os>

3. CPU コア アクティベーション

CPU コア アクティベーションとは、SPARC M12 に搭載されている CPU を CPU コア単位で有効化できる機能です。システム移動中に、業務負荷の変化に合わせて、移動させる CPU コア の数を柔軟に変更できます。

本章では、CPU コア アクティベーション機能を使用した CPU コアの追加手順を説明します。事前に CPU コア アクティベーションを購入する必要があります。

本章では、Oracle Solaris 11.3 のインストール、および修正プログラム SRU17011 (Oracle バージョン: SRU11.3.18.6.0) の適用を行った環境で実施する手順を説明します。

Oracle Solaris 11 のインストールおよび SRU の適用方法は『Oracle Solaris 11 を使ってみよう』をご参照ください。

- 「技術情報 Technical Park」 > ドキュメント > オペレーティングシステム

<http://www.fujitsu.com/jp/sparc-technical/document/solaris/index.html#os>

3.1. OS 上での CPU リソースの確認

以降の手順では、XSCF 環境での操作を行います。コンソールに接続した状態からの操作を想定しています。

1) 物理パーティションを起動します。

```
XSCF> poweron -p 0
PPAR-IDs to power on :00
Continue? [y|n] :y      ← 「y」を入力
00 : Powering on
```

2) XSCF から物理パーティションのコンソールに接続します。

```
XSCF> console -p 0

Console contents may be logged.
Connect to PPAR-ID 0?[y|n] :y          ←「y」を入力
<[Enter]キーを押す>
{0} ok boot
Boot device: disk File and args:
SunOS Release 5.11 Version 11.3 64-bit
Copyright (c) 1983, 2016, Oracle and/or its affiliates. All rights reserved.
Hostname: m12-2s-00

m12-2s-00 console login: user01        ←一般ユーザーでログイン
Password: ****
Last login: Wed Feb 22 12:55:25 2017 on console
Oracle Corporation      SunOS 5.11      11.3      December 2016
$
$ su -
Password: ****                    ←root 権限へ切り替え
```

 コンソールに接続し、一般ユーザーで OS にログインしたあと、root 権限へ切り替えます。

3) OS が認識している CPU コア数を確認します。

```
# psrinfo -vp
The physical processor has 4 cores and 32 virtual processors (0-31)
The core has 8 virtual processors (0-7)
The core has 8 virtual processors (8-15)
The core has 8 virtual processors (16-23)
The core has 8 virtual processors (24-31)
SPARC64-XII (chipid 0, clock 4250 MHz)
```

 実行例では、4CPU コア、32 スレッドが割り当てられています。

4) CPU コアリソースの使用状況を確認します。

OS 上で、使用を許可されている CPU コア数(PERMITTS)と、OS に割り当てられている CPU コア数(IN USE)を確認します。

```
# ldm list-permits

CPU CORE
PERMITTS (PERMANENT)  IN USE  REST
4          (4)        4        0
```

 実行例では、4CPU コアの使用を許可されており、すべて OS に割り当てられています。

5) XSCF 環境へ移動します。

「#.(シャープ+ドット)」を入力して XSCF 環境へ移動します。

```
# #. exit from console.  
XSCF>
```

 実際の実行環境では、「#.(シャープ+ドット)」は画面上には表示されません。

3.2. CPU コア アクティベーション機能による CPU コアの追加【任意設定】

1) CPU コアリソースの使用状況を確認します。

CPU コア アクティベーションで使用を許可された CPU コア数を確認します。

```
XSCF> showcodusage
Resource In Use Installed CoD Permitted Status
-----
PROC          4          24          6 OK: 2 cores available
PPAR-ID/Resource In Use Installed Assigned
-----
0 - PROC          4          24          4 cores
:
-<省略>-
:
Unused - PROC          0          0          2 cores
:
-<省略>-
```

🔊 showcodusage コマンドの出力結果には、次の情報が含まれます。

- ・上段 : システム全体の情報
- ・下段 : 物理パーティション (PPAR) ごとの情報
 - ・SPARC M12-2 は、単一の物理パーティションが表示されます。
 - ・SPARC M12-2S の場合、複数の物理パーティションを構築できます。
- ・In Use : 稼動している CPU コア数
- ・Installed : 物理的に搭載されている CPU コア数
- ・CoD Permitted : CPU コア アクティベーションにより使用を許可された CPU コア数
- ・Status : CPU コア アクティベーションの許諾違反の有無
- ・Assigned : 物理パーティション (PPAR) に割り当てられている CPU コア数

🔊 In Use の値は、XSCF の更新タイミングによって最新の値になっていない場合があります。最新の値が表示されるまで最大で 20 分かかる場合があります。

2) 物理パーティション (PPAR) に割り当てられている CPU コア数を確認します。

showcod コマンドでも、使用を許可された CPU コア数と、PPAR に割り当てられた CPU コア数を確認できます。

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

3) 登録されている CPU コア アクティベーションキーの数を確認します。

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

4) CPU コア アクティベーションキーの詳細情報を確認します。

CPU コア アクティベーションキーのキーデータを確認します。

```
XSCF> showcodactivation -r -i 0
*Index0
Product: SPARC M12-2S
SequenceNumber: 1474
Cpu: noExpiration 1
Text-Signature-SHA256-RSA2048:
CBgTTtiCUF0VzyKyB3jmcFK8lJa/t+ . . . . .
:
-<省略>-
XSCF> showcodactivation -r -i 1
*Index1
Product: SPARC M12-2S
SequenceNumber: 1475
Cpu: noExpiration 1
Text-Signature-SHA256-RSA2048:
EsuU1I+eTigjEwGCFwpN8MwPi09kb1 . . . . .
:
-<省略>-
XSCF> showcodactivation -r -i 2
*Index2
Product: SPARC M12-2S
SequenceNumber: 1476
Cpu: noExpiration 1
Text-Signature-SHA256-RSA2048:
aGTaioi09Pk80wgLxCBigy5eaiW0aM . . . . .
:
-<省略>-
```

5) CPU コア アクティベーションキーを追加します。

""(ダブルクォーテーション)で囲む中に、すべてのキーデータをコピー & ペーストして指定します。

```
XSCF> addcodactivation "Product: SPARC M12-2S
```

```
> SequenceNumber: 1480
```

```
> Cpu: noExpiration 1
```

```
> Text-Signature-SHA256-RSA2048:
```

```
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx==
```

```
> " <[Enter]キーを押す>
```

```
Above Key will be added, Continue?[y|n]:y
```

※ 実際の内容は省略

← 「y」を入力

🔊 CPU コア アクティベーションは事前に購入する必要があります。

🔊 CPU コア アクティベーションキーを 1 つ追加すると、1 コア使用できるようになります。

例) 許諾キー内容 (cod_key_M12_2S_1481.dsf)

```
Product: SPARC M12-2S
```

```
SequenceNumber: 1481
```

```
Cpu: noExpiration 1
```

```
Text-Signature-SHA256-RSA2048:
```

```
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
xxxxxxxxxxxxxxxxxxxxxxxxxxxx==
```

※ 実際の内容は省略

6) CPU コアリソースの使用状況を確認します。

CPU コア アクティベーションにより、1CPU コアを追加で使用できるようになったことを確認します。

```
XSCF> showcodusage
Resource In Use Installed CoD Permitted Status
-----
PROC          4          24          7 OK: 3 cores available
PPAR-ID/Resource In Use Installed Assigned
-----
0 - PROC          4          24          4 cores
:
-<省略>-
:
Unused - PROC          0          0          3 cores
:
-<省略>-
```

7) 物理パーティションに割り当てられている CPU コア数を確認します。

```
XSCF> showcod
PROC Permits installed: 7 cores
PROC Permits assigned for PPAR 0: 4
:
-<省略>-
```

☞ 使用を許可された CPU コア数が 1 コア増えて 7 コアになっています。

8) 物理パーティション(PPAR)に CPU コアを割り当てます。

ここでは、PPAR-ID 0 に 7 コア分の CPU を割り当てています。

```
XSCF> setcod -p 0 -s cpu 7
```

- ☞ PPAR に割り当てる数の CPU コアの総数を指定します。追加する CPU コア数ではないので注意してください。
- ☞ 指定は、1 コア単位で可能です。
- ☞ 使用を許可された CPU コア数を上限値として割り当てることができます。

9) CPU コアリソースの使用状況を確認します。

```
XSCF> showcodusage
Resource In Use Installed CoD Permitted Status
-----
PROC          4          24          7 OK: 3 cores available
PPAR-ID/Resource In Use Installed Assigned
-----
0 - PROC          4          24          7 cores
:
-<省略>-
```

👉 PPAR-ID 0 に 7 コア分の CPU が割り当てられたことを確認します。

10) 物理パーティションに割り当てられている CPU コア数を確認します。

```
XSCF> showcod
PROC Permits installed: 7 cores
PROC Permits assigned for PPAR 0: 7
```

👉 使用を許可された 7 コアのうち、すべての CPU コアを PPAR-ID 0 に割り当てられています。

11) XSCF から物理パーティションのコンソールに接続します。

```
XSCF> console -p 0

Console contents may be logged.
Connect to PPAR-ID 0?[y|n] :y          ← 「y」を入力

m12-2s-00 console login: user01        ←一般ユーザーでログイン
Password:*****
Last login: Wed Feb 22 16:40:35 2017 on console
Oracle Corporation      SunOS 5.11      11.3      December 2016
$
$ su -                                ←root 権限へ切り替え
Password:*****
```

👉 コンソールに接続し、一般ユーザーで OS にログインしたあと、root 権限へ切り替えます。

3.3. OS 環境での動的 CPU リソース追加【任意設定】

XSCF 環境で CPU コアリソースを物理パーティションへ割り当てたあとに、OS 環境で各ドメインに CPU リソースを追加する必要があります。ここでは、物理サーバ環境(制御ドメイン:primary)に CPU を追加します。

1) 現在 OS で使用できる CPU コア数を確認します。

```
# psrinfo -vp
The physical processor has 4 cores and 32 virtual processors (0-31)
The core has 8 virtual processors (0-7)
The core has 8 virtual processors (8-15)
The core has 8 virtual processors (16-23)
The core has 8 virtual processors (24-31)
SPARC64-XII (chipid 0, clock 4250 MHz)
```

2) CPU リソースの使用状況を確認します。

```
# ldm list-permits

CPU CORE
PERMITS (PERMANENT)    IN USE    REST
7                (7)        4         3
```

🟢 使用を許可されている CPU コア数 (PERMITS) が増えていることを確認します。

🟢 追加された CPU コアはまだ OS に割り当てられていないため、未使用の CPU コア数 (REST) が 3 になっています。

3) CPU コア数を変更します。

ここでは、OS に 7 コア分の CPU コア数を割り当てています。

```
# ldm set-core 7 primary
```

4) CPU コア数を再度確認します。

```
# psrinfo -vp
The physical processor has 4 cores and 32 virtual processors (0-31)
  The core has 8 virtual processors (0-7)
  The core has 8 virtual processors (8-15)
  The core has 8 virtual processors (16-23)
  The core has 8 virtual processors (24-31)
  SPARC64-XII (chipid 0, clock 4250 MHz)
The physical processor has 3 cores and 24 virtual processors (32-55)
  The core has 8 virtual processors (32-39)
  The core has 8 virtual processors (40-47)
  The core has 8 virtual processors (48-55)
  SPARC64-XII (chipid 1, clock 4250 MHz)
```

👉 OS が 7 コア分の CPU コアを認識していることを確認します。

5) CPU リソースの使用状況を再度確認します。

```
# ldm list-permits

CPU CORE
PERMITS (PERMANENT)    IN USE    REST
7          (7)         7          0
```

👉 使用中の CPU コア数 (IN USE) が 7 コアに増えていることを確認します。すべての CPU が OS に割り当てられたので、未使用の CPU コア数 (REST) は 0 になります。

3.4. CPU コアの一時利用機能【任意設定】

CPU コアの一時利用機能(Interim Permit)とは、アクティベートされていないすべての CPU コアリソースを 30 日間の期限付きで利用できる機能です。注文した CPU コア アクティベーションが届くまでの間、一時的に追加の CPU リソースを使用できるので、CPU リソースが不足し、直ちに CPU リソースが必要な場合に有効です。

ただし、CPU コアの一時利用機能を一度有効にすると、新規で CPU コア アクティベーションキーを追加して PPAR に割り当てるまで、本機能は再度使用できなくなります。誤って設定しないよう注意してください。

3.4.1. CPU コア一時利用機能の有効化

1) CPU コアの一時利用機能の設定状態を確認します。

```
XSCF> showinterimpermit
Interim Permit for PPAR 0: disabled
Interim Permit for PPAR 1: disabled
:
-<省略>-
```

 CPU コアの一時利用機能を一度も有効にしたことがない場合は「disabled」と表示されます。

2) CPU コアの一時利用機能を有効化します。

```
XSCF> setinterimpermit -p 0 -c enable

Note:
  Please add CPU Activation(s) within 30 days of enabling the Interim Permit.

The Interim Permit for the PPAR will be changed to enabled.
Continue? [y|n] :y          ← 「y」を入力

Completed.
```


3) CPU コアの一時利用機能が有効になったことを確認します。

```
XSCF> showinterimpermit
Interim Permit for PPAR 0: enabled [29 days remaining]
Interim Permit for PPAR 1: disabled
:
-<省略>-
```

- ☞ CPU コアの一時利用機能を有効にすると、「enabled」となり有効期限の日数が表示されます。
- ☞ 一時利用が可能になった CPU コアについては、setcod コマンドで PPAR に割り当てることなく、直接、論理ドメインに割り当てることができます。論理ドメインへの割り当てについては、「3.3 OS 環境での動的 CPU リソース追加【任意設定】」をご参照ください。
- ☞ 有効期限切れとなった場合は「expired」と表示されます。この際、購入した数を上回る CPU コア アクティベーション数が PPAR に割り当てられている場合は、自動的に削除されます。

3.4.2. 一時利用する CPU コアのドメインへの割り当て

1) XSCF から物理パーティションのコンソールに接続します。

```
XSCF> console -p 0

Console contents may be logged.
Connect to PPAR-ID 0?[y|n] :y          ← 「y」を入力

m12-2s-00 console login: user01        ←一般ユーザーでログイン
Password:*****
Last login: Wed Feb 22 12:55:25 2017 on console
Oracle Corporation      SunOS 5.11      11.3      December 2016
$
$ su -
Password:*****                      ←root 権限へ切り替え
```

- ☞ コンソールに接続し、一般ユーザーで OS にログインしたあと、root 権限へ切り替えます。

2) OS で使用できる CPU コア数を確認します。

```
# ldm list-permits

CPU CORE
PERMITS (PERMANENT)    IN USE    REST
24                    11        13
```

- ☞ 上記例は、元々11コアがOSに割り当てられている状態のときに、CPU コアの一時利用機能を有効にした場合の結果です。
- ☞ CPU コアの一時利用機能を有効にしたことで、24 コア (SPARC M12-2S の最大コア数) が使用可能になっています。

3) 使用可能になったコアをドメインに割り当てます。

```
# ldm ls
NAME                STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary             active   -n-c--   UART    88      259584M   0.0%    0.0%    29d 21h 55m
#
# ldm add-core 1 primary      ←1 コアを追加
#
# ldm ls
NAME                STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary             active   -n-c--   UART    96      259584M   0.4%    0.2%    29d 21h 55m
```

☞ 上記例では、1 コアを制御ドメインに追加しています。

☞ SPARC M12-2S は 1 コア 8 スレッドのため、1 コアを追加すると VCPU が 8 個増加します。

4) 現在の使用コアを確認します。

```
# ldm list-permits

CPU CORE
PERMITS (PERMANENT)  IN USE    REST
24      (24)         12        12
```

☞ 使用中の CPU コア数 (IN USE) が 1 コア増加して 12 コアになっています。

5) XSCF 環境へ移動します。

「#.(シャープ+ドット)」を入力して XSCF 環境へ移動します。

```
# #. exit from console.
XSCF>
```

3.4.3. 新規 CPU コア アクティベーションの追加

ここからの手順は、注文した CPU コア アクティベーションキーが届いてから実施してください。

1) CPU コアリソースの使用状況を確認します。

```
XSCF> showcodusage
Resource In Use Installed CoD Permitted Status
-----
PROC      12      24      11 VIOLATION: 1 cores in excess
PPAR-ID/Resource In Use Installed Assigned
-----
0 - PROC      12      24      11 cores
:
-<省略>-
Unused - PROC      0      0      -1 cores
```

🔊 CPU コアの一時利用機能により、OS 上でアクティベートしているコア以上に割り当てているため、コアが超過した状態となります。上記例では、11 コアがアクティベートされているところ、OS 上で 12 コア使用しているため、1 コア超過しています。

2) CPU コア アクティベーションキーを追加します。

```
XSCF> addcodactivation "Product: SPARC M12-2S
> SequenceNumber: 1485
> Cpu: noExpiration 1
> Text-Signature-SHA256-RSA2048:
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
> " <[Enter] キーを押す>
Above Key will be added, Continue?[y|n]:y
```

※ 実際の内容は省略

← 「y」を入力

3) 物理パーティション(PPAR)に CPU コアを割り当てます。

ここでは、PPAR-ID 0 に 12 コア分の CPU を割り当てています。

```
XSCF> setcod -p 0 -s cpu 12
```

- 🔊 PPAR に割り当てる数の CPU コアの総数を指定します。追加する CPU コア数ではないので注意してください。
- 🔊 指定は、1 コア単位で可能です。
- 🔊 使用を許可された CPU コア数を上限値として割り当てることができます。

4) CPU コアリソースの使用状況を確認します。

```
XSCF> showcodusage
Resource In Use Installed CoD Permitted Status
-----
PROC          12          24          12 OK: 0 cores available
PPAR-ID/Resource In Use Installed Assigned
-----
0 - PROC          12          24          12 cores
:
-<省略>-
```

 PPAR-ID 0 に 12 コア分の CPU が割り当てられたことを確認します。

3.4.4. CPU コア一時利用機能の無効化

1) CPU コアの一時利用機能を無効化します。

```
XSCF> setinterimpermit -p 0 -c disable
The Interim Permit will be disabled.
Continue? [y|n] : y
```

Completed.


2) CPU コアの一時利用機能が無効化されたことを確認します。

```
XSCF> showinterimpermit -v
PPAR-ID: 0
Status: Interim Permit is disabled (can be enabled)

CPU Activation Information from the last time Interim Permit was enabled:
Registered CPU Activation Keys (in units of cores): 11
Purchased Cores Assigned to PPAR: 11

Current CPU Activation Information:
Registered CPU Activation Keys (in units of cores): 12
Purchased Cores Assigned to PPAR: 12
:
-<省略>-
```

 Status が「Interim Permit is disabled (can be enabled)」と表示されれば、再度、一時利用機能を有効にすることができます。

 再度、一時利用機能を有効にするためには、前回一時利用機能を使用した際のコア数より、現在のコア数の方が大きい値である必要があります。上記の例では、前回のコア数が 11 コアで現在のコア数が 12 コアです。

《参考》

- showinterimpermit コマンドで-v オプションを使用しない場合

```
XSCF> showinterimpermit
Interim Permit for PPAR 0: cannot be enabled again
Interim Permit for PPAR 1: disabled
:
-<省略>-
```

☞ -v オプションを使用しなかった場合、一時利用機能を再利用する条件を満たしていたとしても、「cannot be enabled again」と表示され、正確な情報が表示されません。確認をする際は必ず-v オプションを付けてください。

3) XSCF から物理パーティションのコンソールに接続します。

```
XSCF> console -p 0

Console contents may be logged.
Connect to PPAR-ID 0?[y|n] :y          ←「y」を入力

m12-2s-00 console login: user01        ←一般ユーザーでログイン
Password:*****
Last login: Wed Feb 22 12:55:25 2017 on console
Oracle Corporation      SunOS 5.11      11.3      December 2016
$
$ su -
Password:*****                        ←root 権限へ切り替え
```

☞ コンソールに接続し、一般ユーザーで OS にログインしたあと、root 権限へ切り替えます。

4) 現在の使用コアを確認します。

```
# ldm list-permits

CPU CORE
PERMITS (PERMANENT)    IN USE    REST
12          (12)        12         0
```

☞ OS 上では 12 コア使用しており、割り当てられているコア数も 12 になっています。

5) ドメインに割り当てられているコア数を確認します。

```
# ldm ls
NAME                STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary             active   -n-c--   UART    96      259584M   0.0%    0.0%    29d 23h 9m
```

☞ 一時利用機能を無効にしても、問題なく 12 コア (96 スレッド) が割り当てられています。

改版履歴

改版日	版数	改版内容
2017 年 8 月	1.0	新規作成

