

Fujitsu Server
PRIMERGY CDI V1.0
システム運用管理者ガイド
- Composable Disaggregated Infrastructure -



はじめに

本書は、PRIMERGY CDI V1.0 システムの運用・保守で行う作業について、システム管理者の方を対象に説明しています。本書を読むにあたっては、Linux に関する基本的な知識が必要になります。

なお、本書は PRIMERGY CDI V1.0 システムの運用・保守以外に利用することはできません。

略称／用語

本書で使用する略称と用語を以下に示します。

正式名称	本書での略称
Composable Disaggregated Infrastructure	CDI
PRIMERGY Composable Disaggregated Infrastructure	PRIMERGY CDI
Compose Manager for CDI	CDI 管理ソフトウェア
Command Line Interface	CLI
Fujitsu Server PRIMERGY CDI V1.0 システム構築手順書	システム構築手順書
Fujitsu Server PRIMERGY CDI V1.0 ソフトウェア管理者ガイド	ソフトウェア管理者ガイド
FUJITSU Software ServerView Suite iRMC S6 Web インターフェース 2.x	iRMC 操作手引き
IB HCA カード(100Gb) [PY-HC331/332] IB HCA カード(100Gb) [PY-HC341/342] IB HCA カード(200Gb) [PY-HC401/402] ファームウェア アップデート Vxx.36.1010 手順書	NIC ファームウェア アップデート手順書
CDI コントローラアプライアンス	Director
PCIe ファブリックスイッチ (48port) for CDI	PCIe スイッチ
PCIe Box (PCIe×8) for CDI	PCIe Box
計算サーバ (PRIMERGY RX2530 M7、PRIMERGY RX2540 M7)	計算サーバ
PCIe SSD-960GB (RI) ×8 for CDI PCIe SSD-800GB (MU) ×8 for CDI	NVMe SSD
PCIe HBA カード for CDI	HBA

関連ドキュメント

関連するドキュメントを以下に示します。必要に応じて参照してください。

ドキュメント	概要	入手方法
CDI コントローラプライアンス 取扱説明書	ハードウェアの機能と仕様を記載しています。	富士通 CDI 製品情報サイトから入手出来ます。
PCIe ファブリックスイッチ (48port) for CDI 取扱説明書	同上	富士通 CDI 製品情報サイトから入手出来ます。
PCIe SSD-960GB (RI) ×8 for CDI 取扱説明書	同上	富士通 CDI 製品情報サイトから入手出来ます。
PCIe SSD-800GB (MU) ×8 for CDI 取扱説明書	同上	富士通 CDI 製品情報サイトから入手出来ます。
PCIe HBA カード for CDI 取扱説明書	同上	富士通 CDI 製品情報サイトから入手出来ます。
PCIe Box (PCIe×8) for CDI 取扱説明書	同上	富士通 CDI 製品情報サイトから入手出来ます。
システム構築手順書	運用前に実施する構築手順について記載しています。	富士通 CDI 製品情報サイトから入手出来ます。
ソフトウェア管理者ガイド	CDI 管理ソフトウェアについて記載しています。	富士通 CDI 製品情報サイトから入手出来ます。
iRMC 操作手引き	iRMC の操作手引き書です。	富士通製品サポートページから入手出来ます。
NIC ファームウェア アップデート手順書	ファームウェアのアップデート手順書について記載しています。	富士通製品サポートページから入手出来ます。

富士通 CDI 製品情報サイト

<https://www.fujitsu.com/jp/products/computing/servers/primergy/solution/cdi/>

富士通製品サポートページ

<https://support.ts.fujitsu.com/IndexDownload.asp?lng=jp>

改版履歴

版数	発行月	変更箇所	内容
初版	2023年7月	—	初版発行
1.1 版	2023年11月	全体 2.2 4.2.1 5.1 5.2.4 4.1.2	誤記を修正 P2P 関連の説明を追記 CDI 管理ソフトウェアのアップデート手順を見直し RAS 制限解除にともなうフローの見直し slot 番号の読み替えを削除 8slot Box 交換手順の追加
1.2 版	2023年11月	全体 2.3.10 3.20 4.2.2	体裁を修正 グループ・論理サーバの再構成ファイルの初期化を追加 グループ・論理サーバの再構成手順を見直し Director ファームウェアのアップデート手順を見直し
1.3 版	2024年1月	3.12, 3.13, 2.3.7, 2.3.8, 5.2.4(1) 2.2.15 4.2.2⑩	章の冒頭に、エラー状態のリソースの定義「op_statusが0以外」を追加。 説明文の中の (op_status=0 以外) ,(op_status=2)の記載を削除 GPU H100 を払い出す場合の注意事項の章を追加 Director ファームウェアのアップデート手順を見直し
1.4 版	2024年1月	5.2.4(1), (3)④ 4.4, 5.2.3, 5.2.3.3	Slot_number の参照先に 10 slot PCIe Box を追加 8S PCIe Box の front/rear 図を削除(5.2.4(1)のみ) "PCIe Box 8slot"の表記を"PCIe Box"に変更
1.5 版	2024年3月	3.1.6, 3.3.1, 3.20.5, 3.20.6, 4.2.1	誤記を修正

目次

第 1 章 概要	13
1.1 運用操作.....	14
1.2 保守操作.....	15
1.3 システムの監視.....	16
第 2 章 運用操作・保守操作・システムの監視で使用する機能	17
2.1 グループ操作.....	17
2.1.1 グループの情報を表示.....	17
2.1.2 グループの詳細情報を表示.....	17
2.1.3 グループの作成（コマンド指定）	18
2.1.4 グループの作成（ファイル指定）	18
2.1.5 グループへのリソース追加（コマンド指定）	19
2.1.6 グループへのリソース追加（ファイル指定）	19
2.1.7 グループからのリソース削除（コマンド指定）	20
2.1.8 グループからのリソース削除（ファイル指定）	20
2.1.9 グループの解体.....	20
2.2 論理サーバ操作.....	21
2.2.1 論理サーバの一覧を表示	21
2.2.2 論理サーバの詳細を表示	21
2.2.3 論理サーバの作成（コマンド指定）	22
2.2.4 論理サーバの作成（ファイル指定）	22
2.2.5 論理サーバへのリソース追加（コマンド指定）	23
2.2.6 論理サーバへのリソース追加（ファイル指定）	23
2.2.7 論理サーバからのリソース削除（コマンド指定）	24
2.2.8 論理サーバからのリソース削除（ファイル指定）	24
2.2.9 論理サーバの解体.....	24
2.2.10 論理サーバの電源オン.....	25
2.2.11 論理サーバの電源オフ.....	25

2.2.12	論理サーバの状態確認.....	25
2.2.13	論理サーバのPCIe バスの Peer to Peer(P2P)機能の設定確認.....	26
2.2.14	論理サーバのPCIe バスの P2P 機能の有効化/無効化.....	26
2.2.15	NVIDIA GPU H100 を払い出す場合の注意事項.....	26
2.3	リソース操作.....	28
2.3.1	リソースの一覧を表示.....	28
2.3.2	ファブリックフリープールのリソースを表示.....	28
2.3.3	グループフリープールのリソースを表示.....	29
2.3.4	グループに登録されているリソースを表示.....	29
2.3.5	論理サーバで使用されているリソースを表示.....	30
2.3.6	リソースの詳細を表示.....	30
2.3.7	グループフリープールのエラー状態のリソースを表示.....	31
2.3.8	論理サーバで使用されているエラー状態のリソースを表示.....	31
2.3.9	リソース ステータスの設定.....	31
2.3.10	リソース情報の同期.....	32
2.4	OS イメージ操作.....	33
2.4.1	OS イメージの一覧を表示.....	33
2.4.2	OS イメージの詳細を表示.....	33
2.4.3	OS イメージの登録.....	34
2.4.4	OS イメージの削除.....	34
2.4.5	OS イメージのインストール.....	34
2.5	構成バックアップファイル操作.....	35
2.5.1	構成バックアップファイルの一覧を表示.....	35
2.5.2	構成バックアップファイルをダウンロード.....	35
2.6	その他の操作.....	36
2.6.1	データの削除.....	36
2.6.2	iRMC 情報の表示 (計算サーバ).....	36
2.6.3	ログファイルの作成.....	36
2.6.4	Director のリセット.....	37

2.6.5 Director の状態表示.....	37
2.6.6 ファームウェアファイル (Director 用) のアップロード.....	37
2.6.7 ファームウェアファイル (Director 用) の一覧を表示.....	37
2.6.8 Director ファームウェアのアップデート.....	38
第 3 章 運用操作	39
3.1 システムを起動する	40
3.1.1 管理 LAN Switch の電源を投入	40
3.1.2 管理サーバの電源を投入	40
3.1.3 管理サーバのネットワーク設定を確認.....	40
3.1.4 CDI コンポーネントの電源を投入.....	41
3.1.5 Director の FAN の状態確認	41
3.1.6 CDI 管理ソフトウェア (管理サーバの Guest OS) を起動.....	42
3.1.7 CDI 管理ソフトウェアにログイン.....	42
3.1.8 論理サーバの電源をオン	42
3.2 システムを終了する	44
3.2.1 全論理サーバの利用停止	44
3.2.2 CDI 管理ソフトウェアにログイン.....	44
3.2.3 論理サーバの電源をオフ	44
3.2.4 CDI コンポーネントの電源を切断.....	44
3.3 CDI 管理ソフトウェアを再起動する	45
3.3.1 CDI 管理ソフトウェア (管理サーバの Guest OS) を停止.....	45
3.3.2 CDI 管理ソフトウェア (管理サーバの Guest OS) を起動.....	45
3.4 CDI 管理ソフトウェアにログインする	46
3.5 リソースを確認する	47
3.6 グループを作成する	48
3.6.1 使用できるリソースの確認.....	48
3.6.2 使用するリソースの選択	48
3.6.3 グループの作成.....	49

3.6.4 作成したグループの確認	49
3.7 グループにリソースを追加する	50
3.7.1 追加できるリソースの確認	50
3.7.2 追加するリソースの選択	50
3.7.3 リソースの追加	51
3.7.4 グループ内のリソースを確認	51
3.8 グループからリソースを削除する	52
3.8.1 削除できるリソースの確認	52
3.8.2 削除するリソースの選択	52
3.8.3 リソースの削除	53
3.8.4 グループ内のリソースを確認	53
3.9 論理サーバを作成する	54
3.9.1 使用できるリソースの確認	54
3.9.2 使用するリソースの選択	54
3.9.3 論理サーバの作成	55
3.9.4 ステータスの確認（論理サーバの起動前）	55
3.9.5 作成した論理サーバの確認	56
3.9.6 作成した論理サーバの電源オン	56
3.9.7 ステータスの確認（論理サーバの起動後）	56
3.10 論理サーバの構成を変更する（リソースの追加）	57
3.10.1 リソースを追加する論理サーバの確認	57
3.10.2 論理サーバの電源オフ	57
3.10.3 ステータスの確認（リソース追加前）	57
3.10.4 追加できるリソースの確認	58
3.10.5 追加するリソースの選択	58
3.10.6 リソースの追加	58
3.10.7 ステータスの確認（リソース追加後）	59
3.10.8 論理サーバのリソースを確認	59
3.10.9 論理サーバの電源オン	59

- 3.10.10 ステータスの確認（論理サーバの起動後）59
- 3.11 論理サーバの構成を変更する（リソースの削除） 60
 - 3.11.1 リソースを削除する論理サーバの確認60
 - 3.11.2 論理サーバの電源オフ60
 - 3.11.3 ステータスの確認（リソース削除前）60
 - 3.11.4 削除できるリソースの確認61
 - 3.11.5 削除するリソースの選択61
 - 3.11.6 リソースの削除61
 - 3.11.7 ステータスの確認（リソース削除後）62
 - 3.11.8 論理サーバのリソースを確認62
 - 3.11.9 論理サーバの電源オン62
 - 3.11.10 ステータスの確認（論理サーバの起動後）62
- 3.12 論理サーバからエラー状態のリソースを削除する 63
 - 3.12.1 エラー状態のリソースを確認63
 - 3.12.2 エラー状態のリソースが組み込まれた論理サーバの特定63
 - 3.12.3 論理サーバの電源オフ63
 - 3.12.4 ステータスの確認（論理サーバの電源オフ後）64
 - 3.12.5 エラー状態のリソースを削除64
 - 3.12.6 ステータスの確認（エラー状態のリソース削除後）64
 - 3.12.7 エラー状態のリソースが削除されたことを確認64
 - 3.12.8 エラー状態のリソースをファブリックフリープールに移動64
- 3.13 グループフリープールからエラー状態のリソースを削除する 65
 - 3.13.1 エラー状態のリソースを確認65
 - 3.13.2 エラー状態のリソースを削除65
- 3.14 論理サーバを解体する 66
 - 3.14.1 論理サーバの確認66
 - 3.14.2 解体する論理サーバの選択66
 - 3.14.3 論理サーバの電源オフ66
 - 3.14.4 ステータスの確認67

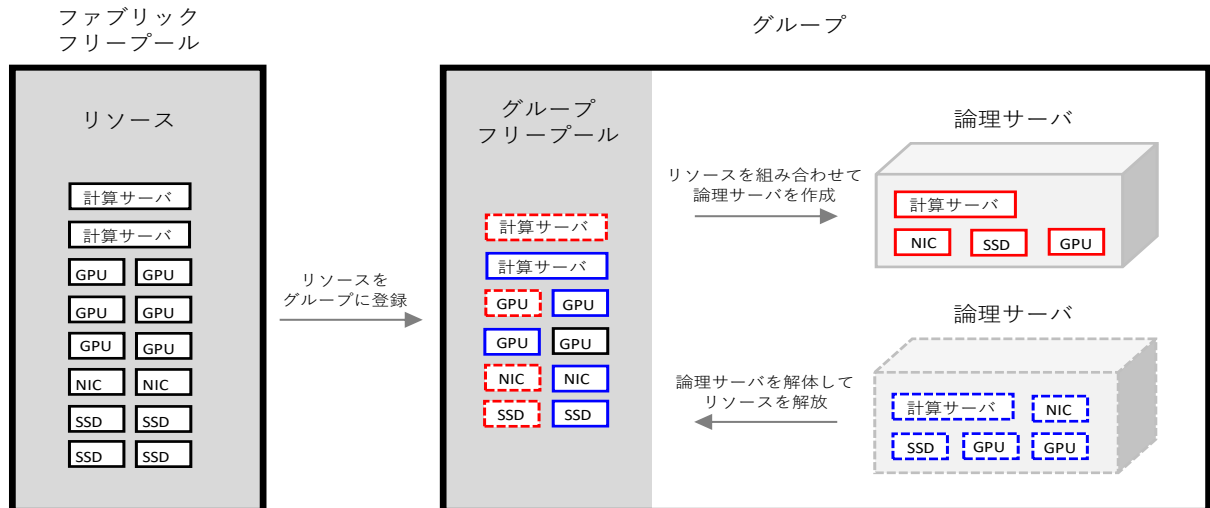
3.14.5 論理サーバを解体.....	67
3.14.6 解体の確認.....	67
3.15 グループを解体する	68
3.15.1 グループの確認.....	68
3.15.2 グループ内の論理サーバを確認.....	68
3.15.3 グループ内の論理サーバを解体.....	68
3.15.4 グループを解体.....	68
3.16 OS イメージを作成する	69
3.16.1 イメージ作成用論理サーバの作成.....	69
3.16.2 OS 種類およびカーネル版数の決定	70
3.16.3 ISO イメージの取得.....	70
3.16.4 ISO イメージのインストール.....	70
3.16.5 grub の設定.....	77
3.16.6 OS イメージの作成.....	77
3.16.7 OS イメージをシステム管理者の PC に転送.....	78
3.16.8 論理サーバの解体.....	78
3.17 論理サーバに OS イメージをインストールする.....	79
3.17.1 使用できる OS イメージの確認.....	79
3.17.2 インストールする OS イメージの選択.....	79
3.17.3 インストール先の論理サーバを確認.....	80
3.17.4 インストールを行うリソース(NVMe SSD)を選択	80
3.17.5 OS イメージをインストール.....	80
3.18 論理サーバへアクセスする	81
3.18.1 論理サーバの確認.....	81
3.18.2 論理サーバの電源オン	81
3.18.3 ステータスの確認.....	81
3.18.4 リソースの確認	81
3.18.5 iRMC 情報の確認.....	82
3.18.6 iRMC へログイン	82

3.18.7	ビデオリダイレクションの起動.....	82
3.18.8	IP アドレスの確認.....	82
3.18.9	論理サーバへのアクセス.....	82
3.19	データを削除する.....	83
3.19.1	データを削除するリソース (NVMe SSD) の確認.....	83
3.19.2	データを削除するリソース (NVMe SSD) の選択.....	83
3.19.3	データを削除.....	83
3.20	グループ・論理サーバを再構成する.....	84
3.20.1	使用できる構成バックアップファイルの確認.....	84
3.20.2	使用する構成バックアップファイルの選択.....	84
3.20.3	構成バックアップファイルのダウンロード.....	84
3.20.4	CDI 管理ソフトウェアからログアウト.....	84
3.20.5	構成バックアップファイルの解凍.....	85
3.20.6	グループ・論理サーバの再構成.....	85
3.20.7	論理サーバのステータスを確認.....	86
第 4 章	保守操作.....	87
4.1	ハードウェアの増設・減設・交換をする.....	88
4.1.1	グループ・論理サーバの再構成を行わない場合.....	88
4.1.2	グループ・論理サーバの再構成を行う場合.....	90
4.2	ソフトウェアをアップデートする.....	96
4.2.1	CDI 管理ソフトウェアのアップデート.....	96
4.2.2	Director ファームウェアのアップデート.....	98
4.2.3	NIC ファームウェアのアップデート.....	100
4.3	設定情報を管理する.....	101
4.3.1	iRMC の設定情報.....	101
4.3.2	BIOS の設定情報.....	101
4.4	活性保守交換の可否.....	102
第 5 章	システムの監視.....	103

5.1 故障箇所切り分けフロー	103
5.2 ハードウェアの状態を監視する	105
5.2.1 計算サーバ, HBA の監視	105
5.2.2 GPU,NVMe SSD,NIC の監視.....	105
5.2.3 その他の保守部品の監視	105
5.2.4 ハードウェアの故障箇所の特定	112
5.2.5 エラー状態となったリソースの扱い.....	117
5.3 NVMe SSD の寿命を確認する	118
5.3.1 フリープールに存在する NVMe SSD の寿命確認.....	118
5.3.2 論理サーバに組み込まれた NVMe SSD の寿命確認	120
第 6 章 システムのセキュリティ	122
6.1 証明書の設定	123
6.2 プロトコル、ポートの設定	124
6.3 初期パスワードの変更.....	125

第1章 概要

Fujitsu PRIMERGY CDI V1.0 システム（以下システムと表記）は、各種リソース（※1）をプール化し、業務に合わせた論理サーバを CDI 管理ソフトウェアで定義することでシステムの性能の最適化を実現します。



論理サーバは複数のリソースを組み合わせることで構成することができます。ファブリックフリープールに存在するリソース（※2）から必要なリソースをグループに登録することで、リソースをグループフリープールに移動させ、グループフリープールに移動したリソースを組み合わせることで論理サーバを作成します。（論理サーバは複数作成することが可能です）

また、使用しなくなった論理サーバを解体（※3）することで構成していたリソースを解放して、他の論理サーバで使用することも可能になります。

なお、グループへのリソースの登録や、論理サーバの作成・解体などの操作は、CLI コマンド、Director 制御コマンド（※4）を使用して行います。（※5）

※1 計算サーバ（CPU、メモリ、オンボード LAN も含む）、GPU、NIC、NVMe SSD

※2 初期状態では全てのリソースはファブリックフリープールにあります。

※3 解体により論理サーバに組み込まれていたリソースはグループフリープールに移動されます。

※4 各コマンドの詳細は、ソフトウェア管理者ガイドを参照してください。

※5 本書に記載している CLI コマンド、Director 制御コマンドは、システム構築手順書に記載している手順で構築された管理サーバの Host OS で使用することを想定しています。

- CLI コマンドの実行は、システム構築手順書「5.1.9 CDI 管理ソフトウェアにログインする」の手順で CDI 管理ソフトウェアにログインした後に `"/home/<Host OS のユーザ名>/cdi/RMcli_<version>/RMcli_<version>_<OSname>"` に格納されている `epcctl` の実行ファイルを使用する。

<Host OS のユーザ名> : システム構築手順書「4.3.1.4 OS インストール手順」で設定した Host OS のログイン ID（以下同様）

<version> : CDI 管理ソフトウェアの提供版数。詳細はシステム構築手順書「4.3.7 Guest OS(仮想マシン)を作成」で作成したマシン名 `"RMimg_<version>"` を確認してください。（以下同様）

<OSname> : 管理サーバ Host OS 名（例 RedHat8.6）（以下同様）

- CLI コマンドの実行で使用する URL は、環境変数 `"URL"` に設定されている。
- Director 制御コマンドの実行は、CDI 管理ソフトウェアの仮想環境で行う。

1.1 運用操作

運用時の行う操作の概要を以下に示します。

○システムの起動・終了

- システムを構成するハードウェアのパワーオン・オフ、CDI 管理ソフトウェアの起動・停止を行います。

○CDI 管理ソフトウェアの再起動

- CDI 管理ソフトウェア（管理サーバの Guest OS）の停止・起動により、再起動を行います。

○CDI 管理ソフトウェアへのログイン

- CDI 管理ソフトウェアにアクセスするためにユーザ認証を行います。

○リソースの確認

- グループ内のリソースや、論理サーバに組み込まれたリソースなど各所のリソースを確認します

○グループの作成・解体

- リソース、論理サーバを要素に持つグループの作成と、使用しなくなったグループの解体を行います。

○グループへのリソース追加・グループからのリソースの削除

- グループへ新たなリソースの追加と、不要となったリソースの削除を行います。

○論理サーバの作成・解体

- 各種リソースで構成される論理サーバの作成と、使用しなくなった論理サーバの解体を行います。

○論理サーバの構成変更

- リソースの追加、削除により、論理サーバの構成を変更します。

○OS イメージの作成

- 論理サーバの NVMe SSD（リソース）にインストールする OS イメージの作成を行います。

○OS イメージのインストール

- NVMe SSD（リソース）に OS イメージをインストールします。

○データの削除

- NVMe SSD（リソース）のデータを削除します。

○グループ、論理サーバの再構成

- 構成バックアップファイルを使用して、グループと論理サーバの再構成を行います。

1.2 保守操作

保守時に行う操作の概要を以下に示します。

○ハードウェアの増設

- 新たなハードウェアの増設を行います。

○ハードウェアの減設

- 必要なくなった場合など、ハードウェアの減設を行います。

○ハードウェアの交換

- 故障した場合など、ハードウェアの交換を行います。

○ソフトウェアのアップデート

- CDI 管理ソフトウェア、Director ファームウェア、NIC ファームウェアのアップデートを行います。

1.3 システムの監視

システムの監視に関連した操作の概要を以下に示します。

○ハードウェアの状態監視

- システムを構成するハードウェアの監視を行います。監視は、iRMC (integrated Remote Management Controller)を使用して行われます。

※ iRMCの詳細は、iRMC 操作手引きを参照してください。

○ハードウェア (NVMe SSD) の寿命確認

- NVMe SSD (リソース) の交換時期を判断するために、書き込み寿命の確認を行います。

第2章 運用操作・保守操作・システムの監視で使用する機能

本章では、運用操作、保守操作、システムの監視で使用される CDI 管理ソフトウェアの機能（CLI コマンド、Director 制御コマンドの操作方法）について説明します。

2.1 グループ操作

ここでは、グループの操作に関連した機能を示します。

2.1.1 グループの情報を表示

作成されているグループの一覧および、その情報が表示されます。

```
cdi: ./epcctl -u $URL group list -o LIST
200
```

grp_uuid	grp_id	grp_name	grp_owner	resources
dcc23c4e-f9ee-4057-9588-254ec8ca8c19	1	group0	owner0	[{'res_uuid': '3ade12c8-bbd4-4306-8d5f-8c374dd82620', 'res_name': 'gpu0', 'res_type': 'gpu', 'res_status': 3, 'res_option_status': '0'}]


```
machines
[{'uuid': '3027d7e8-6bd6-422a-8130-a12c809be254', 'status': 10, 'option_status': '00'}]
```

※表示内容の詳細は、ソフトウェア管理者ガイドを参照してください。

2.1.2 グループの詳細情報を表示

作成したグループを指定することで詳細情報が表示されます。グループの指定には「2.1.1 グループの情報を表示」により表示される grp_name を使用します。

例) group0 の詳細情報を表示

```
cdi: ./epcctl -u $URL group show -g group0 -o
YAML
200
- fbr_id: 1
  grp_id: 1
  grp_id_nonliquid: null
  grp_name: group0
  grp_op_status: '000'
  grp_owner: Fujitsu
  grp_policy: null
  grp_status: 1
  grp_uuid: c50ec5e9-4b32-465b-8ff4-e0921a49b75a
  shelf_id: null
  node_id: null
  networks:
  - uuid: 3b9f159b-3a67-466a-9c8c-c2933293a962
    uuid_type: 5
  resources:
  - res_name: gpu0
    res_option_status: '0'
    res_status: 3
    res_type: gpu
    res_uuid: 3ade12c8-bbd4-4306-8d5f-8c374dd82620
  - res_name: ssd0
    res_option_status: '0'
    res_status: 3
    res_type: storage
    res_uuid: 4f6f4cac-1074-4cf0-ac6b-da37d17642f0
```

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.1.3 グループの作成 (コマンド指定)

指定したリソースで構成されるグループが、指定された名前で作成されます。指定できるリソースはファブリックフリープールの存在するものに限られ、指定には「2.3.2 ファブリックフリープールのリソースを表示」により表示される res_name を使用します。なお、指定されたリソースは本コマンドの実行後にグループフリープールに移動されます。

例) res_name が pcpu0,gpu0,ssd0,nic0 のリソースで構成される group0 を作成

```
cdi: ./epcctl -u $URL group create -g group0 -r pcpu0, gpu0, ssd0, nic0
200
{'grp_uuid': '3fa85f64-5717-4562-b3fc-2c963f66afa6'}
```

作成するグループの名前 ← res_name
※複数指定は`,`で区切ります

2.1.4 グループの作成 (ファイル指定)

設定ファイル (yaml 形式) に記載されたリソース、名前から成るグループが作成されます。設定ファイルに記載するリソースはファブリックフリープールに存在するものに限られます。なお、指定されたリソースは本コマンドの実行後にグループフリープールに移動されます。

例) res_name が pcpu0,gpu0,ssd0,nic0 のリソースで構成される group0 を作成

```
cdi: ./epcctl -u $URL group create -y group0_create.yaml
200
{'grp_uuid': '3fa85f64-5717-4562-b3fc-2c963f66afa6'}
```

```
group0_create.yaml
groups:
  grp_name: group0
  premachines: []
  resources:
  -
    res_specs:
    -
      res_type: 'compute'
      res_uuid: 'b9c59f7d-0a92-4af2-a28e-3778e5b922c1'
    -
      res_type: 'gpu'
      res_uuid: '2df31bfc-723e-4781-a83d-4372630c3498'
    -
      res_type: 'storage'
      res_uuid: '4894c94b-a598-4882-ae2c-dab4c36f5267'
    -
      res_type: 'network'
      res_uuid: '3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309'
  flavors: []
```

作成するグループの名前 ← リソース

※設定内容の詳細はソフトウェア管理者ガイドを参照してください。

2.1.5 グループへのリソース追加（コマンド指定）

指定したリソースが、グループフリープールに追加されます。指定できるリソースはファブリックフリープールに存在するものに限られ、リソース、グループの指定には「2.3.2 ファブリックフリープールのリソースを表示」「2.1.1 グループの情報を表示」により表示される res_name、grp_name を使用します。

例) res_name が gpu1,pcpu1 のリソースを group0 に追加

```
cdi: ./epcctl -u $URL group modify -M add -g group0 -r gpu1,pcpu1
200
{'grp_uuid': '3fa85f64-5717-4562-b3fc-2c963f66afa6'}
```

grp_name ←
res_name ←
※複数指定は`,`で区切ります

2.1.6 グループへのリソース追加（ファイル指定）

設定ファイル（yaml 形式）に記載されたリソースがグループフリープールに追加されます。設定ファイルに記載するリソースはファブリックフリープールに存在するものに限られます。

例) res_name が gpu1,pcpu1 のリソースを group0 に追加

```
cdi: ./epcctl -u $URL group modify -M add -y group0_modify.yaml
200
{'grp_uuid': '3fa85f64-5717-4562-b3fc-2c963f66afa6'}
```

```
group0_modify.yaml
groups:
  grp_uuid: 3fa85f64-5717-4562-b3fc-2c963f66afa6
  resources:
  -
    res_specs:
    -
      res_type: 'gpu'
      res_uuid: '5b30e05a-ef22-4349-b4fd-5cdb45029a2'
    -
      res_type: 'compute'
      res_uuid: '4c69968e-df77-4b37-928a-12bd2d401cc8'
```

grp_uuid ←
リソース ←

※設定内容の詳細はソフトウェア管理者ガイドを参照してください。

2.1.7 グループからのリソース削除（コマンド指定）

指定したリソースが、グループから削除されます。指定できるリソースはグループフリープールに存在するものに限られ（注）、リソース、グループの指定には「2.3.2 ファブリックフリープールのリソースを表示」「2.1.1 グループの情報を表示」により表示される res_name、grp_name を使用します。なお、本コマンドで削除されたリソースはファブリックフリープールに移動されます。

注) 論理サーバに組み込まれているリソースは削除できません。

例) res_name が gpu1,pcpu0 のリソースを group0 から削除

```
cdi: ./epcctl -u $URL group modify -M remove -g group0 -rgpu1,pcpu0
200
{'grp_uuid': '3fa85f64-5717-4562-b3fc-2c963f66afa6'}
```

grp_name
res_name
※複数指定は`,`で区切ります

2.1.8 グループからのリソース削除（ファイル指定）

設定ファイル（yaml 形式）に記載されたリソースがグループから削除されます。設定ファイルに記載できるリソースはグループフリープールに存在するものに限られます（注）。なお、本コマンドで削除されたリソースはファブリックフリープールに移動されます。

注) 論理サーバに組み込まれているリソースは削除できません。

例) res_name が gpu1,pcpu0 のリソースを group0 から削除

```
cdi: ./epcctl -u $URL group modify -M remove -y group0_modify.yaml
200
{'grp_uuid': '3fa85f64-5717-4562-b3fc-2c963f66afa6'}
```

```
group0_modify.yaml
groups:
  grp_uuid: '3fa85f64-5717-4562-b3fc-2c963f66afa6'
  resources:
  -
    res_specs:
    -
      res_type: 'gpu'
      res_uuid: '5b30e05a-ef22-4349-b4fd-5cdb45029a2'
    -
      res_type: 'compute'
      res_uuid: 'b9c59f7d-0a92-4af2-a28e-3778e5b922c1'
```

grp_uuid
リソース

※設定内容の詳細はソフトウェア管理者ガイドを参照してください。

2.1.9 グループの解体

指定されたグループが解体されます。グループの指定には「2.1.1 グループの情報を表示」により表示される grp_name を使用します。なお、グループ内のリソースは本コマンドの実行後にファブリックフリープールに移動されます。

注) 論理サーバが存在した状態で解体はできません。全ての論理サーバを解体した後にグループを解体してください。

例) group0 を解体

```
cdi: ./epcctl -u $URL group destroy -g group0
200
{'grp_uuid': '3fa85f64-5717-4562-b3fc-2c963f66afa6'}
```

grp_name

2.2 論理サーバ操作

ここでは、論理サーバの操作に関連した機能を示します。

2.2.1 論理サーバの一覧を表示

指定したグループの論理サーバが表示されます。グループの指定には「2.1.1 グループの情報を表示」により表示される grp_name を使用します。

例) group0 の論理サーバを表示

```
cdi: ./epcctl -u $URL machine list -g group0 -o LIST
200
```

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0
machine1	2	7d6210ef-8a6a-403e-a980-5542b7852c30	user1

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.2.2 論理サーバの詳細を表示

指定した論理サーバの詳細情報が表示されます。論理サーバの指定には「2.2.1 論理サーバの一覧を表示」により表示される mach_name を使用します。

例) machine0 の詳細情報を表示

```
cdi: ./epcctl -u $URL machine show -m machine0 -o LIST
200
```

item	value
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26
mach_id_nonliquid	
mach_id	
mach_name	machine0
mach_status	13
mach_op_status	00
mach_status_detail	ACTIVE PON
mach_owner	Fujitsu
grp_uuid	7cd208b6-8987-495f-b9bd-5f257ae850b7
p2p	off
boot_ssd	
lanports	
resources	[{'res_uuid': 'f59b7fc1-ad50-494b-9f45-2ff027c6cf55', 'res_name': 'pcpu3', 'res_type': 'compute', 'res_status': 4, 'res_op_status': '0'}]

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.2.3 論理サーバの作成 (コマンド指定)

指定したリソースで構成される論理サーバが、指定された名前グループ内に作成されます。指定できるリソースはグループフリープールに存在するものに限られ、リソース、グループの指定には「2.3.2 ファブリックフリープールのリソースを表示」「2.1.1 グループの情報を表示」により表示される res_name、grp_name を使用します。なお、作成する論理サーバの名前はグループ内で一意になるように指定します。

例) res_name が pcpu0,gpu0,ssd0 のリソースで構成される machine0 を group0 に作成

```
cdi: ./epcctl -u $URL machine create -g group0 -m machine0 -r pcpu0, gpu0, ssd0
200
{'data': {'machines': [{'mach_uuid': ' b9e25b57-521a-4047-b2cd-6159c452ac26' } ]}}
```

grp_name 作成する論理サーバの名前 (グループ内で一意)

res_name

※複数指定は','で区切ります

2.2.4 論理サーバの作成 (ファイル指定)

設定ファイル (yaml 形式) に記載されたリソース、名前から成る論理サーバがグループ内に作成されます。設定ファイルに記載するリソースはグループフリープールに存在するものに限られます。なお、作成する論理サーバの名前はグループ内で一意になるように記載します。

例) res_name が pcpu0,gpu0,ssd0 のリソースで構成される machine0 を group0 に作成

```
cdi: ./epcctl -u $URL machine create -y machine0_create.yaml
200
{'data': {'machines': [{'mach_uuid': ' b9e25b57-521a-4047-b2cd-6159c452ac26' } ]}}
```

```
machine0_create.yaml
groups:
  grp_name: "group0"
  machines:
  - mach_name: machine0
    resources:
    -
      res_specs:
      -
        res_type: 'compute'
        res_name: 'pcpu0'
      -
        res_type: 'gpu'
        res_name: 'gpu0'
      -
        res_type: 'storage'
        res_name: 'ssd0'
```

grp_name

作成する論理サーバの名前 (グループ内で一意)

リソース

※設定内容の詳細はソフトウェア管理者ガイドを参照してください。

2.2.5 論理サーバへのリソース追加（コマンド指定）

指定したリソースが、指定された論理サーバに追加されます。指定できるリソースはグループフリープールに存在するものに限られ、リソース、論理サーバの指定には「2.3.3 グループフリープールのリソースを表示」「2.2.1 論理サーバの一覧を表示」により表示される res_name、mach_name を使用します。

例) res_name が gpu1,pcpu0 のリソースを machine0 に追加

```
cdi: ./epcctl -u $URL machine modify -M add -m machine0 -r gpu1,pcpu0
200
{'data': {'machines': [{'mach_uuid': 'b9e25b57-521a-4047-b2cd-6159c452ac26'}]}}
```

mach_name ←
res_name ←
※複数指定は`,`で区切ります

2.2.6 論理サーバへのリソース追加（ファイル指定）

設定ファイル（yaml 形式）に記載されたリソースが、記載された論理サーバに追加されます。設定ファイルに記載するリソースはグループフリープールに存在するものに限られます。

例) res_name が gpu1,pcpu0 のリソースを machine0 に追加

```
cdi: ./epcctl -u $URL machine modify -M add -y machine0_modify.yaml
200
{'data': {'machines': [{'mach_uuid': 'b9e25b57-521a-4047-b2cd-6159c452ac26'}]}}
```

```
machine0_modify.yaml
machines:
  mach_name: "machine0"
  resources:
  -
    res_specs:
    -
      res_type: 'gpu'
      res_name: 'gpu1'
    -
      res_type: 'compute'
      res_name: 'pcpu0'
```

mach_name ←
リソース ←

※設定内容の詳細はソフトウェア管理者ガイドを参照してください。

2.2.7 論理サーバからのリソース削除（コマンド指定）

指定したリソースが指定された論理サーバから削除されます。リソース、論理サーバの指定には「2.3.5 論理サーバで使用されているリソースを表示」「2.2.1 論理サーバの一覧を表示」により表示される res_name、mach_name を使用します。なお、本コマンドで削除されたリソースはグループフリープールに移動されます。
 注) res_type が storage のリソースを削除した場合は、リソース(NVMe SSD)のデータは自動的に消去されます。

例) res_name が gpu1,pcpu0 のリソースを machine0 から削除

```
cdi: ./epcctl -u $URL machine modify -M remove -m machine0 -r gpu1,pcpu0
200
{'data': {'machines': [{'mach_uuid': 'b9e25b57-521a-4047-b2cd-6159c452ac26'}]}}
```

mach_name
res_name
※複数指定は`,`で区切ります

2.2.8 論理サーバからのリソース削除（ファイル指定）

設定ファイル（yaml 形式）に記載されたリソースが論理サーバから削除されます。なお、本コマンドで削除されたリソースはグループフリープールに移動されます。
 注) res_type が storage のリソースを削除した場合は、リソース(NVMe SSD)のデータは自動的に消去されます。

例) res_name が gpu1,pcpu0 のリソースを machine0 から削除

```
cdi: ./epcctl -u $URL machine modify -M remove -y machine0_modify.yaml
200
{'data': {'machines': [{'mach_uuid': 'b9e25b57-521a-4047-b2cd-6159c452ac26'}]}}
```

```
machine0_modify.yaml
machines:
  mach_name: machine0
  resources:
  -
    res_specs:
      res_type: 'compute'
      res_name: 'pcpu0'
  -
    res_type: 'gpu'
    res_name: 'gpu1'
```

mach_name
リソース

※設定内容の詳細はソフトウェア管理者ガイドを参照してください。

2.2.9 論理サーバの解体

指定された論理サーバが解体されます。論理サーバの指定には「2.2.1 論理サーバの一覧を表示」により表示される mach_name を使用します。なお、解体された論理サーバのリソースは本コマンドの実行後にグループフリープールに移動されます。
 注) 論理サーバに組み込まれていた res_type が storage のリソース(NVMe SSD)のデータは自動的に消去されます。

例) machine0 を解体

```
cdi: ./epcctl -u $URL machine destroy -m machine0
200
{'data': {'machines': [{'mach_uuid': 'b9e25b57-521a-4047-b2cd-6159c452ac26'}]}}
```

mach_name

2.2.10 論理サーバの電源オン

指定した論理サーバの電源がオンされます。論理サーバの指定には「2.2.1 論理サーバの一覧を表示」により表示される mach_name を使用します。

例) machine0 の電源をオン

```
cdi: ./epcctl -u $URL machine power -P on -m machine0
200
{'data': {'machines': [{'mach_uuid': ' b9e25b57-521a-4047-b2cd-6159c452ac26' }]]}
```

なお、論理サーバの PCIe バスの Peer to Peer(P2P)機能を有効にする場合は「2.2.14 論理サーバの PCIe バスの P2P 機能の有効化/無効化」を参照してください。

2.2.11 論理サーバの電源オフ

指定した論理サーバの電源がオフされます。論理サーバの指定には「2.2.1 論理サーバの一覧を表示」により表示される mach_name を使用します。

例) machine0 の電源をオフ

```
cdi: ./epcctl -u $URL machine power -P off -m machine0
200
{'data': {'machines': [{'mach_uuid': ' b9e25b57-521a-4047-b2cd-6159c452ac26' }]]}
```

2.2.12 論理サーバの状態確認

指定した論理サーバのステータスが表示されます。論理サーバの指定には「2.2.1 論理サーバの一覧を表示」により表示される mach_name を使用します。

例) machine0 のステータスを表示

```
cdi: ./epcctl -u $URL machine show -m machine0 -o LIST
200
```

item	value
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26
mach_id_nonliquid	
mach_id	
mach_name	machine0
mach_status	13
mach_op_status	00
mach_status_detail	ACTIVE PON
mach_owner	Fujitsu
grp_uuid	7cd208b6-8987-495f-b9bd-5f257ae850b7
p2p	off
boot_ssd	
lanports	
resources	[{'res_uuid': ' f59b7fc1-ad50-494b-9f45-2ff027c6cf55', 'res_name': ' pcpu3', 'res_type': ' compute', 'res_status': 4, 'res_op_status': '0'}]

※mach_status_detail が論理サーバの詳細ステータスを示しています。詳細はソフトウェア管理者ガイドを参照してください。

2.2.13 論理サーバの PCIe バスの Peer to Peer(P2P)機能の設定確認

指定した論理サーバの P2P の設定状態を確認します。デフォルトは無効状態です。

例) machine0 のステータスを表示

```
cdi: ./epcctl -u $URL machine show -m machine0 -o LIST
200
```

item	value
p2p	off

p2p:on は P2P 設定有効状態、off は無効状態を表します。

2.2.14 論理サーバの PCIe バスの P2P 機能の有効化/無効化

指定した論理サーバの P2P 機能を有効または、無効にします。有効にする場合は論理サーバの電源オン後に実行する必要があります。電源オン直後はエラー終了することがあるため、"machine is powered off."または "Failed to Enable/disable P2P for a machine."でエラー終了した場合には一定時間おいてから再度実行してください。

論理サーバの電源オフにより P2P 設定は自動的に無効になります。再度有効化する場合は、電源オン後に有効化コマンドを実行してください。

例) machine0 の P2P 設定を有効化

```
cdi: ./epcctl -u $URL machine p2p -m machine0 -P on
200
{'mach_uuid': ' b9e25b57-521a-4047-b2cd-6159c452ac26' }
```

2.2.15 NVIDIA GPU H100 を払い出す場合の注意事項

運用管理者が、8 slot PCIe Box に搭載された NVIDIA GPU H100 を使用する論理サーバを払い出す場合、論理サーバの使用者に対して以下の通知を行って下さい。

NVIDIA GPU H100 を利用する場合、nvidia-smi コマンドを実行して、最大消費電力を 300W に設定して使用して下さい。350W で GPU H100 を利用すると正常に動作しない可能性があります。

備考：上記制限は、8 slot PCIe Box に NVIDIA GPU H100 を搭載した場合のみの制限です。他の GPU を払い出す場合、及び 10 slot PCIe Box に NVIDIA GPU H100 を搭載している場合は上記制限は必要ありません。

以下に nvidia-smi コマンドの実行手順を示します。コマンドの実行は、使用者が論理サーバの OS にログインして実行して下さい。なお一部のコマンド実行では su 権限が必要になります。

(1) 以下のコマンドを実行して使用する GPU のタイプと現在の最大電力の設定値を確認して下さい。

```
[epcadmin@localhost ~]$ nvidia-smi
Thu Nov 30 17:11:39 2023
```

NVIDIA-SMI 525.60.13			Driver Version: 525.60.13		CUDA Version: 12.0	
GPU	Name	Persistence-M	Bus-Id	Disp. A	Volatile Uncorr.	ECC
Fan	Temp	Perf		Memory-Usage	GPU-Util	Compute M.
		Pwr:Usage/Cap				MIG M.
0	NVIDIA H100	PCIe Off	00000000:AA:00:0	Off	8%	0
N/A	32C	P0 69W / 350W	0MiB / 81559MiB			Default Disabled

- ※ 1 : NVIDIA H100 であることを確認して下さい (H100 でない場合は、以降のコマンド実行は不要です)。
- ※ 2 : Persistence Mode が Off であることを確認して下さい。
- ※ 3 : 350W であることを確認して下さい (300W 以下である場合は、以降のコマンド実行は不要です)。

(2) 以下のコマンドを実行してパーシステンスモードをオンにして、GPU の最大電力を 300W に設定します。
(このコマンドの実行は su 権限が必要です)

```
[epcadmin@localhost ~]$ sudo nvidia-smi -pm 1
Enabled persistence mode for GPU 00000000:AA:00:0.
All done.
[epcadmin@localhost ~]$ sudo nvidia-smi -pl 300
Power limit for GPU 00000000:AA:00:0 was set to 300.00 W from 350.00 W.
All done.
```

※ : sudo コマンド実行時には su password の入力を求められます。

(3) 以下のコマンドを実行して再度、パーシステンスモードと最大電力の設定値を確認して下さい。

```
[epcadmin@localhost ~]$ nvidia-smi
Thu Nov 30 17:11:39 2023
```

NVIDIA-SMI 525.60.13			Driver Version: 525.60.13		CUDA Version: 12.0	
GPU	Name	Persistence-M	Bus-Id	Disp. A	Volatile Uncorr.	ECC
Fan	Temp	Perf		Memory-Usage	GPU-Util	Compute M.
		Pwr:Usage/Cap				MIG M.
0	NVIDIA H100	PCIe On	00000000:AA:00:0	Off	8%	0
N/A	32C	P0 69W / 300W	0MiB / 81559MiB			Default Disabled

- ※ 1 : Persistence Mode が On であることを確認して下さい。
- ※ 2 : 300W であることを確認して下さい。

(4) Persistence モード、及び最大電力設定は論理サーバの OS 再起動を行うと、元の設定に戻ります。この場合、(1)~(3)を再度実行するか、または、予め libtirpc-devel 等のツールを用いてデバイスドライバの初期設定を変更して下さい。

2.3 リソース操作

ここでは、リソースの操作に関連した機能を示します。

2.3.1 リソースの一覧を表示

システム内の全てのリソースが表示されます。

```
cdi: ./epcctl -u $URL resource list -o LIST
200
```

res_type	res_name	res_uuid	res_status	res_op_status	res_owner	rack_name
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1	4	0	owner1	rack0
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8	3	0	owner1	rack0
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498	n/a	0	n/a	None
gpu	gpu1	5b30e05a-ef22-4349-b4fd-5cdb45029a2	n/a	0	n/a	None
gpu	gpu2	5b30e05a-ef22-4349-b4fd-5cdb45029a2	n/a	0	n/a	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309	n/a	0	n/a	None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267	3	0	n/a	None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490	3	0	n/a	None

position_in_rack	slot_number
5U	8
5U	8
None	None
None	None
None	None
None	None
None	None

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.3.2 ファブリックフリープールのリソースを表示

ファブリックフリープール内のリソース（グループに登録されていないリソース）が表示されます。

```
cdi: ./epcctl -u $URL resource list -F -o LIST
200
```

res_type	res_name	res_uuid	res_status	res_op_status	res_owner	rack_name
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8	3	0	owner1	rack0
gpu	gpu2	5b30e05a-ef22-4349-b4fd-5cdb45029a2	n/a	0	n/a	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309	n/a	0	n/a	None

position_in_rack	slot_number
5U	8
None	None
None	None

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.3.3 グループフリープールのリソースを表示

グループフリープール内のリソース（論理サーバに使用されていないリソース）が表示されます。グループの指定には「2.1.1 グループの情報を表示」により表示される grp_name を使用します。

例) group0 のフリープールのリソースを表示

```
cdi: ./epcctl -u $URL resource list -g group0 -G -o LIST
200
```

res_type	res_name	res_uuid	res_status	res_op_status	res_owner	rack_name
gpu	gpu1	5b30e05a-ef22-4349-b4fd-5cdbc45029a2	n/a	0	n/a	None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490	3	0	n/a	None

position_in_rack	slot_number
None	None
None	None

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.3.4 グループに登録されているリソースを表示

グループ内のリソースの情報が表示されます。グループの指定には「2.1.1 グループの情報を表示」により表示される grp_name を使用します。

例) group0 のリソースを表示

```
cdi: ./epcctl -u $URL resource list -g group0 -o LIST
200
```

res_type	res_name	res_uuid	res_status	res_op_status	res_owner	rack_name
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1	4	0	owner1	rack0
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498	n/a	0	n/a	None
gpu	gpu1	5b30e05a-ef22-4349-b4fd-5cdbc45029a2	n/a	0	n/a	None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267	3	0	n/a	None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490	3	0	n/a	None

position_in_rack	slot_number
5U	8
None	None
None	None
None	None
None	None

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.3.5 論理サーバで使用されているリソースを表示

指定した論理サーバで使用されているリソースの情報が表示されます。論理サーバの指定には「2.2.1 論理サーバの一覧を表示」により表示される mach_name を使用します。

例) machine0 で使用されているリソースを表示

```
cdi: ./epcctl -u $URL resource list -m machine0 -o LIST
200
```

res_type	res_name	res_uuid	res_status	res_op_status	res_owner	rack_name
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1	4	0	owner1	rack0
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498	n/a	0	n/a	None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267	3	0	n/a	None

position_in_rack	slot_number
5U	8
None	None
None	None

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.3.6 リソースの詳細を表示

指定したリソースの詳細情報が表示されます。リソースの指定には「2.3.1 リソースの一覧を表示」により表示される res_name を使用します。

例) gpu0 の詳細情報を表示

```
cdi: ./epcctl -u $URL resource show -r gpu0 -o LIST
200
```

item	value
res_uuid	2df31bfc-723e-4781-a83d-4372630c3498
res_id_nonliquid	
fabr_gid	0x033c00
res_type	gpu
res_name	gpu0
res_status	4
res_op_status	0
res_owner	n/a
grp_uuid	dcc23c4e-f9ee-4057-9588-254ec8ca8c19
premach_uuid	
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26
booting_uuid	
lanports	[]
resspec_uuid	648f5a06-336f-46d0-8511-f9f1a9f657f9
resspec_productname	Tesla T4
resspec_model	Tesla_T4
resspec_vendor	NVIDIA Corporation
resspec_removable	False

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.3.7 グループフリープールのエラー状態のリソースを表示

エラー状態となっているグループフリープールのリソースの情報が表示されます。グループの指定には「2.1.1 グループの情報を表示」により表示される grp_name を使用します。

※エラー状態のリソースとは res_op_status が 0 以外のリソースです。

```
cdi: ./epcctl -u $URL resource list -E -g group0 -G -o YAML
200
- res_name: gpu0
  res_op_status: '2'
  res_owner: n/a
  res_status: 1
  res_type: gpu
  res_uuid: 2df31bfc-723e-4781-a83d-4372630c3498
  rack_name: 'rack0'
  position_in_rack: '10U'
  slot_number: 0
```

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.3.8 論理サーバで使用されているエラー状態のリソースを表示

論理サーバで使用されており、エラー状態となっている全てのリソースの情報が表示されます。

※エラー状態のリソースとは res_op_status が 0 以外のリソースです。

```
cdi: ./epcctl -u $URL resource list -E -M -o YAML
200
- res_name: gpu0
  res_op_status: '2'
  res_owner: n/a
  res_status: 1
  res_type: gpu
  res_uuid: 2df31bfc-723e-4781-a83d-4372630c3498
  rack_name: 'rack0'
  position_in_rack: '10U'
  slot_number: 0
```

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.3.9 リソース ステータスの設定

指定したリソースのオプションステータスに、指定した値が設定されます。リソースの指定には「2.3.1 リソースの一覧を表示」により表示される res_name を使用します。

例) res_name が ssd0 のリソースのオプションステータスに 2:critical を設定

```
cdi: ./epcctl -u $URL resource set_op_status -r ssd0 -s 2
success
```

※0:OK 1:Warning 2:Critical

2.3.10 リソース情報の同期

以下の手順でリソース情報を同期させます。

(1) リソース情報を同期します。

```
cdi: ./epcctl -u $URL resource sync
200
success
```

注) 本コマンドの実行により作成済みのグループや論理サーバの構成情報がクリアされます。CDI 管理ソフトウェアのセットアップ時と保守操作以外での実行は控えてください。

(2) 「3.20.4 CDI 管理ソフトウェアからログアウト」を実行します。

(3) 管理サーバの Guest OS(仮想マシン)にログインします。

※ログイン情報については、システム構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照

(4) 以下のコマンドでグループ・論理サーバの構成ファイルを初期化します。

```
$ cd /var/opt/epc/resource_manager/dump/
$ rm *.yaml script.sh
```

(5) Guest OS(仮想マシン)からログアウトします。

```
$ exit
```


2.4 OS イメージ操作

ここでは、論理サーバに組み込む OS イメージの操作に関連した機能を示します。

2.4.1 OS イメージの一覧を表示

システムに登録された全ての OS イメージが表示されます。

```
cdi: ./epcctl -u $URL image list -o LIST
200
```

bootimg_uuid	bootimg_owner	bootimg_filename	bootimg_type
0627d38b-5d5e-49d4-8439-0a2d4810d55c	n/a	ubuntu18-rx1-20221004.img	img
12e8c1fa-6dd4-4359-b03e-58cd6cf7acc2	n/a	ubuntu18-rx1-20221001.img	img
43fb0b01-5e1d-413e-80a4-237f4eb3a809	n/a	CentOS-7-x86_64-DVD-2009.iso	iso
6db8095e-9ef9-4f0d-aa6c-9f8f7883a8e9	n/a	ubuntu18-rx1-20221003.img	img
a4ff88e4-6162-4e89-ae0a-d79f5008b13e	n/a	ubuntu-20.04.4-live-server-amd64.iso	iso
accf6ed9-d3c7-4c43-af7c-8324203135de	n/a	ubuntu18-rx1-20220615.img	img
fec12e74-421b-48f3-bf34-3075205a5444	n/a	ubuntu20-MLPerf1.img	img
d781ac42-c7ff-424b-982b-739bb6cdf05b	n/a	ubuntu-18.04.6-live-server-amd64.iso	iso

※表示内容の詳細は、ソフトウェア管理者ガイドを参照してください。

2.4.2 OS イメージの詳細を表示

指定した OS イメージの詳細情報が表示されます。OS イメージの指定には「2.4.1 OS イメージの一覧を表示」により表示される bootimg_filename を使用します。

例) ubuntu18-rx1-20221004.img の詳細情報を表示

```
cdi: ./epcctl -u $URL image show <i ubuntu18-rx1-20221004.img > -o LIST
200
```

bootimg_uuid	bootimg_id	bootimg_date	bootimg_owner
0627d38b-5d5e-49d4-8439-0a2d4810d55c	9	2022-12-19T12:52:12+00:00	n/a

bootimg_filename	bootimg_type
ubuntu18-rx1-20221004.img	img

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.4.3 OS イメージの登録

指定した OS イメージがシステムに登録されます。OS イメージは ISO イメージファイル(.ISO 形式のファイル)から作成されます。予め「3.16.1 イメージ作成用論理サーバの作成」～「3.16.8 論理サーバの解体」の手順に沿って OS イメージを準備してください。なお、登録できる OS イメージの大きさは 16GB 以下になります。また、同時に同じ名前の OS イメージを登録することはできません。時間をおいて再登録してください。

例) ubuntu20-MLPerf1.img をシステムに追加

```
cdi: ./epcctl -u $URL image add -t img -i ubuntu20-MLPerf1.img
200
{
  'bootimg_uuid': 'fec12e74-421b-48f3-bf34-3075205a5444'
}
```

登録する OS イメージの種類別
登録する OS イメージのファイル名

※OS イメージの種類別はソフトウェア管理者ガイドを参照してください。

2.4.4 OS イメージの削除

指定した OS イメージがシステムから削除されます。OS イメージの指定には「2.4.1 OS イメージの一覧を表示」により表示される bootimg_filename を使用します。

例) ubuntu20-MLPerf1.img をシステムから削除

```
cdi: ./epcctl -u $URL image remove -i ubuntu20-MLPerf1.img
200
{
  'bootimg_uuid': 'fec12e74-421b-48f3-bf34-3075205a5444'
}
```

bootimg_filename

2.4.5 OS イメージのインストール

指定した OS イメージを「2.3.5 論理サーバで使用されているリソースを表示」により表示される論理サーバに組み込まれた res_type が storage のリソース (NVMe SSD) にインストールします。OS イメージの指定には「2.4.1 OS イメージの一覧を表示」により表示される bootimg_filename を使用します。

例) bootimage.img を ssd0 にインストール

```
cdi: ./epcctl -u $URL resource install -r ssd0 -b bootimage.img
200
success
```

res_name
bootimg_filename

2.5 構成バックアップファイル操作

ここでは、構成バックアップファイルの操作に関連した機能を示します。構成バックアップファイルとは、システム運用管理者がグループや論理サーバの構成を変更するたびに CDI 管理ソフトウェア内に自動的に保存される圧縮ファイルです。このファイルは保守作業時など構成情報を復元する必要がある場合に使用します。

2.5.1 構成バックアップファイルの一覧を表示

作成された構成バックアップファイルが表示されます。

- ファイル名の一部がバックアップを行った時のタイムスタンプになっています。
- ファイルは、グループ・論理サーバが作成・更新・削除されるたびに自動的に生成され、CDI 管理ソフトウェア内に保存されます。
- 最大 10 個のファイルを保存し、10 個を超えた場合はタイムスタンプの古いものから削除されます。

```
cdi: ./epcctl -u $URL system dump -L
200
{
  'data': {
    'dumpfiles': [
      'dumpfile_20230516013915431141.tar.gz',
      'dumpfile_20230516014016816701.tar.gz',
      'dumpfile_20230516014215763132.tar.gz',
      'dumpfile_20230516014352720493.tar.gz',
      'dumpfile_20230516014503466356.tar.gz',
      'dumpfile_20230516014631463656.tar.gz',
      'dumpfile_20230516020212656352.tar.gz',
      'dumpfile_20230516020345644777.tar.gz',
      'dumpfile_20230516020723381957.tar.gz',
      'dumpfile_20230516020845811055.tar.gz'
    ]
  }
}
```

タイムスタンプ

※表示内容の詳細はソフトウェア管理者ガイドを参照してください。

2.5.2 構成バックアップファイルをダウンロード

CDI 管理ソフトウェアに保存されている構成バックアップファイルから、指定した構成バックアップファイルを管理サーバの Host OS の現在のディレクトリ（本コマンドを実行したディレクトリ）にダウンロードします。なお、構成バックアップファイルの指定には「2.5.1 構成バックアップファイルの一覧を表示」により表示されるファイル名を使用します。

```
cdi: ./epcctl -u $URL system dump -F dumpfile_20230516020845811055.tar.gz
200
dumpfile_20230516020845811055.tar.gz
```

ダウンロードするバックアップファイル名

2.6 その他の操作

ここでは、「2.1 グループ操作」～「2.5 構成バックアップファイル操作」に示した操作以外の機能を示します。

2.6.1 データの削除

論理サーバに組み込まれた res_type が storage のリソース (NVMe SSD) のデータを消去します。リソースの指定には「2.3.5 論理サーバで使用されているリソースを表示」により表示される res_name を使用します。

例) ssd0 のデータを削除

```
cdi: ./epcctl -u $URL resource erase -r ssd0 ← res_name
200
Success
```

2.6.2 iRMC 情報の表示 (計算サーバ)

論理サーバに組み込まれた res_type が compute のリソース (計算サーバ) の iRMC 情報を表示します。リソースの指定には「2.3.5 論理サーバで使用されているリソースを表示」により表示される res_name を使用します。

例) res_name が pcpu0 のリソースの iRMC 情報を表示

```
cdi: ./epcctl -u $URL resource bmc -r pcpu0 -o LIST ← res_name
200
```

res_uuid_cpunode	cpunode_ipmi_address	cpunode_ipmi_userid	cpunode_ipmi_password
aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee	AAA.BBB.C.DDD	xxxxxxx	*****

2.6.3 ログファイルの作成

指定された名前でもログファイルの作成を行います。

```
cdi: ./epcctl -u $URL system log -o /home/<Host OS のユーザ名>/cdi/log/20230407.tar.gz ← ログファイル名
200
/home/20230407.tar.gz
```

※ログファイルの詳細はソフトウェア管理者ガイドを参照してください。

2.6.4 Director のリセット

Director をリセットします。

```
$ /opt/epc/bin/fabrctl reset -u https://Director の IP アドレス -c $LIQID_VERIFY_CA/ Director の IP
アドレス.pem
Success
```

※Director の IP アドレスはシステム構築手順書を参照してください。

※本操作は、管理サーバの Guest OS(仮想マシン)にログインして行います。ログイン情報については、構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照してください。

2.6.5 Director の状態表示

Director の状態を表示します。

```
$ /opt/epc/bin/fabrctl state -u https://Director の IP アドレス -c $LIQID_VERIFY_CA/ Director の IP
アドレス.pem
Success
```

※Director の IP アドレスはシステム構築手順書を参照してください。

※本操作は、管理サーバの Guest OS(仮想マシン)にログインして行います。ログイン情報については、構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照してください。

2.6.6 ファームウェアファイル (Director 用) のアップロード

指定したファームウェアファイルを Director へアップロードします。

例) ファームウェアファイル director.upr を Director へアップロード

Director 用ファームウェアのファイル名

```
$ /opt/epc/bin/fabrctl upload -u https://Director の IP アドレス -f director.upr -c $LIQID_VERIFY_CA/
Director の IP アドレス.pem
Success
```

※Director の IP アドレスはシステム構築手順書を参照してください。

※本操作は、管理サーバの Guest OS(仮想マシン)にログインして行います。ログイン情報については、構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照してください。

2.6.7 ファームウェアファイル (Director 用) の一覧を表示

Director にアップロードされたファームウェアファイルを表示します。

```
$ /opt/epc/bin/fabrctl upload -u https://Director の IP アドレス -c $LIQID_VERIFY_CA/Director の IP
アドレス.pem
Index  Filename
-----
0      director-2.5.1.36-2.upr
1      director.upr
```

※Director の IP アドレスはシステム構築手順書を参照してください。

※本操作は、管理サーバの Guest OS(仮想マシン)にログインして行います。ログイン情報については、構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照してください。

2.6.8 Director ファームウェアのアップデート

Director にアップロードされたファームウェアファイルを使用して、Director のファームウェアをアップデートします。

例) Index が 1 のファームウェアファイル (director.upr) を使用して、Director のファームウェアをアップデート

2.6.7 で表示されたアップデートに使用するファームウェアの Index

```
$ /opt/epc/bin/fabrcctl upgrade -u https://Director の IP アドレス L -i 1 -c $LIQID_VERIFY_CA/ Director  
の IP アドレス.pem
```

※Director の IP アドレスはシステム構築手順書を参照してください。

※本操作は、管理サーバの Guest OS(仮想マシン)にログインして行います。ログイン情報については、構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照してください。

※メジャーアップデート (3.2 から 3.3 へのアップデート等) の場合は、-w オプション (ソフトウェア管理者ガイド参照) を指定してください。

第3章 運用操作

本章では、以下の運用時に行われる操作について説明します。詳細については該当する章を参照してください。

○システムの起動・終了	(3.1, 3.2)
○CDI 管理ソフトウェアの再起動	(3.3)
○CDI 管理ソフトウェアへのログイン	(3.4)
○リソースの確認	(3.5)
○グループの作成・解体	(3.6, 3.15)
○グループへのリソース追加・グループからリソース削除	(3.7, 3.8)
○論理サーバの作成・解体	(3.9, 3.14)
○論理サーバの構成変更	(3.10, 3.11, 3.12, 3.13)
○OS イメージの作成	(3.16)
○OS イメージのインストール	(3.17)
○論理サーバへのアクセス	(3.18)
○データの削除	(3.19)
○グループ・論理サーバの再構成	(3.20)

3.1 システムを起動する

システムを起動するには、以下に示す 3.1.1～3.1.8 の手順を実施してください。

3.1.1 管理 LAN Switch の電源を投入

管理 LAN Switch の電源が投入されていない時は電源を投入します。

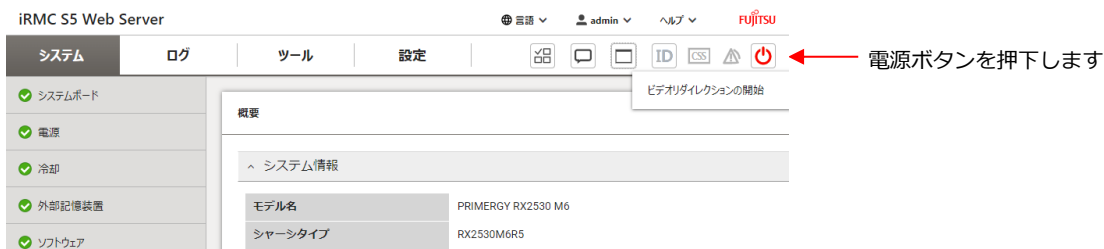
3.1.2 管理サーバの電源を投入

管理サーバの電源が投入されていない場合はサーバ本体の電源ボタンを使用するか、下記(1)～(2)で示す iRMC 画面経由で電源を投入します。

- (1) システム構築手順書「G.1 iRMC へのログイン」の操作により、システム管理者 PC (※1) で管理サーバの iRMC にログインします。

※1 各種作業でシステム管理者が使用する windows PC (以下同様)

- (2) iRMC の Web 画面より電源を投入します。



3.1.3 管理サーバのネットワーク設定を確認

管理サーバ Host OS の起動後にネットワークのブリッジ接続が維持されていることを下記のコマンドで確認します。

```

$ nmcli con
#接続されている場合の例
NAME                UUID                                TYPE    DEVICE
br0                 fa8d24d6-cd1f-4f96-8ba4-9f36a05b95c9 bridge  br0
br0-slave-eno1     e3e090ef-cfcf-4924-a964-df2f48ffb50e ethernet eno1 ← 管理サーバのネットワークデバイスが表示される
...

#接続されていない場合の例
NAME                UUID                                TYPE    DEVICE
br0                 fa8d24d6-cd1f-4f96-8ba4-9f36a05b95c9 bridge  br0
...
    
```


接続されていない場合、下記のコマンドを実行し接続します。

```

$ ip addr
# 管理サーバ Host OS の IP アドレスが指定されたネットワークデバイスを検索
2: eno1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
   link/ether xx:xx:xx:xx:xx:xx brd ff:ff:ff:ff:ff:ff
   inet x.x.x.x/x brd x.x.x.255 scope global dynamic noprefixroute eno1 ← 管理サーバのネットワークデバイス名
   . . .

$ sudo nmcli con add type bridge-slave autoconnect yes con-name br0-slave-{管理サーバのネットワークデバイス名} ifname {管理サーバのネットワークデバイス名} master br0
$ sudo nmcli con down br0
$ sudo nmcli con up br0

$ nmcli con ← 再度確認
#接続されている場合の例
NAME                UUID                                TYPE    DEVICE
br0                  fa8d24d6-cd1f-4f96-8ba4-9f36a05b95c9  bridge br0
br0-slave-eno1      4f0360f4-ae8a-4a11-b740-486a2bb868c6 ethernet eno1 ← 管理サーバのネットワークデバイス名を確認
    
```

3.1.4 CDI コンポーネントの電源を投入

CDI コンポーネントの電源を投入するためにシステム構築手順書「4.9.1 電源投入手順 (2)~(4)」を行います。

3.1.5 Director の FAN の状態確認

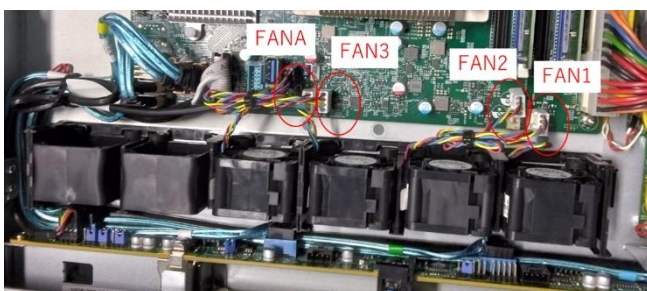
システムの電源を投入後は、管理サーバから Director に root ログインし、以下のコマンドにてファンの状態を確認してください。

```

# ipmitool sdr |grep FAN
FAN1 | 9900 RPM | ok
FAN2 | 9300 RPM | ok
FAN3 | 9900 RPM | ok
FAN4 | no reading | ns
FANA | 9500 RPM | ok
    
```

} ファンが正常な場合の表示例
注)FAN4 は、常に no reading,ns が表示されます。

- FAN1,FAN2,FAN3,FANA に、"ok"と表示されていれば問題ありません。
- FAN1,FAN2,FAN3,FANA に、"ok"以外が表示されていればファン異常です。ファン電源のケーブル抜け等を確認してください。解消しない場合はファン交換を実施してください。
- Director 起動後、稼働中に発生したファン故障は、Information LED にて確認できます。（「5.2.3.1 Director 目視による異常検出」を参照）



3.1.6 CDI 管理ソフトウェア（管理サーバの Guest OS）を起動

CDI 管理ソフトウェア（管理サーバの Guest OS）を起動するために下記(1)~(4)を行います。

※(4)は必要に応じて任意で実施してください。

(1) 管理サーバ Guest OS 名の一覧を以下のコマンドで取得します。

```
$ sudo virsh list --all
Id      名前                状態
-      RMimg_<version>    shut off
```

(2) 表示した CDI 管理ソフトウェア（管理サーバの Guest OS）の状態が shut off の場合は、以下のコマンドで CDI 管理ソフトウェアを起動します。

```
$ sudo virsh start RMimg_<version>
```

(3) 起動されたことを以下のコマンドで確認します。

```
$ sudo virsh list --all
Id      名前                状態
-      RMimg_<version>    running ← 状態が running であることを確認
```

(4) 次回以降の管理サーバのブート時に、自動的に Guest OS を起動させたい場合は以下のコマンドを実施します。
なお、使用の際は必ず「3.1.3 管理サーバのネットワーク設定を確認」にてネットワークのブリッジ接続が維持されていることを確認してください。

```
$ sudo virsh autostart RMimg_<version>
```

3.1.7 CDI 管理ソフトウェアにログイン

システム管理者 PC から管理サーバの Host OS に接続して、「3.4 CDI 管理ソフトウェアにログインする」の手順を行い CDI 管理ソフトウェアにログインします。

3.1.8 論理サーバの電源をオン

全ての論理サーバの電源をオンするために下記(1)~(3)を行います。

(1) 「2.2.1 論理サーバの一覧を表示」の操作により、グループに存在する論理サーバを確認します。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0
machine1	2	7d6210ef-8a6a-403e-a980-5542b7852c30	user1

(2) 「2.2.10 論理サーバの電源オン」の操作により、確認した全ての論理サーバの電源を ON します。

```
cdi: ./epcctl -u $URL machine power -P on -m machine0
200
{'data': {'machines': [{'mach_uuid': ' b9e25b57-521a-4047-b2cd-6159c452ac26' } ]}}
cdi: ./epcctl -u $URL machine power -P on -m machine1
200
{'data': {'machines': [{'mach_uuid': ' 7d6210ef-8a6a-403e-a980-5542b7852c30' } ]}}
```

- (3) 「2.2.12 論理サーバの状態確認」の操作により、全ての論理サーバの詳細ステータス(mach_status_detail)が "ACTIVE PON"であることを確認します。

```
cdi: ./epcctl -u $URL machine show -m machine0
```

```
...
```

```
"mach_status_detail": "ACTIVE PON",
```

```
cdi: ./epcctl -u $URL machine show -m machine1
```

```
...
```

```
"mach_status_detail": "ACTIVE PON",
```

※mach_status_detailが"ACTIVE PON"以外の場合は、"ACTIVE PON"が表示されるまで確認を繰り返してください。

3.2 システムを終了する

システムを終了するには、以下に示す 3.2.1～3.2.4 の手順を実施してください。

3.2.1 全論理サーバの利用停止

論理サーバの電源オフが行われるため、全ての利用者に対して利用停止を通知します。

3.2.2 CDI 管理ソフトウェアにログイン

システム管理者 PC から管理サーバの Host OS に接続して、「3.4 CDI 管理ソフトウェアにログインする」の手順を行い CDI 管理ソフトウェアにログインします。

3.2.3 論理サーバの電源をオフ

作成した全ての論理サーバの電源をオフするために下記(1)～(3)を行います。

(1) 「2.2.1 論理サーバの一覧を表示」の操作により、グループに存在する論理サーバを確認します。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0
machine1	2	7d6210ef-8a6a-403e-a980-5542b7852c30	user1

(2) 「2.2.11 論理サーバの電源オフ」の操作により、確認した全ての論理サーバの電源を OFF します。

```
cdi: ./epcctl -u $URL machine power -P off -m machine0
200
{'data': {'machines': [{'mach_uuid': ' b9e25b57-521a-4047-b2cd-6159c452ac26' } ]}}
cdi: ./epcctl -u $URL machine power -P off -m machine1
200
{'data': {'machines': [ {'mach_uuid': ' 7d6210ef-8a6a-403e-a980-5542b7852c30' } ]}}
```

(3) 「2.2.12 論理サーバの状態確認」の操作により、全ての論理サーバの詳細ステータス(mach_status_detail)が "INACTIVE POFF"であることを確認します。

```
cdi: ./epcctl -u $URL machine show -m machine0
...
"mach_status_detail": " INACTIVE POFF",

cdi: ./epcctl -u $URL machine show -m machine1
...
"mach_status_detail": " INACTIVE POFF",
```

※mach_status_detail が "INACTIVE POFF"以外の場合は、"INACTIVE POFF"が表示されるまで確認を繰り返してください。

3.2.4 CDI コンポーネントの電源を切断

CDI コンポーネントの電源を切断するために、システム構築手順書「4.9.2 電源切断手順 (2)～(6)」を行います。

3.3 CDI 管理ソフトウェアを再起動する

CDI 管理ソフトウェアを再起動するには、以下に示す 3.3.1~3.3.2 の手順を実施してください。

3.3.1 CDI 管理ソフトウェア（管理サーバの Guest OS）を停止

CDI 管理ソフトウェアを停止するために下記(1)~(3)を行います。

(1) 管理サーバ Guest OS 名の一覧を以下のコマンドで取得します。

```
$ sudo virsh list --all
Id   名前                状態
0    RMimg_<version>    running
```

(2) 表示された CDI 管理ソフトウェア（管理サーバの Guest OS）を下記のコマンドで停止します。

```
$ sudo virsh shutdown RMimg_<version>
```

(3) 停止されたことを下記のコマンドで確認します。

```
$ sudo virsh list --all
Id   名前                状態
-    RMimg_<version>    shut off ← 状態が shut off であることを確認
```

3.3.2 CDI 管理ソフトウェア（管理サーバの Guest OS）を起動

CDI 管理ソフトウェアを起動するために下記(1)~(2)を行います。

(1) 表示した CDI 管理ソフトウェア（管理サーバの Guest OS）の状態が shut off の場合は、以下のコマンドで CDI 管理ソフトウェアを起動します。

```
$ sudo virsh start RMimg_<version>
```

(2) 起動されたことを以下のコマンドで確認します。

```
$ sudo virsh list --all
Id   名前                状態
0    RMimg_<version>    running ← 状態が running であることを確認
```

3.4 CDI 管理ソフトウェアにログインする

CLI コマンドを使用する運用操作、保守操作、システムの監視を行うためには、CDI 管理ソフトウェアへのログインが必要になります。CDI 管理ソフトウェアにログインするには、以下のコマンドを実施してください。なお、使用するユーザ名とパスワードは、システム構築手順書「5.1.5(4) CDI 管理ソフトウェアのユーザとパスワード」で設定したものを使用します。

```
$ cd /home/<管理サーバのユーザ名>/cdi/RMcli<version>/RMcli<version>_<OSname>
↑
RMcli<version>.tar.gz (CLI コマンドのイメージ圧縮ファイル)を展開したディレクトリ (※1) に移動

$ ./epcctl -u $URL login
Enter Your Username : xxxx ← ユーザ名を入力
Enter Your Password : **** ← パスワードを入力
login success
cdi:
```

※1 詳細は、システム構築手順書「4.3.3 作業用 PC から管理サーバへのファイル転送」「4.3.4 tar ファイルの解凍」を参照してください。

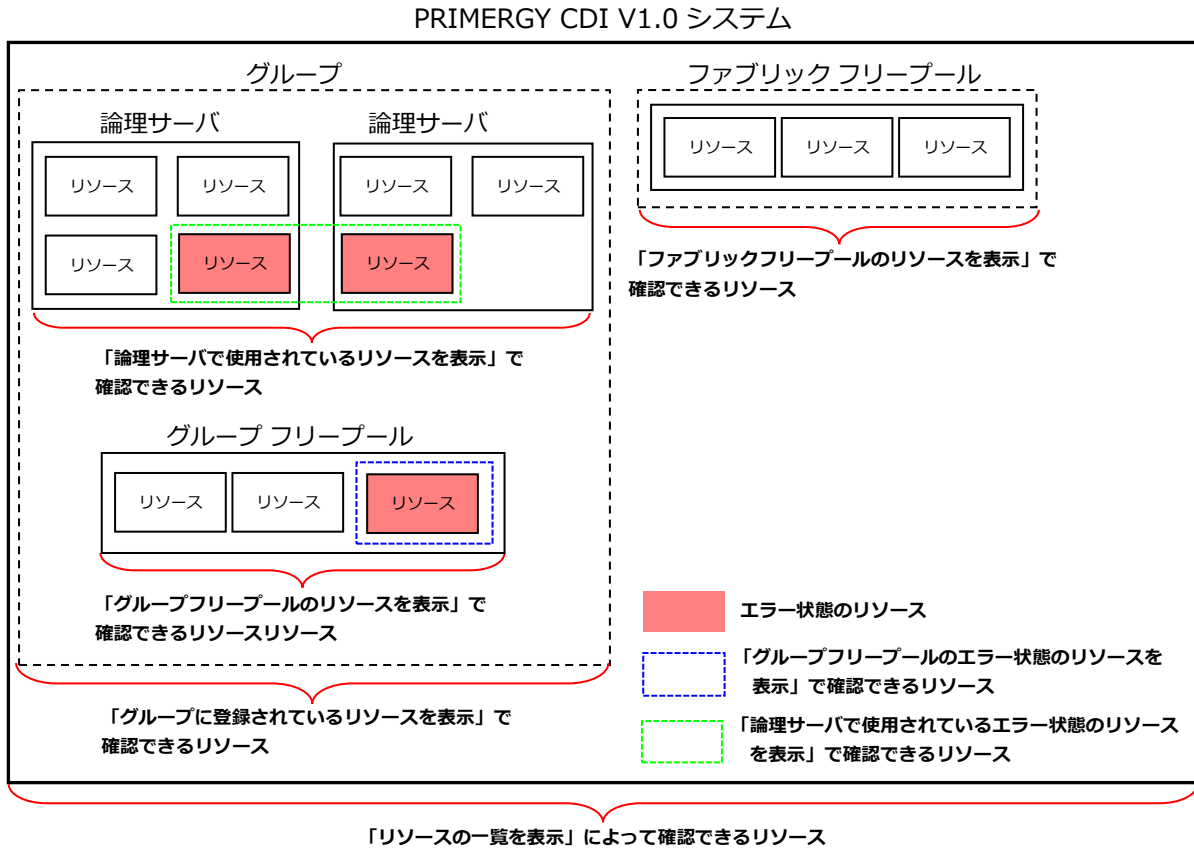
ログインに成功すると"login success"が表示され、プロンプト (cdi:) が表示されます。CLI コマンドによる操作は、ログイン後のプロンプト内で実施してください。

なお、ログインに失敗すると"Unauthorized"が表示されます。ユーザ名、パスワードを確認して再ログインをしてください。

また、ログイン後、一定時間 (30 分) が経過して CLI コマンドを実行すると"Authentication expired"が表示されます。cdi: プロンプトで exit を実行してログアウトした後で、再ログインを実行してください。

3.5 リソースを確認する

CLI コマンドの操作で確認できるシステム内のリソースは以下の通りです。作業に合わせて実施してください。



3.6 グループを作成する

運用で使用するグループを作成するには、以下に示す 3.6.1~3.6.4 の手順を実施してください。なお、作成できるグループの数は 1 つで、論理サーバを作成する前にグループを作成します。

3.6.1 使用できるリソースの確認

グループの作成で使用できるリソース（ファブリックフリープール内のリソース）を確認するために「2.3.2 ファブリックフリープールのリソースを表示」の操作を行います。

※必要に応じて「2.3.6 リソースの詳細を表示」の操作により個々のリソースの詳細情報も確認してください。

res_type	res_name	res_uuid	...	res_op_status	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		0		8
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8		0		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498		0		None
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309		0		None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267		0		None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490		0		None

リソース(gpu0)の詳細情報

item	value
res_uuid	2df31bfc-723e-4781-a83d-4372630c3498
res_name	gpu0
...	
resspec_model	Tesla_T4
resspec_vendor	NVIDIA Corporation
...	

3.6.2 使用するリソースの選択

「3.6.1 使用できるリソースの確認」で確認したリソースから、グループの作成に使用するリソースを選択します。

res_type	res_name	res_uuid	...	res_op_status	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		0		8
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8		0		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498		0		None
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309		0		None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267		0		None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490		0		None

選択

注) res_op_status が 0 以外のリソースは、故障など何らかの問題があるリソースのため選択しないでください。

3.6.3 グループの作成

「3.6.2 使用するリソースの選択」で選択したリソースを使用してグループを作成します。グループの作成は「2.1.3 グループの作成（コマンド指定）」または「2.1.4 グループの作成（ファイル指定）」の操作により行います。

・ **コマンド指定操作(2.1.3)によるグループの作成**

(1) 選択したリソースの res_name を指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL group create -g group0 -r pcpu0, gpu0, ssd0
```

作成するグループの名前 (group0)
 選択したリソースの res_name (pcpu0, gpu0, ssd0)

・ **ファイル指定操作(2.1.4)によるグループの作成**

(1) 設定ファイル (yaml 形式) を作成します。

```
groups:
  grp_name: 'group0'
  premachines: []
  resources:
  -
    res_specs:
    -
      res_type: 'compute'
      res_uuid: 'b9c59f7d-0a92-4af2-a28e-3778e5b922c1'
    -
      res_type: 'gpu'
      res_uuid: '2df31bfc-723e-4781-a83d-4372630c3498'
    -
      res_type: 'storage'
      res_uuid: '4894c94b-a598-4882-ae2c-dab4c36f5267'
  flavors: []
```

作成するグループの名前 (group0)
 選択したリソースの res_uuid (b9c59f7d-0a92-4af2-a28e-3778e5b922c1, 2df31bfc-723e-4781-a83d-4372630c3498, 4894c94b-a598-4882-ae2c-dab4c36f5267)

(2) 作成した設定ファイルを指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL group create -y (1)で作成した設定ファイル名
```

3.6.4 作成したグループの確認

グループフリープールに「3.6.2 使用するリソースの選択」で選択したリソースが存在するグループが作成されたことを「2.1.1 グループの情報を表示」および「2.3.3 グループフリープールのリソースを表示」の操作により確認します。

・ **グループが作成されたことの確認(2.1.1)**

作成したグループの名前が表示されることを確認します。

grp_uuid	grp_id	grp_name	...
dcc23c4e-f9ee-4057-9588-254ec8ca8c19	1	group0	...

・ **グループフリープールのリソースを確認(2.3.3)**

3.6.2 で選択したリソースが表示されることを確認します。

res_type	res_name	...
compute	pcpu0	...
gpu	gpu0	...
storage	ssd0	...

3.7 グループにリソースを追加する

グループフリープールにリソースを追加するには、以下に示す 3.7.1~3.7.4 の手順を実施してください。なお、追加できるリソースはファブリックフリープール内のものに限られ、グループを作成した後に追加が行えます。

3.7.1 追加できるリソースの確認

グループに追加できるリソース（ファブリックフリープール内のリソース）を確認するために「2.3.2 ファブリックフリープールリソースを表示」の操作を行います。

※必要に応じて「2.3.6 リソースの詳細を表示」の操作により個々のリソースの詳細情報も確認してください。

res_type	res_name	res_uuid	...	res_op_status	...	slot_number
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8		0		8
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309		0		None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490		0		None

リソース(gpu1)の詳細情報

item	value
res_uuid	dcf272f4-c108-4a4c-be92-091c4725dabc
res_name	gpu1
...	
resspec_model	Tesla_T4
resspec_vendor	NVIDIA Corporation
...	

3.7.2 追加するリソースの選択

「3.7.1 追加できるリソースの確認」で確認したリソースからグループに追加するリソースを選択します。

res_type	res_name	res_uuid	...	res_op_status	...	slot_number
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8		0		8
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309		0		None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490		0		None

選択

注) res_op_status が 0 以外のリソースは、故障など何らかの問題があるリソースのため選択しないでください。

3.7.3 リソースの追加

「3.7.2 追加するリソースの選択」で選択したリソースをグループに追加します。リソースの追加は「2.1.5 グループへのリソース追加（コマンド指定）」または「2.1.6 グループへのリソース追加（ファイル指定）」の操作により行います。

・ **コマンド指定操作(2.1.5)によるリソースの追加**

(1) 選択したリソースの res_name を指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL group modify -M add -g group0 -r gpu1,ssd1
```

grp_name ← group0
 選択したリソースの res_name ← gpu1,ssd1

・ **ファイル指定操作(2.1.6)によるリソースの追加**

(1) 設定ファイル（yaml 形式）を作成します。

```
groups:
  grp_name: dcc23c4e-f9ee-4057-9588-254ec8ca8c19
  premachines: []
  resources:
  -
    res_specs:
    -
      res_type: 'gpu'
      res_uuid: 'dcf272f4-c108-4a4c-be92-091c4725dabc'
    -
      res_type: 'storage'
      res_uuid: 'a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490'
  flavors: []
```

grp_uuid ← dcc23c4e-f9ee-4057-9588-254ec8ca8c19
 選択したリソースの res_uuid ← dcf272f4-c108-4a4c-be92-091c4725dabc, a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490

(2) 作成した設定ファイルを指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL group modify -M add -y (1)で作成した設定ファイル名
```

3.7.4 グループ内のリソースを確認

グループフリープールに「3.7.2 追加するリソースの選択」で選択したリソースが追加されたことを「2.3.3 グループフリープールのリソースを表示」の操作により確認します。

3.7.2 で選択したリソースがグループフリープールに表示されることを確認します。

res_type	res_name	...
compute	pcpu0	
gpu	gpu0	
gpu	gpu1	...
storage	ssd0	
storage	ssd1	

3.8 グループからリソースを削除する

グループフリープールからリソースを削除するには、以下に示す 3.8.1～3.8.4 の手順を実施してください。なお削除できるリソースはファブリックフリープール内のものに限られ、グループを作成した後に削除が行えます。

3.8.1 削除できるリソースの確認

グループから削除できるリソース（グループフリープール内のリソース）を確認するために「2.3.3 グループフリープールのリソースを表示」の操作を行います。

※必要に応じて「2.3.6 リソースの詳細を表示」の操作により個々のリソースの詳細情報も確認してください。

res_type	res_name	res_uuid	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498		None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267		None

リソース(gpu0)の詳細情報

item	value
res_uuid	2df31bfc-723e-4781-a83d-4372630c3498
res_name	gpu0
...	
resspec_model	Tesla_T4
resspec_vendor	NVIDIA Corporation
...	

3.8.2 削除するリソースの選択

「3.8.1 削除できるリソースの確認」で確認したリソースのうち、グループから削除するリソースを選択します。

res_type	res_name	res_uuid	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498		None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267		None

選択

3.8.3 リソースの削除

「3.8.2 削除するリソースの選択」で選択したリソースをグループから削除します。リソースの削除は「2.1.7 グループからのリソース削除（コマンド指定）」または「2.1.8 グループからのリソース削除（ファイル指定）」の操作により行います。

・ **コマンド指定操作(2.1.7)によるリソースの削除**

(1) 選択したリソースの res_name を指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL group modify -M remove -g group0 -r gpu0
```

grp_name
← 選択したリソースの res_name

・ **ファイル指定操作(2.1.8)によるリソースの削除**

(1) 設定ファイル（yaml 形式）を作成します。

```
groups:
  grp_name: 'd0c23c4e-f9ee-4057-9588-254ec8ca8c19'
  premachines: []
  resources:
  -
    res_specs:
    -
      res_type: 'gpu'
      res_uuid: '2df31bfc-723e-4781-a83d-4372630c3498'
    flavors: []
```

grp_uuid
← 選択したリソースの res_uuid

(2) 作成した設定ファイルを指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL group modify -M remove -y (1)で作成した設定ファイル名
```

3.8.4 グループ内のリソースを確認

グループフリープールから「3.8.2 削除するリソースの選択」で選択したリソースが削除されたことを「2.3.3 グループフリープールのリソースを表示」の操作により確認します。

3.8.2 で選択したリソースがグループフリープールに表示されないことを確認します。

res_type	res_name	...
compute	pcpu0	...
storage	ssd0	

3.9 論理サーバを作成する

論理サーバを作成するには、以下に示す 3.9.1～3.9.7 の手順を実施してください。論理サーバの作成はグループを作成した後に行えます。

3.9.1 使用できるリソースの確認

論理サーバの作成で使用できるリソース（グループフリープール内のリソース）を確認するために「2.3.3 グループフリープールのリソースを表示」の操作を行います。

※必要に応じて「2.3.6 リソースの詳細を表示」の操作を実施して個々のリソースの詳細情報も確認してください。

※使用するリソースとして適当なものがなければ「3.7 グループにリソースを追加する」の操作によりグループフリープールにリソースを追加してください。

res_type	res_name	res_uuid	...	res_op_status	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		0		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498		0		None
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309		0		None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267		0		None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490		0		None

リソース(gpu0)の詳細情報

item	value
res_uuid	2df31bfc-723e-4781-a83d-4372630c3498
res_name	gpu0
...	...
resspec_model	Tesla_T4
resspec_vendor	NVIDIA Corporation
...	...

3.9.2 使用するリソースの選択

「3.9.1 使用できるリソースの確認」で確認したリソースから、論理サーバの作成に使用するリソースを選択します。1つの論理サーバに対し、res_typeがcomputeのリソース（計算サーバ）を必ず1台、res_typeがstorageのリソース（NVMe SSD）を1台以上選択してください。それ以外のリソースは任意台数選択することができます。

res_type	res_name	res_uuid	...	res_op_status	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		0		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498		0		None
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309		0		None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267		0		None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490		0		None

選択

注) res_op_statusが0以外のリソースは、故障など何らかの問題があるリソースのため選択しないでください。

3.9.3 論理サーバの作成

「3.9.2 使用するリソースの選択」で選択したリソースを使用して論理サーバを作成します。論理サーバの作成は、「2.2.3 論理サーバの作成（コマンド指定）」または「2.2.4 論理サーバの作成（ファイル指定）」の操作により行います。

・ **コマンド指定操作(2.2.3)による論理サーバの作成**

(1) 選択したリソースの res_name を指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL machine create -g group0 -m machine0 -r pcpu0, gpu0, ssd0
```

grp_name 作成する論理サーバの名前

← 選択したリソースの res_name

・ **ファイル指定操作(2.2.4)による論理サーバの作成**

(1) 設定ファイル (yaml 形式) を作成します。

```
groups:
  grp_name: "group0"
  machines:
    - mach_name: "machine0"
      resources:
        - res_specs:
            res_type: 'compute'
            res_name: 'pcpu0'
        - res_type: 'gpu'
          res_name: 'gpu0'
        - res_type: 'storage'
          res_name: 'ssd0'
```

grp_name

作成する論理サーバの名前

← 選択したリソースの res_name

(2) 作成した設定ファイルを指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL machine create -y (1)で作成した設定ファイル名
```

3.9.4 ステータスの確認（論理サーバの起動前）

「3.9.3 論理サーバの作成」で作成した論理サーバの詳細ステータス(mach_status_detail)が“INACTIVE POFF”であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ “INACTIVE POFF” 以外が表示された場合は、“INACTIVE POFF”が表示されるまで確認を繰り返します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": "INACTIVE POFF",
"power_status": "unknown",
"lanports": null
```

3.9.5 作成した論理サーバの確認

「3.9.2 使用するリソースの選択」で選択されたリソースで構成される論理サーバが作成されたことを「2.2.1 論理サーバの一覧を表示」

および「2.3.5 論理サーバで使用されているリソースを表示」の操作により確認します。

- ・論理サーバが作成されたことの確認(2.2.1)

作成した論理サーバの名前が表示されることを確認します。

mach_name	mach_id	...
machine0	1	...

- ・論理サーバのリソースの確認(2.3.5)

「3.9.2 使用するリソースの選択」で選択したリソースが表示されることを確認します。

res_type	res_name	...
compute	pcpu0	...
gpu	gpu0	...
storage	ssd0	...

3.9.6 作成した論理サーバの電源オン

作成した論理サーバの電源をオンするために「2.2.10 論理サーバの電源オン」の操作を行います。

```
cdi: ./epcctl -u $URL machine power -P on -m machine0 ← 作成した論理サーバの mach_name
```

3.9.7 ステータスの確認（論理サーバの起動後）

「3.9.6 作成した論理サーバの電源オン」で電源をオンした論理サーバの詳細ステータス(mach_status_detail)が"ACTIVE PON"であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ "ACTIVE PON"以外が表示された場合は、"ACTIVE PON"が表示されるまで確認を繰り返します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 13,
"mach_status_detail": " ACTIVE PON",
"power_status": "unknown",
"lanports": null
```


3.10 論理サーバの構成を変更する（リソースの追加）

論理サーバへリソースを追加するには、以下に示す 3.10.1～3.10.10 の手順を実施してください。リソースの追加は論理サーバを作成した後に実施できます。

3.10.1 リソースを追加する論理サーバの確認

リソースを追加する論理サーバを確認するために「2.2.1 論理サーバの一覧を表示」の操作を行います。

※必要に応じて「2.2.2 論理サーバの詳細を表示」の操作により論理サーバの詳細情報も確認してください。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0

論理サーバ(machine0)の詳細情報

item	value
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26
mach_id_nonliquid	
mach_id	
mach_name	machine0
mach_status	13
mach_op_status	00
mach_status_detail	ACTIVE PON
...	...

3.10.2 論理サーバの電源オフ

「3.10.1 リソースを追加する論理サーバの確認」で確認した論理サーバの電源をオフするために「2.2.11 論理サーバの電源オフ」の操作を行います。

```
cdi: ./epcctl -u $URL machine power -P off -m machine0
```

← 確認した論理サーバの mach_name

3.10.3 ステータスの確認（リソース追加前）

「3.10.2 論理サーバの電源オフ」で電源をオフした論理サーバの詳細ステータス(mach_status_detail)が“INACTIVE POFF”であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ “INACTIVE POFF”以外が表示された場合は、“INACTIVE POFF”が表示されるまで確認を繰り返します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": " INACTIVE POFF",
"power_status": "unknown",
"lanports": null
```

3.10.4 追加できるリソースの確認

論理サーバに追加できるリソース（グループフリープール内のリソース）を確認するために「2.3.3 グループフリープールのリソースを表示」の操作を行います。

※必要に応じて「2.3.6 リソースの詳細を表示」の操作によりリソースの詳細情報も確認してください。

※使用するリソースとして適当なものがなければ「3.7 グループにリソースを追加する」の操作によりグループフリープールにリソースを追加してください。

res_type	res_name	res_uid	...	res_op_status	...	slot_number
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309	...	0	...	None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490	...	0	...	None

リソース(gpu1)の詳細情報

item	value
res_uid	dcf272f4-c108-4a4c-be92-091c4725dabc
res_name	gpu1
...	...
resspec_model	Tesla_T4
resspec_vendor	NVIDIA Corporation
...	...

3.10.5 追加するリソースの選択

「3.10.4 追加できるリソースの確認」で確認したリソースから、論理サーバに追加するリソースを選択します。

res_type	res_name	res_uid	...	res_op_status	...	slot_number
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309	...	0	...	None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490	...	0	...	None

注) res_op_status が 0 以外のリソースは、故障など何らかの問題があるリソースのため選択しないでください。

注) res_type が compute のリソースは選択できません。（追加できません）

3.10.6 リソースの追加

「3.10.5 追加するリソースの選択」で選択したリソースを論理サーバに追加します。リソースの追加は「2.2.5 論理サーバへのリソース追加（コマンド指定）」または「2.2.6 論理サーバへのリソース追加（ファイル指定）」の操作により行います。

・ **コマンド指定操作(2.2.5)によるリソースの追加**

(1) 選択したリソースの res_name を指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL machine modify -M add -m machine0 -r gpu1
```

mach_name ← machine0
← 選択したリソースの res_name ← gpu1

・ **ファイル指定操作(2.2.6)によるリソースの追加**

(1) 設定ファイル (yaml 形式) を作成します。

```
machines:
  mach_name: "machine0"
  resources:
  -
    res_specs:
    -
      res_type: 'gpu'
      res_name: 'gpu1'
```

← mach_name ← machine0
← 選択したリソースの res_name ← gpu1

(2) 作成した設定ファイルを指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL machine modify -M add -y (1)で作成した設定ファイル名
```

3.10.7 ステータスの確認（リソース追加後）

「3.10.6 リソースの追加」でリソースを追加した論理サーバの詳細ステータス(mach_status_detail)が "INACTIVE POFF"であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ "INACTIVE POFF" 以外が表示された場合は、"INACTIVE POFF"が表示されるまで確認を繰り返します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": " INACTIVE POFF",
"power_status": "unknown",
"lanports": null
```

3.10.8 論理サーバのリソースを確認

「3.10.5 追加するリソースの選択」で選択したリソースが論理サーバに追加されたことを「2.3.5 論理サーバで使用されているリソースを表示」の操作により確認します。

選択したリソースが表示されることを確認します。

res_type	res_name	...
compute	pcpu0	
gpu	gpu0	...
gpu	gpu1	
storage	ssd0	

3.10.9 論理サーバの電源オン

リソースを追加した論理サーバの電源をオンするために「2.2.10 論理サーバの電源オン」の操作を行います。

```
cdi: ./epcctl -u $URL machine power -P on -m machine0 ← mach_name
```

3.10.10 ステータスの確認（論理サーバの起動後）

「3.10.9 論理サーバの電源オン」で電源をオンした論理サーバの詳細ステータス(mach_status_detail)が "ACTIVE PON"であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ "ACTIVE PON"以外が表示された場合は、"ACTIVE PON"が表示されるまで確認を繰り返します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 13,
"mach_status_detail": " ACTIVE PON",
"power_status": "unknown",
"lanports": null
```

3.11 論理サーバの構成を変更する（リソースの削除）

論理サーバからリソースを削除するには、以下に示す 3.11.1~3.11.10 の手順を実施してください。リソースの削除は論理サーバを作成した後に実施できます。

3.11.1 リソースを削除する論理サーバの確認

グループに存在する論理サーバを確認するために「2.2.1 論理サーバの一覧を表示」の操作を行います。
 ※必要に応じて「2.2.2 論理サーバの詳細を表示」の操作により論理サーバの詳細情報も確認してください。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0

論理サーバ(machine0)の詳細情報

item	value
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26
mach_id_nonliquid	
mach_id	
mach_name	machine0
mach_status	13
mach_op_status	00
mach_status_detail	ACTIVE PON
...	...

3.11.2 論理サーバの電源オフ

「3.11.1 リソースを削除する論理サーバの確認」で確認した論理サーバの電源をオフするために「2.2.11 論理サーバの電源オフ」の操作を行います。

cdi: ./epcctl -u \$URL machine power -P off **-m machine0** ← 確認した論理サーバの mach_name

3.11.3 ステータスの確認（リソース削除前）

「3.11.2 論理サーバの電源オフ」で電源をオフした論理サーバの詳細ステータス(mach_status_detail)が“INACTIVE POFF”であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ “INACTIVE POFF”以外が表示された場合は、“INACTIVE POFF”が表示されるまで確認を繰り返します。

```

"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": " INACTIVE POFF",
"power_status": "unknown",
"lanports": null
    
```

3.11.4 削除できるリソースの確認

論理サーバに存在するリソースを確認するために「2.3.5 論理サーバで使用されているリソースを表示」の操作を行います。

※必要に応じて「2.3.6 リソースの詳細を表示」の操作によりリソースの詳細情報も確認してください。

res_type	res_name	res_uuid	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498	...	None
storage	ssd0	8e2abd0a-61a6-4b44-87f2-648aa0a0031a		None

リソース(gpu0)の詳細情報

item	value
res_uuid	e2ba154d-d7fa-4db7-8a6d-816337d92079
res_name	gpu0
...	

3.11.5 削除するリソースの選択

「3.11.4 削除できるリソースの確認」で確認したリソースの中で、論理サーバから削除するリソースを選択します。

res_type	res_name	res_uuid	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498	...	None
storage	ssd0	8e2abd0a-61a6-4b44-87f2-648aa0a0031a		None

選択

3.11.6 リソースの削除

「3.11.5 削除するリソースの選択」で選択したリソースを論理サーバから追加します。リソースの削除は「2.2.7 論理サーバからのリソース削除（コマンド指定）」または「2.2.8 論理サーバからのリソース削除（ファイル指定）」の操作により行います。

・ **コマンド指定操作(2.2.7)によるリソースの削除**

(1) 選択したリソースの res_name を指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL machine modify -M remove -m machine0 -r gpu0
```

mach_name (pointing to machine0)
 選択したリソースの res_name (pointing to gpu0)

・ **コマンド指定操作(2.2.8)によるリソースの削除**

(1) 設定ファイル (yaml 形式) を作成します。

```
machines:
  mach_name: machine0
  resources:
    -
      res_specs:
        -
          res_type: 'gpu'
          res_name: 'gpu0'
```

mach_name (pointing to machine0)
 選択したリソースの res_name (pointing to gpu0)

(2) 作成した設定ファイルを指定してコマンドを実行します。

```
cdi: ./epcctl -u $URL machine modify -M remove -y (1)で作成した設定ファイル名
```

3.11.7 ステータスの確認（リソース削除後）

「3.11.6 リソースの削除」でリソースを削除した論理サーバの詳細ステータス(mach_status_detail)が“INACTIVE POFF”であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ “INACTIVE POFF”以外が表示された場合は、“INACTIVE POFF”が表示されるまで確認を繰り返します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": " INACTIVE POFF",
"power_status": "unknown",
"lanports": null
```

3.11.8 論理サーバのリソースを確認

「3.11.5 削除するリソースの選択」で選択したリソースが論理サーバから削除されたことを「2.3.5 論理サーバで使用されているリソースを表示」の操作により確認します。

選択したリソースが表示されないことを確認します。

res_type	res_name	...
compute	pccpu0	...
storage	ssd0	...

3.11.9 論理サーバの電源オン

リソースを削除した論理サーバの電源をオンするために「2.2.10 論理サーバの電源オン」の操作を行います。

```
cdi: ./epcctl -u $URL machine power -P on -m machine0 ← mach_name
```

3.11.10 ステータスの確認（論理サーバの起動後）

「3.11.9 論理サーバの電源オン」で電源をオンした論理サーバの詳細ステータス(mach_status_detail)が“ACTIVE PON”であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ “ACTIVE PON”以外が表示された場合は、“ACTIVE PON”が表示されるまで確認を繰り返します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 13,
"mach_status_detail": " ACTIVE PON",
"power_status": "unknown",
"lanports": null
```

3.12 論理サーバからエラー状態のリソースを削除する

論理サーバからエラー状態のリソースを削除するには、以下に示す 3.12.1~3.12.8 の手順を実施してください。
 ※エラー状態のリソースとは res_op_status が 0 以外のリソースです。

3.12.1 エラー状態のリソースを確認

エラー状態のリソースが組み込まれた論理サーバの mach_uuid を確認するために下記(1)~(2)を行います。

- (1) 「2.3.8 論理サーバで使用されているエラー状態のリソースを表示」の操作により、エラー状態のリソースの res_name を確認します。

```

- res_name: gpu0
  res_op_status: '2'
  res_owner: n/a
  res_status: 1
  res_type: gpu
  res_uuid: 2df31bfc-723e-4781-a83d-4372630c3498
  rack_name: 'rack0'
  position_in_rack: '10U'
  slot_number: 0
    
```

- (2) 「2.3.6 リソースの詳細を表示」の操作により、(1)で確認した res_name のリソースの詳細情報を表示して論理サーバの mach_uuid を確認します。

リソース(gpu0)の詳細情報

item	value
res_uuid	2df31bfc-723e-4781-a83d-4372630c3498
res_id_nonliquid	
fabr_gid	0x033c00
res_type	gpu
res_name	gpu0
res_status	4
...	
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26

3.12.2 エラー状態のリソースが組み込まれた論理サーバの特定

「2.2.1 論理サーバの一覧を表示」の操作で表示される論理サーバの中から、mach_uuid が 3.12.1 で確認したものと同一論理サーバの mach_name を特定します。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0
machine1	2	7d6210ef-8a6a-403e-a980-5542b7852c30	user1

3.12.3 論理サーバの電源オフ

「3.11.5 エラー状態のリソースが組み込まれた論理サーバの特定」で特定した論理サーバの電源をオフするために「2.2.11 論理サーバの電源オフ」の操作を行います。

```

cdi: ./epcctl -u $URL machine power -P off -m machine0
    
```

3.12.4 ステータスの確認（論理サーバの電源オフ後）

「3.12.3 論理サーバの電源オフ」で電源をオフした論理サーバの詳細ステータス(mach_status_detail)が、“INACTIVE POFF”であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ “INACTIVE POFF”以外が表示された場合は、“INACTIVE POFF”が表示されるまで確認を繰り返します。

```

"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": " INACTIVE POFF",
"power_status": "unknown",
"lanports": null
    
```

3.12.5 エラー状態のリソースを削除

「3.12.1 エラー状態のリソースを確認」で確認したエラー状態のリソースを「2.2.7 論理サーバからのリソース削除（コマンド指定）」の操作により論理サーバから削除します。

```

cdi: ./epcctl -u $URL machine modify -M remove -m machine0 -r gpu0
    
```

← 3.12.2 で特定した mach_uuid
← 3.12.1 で確認した res_name

3.12.6 ステータスの確認（エラー状態のリソース削除後）

「2.2.12 論理サーバの状態確認」の操作により論理サーバの詳細ステータス(mach_status_detail)が“INACTIVE POFF”であることを確認します。

※ “INACTIVE POFF”以外が表示された場合は、“INACTIVE POFF”が表示されるまで確認を繰り返します。

```

"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": " INACTIVE POFF",
"power_status": "unknown",
"lanports": null
    
```

3.12.7 エラー状態のリソースが削除されたことを確認

グループフリープールに「3.12.1 エラー状態のリソースを削除」で削除したエラー状態のリソースが移動したことを「2.3.3 グループフリープールのリソースを表示」の操作により確認します。

※エラー状態のリソースが表示されることを確認します。

res_type	res_name	res_uuid	...	slot_number
compute	pcpu0	9058a156-9c83-4824-a733-fb0c88fe4ada		8
gpu	gpu0	5b30e05a-ef22-4349-b4fd-5cdb45029a2		None
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309		None
storage	ssd0	b307aff0-69f1-4a26-8785-6559c042d096		None
storage	ssd1	277c019b-2cf2-43a6-a3c3-55ac277ac0ff		None

3.12.8 エラー状態のリソースをファブリックフリープールに移動

「3.12.1 エラー状態のリソースを削除」で削除されたエラー状態のリソースをファブリックフリープールに移動するために「2.1.7 グループからのリソース削除（コマンド指定）」の操作を行います。

```

cdi: ./epcctl -u $URL group modify -M remove -g group0 -r gpu0
    
```

← grp_name
← 3.12.5 で削除されたエラー状態のリソースの res_name

3.13 グループフリープールからエラー状態のリソースを削除する

グループフリープールからエラー状態のリソースを削除するには、以下に示す 3.13.1～3.13.2 の手順を実施してください。なお、寿命が近づいた NVMe SSD もエラー状態として扱います。

※エラー状態のリソースとは res_op_status が 0 以外のリソースです。

3.13.1 エラー状態のリソースを確認

グループフリープールのエラー状態のリソースを確認するために「2.3.7 グループフリープールのエラー状態のリソースを表示」の操作を行います。

```
- res_name: gpu0
  res_op_status: 2
  res_owner: n/a
  res_status: 1
  res_type: gpu
  res_uuid: 2df31bfc-723e-4781-a83d-4372630c3498
  rack_name: 'rack0'
  position_in_rack: '10U'
  slot_number: 0
```

3.13.2 エラー状態のリソースを削除

「3.13.1 エラー状態のリソースを確認」で確認したエラー状態のリソースをファブリックフリープールに移動するために「2.1.7 グループからのリソース削除（コマンド指定）」の操作を行います。

```
cdi: ./epcctl -u $URL group modify -M remove -g group0 -r gpu0
```

grp_name
3.13.1 で確認したエラー状態のリソースの res_name

3.14 論理サーバを解体する

論理サーバを解体するには、以下に示す 3.14.1～3.14.6 の手順を実施してください。

3.14.1 論理サーバの確認

グループに存在する論理サーバを確認するために「2.2.1 論理サーバの一覧を表示」の操作を行います。
 ※必要に応じて「2.2.2 論理サーバの詳細を表示」の操作により論理サーバの詳細情報も確認してください。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0
machine1	2	7d6210ef-8a6a-403e-a980-5542b7852c30	user1

論理サーバ(machine0)の詳細情報

item	value
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26
mach_id_nonliquid	
mach_id	
mach_name	machine0
mach_status	13
mach_op_status	00
mach_status_detail	ACTIVE PON
...	...

3.14.2 解体する論理サーバの選択

「3.14.1 論理サーバの確認」で確認した論理サーバから解体するものを選択します。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0
machine1	2	7d6210ef-8a6a-403e-a980-5542b7852c30	user1

→ 選択

3.14.3 論理サーバの電源オフ

「3.14.2 解体する論理サーバの選択」で選択した論理サーバの電源をオフするために「2.2.11 論理サーバの電源オフ」の操作を行います。

cdi: ./epcctl -u \$URL machine power -P off **-m machine0** ← 選択した論理サーバの mach_name

3.14.4 ステータスの確認

「2.2.12 論理サーバの状態確認」の操作により解体する論理サーバの詳細ステータス(mach_status_detail)が“INACTIVE POFF”であることを確認します。

※ “INACTIVE POFF”以外が表示された場合は、“INACTIVE POFF”が表示されるまで確認を繰り返します。

```

"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": " INACTIVE POFF",
"power_status": "unknown",
"lanports": null
    
```

3.14.5 論理サーバを解体

「3.14.2 解体する論理サーバの選択」で選択した論理サーバを解体するために「2.2.9 論理サーバの解体」の操作を行います。

※論理サーバに res_type が storage のリソースが組み込まれていた場合は、リソース(NVMe SSD)のデータは消去されます。

```

cdi: ./epcctl -u $URL machine destroy -m machine0 ← 選択した論理サーバの mach_name
    
```

3.14.6 解体の確認

「3.14.5 論理サーバを解体」で解体した論理サーバが削除されたことを「2.2.1 論理サーバの一覧を表示」の操作により確認します。

※解体した論理サーバが表示されないことを確認します。

mach_name	mach_id	mach_uuid	mach_owner
machine1	2	7d6210ef-8a6a-403e-a980-5542b7852c30	user1

3.15 グループを解体する

グループを解体するには、以下に示す 3.15.1~3.15.4 の手順を実施してください。なお、グループの解体はグループ内の全ての論理サーバを解体した後に行います。

3.15.1 グループの確認

解体するグループを確認するために「2.1.1 グループの情報を表示」の操作を行います。

※必要に応じて「2.1.2 グループの詳細情報を表示」の操作によりグループの詳細情報も確認してください。

grp_uuid	grp_id	grp_name	machines
dcc23c4e-f9ee-4057-9588-254ec8ca8c19	1	group0	[{'uuid': '3027d7e8-6bd6-422a-8130-a12c809be254', 'status': 10, 'option_status': '00'}]

グループ(group0)の詳細情報

```
- fbr_id: 1
  grp_id: 1
  grp_id_nonliquid: null
  grp_name: g0
  grp_op_status: '000'
  grp_owner: Fujitsu
  . . .
```

3.15.2 グループ内の論理サーバを確認

グループに存在する論理サーバを確認するために「2.2.1 論理サーバの一覧を表示」の操作を行います。

※必要に応じて「2.2.2 論理サーバの詳細を表示」の操作で個々の論理サーバの詳細情報も確認してください。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0
machine1	2	7d6210ef-8a6a-403e-a980-5542b7852c30	user1

論理サーバ(machine0)の詳細情報

item	value
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26
mach_id_nonliquid	
mach_id	
mach_name	machine0
mach_status	10
. . .	

3.15.3 グループ内の論理サーバを解体

「3.15.2 グループ内の論理サーバを確認」で確認したグループ内の全ての論理サーバを「3.14 論理サーバを解体する」の操作により解体します。

3.15.4 グループを解体

グループを解体するために「2.1.9 グループの解体」の操作を実施します。

※解体により、グループ内の全てのリソースはファブリックプールに移動します。

cdi: ./epcctl -u \$URL group destroy **-g group0** ← グループの grp_name

3.16 OS イメージを作成する

論理サーバに組み込む OS イメージを作成するには、以下に示す 3.16.1~3.16.8 の手順を実施してください。
 なお、作成では計算サーバ:1 台と NVMe SSD:1~2 台で構成されるイメージ作成用論理サーバを使用するため、
 利用可能な計算サーバと NVMe SSD がそれぞれ 1 台以上残存していることが OS イメージを作成する条件となります。

3.16.1 イメージ作成用論理サーバの作成

イメージ作成用論理サーバを作成するために下記①~⑤を行います。

- ① リソース（計算サーバ, NVMe SSD）を確認するために「2.3.3 グループフリープールのリソースを表示」の操作を行います。

※必要に応じて「2.3.6 リソースの詳細を表示」の操作により個々のリソースの詳細情報も確認してください。

※使用するリソースとして適当なものがないければ「3.7 グループにリソースを追加する」の操作によりグループフリープールにリソースを追加してください。

res_type	res_name	res_uuid	...	res_op_status	...	slot_number
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8	...	0	...	8
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490	...	0	...	None

リソース(ssd1)の詳細情報

item	value
res_uuid	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490
res_name	ssd1
...	...

- ② ①で確認したリソースから作成で使用するリソース（計算サーバ, NVMe SSD）を選択します。後述の「3.16.4 ISO イメージのインストール」でバーチャルメディア機能を利用する場合は res_type が storage の NVMe SSD を 1 台、CLI コマンドを利用する場合は 2 台選択ください。

res_type	res_name	res_uuid	...	res_op_status	...	slot_number
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8	...	0	...	8
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	...	0	...	None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490	...	0	...	None

選択

注)res_op_status が 0 以外のリソースは、故障など何らかの問題があるリソースのため選択しないでください。

- ③ ②で選択したリソースを使用してイメージ作成用の論理サーバを作成します。論理サーバの作成は「2.2.3 論理サーバの作成（コマンド指定）」の操作により行います。

```
cdi: ./epcctl -u $URL machine create -g group0 -m machine0 -r pcpu1,ssd1
```

グループの grp_name (group0) 作成する論理サーバの名前 (machine0) 選択したリソースの res_name (pcpu1,ssd1)

- ④ ③で作成した論理サーバの詳細ステータス(mach_status_detail)が“INACTIVE POFF”であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": " INACTIVE POFF",
"power_status": "unknown",
"lanports": null
```

※ “マシン作成中(CREATING)”が表示された場合は“INACTIVE POFF”が表示されるまで確認を繰り返します。

- ⑤ ②で選択されたリソースで構成される論理サーバが作成されたことを「2.2.1 論理サーバの一覧を表示」および「2.3.5 論理サーバで使用されているリソースを表示」の操作により確認します。

- ・作成した論理サーバの名前が表示されることを確認します。
- ・選択したリソースが表示されることを確認します。

mach_name	mach_id	...
machine0	1	...

res_type	res_name	...
compute storage	pcpu1 ssd1	...

3.16.2 OS 種類およびカーネル版数の決定

作成可能な OS イメージを確認し、OS の種類およびカーネル版数を決定します。下記のサイトにアクセスして OS の種類および、カーネル版数を決定下さい。

<https://www.fujitsu.com/jp/products/computing/servers/primergy/software/>

3.16.3 ISO イメージの取得

OS イメージに使用する ISO イメージファイル(.ISO 形式のファイル)を取得します。OS イメージに Linux OS を使用する場合は、下記のサイトに記載された手順に沿って ISO イメージファイルを取得してください。

<https://www.fujitsu.com/jp/products/computing/servers/primergy/software/linux/install/>

※サイトには複数の取得方法が記載されていますが、サポート契約を確認のうえ適当な取得手順を選択ください。以降は、OS イメージに Red Hat Enterprise Linux を使用するものとし、ISO イメージファイル (RedHat.ISO) をシステム管理者 PC で取得した場合を例に記載します。

3.16.4 ISO イメージのインストール

「3.16.3 ISO イメージの取得」で取得した ISO イメージを「3.16.1 イメージ作成用論理サーバの作成」で作成したイメージ作成用論理サーバの NVMe SSD にインストールします。以下の「A. バーチャルメディア機能を利用する場合」または「B. CLI コマンドを利用する場合」のいずれかを選択ください。

※操作で使用している iRMC の詳細は、システム構築手順書および、iRMC の操作手引きを参照してください。

※操作で使用している iRMC の一部機能（アドバンストビデオリダイレクション機能、バーチャルメディア機能）を利用するためには、予め「リモートマネジメントコントローラアップグレード:PY-RMC44」のご購入が必要になります。

A. バーチャルメディア機能を利用する場合

下記(A-1)~(A-7)の操作により ISO イメージをインストールします。

- (A-1) システム構築手順書「G.1 iRMC へのログイン」を参考にシステム管理者 PC から計算サーバの iRMC にログインします。IP アドレス等のログイン情報は、システム構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照してください。

① iRMCのIPアドレスを指定して下さい。
 ② Username, Passwordを入力して下さい

工場出荷時のUsername/Passwordは以下

- Username: admin
- Password: admin

(Passwordは適切に変更して下さい)

(A-2) システム構築手順書「G.4 ビデオリダイレクション機能」を参考に機能の設定とビデオリダイレクションの起動を行います。本機能を使用する前に画面制御の方式を以下で選択します。

①～③ TOP画面から、Settings > Services > Advanced Video Redirection(AVR)を選択して下さい。
 ④ ブルダウンメニューで、HTML5かJviewerを選択して下さい。

- システム管理者 PC の Web ブラウザが HTML5 に対応している場合は HTML5 を選択
- システム管理者 PC に Java をインストールしてある場合は Jviewer(Java)を選択

設定後、ビデオリダイレクションを起動ください。

① ビデオリダイレクションボタン：クリックするとビデオリダイレクション機能が開始されます。

② この画面を出した状態で、主電源ボタンを押すと、BIOSの起動が開始されF2を押す事でBIOS画面が表示されます。
注意：バーチャルメディア機能を使う場合は、まだ主電源ボタンを押さないで下さい。

(A-3) システム構築手順書の「G.5 パーチャルメディア機能」を参考にシステム管理者 PC にある ISO ファイルを対象サーバにリモートマウントします。

① ビデオリダイレクション機能を起動した後、Selectボタンを押して下さい。

② ダイアログボックスが開きますので、ISOファイルを選択して下さい。

③ 選択したISOファイルが表示されている事を確認して下さい。

④ Start MediaボタンをクリックするとCDイメージがマウントされます。

(A-4) システム構築手順書の「G.2 主電源のオン・オフ機能」を参考に PRIMERGY 主電源を投入します。

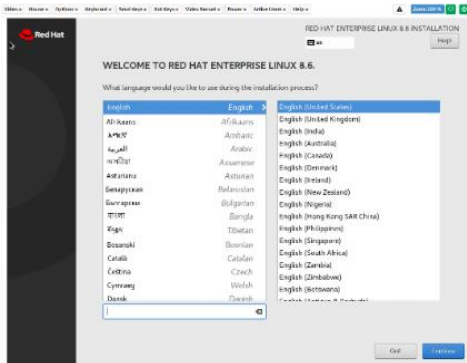
表示の意味

- 主電源がオフ
- 主電源がオン

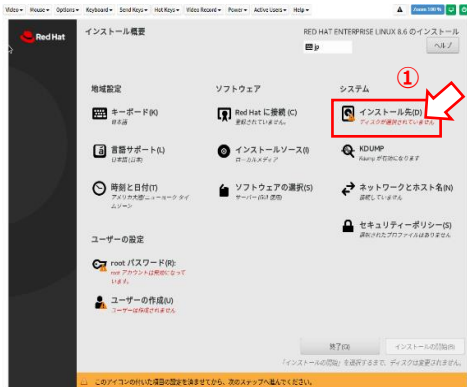
① 主電源オン・オフボタン：現在電源がオフの時にクリックすると主電源がオンになります。現在電源がオンに時にクリックすると主電源がオフになります。

(A-5) PRIMERGY が起動し、リモートマウントした ISO ファイルからのブートが行われます。ブートが失敗する場合は、一度電源オフにしたうえでビデオリダイレクションメニューの「Power」→「Boot Options」を選択し、「Boot Options」で「CDROM/DVD」を選択したうえで再度電源を投入してください。

(A-6) ISO ファイルからのブートが成功すると OS インストール画面に移行します。移行後、手順に沿って設定を行います。



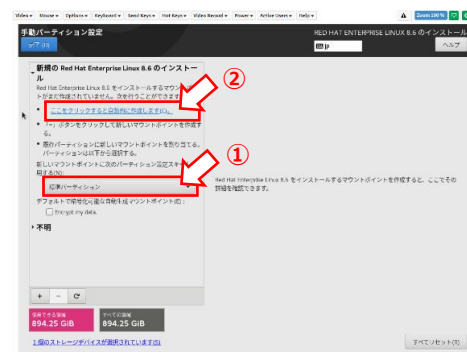
OS イメージのサイズ上限は 16GB のためパーティションを調整し、総サイズが 12GB 以下となるように設定します。



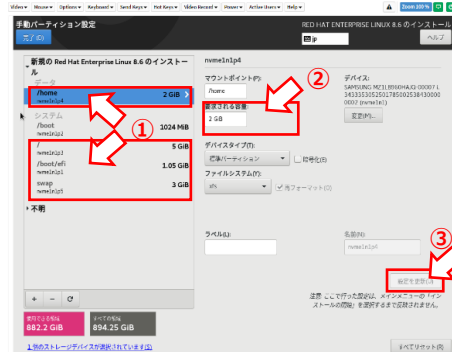
① 「インストール概要」で「インストール先」を選択します。



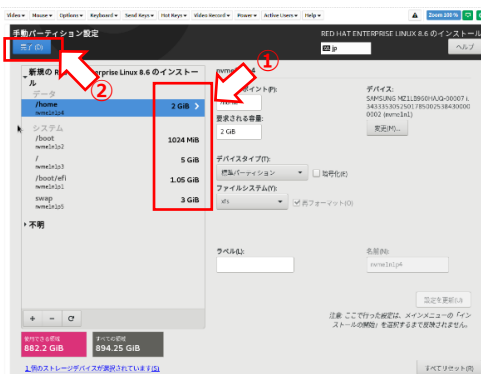
② 「デバイスの選択」で「ローカル標準ディスク」を選択します。
③ 「ストレージの設定」で「カスタム」を選択します。
④完了を選択します。



① 「標準パーティション」を選択します。
② 「ここをクリックすると自動的に作成します」を選択します。



① /boot を除くパーティションのサイズを変更します。各パーティションを選択して下さい。
② 「要求される容量」にてサイズを変更してください。
③ 「設定を変更」を選択します。



④各パーティションサイズの合計が 12GB になるようにサイズ設定を行ってください。
⑤サイズ変更後に「完了」を押下します。「変更の概略」が表示された場合は「変更を許可する」を選択します。

その他以下の設定は指定がありません。システム管理者は任意のものを設定ください。

- ・ OS のホスト名、ログインユーザ名、ログインパスワード
- ・ ソフトウェアパッケージ
- ・ ネットワーク設定

(A-7) 設定完了後、ディスクへの OS インストールを開始します。インストールが完了後、OS をリブートして下さい。リブート後ログイン画面が表示されればインストールは完了です。

B. CLI コマンドを利用する場合

下記(B-1)~(B-8)の操作によりインストールします。「3.16.1 イメージ作成用論理サーバの作成」の②で NVMe SSD 2 台が論理サーバに必ず接続されていることを確認したうえで以降の操作を実施ください。

(B-1) CLI コマンドを使用するために、「3.4 CDI 管理ソフトウェアにログインする」手順に沿って CDI 管理ソフトウェアにログインします

(B-2) 「2.4.3 OS イメージの登録」を参考に、「3.16.3 ISO イメージの取得」で取得した ISO イメージファイルをシステムに登録します。

※登録できる ISO イメージファイルの大きさは、16GB 以下になります。また、同じ名前の ISO イメージファイルに登録することはできません。

```
cdi: ./epcctl -u $URL image add -t iso -i xxx.iso
200
{
  'booting_uuid': 'fec12e74-421b-48f3-bf34-3075205a5444'
}
```

登録する OS イメージの種別(必ず「iso」を指定)

登録する ISO イメージのファイル名

(B-3) 「2.4.5 OS イメージのインストール」を参考に、(B-2)で登録した ISO イメージファイルを論理サーバにインストールします。

※インストール先となる NVMe SSD は「2.3.5 論理サーバで使用されているリソースを表示」により表示される res_name で指定します。

```
cdi: ./epcctl -u $URL resource install -r ssd1 -b xxx.iso
200
{
  'booting_uuid': 'fec12e74-421b-48f3-bf34-3075205a5444'
}
```

インストール先となる NVMe SSD の res_name

ISO イメージのファイル名

(B-4) 「2.2.10 論理サーバの電源オン」を参考に、ISO イメージファイルをインストールした論理サーバの電源を投入します。

※論理サーバは「2.2.1 論理サーバの一覧を表示」により表示される mach_name で指定します。

例) machine0 の電源をオン

```
cdi: ./epcctl -u $URL machine power -P on -m machine0
200
{ 'data': { 'machines': [ { 'mach_uuid': ' b9e25b57-521a-4047-b2cd-6159c452ac26' } ] } }
```

mach_name

(B-5) システム構築手順書「G.1 iRMC へのログイン」を参考にシステム管理者 PC から計算サーバの iRMC にログインします。IP アドレス等のログイン情報は、システム構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照してください。

① iRMCのIPアドレスを指定して下さい。
 ② Username, Passwordを入力して下さい

工場出荷時のUsername/Passwordは以下

- Username: admin
- Password: admin

(Passwordは適切に変更して下さい)

(B-6) システム構築手順書「G.4 ビデオリダイレクション機能」を参考に機能の設定とビデオリダイレクションの起動を行います。本機能を使用する前に画面制御の方式を以下で選択します。

①～③ TOP画面から、Settings > Services > Advanced Video Redirection(AVR)を選択して下さい。
 ④ ブルダウンメニューで、HTML5かJviewerを選択して下さい。

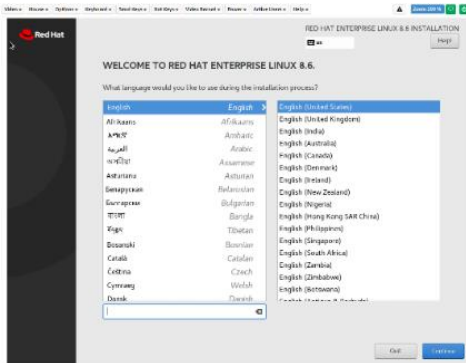
- システム管理者 PC の Web ブラウザが HTML5 に対応している場合は HTML5 を選択
- システム管理者 PC に Java をインストールしてある場合は JViewer(Java)を選択

設定後、ビデオリダイレクションを起動ください。

① ビデオリダイレクションボタン：クリックするとビデオリダイレクション機能が開始されます。

② この画面を出した状態で、主電源ボタンを押すと、BIOSの起動が開始されF2を押す事でBIOS画面が表示されます。
注意：バーチャルメディア機能を使う場合は、まだ主電源ボタンを押さないで下さい。

(B-7) ISO ファイルからのブートが成功すると OS インストール画面に移行します。移行後、手順に沿って設定を行います。



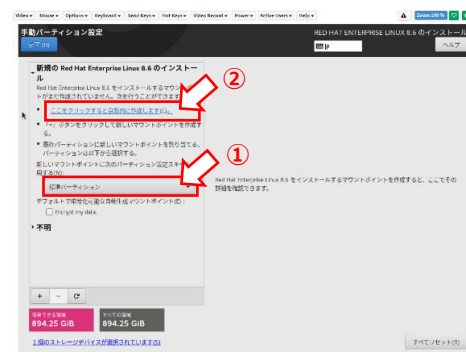
OS イメージのサイズ上限は 16GB のためパーティションを調整し、総サイズが 12GB 以下となるように設定します。



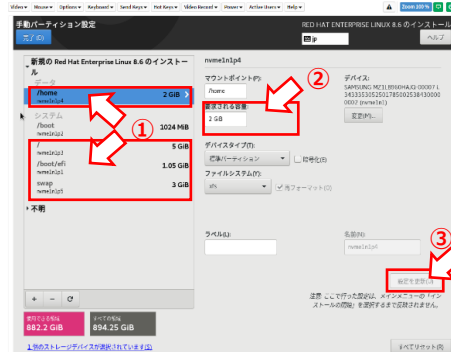
① 「インストール概要」で「インストール先」を選択します。



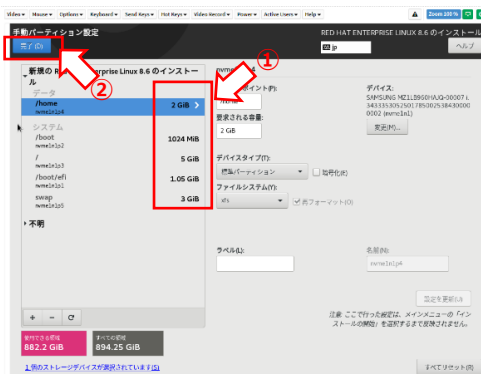
② 「デバイスの選択」で「ローカルの標準ディスク」を選択します。
 ③ 「ストレージの設定」で「カスタム」を選択します。
 ④完了を選択します。



① 「標準パーティション」を選択します。
 ② 「ここをクリックすると自動的に作成します」を選択します。



① /boot を除くパーティションのサイズを変更します。各パーティションを選択して下さい。
 ② 「要求される容量」にてサイズを変更してください。
 ③ 「設定を変更」を選択します。



①各パーティションサイズの合計が 12GB になるようにサイズ設定を行ってください。
 ②サイズ変更後に「完了」を押下します。「変更の概略」が表示された場合は「変更を許可する」を選択します。

その他以下の設定は指定がありません。システム管理者は任意のものを設定ください。

- ・ OS のホスト名、ログインユーザ名、ログインパスワード
- ・ ソフトウェアパッケージ
- ・ ネットワーク設定

(B-8) 設定完了後、ディスクへの OS インストールを開始します。インストールが完了後、OS をリブートして下さい。リブート後ログイン画面が表示されればインストールは完了です。

3.16.5 grub の設定

「3.16.4 ISO イメージのインストール」で使用したビデオリダイレクション上で 3.16.4 の (A-6)または (B-7) で設定した OS のユーザ名とパスワードで論理サーバにログインし、以下のコマンドを使用して grub の設定を行います。

```
# grub 設定を行います
$ sudo vim /etc/default/grub

# 以下の記述を変更下さい
GRUB_DEFAULT=saved
GRUB_SAVEDEFAULT=true
GRUB_CMDLINE_LINUX="pcie_aspm=off pcie_ports=native vt.handoff=1"
GRUB_CMDLINE_LINUX_DEFAULT="nvme_core.default_ps_max_latency_us=0"

# 記載後設定の更新を行います
$ sudo update-grub
```

設定後以下の shutdown コマンド、またはシステム構築手順書「G.2 主電源のオン・オフ機能」を参考に、PRIMERGY 主電源をオフ操作により論理サーバをシャットダウンしてください。

```
$ shutdown now
```

3.16.6 OS イメージの作成

「3.16.5 grub の設定」で使用したビデオリダイレクションを引き続き使用し、「3.16.4 ISO イメージのインストール」で NVMe SSD にインストールした OS から OS イメージを作成します。

- ① 3.16.4 の (A-3)~(A-5)または (B-4)を再度実施し、論理サーバの起動とリモートマウントした ISO ファイルからのブートを実施します。
- ② インストール設定画面で Ctl-Alt-F2 を押してターミナル画面に遷移します。
- ③ イメージファイルを保存するパーティションを作成します。

```
$ sudo fdisk /dev/nvme0n1
Command (m for help): n
Partition number (3-128, default 3):
First sector (...):
Last sector, ...):
Command (m for help): w
```

← 「n」を入力し「Enter」を押してください。
 ←何も入力せず「Enter」を押してください。
 ←何も入力せず「Enter」を押してください。
 ←何も入力せず「Enter」を押してください。
 ← 「w」を入力し「Enter」を押してください。設定が保存され終了します

- ④ パーティションをマウントします。

```
$ sudo mkfs.ext4 /dev/nvme0n1p3
$ sudo mount /dev/nvme0n1p3 /mnt
```

←nvme0n1p x の x は(3)で新たに作成したパーティション番号になります。

- ⑤ NVMe SSD にインストールした OS をコピーします。コマンドは 3.16.4 で論理サーバにインストールした OS のサイズを 5GB に設定した場合の例になります。

```
$ sudo dd if=/dev/nvme0n1 of=/mnt/OSimage.img bs=1G count=6 status=progress
6+0 records in
6+0 records out
6368709120 bytes (6.4 GB, 6.0 GiB) copied, 70 s, 76.4 MB/s
```

注) bs の値と count の値の積が常にインストールした OS のサイズよりも 1GB 以上大きい値になるように設定してください。

3.16.4 でインストールした OS のサイズが 8GB の場合、「bs=1G count=9」を設定

3.16.4 でインストールした OS のサイズが 12GB の場合、「bs=1G count=13」を設定

- ⑥ 以下の手順で OS をコピーします。

```
$ sudo gdisk /mnt/OSimage.img
```

```
Command (? for help): d
```

```
Partition number (1-X): X
```

```
Command (? for help): x
```

```
Expert command (? for help): e
```

```
Relocating backup data structures to the end of the disk
```

```
Expert command (? for help): w
```

```
Final checks complete. About to write GPT data. THIS WILL
```

```
OVERWRITE EXISTING PARTITIONS!!
```

```
Do you want to proceed? (Y/N): y
```

```
OK:
```

← 「d」を入力し「Enter」を押してください。

← 「X」を入力し「Enter」を押してください。

X は (3) で作成したパーティション番号になります

← 「x」を入力し「Enter」を押してください。

← 「e」を入力し「Enter」を押してください。

← 「w」を入力し「Enter」を押してください。

← 「y」を入力し「Enter」を押してください。

3.16.7 OS イメージをシステム管理者の PC に転送

「3.16.6 OS イメージの作成」で使用したビデオリダイレクションを引き続き使用し、3.16.6 で作成した OS イメージを Host OS に転送します。以下は scp コマンドを用いたコマンド例になります。

```
$ sudo scp /mnt/OSimage.img <Host OS のユーザ名>@<Host OS の IP Address>:/home/<Host OS のユーザ名>
>/cdi/RMcli<version>/RMcli<version>_<OSname>
<Host OS のユーザ名>@<Host OS の IP Address>'s password: パスワード (非表示)
```

転送終了後、論理サーバをシャットダウンします。以下の shutdown コマンド、またはシステム構築手順書

「G.2 主電源のオン・オフ機能」を参考に、PRIMERGY 主電源をオフ操作により論理サーバをシャットダウンしてください。

```
$ shutdown now
```

論理サーバのシャットダウン後、ビデオリダイレクションおよび iRMC の操作画面を閉じてください。

3.16.8 論理サーバの解体

システム管理者 PC から管理サーバの Host OS に接続し、「3.4 CDI 管理ソフトウェアにログインする」の手順で CDI 管理ソフトウェアにログインします。続いてイメージ作成用論理サーバを解体するため「2.2.9 論理サーバの解体」の操作を行います。

```
cdi: ./epcctl -u $URL machine destroy -m imgmach
```

3.17 論理サーバに OS イメージをインストールする

論理サーバに組み込まれた res_type が storage のリソース(NVMe SSD)に OS イメージをインストールするには以下に示す 3.17.1~3.17.5 の手順を実施してください。なお、インストールできる OS イメージの大きさは 16GB 以下になります。

3.17.1 使用できる OS イメージの確認

インストールする OS イメージ（システムに登録された OS イメージ）を確認するために「2.4.1 OS イメージの一覧を表示」の操作を行います。

※必要に応じて「2.4.2 OS イメージの詳細を表示」の操作で個々の OS イメージの詳細情報も確認してください。

※使用する OS イメージとして適当なものがないければ「2.4.3 OS イメージの登録」の操作でシステムに OS イメージを追加してください。

bootimg_uuid	bootimg_owner	bootimg_filename	bootimg_type
0627d38b-5d5e-49d4-8439-0a2d4810d55c	n/a	ubuntu18-rx1-20221004.img	img
12e8c1fa-6dd4-4359-b03e-58cd6cf7acc2	n/a	ubuntu18-rx1-20221001.img	img
43fb0b01-5e1d-413e-80a4-237f4eb3a809	n/a	CentOS-7-x86_64-DVD-2009.iso	iso
6db8095e-9ef9-4f0d-aa6c-9f8f7883a8e9	n/a	ubuntu18-rx1-20221003.img	img
a4ff88e4-6162-4e89-ae0a-d79f5008b13e	n/a	ubuntu-20.04.4-live-server-amd64.iso	iso
accf6ed9-d3c7-4c43-af7c-8324203135de	n/a	ubuntu18-rx1-20220615.img	img
fec12e74-421b-48f3-bf34-3075205a5444	n/a	ubuntu20-MLPerf1.img	img
d781ac42-c7ff-424b-982b-739bb6cdf05b	n/a	ubuntu-18.04.6-live-server-amd64.iso	iso

bootimg_uuid	bootimg_id	bootimg_date	bootimg_type
fec12e74-421b-48f3-bf34-3075205a5444	9	2022-12-19T12:52:12+00:00	img

3.17.2 インストールする OS イメージの選択

「3.17.1 使用できる OS イメージの確認」で確認した OS イメージから、インストールする OS イメージを選択します。

bootimg_uuid	bootimg_filename	bootimg_type
0627d38b-5d5e-49d4-8439-0a2d4810d55c	ubuntu18-rx1-20221004.img	img
12e8c1fa-6dd4-4359-b03e-58cd6cf7acc2	ubuntu18-rx1-20221001.img	img
43fb0b01-5e1d-413e-80a4-237f4eb3a809	CentOS-7-x86_64-DVD-2009.iso	iso
6db8095e-9ef9-4f0d-aa6c-9f8f7883a8e9	ubuntu18-rx1-20221003.img	img
a4ff88e4-6162-4e89-ae0a-d79f5008b13e	ubuntu-20.04.4-live-server-amd64.iso	iso
accf6ed9-d3c7-4c43-af7c-8324203135de	ubuntu18-rx1-20220615.img	img
fec12e74-421b-48f3-bf34-3075205a5444	ubuntu20-MLPerf1.img	img
d781ac42-c7ff-424b-982b-739bb6cdf05b	ubuntu-18.04.6-live-server-amd64.iso	iso

— 選択

3.17.3 インストール先の論理サーバを確認

インストール先の論理サーバを確認するために「2.2.1 論理サーバの一覧を表示」および「2.2.2 論理サーバの詳細を表示」の操作を行います。

- 2.2.1 の操作によりインストール先の論理サーバの mach_name を確認します。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0
machine1	2	7d6210ef-8a6a-403e-a980-5542b7852c30	user1

- 2.2.2 の操作によりインストール先の論理サーバに res_type が storage のリソースが組み込まれていることを確認します。

※論理サーバ(machine0)の詳細情報

item	value
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26
mach_id_nonliquid	
mach_id	
mach_name	machine0
resources	[{'res_uuid': '4f6f4cac-1074-4cf0-ac6b-da37d17642f0', 'res_name': 'ssd0', 'res_type': 'storage', 'res_status': 4, 'res_op_status': '0'}]

3.17.4 インストールを行うリソース(NVMe SSD)を選択

「2.3.5 論理サーバで使用されているリソースを表示」の操作により、論理サーバに組み込まれた res_type が storage のリソースからインストールを行うリソース(NVMe SSD)を選択します。

※必要に応じて「2.3.6 リソースの詳細を表示」の操作でリソース(NVMe SSD)の詳細情報も確認してください。

res_type	res_name	res_uuid	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1	8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498	None
gpu	gpu1	dcf272f4-c108-4a4c-be92-091c4725dabc	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebaef309	None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267	None

※NVMe SSD(ssd0)の詳細情報

item	value
res_uuid	2df31bfc-723e-4781-a83d-4372630c3498
res_id_nonliquid	
fabr_gid	0x033c00
res_type	gpu
res_name	ssd0

3.17.5 OS イメージをインストール

「3.17.4 インストールを行うリソース(NVMe SSD)を選択」で選択したリソース(NVMe SSD)に「3.17.4 インストールする OS イメージの選択」で選択した OS イメージをインストールするために「2.4.5 OS イメージのインストール」の操作を行います。

cdi: ./epcctl -u \$URL resource install **-r ssd0** **-b ubuntu18-rx1-20221004.img**

選択したリソース(NVMe SSD)の res_name
 選択した OS イメージの bootimg_filename

3.18 論理サーバへアクセスする

OS イメージをインストールした論理サーバにアクセスするには、以下に示す 3.18.1~3.18.9 の手順を実施してください。

※OS のインストールは「3.17 論理サーバに OS イメージをインストールする」の操作により行います。

3.18.1 論理サーバの確認

アクセスする論理サーバを確認するために「2.2.1 論理サーバの一覧を表示」の操作を行います。
 ※必要に応じて「2.2.2 論理サーバの詳細を表示」の操作により論理サーバの詳細情報も確認してください。

mach_name	mach_id	mach_uuid	mach_owner
machine0	1	b9e25b57-521a-4047-b2cd-6159c452ac26	user0

↓ 論理サーバ(machine0)の詳細情報

item	value
mach_uuid	b9e25b57-521a-4047-b2cd-6159c452ac26
mach_id_nonliquid	
mach_id	
mach_name	machine0
...	...

3.18.2 論理サーバの電源オン

「3.18.1 論理サーバの確認」確認した論理サーバの電源が投入されていなければ「2.2.10 論理サーバの電源オン」を行います。

```
cdi: ./epcctl -u $URL machine power -P on -m machine0 ← 確認した論理サーバの mach_name
```

3.18.3 ステータスの確認

アクセスする論理サーバの詳細ステータス(mach_status_detail)が“ACTIVE PON”であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

※ “ACTIVE PON”以外が表示された場合は、“ACTIVE PON”が表示されるまで確認を繰り返します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 13,
"mach_status_detail": " ACTIVE PON",
...
```

3.18.4 リソースの確認

論理サーバに存在するリソースを確認するために「2.3.5 論理サーバで使用されているリソースを表示」の操作を行います。

※res_type が compute のリソース (計算サーバ) の res_name を確認します。

res_type	res_name	res_uuid	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498	...	None
storage	ssd0	8e2abd0a-61a6-4b44-87f2-648aa0a0031a		None

3.18.5 iRMC 情報の確認

「3.18.4 リソースの確認」で確認した計算サーバの iRMC 情報を確認するために「2.6.2 iRMC 情報の表示（計算サーバ）」の操作を行います。

```
cdi: ./epcctl -u $URL resource bmc -r pcpu0 -o LIST
200
```

計算サーバの res_name

res_uuid_cpunode	cpunode_ipmi_address	cpunode_ipmi_userid	cpunode_ipmi_password
aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee	AAA. BBB. C. DDD	xxxxxxx	*****

3.18.6 iRMC ヘログイン

「3.18.4 iRMC 情報の確認」で確認した iRMC 情報を使用して、システム管理者 PC で計算サーバの iRMC にログインするために、システム構築手順書「G.1 iRMC へのログイン」の操作を行います。

3.18.7 ビデオリダイレクションの起動

ビデオリダイレクションを起動するために、システム構築手順書「G.4 ビデオリダイレクション機能」の操作を行います。

3.18.8 IP アドレスの確認

アクセスする論理サーバの IP アドレスを確認するために、「3.18.7 ビデオリダイレクションの起動」で起動したビデオリダイレクションのコンソール画面からログイン(※1)し、論理サーバのコンソールで下記のコマンドにより IP アドレスを確認します。

※1 ユーザ ID とパスワードは「3.16.4 ISO イメージのインストール」の (A-6)または (B-7)で設定したものを
使用します。

```
$ ip addr
```

なお、IP アドレスが設定されていない場合は、論理サーバにインストールされている OS に合わせた手順により手動で IP アドレスを設定してください。

3.18.9 論理サーバへのアクセス

「3.18.8 IP アドレスの確認」で確認した IP アドレスを使用して、論理サーバにアクセスします。

3.19 データを削除する

res_type が storage のリソース(NVMe SSD)のデータを削除するには、以下に示す 3.19.1~3.19.3 の手順を実施してください。本手順によって SSD 上の全てのデータを完全に削除することができます。

計算サーバに組み込まれた SSD を論理的に取り外す場合や計算サーバを解体する場合は SSD の内容は自動的に消去されますが、SSD カードを交換する場合などで SSD の内容を明示的に消去する場合に実施してください。

注) OS が動作中の計算サーバの SSD を消去する場合は、計算サーバを停止するか対象 SSD のマウントを解除してください。また OS がインストールされている SSD を消去すると OS の再インストールが必要となります。

3.19.1 データを削除するリソース (NVMe SSD) の確認

データを削除する res_type が storage のリソース(NVMe SSD)を確認するために「2.3.1 リソースの一覧を表示」の操作を行います。

※必要に応じて「2.3.6 リソースの詳細を表示」の操作によりリソースの詳細情報も確認してください。

res_type	res_name	res_uuid	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		8
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498		None
gpu	gpu1	5b30e05a-ef22-4349-b4fd-5cdb45029a2	...	None
gpu	gpu2	5b30e05a-ef22-4349-b4fd-5cdb45029a2		None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309		None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267		None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490		None

item	value
res_uuid	2df31bfc-723e-4781-a83d-4372630c3498
res_id_nonliquid	
fabr_gid	0x033c00
res_type	gpu
res_name	ssd0
...	...

3.19.2 データを削除するリソース (NVMe SSD) の選択

「3.19.1 データを削除するリソース (NVMe SSD) の確認」で確認したリソースからデータを削除するリソースを選択します。

res_type	res_name	res_uuid	...	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1		8
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8		8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498		None
gpu	gpu1	5b30e05a-ef22-4349-b4fd-5cdb45029a2	...	None
gpu	gpu2	5b30e05a-ef22-4349-b4fd-5cdb45029a2		None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebafef309		None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267		None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490		None

3.19.3 データを削除

「3.19.2 データを削除するリソース (NVMe SSD) の選択」で選択したリソースからデータを削除するために「2.6.1 データの削除」の操作を行います。

cdi: ./epcctl -u \$URL resource erase -r **ssd0** ← 選択したリソース(NVMe SSD)の res_name

3.20 グループ・論理サーバを再構成する

構成バックアップファイル（グループ構成と論理サーバ構成が保存された圧縮ファイル）を使用してグループ・論理サーバを再構成するには、以下に示す 3.20.1～3.20.7 の手順を実施してください。

3.20.1 使用できる構成バックアップファイルの確認

「2.5.1 構成バックアップファイルの一覧を表示」の操作により、保存されている構成バックアップファイルを確認します。

```
'dumpfiles': [
  'dumpfile_20230516013915431141.tar.gz',
  'dumpfile_20230516014016816701.tar.gz',
  'dumpfile_20230516014215763132.tar.gz',
  'dumpfile_20230516014352720493.tar.gz',
  'dumpfile_20230516014503466356.tar.gz',
  'dumpfile_20230516014631463656.tar.gz',
  'dumpfile_20230516020212656352.tar.gz',
  'dumpfile_20230516020345644777.tar.gz',
  'dumpfile_20230516020723381957.tar.gz',
  'dumpfile_20230516020845811055.tar.gz'
```

3.20.2 使用する構成バックアップファイルの選択

「3.20.1 使用できる構成バックアップファイルの確認」で確認した構成バックアップファイルから再構成に使用するファイルを選択します。

※ファイル名に記載されたタイムスタンプはバックアップ取得時間になります。ファイル名のタイムスタンプを参考にバックアップファイルを選択してください。（最新のファイルが一番下に表示されています）

```
'dumpfiles': [
  'dumpfile_20230516013915431141.tar.gz',
  'dumpfile_20230516014016816701.tar.gz',
  'dumpfile_20230516014215763132.tar.gz',
  'dumpfile_20230516014352720493.tar.gz',
  'dumpfile_20230516014503466356.tar.gz',
  'dumpfile_20230516014631463656.tar.gz',
  'dumpfile_20230516020212656352.tar.gz',
  'dumpfile_20230516020345644777.tar.gz',
  'dumpfile_20230516020723381957.tar.gz',
  'dumpfile_20230516020845811055.tar.gz' ← 選択
```

3.20.3 構成バックアップファイルのダウンロード

「3.20.2 使用できる構成バックアップファイルの選択」で選択した構成バックアップファイルをダウンロードするために「2.5.2 構成バックアップファイルをダウンロード」の操作を行います。

```
cdi: ./epcctl -u $URL system dump -F dumpfile_20230516020845811055.tar.gz ← 選択したファイル名
200
dumpfile_20230516020845811055.tar.gz
```

3.20.4 CDI 管理ソフトウェアからログアウト

システムログ採取後、CDI 管理ソフトウェアからログアウトします。

```
cdi: exit
$
```

3.20.5 構成バックアップファイルの解凍

以下のコマンドでダウンロードした構成バックアップファイルを解凍します。

```
$ tar xzvf dumpfile_20230516020845811055.tar.gz
```

※解凍により以下のファイルが展開されます。

- ・ g_{grp_name}.yaml
グループ再構成に使用する yaml ファイルです。{grp_name}にはグループ名が入ります。
- ・ m_{grp_name}_{mach_name}.yaml
論理サーバ再構成に使用する yaml ファイルです。{grp_name}には論理サーバのグループ名、{mach_name}には論理サーバ名が入ります。
- ・ script.sh
グループ・論理サーバを再構成するためのコマンドリストを記述したファイルです。

解凍した構成バックアップ用の yaml ファイルは、cp コマンド等で epccctl の実行コマンドのあるパスに移しておいてください。以下は一例になります。構成バックアップファイルのディレクトリ名、バックアップ用の yaml ファイル名、epccctl の実行コマンドのあるパスは実際の環境に合わせて変更して下さい。

```
$ cp ./dumpfile_20230516020845811055/g_{grp_name}.yaml /home/<管理サーバのユーザ名>/<cdi/RMcli<version>/RMcli<version>_<OSname>
$ cp ./dumpfile_20230516020845811055/m_{grp_name}_{mach_name}.yaml /home/<管理サーバのユーザ名>/<cdi/RMcli<version>/RMcli<version>_<OSname>
```

3.20.6 グループ・論理サーバの再構成

システム管理者 PC から管理サーバの Host OS に接続し、「3.4 CDI 管理ソフトウェアにログインする」の手順で CDI 管理ソフトウェアに再度ログインします。次に、コマンドリストファイル(script.sh)に記載されたコマンドを書き換え、2行目から順に実行していくことでバックアップを行った時のグループと論理サーバが再構成されます。

○script.sh (例)

```
# url=http://localhost:8000/resource_manager/api/v1
# ./epccctl -u $url group create -y /var/opt/epc/resource_manager/dump/g_group0.yaml
# ./epccctl -u $url machine create -y /var/opt/epc/resource_manager/dump/m_group0_machine0.yaml
# ./epccctl -u $url machine create -y /var/opt/epc/resource_manager/dump/m_group0_machine1.yaml
```

○script.sh に記載されているコマンドを以下の様に変更して 2 行目から実行

- ・ 先頭の"#"を削除
- ・ "\$url"を"\$URL"に置換
- ・ バックアップ用 yaml ファイルのパス"/var/opt/epc/resource_manager/dump/"を、3.20.5 章で epccctl と同パスに置いた場合、"/var/opt/epc/resource_manager/dump/"を削除 (epccctl と同パスに置いてない場合、yaml ファイルが置いてあるパス (例: "dumpfile_20230516020845811055/") に変更)

○実行例 (3.20.5 章で yaml ファイルを epccctl と同パスに置いた場合)

```
cdi: ./epccctl -u $URL group create -y g_group0.yaml ← group0 (グループ) を再構成
200
{xxxxx}
cdi: ./epccctl -u $URL machine create -y m_group0_machine0.yaml ← group0 の group0_machine0
200 (論理サーバ) を再構成
{xxxxx}
cdi: ./epccctl -u $URL machine create -y m_group0_machine1.yaml ← group0 の group0_machine1
200 (論理サーバ) を再構成
{xxxxx}
```

3.20.7 論理サーバのステータスを確認

「2.2.12 論理サーバの状態確認」の操作により再構成した全ての論理サーバの詳細ステータス(mach_status_detail)が“INACTIVE POFF”であることを確認します。

※ 再構成を行った個々の論理サーバについて詳細ステータスを確認します。

※ “INACTIVE POFF”以外が表示された場合は、“INACTIVE POFF”が表示されるまで確認を繰り返します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",  
"mach_status": 15,  
"mach_status_detail": " INACTIVE POFF",  
"power_status": "unknown",  
"lanports": null
```

第4章 保守操作

本章では、以下の保守時に行われる操作について説明します。詳細については、該当する章を参照してください。

- ハードウェアの増設・減設・交換 (4.1)
- ソフトウェアのアップデート (4.2)
- 設定情報の管理 (4.3)

4.1 ハードウェアの増設・減設・交換をする

ここでは、ハードウェアの増設・減設・交換作業について説明します。実施できる作業は、増設・減設・交換後に構成バックアップファイルから既存のグループ・論理サーバを再構成（過去の論理サーバ構成を復元）するかどうかで異なります。以下で各々についての流れを示します。これらの作業は、システム運用管理者と保守技術員が手順を分担して行います。

4.1.1 グループ・論理サーバの再構成を行わない場合

増設・減設・交換作業を実施後に、作業前のグループ・論理サーバの再構成を行う必要が無い場合には、以下の①～⑤の手順で作業を実施し、その後、必要に応じてグループ・論理サーバの作成を行います。

- ①全論理サーバの利用停止 **システム運用管理者作業**
- ②全論理サーバの電源オフ **システム運用管理者作業**
全論理サーバの電源状態を確認し、電源オンの論理サーバがあればオフします。
論理サーバの OS 上から shutdown が実行不可となる場合は、iRMC の Web 画面より電源オフを行います。
→ システム構築手順書 G.2「主電源のオン・オフ機能」参照
- ③ハードウェアの増設・減設・交換 **保守技術員作業**
システム構築手順書の「第 6 章 ハードウェアの増設・減設・交換手順」を実施します。
- ④ CDI 管理ソフトウェアの削除（管理サーバの交換保守の場合は不要） **システム運用管理者作業**
「4.2.1 CDI 管理ソフトウェアのアップデート」(3)の手順で CDI 管理ソフトウェアを削除します。
- ⑤ CDI 管理ソフトウェアのセットアップ **システム運用管理者作業**
システム構築手順書の「第 5 章 CDI 管理ソフトウェアのセットアップ」を実施します。
ただし、増設・減設・交換する対象別に手順の一部の要・不要が分かれます。下表を参照ください。

	計算サーバ	計算サーバ	計算サーバ	管理サーバ	Director
	増設	減設	交換	交換	交換
5.1.2. 計算サーバ証明書を作成する	要	不要	要	不要	不要
5.1.3 Director の HTTPS 設定および証明書を作成する	不要	不要	不要	不要	要
5.1.4 Host OS から Guest OS へのアクセス設定を行う	不要	不要	不要	要	不要
5.1.5 CDI 管理ソフトウェアの環境設定を行う	不要	不要	不要	要	不要
5.1.6 CDI 管理ソフトウェアに証明書を登録する(計算サーバ証明書の登録)	対象サーバ：要 他：不要	不要	対象サーバ：要 他：不要	不要	不要
5.1.6 CDI 管理ソフトウェアに証明書を登録する(Director 証明書の登録)	不要	不要	不要	要	要
5.1.7 CDI 管理ソフトウェアの証明書を作成する	不要	不要	不要	要	不要
5.1.8 スペックファイルを編集する(3)~(5)	要	要	要	要	不要
5.1.8 スペックファイルを編集する(6)~(8)	※	※	※	要	要
5.1.9 CDI 管理ソフトウェアにログインする	要	要	要	要	要
5.1.10 リソーススペックテーブルを作成する	要	要	要	要	不要
5.1.11 PCIe BOX のスペックテーブルを作成する	不要	不要	不要	要	不要
5.1.12 PCIe Box テーブルを作成する	※	※	※	要	要
5.1.13 リソース情報を同期させる	要	要	要	要	要
5.1.14 SSD 監視用 OS イメージを登録する	不要	不要	不要	不要	要
5.1.15 計算サーバの iRMC 情報を登録する	要	要	要	要	不要

※CDI 構成パターン名の変更有りの場合：要、無い場合：不要

4.1.2 グループ・論理サーバの再構成を行う場合

増設・減設・交換作業を実施後に、作業前のグループ・論理サーバの再構成を行うことが必要な場合でのコンポーネント毎の増設/減設/交換の可・不可を下表に示します。ただし、ここでの増設・減設・交換とは、既存のコンポーネントの結線状態や構成設定の変更が無い場合に限られます。また交換は、スペックが同一のコンポーネント同士の交換に限られます。

対象 コンポーネント	増設 (参照章)	減設 (参照章)	交換 (参照章)	備考
Director	不可	不可	不可	
Fabric Switch	不可	不可	可 (4.1.2.1 章)	交換は、同スペックかつケーブル接続は変えないことが前提です。
PCIe Box	不可	不可	可 (4.1.2.1 章)	交換は、同スペックかつケーブル接続や搭載カードは変えないことが前提です。
PCIe カード	可能 (4.1.2.1 章)	可能 (4.1.2.1 章)	可能 (4.1.2.1 章)	
計算サーバ	可能 (4.1.2.3 章)	可能 (4.1.2.4 章)	可能 (4.1.2.2 章)	交換は計算サーバの部品単位の交換を意味し搭載しているカード類（例：HBA カード）も含まれます。
管理サーバ	不可	不可	不可	
管理 LAN スイッチ	不可	不可	可能	システム構築手順書の「第 6 章 ハードウェアの増設・減設・交換手順」を実施します
PDU	可能	可能	可能	同上
業務 LAN スイッチ	任意	任意	任意	同上

4.1.2.1 PCIe カード・Fabric Switch・PCIe BOX の増設・減設・交換保守手順

PCIe カード・Fabric Switch・PCIe BOX の増設・減設・交換保守は、下記の手順を実施してください。

- ①全論理サーバの利用停止 **システム運用管理者作業**
論理サーバ利用者に使用停止および、各自任意でのデータのバックアップ実施を通知してください。
- ②全論理サーバの電源オフ **システム運用管理者作業**
全論理サーバの電源状態を確認し、電源オンの論理サーバがあればオフします。
論理サーバの OS 上から shutdown が実行不可となる場合は、iRMC の Web 画面より電源オフを行います。
→システム構築手順書 G.2「主電源のオン・オフ機能」参照
- ③（減設の場合のみ）対象リソースを論理サーバから削除 **システム運用管理者作業**
対象のリソースが論理サーバで使用されている場合は、論理サーバから対象リソースを削除します。
 - ・対象のリソースの詳細情報を表示し、mach_uuid を確認する。
mach_uuid が空白の場合は論理サーバで使用されていないため削除手順は不要です。
→「2.3.6 リソースの詳細を表示」参照
 - ・mach_uuid を用いて論理サーバ詳細情報を表示し、論理サーバ名を確認する。
→「2.2.2 論理サーバの詳細を表示」参照
 - ・論理サーバから対象リソースを削除する
→「2.2.7 論理サーバからのリソース削除（コマンド指定）」参照
- ④（減設の場合のみ）対象リソースをグループから削除 **システム運用管理者作業**
対象のリソースがグループに追加されている場合は、グループから対象リソースを削除します。
 - ・対象のリソースの詳細情報を表示し、grp_uuid を確認する。
grp_uuid が空白の場合はグループに追加されていないため削除手順は不要です。
→「2.3.6 リソースの詳細を表示」参照
 - ・grp_uuid を用いてグループから対象リソースを削除する
→「2.1.7 グループからのリソース削除（コマンド指定）」参照
- ⑤対象リソースの増設・減設・交換保守作業 **保守技術員作業**
PCIe カードの場合、以下のように対象リソースの設置場所を確認し、増設・減設・交換保守作業を行います。
 - ・対象リソースの設置場所（JBOX が設置されているラック名・ラック位置、保守対象リソースが挿されている JBOX 内のスロット番号）を表示する
→「2.3.6 リソースの詳細を表示」参照
 - ・対象リソースのスロット位置を確認し、増設・減設・交換保守を実施する
→システム構築手順書「6.6.2 PCIe カードの増設/減設/交換手順」参照
 Fabric Switch・PCIe BOX の交換の場合、以下の手順で保守作業を行います。
→システム構築手順書「6.6.4 Fabric Switch/PCIe Box の交換手順」参照
- ⑥リソース情報を同期する **システム運用管理者作業**
「3.4 CDI 管理ソフトウェアにログインする」を実施した後に、「2.3.10 リソース情報の同期」を実施します。
- ⑦計算サーバの iRMC 情報を再登録する **システム運用管理者作業**
CDI 管理ソフトウェアに対し計算サーバの iRMC のユーザ名・パスワードの再登録を実施します。操作手順はシステム構築手順書「5.1.15 計算サーバの iRMC 情報を登録する」を参照してください。
- ⑧グループ・論理サーバの再構成 **システム運用管理者作業**
自動保存されている構成バックアップファイルを使用して、グループ、論理サーバ構成を再構築します。
→「3.20 グループ・論理サーバを再構成する」参照
- ⑨増設したリソースをグループに追加 **システム運用管理者作業**
保守交換・増設したリソースはファブリックフリーブルにあるためグループに追加します。
→「2.1.5 グループへのリソース追加（コマンド指定）」参照

- ⑩増設したリソースを論理サーバに追加 **システム運用管理者作業**
→「2.2.5 論理サーバへのリソース追加（コマンド指定）」参照
- ⑪ログ情報を採取保存 **システム運用管理者作業**
→「システム構築手順書「5.1.16 ログ情報を採取して保存しておく」参照
- ⑫論理サーバの電源オン **システム運用管理者作業**
→「2.2.10 論理サーバの電源オン」参照

4.1.2.2 計算サーバの部品交換保守手順

計算サーバの部品交換保守は、下記の手順を実施してください。

- ①対象論理サーバの電源状態を確認し、電源オン状態の場合は電源オフします。 **システム運用管理者作業**
→「2.2.12 論理サーバの状態確認」「2.2.11 論理サーバの電源オフ」参照
- ②対象リソースを交換します。 **保守技術員作業**
→システム構築手順書「6.6.5 計算サーバの交換」参照
ただし、システム構築手順書「6.6.5 計算サーバの交換」は異なる機種への交換も含むと記載されていますが、本手順では異なる機種への交換は対象外です。
また、Director の電源をオフにしない状態での交換を前提とします。
また、手順番号 step0「計算サーバの交換が出来る事の確認」の作業は実施不要です。
- ③計算サーバ証明書を生成します。 **システム運用管理者作業**
→システム構築手順書「5.1.2 計算サーバ証明書を生成する」参照
- ④CDI 管理ソフトウェアにサーバ証明書を登録します。 **システム運用管理者作業**
→システム構築手順書「5.1.6 CDI 管理ソフトウェアに証明書を登録する」参照
ただし、システム構築手順書「5.1.6 CDI 管理ソフトウェアに証明書を登録する」にはすべての計算サーバ、Director の証明書登録をすると記載されていますが、本手順では保守交換した計算サーバ以外の証明書登録は実施不要です。
- ⑤ログ情報を採取保存 **システム運用管理者作業**
→システム構築手順書「5.1.16 ログ情報を採取して保存しておく」参照
- ⑥対象論理サーバの電源をオンします。 **システム運用管理者作業**
→「2.2.10 論理サーバの電源オン」参照

4.1.2.3 計算サーバの増設

計算サーバを増設するには、下記の手順を実施してください。

- ①全論理サーバの利用停止 **システム運用管理者作業**
- ②全論理サーバの電源オフ **システム運用管理者作業**
全論理サーバの電源状態を確認し、電源オン状態の場合は電源オフします。
- ③増設する計算サーバ（ハードウェア）のセットアップ **保守技術員作業**
システム構築手順書「6.6.3 計算サーバの増設/減設」を参照
- ④スペックファイルを編集する **システム運用管理者作業**
システム構築手順書「5.1.8 スペックファイルを編集する」(1)~(5)を参照。
ここで編集する yml ファイルは、手順 (3) においてシステム構築手順書「6.6.3 計算サーバの増設/減設」の追加手順 1「4.4.4 HOST IPMI の設定」で Director に設定した計算サーバの論理 CPU 名に基づいて選択します。
- ⑤CDI 管理ソフトウェアにログインする **システム運用管理者作業**
システム構築手順書「5.1.9 CDI 管理ソフトウェアにログインする」を参照。
- ⑥増設された計算サーバのリソーススペックテーブルを作成 **システム運用管理者作業**
システム構築手順書「5.1.10 リソース スペック テーブルを作成する」を参照。
ここで yml ファイルは、手順 (4) で編集した yml ファイルのみを使用します。
- ⑦増設された計算サーバのリソース情報を同期 **システム運用管理者作業**
「2.3.10 リソース情報の同期」を参照。
- ⑧増設された計算サーバの iRMC 情報を登録 **システム運用管理者作業**
システム構築手順書「5.1.15 計算サーバの iRMC 情報を登録する」の(2)を参照。
ただし、システム構築手順書「5.1.15 計算サーバの iRMC 情報を登録する」の(2)にはすべての計算サーバに対して実施すると記載されていますが、本手順では増設した計算サーバに対してのみ実施します。
ここで指定するリソース名称は手順 (3) においてシステム構築手順書「6.6.3 計算サーバの増設/減設」の追加手順 1「4.4.4 HOST IPMI の設定」で Director に追加設定した計算サーバの論理 CPU 名を使用します。
- ⑨増設された計算サーバのサーバ証明書を取得・設定・登録 **システム運用管理者作業**
システム構築手順書「5.1.2 計算サーバ証明書を生成する」を参照し、増設された計算サーバのサーバ証明書を生成します。本手順では増設された計算サーバ以外の計算サーバに対するサーバ証明書の生成は不要です。次に、システム構築手順書「5.1.6 CDI 管理ソフトウェアに証明書を登録する」を参照し、CDI 管理ソフトウェアに増設された計算サーバのサーバ証明書を登録します。本手順では増設された計算サーバ以外の計算サーバおよび Director の証明書登録は不要です。
- ⑩ログ情報を採取保存 **システム運用管理者作業**
→ システム構築手順書「5.1.16 ログ情報を採取して保存しておく」参照
- ⑪グループ・論理サーバの再構成 **システム運用管理者作業**
「3.20 グループ・論理サーバを再構成する」を参照
- ⑫論理サーバの電源オン **システム運用管理者作業**
→ 「2.2.10 論理サーバの電源オン」参照

4.1.2.4 計算サーバの減設

計算サーバを減設するには、下記の手順を実施してください。

- ①全論理サーバの利用停止 **システム運用管理者作業**
- ②全論理サーバの電源オフ **システム運用管理者作業**
全論理サーバの電源状態を確認し、電源オン状態の場合は電源オフします。
- ③減設する計算サーバの論理サーバを解体 **システム運用管理者作業**
減設する計算サーバで構成された論理サーバを解体
- ④リソースをグループから削除 **システム運用管理者作業**
③で解体したリソースをグループから削除
- ⑤減設する計算サーバ（ハードウェア）の取り外し **保守技術員作業**
システム構築手順書「6.6.3. 計算サーバの増設/減設手順」を参照
ただし、手順番号：前手順1「減設の場合、計算サーバの減設が出来る事の確認」の手順は不要です。
- ⑥減設する計算サーバのリソース情報をリソーステーブルから削除 **システム運用管理者作業**
・減設する計算サーバのリソース名を指定してリソーステーブルから削除する。

```
cdi: ./epcctl -u $URL resource remove -r <リソース名>
```
- ⑦減設する計算サーバのスペック情報をスペックテーブルから削除 **システム運用管理者作業**
・減設する計算サーバのスペックテーブル上の UUID を確認する。

```
cdi: ./epcctl -u $URL spec list
```


・減設する計算サーバのスペックの UUID を指定してスペックテーブルから削除する。

```
cdi: ./epcctl -u $URL spec remove -u UUID
```
- ⑧リソース情報を再取得する **システム運用管理者作業**
→「2.3.10 リソース情報の同期」参照
- ⑨ログ情報を採取保存 **システム運用管理者作業**
→システム構築手順書「5.1.16 ログ情報を採取して保存しておく」参照
- ⑩グループ・論理サーバの再構成 **システム運用管理者作業**
「3.20 グループ・論理サーバを再構成する」を参照
- ⑪論理サーバの電源オン **システム運用管理者作業**
→「2.2.10 論理サーバの電源オン」参照

4.2 ソフトウェアをアップデートする

ここでは、CDI 管理ソフトウェア、Director ファームウェア、PCIe Box 内部の NIC ファームウェアのアップデートについて記載します。

4.2.1 CDI 管理ソフトウェアのアップデート

CDI 管理ソフトウェアをアップデートするには下記①～⑦を実施してください。

- ①論理サーバの利用停止を通知
 - ・論理サーバの電源オフが行われるため、全ての利用者に対して利用停止を通知します。必要であればデータのバックアップの実施も通知してください。
- ②論理サーバの電源をオフ
 - ・グループに存在する全ての論理サーバの電源をオフするために「3.2.3 論理サーバの電源をオフ」の操作を行います。
- ③ダンプファイルの取得
 - ・CDI 管理ソフトウェアに保存されている構成バックアップファイルをダウンロードするために「3.20.1 使用できる構成バックアップファイルを確認」、「3.20.2 使用できる構成バックアップファイルの選択」、および「3.20.3 構成バックアップファイルのダウンロード」の操作を行います。
※ダンプファイルはタイムスタンプが最新のものを使用ください。
- ④管理サーバの iRMC にログイン
 - ・システム管理者 PC で管理サーバの iRMC にログインするために、システム構築手順書「G.1 iRMC へのログイン」の操作を行います。
- ⑤ビデオリダイレクションの起動
 - ・ビデオリダイレクションを起動するために、システム構築手順書「G.4 ビデオリダイレクション機能」の操作を行います。
- ⑥仮想マシンマネージャの起動
 - ・アプリケーションの一覧から“仮想マシンマネージャ”のアイコンを選択し、仮想マシンマネージャを起動します。



- ⑦管理サーバの Guest OS の削除
 - ・登録している仮想マシン “RMimg_<version>” を選択し、メニューの[削除]により削除します。



⑧ CDI 管理ソフトウェアのイメージファイルを削除

- ・以下のコマンドを実行して、使用していた CDI 管理ソフトウェアのイメージファイルを削除します。

```
$ sudo rm /var/lib/libvirt/images/RMimg_<version>.qcow2  
$ sudo rm /etc/libvirt/qemu/RMimg_<version>.xml
```

⑨ CDI 管理ソフトウェアの再準備

- ・アップデートで使用する CDI 管理ソフトウェアを準備するために、システム構築手順書「4.3.3 作業用 PC から管理サーバへのファイル転送」から「4.3.9 CDI ツールキットの確認」までを実施します。

⑩ CDI 管理ソフトウェアのアップデート

- ・CDI 管理ソフトウェアをアップデートするために、システム構築手順書「5 ソフトウェアのセットアップ」を実施します。

⑪ グループ、論理サーバ構成の再構築

- ・⑧で Director のリセットを実施した場合は、構成バックアップファイルを使用してグループ、論理サーバ構成を再構築します。③で取得したダンプファイルに対して、「3.20.5 構成バックアップファイルの解凍」、「3.20.6 グループ・論理サーバの再構成」、および「3.20.7 論理サーバのステータスを確認」の操作を行います。Director のリセットを実施していない場合本操作は不要です。

⑫ 論理サーバの電源オン

- ・グループに存在する全ての論理サーバの電源をオンするために「3.1.8 論理サーバの電源をオン」の操作を行います。

4.2.2 Director ファームウェアのアップデート

Director ファームウェアをアップデートするには下記①～③を実施してください。

①論理サーバの利用停止を通知

- ・論理サーバの電源オフが行われるため、全ての利用者に対して利用停止を通知します。必要であればデータのバックアップの実施も通知してください。

②論理サーバの電源をオフ

- ・グループに存在する全ての論理サーバの電源をオフするために「3.2.3 論理サーバの電源をオフ」の操作を行います。

③ダンプファイルの取得

- ・CDI 管理ソフトウェアに保存されている構成バックアップファイルをダウンロードするために「3.20.1 使用できる構成バックアップファイルを確認」、「3.20.2 使用できる構成バックアップファイルの選択」、および「3.20.3 構成バックアップファイルのダウンロード」の操作を行います。
※ダンプファイルはタイムスタンプが最新のものを使用ください。

④ファームウェアのアップロード

- ・Director にファームウェアをアップロードするために「2.6.6 ファームウェアファイル (Director 用) のアップロード」の操作を行います。管理サーバの Guest OS(仮想マシン)にログインし、以下のコマンドを実行してください。ログイン情報については構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照してください。

```
$ /opt/epc/bin/fabrctl upload -u https://Director の IP アドレス -f director-xxx.upr -c $LIQID_VERIFY_CA/Director の IP アドレス.pem
Uploading liqid-xxx.upr
Success
```

※Director の IP アドレスについては、システム構築手順書を参照してください。(以下同様)

⑤ファームウェアの一覧を表示

- ・Director にアップロードされているファームウェアの一覧を表示するために「2.6.7 ファームウェアファイル (Director 用) の一覧を表示」の操作を行います。

```
$ /opt/epc/bin/fabrctl upload -u https://Director の IP アドレス -c $LIQID_VERIFY_CA/Director の IP アドレス.pem
Index      Filename
-----
0          director-2.5.1.36-2.upr
1          director-3.0.0.106.upr
2          director-xxx.upr
```

※ アップデート実行時にファイルの Index 番号を使用するため、対象ファイルの Index 番号を記録しておきます。

⑥ファームウェアのアップデート

- ・Director にアップロードされているファームウェアを使用してアップデートするために「2.6.8 Director ファームウェアのアップデート」の操作を行います。

```
$ /opt/epc/bin/fabrctl upgrade -u https://Director の IP アドレス L -i 2 -c $LIQID_VERIFY_CA/Director の IP アドレス.pem
Are you sure you want to upgrade? [y/N]: y ← y を入力
Upgrading liqid-xxx
Success
```

注) 所要時間は 45 分程度です。アップデート実施中は system シャットダウンの操作は行わないでください。

注) メジャーアップデート (3.2 から 3.3 へのアップデート等) の場合は、-w オプション (ソフトウェア管理者ガイド参照) を指定してください。

⑦アップデートの確認

- ・所要時間が経過したところで、アップデートの確認を行うために「2.6.5 Director の状態表示」の操作を行います。

```
$ /opt/epc/bin/fabrctl state -u https://Director の IP アドレス -c $LIQID_VERIFY_CA/Director の IP アドレス.pem
```

Success ←Success ならば正常状態

⑧Director のリセット

- ・⑦で Error が表示される場合は Director のリセットするために「2.6.4 Director のリセット」の操作を行います。

```
$ /opt/epc/bin/fabrctl state -u https://Director の IP アドレス -c $LIQID_VERIFY_CA/Director の IP アドレス.pem
Error: LiquidDegradedSteps: Should call 'reset' ←Error 表示
$ /opt/epc/bin/fabrctl reset -u https://Director の IP アドレス -c $LIQID_VERIFY_CA/Director の IP アドレス.pem
Success
```

⑨アップデートの完了を確認

- ・Director にアップロードされているファームウェアを再確認するために「2.6.7 ファームウェアファイル (Director 用) の一覧を表示」の操作を行います。

※アップデートで使用したファイルがリストより消えていれば、アップデートが完了しています。

```
$ /opt/epc/bin/fabrctl upload -u https://Director の IP アドレス -c $LIQID_VERIFY_CA/Director の IP アドレス.pem
Index      Filename
-----
0          director-2.5.1.36-2.upr
1          director-3.0.0.106.upr
```

⑩Director の電源オフオン

- ・以下の手順を実行して、Director の電源を OFF-ON してください。
システム構築手順書「4.9.2 電源切断手順」を参照して、PCIe Box、Fabric Switch、Director の電源を切断してください。
システム構築手順書「4.9.1 電源投入手順」を参照して、PCIe Box、Fabric Switch、Director の順に電源を再度投入してください。

⑪リソース情報を同期する

- ・⑧で Director のリセットを実施した場合は「3.4 CDI 管理ソフトウェアにログインする」を実行後「2.3.10 リソース情報の同期」の操作を行います。Director のリセットを実施していない場合、本操作は不要です。

⑫計算サーバの iRMC 情報を再登録する

- ・⑧で Director のリセットを実施した場合は、CDI 管理ソフトウェアに対し計算サーバの iRMC のユーザ名・パスワードの再登録を実施します。操作手順はシステム構築手順書「5.1.15 計算サーバの iRMC 情報を登録する」を参照してください。Director のリセットを実施していない場合本操作は不要です。

⑬グループ、論理サーバ構成の再構築

- ・⑧で Director のリセットを実施した場合は、構成バックアップファイルを使用してグループ、論理サーバ構成を再構築します。③で取得したダンプファイルに対して、「3.20.5 構成バックアップファイルの解凍」、「3.20.6 グループ・論理サーバの再構成」、および「3.20.7 論理サーバのステータスを確認」の操作を行います。Director のリセットを実施していない場合本操作は不要です。

⑭論理サーバの電源オン

- ・グループに存在する全ての論理サーバの電源をオンするために「3.1.8 論理サーバの電源をオン」の操作を行います。

4.2.3 NIC ファームウェアのアップデート

PCIe Box 内部の NIC ファームウェアをアップデートするには下記①～⑤を実施してください。

- ①論理サーバの利用停止を通知
 - ・論理サーバの電源オフが行われるため、全ての利用者に対して利用停止を通知します。必要であればデータのバックアップの実施も通知してください。
- ②ファームウェアのアップデート
 - ・NIC ファームウェア アップデート手順書を参考にして、ファームウェアをアップデートします。
- ③論理サーバの電源をオフ
 - ・グループに存在する全ての論理サーバの電源をオフするために「3.4 CDI 管理ソフトウェアにログインする」を実行後「3.2.3 論理サーバの電源をオフ」の操作を行います。
- ④CDI コンポーネントの電源を切断
 - ・CDI コンポーネントの電源を切断するために、システム構築手順書「4.9.2 電源切断手順」(2)および(3)の操作を行います。
- ⑤CDI コンポーネントの電源を投入
 - ・CDI コンポーネントの電源を投入するためにシステム構築手順書「4.9.1 電源投入手順」(2)～(4)を行います。

4.3 設定情報を管理する

ここでは、本システムで使用する設定情報の管理について記載します。

4.3.1 iRMC の設定情報

計算サーバの iRMC の IP アドレス・ユーザ名・パスワード情報は CDI 管理ソフトウェアおよび Director 制御ソフトウェアへの登録・更新が必須となります。iRMC のユーザ名・パスワードを変更した場合は、必ず下記に示す①～②を実施してください。

※他の利用者が iRMC のユーザ・パスワードを変更した場合も同様の手順が必要となります。

① Director 制御ソフトウェアに対し、計算サーバの iRMC のユーザ名・パスワードの変更
変更方法は、システム構築手順書「4.6.5 PDU IPMI の設定」を参照してください。

② CDI 管理ソフトウェアに対し、計算サーバの iRMC のユーザ名・パスワードの変更
変更方法は、システム構築手順書「5.1.15 計算サーバの iRMC 情報を登録する」を参照してください。

4.3.2 BIOS の設定情報

計算サーバの BIOS パスワード情報は CDI 管理ソフトウェアおよび Director 制御ソフトウェアへの登録不要です。論理サーバ作成後 BIOS パスワードを設定することは可能ですが、論理サーバを解体しても計算サーバの BIOS パスワード情報は残り、次回論理サーバ生成後にもパスワード入力が必要とされます。BIOS パスワード紛失防止の為、論理サーバを解体する前に必ず BIOS パスワードを指定なしの状態にしてください。

※万が一 BIOS パスワードを失念した場合は、計算サーバの修理が必要となります。

4.4 活性保守交換の可否

以下に FRU 単位での活性交換の可否をまとめて記載します。

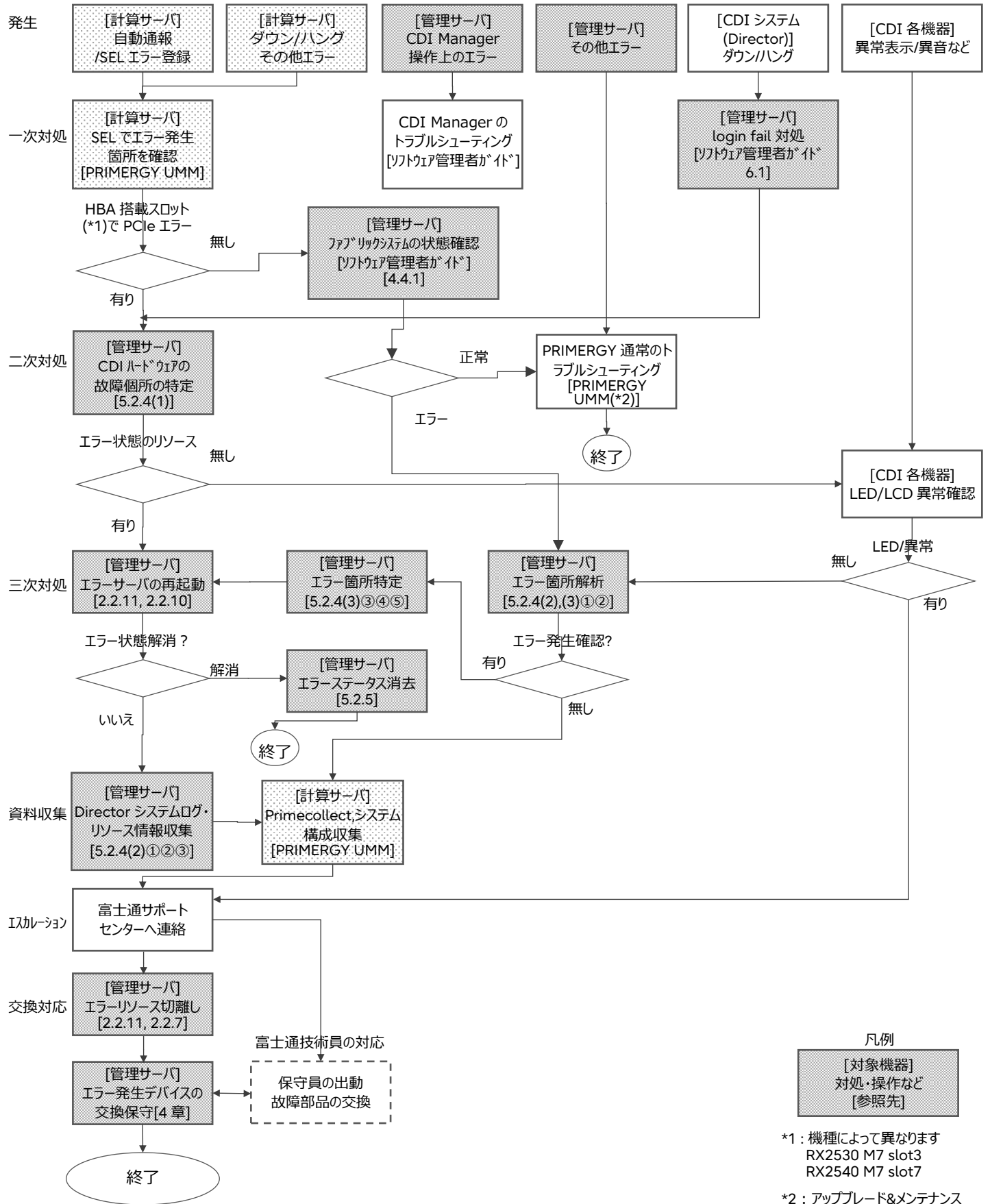
#	FRU	交換条件	活性交換	備考
1	Director	システム AC Off	不可	
2	PCIe Switch	システム AC Off	不可	
3	PCIe Box	システム AC Off	不可	
4	PCIe Cable(データ)	システム AC Off	不可	
5	PCIe Cable(管理)	システム AC Off	不可	
6	HBA	PRIMERGY AC Off	不可	PRIMERGY に搭載されるカード交換ではシステムの AC Off は不要 (計算サーバ交換の場合もシステムの AC Off は不要)
7	NVMe SSD	システム AC Off	不可	Liquid NVMe SSD は、SSD 8 個単位で交換
8	PCI Card	システム AC Off	不可	PCIe Box に搭載されるカード
9	PCI Card 補助電源 Cable	システム AC Off	不可	
10	PSU	システム AC On または Off	可	Director/PCI Switch/PCI Box に搭載されているもの
11	FAN	システム AC Off	不可	Director/PCI Switch/PCI Box に搭載されているもの
12	装置用外部電源ケーブル	システム AC On または Off	可	Director/PCI Switch/PCI Box 用の外部電源ケーブル
13	SmartPDU	システム AC Off	不可	
14	PRIMERGY 部品	PRIMERGY 仕様に従う	←	システムの AC Off は不要

第5章 システムの監視

本章では、システムで使用するハードウェアの状態監視と NVMe SSD の寿命確認について説明します。

5.1 故障箇所切り分けフロー

以下に、故障箇所切り分けフローの概要を示します。



凡例
[対象機器] 対処・操作など [参照先]

*1: 機種によって異なります
RX2530 M7 slot3
RX2540 M7 slot7

*2: アップグレード&メンテナンス マニュアル

5.2 ハードウェアの状態を監視する

ここでは、ハードウェアの状態監視について説明します。なお、監視対象となるハードウェアは、計算サーバ、HBA、PCIe カード（PCIe Box 内の GPU,NVMe SSD,NIC）になります。

5.2.1 計算サーバ, HBA の監視

計算サーバ, HBA は iRMC により状態監視を行います。詳細は、iRMC 操作手引きを参照してください。

5.2.2 GPU,NVMe SSD,NIC の監視

論理サーバに組み込まれた PCIe カード（GPU,NVMe SSD,NIC）についても iRMC により状態監視されます。ただし、iRMC の SEL（system event log）に示されるスロット位置は論理的なものであり、実際の PCIe BOX のスロット位置と異なるため物理的な位置の確認が必要になります。

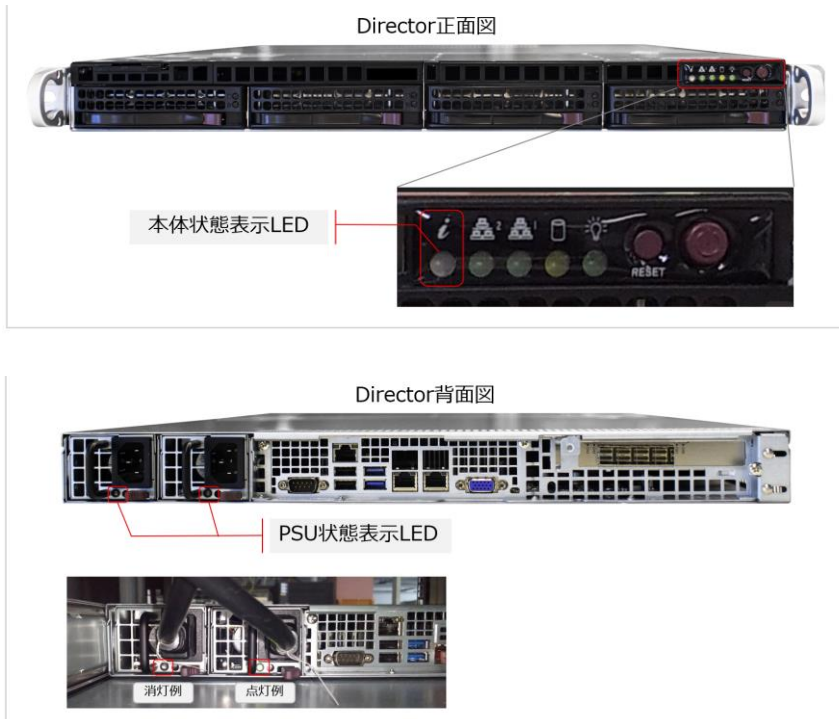
※ 物理的な位置の特定については「5.2.4 ハードウェアの故障個所の特定」を参照して下さい。

5.2.3 その他の保守部品の監視

本システムでは、保守部品である Director（ファン、PSU）、PCIe SW（ファン、PSU）、PCIe BOX（ファン、PSU）、Medusa Cable、PCIe Data Cable、GPU Pow Cable の状態監視を管理サーバから行うことはできませんが、5.1.3.1～5.1.2.3 に示す方法（直接外観を目視）で異常状態の検出が可能です。

5.2.3.1 Director 目視による異常検出

Director に対する目視検査では、以下の図に示す Director 正面の本体状態表示 LED、及び背面の PSU 状態表示 LED を確認して下さい。



以下の表に状態表示 LED の表示内容を示します。

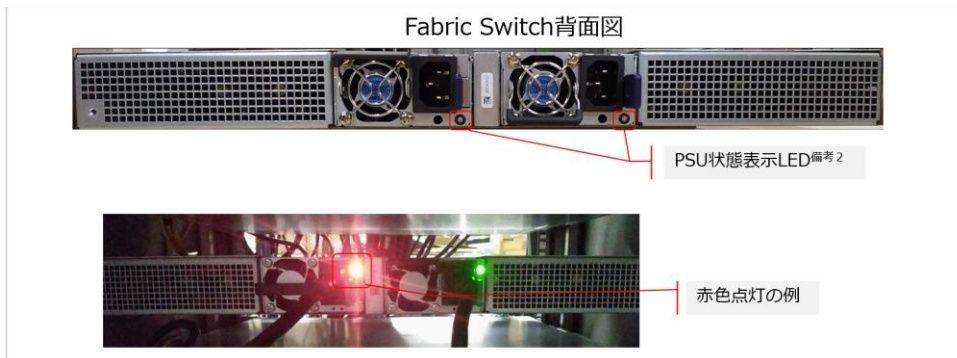
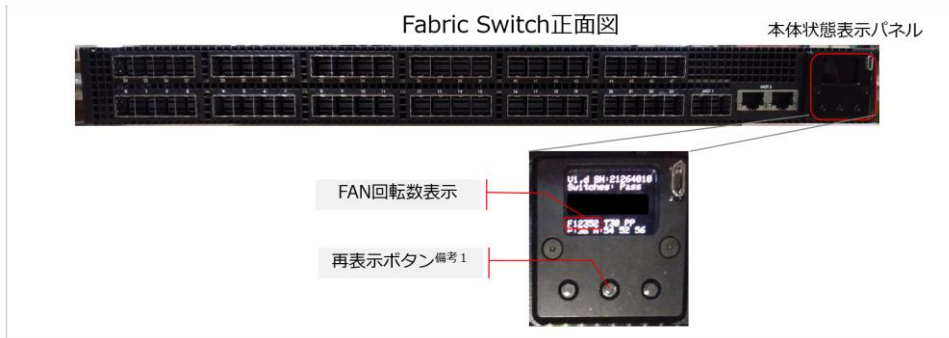
状態表示 LED	Director の状態	第一被疑部品	第二被疑部品	交換単位
消灯、または緑色点灯 点滅	正常状態	なし	—	—
赤色点灯	高温異常	Director FAN	Director 本体	FAN の交換単位は 4 つ同時
1 秒間隔で赤色点滅	FAN 異常	Director FAN	—	FAN の交換単位は 4 つ同時
0.5 秒間隔で赤色点滅	PSU 異常	Director PSU	—	交換単位は 1 台 被疑 PSU は下記参照 <small>備考 1 備考 2</small>

備考 1：PSU 状態表示 LED が緑点灯の場合 PSU は正常動作中であり、消灯の場合は AC 給電されていないか、または PSU 本体の異常である事を示します。

備考 2：PSU 状態表示 LED が消灯の場合、電源ケーブルの半抜け等によって AC 給電がされていない可能性があります。PSU 交換前に電源ケーブルの接続状態の確認をお願いします。

5.2.3.2 Fabric Switch 目視による異常検出

Fabric Switch に対する目視検査では、以下の図に示す Fabric Switch 正面の本体状態表示パネル、及び背面の PSU 状態表示 LED を確認して下さい。なお、Fabric Switch 装置は、ラック背面側に Fabric Switch のコネクタ面が来る様にラックに搭載されます。従って本体状態表示パネルを目視する場合はラック背面側から、PSU 状態表示 LED を目視する場合はラック正面側から目視して下さい。



備考 1 : Fabric Switch が通常動作中は、本体状態表示パネルの表示は消灯しています。この場合、図中の再表示ボタンを押して再表示を行って下さい。

備考 2 : Fabric Switch の PSU は複数のモデルがあり LED の実装位置が異なっていますが、表示が示す内容は同じです。

以下に本体状態表示パネルの FAN 回転数表示の内容、及び PSU 状態表示 LED の表示内容を示します。

状態表示 LED	Fabric Switch の状態	第一被疑部品	第二被疑部品	交換単位
12 万回転前後 ^{備考3}	正常状態	なし	—	—
11 万回転以下 ^{備考3}	FAN 異常	Fabric Switch FAN	—	FAN の交換単位は 6 つ同時

PSU 状態表示 LED	Fabric Switch の状態	第一被疑部品	第二被疑部品	交換単位
緑色点灯	正常状態	なし	—	—
赤色点灯	PSU 異常	Fabric Switch PSU	—	交換単位は 1 台 被疑 PSU は下記参照 ^{備考4 備考5}

備考 3 : FAN 回転数表示の FXXXXX は 5 桁の数値であり、FAN の回転数の平均値を示しています。通常時は概ね 1 万 2 千回転前後の値が表示されます。回転数が 1 万 1 千回転を下回っている場合 FAN 異常が疑われます。

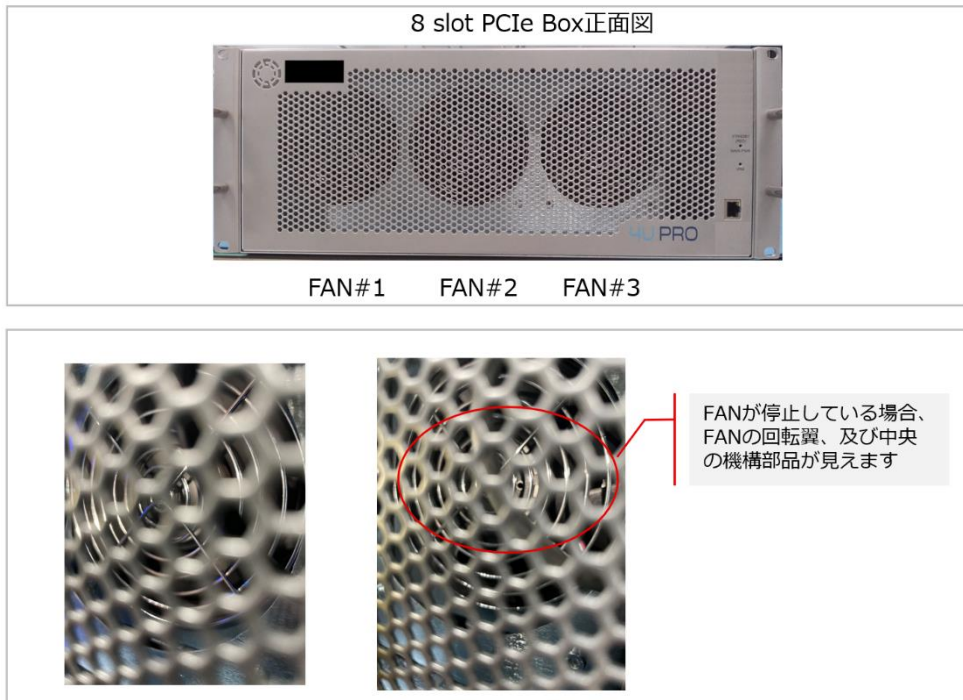
備考 4 : PSU 状態表示 LED が緑点灯の場合 PSU は正常動作中であり、赤色点灯の場合は AC 給電されていないか、または PSU 本体の異常である事を示します。

備考 5 : PSU 状態表示 LED が赤色点灯の場合、電源ケーブルの半抜け等によって AC 給電がされていない可能性があります。PSU 交換前に電源ケーブルの接続状態の確認をお願いします。

5.2.3.3 PCIe BOX 目視による異常検出

(1) 8 slot PCIe Box に対する目視検査では、管理サーバから遠隔監視する事が出来ません。代わりに直接外観を目視することで異常状態の検出を行って下さい。その方法を以下に記載します。

- ① FAN の回転異常は以下の様に FAN を目視して動作を確認して下さい。



FAN 目視作業時は、光源（懐中電灯等）を用意し、なるべく本体正面に近い位置から光を当てる事で FAN 本体が見やすくなります。

- ② PSU の状態は、背面の PSU 状態表示 LED を目視して動作を確認して下さい。



以下に FAN 目視、及び PSU 状態表示 LED の表示内容を示します。

本体正面 FAN 目視	PCIe Box の状態	第一被疑部品	第二被疑部品	交換単位
FAN 回転中	正常状態	なし	—	—
FAN 停止	FAN 異常	PCIe Box FAN	—	FAN の交換単位は 1 個

PSU 状態表示 LED	PCIe Box の状態	第一被疑部品	第二被疑部品	交換単位
緑色点灯	正常状態	なし	—	—

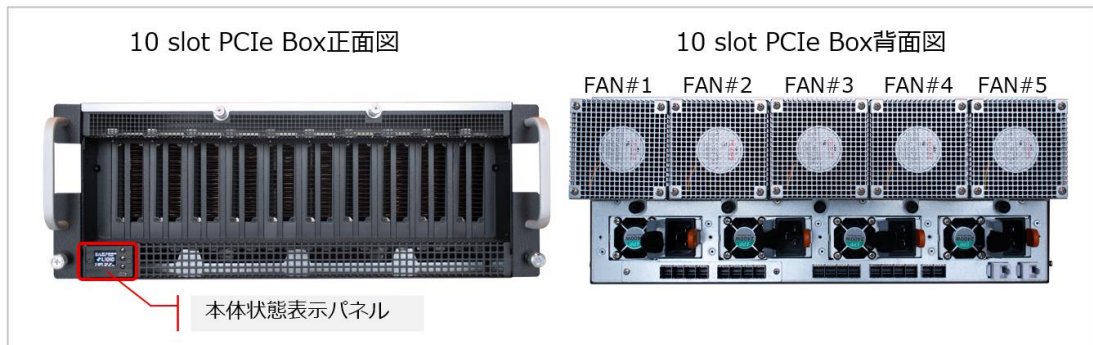
オレンジ点灯	PSU 異常	PCIe Box PSU	—	交換単位は 1 台 被疑 PSU は下記参照 ^{備考 1 備考 2}
--------	--------	--------------	---	--

備考 1：PSU 状態表示 LED が緑点灯の場合 PSU は正常動作中であり、オレンジ点灯の場合は AC 給電されていないか、または PSU 本体の異常である事を示します。

備考 2：PSU 状態表示 LED がオレンジ点灯の場合、電源ケーブルの半抜け等によって AC 給電がされていない可能性があります。PSU 交換前に電源ケーブルの接続状態の確認をお願いします。

(2) 10 slot PCIe Box に対する目視検査では、以下の図に示す正面の本体状態表示パネル、及び背面の PSU 状態表示 LED を確認して下さい。

- ① FAN の回転異常は、以下の図に示すとおり本体前面の表示パネルを目視して下さい。以下に FAN 異常時の表示例を示します。



本体状態表示パネルの表示が消灯している場合、以下の再表示ボタンを押して下さい。



故障したファンがある場合、表示パネルの下から 3 行目に FAN 番号と failure が表示されます。また下から 2 行目にファンの速度が順番に表示されます。(5 つの FAN 情報が自動的に循環表示されます) 故障したファンは赤色になり RPM は 0k (0 RPM) と表示されます。複数の FAN が故障している場合、下から 3 行目には FAN 番号の大きい方が赤字表示されて FAN 番号の小さい方は表示されません。

以下に例として FAN#4 と FAN#5 が故障した場合の表示例を示します。



- ② PSU の状態は、本体状態表示パネル、及び背面の PSU 状態表示 LED を目視して動作を確認して下さい。



PSU の状態は、上図の通り本体状態表示パネルに 4 つの文字で表示されます。4 つの文字はそれぞれ左から PSU#3、PSU#2、PSU#1、PSU#0 の状態を 1 文字で表します。"P"が表示されている場合正常動作中、"N"が表示されている給電されていないか、または PSU 異常であることを示します。

以下に FAN 目視、及び PSU 状態表示 LED の表示内容を示します。

表示パネル目視	PCIe Box の状態	第一被疑部品	交換単位
Fan N failure 表示なし	正常状態	なし	—
Fan N failure 表示あり	FAN 異常	PCIe Box FAN	FAN の交換単位は 1 個 交換する FAN は表示パネルを参照

表示パネル目視、 PSU 状態表示 LED	PCIe Box の状態	第一被疑部品	交換単位
表示パネルが"P"表示、かつ LED が緑色点灯	正常状態	なし	—
表示パネルが"N"表示 かつ LED がオレンジ色点灯	PSU 異常	PCIe Box PSU	交換単位は 1 台 被疑 PSU は下記参照 <small>備考 1 備考 2</small>

備考 1 : PSU 状態表示 LED が緑点灯の場合 PSU は正常動作中であり、オレンジ点灯の場合は AC 給電されていないか、または PSU 本体の異常である事を示します。

備考 2 : PSU 状態表示 LED がオレンジ点灯の場合、電源ケーブルの半抜け等によって AC 給電がされていない可能性があります。PSU 交換前に電源ケーブルの接続状態の確認をお願いします。

5.2.4 ハードウェアの故障個所の特定

iRMC によりハードウェアの故障が検知された時は、アップグレード&メンテナンスマニュアルや SEL 情報に従って故障箇所を特定した後に適当な対処を行って下さい。

なお、故障箇所が PCI スロットの場合は PCIe デバイスの物理位置特定が必要になるため下記 (1) ~ (3) を行って PCIe カードにエラーが発生しているかを確認し、適当な対処をしてください。

また、iRMC によるハードウェアの故障検知が無くとも、Input/output error 等により PCIe カードにアクセスできないことによる論理サーバのハングアップ状態を検出した場合や、PCIe カードのドライバエラーを検出し、ハード故障を疑われる場合にも、故障箇所の切り分けのために (1) ~ (3) の手順を実施してください。

(1) PCIe カードの故障有無と故障箇所を確認

- エラー状態のリソースの有無を「2.3.8 論理サーバで使用されているエラー状態のリソースを表示」の操作により故障の有無と故障箇所を確認します。「3.4 CDI 管理ソフトウェアにログインする」の手順で CDI 管理ソフトウェアにログインし、以下を実行します。

※下記の resource list -E コマンドでは res_op_status が 0 以外のエラー状態のリソース一覧が表示されます。
res_op_status=2 のリソースが保守交換対象のリソースです。

```
cdi: ./epcctl -u $URL resource list -E -M -o YAML
200
- res_name: gpu0
  res_op_status: '2' ← 2: Critical (= 保守交換対象を示す)
  res_owner: n/a
  res_status: 1
  res_type: gpu
  res_uuid: 2df31bfc-723e-4781-a83d-4372630c3498
  rack_name: 'rack0'
  position_in_rack: '10U'
  slot_number: 0
```

} エラー状態のリソース

↑ エラー状態リソースの物理位置

rack_name, position_in_rack : システム構築手順書「5.1.8 (8) スペックファイルを編集する」で設定した情報に対応
slot_number : システム構築手順書「2.3.1 8-slot PCIe Box のスロット番号」および「2.3.2 10 slot PCIe Box のスロット番号」参照

(2) エラーへの対応

- エラー状態のリソース表示有りの場合：(1)でリソースが表示される時
PCIe カードの故障と考えられるため、SEL でエラーの詳細を確認します。カードの故障と判断し、物理交換保守を実施するまでの間、その故障カードを含む論理サーバの利用を続けたい場合は「3.12 論理サーバからエラー状態のリソースを削除する」に従って故障カードを切り離すことで利用が続けられます。
また、故障カードの物理交換保守を実施する場合は、エラー状態リソースの物理位置を確認し「4.1 ハードウェアの増設・減設・交換をする」以降の手順で保守交換を実施してください。
- エラー状態のリソース表示なしの場合：(1)でリソースが表示されない時
PCIe カード以外の故障と考えられるため、以下の①～④の情報を採取し富士通サポートにご連絡ください。
※②～③については「3.4 CDI 管理ソフトウェアにログインする」の手順で CDI 管理ソフトウェアにログインして実施します。

①構築時や増設・減設・交換保守時に採取しておいたログ

ディレクトリ"/home/<Host OS のユーザ名>/cdi/log"に格納された"setup_<日付:yyyymmdd>.tar.gz"のうち最新の日付もの。

※<Host OS のユーザ名>はシステム構築手順書「4.3.1.4 OS インストール手順」で設定した Host OS のログイン ID になります。

②システムログ

「2.6.3 ログファイルの作成」の操作により作成されるファイル

③リソースの一覧

「2.3.1 リソースの一覧を表示」の操作結果を保存したファイル

```
cdi: ./epcctl -u $URL resource list -o LIST > /home/<Host OS のユーザ名>/cdi/log/res_list20230407.txt
res_list20230407.txt
```

res_type	res_name	res_uid	slot_number
compute	pcpu0	b9c59f7d-0a92-4af2-a28e-3778e5b922c1	8
compute	pcpu1	4c69968e-df77-4b37-928a-12bd2d401cc8	8
gpu	gpu0	2df31bfc-723e-4781-a83d-4372630c3498	None
gpu	gpu1	5b30e05a-ef22-4349-b4fd-5cdb45029a2	None
gpu	gpu2	5b30e05a-ef22-4349-b4fd-5cdb45029a2	None
network	nic0	3fc2e7b3-f86d-47b1-9f1f-7d7ebaefef309	None
storage	ssd0	4894c94b-a598-4882-ae2c-dab4c36f5267	None
storage	ssd1	a243dbdf-b7e3-4aa9-b8b2-bb2bb56cf490	None

④全リソースの詳細情報

③で得られた全てのリソースについて「2.3.6 リソースの詳細を表示」の操作結果を保存したファイル

```
cdi: ./epcctl -u $URL resource show -r pcpu0 -o LIST > /home/<Host OS のユーザ名>/cdi/log/respcpu0_show20230407.txt
cdi: ./epcctl -u $URL resource show -r pcpu1 -o LIST > /home/<Host OS のユーザ名>/cdi/log/respcpu1_show20230407.txt
cdi: ./epcctl -u $URL resource show -r gpu0 -o LIST > /home/<Host OS のユーザ名>/cdi/log/resgpu0_show20230407.txt
:
cdi: ./epcctl -u $URL resource show -r ssd1 -o LIST > /home/<Host OS のユーザ名>/cdi/log/resssd1_show20230407.txt
```

resgpu0_show20230407.txt

item	value
res_uid	2df31bfc-723e-4781-a83d-4372630c3498
res_id_nonliquid	
fabr_gid	0x033c00
res_type	gpu
res_name	gpu0
...	

(3) エラー発生した PCIe カードや計算サーバの特定

- ・(2)でエラー状態のリソース表示なしの場合でも、Director のログからエラー発生した PCIe カードを特定することが可能なケースがあります。必要に応じて以下の①～⑤の手順で特定してください。

①システムログを採取

「3.4 CDI 管理ソフトウェアにログインする」の手順で CDI 管理ソフトウェアにログインし、(2)の②と同様の操作でシステムログを採取します。

```
# CDI 管理ソフトウェアにログインする
```

```
$ cd /home/<Host OS のユーザ名>/cdi/RMcli<version>/RMcli<version>_<OSname>
```

```
$ ./epcctl -u $URL login
```

```
Enter Your Username : xxxx ← ユーザ名を入力
```

```
Enter Your Password : **** ← パスワードを入力
```

```
login success
```

```
# システムログを採取する
```

```
cdi: ./epcctl -u $URL system log -o /home/<Host OS のユーザ名>/cdi/log/20230407.tar.gz
```

システムログ採取後、CDI 管理ソフトウェアからログアウトします。

```
cdi: exit
```

```
$
```

②エラー発生デバイスのバス番号を確認

ログを展開し、以下のファイルに"Plx_Mgr ERR[XX]ds_dev_remove: DPC event for XXXX:XX:XX.X"のメッセージが含まれていないかを確認します。含まれていた場合、そのメッセージにエラー発生デバイスのバス番号が示されま

```
$ tar -zxvf /home/<Host OS のユーザ名>/cdi/log/20230407.tar.gz
var/log/epc/resource_manager/nginx_conf_access.log-20230406.gz
var/log/epc/resource_manager/nginx_error.log-20230406.gz
...
./liquid_log_*/liquid_log.tar.gz
...

$ tar -zxvf liquid_log_*/liquid_log.tar.gz
liqsysinfo-*/
liqsysinfo-*/system/
liqsysinfo-*/configs/
...
liqsysinfo-*/dmesg.syslog
...

$ grep 'ds_dev_remove: DPC event' ./liqsysinfo-*/dmesg.syslog
[Thu Sep 7 02:27:29 2023] Plx_Mgr ERR[53]ds_dev_remove: DPC event for 0000:11:00.0. ds_port=...(以下略)
```

この例ではバス番号は 11

※"*****"は自動的に付与される文字列になります。ls コマンドでディレクトリ名を確認し、付与された文字列を記載ください。

③バス番号に相当するリソース名を特定

「3.4 CDI 管理ソフトウェアにログインする」の手順で CDI 管理ソフトウェアに再度ログインし、②で確認したバス番号と一致するものをリソース詳細情報の"fabr_gid"から探し、そのリソース名"res_name"を特定します。

```
# CDI 管理ソフトウェアにログインする
$ cd /home/<Host OS のユーザ名>/cdi/RMcli<version>/RMcli<version>_<OSname>
$ ./epcctl -u $URL login
Enter Your Username : xxxx ← ユーザ名を入力
Enter Your Password : **** ← パスワードを入力
login success

# リソース詳細情報を取得する (以下は res_name:gpu0 の例)
cdi: ./epcctl -u $URL resource show -r gpu0
200
{
  "res_uuid": "c7d2a177-371c-46b2-acc6-33e812db6c99",
  "res_id_nonliquid": null,
  "fabr_gid": "0x033c00", ← fabr_gid の 3 桁目と 4 桁目がバス番号となる (この例では"3c" がバス番号)
  "res_type": "gpu",
  "res_name": "gpu0", ← リソース名
  ...
}
```

②で取得したバス番号と一致しない場合は、他のリソースを指定して詳細情報を確認してください。

④物理位置を特定

resource list コマンドで表示されるリソースリストの中から③で特定したリソース名から物理位置を特定します。

```

cdi: ./epcctl -u $URL resource list -g g0
200
{
  "data": {
    "resources": [
      {
        "res_uuid": "2df31bfc-723e-4781-a83d-4372630c3498",
        "res_type": "gpu",
        "res_name": "gpu0",
        "res_status": 4,
        "res_op_status": "2",
        "res_owner": "Fujitsu"
        "rack_name": "rack0",
        "position_in_rack": "5U",
        "slot_number": 8
      },
    ],
  },
}
    
```

リソースの物理位置

Rack_name, position_in_rack : システム構築手順書「5.1.8 (8) スペックファイルを編集する」で設定した情報に対応
 Slot_number : システム構築手順書「2.3.1 8-slot PCIe Box のスロット番号」および「2.3.2 10 slot PCIe Box のスロット番号」参照

⑤エラー発生デバイスの属する計算サーバの特定

③で探したリソースの詳細情報中のマシン番号 (mach_uuid) と一致するものを、machine list コマンドで示されるマシンのリストの中から探してください。該当するマシン名(mach_name)がエラー発生デバイスの属する計算サーバを特定できます。

```

cdi: ./epcctl -u $URL machine list -g g0
200
{
  "data": {
    "machines": [
      {
        "mach_uuid": "7d6210ef-8a6a-403e-a980-5542b7852c30",
        "mach_id": 1 ,
        "mach_name": " FJMachine ",
        "mach_owner": "Fujitsu",
        "resources": [
          {
            "res_uuid": "2df31bfc-723e-4781-a83d-4372630c3498",
            "res_name": "gpu0",
            "res_type": "gpu",
            "res_status": 4,
            "res_op_status": "2"
          },
          {
            ...
          }
        ],
      },
    ],
  },
}
    
```

5.2.5 エラー状態となったリソースの扱い

本システムでは、論理サーバで使用中の PCIe デバイスの稼働状態に問題があると、CDI 管理ソフトウェア内にエラー状態リソースとして登録されます。エラー状態として登録されたリソースは「2.3.8 論理サーバで使用されているエラー状態のリソースを表示」の操作により確認できます。

論理サーバの再起動等の対応によって問題が解消し、リソースの交換が不要と判断した場合には「2.3.9 リソースステータスの設定」でオプションステータスに“0:OK”を設定してください。

問題が解消しない場合は、「3.12 論理サーバからエラー状態のリソースを削除する」および「3.13 グループフリープールからエラー状態のリソースを削除する」の操作を行った後に交換などの対応を検討してください。リソースのエラー状態は自動的に登録解除されませんので、PCIe デバイスの問題の解消方法に応じて適宜登録解除する必要があります。

- 論理サーバの再起動等 PCIe デバイスの交換を行わずに問題が解消した場合
「2.3.9 リソースステータスの設定」でオプションステータスに“0:OK”を設定してください。
- 再起動などで問題が解消せず、PCIe デバイスの交換を行う場合
「4.1.2.1 PCIe カードの増設・減設・交換保守手順」に沿って PCIe デバイスの交換を実施してください。

5.3 NVMe SSD の寿命を確認する

本システムでは、交換時期を判断するための NVMe SSD（リソース）の書き込み寿命の確認が行えます。なお、確認手順は、NVMe SSD がグループのフリープールに存在する場合と、論理サーバに組み込まれている場合で異なります。

5.3.1 フリープールに存在する NVMe SSD の寿命確認

グループフリープール内の NVMe SSD の寿命を確認するには下記①～⑤を実施して下さい。

① CDI 管理ソフトウェアにログイン

- ・「システム管理者 PC から管理サーバの Host OS に接続して、「3.4 CDI 管理ソフトウェアにログインする」の手順を行い CDI 管理ソフトウェアにログインします。

② 寿命確認の対象 NVMe SSD を組み込んだ論理サーバの作成

- ・確認対象の NVMe SSD と、何れかの計算サーバを使用して論理サーバを作成します。「2.2.3 論理サーバの作成（コマンド指定）」の操作により作成します。

例) 計算サーバ pcpu0 を使用して、ssd0 を確認対象とする論理サーバを作成

```
cdi: ./epcctl -u $URL machine create -g group0 -m machine0 -r pcpu0,ssd0
200
{'data': {'machines': [{'mach_uuid': 'b9e25b57-521a-4047-b2cd-6159c452ac26'}]}}
```

- ※使用できる計算サーバが無い場合は「3.7 グループにリソースを追加する」の操作でグループフリープールに追加してください。

③ ステータスの確認

- ・②で作成した論理サーバの詳細ステータス(mach_status_detail)が“INACTIVE POFF”であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

```
"mach_uuid": "b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": "INACTIVE POFF",
...
```

- ※ “INACTIVE POFF”以外が表示された場合は、“INACTIVE POFF”が表示されるまで確認を繰り返します。

④ 寿命確認用の OS イメージをインストール

- ・確認対象の NVMe SSD に寿命確認用 OS イメージをインストールするために「2.4.5 OS イメージのインストール」の操作を行います。

```
cdi: ./epcctl -u $URL resource install -r ssd0 -b OSsmartctl<version>.img
200
success
```

- ※寿命確認用 OS イメージ (OSsmartctl<version>.img) はシステムの構築時に管理サーバに格納されています。詳細は、システムの構築手順書を参照してください。

⑤ 論理サーバの電源をオン

- ・②で作成した論理サーバの電源をオンするために「2.2.10 論理サーバの電源オン」の操作を行います。

```
cdi: ./epcctl -u $URL machine power -P on -m machine0
```

⑥ ステータスの確認

- ⑤で電源をオンした論理サーバの詳細ステータス(mach_status_detail)が"ACTIVE PON"であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 13,
"mach_status_detail": " ACTIVE PON",
. . .
```

※ "ACTIVE PON"以外が表示された場合は、"ACTIVE PON"が表示されるまで確認を繰り返します。

⑦ 計算サーバの iRMC 情報を確認

- 論理サーバに組み込まれた計算サーバの iRMC 情報を「2.6.2 iRMC 情報の表示（計算サーバ）」の操作により確認します。

res_uuid_cpunode	cpunode_ipmi_address	cpunode_ipmi_userid	cpunode_ipmi_password
b9c59f7d-0a92-4af2-a28e-3778e5b922c1	192.168.1.xxx	XXrt08y	Lk2!4Q@We

⑧ ビデオリダイレクションの実行

- ⑦で表示された iRMC 情報を使用して論理サーバの iRMC にログインし、ビデオリダイレクションを開きます。

⑨ 寿命確認用 OS にログイン

- ユーザ : smartctl、パスワード : smartctl を入力し、寿命確認用 OS にログインします。

⑩ 寿命の確認

- smartctl コマンドを実行し NVMe SSD の寿命情報を表示します。

```
$ lsblk ← 監視対象 NVMe SSD のディスクドライブ名を確認
/dev/nvme0n1
$ sudo smartctl -x /dev/nvme0n1 ← smartctl コマンドの実行
smartctl 7.1 2019-12-30 r5022 [x86_64-linux-5.4.0-125-generic] (local build)
Copyright (C) 2002-19, Bruce Allen, Christian Franke, www.smartmontools.org
...
=== START OF SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED

SMART/Health Information (NVMe Log 0x02)
Critical Warning:          0x00
Temperature:              33 Celsius
Available Spare:          100%
Available Spare Threshold: 10%
Percentage Used:          3%
Data Units Read:          440,607 [225 GB]
Data Units Written:       649,082 [332 GB]
```

- Percentage Used 値を確認します。

```
=== START OF SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED

SMART/Health Information (NVMe Log 0x02)
. . .
Percentage Used:          3%
. . .
```

※ 初期値はほぼ 0% で、書き込み回数の累積に伴い増加し 100% で書き込み寿命となります。

⑪ 論理サーバの電源をオフ

- ②で作成した論理サーバの電源をオフするために「2.2.11 論理サーバの電源オフ」の操作を行います。

```
cdi: ./epcctl -u $URL machine power -P off -m machine0
```

⑫ ステータスの確認

- ⑪ で電源をオフした論理サーバの詳細ステータス(mach_status_detail)が" INACTIVE POFF"であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 15,
"mach_status_detail": " INACTIVE POFF",
. . .
```

※ "INACTIVE POFF"以外 が表示された場合は、"INACTIVE POFF"が表示されるまで確認を繰り返します。

⑬ 論理サーバを解体

- ① で作成した論理サーバを解体するために「3.14 論理サーバを解体する」の操作を行います。

⑭ 寿命確認の対象 NVMe SSD のオプションステータスを設定

- 寿命が近づいた(Percentage Used の値が 100 に近づいた)NVMe SSD のオプションステータスを変更するために「2.3.9 リソース ステータスの設定」の操作を行います。

```
cdi: ./epcctl -u $URL resource set_op_status -r ssd0 -s 1 ← 1:Warning を設定します。
```

⑮ 寿命が近づいた NVMe SSD をファブリックフリープールに移動

- 寿命が近づいた NVMe SSD をファブリックフリープールに移動して、交換などの対応をするために「3.13 グループフリープールからエラー状態のリソースを削除する」の操作を行います。

5.3.2 論理サーバに組み込まれた NVMe SSD の寿命確認

論理サーバに組み込まれた NVMe SSD の寿命を確認するには下記①～⑨の手順を実施してください。

① CDI 管理ソフトウェアにログイン

- 「システム管理者 PC から管理サーバの Host OS に接続して、「3.4 CDI 管理ソフトウェアにログインする」の手順を行い CDI 管理ソフトウェアにログインします。

② 論理サーバの電源をオン

- 確認対象の NVMe SSD が組み込まれた論理サーバの電源がオフの時は「2.2.10 論理サーバの電源オン」の操作を行います。

```
cdi: ./epcctl -u $URL machine power -P on -m machine0
```

③ ステータスの確認

- 確認対象の NVMe SSD が組み込まれた論理サーバの詳細ステータス(mach_status_detail)が"ACTIVE PON"であることを「2.2.12 論理サーバの状態確認」の操作により確認します。

```
"mach_uuid": " b9e25b57-521a-4047-b2cd-6159c452ac26",
"mach_status": 13,
"mach_status_detail": " ACTIVE PON",
. . .
```

※ "ACTIVE PON"以外 が表示された場合は、"ACTIVE PON"が表示されるまで確認を繰り返します。

④ 計算サーバの iRMC 情報を確認

- 論理サーバに組み込まれた計算サーバの iRMC 情報を「2.6.2 iRMC 情報の表示 (計算サーバ)」の操作により確認します。

res_uuid_cpunode	cpunode_ipmi_address	cpunode_ipmi_userid	cpunode_ipmi_password
b9c59f7d-0a92-4af2-a28e-3778e5b922c1	192.168.1.xxx	XXrt08y	Lk2!4Q@We

⑤ ビデオリダイレクションの実行

- ・④で表示された iRMC 情報を使用して論理サーバの iRMC にログインし、ビデオリダイレクションを開きます。

⑥ 寿命確認用 OS にログイン

- ・ユーザ : smartctl、パスワード : smartctl を入力し、寿命確認用 OS にログインします。

⑦ smartctl をインストール

`$ sudo apt -y install smartmontools` ← 使用 OS が Ubuntu の場合のインストールコマンド
 ※ OS に合わせた方法でインストールしてください。

⑧ 寿命の確認

- ・ smartctl コマンドを実行し NVMe SSD の寿命情報を表示します。

```
$ lsblk ← 監視対象 NVMe SSD のディスクドライブ名を確認
/dev/nvme0n1
$ sudo smartctl -x /dev/nvme0n1 ← smartctl コマンドの実行
smartctl 7.1 2019-12-30 r5022 [x86_64-linux-5.4.0-125-generic] (local build)
Copyright (C) 2002-19, Bruce Allen, Christian Franke, www.smartmontools.org
...
=== START OF SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED

SMART/Health Information (NVMe Log 0x02)
Critical Warning:          0x00
Temperature:              33 Celsius
Available Spare:          100%
Available Spare Threshold: 10%
Percentage Used:          3%
Data Units Read:          440,607 [225 GB]
Data Units Written:       649,082 [332 GB]
```

- ・ Percentage Used 値を確認します。

```
=== START OF SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED

SMART/Health Information (NVMe Log 0x02)
Percentage Used:          3%
```

※ 初期値はほぼ 0% で、書き込み回数の累積に伴い増加し 100% で書き込み寿命となります。

⑨ 寿命確認の対象 NVMe SSD のオプションステータスを設定

- ・ 寿命が近づいた (Percentage Used の値が 100 に近づいた) NVMe SSD のオプションステータスを変更するために「2.3.9 リソース ステータスの設定」の操作を行います。

`cdi: ./epcctl -u $URL resource set_op_status -r ssd0 -s 1` ← 1:Warning を設定します。

※ ステータスを変更した NVMe SSD は、交換などの対応を検討してください。

第6章 システムのセキュリティ

本章では、本システムの運用にあたり、情報漏洩やデータ消失等のセキュリティリスクを回避するために確認すべき観点について説明します。

※ 本システムはイントラネット環境での稼働を前提としており、個人情報の取り扱い等、固有のセキュリティ対策は実装されておりませんので、お客様にて運用を踏まえたセキュリティポリシーを策定いただき、対策を実施してください。

下記事項について、運用開始前にご対応ください。

項目	内容
証明書の設定	<p>本システムは、CDI 管理ソフトウェアと特定の機器との間で証明書を利用した通信を行います。</p> <p>証明書の設定については「6.1 証明書の設定」を参照ください。</p>
プロトコル、ポートの設定	<p>本システムでは複数の機器が使用されますが、セキュリティの弱点となる機器を保護するために、ネットワーク環境の設定を行ってください。</p> <p>プロトコル、ポートの設定については「6.2 プロトコル、ポートの設定」を参照ください。</p>
初期パスワードの変更	<p>本システムでは、ユーザー名/パスワードを用いてアクセスする機能が存在します。この中で初期パスワードを用いる機能について、お客様のパスワードに再設定してから運用を開始してください。</p> <p>初期パスワードを使用する箇所と変更方法については「6.3 初期パスワードの変更」を参照ください。</p>
ソフトウェアのアップデート	<p>本システムが利用するソフトウェアは、配布時に公開された修正内容や適用条件等を確認し、運用環境に合わせてアップデートを実施してください。</p> <p>なお、アップデート適用中は電源の切断またはリセットを行わないでください。更新中に電源を切断またはリセットした場合は、起動しなくなる可能性があります。</p>
OS、OSS のセキュリティパッチ適用	<p>システム内にて使用している OS、OSS については、お客様にて、常に最新の情報をご確認いただきますようお願い申し上げます。</p> <p>なお、公開されている情報に従い対応を行った（あるいは行わなかった）ことによりお客様に損害が生じた場合には、当社はその損害につき責任を負いかねます。</p>
MFA(多要素認証)	<p>本システムは、MFA(多要素認証)には対応しておりません。</p>

6.1 証明書の設定

本システムでは、以下の通信経路で証明書を用いた通信を行います。各経路について、以降の手順に従い証明書の設定を行ってください。

通信元 (クライアント)	通信先 (サーバ)	設定手順
CDI 管理ソフトウェア	計算サーバ(iRMC)	システム構築手順書 「5.1.2 計算サーバ証明書を生成する」 「5.1.6 CDI 管理ソフトウェアに証明書を登録する」
CDI 管理ソフトウェア	Director	システム構築手順書 「5.1.3 Director の HTTPS 設定および証明書を生成する」 「5.1.6 CDI 管理ソフトウェアに証明書を登録する」
システム管理者 PC	CDI 管理ソフトウェア	システム構築手順書 「5.1.7 CDI 管理ソフトウェアの証明書を作成する」

補足事項:

- ・本資料では、自己署名証明書を使用する方法について説明します。自己署名証明書以外を使用する場合は、お客様にてサーバ証明書およびルート証明書をご用意のうえ、設定を行ってください。
- ・上記以外の機器間にて証明書(https)による通信を行っている箇所については、通信先(サーバ)に関するドキュメントを参照して証明書の設定を行ってください。証明書(https)による通信を行っている箇所については「6.2 プロトコル、ポートの設定」を参照ください。

6.2 プロトコル、ポートの設定

本製品の機器が使用するプロトコル、ポートを以下に示します。

通信元	通信先	通信方法		備考	
		プロトコル	ポート		
システム管理者 PC	CDI 管理ソフトウェア ※REST API	TCP	443(HTTPS)		
	管理サーバ	TCP	22(SSH)		
	計算サーバ ※iRMC	TCP	443(HTTPS)		
	CDI 管理ソフトウェア ※VM	TCP	22(SSH)		
	Director		TCP	443(HTTPS)	
			TCP	22(SSH)	
Smart PDU	TCP	443(HTTPS)			
論理サーバ利用者 PC	計算サーバ ※iRMC	TCP	443(HTTPS)		
CDI 管理ソフトウェア	計算サーバ ※iRMC	TCP	443(HTTPS)		
	Director	TCP	443(HTTPS)		
計算サーバ ※iRMC	システム管理者 PC	TCP	25(SMTP)		
Director	計算サーバ ※iRMC	UDP	623(IPMI)		
	Smart PDU	TCP	443(HTTPS)		
管理 LAN に接続された 各装置	NTP サーバ	UDP	123	お客様にてご用意ください	
	DNS サーバ	UDP/TCP	53	お客様にてご用意ください	

上記機器において、より高いセキュリティレベルでの運用が求められる機器について、個別にセキュリティの強化を行ってください。具体的には、以下の表において「保護レベル」列に「Security」と記載された機器が該当します。これらの機器は、それ以外の機器とは独立したセグメントに配置していただき、「Public」の機器を経由してのみ外部ネットワークからアクセス可能となるよう、ネットワーク環境を設計してください。

機器情報		保護レベル	備考
大分類	小分類		
システム管理者 PC	—	Public	
論理サーバ利用者 PC	—	Public	
管理サーバ	本体	Public	管理 LAN
	CDI 管理ソフトウェア	Security	管理 LAN
計算サーバ	本体	Public	
	iRMC	Public	管理 LAN
Director	—	Security	管理 LAN
Smart PDU	—	Security	管理 LAN

6.3 初期パスワードの変更

初期パスワードの変更方法について、以下に記述します。

設定先	設定手順
CDI 管理ソフトウェア Guest OS (仮想マシン)	システム構築手順書 「5.1.5 CDI 管理ソフトウェアの環境設定を行う」(2)(4)
CDI 管理ソフトウェア RabbitMQ	システム構築手順書 「5.1.5 CDI 管理ソフトウェアの環境設定を行う」(3)
CDI 管理ソフトウェア MySQL	システム構築手順書 「5.1.5 CDI 管理ソフトウェアの環境設定を行う」(3)

補足事項:

- ・ 上記以外の各コンポーネント(Director、PDU、Ethernet Switch 等)の初期パスワードおよび変更方法につきましては、システム構築手順書「4.1.3 コンポーネントのログイン情報の初期値」を参照ください。

