NEC

SX-Aurora TSUBASA

インストレーションガイド

SX-Aurora TSUBASA

輸出する際の注意事項

本製品（ソフトウェアを含む）は、外国為替および外国貿易法で規定される規制貨物（または役務）に該当することがあります。

その場合、日本国外へ輸出する場合には日本国政府の輸出許可が必要です。

なお、輸出許可申請手続きにあたり資料等が必要な場合には、お買い上げの販売店またはお近くの当社営業拠点にご相談ください。

は　し　が　き

本書は、VE カードを搭載したマシン Vector Island（VI）上で SX-Aurora TSUBASA システムの運用が行えるようにするため、SX-Aurora TSUBASA システムの運用に必要なソフトウェアのインストール、環境設定及びソフトウェアのアップデートの手順について説明したものです。

本書の最新版は、以下のサイトから入手可能です。

https://sxauroratsubasa.sakura.ne.jp/documents/guide/pdfs/InstallationGuide_J.pdf

なお、SX-Aurora TSUBASA を利用するにあたって、OS およびソフトウェアのインストールからサンプルプログラムの実行までの、基本的な環境構築についての手順を画面イメージ付きで説明した「セットアップガイド」を用意していますので、合わせてご利用ください。

https://sxauroratsubasa.sakura.ne.jp/documents/guide/pdfs/SetupGuide_J.pdf

なお、セットアップガイドはモデル A100-1 のみに対応し、また、ScaTeFS、NQSV のインストールの説明を含んでいません。

⚠ 注意

本書に記載されている'#'プロンプトで始まるコマンド行は、スーパーユーザー権限で実行してください。

商標、著作権について

- Linux はアメリカ合衆国及びその他の国におけるLinus Torvalds の商標です。
- Red Hat、Red Hat Enterprise Linuxは米国およびその他の国において登録されたRed Hat, Inc.の商標です。
- Apacheは、Apache Software Foundationの商標または登録商標です。
- InfiniBand は、InfiniBand Trade Association の商標またはサービスマークです。
- NVIDIAは米国および/または他国のNVIDIA Corporationの商標および/または登録商標です。
- Windowsは、米国Microsoft Corporationの米国およびその他の国における登録商標です。
- その他、記載されている会社名、製品名は、各社の登録商標または商標です。

用語定義・略語

用語・略語	説　明
ベクトルエンジン (VE、Vector Engine)	SX-Aurora TSUBASAの中核であり、ベクトル演算を行う部分です。PCI Expressカードであり、x86サーバーに搭載して使用します。
ベクトルホスト (VH、Vector Host)	ベクトルエンジンを保持するサーバー、つまり、ホストコンピュータを指します。
ベクトルアイランド (VI、Vector Island)	VH1台にVEを1枚ないし複数枚組み込んだ単位を指します。後述のエッジモデル、オンサイトモデルはVI単位でのご利用となります。
エッジモデル	SX-Aurora TSUBASAの製品モデルのうち、デスクサイドに置くことができ、容易にセッティングを行うことができる製品モデルを指します。
オンサイトモデル	SX-Aurora TSUBASAの製品モデルのうち、1Uないし4Uサイズで、サーバーラックに搭載する製品モデルを指します。小規模~大規模まで幅広い構成のユーザーをカバーします。
データセンターモデル	SX-Aurora TSUBASAの製品モデルのうち、これまでのSXの後継に位置する製品モデルを指します。ベクトルエンジンを高密度に搭載した4Uラックマウントサーバーを最大8台搭載したモデルであり、すべてのベクトルエンジンに直接水冷の器具を装着します。
VE1	本書において、Vector EngineのTypeの10の位が1である製品を表します。
VE2	本書において、Vector EngineのTypeの10の位が2である製品を表します。
VE3	本書において、Vector EngineのTypeの10の位が3である製品を表します。
VMC	VE管理コントローラー(VE Management Controller)の略語です。
IB	InfiniBandの略語です。
HCA	Host Channel Adapterの略語です。IBネットワークに接続するためにサーバー側に取り付けるPCIeカードです。
MPI	Message Passing Interfaceの略語です。主にノード間で並列コンピューティングを行うための標準化規格です。同一ノード内であっても、プロセス間の通信にMPIを使用できます。OpenMPとの併用も可能です。
NEC yumリポジトリ	NECのAuroraソフトウェア用のリポジトリ。無償yumリポジトリには無償ソフトウェアが登録されており、誰でもアクセス可能です。アクセス制限付きyumリポジトリには有償ソフトウェアが登録されており、有償ソフトウェアを購入してPPサポートを契約された方のみアクセス可能です。
ライセンスサーバー	SX-Aurora TSUBASA 用ソフトウェアのライセンスの管理を行うサーバー機能です。有償ソフトウェアを使用する場合に必要です。
フロントエンド、または フロントエンドマシン	本書において、SX-Aurora TSUBASAシステムと別に、プログラムのコンパイルを行うこと(およびそのマシン)を指します。

PPサポート	お客様がご購入されたソフトウェア製品の技術サポートを有償にて提供するサポートサービスです。

目　次

はしがき i
用語定義・略語 iii
目　次 v
表目次 ix
図目次 x
第 1 章　はじめに 1
1.1　本書が対象とするソフトウェア 1
1.1.1　ソフトウェア一覧 1
1.1.2　ソフトウェアの入手方法 2
1.2　システム要件 3
1.2.1　ハードウェア 3
1.2.2　対象 OS 3
1.2.3　関連情報 4
1.3　システム構成例 5
1.4　ソフトウェア入手方法の違い 7
1.4.1　有償ソフトウェアを購入して PP サポート契約をしている場合 7
1.4.2　有償ソフトウェアを未購入、または、PP サポート契約をしていない場合 8
1.5　VH へのインストール作業の流れ 9
第 2 章　OS および各種ソフトウェアのインストール 10
2.1　OS のインストール 10
2.1.1　Linux OS のインストール 10
2.1.2　Linux OS の yum リポジトリの設定 10
2.2　動作検証済みカーネルへのアップデート 13
2.3　MLNX_OFED のインストール(選択) 15
2.3.1　モデル B403-8、C401-8 または IB の HCA として ConnectX-7 を搭載する機種を使用する場合 16
第 3 章　SX-Aurora TSUBASA ソフトウェアのインストール 18
3.1　インストールのための事前準備 18
3.1.1　yum リポジトリを使用するために必要なファイル 18
3.1.2　アクセス制限付き yum リポジトリへのアクセス用 ID 18
3.1.3　VH から NEC yum リポジトリへ直接アクセスできない場合 20
3.2　インストールの実行 24

3.2.1 有償ソフトウェアを購入して PP サポート契約あり/オンラインの場合..... 25
3.2.2 有償ソフトウェアを購入して PP サポート契約あり/オフラインの場合..... 27
3.2.3 有償ソフトウェアを未購入、または、PP サポート契約なしの場合 28
3.2.4 シェル変数 TSUBASA_GROUPS に指定可能なグループ 30
3.3 VE ドライバと Linux カーネルの互換性確認..... 31
3.3.1 ダウングレードの要否確認 32
3.3.2 ダウングレードの実行..... 32
3.4 VE 番号定義ファイルの確認・設定 33
3.5 VE の状態確認..... 33
3.6 VMC Firmware (VMCFW) の アップデート..... 34
3.7 コマンド互換パッケージのインストール(選択) 35
3.8 HugePages の設定..... 35
3.9 ScaTeFS の起動(選択) 36
第 4 章 SX-Aurora TSUBASA ソフトウェアの設定 37
4.1 運用ネットワークの設定 38
4.1.1 InfiniBand (IP over IB)を使用する場合 38
4.1.2 Ethernet を使用する場合 39
4.1.3 ネットワークのサービスのリスタート 40
4.2 ライセンスサーバーの指定..... 40
4.3 ライセンスサーバーの設定..... 41
4.4 InfiniBand HCA の Relaxed Ordering 設定(選択) 41
4.5 InfiniBand HCA の PCIe クレジット数の設定(選択) 42
4.6 ScaTeFS の設定..... 44
4.7 NQSV の設定..... 45
4.8 NEC MPI の設定..... 45
4.8.1 SELinux の設定 45
4.8.2 ファイアウォールの設定..... 45
4.8.3 InfiniBand QoS 向けの設定..... 46
4.8.4 NVIDIA SHARP の設定 46
4.8.5 HCA デバイス名の固定化..... 47
4.8.6 各 VE が使用する HCA と通信方式の設定（選択）..... 48
4.8.7 ホスト名の名前解決の設定 49
4.8.8 InfiniBand Adaptive Routing 利用時の設定（選択） 49
4.8.9 Partial Process Swapping 利用時の設定（選択） 49

4.9 NEC Parallel Debugger の設定 49
4.9.1 Eclipse PTP のインストール 50
4.9.2 NEC Parallel Debugger plugin のインストール 50
4.9.3 Eclipse PTP 必須ソフトウェアのインストール 50
4.9.4 ファイアウォールの設定 51
4.10 仮想メモリの設定の確認 52
4.11 memlock リソースの設定の確認 52
4.12 HugePages の設定 53
4.13 Partial Process Swapping 機能の設定 54
4.13.1 veswap.option の設定 54
4.13.2 VEOS の再起動 57
4.14 Process accounting の設定 58
4.14.1 プロセスアカウントサービスの開始 58
4.14.2 プロセスアカウントサービスの停止 58
4.15 SX-Aurora TSUBASA でのプログラム実行について 58
第 5 章 アップデート 60
5.1 VH を運用から外す 63
5.1.1 対象 VH をジョブスケジューラから外す 63
5.1.2 対象 VH を運用監視対象から外す 63
5.2 ScaTeFS Client のアンインストール(選択) 64
5.3 リリースパッケージの更新 66
5.4 yum リポジトリ無効化(OS アップデートを行う場合) 67
5.5 MLNX_OFED のアンインストール(選択) 67
5.6 OS のアップデート(選択) 68
5.7 カーネルのアップデート(選択) 68
5.8 MLNX_OFED のインストール(アンインストールした場合) 69
5.9 yum リポジトリの更新(OS アップデートを行った場合) 69
5.10 不要なパッケージのアンインストール 69
5.11 アップデートの実行 69
5.12 ScaTeFS Client の再インストール(選択) 72
5.13 VE ドライバと Linux カーネルの互換性確認 72
5.13.1 ダウングレードの要否確認 72
5.13.2 ダウングレードの実行 73
5.14 VE 番号定義ファイルの確認・設定 73

5.15 VE の状態確認 74
5.16 VMC Firmware (VMCFW) の アップデート 75
5.17 コマンド互換パッケージのインストール(選択) 76
5.18 ScaTeFS の起動(選択) 76
5.19 ソフトウェアの設定の更新 76
5.20 VH の運用を開始する 77
5.20.1 対象 VH で全ての VE の ONLINE 状態を確認する 77
5.20.2 対象 VH を運用監視対象に入れる 77
5.20.3 対象 VH をジョブスケジューラに追加する 77
第 6 章 アンインストール 79
6.1 VH を運用から外す 79
6.1.1 対象 VH をジョブスケジューラから外す 79
6.1.2 対象 VH を運用監視対象から外す 80
6.2 ScaTeFS Client のアンインストール(選択) 80
6.3 アンインストールの実行 82
付録 A マルチインスタンス対応のソフトウェアのインストール 83
コンパイラ 83
MPI 85
Numeric Library Collection 86
NLCPy 86
付録 B インストールされるパッケージ 87
SX-Aurora ソフトウェアパッケージ一覧 87
パッケージグループに含まれるパッケージ 91
付録 C ネットワーク構成 96
運用ネットワーク 96
管理ネットワーク 97
付録 D glibc 環境への移行のお願い 99
付録 E HugePages の手動設定方法 100
(1) HugePages 数の計算 100
(2) HugePages の設定 101
付録 F 発行履歴 103
発行履歴一覧表 103
追加・変更点詳細 103
索 引 104

表目次

表 1 ソフトウェア一覧 1
表 2 ソフトウェアコンポーネントと入手方法 2
表 3 サーバーモデル 3
表 4 MLNX_OFED の対応バージョン 15
表 5 パッケージグループ 30
表 6 パッケージグループ/NQSV 31
表 7 パッケージグループ/ScaTeFS Client 31
表 8 パッケージグループ/ScaTeFS Server 31
表 9 SX-Aurora TSUBASA ソフトウェアの設定 37
表 10 ライセンスサーバーの指定 40
表 11 ライセンスサーバーの指定（環境変数） 40
表 12 veswap.options ファイルの設定内容 54
表 13 インストールパッケージ一覧 87
表 14 パッケージグループ/InfiniBand for SX-Aurora TSUBASA 91
表 15 パッケージグループ/VE Application 91
表 16 パッケージグループ/SDK (MPI 以外) 93
表 17 パッケージグループ/SDK (MPI) 95
表 18 必要な HugePages のページ数 100

図目次

図 1 構成１：スタンドアロン（VI 1 台）で利用する場合 5
図 2 構成 2：複数台の VI、管理用サーバー、およびフロントエンドマシンを用意する場合 5
図 3 構成 3：大規模構成 6
図 4 yum リポジトリからのインストール 7
図 5 ローカル環境の yum リポジトリからのインストール 8
図 6 インターネット配信製品ダウンロードサービスでダウンロードしたパッケージのインストール 8
図 7 インストール作業 9
図 8 シリアル No カードの例 19
図 9 アップデート作業 62
図 10 ネットワーク構成 96

第1章　はじめに

1.1　本書が対象とするソフトウェア

1.1.1　ソフトウェア一覧

本書が対象とするのは、SX-Aurora TSUBASA システムの運用に必要なソフトウェアである下表のソフトウェアです。

表 1 ソフトウェア一覧

ソフトウェア名	説明
VEOS	VE制御ソフトウェア
MMM	Monitoring & Maintenance Manager
VMC Firmware	VMCファームウェア
SX-Aurora TSUBASA用InfiniBand	InfiniBand制御ソフトウェア
ライセンスサーバー	ライセンス管理ソフトウェア
ライセンスアクセスライブラリ	ライセンスチェック用ライブラリ
NEC Software Development Kit for Vector Engine **（本書では以降 SDK と記載）**	ソフトウェア開発キット SDK Community Edition（本書では以降SDK-CEと記載）とSDK Standard Edition（本書では以降SDK-SEと記載）の2種類があり、SDK-CEは2023年1月にリリースされた無償のSDK、SDK-SEは2023年1月以前から購入可能な有償のSDKで以下の旧型番の製品を指します。 UWAA00-N10-I UWAA00-N11-I UWAA00-N12-I UWAA00-N1U-I なお、単にSDKと記載した場合は、SDK-CE、SDK-SEの両方を指します。
SDKランタイム	VEプログラムおよびMPIプログラムの実行に必要なSDKのランタイムライブラリおよびコマンド
NEC Compat C++ **標準ライブラリ** for Vector Engine **（本書では以降 Compat C++と記載）**	Version 3.5.x以前のC++コンパイラとの互換のためのC++標準ライブラリ SDK-CEの拡張ライブラリで、有償ソフトウェアであり、SDK-CEとは別に購入しなければなりません。
NEC Scalable Technology File System **（本書では以降 ScaTeFS と記載）**	スケーラブルファイルシステム
NEC Network Queuing System V **（本書では以降NQSV と記載）**	バッチ実行制御システム

注1）ScaTeFSは、運用ネットワークにInfiniBand を使用する環境でのみ利用できます。運用ネットワークにEthernetのみを使用する環境では利用できません。

1.1.2　ソフトウェアの入手方法

SX-Aurora TSUBASA システムの運用に必要なソフトウェアは、入手方法に応じて以下の二種類に分けられます。なお、アクセス制限付き yum リポジトリへのアクセス条件は、1.4.1 を参照して下さい。

A) 無償ソフトウェア: アクセス制限のない yum リポジトリ(本書では無償 yum リポジトリと記載)から yum コマンドを使ってインストール

B) 有償ソフトウェア: PP サポートの契約をしている場合、アクセス制限付き yum リポジトリから yum コマンドを使ってインストール。そうでない場合は、「インターネット配信製品ダウンロードサービス」で入手したパッケージをインストール

本書では以降、無償 yum リポジトリ、および、アクセス制限付き yum リポジトリをまとめて NEC yum リポジトリと記載します。

各ソフトウェアに含まれるコンポーネント、および、その入手方法は以下のとおりです。

表 2 ソフトウェアコンポーネントと入手方法

ソフトウェア名	含まれるコンポーネント	入手方法
VEOS	VEOS	A
MMM	MMM	A
VMC Firmware	VMC Firmware	A
SX-Aurora TSUBASA 用 InfiniBand	SX-Aurora TSUBASA用InfiniBand	A
ライセンスサーバー	ライセンスサーバー	A
ライセンスアクセスライブラリ	ライセンスアクセスライブラリ	A
SDK	C/C++コンパイラ Fortranコンパイラ binutils Numeric Library Collection NLCPy mpi4py-ve NEC Parallel Debugger (*) チューニングツール NEC MPI (NEC MPI/Scalar-Vector Hybridを含む(注1))	A (SDK-CE) B (SDK-SE) (*)はBのみ
SDKランタイム	SDKに含まれるbinutils、ランタイムライブラリおよびMPI実行コマンド	A
Compat C++	NEC Compat C++ 標準ライブラリ for Vector Engine	B
ScaTeFS	ScaTeFS/Client	B
NQSV	NQSV/JobServer NQSV/Client	B

注1)　NEC MPI/Scalar-Vector HybridはNEC MPIとパッケージが共通なのでインストール作業は不要です。

1.2　システム要件

1.2.1　ハードウェア

SX-Aurora TSUBASA システムの運用を行うためにベクトルエンジンを搭載するためのサーバーとして、以下のモデルがあります。

表 3 サーバーモデル

	エッジモデル	オンサイトモデル				データセンターモデル
モデル名称(注1)	A100-1 A101-1 A111-1	A300-2	A300-4 A311-4	A300-8 A311-8 B300-8 B302-8	A412-8 B401-8 B403-8 C401-8	A500-64 A511-64
最大搭載ベクトルエンジン(VE)数	1	2	4	8	8	64
ベクトルホスト(VH)数	1	1	1	1	1	8

注1）　モデル名称Axxx-xに搭載されているVEはType 1*(VE1)。モデル名称Bxxx-xに搭載されているVEはType 2*(VE2)。モデル名称Cxxx-xに搭載されているVEはType 3*(VE3)。インストールするパッケージは搭載VEに対応したものを選択してください。

詳細については、SX-Aurora TSUBASA の製品カタログを参照してください。

⚠ 注意　VHのBIOS設定

Boot Modeは、工場出荷時設定値のUEFIのままお使いください。本機では、UEFI以外のモードをサポートしておりません。

1.2.2　対象 OS

SX-Aurora TSUBASA システムは、RedHat 系のオペレーティングシステム上で動作します。

SX-Aurora TSUBASA システムのソフトウェアの動作検証済みの LinuxOS とカーネルバージョンについては NEC サポートポータルの下記のコンテンツをご確認ください。

【SX-Aurora TSUBASA】検証済み Linux カーネル
https://www.support.nec.co.jp/View.aspx?id=3140106285

上記コンテンツに記載のとおり、対応カーネルはアップデート版のみとなります。OS インストーラでインストールされる標準カーネルには対応していませんので、必ずカーネルをアップデートしてご利用ください。また動作検証していないカーネルへのアップデートを避けるため、　設定ファイル /etc/yum.conf において kernel パッケージのアップデートを対象外としてください。設定ファイルの記述方法については、2.2 節を参照してください。

1.2.3　関連情報

以下のサイトにて SX-Aurora TSUBASA に関する各種情報を公開していますので、合わせてご確認ください。

- NEC Aurora Forum(https://sxauroratsubasa.sakura.ne.jp/)

1.3　システム構成例

SX-Aurora TSUBASA の構成例を紹介します。

(1)　構成 1 ：　スタンドアロン（ベクトルアイランド(VI) 1台）で利用する場合

以下のソフトウェアをベクトルホスト(VH)上にインストールしてください。

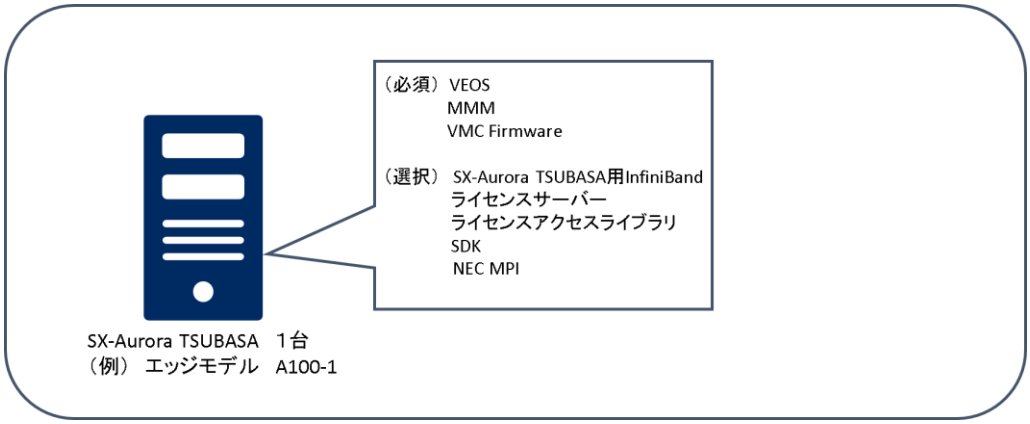


図 1 構成 1 ：スタンドアロン（VI 1 台）で利用する場合

(2)　構成 2 ：　複数台のVI、管理用サーバー、およびフロントエンドマシンを用意する場合

管理用サーバーでソフトウェアのライセンス管理を行い、フロントエンドマシンはプログラムのコンパイルを行うために使用する例です。

各VH、管理用サーバー、およびフロントエンドにインストールするソフトウェアを示します。

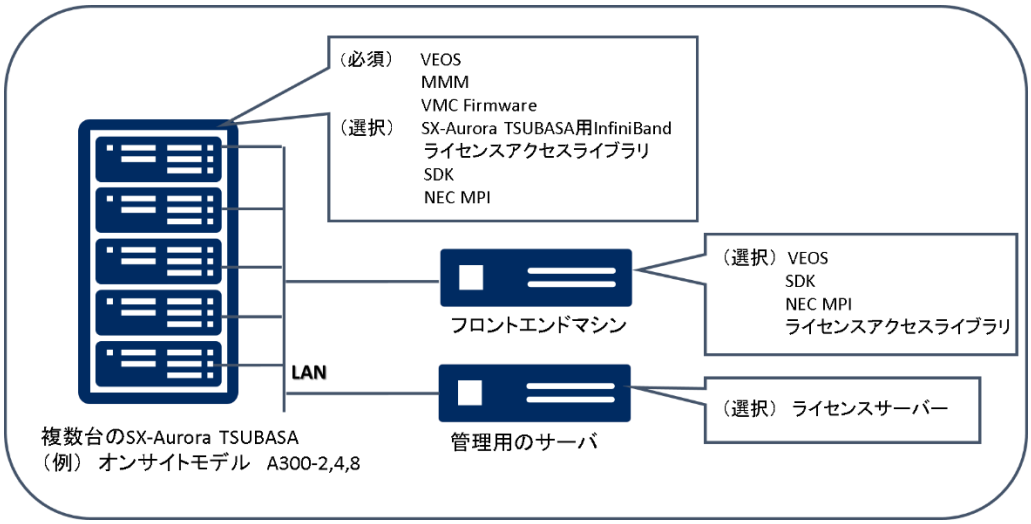


図 2 構成 2：複数台の VI、管理用サーバー、およびフロントエンドマシンを用意する場合

（3）　構成３：　大規模構成

営業／SEまで、ご相談下さい。

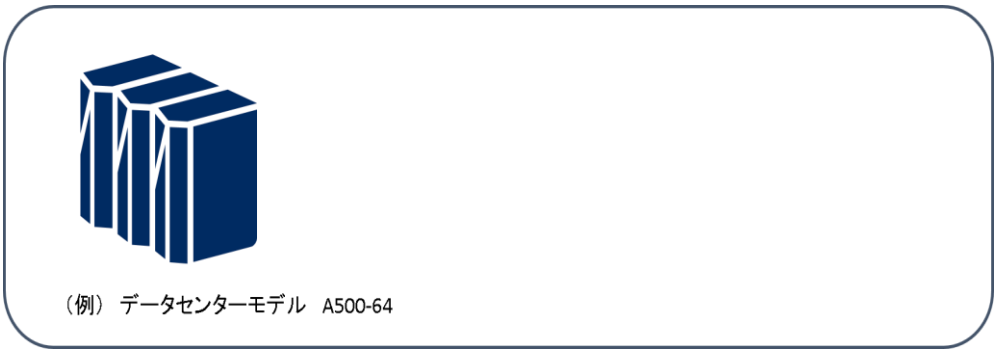


図 3 構成 3：大規模構成

1.4　ソフトウェア入手方法の違い

インストール作業は、有償ソフトウェア(1.1.2)を購入しており、PP サポートの契約をしている場合とそうでない場合とで手順が異なります。また、VH から NEC yum リポジトリに直接アクセスする場合としない場合とで手順が異なります。

1.4.1　有償ソフトウェアを購入して PP サポート契約をしている場合

有償ソフトウェアを購入しており、PP サポートの契約をしている場合は、アクセス制限付き yum リポジトリにアクセスしてインストールします。有償ソフトウェアが登録されている yum リポジトリへのアクセスには、有償ソフトウェアのサポートパックのシリアル No.が必要です。シリアル No.については、3.1.2 節を参照してください。

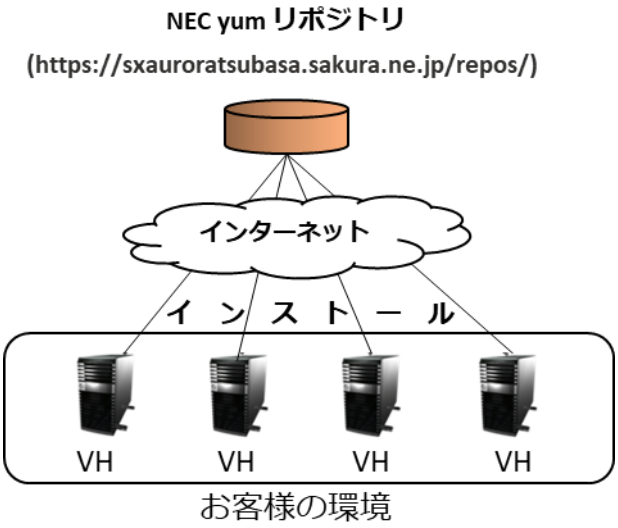


図 4 yum リポジトリからのインストール

インターネットに直接アクセスできない場合は、ローカル環境に yum リポジトリを構築することでパッケージをインストールできます。ローカルに yum リポジトリを構築する手順は、3.1.3 節をご参照ください。

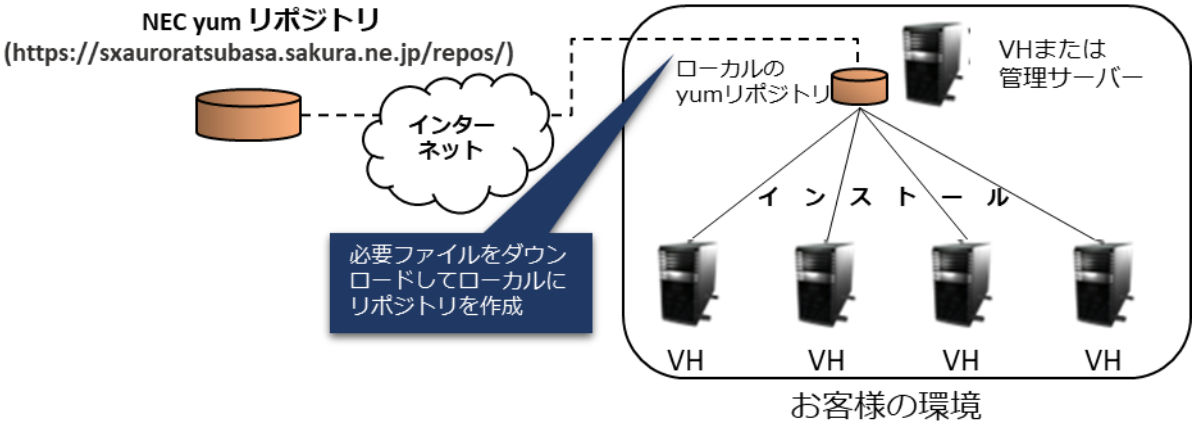


図 5 ローカル環境の yum リポジトリからのインストール

1.4.2　有償ソフトウェアを未購入、または、PP サポート契約をしていない場合

有償ソフトウェアを購入しており、PP サポートの契約をしていない場合は、「インターネット配信製品ダウンロードサービス」より入手した有償ソフトウェアのパッケージを yum コマンドでインストールします。

無償ソフトウェアは、無償 yum リポジトリにアクセスしてインストールします(図 4 参照)。インターネットに直接アクセスできない場合は、ローカル環境に無償ソフトウェアの yum リポジトリを構築することでパッケージをインストールできます(図 5 参照)。

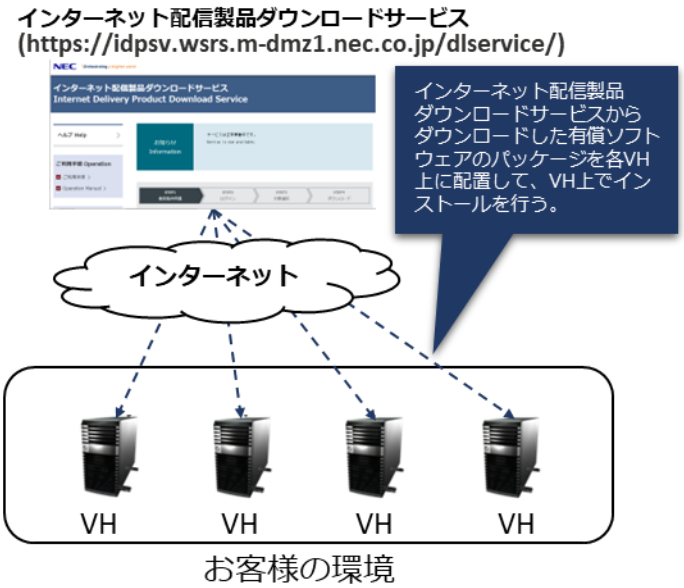


図 6 インターネット配信製品ダウンロードサービスでダウンロードしたパッケージのインストール

1.5　VHへのインストール作業の流れ

SX-Aurora TSUBASA システムの運用に必要なソフトウェアの VH へのインストール作業の流れを以下に示します。この手順でのインストールの詳細は第 2 章をご参照ください。また、インストール後のソフトウェアの設定については第 4 章をご参照ください。

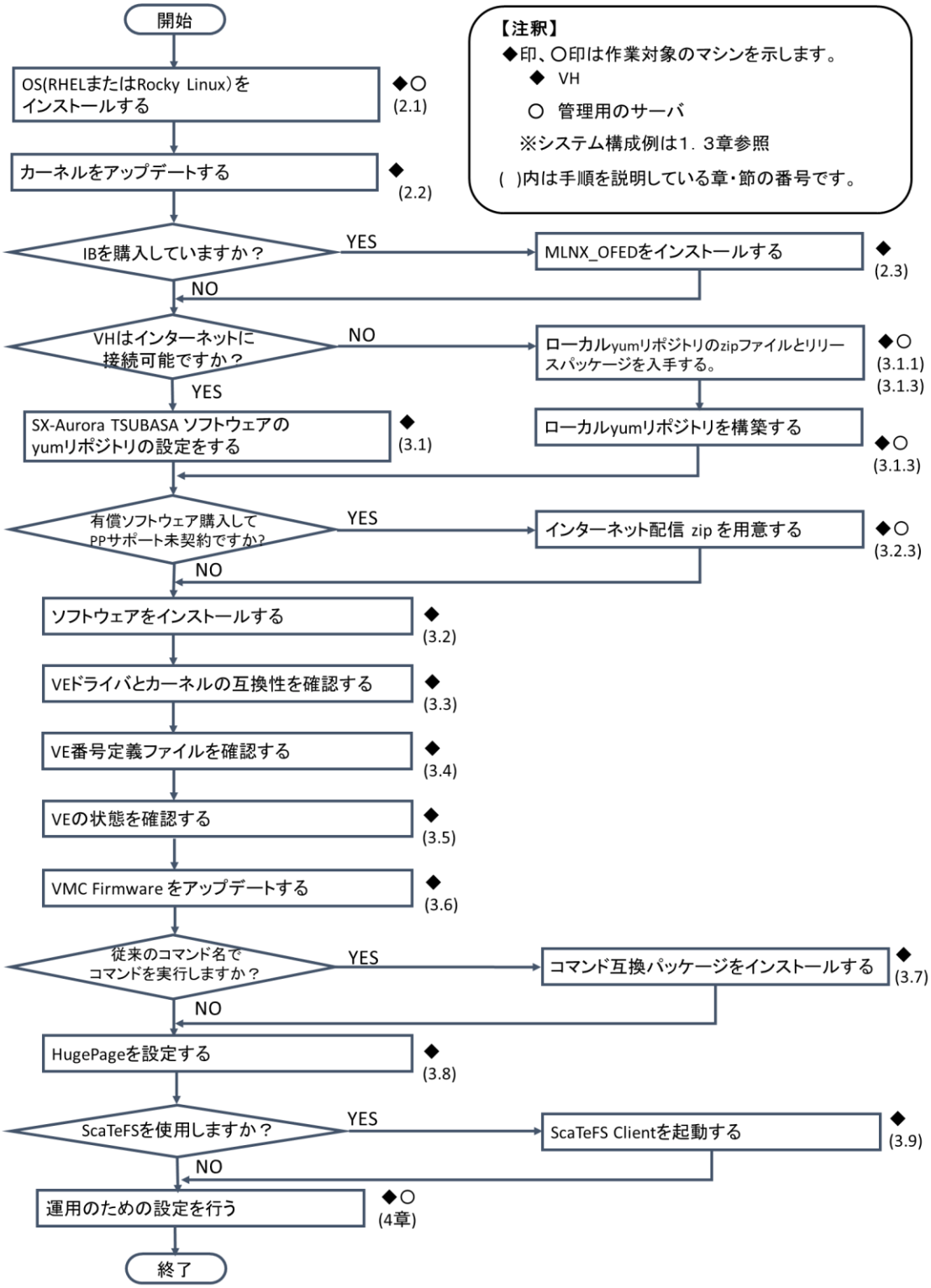


図 7 インストール作業

第2章　OS および各種ソフトウェアのインストール

本章では、Aurora ソフトウェアを利用するための OS インストールおよび各種ソフトウェアのインストールの手順を説明します。

VH がインターネットに接続している場合は、インターネット上の yum リポジトリをオンラインで利用することができます（図 4 参照）。

VH がインターネットに直接アクセスできない場合は、有償ソフトウェア、無償ソフトウェアともに、インターネットにアクセスできる環境でダウンロードしたソフトウェアパッケージを利用して、VH または管理サーバー上に yum リポジトリを構築することでインストールを行います（図 5 参照）。

2.1　OSのインストール

2.1.1　Linux OS のインストール

Aurora がサポート中の OS（Linux ディストリビューションとカーネルのバージョンの組み合わせ）を、OS のインストール方法に従って、VH にインストールしてください。

- Aurora がサポート中の OS（【SX-Aurora TSUBASA】検証済み Linux カーネル）
  https://www.support.nec.co.jp/View.aspx?id=3140106285
- OS のインストール方法
  - Red Hat Enterprise Linuxの場合
    * Product Documentation for Red Hat Enterprise Linux 8 -- Installing
  - Rocky Linuxの場合
    * Rocky Linux Documentation, GuidesのInstalling Rokcy Linuxなど

2.1.2　Linux OS の yum リポジトリの設定

SX-Aurora TSUBASA のソフトウェアをインストールするために、Linux OS に関する追加のパッケージを yum コマンドによってインストールできるよう、Linux OS の yum リポジトリの設定をしておいてください。yum リポジトリの設定については、Linux OS のインストール媒体を利用する方法とインターネット上の公式リポジトリを利用する方法があります。

ここでは、Rocky Linux 8 の OS のインストール媒体（DVD）を使用した yum リポジトリを利用する際の設定の手順について説明します。その他の OS については、リポジトリの名前、ファイル名、GPG キーファイルを対象の OS に合わせて変更してください。

(1)　OSインストール媒体（DVD）のマウント

OSインストール媒体（DVD）をDVDドライブにセットして適当なディレクトリにマウントしてください。以下の例では/media/cdromディレクトリにマウントします。

```
# mkdir /media/cdrom
# mount /dev/cdrom /media/cdrom
```

(2)　yumリポジトリの設定

OSインストール媒体（DVD）からyumコマンドでパッケージをインストールできるようにするため、既存のリポジトリ設定ファイルを退避して、OSインストールDVD用のリポジトリ設定ファイルを設定します。

```
# cd /etc/yum.repos.d
# ls
Rocky-AppStream.repo          Rocky-FastTrack.repo
Rocky-BaseOS.repo             Rocky-HighAvailability.repo
Rocky-ContinuousRelease.repo  Rocky-Media.repo
Rocky-Debuginfo.repo          Rocky-Plus.repo
Rocky-Devel.repo              Rocky-PowerTools.repo
Rocky-Extras.repo             Rocky-Sources.repo
# mkdir Rocky-repos.d
# mv *.repo Rocky-repos.d
# ls
Rocky-repos.d
# cp Rocky-repos.d/Rocky-Media.repo .
```

/etc/yum.repos.d/Rocky-Media.repo を以下のように修正します。

```
[media-baseos]
name=Rocky Linux $releasever - Media - BaseOS
baseurl=file:///media/cdrom/BaseOS
gpgcheck=1
enabled=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-rockyofficial

[media-appstream]
name=Rocky Linux $releasever - Media - AppStream
baseurl=file:///media/cdrom/AppStream
gpgcheck=1
enabled=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-rockyofficial
```

SX-Aurora TSUBASA ソフトウェアのインストールが終わるまで、DVD は mount しておいてください。

2.2　動作検証済みカーネルへのアップデート

VH の Linux カーネルを動作検証済みのカーネルにアップデートし、VH を再起動します。

SX-Aurora TSUBASA システムのソフトウェアの動作検証済みの LinuxOS とカーネルバージョンについては NEC サポートポータルの下記のコンテンツをご確認ください。

【SX-Aurora TSUBASA】検証済み Linux カーネル
https://www.support.nec.co.jp/View.aspx?id=3140106285

動作検証していないカーネルへのアップデートを避けるため、/etc/yum.conf において kernel パッケージのアップデートを対象外としてください。

/etc/yum.conf への記載例を下記に示します。'exclude=kernel*' を yum.conf に追記し、kernel パッケージのアップデートを抑止します。

```
# vi /etc/yum.conf
[main]
exclude=kernel*
 …
```

Rocky Linux 8.10 において、検証済みカーネル(kernel-4.18.0-553.8.1.el8_10.x86_64)をインストールする場合は、以下のように実行します。

(1) Rocky Linux 8.10用のリポジトリの設定

以下のように/etc/yum.repos.d/Rocky-BaseOS.repoを作成して、BaseOS リポジトリの接続先を http://dl.rockylinux.org/pub/rocky/8.10/ に設定します。ファイルを編集して、
"mirrorlist=https://mirrors.rockylinux.org/mirrorlist?arch=$basearch&repo=BaseOS-$releasever"
をコメントアウトして、
"baseurl=http://dl.rockylinux.org/pub/rocky/8.10/BaseOS/$basearch/os/"
を追加します。

```
# cd /etc/yum.repos.d
# cp Rocky-repos.d/Rocky-BaseOS.repo .
# vi Rocky-Linux-BaseOS.repo
# diff Rocky-repos.d/Rocky-BaseOS.repo Rocky-BaseOS.repo
13c13
<
mirrorlist=https://mirrors.rockylinux.org/mirrorlist?arch=$basearch&repo=BaseOS-$releasever
---
```

```
>
# mirrorlist=https://mirrors.rockylinux.org/mirrorlist?arch=$basearch&repo=BaseOS-$releasever
14a15
> baseurl=http://dl.rockylinux.org/pub/rocky/8.10/BaseOS/$basearch/os/
```

(2) カーネル(kernel-4.18.0-553.8.1.el8_10.x86_64)をインストールします。

```
#   dnf   --disableexcludes=all   install   kernel-4.18.0-553.8.1.el8_10   kernel-headers-4.18.0-
553.8.1.el8_10
Last metadata expiration check: 0:03:09 ago on Wed Sep 25 10:01:15 2024.
Dependencies resolved.
================================================================================
 Package           Arch       Version                   Repository   Size
================================================================================
Installing:
 kernel            x86_64     4.18.0-553.8.1.el8_10       baseos      10 M
 kernel-headers     x86_64     4.18.0-553.8.1.el8_10       baseos      12 M
Installing dependencies:
 kernel-core        x86_64     4.18.0-553.8.1.el8_10       baseos      43 M
 kernel-modules     x86_64     4.18.0-553.8.1.el8_10       baseos      36 M

Transaction Summary
================================================================================
Install  4 Packages

Total download size: 101 M
Installed size: 102 M
Is this ok [y/N]: y
    :
Installed:
  kernel-4.18.0-553.8.1.el8_10.x86_64
  kernel-core-4.18.0-553.8.1.el8_10.x86_64
  kernel-headers-4.18.0-553.8.1.el8_10.x86_64
  kernel-modules-4.18.0-553.8.1.el8_10.x86_64

Complete!

# reboot
```

⚠ **注意**

ホストマシンのカーネルバージョンについて、原則として動作検証済みのカーネルの中で最新のものにアップデートすることを推奨します。最新バージョンをご利用されない場合、Aurora ソフトウェアをインストールしても、ve_drv-kmodパッケージに含まれるve_drvモジュールがロードされずにVEがONLINEにならない場合があります。その場合、vp-kmod、ve_drv-kmod パッケージをダウングレードする必要があります。詳細は「VEドライバとLinuxカーネルの互換性確認」を参照してください。

2.3　MLNX_OFEDのインストール(選択)

InfiniBand を使用するシステムでは、インストール対象 VH に MLNX_OFED をインストールしてください。

使用する VH の OS とインストールする MLNX_OFED のバージョン対応は以下のとおりです。対応するバージョンの MLNX_OFED をインストールしてください。

表 4 MLNX_OFED の対応バージョン

OS	MLNX_OFEDのバージョン	対応VE	
		VE1/VE2	VE3
RHEL/Rocky Linux 8.10	MLNX_OFED 23.10-3.2.2.0-LTS	○	○

MLNX_OFED の ISO イメージとドキュメントは、次の NVIDIA 公式ホームページから入手できます。

https://network.nvidia.com/products/infiniband-drivers/linux/mlnx_ofed/

⚠ **注意**

- モデルC401-8またはIBのHCAとしてConnectX-7を搭載する機種を利用する場合、MLNX_OFED 23.10-3.2.2.0-LTSをインストールする際に"--without-fw-update"オプションを付けてインストールしてください。このオプションを指定せずにMLNX_OFED 23.10-3.2.2.0-LTSをインストールすると、自動的にHCA-FWは28.39.3560がインストールされます。HCA-FW 28.39.3560がインストールされた場合は、再起動後、HCA-FWをダウングレードしてください(2.3.1参照)。
- MLNX_OFED をインストールする前に、Release NotesのHardware and Software Requirementsを参照し、必要なパッケージを事前にインストールしてください。また、MLNX_OFEDのインストール中に不足するパッケージが表示されてエラーになった場合、表示されたパッケージをインストールしてください。
  MLNX_OFEDのドキュメントは以下から参照できます。
  https://docs.nvidia.com/networking/software/adapter-software/index.html
- カーネルのアップデートを行っている場合、mlnxofedinstallによるインストールが失敗することがあります。この場合、mlnx_add_kernel_support.shを実行するか、--add-kernel-supportオプションを付けてmlnxofedinstallを実行する必要があります。これらのコマンドを実行する際には--kmpオプションを追加してください。
- mlnxofedinstall --add-kernel-support --kmpの実行に、kernel-rpm-macrosパッケージが必要になる場合があります。この場合、/etc/yum.confのexclude=kernel*を一時的にコメ

ントアウトしてからkernel-rpm-macrosパッケージをインストールしてください。

- InfiniBandに依存するソフトウェアがインストールされている場合、以下のメッセージが出力されてインストールが失敗することがあります。この場合、表示されたパッケージをアンインストールしてからインストールを再実行してください。

```
Error: One or more packages depends on MLNX_OFED_LINUX.
These packages should be removed before uninstalling MLNX_OFED_LINUX:
```

インストール後、InfiniBand ドライバのリロードまたは VH の再起動を行ってください。

```
# /etc/init.d/openibd restart
```

⚠ 注意

- ファームウェアのアップデートが行われた場合VHの再起動は必須です。詳しくはMLNX_OFED のドキュメントを参照してください。
- InfiniBandドライバのリロードに失敗した場合、VHを再起動してください。
- 再起動後、MLNX_OFEDが動作しているかどうかは以下のコマンドで確認できます。

```
# systemctl status openibd
```

2.3.1　モデル B403-8、C401-8 または IB の HCA として ConnectX-7 を搭載する機種を使用する場合

モデル B403-8、C401-8 または IB の HCA として ConnectX-7 を搭載する機種を使用する場合、IB HCA(NDR200 対応 ConnectX-7)の HCA-FW は 28.39.1002 を使用してください。

1. HCA-FW のバージョン確認

   HCA-FW のバージョンが 28.39.1002 かどうか確認してください。

```
# ibstat | grep -A2 MT4129
        CA type: MT4129
        Number of ports: 1
        Firmware version: 28.39.1002
--
        CA type: MT4129
        Number of ports: 1
        Firmware version: 28.39.1002
```

2. HCA-FW の更新

   表示された Firmware version が 28.39.1002 と異なる場合は、下記の NVIDIA のサイトから

Firmware version 28.39.1002 をダウンロードし、インストールしてください。

https://network.nvidia.com/support/firmware/connectx7

- ダウンロード手順

  ARCHIVE VERSION タブをクリックする

  →Version 28.39.1002 , OPN MCX75310AAS-HEA, PSID MT_0000000844 を選択する

  → Download fw-ConnectX7-rel-28_39_1002-MCX75310AAS-HEA_Ax-UEFI-14.32.12-FlexBoot-3.7.201.signed を選択して HCA-FW をダウンロードする

  ダウンロード後、zip 圧縮されているファイルを展開してください。

- HCA-FW のアップデート手順

  表示された HCA-FW のバージョンが 28.39.1002 よりも古い場合、インストール対象の VH で、root 権限で以下を実行してください。

```
# mst start
# mlxfwmanager -d /dev/mst/mt4129_pciconf0 -i fw-ConnectX7-rel-28_39_1002-MCX75310AAS-HEA_Ax-UEFI-14.32.12-FlexBoot-3.7.201.signed.bin -u -y
# mlxfwmanager -d /dev/mst/mt4129_pciconf1 -i fw-ConnectX7-rel-28_39_1002-MCX75310AAS-HEA_Ax-UEFI-14.32.12-FlexBoot-3.7.201.signed.bin -u -y
```

  適用後、VH を再起動してください。

```
# reboot
```

- HCA-FW のダウングレード手順

  表示された HCA-FW のバージョンが 28.39.1002 よりも新しい場合、インストール対象の VH で、root 権限で以下を実行してください。

```
# mst start
# mlxfwmanager -d /dev/mst/mt4129_pciconf0 -i fw-ConnectX7-rel-28_39_1002-MCX75310AAS-HEA_Ax-UEFI-14.32.12-FlexBoot-3.7.201.signed.bin -u -y -f
# mlxfwmanager -d /dev/mst/mt4129_pciconf1 -i fw-ConnectX7-rel-28_39_1002-MCX75310AAS-HEA_Ax-UEFI-14.32.12-FlexBoot-3.7.201.signed.bin -u -y -f
```

  適用後、VH を再起動してください。

```
# reboot
```

第3章　SX-Aurora TSUBASA ソフトウェアのインストール

3.1　インストールのための事前準備

SX-Aurora TSUBASA ソフトウェアは yum リポジトリから yum コマンドを用いてインストールします。このため、事前にそれらを利用できるよう準備します。

yum リポジトリは、インターネット上のものをオンラインで利用する方法と、VH 上または管理サーバー上に構築してオフラインで利用する方法とがあります。

⚠ 注意

インターネット上の yumリポジトリへのアクセスにおいては、本サービスの安定稼働確保および不正利用防止を目的に、アクセス元のIPアドレスとともにアクセス用IDの情報を記録しています。

3.1.1　yum リポジトリを使用するために必要なファイル

オンライン、またはオフラインで使用するいずれの場合も、以下の設定ファイル、スクリプトを含む rpm ファイルを事前にダウンロードしておきます。この rpm ファイルに含まれるアーキテクチャ設定ファイルがデフォルトのアーキテクチャを決定します。アーキテクチャ設定ファイルの情報をもとに、NEC コンパイラがコンパイル時に VE1 または VE3 アーキテクチャのバイナリを生成します。

- デフォルトのアーキテクチャを VE1 にする場合
  https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve1-3.0-1.noarch.rpm
- デフォルトのアーキテクチャを VE3 にする場合
  https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve3-3.0-1.noarch.rpm

上記の rpm に以下のファイルが含まれます。

- /etc/opt/nec/ve/default.conf　アーキテクチャ設定ファイル
- /etc/pki/rpm-gpg/RPM-GPG-KEY-TSUBASA-soft　GPG公開鍵
- /etc/yum.repos.d/TSUBASA-restricted.repo　アクセス制限付きyumリポジトリ設定ファイル
- /etc/yum.repos.d/TSUBASA-repo.repo　無償yumリポジトリ設定ファイル
- /opt/nec/ve/sbin/TSUBASA-groups-remark.sh　グループ状態更新スクリプト
- /opt/nec/ve/sbin/terminate-all-ve-services　サービス停止スクリプト
- /opt/nec/ve/sbin/start-all-ve-services　サービス再開スクリプト
- /opt/nec/ve/sbin/setup-ve-infiniband.sh　InfiniBand設定スクリプト

3.1.2　アクセス制限付き yum リポジトリへのアクセス用 ID

有償ソフトウェアが含まれるアクセス制限付き yum リポジトリへアクセスするためには、各有償ソフトウェアのサポートパックの購入またはサポート契約が必要です。(1.4.1)

契約内容に基づいて、アクセス制限付き yum リポジトリ設定ファイル(/etc/yum.repos.d/TSUBASA-restricted.repo)中の、各 yum リポジトリの設定欄を編集します。「PP・サポートサービス 製品シリアル No.カード」に記載されたシリアル No.の 16 桁（"-"を除く）を username に、右側８桁を password に入力し、enabled は 1 に設定してください。シリアル No.カードは、サポートパックを購入されたお客様は、「インターネット配信製品ダウンロードサービス」からダウンロードできるファイル中に含まれています。サポート契約されたお客様は、営業にお問合せ願います。以下は、シリアル No.カードの例です。

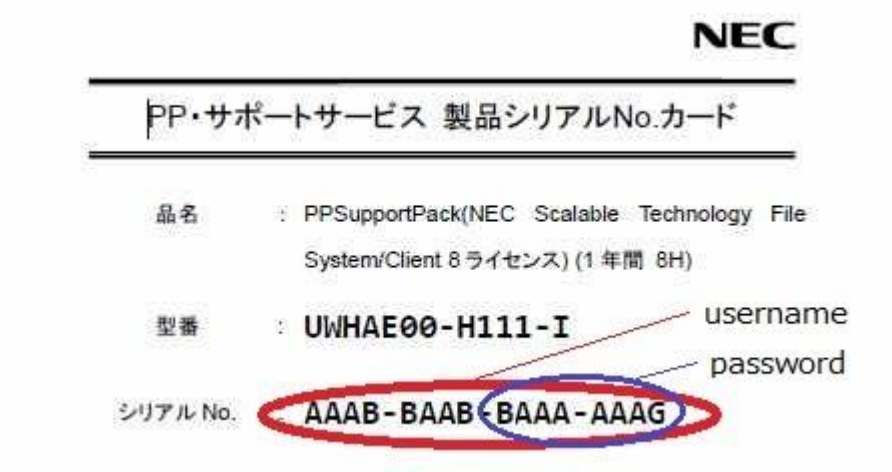


図 8 シリアル No カードの例

- オンラインの場合

アクセス制限付き yum リポジトリ設定ファイルの編集例は、下記のとおりです。アクセス用 ID は有償ソフトウェアごとに異なります。

```
# vi /etc/yum.repos.d/TSUBASA-restricted.repo
[nec-sdk]
  …
username=<NEC SDKのシリアルNo>
password=<NEC SDKのシリアルNoの右側の8桁>
enabled=1     ←0を1に変更
  …
[nec-mpi]
  …
username=<NEC SDKのシリアルNo　※>
password=<NEC SDKのシリアルNoの右側の8桁　※>
enabled=1     ←0を1に変更
  …
```

※2021 年 1 月以降に PPSP for NEC SDK をご購入されている場合は、nec-mpi に PPSP for NEC SDK のシリアル No を設定してください。2020 年 12 月以前に PPSP for NEC SDK をご購入されている場合は、nec-mpi に PPSP for NEC MPI のシリアル No を設定してください。
※PP サポートを契約されている方で、SDK-SE または Compat C++をご購入されていない場合は、nec-sdk の enabled を 0 に設定してください。

- オフラインの場合
  次節で説明するローカル yum リポジトリ構築用 zip ファイルをダウンロードする際に、ブラウザから入力を求められるユーザー名とパスワードに指定してください。

3.1.3　VH から NEC yum リポジトリへ直接アクセスできない場合

VH からオンラインでインターネット上の NEC yum リポジトリへ直接アクセスできない場合、VH 上または管理サーバー上に yum リポジトリを構築する必要があります。オンラインでアクセスできる場合、本項の手順は不要です。

(1) ローカル yum リポジトリ構築用 zip ファイル

yum リポジトリが含まれる zip ファイルは以下の URL へのアクセスにより入手できます。

RHEL/Rocky Linux 8.*x* の *x* はお客様がご利用の RHEL/Rocky Linux のバージョンをご確認ください。

https://sxauroratsubasa.sakura.ne.jp/repos/ 配下

	RHEL/Rocky Linux 8.x 系
共通(yum リポジトリ設定ファイル等)	TSUBASA-soft-release-ve1-3.0-1.noarch.rpm または TSUBASA-soft-release-ve3-3.0-1.noarch.rpm
VEOS, MMM 等	TSUBASA-repo_el8.x.zip
SDK-CE	community/sdk/sdk_el8.zip community/mpi/mpi_mofed5_el8.zip
SDK-SE, Compat C++	restricted/sdk/sdk_el8.zip restricted/mpi/mpi_mofed5_el8.zip
SDK ランタイム	runtime/sdk/sdk_el8.zip runtime/mpi/mpi_mofed5_el8.zip
NQSV	restricted/nqsv/nqsv_el8.zip
ScaTeFS	restricted/scatefs/scatefs_el8.x.zip

community、および、restricted 配下の SDK-CE、SDK-SE、Compat C++を取得するための zip ファイルは、runtime 配下の同名の zip ファイルの内容を含みます。そのため、community、または、restricted から zip ファイルを取得する場合、runtime からの zip ファイルは取得不要です。

アクセス制限付き yum リポジトリ配布ページへアクセスする際は PP サポートのユーザー名(シリアル No.の 16 桁)とパスワード(シリアル No.の右側 8 桁)が必要になります。

以下より yum リポジトリの設定ファイルを含むリリースパッケージを入手してください。なお、リポジトリの設定変更が含まれるため、更新の場合も入手してください。

- デフォルトでVE1向けのバイナリを生成する場合
  TSUBASA-soft-release-ve1-3.0-1.noarch.rpm
- デフォルトでVE3向けのバイナリを生成する場合
  TSUBASA-soft-release-ve3-3.0-1.noarch.rpm

VEOS, MMM 等、および、ScaTeFS を利用する場合、以下より RHEL/Rocky Linux のバージョンに応じてリリースパッケージを入手して下さい。

	OSバージョン	ファイルリンク
VEOS, MMM等	8.10	TSUBASA-repo_el8.10.zip
ScaTeFS	8.10	restricted/scatefs/scatefs_el8.10.zip

RHEL/Rocky Linux 8.x 系上で SDK、NQSV を利用する場合、以下のリリースパッケージを入手してください。

		ファイルリンク（RHEL/Rocky Linux 8.x系向け）
SDK (MPI以外)	SDK-CE	community/sdk/sdk_el8.zip
	SDK-SE, Compat C++	restricted/sdk/sdk_el8.zip
	SDKランタイム	runtime/sdk/sdk_el8.zip
SDK (MPI)	SDK-CE	community/mpi/mpi_mofed5_el8.zip
	SDK-SE, Compat C++	restricted/mpi/mpi_mofed5_el8.zip
	SDKランタイム	runtime/mpi/mpi_mofed5_el8.zip
NQSV	NQSV	restricted/nqsv/nqsv_el8.zip

(2) ローカル yum リポジトリの構築

SX-Aurora TSUBASA ソフトウェア用の yum リポジトリを構築するホスト（以降、リポジトリサーバーと呼びます）上で、以下の手順により SX-Aurora TSUBASA ソフトウェア用の yum リポジトリを構築してください。

スタンドアロン構成の場合、VH 上に yum リポジトリを構築します。

複数の VH が存在する場合は、そのうちの 1 台または管理サーバー上に yum リポジトリを構築します。

スタンドアロン構成以外の場合は、リポジトリサーバーに、Apache HTTP Server をインストールし起動します。スタンドアロン構成の場合は、Apache HTTP Server サーバーは不要です。

```
# yum -y install httpd
# systemctl start httpd
# systemctl enable httpd
```

次に、(1)で入手したファイルの格納ディレクトリを作成してください。

スタンドアロン構成の場合は、任意のディレクトリでかまいません。

スタンドアロン構成以外の場合は、HTTP サーバーが公開するディレクトリ(/var/www/html 等)配下にディレクトリを作成し、(1)にて入手したリリースパッケージと zip ファイル一式を配置してください。

```
# mkdir -p /path/to/repos
```

その後リリースパッケージをインストールしてください。

- デフォルトのアーキテクチャを VE1 にする場合

```
# cd /path/to/repos
# yum install ./TSUBASA-soft-release-ve1-3.0-1.noarch.rpm
```

- デフォルトのアーキテクチャを VE3 にする場合

```
# cd /path/to/repos
# yum install ./TSUBASA-soft-release-ve3-3.0-1.noarch.rpm
```

⚠ 注意

インストール中に以下のようなメッセージが表示された場合は、既存の設定を上書きしないために設定ファイルを異なる名前(拡張子が.rpmnew)でインストールしたことを示しており、新機能のための設定が追加されている場合があります。新機能を使わない場合は *.rpmnew について特に作業はありません。新機能を使いたい場合は *.rpmnew ファイルの内容を確認し、更新されている設定を既存の設定ファイルに転記し、環境に合わせて修正を行ってください。

```
warning: /etc/yum.repos.d/TSUBASA-repo.repo created as /etc/yum.repos.d/TSUBASA-re
po.repo.rpmnew
warning: /etc/yum.repos.d/TSUBASA-restricted.repo created as /etc/yum.repos.d/TSUB
ASA-restricted.repo.rpmnew
```

無償 yum リポジトリ設定ファイル(/etc/yum.repos.d/TSUBASA-repo.repo)と、アクセス制限付き yum リポジトリ設定ファイル(/etc/yum.repos.d/TSUBASA-restricted.repo)の yum リポジトリの設定欄を編集します。

各 yum リポジトリの設定で、ローカルに展開した yum リポジトリの URL(http アドレス)を、下記のように baseurl に設定し、enabled に 1 を設定してください。

(例) VEOS, MMM 等

変更前

```
baseurl=https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-repo_el8.10
```

変更後

```
baseurl=file:///path/to/repos/TSUBASA-repo_el8.10
```

(例) SDK-CE

変更前

```
baseurl=https://sxauroratsubasa.sakura.ne.jp/repos/community/sdk/sdk_el8
```

変更後(リポジトリサーバーがスタンドアロン構成)

```
baseurl=file:///path/to/repos/sdk_el8
```

変更後(リポジトリサーバーがスタンドアロン構成以外)

※ DocumentRoot にしたディレクトリ(/var/www/html)は指定する必要はありません。

```
baseurl=http://address.or.name.of.server/path/to/repos/sdk_el8
```

yum のキャッシュを削除します。

```
# yum clean all
```

最後に zip ファイルからリポジトリを展開してください。

※展開先に過去に展開した古いファイルが既にある場合は、それらを退避した上で zip ファイルを展開してください。

```
# cd /path/to/repos/
# find . -name '*.zip' | xargs -n1 unzip -o
# rm *.zip
```

(3) ローカル yum リポジトリの更新

リポジトリサーバー上で、(1)に記載された最新の zip ファイルを、yum リポジトリを作成したディレクトリ(以下の例では、/path/to/repos)にコピーした上で展開します。

```
# cd /path/to/repos/
# find . -name '*.zip' | xargs -n1 unzip -o
# rm *.zip
```

各 VH において、OS のバージョンアップを行った場合は、yum リポジトリの URL に含まれるバージョン番号を更新してください。

(例) VEOS, MMM 等

変更前

```
baseurl=file:///path/to/repos/TSUBASA-repo_el7.5
```

変更後

```
baseurl=file:///path/to/repos/TSUBASA-repo_el8.10
```

(例) SDK

変更前

```
baseurl=file:///path/to/repos/sdk_el7
```

変更後(リポジトリサーバーがスタンドアロン構成)

```
baseurl=file:///path/to/repos/sdk_el8
```

yum のキャッシュを削除します。

```
# yum clean all
```

3.2　インストールの実行

各環境に合わせてインストールを実行します。

シェル変数 TSUBASA_GROUPS に指定するグループは、InifiniBand HCA の搭載有無、有償ソフトウェアの購入有無に合わせて変更してください。指定可能なグループについては、3.2.4 を参照してください。

⚠ 注意

NEC yum リポジトリにアクセスできない場合は、時間をあけて再度試してください。NEC yum リポジトリがメンテナンス中の可能性もありますので、以下もあわせてご確認ください。

https://sxauroratsubasa.sakura.ne.jp/forums/

3.2.1　有償ソフトウェアを購入して PP サポート契約あり/オンラインの場合

- リポジトリ設定ファイルのインストール

  以下のコマンドを実行し、アクセス制限付きyumリポジトリへのアクセス設定をおこなってください。

  - デフォルトのアーキテクチャをVE1にする場合

```
# yum install https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve1-3.0-1.noarch.rpm
```

  - デフォルトのアーキテクチャをVE3にする場合

```
# yum install https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve3-3.0-1.noarch.rpm
```

⚠ 注意

インストール中に以下のようなメッセージが表示された場合は、既存の設定を上書きしないために設定ファイルを異なる名前(拡張子が.rpmnew)でインストールしたことを示しており、新機能のための設定が追加されている場合があります。新機能を使わない場合は *.rpmnew について特に作業はありません。新機能を使いたい場合は *.rpmnew ファイルの内容を確認し、更新されている設定を既存の設定ファイルに転記し、環境に合わせて修正を行ってください。

```
warning: /etc/yum.repos.d/TSUBASA-repo.repo created as /etc/yum.repos.d/TSUBASA-re
po.repo.rpmnew
warning: /etc/yum.repos.d/TSUBASA-restricted.repo created as /etc/yum.repos.d/TSUB
ASA-restricted.repo.rpmnew
```

- Auroraソフトウェアのインストール

  実行・コンパイル環境の例

```
# TSUBASA_GROUPS="ve-devel ve-infiniband nec-sdk-devel nec-mpi-devel nqsv-execution
scatefs-client-tsubasa"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

  実行環境の例

```
# TSUBASA_GROUPS="ve-runtime ve-infiniband nec-sdk-runtime nec-mpi-runtime nqsv-execution
scatefs-client-tsubasa"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

フロントエンドマシンの例

```
# TSUBASA_GROUPS="ve-frontend nec-sdk-frontend nec-mpi-frontend"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

3.2.2　有償ソフトウェアを購入して PP サポート契約あり/オフラインの場合

- ローカル yum リポジトリの構築

  オンラインでインターネット上の yum リポジトリへアクセスできない場合、無償 yum リポジトリとアクセス制限付き yum リポジトリを、ローカルにアクセス可能な yum リポジトリとして構築してください。構築方法は、「3.1.3　VH から NEC yum リポジトリへ直接アクセスできない場合」を参照してください。

- リポジトリ設定ファイルのインストール

  以下のコマンドを実行し、ローカルyumリポジトリへのアクセス設定をおこなってください。

  - デフォルトのアーキテクチャをVE1にする場合

```
# cd /path/to/repos
# yum install ./TSUBASA-soft-release-ve1-3.0-1.noarch.rpm
```

  - デフォルトのアーキテクチャをVE3にする場合

```
# cd /path/to/repos
# yum install ./TSUBASA-soft-release-ve3-3.0-1.noarch.rpm
```

⚠ 注意

インストール中に以下のようなメッセージが表示された場合は、既存の設定を上書きしないために設定ファイルを異なる名前(拡張子が.rpmnew)でインストールしたことを示しており、新機能のための設定が追加されている場合があります。新機能を使わない場合は *.rpmnew について特に作業はありません。新機能を使いたい場合は *.rpmnew ファイルの内容を確認し、更新されている設定を既存の設定ファイルに転記し、環境に合わせて修正を行ってください。

```
warning: /etc/yum.repos.d/TSUBASA-repo.repo created as /etc/yum.repos.d/TSUBASA-re
po.repo.rpmnew
warning: /etc/yum.repos.d/TSUBASA-restricted.repo created as /etc/yum.repos.d/TSUB
ASA-restricted.repo.rpmnew
```

- Auroraソフトウェアのインストール

  実行・コンパイル環境の例

```
# TSUBASA_GROUPS="ve-devel ve-infiniband nec-sdk-devel nec-mpi-devel nqsv-execution
scatefs-client-tsubasa"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
```

```
# yum group install $TSUBASA_GROUPS
```

実行環境の例

```
# TSUBASA_GROUPS="ve-runtime ve-infiniband nec-sdk-runtime nec-mpi-runtime nqsv-execution
scatefs-client-tsubasa"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

フロントエンドマシンの例

```
# TSUBASA_GROUPS="ve-frontend nec-sdk-frontend nec-mpi-frontend"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

3.2.3　有償ソフトウェアを未購入、または、PP サポート契約なしの場合

(1) リポジトリ設定ファイルのインストール

✓　オンラインの場合

オンラインでインターネット上のyumリポジトリを利用可能な場合、以下のコマンドを実行してください。

- デフォルトのアーキテクチャをVE1にする場合

```
# yum install https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve1-3.0-1.noarch.rpm
```

- デフォルトのアーキテクチャをVE3にする場合

```
# yum install https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve3-3.0-1.noarch.rpm
```

✓　オフラインの場合

無償yumリポジトリのみをローカルに構築し、以下のコマンドを実行し、ローカルyumリポジトリへのアクセス設定をおこなってください。

- デフォルトのアーキテクチャをVE1にする場合

```
# yum install https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve1-3.0-1.noarch.rpm
```

- デフォルトのアーキテクチャをVE3にする場合

```
# yum install https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve3-3.0-1.noarch.rpm
```

⚠ 注意

インストール中に以下のようなメッセージが表示された場合は、既存の設定を上書きしないために設定ファイルを異なる名前(拡張子が.rpmnew)でインストールしたことを示しており、新機能のための設定が追加されている場合があります。新機能を使わない場合は *.rpmnew について特に作業はありません。新機能を使いたい場合は *.rpmnew ファイルの内容を確認し、更新されている設定を既存の設定ファイルに転記し、環境に合わせて修正を行ってください。

```
warning: /etc/yum.repos.d/TSUBASA-repo.repo created as /etc/yum.repos.d/TSUBASA-re
po.repo.rpmnew
warning: /etc/yum.repos.d/TSUBASA-restricted.repo created as /etc/yum.repos.d/TSUB
ASA-restricted.repo.rpmnew
```

(2) 有償ソフトウェアの展開

「インターネット配信製品ダウンロードサービス」で最新の有償ソフトウェアの zip ファイルをダウンロードして、VH に転送してください。

古い有償ソフトウェアの zip ファイルが有る場合は削除後、zip ファイルを展開してください。

```
# find . -name '*.zip' | xargs -n1 unzip -o
```

(3) Auroraソフトウェアのインストール

ご利用の環境に合わせて、Auroraソフトウェアをインストールしてください。

- 実行・コンパイル環境の場合

```
# yum install devel/*.rpm
# TSUBASA_GROUPS="ve-devel ve-infiniband"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

- 実行環境のインストールの場合

```
# yum install runtime/*.rpm
# TSUBASA_GROUPS="ve-runtime ve-infiniband"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

- フロントエンドマシンのインストールの場合

```
# yum install frontend/*.rpm
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh ve-frontend
# yum group install ve-frontend
```

- NQSVを使用する場合

  NQSVを使用する場合は、以下の手順でインストールしてください。

```
# yum install NQSV-client/*.rpm
```

- ScaTeFS Clientを使用する場合

  ScaTeFS Clientを使用する場合は、以下の手順でインストールしてください。
  事前にve-infinibandパッケージグループまたはve-infiniband-mofedX.Xパッケージグループがインストールされている必要があります。

  ・ RHEL/Rocky Linux 8.10 にインストールする場合

```
# yum install scatefs-client/el8.10/scatefs-client-tsubasa/*.rpm
```

3.2.4　シェル変数 TSUBASA_GROUPS に指定可能なグループ

各マシンタイプに応じてパッケージグループを指定してください。シェル変数 TSUBASA_GROUPS に指定可能なグループとインストール・アップデート対象となるパッケージファイルの関係は以下の通りです。

表 5 パッケージグループ

パッケージグループ	機能	実行・コンパイル環境	実行環境	フロントエンドマシン
ve-devel	VE Application	○	-	-
ve-runtime		-	○	-
ve-frontend		-	-	○
ve-infiniband	InfiniBand for SX-Aurora TSUBASA	○	○	-
nec-sdk-devel	SDK (MPI 以外)	○	-	-
nec-sdk-frontend		-	-	○
nec-sdk-runtime	SDK ランタイム（MPI 以外）	-	○	-
nec-mpi-devel	SDK (MPI)	○	-	-
nec-mpi-frontend		-	-	○
nec-mpi-runtime	SDK ランタイム（MPI）	-	○	-
nqsv-execution	NQSV	○	○	-
scatefs-client-tsubasa	ScaTeFS Client	○	○	-

※ 各パッケージグループに含まれるパッケージについては、付録 B を参照してください。

表 6 パッケージグループ/NQSV

パッケージファイル	グループ名	
	nqsv-batch	nqsv-execution
NQSV-API	○	-
NQSV-Client	○	○
NQSV-JobManipulator	○	-
NQSV-JobServer	-	○
NQSV-ResourceManager	○	-

表 7 パッケージグループ/ScaTeFS Client

パッケージファイル	グループ名		
	scatefs-client-tsubasa	scatefs-client-tsubasa-mofedX.X (※1)	scatefs-client-tsubasa-container
kmod-scatefs-client-modules-mofedX.X	○	○	-
scatefs-client-libscatefsib_ve	○	○	○
scatefs-client-mount-utils	○	○	-
scatefs-client-rcli-utils	○	○	-
scatefs-client-sys-utils	○	○	-
scatefs-client-usr-utils	○	○	○

(※1) このパッケージグループは、特定のバージョンの MLNX_OFED に対応した ScaTeFS Client をインストールする場合に使用します。特別な理由がない限りは、scatefs-client-tsubasa-mofedX.X グループではなく scatefs-client-tsubasa グループを使用してください。

表 8 パッケージグループ/ScaTeFS Server

パッケージファイル	グループ名	
	scatefs-server	scatefs-server-monitoring (※1)
scatefs-srv	○	-
scatefs-mon	-	○

(※1) "scatefs-server-monitoring" グループは RHEL 7.7 以降で対応しています。

3.3　VEドライバとLinuxカーネルの互換性確認

インストールした VE ドライバがホストマシンの Linux カーネルと互換性がない場合、VE ドライバのパッケージ（vp-kmod、ve_drv-kmod パッケージ）のダウングレードが必要な場合があります。

ホストマシンのカーネルは、原則として動作検証済みのカーネルの中で最新のものにアップデートすること(2.2 動作検証済みカーネルへのアップデート)を推奨します。最新バージョンをご利用されない場合、Aurora ソフトウェアをインストールしても、ve_drv-kmod パッケージに含まれる ve_drv モジュールがロードされずに VE が ONLINE にならない場合があります。その場合、vp-kmod、ve_drv-kmod パッケージをダウングレードする必要があります。

3.3.1　ダウングレードの要否確認

ダウングレードの要否を確認します。まずは、ve_drv モジュールがロードされているかを確認します。

```
# lsmod | grep ve_drv
ve_drv              262144  128
vp                   20480  1 ve_drv
```

ve_drv が表示された場合、ダウングレードは不要です。3.4 に進んでください。

```
# lsmod | grep ve_drv
（何も表示されない）
```

何も表示されなかった場合、ロードできていません。次のコマンドで VH の Linux カーネルバージョンを確認してください。

```
# uname -r
4.18.0-553.8.1.el8_10.x86_64
```

次に、vp-kmod および ve_drv-kmod パッケージのバージョンを確認して、対応カーネルモジュールがインストールされていることを確認してください。

【SX-Aurora TSUBASA】検証済み Linux カーネル
https://www.support.nec.co.jp/View.aspx?id=3140106285

```
# rpm -q vp-kmod
vp-kmod-3.0.1-4.18.0_372.32.1_1.el8.x86_64
# rpm -q ve_drv-kmod
ve_drv-kmod-3.0.1-4.18.0_372.32.1.el8_6.x86_64_1.el8.x86_64
```

Linux カーネルバージョンより vp-kmod または ve_drv-kmod パッケージのバージョンが新しい場合、vp-kmod、ve_drv-kmod パッケージのダウングレードが必要です。

3.3.2　ダウングレードの実行

次のコマンドを実行して、vp-kmod、ve_drv-kmod パッケージのダウングレードを実行してください。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
# yum downgrade vp-kmod ve_drv-kmod
# /opt/nec/ve/sbin/start-all-ve-services
```

3.4　VE番号定義ファイルの確認・設定

1.2.1 ハードウェアの表 3 サーバーモデルに記載されたモデルをご利用のお客様は、以下のコマンドで、VE 番号定義ファイルの設定・確認が必要です。

以下のコマンドで、VE 番号定義ファイルが存在するか確認します。VE 番号定義ファイルが存在する場合は 3.5 に進んでください。

```
# ls -l /etc/opt/nec/ve/veos/ve_nodes.json
-rw-r--r--. 1 root root 274  2月 22  2023 /etc/opt/nec/ve/veos/ve_nodes.json
```

VE 番号定義ファイルが存在しない場合、以下のコマンドで VE 番号定義ファイルを配置します。

```
# /opt/nec/ve/mmm/analysis/sbin/mmm-config-json.sh 2
```

VE 番号定義ファイルが配置されたことを確認し、VE サービスを再起動します。

```
# ls -l /etc/opt/nec/ve/veos/ve_nodes.json
-rw-r--r--. 1 root root 274  2月 22  2023 /etc/opt/nec/ve/veos/ve_nodes.json
# /opt/nec/ve/sbin/terminate-all-ve-services
# /opt/nec/ve/sbin/start-all-ve-services
```

3.5　VEの状態確認

vecmd コマンドで VE カードが ONLINE であることを確認します。VE が ONLINE になるまでには数分程度かかることがあります。

```
# /opt/nec/ve/bin/vecmd state get
Vector Engine MMM-Command v1.0.0
Command:
state -N 0 get
--------------------------------------------------------------------------------
VE0 [03:00.0] [ ONLINE ] Last Modif:2017/11/29 10:18:00
--------------------------------------------------------------------------------
Result: Success
```

⚠ 注意

"UNINITIALIZED"や"<OFFLINE>"が表示される場合は、しばらくお待ちください。
しばらくお待ちいただいてもVE stateが"ONLINE"へ切り替わらない場合、以下のコマンドを実行し、サービスを再起動してください。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
# /opt/nec/ve/sbin/start-all-ve-services
```

また、"There is no executable ve card!"と表示される場合についても、同様の方法でサービスを再起動してください。

"UNAVAILABLE"が表示された場合は、rebootコマンドにて、VHを再起動してください。

```
# reboot
```

再起動しても、なお"UNAVAILABLE"が表示される場合、故障の可能性があります。以下の手順書を参考に切り分け作業を行って下さい。

- NEC Aurora Forum Documentation
  "Vector Engine 2.0 障害切り分け手順書"

3.6　VMC Firmware (VMCFW) の アップデート

次のコマンドを実行して、VMCFW の更新要否を判定します。

```
# /opt/nec/ve/bin/vecmd fwup check
```

「Updating VMCFW is required.」と出力された場合は VMCFW の更新が必要です。「Updating VMCFW is not required.」と出力された場合は VMCFW の更新が不要です。

VMCFW の更新が必要な場合は、次の手順でファームウェアをアップデートします。

(1)　vecmdコマンドにて、VEの状態を変更し、ファームウェアをアップデートします。

- VE1/VE2に対してアップデートする場合

```
# /opt/nec/ve/bin/vecmd state set off
# /opt/nec/ve/bin/vecmd state set mnt
# /opt/nec/ve/bin/vecmd fwup vmc aurora_MK10.bin
```

※ファームウェアの更新には、数分程度かかります。

VHを再起動します。

```
#reboot
```

- VE3に対してアップデートする場合

```
# /opt/nec/ve/bin/vecmd state set off
# /opt/nec/ve/bin/vecmd fwup
```

※ファームウェアの更新には、10分程度かかります。

VHをshutdownして再度電源ONします(VE1と違うので注意してください)

```
# shutdown -h now
```

(2)　VH起動後、rootユーザーでログインします。

(3)　VEの状態確認

vecmdコマンドでVEカードがONLINEであることを確認します。VEがONLINEになるまでには数分程度かかることがあります。

```
# /opt/nec/ve/bin/vecmd state get
Vector Engine MMM-Command v1.0.0
Command:
state -N 0 get
--------------------------------------------------------------------------------
VE0 [03:00.0] [ ONLINE ] Last Modif:2017/11/29 10:18:00
--------------------------------------------------------------------------------
Result: Success
```

⚠ 注意

"UNINITIALIZED"や"<OFFLINE>"が表示される場合は、しばらくお待ちください。また、"ONLINE"にならない、もしくは表示されないVEがある場合、一度VEをshutdownしてから電源を入れ直してください。

3.7　コマンド互換パッケージのインストール(選択)

2023 年 3 月リリースの SX-Aurora TSUBASA ソフトウェアより、Linux から移植したコマンドは ve-というプレフィックスが付きます。2023 年 3 月リリース以前の Linux から移植したコマンドと互換性を保つ必要がある場合は、コマンド互換パッケージをインストールしてください。

環境により、コマンド互換パッケージが異なります。各環境に合わせて適当なパッケージをインストールしてください。

- 実行・コンパイル環境

```
# yum install ve-command-name-compat-devel
```

- 実行環境

```
# yum install ve-command-name-compat-runtime
```

- フロントエンド環境

```
# yum install ve-command-name-compat-front-end
```

3.8　HugePages の設定

VEOS、NEC MPI、およびScaTeFS Clientは、HugePagesを使用します。次のコマンドを実行して、HugePagesの設定をします。VMCFWの更新のため、VHを再起動済みの場合は、自動設定されていますので実行する必要はありません。

(1)　HugePages の設定

```
# /opt/nec/ve/sbin/ve-set-hugepages
```

(2) HugePages の確認
nr_hugepages, nr_overcommit_hugepages の値が 0 ではないことを確認します。

```
# /opt/nec/ve/sbin/ve-set-hugepages -s
nr_hugepages:256
nr_overcommit_hugepages:23790
```

- 起動中の自動設定の確認

Linux の systemd-sysctl サービスもVH起動時にHugePages設定ができますが、本自動設定コマンド(ve-set-hugepages)による設定とは排他となります。/etc/sysctl.confや、/etc/sysctl.d 配下等でHugePages、HugePages オーバーコミットの設定を行っている場合、削除してください。詳細は「4.12 HugePagesの設定」を参照してください。

3.9　ScaTeFS の起動(選択)

ScaTeFS Client をインストールした場合は、VH を再起動して、ScaTeFS Client を起動します。VMCFW の更新のため、VH を再起動済みの場合は、改めて再起動する必要はありません。

```
# reboot
```

以上でソフトウェアのインストールは終わりです。
次に、第 4 章を参照して、ソフトウェアの設定を行ってください。

第4章　SX-Aurora TSUBASA ソフトウェアの設定

SX-Aurora TSUBASA システムの運用を開始するにあたって、ソフトウェアごとに次の設定をしてください。

以下の表に各設定の概要を示します。

表 9 SX-Aurora TSUBASA ソフトウェアの設定

項目	要件
4.1 運用ネットワークの設定	Aurora システムを運用ネットワークに接続する場合
4.2 ライセンスサーバーの指定	SDK, NEC MPI (*), ScaTeFS, NQSV を使用する場合 (*) Version 2.13.0 以降は不要
4.3 ライセンスサーバーの設定	SDK, NEC MPI (*), ScaTeFS, NQSV を使用する場合 (*) Version 2.13.0 以降は不要
4.4 InfiniBand HCA の Relaxed Ordering 設定(選択)	モデル A412-8、B401-8、B403-8 または C401-8 を使用する場合
4.5 InfiniBand HCA の PCIe クレジット数の設定(選択)	モデル B302-8、B401-8 または C401-8 を使用する場合
4.6 ScaTeFS の設定	ScaTeFS を使用する場合
4.7 NQSV の設定	NQSV を使用する場合
4.8 NEC MPI の設定	NEC MPI を使用する場合
4.8.1 SELinux の設定	NEC MPI を使用し、SELinux が有効の場合
4.8.2 ファイアウォールの設定	NEC MPI を使用し、ファイアウォールが有効な場合
4.8.3 InfiniBand QoS 向けの設定	NEC MPI を使用し、InfiniBand サブネットマネージャにおいて Quality of Service(QoS)を有効にしている場合
4.8.4 NVIDIA SHARP の設定	NEC MPI を使用し、InfiniBand の NVIDIA SHARP 機能を使用する場合
4.8.5 HCA デバイス名の固定化	NEC MPI を使用する場合
4.8.6 各 VE が使用する HCA と通信方式の設定（選択）	NEC MPI を使用し、モデル A412-8、B401-8、B403-8 または C401-8 を使用する場合

項目	要件
4.8.7 ホスト名の名前解決の設定	NEC MPI を使用する場合
4.8.8 InfiniBand Adaptive Routing 利用時の設定（選択）	NEC MPI を使用し、InfiniBand ネットワークの設定において、Adaptive Routing 機能を有効にする場合
4.8.9 Partial Process Swapping 利用時の設定（選択）	VE 上で動作する MPI プログラムを NQSV から実行する際の Partial Process Swapping を利用する場合
4.9 NEC Parallel Debugger の設定	NEC Parallel Debugger を使用する場合
4.10 仮想メモリの設定の確認	ve_exec コマンドを使用する場合
4.11 memlock リソースの設定の確認	プログラム実行する場合
4.12 HugePages の設定	VEOS（高速 I/O ライブラリ）、NEC MPI、または ScaTeFS Client で HugePages を使用する場合
4.13 Partial Process Swapping機能の設定	停止状態のVEプロセスのメモリの一時解放と復元機能を使用する場合
4.14 Process accounting の設定	VE プロセスのプロセスアカウント情報採取機能を使用する場合
4.15 SX-Aurora TSUBASA でのプログラム実行について	プログラム実行する場合

4.1　運用ネットワークの設定

4.1.1　InfiniBand (IP over IB)を使用する場合

運用ネットワークに InfiniBand (IP over IB)を使用する場合は、ib0 インタフェースに対してネットワークの設定を行ってください。

ネットワークの構成に関しては、「付録 C ネットワーク構成」を参照してください。

```
# nmcli connection add type infiniband autoconnect yes con-name ib0 ifname ib0
# nmcli connection mod ib0 ipv4.method manual ipv4.address "XX.XX.XX.XX/YY" (*注)
```

(*注: XX.XX.XX.XX はIP over IBのIPアドレス、YYはネットマスクのビット数)

InfiniBand カード(HCA)を 2 枚以上搭載している場合は、2 枚の HCA を使用して、active-backup の bond0 インタフェースを作成して、bond0 インタフェースに対してネットワークの設定を行ってください。

```
# nmcli connection add type bond con-name bond0 ifname bond0 mode active-backup
# nmcli connection mod bond0 ipv4.method disabled ipv6.method ignore
# nmcli connection add type infiniband autoconnect yes ifname ib0 master bond0
# nmcli connection add type infiniband autoconnect yes ifname ib1 master bond0
# nmcli connection mod bond0 +bond.options primary=ib0
# nmcli connection mod bond0 +bond.options miimon=100,updelay=100,downdelay=100
# nmcli connection mod bond0 ipv4.method manual ipv4.address "XX.XX.XX.XX/YY" (*注)
```

(*注: XX.XX.XX.XX はIP over IBのIPアドレス、YYはネットマスクのビット数)

4.1.2　Ethernet を使用する場合

運用ネットワークに Ethernet を使用する場合は、運用ネットワークを接続する Ethernet インタフェースに対してネットワークの設定を行ってください。Ethernet インタフェース名は、機種毎に異なりますので、ifconfig または ip コマンドでネットワークインタフェース名を確認してから設定を行います。

```
# ifconfig -a
enp129s0f0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
       inet xx.xx.xx.xx  netmask yy.yy.yy.yy  broadcast zz.zz.zz.zz
       inet6 fe80::ec4:7aff:feea:d338  prefixlen 64  scopeid 0x20<link>
       ether 0c:c4:7a:ea:d3:38  txqueuelen 1000  (Ethernet)
       RX packets 215948  bytes 27415658 (26.1 MiB)
       RX errors 0  dropped 0  overruns 0  frame 0
       TX packets 943  bytes 475319 (464.1 KiB)
       TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp129s0f1: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
       ether 0c:c4:7a:ea:d3:39  txqueuelen 1000  (Ethernet)
       RX packets 0  bytes 0 (0.0 B)
       RX errors 0  dropped 0  overruns 0  frame 0
       TX packets 0  bytes 0 (0.0 B)
       TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

# nmcli connection add type ethernet autoconnect yes con-name enp129s0f1 ifname enp129s0f1
# nmcli connection mod enp129s0f1 ipv4.method manual ipv4.address "XX.XX.XX.XX/YY"(*注)
```

(*注: enp129s0f1は運用ネットワークのインタフェース名の例、XX.XX.XX.XX はIPアドレス、YYはネットマスクのビット数)

4.1.3　ネットワークのサービスのリスタート

4.1.1 または 4.1.2 の設定後に、ネットワークのサービスをリスタートしてください。

RHEL/Rocky Linux 8.x の場合:

```
# systemctl restart NetworkManager.service
# nmcli con up インタフェース名(*注)
```

(*注: インタフェース名は、設定した運用ネットワークのインタフェース名を指定してください)

4.2　ライセンスサーバーの指定

有償ソフトウェアをインストールした VH 上では、有償ソフトウェアがライセンスの払い出しを受けるために接続するライセンスサーバーの情報を設定しておく必要があります。

ライセンスサーバーの情報を設定するには、次の 2 つの方法があります。

- 設定ファイルによる指定

  設定ファイル/opt/nec/aur_license/aur_license.conf 中に、接続先のライセンスサーバーのホスト名とポート番号を以下のように設定します。

表 10 ライセンスサーバーの指定

設定項目	ヘッダー文字列	値の指定
ライセンスサーバーのホスト名	License_server_host	ホスト名の文字列
待ち受けポート番号	License_server_port	ポート番号（数字）

aur_license.confファイルの記述例

```
License_server_host=sv_host
License_server_port=7300
```

ScaTeFS Clientの場合、この方法で設定してください。

- 環境変数による指定

  VH上で起動するソフトウェア（NEC MPI、C/C++ compiler、Fortran compiler）に対して個別に接続先のライセンスサーバーを指定する必要がある場合は、環境変数により指定をしてください。使用する環境変数は次のとおりです。

表 11 ライセンスサーバーの指定（環境変数）

設定項目	環境変数
ライセンスサーバーのホスト名	AURLIC_SERVER_HOSTNAME
待ち受けポート番号	AURLIC_SERVER_PORT

bashの場合の指定例

```
$ export AURLIC_SERVER_HOSTNAME="sv_host"
$ export AURLIC_SERVER_PORT="7300"
```

環境変数による指定は、設定ファイルによる指定より優先されます。

4.3　ライセンスサーバーの設定

スタンドアロン構成の場合は、VH 上でライセンスサーバーの設定を行います。スタンドアロン構成でない場合は、別途、管理サーバーにライセンスサーバーをインストールし、設定を行います。ライセンス管理対象の有償ソフトウェアに添付されている「HPC ソフトウェアライセンス管理説明書」を参照のうえ実施してください。

4.4　InfiniBand HCAのRelaxed Ordering設定(選択)

モデル A412-8、B401-8、B403-8 または C401-8 をご利用の場合は、性能向上のために、InfiniBand HCA の PCIe の Relaxed Ordering 機能を有効に設定してください。

⚠ 注意

- Relaxed Ordering機能を有効にする場合は、SX-Aurora TSUBASAで動作させるミドルウェア、およびアプリケーション(InfiniBand Verbs APIを使用するソフトウェア)が、Relaxed Orderingに対応していることを確認してください。対応の可否は各提供ベンダへご確認ください。
- NEC MPI(2.6.0以降)および、ScaTeFSはRelaxed Ordering機能に対応しています。
- **モデルA412-8、B401-8、B403-8、C401-8以外は、Relaxed Orderingは有効にしないでください。結果不正となる可能性があります。**

以下の手順で、実装している HCA の枚数分、Relaxed Ordering の機能を有効にしてください。

注）　手順の中の"/dev/mst/mtXXXX_pciconf0"、 "/dev/mtXXXX_pciconf1" で示している XXXX は、モデルにより異なります。下記の値を指定してください。

モデル A412-8、B401-8 … 4123 (例: /dev/mst/mt4123_pciconf0)

モデル B403-8、C401-8 … 4129 (例: /dev/mst/mt4129_pciconf0)

```
# mst start
# mlxconfig -d /dev/mst/mtXXXX_pciconf0 --yes set PCI_WR_ORDERING=1
…
```

```
Configurations:                          Next Boot      New
        PCI_WR_ORDERING                  per_mkey(0)     force_relax(1)


 Apply new Configuration? (y/n) [n] : y
Applying... Done!
-I- Please reboot machine to load new configurations.


# mlxconfig -d /dev/mst/mtXXXX_pciconf1 --yes set PCI_WR_ORDERING=1
…
Configurations:                          Next Boot      New
        PCI_WR_ORDERING                  per_mkey(0)     force_relax(1)


 Apply new Configuration? (y/n) [n] : y
Applying... Done!
-I- Please reboot machine to load new configurations
```

Relaxed Ordering の設定を反映するために、再起動を行ってください。

```
# reboot
```

なお、mlxconfig のコマンドオプションに"PCI_WR_ORDERING=0"を指定すると、Relaxed Ordering の機能を無効にできます。設定を変更した場合は、再起動を行ってください。

再起動後、以下のように、Relaxed Ordering 機能が有効になっていることを確認してください。

```
# mst start
# mlxconfig -d /dev/mst/mtXXXX_pciconf0 query | grep PCI_WR_ORDER
        PCI_WR_ORDERING                  force_relax(1)
# mlxconfig -d /dev/mst/mtXXXX_pciconf1 query | grep PCI_WR_ORDER
        PCI_WR_ORDERING                  force_relax(1)
```

PCI_WR_ORDERING が"force_relax(1)"になっている場合は Relaxed Ordering 有効

PCI_WR_ORDERING が"per_mkey(0)"になっている場合は Relaxed Ordering 無効

を示します。

4.5　InfiniBand HCAのPCIeクレジット数の設定(選択)

モデル B302-8、B401-8、C401-8 をご利用の場合は、性能向上のために、InfiniBand HCA の PCIe クレジット数の設定を行ってください。モデルにより指定する値が異なります。下記の値を指定してください。

モデル B302-8 … 32

モデル B401-8 … 32

モデル C401-8 … 36

本設定で PCIe クレジット数が 16 から指定した値に増加します。

⚠ 注意

- HCAのPCIeクレジット数の設定は、HCA Firmwareバージョンが20.28.2006以降のHCAをご利用の場合に設定できます。
- PCIeクレジット数は上記で記載した以外の値は設定しないようにしてください。

以下の手順で、実装している HCA の枚数分、設定を行ってください。

注）　手順の中の"/dev/mst/mtXXXX_pciconf0"、"/dev/mtXXXX_pciconf1"で示している XXXX は、モデルにより異なります。下記の値を指定してください。

モデル B302-8、B401-8 … 4123 (例: /dev/mst/mt4123_pciconf0)

モデル C401-8 … 4129 (例: /dev/mst/mt4129_pciconf0)

また、YY は PCIe クレジット数を設定します。

```
# mst start
# mlxconfig -d /dev/mst/mtXXXX_pciconf0 --yes set ADVANCED_PCI_SETTINGS=1
…
Configurations:                          Next Boot      New
        ADVANCED_PCI_SETTINGS             False(0)       True(1)

 Apply new Configuration? (y/n) [n] : y
Applying... Done!
-I- Please reboot machine to load new configurations.

# mlxconfig -d /dev/mst/mtXXXX_pciconf0 --yes set MAX_ACC_OUT_READ=YY
…
Configurations:                          Next Boot      New
        MAX_ACC_OUT_READ                  0             YY

 Apply new Configuration? (y/n) [n] : y
Applying... Done!
-I- Please reboot machine to load new configurations.

# mlxconfig -d /dev/mst/mtXXXX_pciconf1 --yes set ADVANCED_PCI_SETTINGS=1
…
Configurations:                          Next Boot      New
        ADVANCED_PCI_SETTINGS              False(0)      True(1)

 Apply new Configuration? (y/n) [n] : y
```

```
Applying... Done!
-I- Please reboot machine to load new configurations.


# mlxconfig -d /dev/mst/mtXXXX_pciconf1 --yes set MAX_ACC_OUT_READ=YY
…
Configurations:                          Next Boot      New
        MAX_ACC_OUT_READ                  0            YY


 Apply new Configuration? (y/n) [n] : y
Applying... Done!
-I- Please reboot machine to load new configurations.
```

設定を反映するために、再起動を行ってください。

```
# reboot
```

再起動後、以下のように、PCIe クレジット数の変更が反映されていることを確認してください。

```
 # mst start
 # mlxconfig -d /dev/mst/mtXXXX_pciconf0 query | grep ADVANCED_PCI_SETTINGS
        ADVANCED_PCI_SETTINGS             True(1)
 # mlxconfig -d /dev/mst/mtXXXX_pciconf0 query | grep MAX_ACC_OUT_READ
        MAX_ACC_OUT_READ                  YY
 # mlxconfig -d /dev/mst/mtXXXX_pciconf1 query | grep ADVANCED_PCI_SETTINGS
        ADVANCED_PCI_SETTINGS             True(1)
 # mlxconfig -d /dev/mst/mtXXXX_pciconf1 query | grep MAX_ACC_OUT_READ
        MAX_ACC_OUT_READ                  YY
```

ADVANCED_PCI_SETTINGS が"True(1)"、MAX_ACC_OUT_READ が指定した PCIe クレジット数となっていることを確認してください。

4.6　ScaTeFSの設定

ScaTeFS Client を使用する場合は、「NEC Scalable Technology File System (ScaTeFS) 運用の手引」の「第 6 章 Linux クライアントの設定」を参照して、設定してください。

SX-Aurora TSUBASA で利用するには、HugePages の設定も必要です。HugePages の設定については本書の「4.12 HugePages の設定」を参照してください。

ScaTeFS Server については、「NEC Scalable Technology File System (ScaTeFS) 運用の手引」を参照して構築してください。

4.7　NQSVの設定

NQSV を使用する場合は、「NEC Network Queuing System V (NQSV) 利用の手引 [管理編]」を参照して、ジョブサーバーとクライアントの設定をしてください。

また、「NEC Network Queuing System V (NQSV) 利用の手引 [JobManipulator 編]」の「5.4 HCA 割り当て機能」に従って、各ジョブサーバーホスト（VH）上にデバイスリソース定義ファイル（/etc/opt/nec/nqsv/resource.def）を作成して配置してください。

NQSV のサーバー機能（バッチサーバー、スケジューラー、アカウンティング）については、「NEC Network Queuing System V (NQSV) 利用の手引 [導入編]」を参照して、環境を構築してください。

4.8　NEC MPIの設定

本節では，MPI プログラムを起動する全 VH で必要となる設定を説明します。

あわせて、HugePages の設定も必要です。HugePages の設定については本書の「4.12 HugePages の設定」を参照してください。

4.8.1　SELinux の設定

SELinux が有効の場合，次の設定を行ってください。SELinux が無効の場合は必要ありません。

複数 VH 間でインタラクティブ MPI 実行を行い，かつ ホームディレクトリを NFS 又は ScaTeFS でマウントしている場合，SELinux の use_nfs_home_dirs を有効にしてください。

```
# setsebool -P use_nfs_home_dirs 1
```

以下のコマンドで"use_nfs_home_dirs --> on"が出力されれば、SELinux の設定は完了です。

```
# getsebool use_nfs_home_dirs
use_nfs_home_dirs --> on
```

4.8.2　ファイアウォールの設定

ファイアウォールが有効の場合，次の設定を行ってください。ファイアウォールが無効の場合は必要ありません。

NEC MPI は既定で 25257 番ポートから 25266 番ポートを使用して外部からの接続を待ち受けます。以下のコマンドを実行して，それらのポートを外部に開放してください。

```
# firewall-cmd --zone=public --permanent --add-port=25257-25266/tcp
# firewall-cmd --reload
```

以下のコマンドで開放したポートが出力されれば，ファイアウォールの設定は完了です。

```
# firewall-cmd --list-port --zone=public
25257-25266/tcp
```

使用するポートは，環境変数 NMPI_PORT_RANGE を使用して変更できます。以下に、25257 番ポートから 25266 番ポートを使用するように設定する例を示します。

```
# export NMPI_PORT_RANGE=25257:25266
```

NQSV で、Partial Process Swapping 機能(4.13 節)を利用した緊急ジョブを実行する場合は、追加のポートが必要になる場合があります。緊急ジョブのために停止するジョブも含めて、1VH 上のジョブの個数が 11 以上となる場合、ジョブの個数と同数以上のポートを firewall-cmd コマンドで開放し、環境変数 NMPI_PORT_RANGE で使用するポートを設定してください。

4.8.3　InfiniBand QoS 向けの設定

InfiniBand サブネットマネージャにおいて Quality of Service(QoS)を有効にしている場合、NECMPI が使用するサービスレベルの設定を行います。QoS が無効の場合、本設定は不要です。

NECMPI が使用するサービスレベルは、設定ファイル /etc/opt/nec/ve/mpi/necmpi.conf 中で設定します。本ファイル中で、キーワード "ib_qos_venode" の後に空白を開けてサービスレベルを設定します。設定がない場合、サービスレベル 0 が適用されます。先頭が # で始まる行は、コメントとみなされ無視されます。

```
# 例）サービスレベル 2 を設定する場合
ib_qos_venode      2
```

4.8.4　NVIDIA SHARP の設定

NEC MPI は、NVIDIA SHARP 機能に対応しています。SHARP を使用するためには、Aggregation Manager の設定を行う必要があります。Aggregation Manager は、InfiniBand の Subnet Manager と同一ホストで起動します。

Aggregation Manager、SHARP ライブラリは同一のバージョンにそろえる必要があります。SX-Aurora TSUBASA では以下のバージョンのみをサポートしています。

MLNX_OFED バージョン	SHARP バージョン
23.10-3.2.2.0	3.5.2

⚠ 注意

Subnet ManagerとしてNVIDIA UFMを使用する場合、NVIDIA UFMが提供するSHARPの

バージョンと、MLNX_OFEDが提供するSHARPのバージョンが一致しない場合があります。
バージョン間の互換性についてはNECのサポートにお問い合わせください。

VH の設定

VH のファイアウォール設定が必要です。Subnet Manager からの通信を許可するようにファイアウォールの設定を行ってください。以下は設定方法の例です。

```
# firewall-cmd --add-source=<Subnet ManagerのIPoIBアドレス> --zone=trusted --permanent
# firewall-cmd --add-source=<Subnet ManagerのIPoIBアドレス> --zone=trusted
```

Aggregation Manager の設定

Aggregation Manager の設定については、以下の NVIDIA 公式ホームページに記載されているドキュメントを参照してください。

https://docs.nvidia.com/networking/category/mlnxsharp

Aggregation Manager は、SHARP デーモンと TCP 通信を行います。Aggregation Manager が動作するホストでファイアウォールが有効になっている場合、Aggregation Manager が動作するホストで以下を実行し、TCP のポート 6126 を解放してください。

```
# firewall-cmd --add-port=6126/tcp --permanent
# firewall-cmd --add-port=6126/tcp
```

Aggregation Manager が動作するホストが複数のネットワークインタフェースを備えており、VH と直接通信できないインタフェースが存在する場合、Aggregation Manager が使用するインタフェースを明示的に指定する必要があります。この場合、Aggregation Manager が動作するホストの設定ファイル /opt/mellanox/sharp/conf/sharp_am.cfg 中で、パラメータ smx_sock_interface にインタフェース名を指定してください。以下の例は、ネットワークインタフェースとして ib0 を使用する場合の記述です。

```
smx_sock_interface ib0
```

4.8.5　HCA デバイス名の固定化

デフォルトの HCA デバイス名(mlx5_X)は、MLNX_OFED のアップデートや VH の BIOS のアップデートなどを行った場合に変更される可能性があります。InfiniBand を使用するシステムでは、以下の内容で /etc/udev/rules.d/60-rdma-persistent-naming.rules を作成し、デバイス名の固定化を行ってください。

```
ACTION=="add", SUBSYSTEM=="infiniband", PROGRAM="rdma_rename %k NAME_PCI"
```

設定後、再起動を行ってください。

```
# reboot
```

⚠ 注意

この手順を実施することにより、HCAデバイス名はibpXXs0に変更されます(XXはHCAのPCIバス番号です)。SX-Aurora TSUBASAソフトウェア以外の、HCAデバイス名を直接指定するソフトウェアを使用している場合、設定変更が必要になる場合があります。

4.8.6　各 VE が使用する HCA と通信方式の設定（選択）

モデル A412-8、B401-8、B403-8 または C401-8 をご利用の場合は、設定ファイル /etc/opt/nec/ve/mpi/necmpi.conf に以下の設定を追加してください。アプリケーション性能が向上する可能性があります。そのほかのモデルでは不要です。

- モデル A412-8 または B401-8

```
ib_adapter_name_list_pve 0=ibp69s0:1 1=ibp69s0:1 2=ibp69s0:1 3=ibp69s0:1 4=ibp1s0:1 5=ibp1s0:1
6=ibp1s0:1 7=ibp1s0:1
```

- モデル B403-8

```
ib_adapter_name_list_pve 0=ibp78s0:1 1=ibp78s0:1 2=ibp78s0:1 3=ibp78s0:1 4=ibp1s0:1 5=ibp1s0:1
6=ibp1s0:1 7=ibp1s0:1
```

- モデル C401-8

```
ib_adapter_name_list_pve 0=ibp78s0:1 1=ibp78s0:1 2=ibp78s0:1 3=ibp78s0:1 4=ibp1s0:1 5=ibp1s0:1
6=ibp1s0:1 7=ibp1s0:1
dma_vh_memcpy     off
ib_vh_memcpy_send  off
ib_vh_memcpy_recv  off
```

「ib_adapter_name_list_pve」から始まる行は、各 VE が MPI 通信時に使用する HCA を指定する設定です。キーワード "ib_adapter_name_list_pve" の後に空白を開けて "<VE 番号 0>=<HCA1>[,<HCA2> …] [<VE 番号 1>=<HCA1>[,<HCA2> …] …]"の書式で各 VE が使用する HCA を一行で記述します。「dma_vh_memcpy」「ib_vh_memcpy_send」「ib_vh_memcpy_recv」から始まる各行は MPI 通信時の VH メモリコピー機能の使用を指定する設定です。モデル C401-8 の場合、これらすべてに off を指定します。先頭が # で始まる行は、コメントとみなされ無視されます。

4.8.7　ホスト名の名前解決の設定

MPI 実行に使用するホストは、当該ホスト上でホスト名を IP アドレスに解決できる必要があります。シングルノード実行においても同様です。/etc/hosts や DNS などを使用し、ホスト名が IP アドレスに解決できるように当該ホスト上で設定を行ってください。

4.8.8　InfiniBand Adaptive Routing 利用時の設定（選択）

InfiniBand ネットワークの設定において、Adaptive Routing 機能を有効にする場合、NEC MPI の設定ファイル /etc/opt/nec/ve/mpi/necmpi.conf に以下の設定を追加してください。

```
# Adaptive Routing 有効時の指定
ib_ar 1
```

キーワード "ib_ar"に続けて空白および"1"を記載することにより、NEC MPI は Adaptive Routing 向けの動作となります。先頭が＃で始まる行はコメントとみなされ、無視されます。

本設定が無い場合、Adaptive Routing 機能は不使用と仮定した動作が既定値となります。なお、InfiniBand ネットワークの設定と本設定が一致しない場合、MPI 通信性能の低下などの原因となることがあります。

4.8.9　Partial Process Swapping 利用時の設定（選択）

NQSV を利用した実行において、Partial Process Swapping 機能を使用し、通常リクエストをサスペンドする際、VE 上で動作する MPI プロセスの Non Swappable Memory 領域を解放する場合、NEC MPI の設定ファイル /etc/opt/nec/ve/mpi/necmpi.conf に以下の設定を追加してください。

```
# Non Swappable Memory 領域を解放する場合の指定
swap_on_hold 1
```

キーワード "swap_on_hold"に続けて空白および"1"を記載してください。また、4.8.6 で「ib_vh_memcpy_send」と「ib_vh_memcpy_recv」に off を指定した場合、これらの 2 行の行頭に ＃を追加して無効化してください。これにより、Non Swappable Memory 領域を解放する動作となります。先頭が＃で始まる行はコメントとみなされ、無視されます。なお、MPI プログラムの実行に利用する NQSV キューのプロセスマネージャは Hydra の設定が必須となります。プロセスマネージャとして MPD を設定した場合、先行リクエストが停止した状態で緊急リクエストを実行する際、MPI プログラムが正常に実行できないことがあります。

4.9　NEC Parallel Debuggerの設定

NEC Parallel Debugger は、Eclipse PTP に対するデバッグ機能向けプラグインです。NEC Parallel

Debugger を使用する場合は、Eclipse　PTP を起動するホストおよびデバッグ対象のプログラムが動作する VH で、以下の設定を実施してください。

4.9.1　Eclipse PTP のインストール

Eclipse PTP を起動するホスト上に、Eclipse PTP をダウンロードします。Eclipse の公式サイトより最新版である Oxygen v3 をダウンロードして下さい。

次に、ダウンロードした圧縮ファイルをインストールディレクトリ（下記例では/INST-DIR）に展開します。

```
# tar zxvf eclipse-parallel-oxygen-3-linux-gtk-x86_64.tar.gz -C /INST-DIR
```

Eclipse PTP の実行コマンドは/INST-DIR/eclipse/eclipse になります。本コマンドが検索できるように環境変数 PATH を設定してください。

4.9.2　NEC Parallel Debugger plugin のインストール

Eclipse PTP を起動するホストで、インストールディレクトリ配下にある plugins ディレクトリに NEC Parallel Debugger プラグインを配置します。

```
# cp /opt/nec/ve/npdb/plugins/com.nec.ParallelDebugger_*.jar /INST-DIR/eclipse/plugins/
```

4.9.3　Eclipse PTP 必須ソフトウェアのインストール

VH に、Eclipse PTP のデバッグ実行に必須のソフトウェアをインストールします。インストールが必要なソフトウェアは以下の通りです。CPAN Search Site 等から最新版をダウンロードして下さい。

- Test-Simple
- Devel-GDB
- Expect
- IO-Tty

以下に各ソフトウェアのインストール手順を説明します。

- Test-Simple
  ダウンロードした圧縮ファイルを任意のディレクトリに展開し、make後、インストールします。

```
# tar zxvf Test-Simple-1.302183.tar.gz
# cd Test-Simple-1.302183
# perl Makefile.PL
```

```
# make
# make test
# make install
```

- Devel-GDB
  ダウンロードした圧縮ファイルを任意のディレクトリに展開し、make後、インストールします。

```
# tar zxvf Devel-GDB-2.02.tar.gz
# cd Devel-GDB-2.02
# perl Makefile.PL
# make
# make test
# make install
```

- IO-Tty
  ダウンロードした圧縮ファイルを任意のディレクトリに展開し、make後、インストールします。

```
# tar zxvf IO-Tty-1.16.tar.gz
# cd IO-Tty-1.16
# perl Makefile.PL
# make
# make test
# make install
```

- Expect
  ダウンロードした圧縮ファイルを任意のディレクトリに展開し、make後、インストールします。

```
# tar zxvf Expect-1.35.tar.gz
# cd Expect-1.35
# perl Makefile.PL
# make
# make test
# make install
```

4.9.4　ファイアウォールの設定

VH で、ファイアウォールが有効の場合，次の設定を行ってください。ファイアウォールが無効の場合は必要ありません。

NEC Parallel Debugger で利用する Eclipse PTP コンポーネントである SDM(Scalable Debug Manager)は、デバッグ対象のプロセス毎に起動し、既定値で 50000 番から 50079 番の範囲にある任意のポート番号を使用して外部からの接続を待ち受けます。以下のコマンドを実行して 50000 番ポートから 50079 番ポートを外部に開放して下さい。

```
# firewall-cmd --zone=public --permanent --add-port=50000-50079/tcp
# firewall-cmd --reload
```

以下のコマンドで先ほど開放したポートが出力されれば、ファイアウォールの設定は完了です。

```
# firewall-cmd --list-port --zone=public
50000-50079/tcp
```

なお、SDM が利用するポート番号は NPDB_SDM_PORTRANGE 環境変数で変更できます。以下に 51000 番ポートから 51100 番ポートを使用するように変更する例を示します。

```
# export NPDB_SDM_PORTRANGE=51000:51100
```

4.10 仮想メモリの設定の確認

VEOS の一部である ve_exec コマンド は 、"overcommit" 機能を利用して、1TB 以上の VH の仮想アドレス空間を使用します。"overcommit" 機能を有効にするためには、パラメータ vm.overcommit_memory の値が"2" 以外である必要があります。次のコマンドで確認してください。

```
# sysctl vm.overcommit_memory
vm.overcommit_memory = 0
```

⚠ 注意

VHの物理メモリを、1TB 以上のVH仮想アドレス空間に割り当てることはありません。

4.11 memlockリソースの設定の確認

memlock リソースは無制限に設定されている必要があります。本設定は VEOS のインストール時に自動的に行われます。次のコマンドで確認してください。

```
# ulimit -l
unlimited
```

memlock の変更はログイン中のユーザーには即時に適用されません。unlimited と表示されない場合は、ログインしなおして、再確認してください。

4.12 HugePagesの設定

VEOS(高速I/Oライブラリ、Partial Process Swapping)、NEC MPI、およびScaTeFS Clientは、HugePagesを使用します。HugePages設定コマンドを実行することで、HugePages と HugePages のオーバーコミット（利用可能な場合 HugePages 設定数を超えて利用可能とする限界値）を自動設定します。なお、HugePages は VH の環境に応じた値となり、HugePages のオーバーコミットは実装メモリの半分がデフォルト値です。

- Partial Process Swapping については次節を参照してください。
- NQSV のソケットスケジューリングを利用する場合は、NEC Network Queuing System V (NQSV) 利用の手引［管理編］18章ソケットスケジューリング 18.1.3 メモリアロケーションポリシー を必ず参照してください。ソケットスケジューリングで membind ポリシーを指定する場合、指定しない時の "VH numa ノード数倍" の HugePages 確保が必要です。/etc/opt/nec/ve/veos/ve-hugepages.conf で MEMBIND オプションを有効にしてください。

```
# vi /etc/opt/nec/ve/veos/ve-hugepages.conf
MEMBIND=YES
```

- SX-Aurora TSUBASAソフトウェア以外のソフトウェア用にHugePagesの設定を行っている場合は、そのソフトウェアが必要とするHugePagesのページ数を/etc/opt/nec/ve/veos/ve-hugepages.conf に指定します。

```
# vi /etc/opt/nec/ve/veos/ve-hugepages.conf
ADDITIONAL_HUGE_PAGES=<the number of Hugepages>
```

HugePages 設定コマンドの詳細は、VEOSドキュメント「VEプログラムの実行方法」を参照してください。

HugePages 設定コマンドの設定や、NEC MPI, ScaTeFS Client のパッケージの変更を行った場合は再度設定コマンドを実行して、設定値を更新してください。

```
# /opt/nec/ve/sbin/ve-set-hugepages
```

HugePages 設定は VH 起動時に自動的に行われます。systemd-sysctl コマンドによる設定とは

HugePages に関して排他となりますので、/etc/sysctl.confや、/etc/sysctl.d 配下等で HugePages、HugePages オーバーコミットの設定を行わないでください。

⚠ 注意

systemd-sysctl.service （sysctl コマンド）の設定ファイルに HugePages(nr_hugepages)、HugePages のオーバーコミット設定(nr_overcommit_hugepages) が有る場合、systemd-sysctl.service の値が設定される場合があります。 systemd-sysctl.service の設定ファイルについては Linux の manpage sysctl.d(5) を参照してください。

4.13 Partial Process Swapping機能の設定

Partial Process Swapping 機能は、プロセスの VE メモリを VH メモリまたはファイルにスワップアウトし、VE メモリを解放するための機能です。スワップアウトすることにより空いた VE メモリで、別のプロセスを実行する際に利用します。別のプロセスの実行完了後、VH メモリから VE メモリへスワップインし、元のプロセスを再開できます。

Partial Process Swapping 機能を使わない場合は、本節を飛ばしてください。

4.13.1 veswap.option の設定

各 VE で共通の設定をする場合は、/etc/opt/nec/ve/veos/ve-os-launcher.d/veswap.options を編集します。VE ごとに固有の値を設定する場合は、N を VE ノード番号として、/etc/opt/nec/ve/veos/ve-os-launcher.d/N ディレクトリを作成し、その配下に veswap.options を作成します。

veswap.options ファイルの設定内容は次の通りです。 ve-os-launcher@*=--ve-swap-mem-max=オプションを指定すると、VH メモリへのスワップアウトになります。ve-os-launcher@*=--ve-swap-file-path=と ve-os-launcher@*=--ve-swap-file-max=の両方を指定すると、ファイルへのスワップアウトになります。

表 12 veswap.options ファイルの設定内容

オプション	値	説明	補足
ve-os-launcher@*=--ve-swap-mem-max=	整数（MB）	VH メモリにスワップアウトする VE メモリの最大値	1VE あたりの値です。
ve-os-launcher@*=--ve-swap-file-path=	文字列	スワップアウトする先のファイルのパス	指定されたパスに、ホスト名と VE ノード番号が追加されます
ve-os-launcher@*=--ve-	整数（MB）	ファイルにスワップ	1VE あたりの値で

オプション	値	説明	補足
swap-file-max=		アウトする VE メモリの最大値	す。
ve-os-launcher@*=--ve-swap-file-user=	文字列	スワップファイルの所有者	オプションが指定されない場合のデフォルト値は"root"です。
ve-os-launcher@*=--ve-swap-file-group=	文字列	スワップファイルのグループ	オプションが指定されない場合のデフォルト値は"root"です。

次の 4 つの設定例について説明します。

1. VH メモリにスワップアウトする場合
2. ローカルファイルシステムにスワップアウトする場合
3. 共有ファイルシステムにスワップアウトする場合
4. VH メモリと、2 台の NVMe 上のファイルシステムに並列でスワップアウトする場合

例 1: VH メモリにスワップアウトする場合

veswap.options ファイルに次の行を追加してください。

```
ve-os-launcher@*=--ve-swap-mem-max=49152
```

この例では、1VE あたり最大で 48GB(49,152MB)の VE メモリをスワップアウトします。

例 2: ローカルファイルシステムにスワップアウトする場合

veswap.options ファイルに次の行を追加してください。

```
ve-os-launcher@*=--ve-swap-file-path=/var/opt/nec/ve/veos/swap-file
ve-os-launcher@*=--ve-swap-file-max=49152
```

この例では、1VE あたり最大で 48GB(49,152MB)のメモリを、/var/opt/nec/ve/veos/swap-file_<hostname>.N というパスのファイルにスワップアウトします。<hostname>はホスト名、N は VE ノード番号です。

例 3: 共有ファイルシステムにスワップアウトする場合

/mnt/share に共有ファイルシステムを mount して、/mnt/share/veswap ディレクトリの所有者とグループが"veswap"ユーザーと"veswap"グループである場合の例を説明します。

veswap.options ファイルに次の行を追加してください。

```
ve-os-launcher@*=--ve-swap-file-path=/mnt/share/veswap/swap-file
ve-os-launcher@*=--ve-swap-file-max=49152
ve-os-launcher@*=--ve-swap-file-user=veswap
ve-os-launcher@*=--ve-swap-file-group=veswap
```

この例では、1VE あたり最大で 48GB(49,152MB)のメモリを、/mnt/share/veswap/swap-file_<hostname>.N というパスのファイルにスワップアウトします。VEOS は、"veswap"ユーザーと"veswap"グループの権限で、ファイルを作成します。<hostname>はホスト名、N は VE ノード番号です。

例 4: VH メモリと、2 台の NVMe 上のファイルシステムに並列でスワップアウトする場合

1VE あたり最大で 48GB(49,152MB)の VE メモリをスワップアウトし、並列で、1VE あたり最大で 48GB(49,152MB)のメモリを NVMe 上のファイルシステムへスワップアウトする例を説明します。VE は 8 台搭載され、NVMe は 2 台搭載されている VH を想定し、VE0 から VE3 は 1 台の NVMe 上のファイルシステムへ、VE4 から VE7 はもう 1 台の NVMe 上のファイルシステムにスワップアウトします。2 台の NVMe 上にはそれぞれファイルシステムが作成され、/mnt/nvme0 と、/mnt/nvme1 に mount されている場合の例を説明します。

/etc/opt/nec/ve/veos/ve-os-launcher.d/veswap.options を確認し、設定がすべてコメントアウトされていることを確認します。

以下のコマンドを実行して、設定ファイルを保存する 8 つのディレクトリを作成してください。

```
# for d in `seq 0 7`; do mkdir /etc/opt/nec/ve/veos/ve-os-launcher.d/$d/; done
```

以下の 4 つのファイルを作成します。

```
/etc/opt/nec/ve/veos/ve-os-launcher.d/0/veswap.options
/etc/opt/nec/ve/veos/ve-os-launcher.d/1/veswap.options
/etc/opt/nec/ve/veos/ve-os-launcher.d/2/veswap.options
/etc/opt/nec/ve/veos/ve-os-launcher.d/3/veswap.options
```

内容は次の通りです。

```
ve-os-launcher@*=--ve-swap-mem-max=49152
```

```
ve-os-launcher@*=--ve-swap-file-path=/mnt/nvme0/swap-file
ve-os-launcher@*=--ve-swap-file-max=49152
```

以下の 4 つのファイルを作成します。

```
/etc/opt/nec/ve/veos/ve-os-launcher.d/4/veswap.options
/etc/opt/nec/ve/veos/ve-os-launcher.d/5/veswap.options
/etc/opt/nec/ve/veos/ve-os-launcher.d/6/veswap.options
/etc/opt/nec/ve/veos/ve-os-launcher.d/7/veswap.options
```

内容は次の通りです。

```
ve-os-launcher@*=--ve-swap-mem-max=49152
ve-os-launcher@*=--ve-swap-file-path=/mnt/nvme1/swap-file
ve-os-launcher@*=--ve-swap-file-max=49152
```

このような設定を行うと、スワップアウト先ファイルのパスは/mnt/nvme0/swap-file_<hostname>.N と/mnt/nvme1/swap-file_<hostname>.N になります。<hostname>はホスト名、N は VE ノード番号です。

4.13.2 VEOS の再起動

設定の反映のため、Hugepages 設定コマンドを実行し、VEOS を再起動してください。

```
# /opt/nec/ve/sbin/ve-set-hugepages
# systemctl restart 've-os-launcher@*'
```

"veswap -s"コマンドを実行して、Partial Process Swapping 機能が有効になっているか確認してください。もし、Partial Process Swapping 機能が有効になっていれば、"SWAPPED"列の値は"0"になります。Partial Process Swapping 機能が無効の場合は、"SWAPPED"列の値は"N/A"になります。veswap コマンドが"Failed to get online VE Nodes: No such file or directory"と表示する場合は、VEOS が再起動中です。その場合は、数秒待って再実行してください。

```
# /opt/nec/ve/bin/veswap -s
    VE     SWAPPED
    VE7          0
    VE6          0
    VE5          0
    VE4          0
    VE2          0
    VE3          0
    VE1          0
```

```
    VE0          0
```

NQSV では、Partial Process Swapping 機能を利用して、緊急ジョブを実行できます。詳細な設定方法は、「NEC Network Queuing System V (NQSV)　利用の手引[JobManipulator 編]」の「5.6 VE ジョブのサスペンド」を参照してください。

4.14 Process accountingの設定

Process accounting 機能は VE プロセスのプロセスアカウント情報を採取する機能です。Process accounting 機能が有効になっている場合、各プロセスの終了時にプロセスアカウントファイルに情報が記録されます。

4.14.1 プロセスアカウントサービスの開始

Process accounting を利用するには、次のコマンドで psacct-ve サービスを開始してください。

```
# for i in `seq 0 7`; do if [ -e /dev/veslot$i ]; then  systemctl enable psacct-ve@$i;  systemctl
restart psacct-ve@$i; fi done
```

psacct-ve サービスが有効になっていれば、VE プロセスの終了時にプロセスアカウント情報が記録されます。ve-lastcomm コマンドでこの情報を確認することが出来ます。ノード番号に対応したプロセスアカウントファイルを指定してください。

```
# /opt/nec/ve/bin/ve-lastcomm -f /var/opt/nec/ve/account/pacct_*
```

例えば、ノード番号 0 のプロセスアカウントファイルを見る方法は以下の通りです。

```
# /opt/nec/ve/bin/ve-lastcomm -f /var/opt/nec/ve/account/pacct_0
```

4.14.2 プロセスアカウントサービスの停止

Process accounting を利用しない場合は、次のコマンドで psacct-ve サービスを停止してください。

```
# for i in `seq 0 7`; do systemctl stop psacct-ve@$i; systemctl disable psacct-ve@$i $i; done
```

4.15 SX-Aurora TSUBASAでのプログラム実行について

SX-Aurora TSUBASA でプログラムを実行させるには大きく、以下の 2 つの方法があります。

- ve_execコマンドを直接指定して実行
- ve_execコマンド指定を省略して実行

- ve_execコマンドを直接使用して実行

SX-Aurora TSUBASAで、VE上で実行プログラムを実行させるためには、ve_execコマンドを利用する必要があります。本コマンドを省略する方法は、以下の「 ve_execコマンド指定を省略して実行」で説明します。

```
実行例：
MPIプログラムのケース
  $ mpirun -host 0 -ve 0-1 -np 16 /opt/nec/ve/bin/ve_exec ./mpi.lm
    mpi.lmは、お客様の実行プログラム名です。

非MPIプログラムのケース
  $ /opt/nec/ve/bin/ve_exec ./nonmpi.lm
    nonmpi.lmは、お客様の実行プログラム名です。
```

- ve_execコマンド指定を省略して実行

  VEOS 1.0.3以降では、Linuxのbinfmt機能を利用して、下記のようにve_execの指定を省略してプログラムを実行できるよう設定されています。

```
MPIプログラムのケース
  $ mpirun -host 0 -ve 0-1 -np 16 ./mpi.lm
    mpi.lmは、お客様の実行プログラム名です。

非MPIプログラムのケース
  $ ./nonmpi.lm
    nonmpi.lmは、お客様の実行プログラム名です。
```

上記の実行方法につきましては、バッチジョブ実行、会話型バッチ実行およびインタラクティブ実行に関わらず、共通です。

第5章　　アップデート

VH 上の SX-Aurora TSUBASA ソフトウェアを手動でアップデートする手順を説明します。なお、ローカルに yum リポジトリを構築している場合は、ローカルの yum リポジトリを更新してください(3.1.3 節参照)。

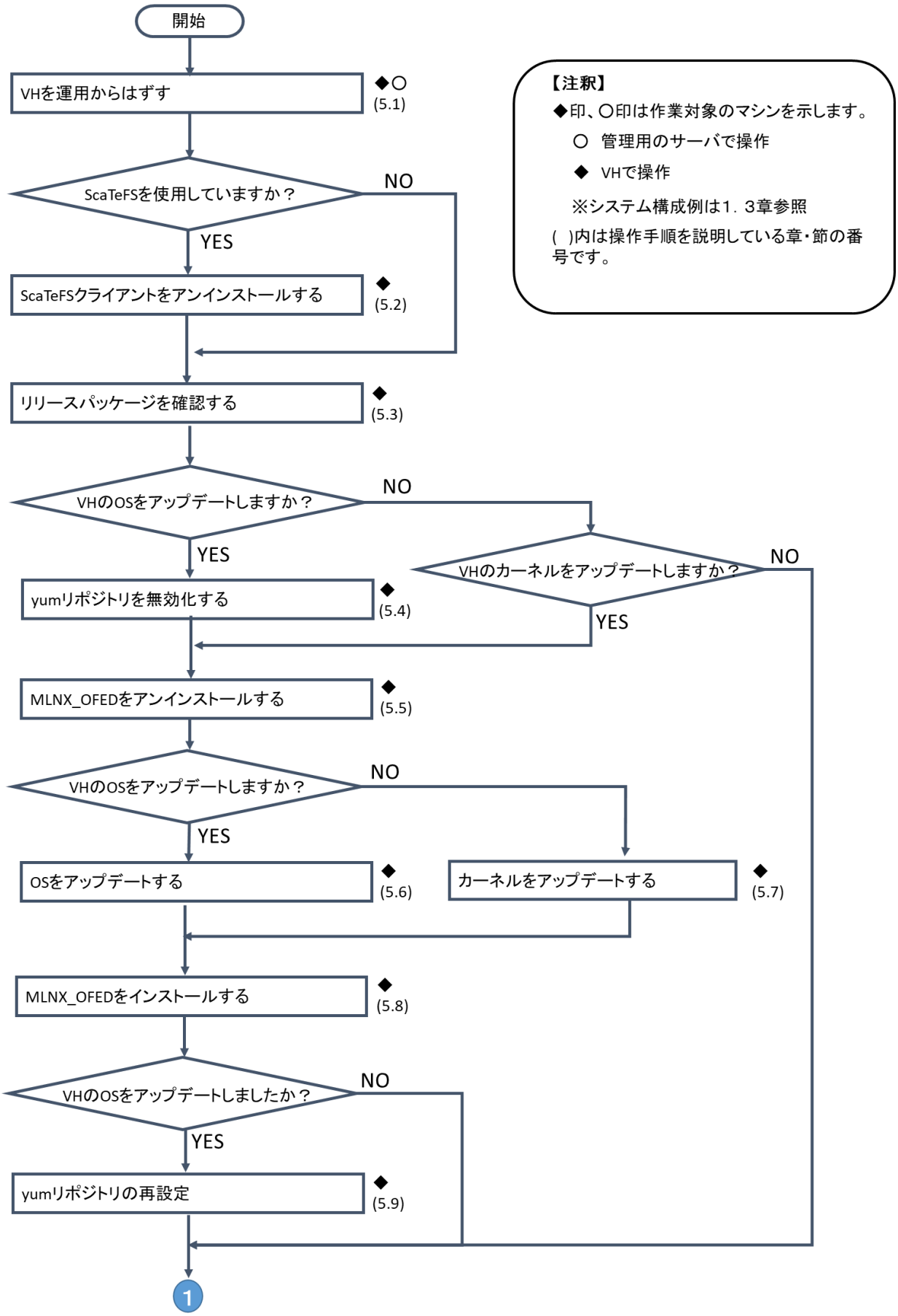
無償ソフトウェアのアップデートは、どなたでも可能です。

有償ソフトウェアのアップデートは、有償ソフトウェアを購入してPPサポートを契約している必要です。

VHにインストールされているSX-Aurora TSUBASAソフトウェアのアップデート作業の流れを以下に示します。

⚠ 注意

OS をアップデートする場合、MLNX_OFED もバージョン 4.x から 23.x へアップデートが 必要になる場合があります(2.3 節参照)。
この場合、MLNX_OFED のバージョン 4.x と 23.x 間では一部機能に互換性がないため、 アップデート前の環境で NEC MPI をスタティックリンク(既定値)した実行ファイルは アップデート後の環境で実行できません。


開始
VHを運用からはずす
◆○
(5.1)
ScaTeFSを使用していますか？
NO
YES
ScaTeFSクライアントをアンインストールする
◆
(5.2)
リリースパッケージを確認する
◆
(5.3)
VHのOSをアップデートしますか？
NO
YES
VHのカーネルをアップデートしますか？
NO
YES
yumリポジトリを無効化する
◆
(5.4)
MLNX_OFEDをアンインストールする
◆
(5.5)
VHのOSをアップデートしますか？
NO
YES
OSをアップデートする
◆
(5.6)
カーネルをアップデートする
◆
(5.7)
MLNX_OFEDをインストールする
◆
(5.8)
VHのOSをアップデートしましたか？
NO
YES
yumリポジトリの再設定
◆
(5.9)
1
【注釈】
◆印、○印は作業対象のマシンを示します。
○　管理用のサーバで操作
◆　VHで操作
※システム構成例は1．3章参照
(　)内は操作手順を説明している章・節の番号です。

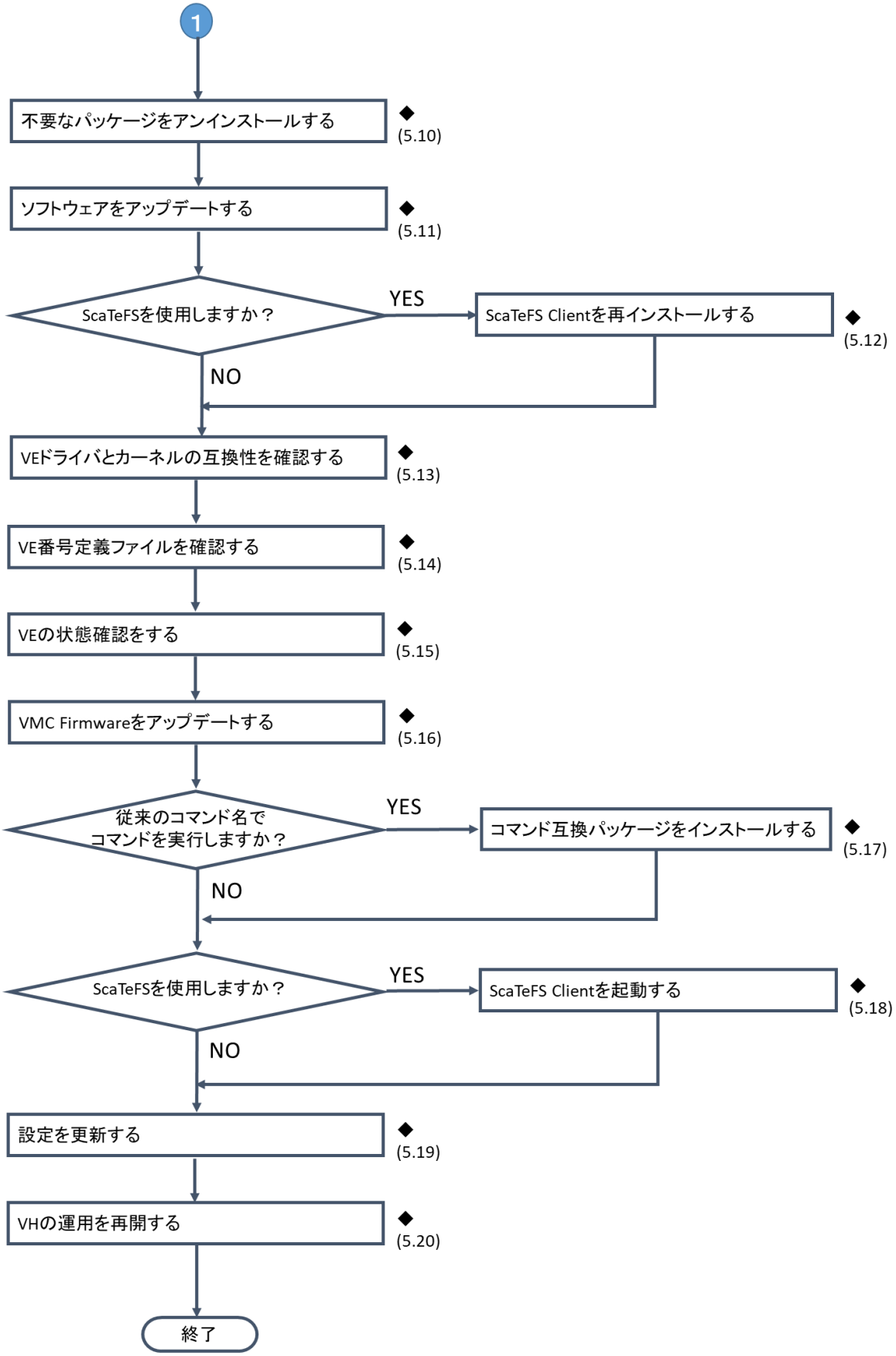

1
不要なパッケージをアンインストールする
(5.10)
ソフトウェアをアップデートする
(5.11)
ScaTeFSを使用しますか？
YES
ScaTeFS Clientを再インストールする
(5.12)
NO
VEドライバとカーネルの互換性を確認する
(5.13)
VE番号定義ファイルを確認する
(5.14)
VEの状態確認をする
(5.15)
VMC Firmwareをアップデートする
(5.16)
従来のコマンド名で
コマンドを実行しますか？
YES
コマンド互換パッケージをインストールする
(5.17)
NO
ScaTeFSを使用しますか？
YES
ScaTeFS Clientを起動する
(5.18)
NO
設定を更新する
(5.19)
VHの運用を再開する
(5.20)
終了

図 9 アップデート作業

5.1　VHを運用から外す

5.1.1　対象 VH をジョブスケジューラから外す

ジョブスケジューラを使用している場合は、以下の手順で、対象の VH をジョブスケジューラの運用から切り離してください。

(1) NQSV/Clientをインストールしているホスト上で、qmgr(1M)のunbind execution_queueサブコマンドを使って対象VH上のジョブサーバーを運用中のキューからアンバインドします。例えば、ジョブサーバーID が100 番のジョブサーバーをbqキューからアンバインドする場合は、以下のコマンドを実行します。

```
$ qmgr -Pm
Mgr: unbind execution_queue job_server bq job_server_id = 100
```

※会話キューの場合は unbind interactive_queue サブコマンドを使用してください。

(2) 上記操作の後、対象ジョブサーバー上にジョブが存在しないことをNQSV/Clientをインストールしているホスト上で、qstat(1)コマンドの -Jオプションで確認します。

```
$ qstat -J -Pm
```

qstat -Jの実行結果のJSVNO欄にステップ(1)でアンバインドしたジョブサーバーのジョブサーバーIDが表示されないことを確認してください。

対象ジョブサーバー上にジョブが存在した場合は、運用方針に従って以下のいずれかの操作を行ってください。

- 対象ジョブの実行終了を待つ
- 対象ジョブをqrerunによって別の実行ホストで再実行する
- 対象ジョブをqdelで削除する

(3) 対象VH上でsystemctlを使って対象VH上のジョブサーバーおよびランチャーを停止します。

```
# systemctl stop nqs-jsv.target
```

5.1.2　対象 VH を運用監視対象から外す

対象の VH に対して、Zabbix 等の運用監視ソフトウェアを使用した監視を行っている場合は、対象の VH の監視をメンテナンスモードにしてください。

5.2　ScaTeFS Clientのアンインストール(選択)

ScaTeFS Client を使用している場合は、以下の手順で、ScaTeFS Client サービスを停止した後に ScaTeFS Client をアンインストールしてください。

(1) アップデート対象VHでScaTeFSのファイルシステムをNFSでエクスポートしている場合、そのファイルシステムをマウントしている全てのNFSクライアントからそのファイルシステムをアンマウントしてください。アンマウントが完了したら(2)へ進んでください。

アップデート対象VHでScaTeFSのファイルシステムをNFSでエクスポートしていない場合は、(3)に進んでください。

(2) アップデート対象VHでnfs-serverサービスを停止します。

```
# systemctl stop nfs-server
```

(3) アップデート対象VHでScaTeFSのファイルシステムをSambaで共有として公開している場合、その共有を使用している全てのWindows等のCIFSクライアントでその共有へのアクセスを止めてください。アクセスの停止が完了したら(4)へ進んでください。

アップデート対象VHでScaTeFSのファイルシステムをSambaで共有として公開していない場合は、(5)に進んでください。

(4) アップデート対象VHでsmbサービスとnmbサービスを停止します。

```
# systemctl stop smb
# systemctl stop nmb
```

(5) アップデート対象VHでScaTeFSのファイルシステムをアンマウントします。

```
# umount -a -t scatefs
```

(6) アップデート対象VHでscatefs-clientサービスを停止します。

```
# systemctl stop scatefs-client
```

(7) 現在インストールされているScaTeFSのパッケージグループのIDを確認します(下線部)。

```
# yum group list installed -v 'scatefs-client-*'
(...)
インストール済みグループ:
   ScaTeFS Client for SX-Aurora TSUBASA and Mellanox OFED 4.9-0.1.7.0 (scatefs-client-
```

```
tsubasa)
完了
```

パッケージグループのIDを指定して、アンインストールを行います。

以下はパッケージグループのIDが "scatefs-client-tsubasa" の例です。

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh scatefs-client-tsubasa
# yum group remove scatefs-client-tsubasa
```

⚠ 注意

- /opt/nec/ve/sbin/TSUBASA-groups-remark.sh が存在しない場合、TSUBASA-groups-remark.shの実行はスキップしてください。

- yum group remove実行時に「グループから削除するパッケージがありません」というエラーになった場合は、以下のようにyum groups mark removeで削除を行ってください。

```
# yum group remove scatefs-client-tsubasa
No environment named scatefs-client-tsubasa exists
Maybe run: yum groups mark remove (see man yum)
グループから削除するパッケージがありません
# yum groups mark remove scatefs-client-tsubasa
Marked remove: scatefs-client-tsubasa
```

- PPサポート契約がなくパッケージグループ指定が利用できない場合、以下のようにScaTeFS Clientのパッケージを直接アンインストールしてください。

```
# yum remove '*scatefs-client-*'
```

(8)　ScaTeFS Clientのパッケージがすべてアンインストールされていることを確認します。

```
# rpm -qa | grep scatefs-client
(何も表示されない)
```

もしまだパッケージが残っている場合は、以下のように直接アンインストールしてください。

```
# rpm -qa | grep scatefs-client
scatefs-client-xxxxx
# yum remove scatefs-client-xxxxx
```

5.3　リリースパッケージの更新

yum リポジトリは、インターネット上のものをオンラインで利用する、もしくはローカルに構築してオフラインで利用することができます。

➢ オンラインの場合

以下のコマンドを実行し、インストールしてください。

- デフォルトのアーキテクチャをVE1にする場合

```
# yum install https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve1-3.0-1.noarch.rpm
```

- デフォルトのアーキテクチャをVE3にする場合

```
# yum install https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve3-3.0-1.noarch.rpm
```

➢ オフラインの場合

リリースパッケージを以下のURLから取得し、ローカルに構築した無償 yum リポジトリに配置してください。後述のコマンド例の場合、/path/to/repos 配下に配置してください。

- デフォルトのアーキテクチャをVE1にする場合

https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve1-3.0-1.noarch.rpm

その後、以下のコマンドでインストールしてください。

```
# cd /path/to/repos
# yum install ./TSUBASA-soft-release-ve1-3.0-1.noarch.rpm
```

- デフォルトのアーキテクチャをVE3にする場合

https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-ve3-3.0-1.noarch.rpm

その後、以下のコマンドでインストールしてください。

```
# cd /path/to/repos
# yum install ./TSUBASA-soft-release-ve3-3.0-1.noarch.rpm
```

⚠ 注意

インストール中に以下のようなメッセージが表示された場合は、既存の設定を上書きしないために設定ファイルを異なる名前(拡張子が.rpmnew)でインストールしたことを示しており、新機能のための設定が追加されている場合があります。新機能を使わない場合は *.rpmnew について特に作業はありません。新機能を使いたい場合は *.rpmnew ファイルの内容を確認し、更新されている設定を既存の設定ファイルに転記し、環境に合わせて修正を行ってください。

warning: /etc/yum.repos.d/TSUBASA-repo.repo created as /etc/yum.repos.d/TSUBASA-re

```
po.repo.rpmnew
warning: /etc/yum.repos.d/TSUBASA-restricted.repo created as /etc/yum.repos.d/TSUB
ASA-restricted.repo.rpmnew
```

詳細については「3.1 インストールのための事前準備」を参照してください。

5.4　yumリポジトリ無効化(OSアップデートを行う場合)

OS のアップデートを行う場合、以下の設定ファイルを編集し、各 yum リポジトリを無効化します。各 yum リポジトリの設定の enabled の値を 0 にしてください。

- /etc/yum.repos.d/TSUBASA-restricted.repo　アクセス制限付きyumリポジトリ設定ファイル
- /etc/yum.repos.d/TSUBASA-repo.repo　無償yumリポジトリ設定ファイル

5.5　MLNX_OFEDのアンインストール(選択)

OS のアップデート、カーネルのアップデート、または MLNX_OFED のアップデートを行う場合は、アップデート作業を行う前にインストール済みの MLNX_OFED を削除していただく必要があります。

まず、全ての VE プログラムが動作していないことを確認し、サービス停止スクリプトを実行してください

```
# /opt/nec/ve/sbin/terminate-all-ve-services
```

次に MLNX_OFED に依存関係のあるソフトウェアをアンインストールしてください。

```
# yum remove 'libibverbs-ve*' 'libmlx5-ve*' 'libveib*' 'kmod-ve_peermem*' 'libibumad-ve*' 'sharp-
  ve*' ve-memory-mapping 'libvedma-ve*' 'rdma-core-ve*' ve_peermem 'libvesharp*'
```

最後に MLNX_OFED をアンインストールしてください。

```
# /usr/sbin/ofed_uninstall.sh
```

(参考) NVIDIA 公式ホームページから MLNX_OFED のマニュアルをご参照頂けます。
https://docs.nvidia.com/networking/software/adapter-software/index.html

RHEL/CentOS 8.4 を使用している場合、MLNX_OFED のアンインストール後に libibverbs を再インストールしてください。

```
# yum install libibverbs
```

5.6　OSのアップデート(選択)

動作検証済みの OS へアップデートします。

SX-Aurora TSUBASA システムのソフトウェアの動作検証済みの LinuxOS とカーネルバージョンについては NEC サポートポータルの下記のコンテンツをご確認ください。

【SX-Aurora TSUBASA】検証済み Linux カーネル
https://www.support.nec.co.jp/View.aspx?id=3140106285

5.7　カーネルのアップデート(選択)

VH の Linux カーネルを、動作検証済みのカーネルにアップデートし、VH を再起動します。

SX-Aurora TSUBASA システムのソフトウェアの動作検証済みの LinuxOS とカーネルバージョンについては NEC サポートポータルの下記のコンテンツをご確認ください。

【SX-Aurora TSUBASA】検証済み Linux カーネル
https://www.support.nec.co.jp/View.aspx?id=3140106285

動作を検証していないカーネルへのアップデートを避けるため、/etc/yum.conf において kernel パッケージのアップデートを対象外としてください。/etc/yum.conf への記載例を下記に示します。'exclude=kernel*' を yum.conf に追記し、kernel パッケージのアップデートを抑止します。

```
# vi /etc/yum.conf
[main]
exclude=kernel*
 …
```

⚠ 注意

ホストマシンのカーネルバージョンについて、原則として動作検証済みのカーネルの中で最新のものにアップデートすることを推奨します。最新バージョンをご利用されない場合、Aurora ソフトウェアをアップデート後、ve_drv-kmodパッケージに含まれるve_drvモジュールがロードされずにVEがONLINEにならない場合があります。その場合、vp-kmod、ve_drv-kmodパッケージをダウングレードする必要があります。詳細は「VEドライバとLinuxカーネルの互換性確認」を参照してください。

5.8　MLNX_OFEDのインストール(アンインストールした場合)

InfiniBand を使用するシステムでは、「2.3 MLNX_OFED のインストール(選択)」を参照し、インストール対象 VH に MLNX_OFED をインストールしてください。

MLNX_OFED の SHARP 機能を使用しており、SHARP バージョンが 2.6.1 以前から 2.7.0 以降に変更になった場合、「4.8.4 NVIDIA SHARP の設定」を参照し、VH のファイアウォール設定を行ってください。

5.9　yumリポジトリの更新(OSアップデートを行った場合)

アップデートの場合も、インストール時と同様に、SX-Aurora TSUBASA ソフトウェア用の yum リポジトリを使用します。

5.5 節において、OS のバージョンアップを行った場合、yum リポジトリの更新が必要となります。

オンラインでインターネット上の yum リポジトリを利用可能な場合、アクセス制限付き yum リポジトリ設定ファイル(/etc/yum.repos.d/TSUBASA-restricted.repo)と無償 yum リポジトリ設定ファイル(/etc/yum.reops.d/TSUBASA-repo.repo)を編集し、各 yum リポジトリの baseurl を OS のバージョンにあった url に更新してください。また、5.4 節で enabled の値を 0 にした yum リポジトリについては、enabled の値を 1 に戻してください。

5.10 不要なパッケージのアンインストール

- **ve-memory-mapping パッケージのアンインストール（InfiniBand HCA 非搭載の場合）**

VH に InfiniBand HCA が搭載されていない場合は、ve-memory-mapping パッケージをアンインストールしてください。アンインストールせずにアップデートを実行しようとすると、依存関係によってアップデートが失敗します。

まず、全ての VE プログラムが動作していないことを確認し、サービス停止スクリプトを実行してください。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
```

次に、ve-memory-mapping パッケージをアンインストールしてください。

```
# yum remove ve-memory-mapping
```

5.11 アップデートの実行

各環境に合わせてアップデートを実行します。

シェル変数 TSUBASA_GROUPS に指定するグループは、InifiniBand HCA の搭載有無、有償ソフトウェアの購入有無に合わせて変更してください。また、アップデートしたくないソフトウェアはシェル変数 TSUBASA_GROUPS から除外してください。指定可能なグループについては 3.2.4 シェル変数 TSUBASA_GROUPS に指定可能なグループを参照してください。

⚠ 注意

- シェル変数TSUBASA_GROUPSには"scatefs-client-tsubasa"を指定しないでください。ScaTeFS Clientは、次の「5.12 ScaTeFS Clientの再インストール(選択)」でインストールします。
- SDK(コンパイラ, Numeric Library Collection, NLCPy)、MPIのマルチインスタンス対応のソフトウェアのインストールについては付録Aを参照してください。
- VEOS v3.0.2 のプロセスアカウンティング機能拡張に伴い、アカウンティングファイルのフォーマットが変更されます。既存の未圧縮のアカウンティングファイルは pacct-ve パッケージインストール時に新しい構造体を使用したフォーマットに自動変換されます。従来のファイルは同一ディレクトリに保存されます。

⚠ 注意

アップデート前にNEC MPI 2.xを使用していた場合、リポジトリ設定ファイルに記述されているbaseurlのmpi_el8をmpi_mofed5_el8に変更してください。

変更が必要なリポジトリ設定ファイルとリポジトリ名は以下の通りです。
設定ファイル：/etc/yum.repos.d/TSUBASA-repo.repo
リポジトリ名：nec-mpi-runtime, nec-mpi-community

SDK-SE、または、Compat C++を利用する場合は上記に加えて、以下のリポジトリ設定ファイルのリポジトリも変更してください。
設定ファイル：/etc/yum.repos.d/TSUBASA-restricted.repo
リポジトリ名：nec-mpi
(例)

```
/etc/yum.repos.d/TSUBASA-repo.repo
  [nec-mpi-runtime]
  …
  baseurl=https://sxauroratsubasa.sakura.ne.jp/repos/runtime/mpi/mpi_mofed5_el8

  [nec-mpi-community]
  …
  baseurl=https://sxauroratsubasa.sakura.ne.jp/repos/community/mpi/mpi_mofed5_el8
```

- 実行・コンパイル環境の例

  全ての VE プログラムが動作していないことを確認し、サービス停止スクリプトを実行してください。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
```

Auroraソフトウェアをアップデートします。

```
# TSUBASA_GROUPS="ve-devel ve-infiniband nec-sdk-devel nec-mpi-devel nqsv-execution"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

※上の例では、/opt/nec/ve/bin/nfort のパスで起動されるコンパイラは新しいバージョンに変更されます(ncc、nc++も同様です)。/opt/nec/ve/bin/nfort のパスで起動されるコンパイラの変更を行わない場合の手順は、付録 A を参照してください。

サービス再開スクリプトを実行してください。

```
# /opt/nec/ve/sbin/start-all-ve-services
```

- 実行環境の例

  全ての VE プログラムが動作していないことを確認し、サービス停止スクリプトを実行してください。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
```

Auroraソフトウェアをアップデートします。

```
# TSUBASA_GROUPS="ve-runtime ve-infiniband nec-sdk-runtime nec-mpi-runtime nqsv-execution"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install  $TSUBASA_GROUPS
```

サービス再開スクリプトを実行してください。

```
# /opt/nec/ve/sbin/start-all-ve-services
```

- フロントエンドの例

```
# TSUBASA_GROUPS="ve-frontend nec-sdk-frontend nec-mpi-frontend"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

※サービス停止スクリプトおよびサービス再開スクリプトを実行する必要はありません。

※上の例では、/opt/nec/ve/bin/nfortのパスで起動されるコンパイラは新しいバージョンに変更されます(ncc、nc++も同様です)。/opt/nec/ve/bin/nfortのパスで起動されるコンパイラの変更を行わない場合の手順は、付録 Aを参照してください。

5.12 ScaTeFS Clientの再インストール(選択)

「5.2 ScaTeFS Client のアンインストール(選択)」で ScaTeFS Client をアンインストールした場合は、以下の手順で ScaTeFS Client を再インストールしてください。このときインストールされる ScaTeFS Client は、yum リポジトリに格納されている最新のパッケージになります。

(1) ScaTeFS Clientを再インストールします。

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh scatefs-client-tsubasa
# yum group install scatefs-client-tsubasa
```

⚠ 注意

- PPサポート契約がなくパッケージグループ指定が利用できない場合は、「インターネット配信製品ダウンロードサービス」で入手したzipファイルを用意し、ScaTeFS Clientのパッケージを直接インストールしてください。この手順については「3.2.3 有償ソフトウェアを未購入、または、PPサポート契約なしの場合」を参照してください。

(2) ScaTeFS Clientをアンインストールした際に、設定ファイルのバックアップファイル /etc/scatefs/client/ibdevice.conf.rpmsave が作成されている場合は、以下の手順で設定ファイルのリストアを行います。ファイルが存在しない場合、この手順はスキップしてください。

```
# mv /etc/scatefs/client/ibdevice.conf.rpmsave /etc/scatefs/client/ibdevice.conf
mv: overwrite '/etc/scatefs/client/ibdevice.conf'? y
```

5.13 VEドライバとLinuxカーネルの互換性確認

アップデートした VE ドライバがホストマシンの Linux カーネルと互換性がない場合、VE ドライバのパッケージ（vp-kmod、ve_drv-kmod パッケージ）のダウングレードが必要な場合があります。

ホストマシンのカーネルは、原則として動作検証済みのカーネルの中で最新のものにアップデートすること(2.2 動作検証済みカーネルへのアップデート)を推奨します。最新バージョンをご利用されない場合、Aurora ソフトウェアをインストールしても、ve_drv-kmod パッケージに含まれる ve_drv モジュールがロードされずに VE が ONLINE にならない場合があります。その場合、vp-kmod、ve_drv-kmod パッケージをダウングレードする必要があります。

5.13.1 ダウングレードの要否確認

ダウングレードの要否を確認します。まずは、ve_drv モジュールがロードされているかを確認します。

```
# lsmod | grep ve_drv
ve_drv              262144  128
vp                   20480  1 ve_drv
```

ve_drv が表示された場合、ダウングレードは不要です。5.14 に進んでください。

```
# lsmod | grep ve_drv
（何も表示されない）
```

何も表示されなかった場合、ロードできていません。次のコマンドで VH の Linux カーネルバージョンを確認してください。

```
# uname -r
4.18.0-553.8.1.el8_10.x86_64
```

次に、vp-kmod および ve_drv-kmod パッケージのバージョンを確認して、対応カーネルモジュールがインストールされていることを確認してください。

【SX-Aurora TSUBASA】検証済み Linux カーネル
https://www.support.nec.co.jp/View.aspx?id=3140106285

```
# rpm -q vp-kmod
vp-kmod-3.0.1-4.18.0_372.32.1_1.el8.x86_64
# rpm -q ve_drv-kmod
ve_drv-kmod-3.0.1-4.18.0_372.32.1.el8_6.x86_64_1.el8.x86_64
```

Linux カーネルバージョンより vp-kmod または ve_drv-kmod パッケージのバージョンが新しい場合、vp-kmod、ve_drv-kmod パッケージのダウングレードが必要です。

5.13.2 ダウングレードの実行

次のコマンドを実行して、vp-kmod、ve_drv-kmod パッケージのダウングレードを実行してください。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
# yum downgrade vp-kmod ve_drv-kmod
# /opt/nec/ve/sbin/start-all-ve-services
```

5.14 VE番号定義ファイルの確認・設定

1.2.1 ハードウェアの表 3 サーバーモデルに記載されたモデルをご利用のお客様は、以下のコマンドで、VE 番号定義ファイルの設定・確認が必要です。

以下のコマンドで、VE 番号定義ファイルが存在するか確認します。VE 番号定義ファイルが存在する場合は 5.15 に進んでください。

```
# ls -l /etc/opt/nec/ve/veos/ve_nodes.json
-rw-r--r--. 1 root root 274  2月 22  2023 /etc/opt/nec/ve/veos/ve_nodes.json
```

VE 番号定義ファイルが存在しない場合、以下のコマンドで VE 番号定義ファイルを配置します。

```
# /opt/nec/ve/mmm/analysis/sbin/mmm-config-json.sh 2
```

VE 番号定義ファイルが配置されたことを確認し、VE サービスを再起動します。

```
# ls -l /etc/opt/nec/ve/veos/ve_nodes.json
-rw-r--r--. 1 root root 274  2月 22  2023 /etc/opt/nec/ve/veos/ve_nodes.json
# /opt/nec/ve/sbin/terminate-all-ve-services
# /opt/nec/ve/sbin/start-all-ve-services
```

5.15 VEの状態確認

vecmd コマンドで VE カードが ONLINE であることを確認します。VE が ONLINE になるまでには数分程度かかることがあります。

```
# /opt/nec/ve/bin/vecmd state get
Vector Engine MMM-Command v1.0.0
Command:
state -N 0 get
--------------------------------------------------------------------------------
VE0 [03:00.0] [ ONLINE ] Last Modif:2017/11/29 10:18:00
--------------------------------------------------------------------------------
Result: Success
```

⚠ 注意

"UNINITIALIZED"や"<OFFLINE>"が表示される場合は、しばらくお待ちください。
しばらくお待ちいただいてもVE stateが"ONLINE"へ切り替わらない場合、以下のコマンドを実行し、サービスを再起動してください。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
# /opt/nec/ve/sbin/start-all-ve-services
```

また、"There is no executable ve card!"と表示される場合についても、同様の方法でサービスを再起動してください。

"UNAVAILABLE"が表示された場合は、rebootコマンドにて、VHを再起動してください。

```
# reboot
```

再起動しても、なお"UNAVAILABLE"が表示される場合、故障の可能性があります。以下の手順書を参考に切り分け作業を行って下さい。

NEC Aurora Forum (https://sxauroratsubasa.sakura.ne.jp/Documentation(Japanese))
"Vector Engine 2.0 障害切り分け手順書"

5.16 VMC Firmware (VMCFW) の アップデート

次のコマンドを実行して、VMCFWの更新要否を判定します。

```
# /opt/nec/ve/bin/vecmd fwup check
```

「Updating VMCFW is required.」と出力された場合はVMCFWの更新が必要です。「Updating VMCFW is not required.」と出力された場合はVMCFWの更新が不要です。

VMCFWの更新が必要な場合は、次の手順でファームウェアをアップデートします。

(1)　vecmdコマンドにて、VEの状態を変更し、ファームウェアをアップデートします。

- VE1/VE2に対してアップデートする場合

```
# /opt/nec/ve/bin/vecmd state set off
# /opt/nec/ve/bin/vecmd state set mnt
# /opt/nec/ve/bin/vecmd fwup vmc aurora_MK10.bin
```

※ファームウェアの更新には、数分程度かかります。

VHを再起動します。

```
#reboot
```

- VE3に対してアップデートする場合

```
# /opt/nec/ve/bin/vecmd state set off
# /opt/nec/ve/bin/vecmd fwup
```

※ファームウェアの更新には、10分程度かかります。

(2)　VHをshutdownして再度電源ONします（VE1と異なるので注意してください）

```
# shutdown -h now
```

(3)　VH起動後、rootユーザーでログインします。

(4)　VEの状態確認

vecmd コマンドで VE カードが ONLINE であることを確認します。VE が ONLINE になるまでには数分程度かかることがあります。

```
# /opt/nec/ve/bin/vecmd state get
Vector Engine MMM-Command v1.0.0
Command:
state -N 0 get
--------------------------------------------------------------------------------
VE0 [03:00.0] [ ONLINE ] Last Modif:2017/11/29 10:18:00
--------------------------------------------------------------------------------
```

```
Result: Success
```

⚠ 注意

"UNINITIALIZED"や"<OFFLINE>"が表示される場合は、しばらくお待ちください。また、"ONLINE"にならない、もしくは表示されないVEがある場合、一度VEをshutdownしてから電源を入れ直してください。

5.17 コマンド互換パッケージのインストール(選択)

2023 年 3 月リリースの SX-Aurora TSUBASA ソフトウェアより、Linux から移植したコマンドは ve-というプレフィックスが付きます。2023 年 3 月リリース以前の Linux から移植したコマンドと互換性を保つ必要がある場合は、コマンド互換パッケージをインストールしてください。

環境により、コマンド互換パッケージが異なります。各環境に合わせて適当なパッケージをインストールしてください。

- 実行・コンパイル環境

```
# yum install ve-command-name-compat-devel
```

- 実行環境

```
# yum install ve-command-name-compat-runtime
```

- フロントエンド環境

```
# yum install ve-command-name-compat-front-end
```

5.18 ScaTeFS の起動(選択)

ScaTeFS Client を使用している場合は、VH を再起動して、ScaTeFS Client を起動します。VMCFW の更新のため、VH を再起動済みの場合は改めて再起動する必要はありません。

```
# reboot
```

5.19 ソフトウェアの設定の更新

アップデート後、必要なリソースが変わることや、新しい機能が利用可能になることがあります。以下の節を参照して、設定を更新してください。

- 4.8.5 HCAデバイス名の固定化

  2024/10月末のリリースからこの設定が必要になりました。InfiniBandを使用するシステ

ムでは、未設定の場合、設定を行ってください。設定を行った後は、「4.8.6 各VEが使用するHCAと通信方式の設定（選択）」を参照し、necmpi.confの内容を変更してください。

- 4.12 HugePagesの設定

  2021/9月末のリリースから HugePages は自動設定されるようになりました。本機能を利用するためには /etc/sysctl.conf 等 sysctl 用の設定ファイルから HugePages の設定をコメントアウトする必要があります。

```
# vi /etc/sysctl.conf
## vm.nr_hugepages = 384 # コメントアウト
```

- 4.13 Partial Process Swapping機能の設定

5.20 VHの運用を開始する

5.20.1 対象 VH で全ての VE の ONLINE 状態を確認する

vecmd コマンドで VE カードが ONLINE であることを確認します。VE が ONLINE になるまでには数分程度かかることがあります。

```
# /opt/nec/ve/bin/vecmd state get
Vector Engine MMM-Command v1.0.0
Command:
state -N 0 get
--------------------------------------------------------------------------------
VE0 [03:00.0] [ ONLINE           ] Last Modif:2017/11/29 10:18:00
--------------------------------------------------------------------------------
Result: Success
```

⚠ 注意

"UNINITIALIZED"や"<OFFLINE>"が表示される場合は、しばらくお待ちください。

5.20.2 対象 VH を運用監視対象に入れる

対象の VH に対して、Zabbix 等の運用監視ソフトウェアを使用した監視を行っている場合は、ここでメンテナンスモードを解除し監視対象に戻してください。

5.20.3 対象 VH をジョブスケジューラに追加する

ジョブスケジューラを使用している場合は、以下の手順で、対象の VH をジョブスケジューラの運用に組み込んでください。

(1) 対象VH上でsystemctlを使って対象VH上のジョブサーバーおよびランチャーを起動します。

```
# systemctl start nqs-jsv
```

(2) NQSV/Clientをインストールしているホスト上で、qmgr(1M)のbind execution_queueサブコマンドを使って対象VH上のジョブサーバーを運用するキューにバインドします。例えば、ジョブサーバーID が100 番のジョブサーバーをbqキューにバインドする場合は、以下のコマンドを実行します。

```
$ qmgr -Pm
Mgr: bind execution_queue job_server bq job_server_id = 100
```

※会話キューの場合はbind interactive_queueサブコマンドを使用してください。

第6章　アンインストール

各ソフトウェアのアンインストール手順を説明します。

6.1　VHを運用から外す

6.1.1　対象 VH をジョブスケジューラから外す

ジョブスケジューラを使用している場合は、以下の手順で、対象の VH をジョブスケジューラの運用から切り離してください。

(1) NQSV/Clientをインストールしているホスト上で、qmgr(1M)のunbind execution_queueサブコマンドを使って対象VH上のジョブサーバーを運用中のキューからアンバインドします。例えば、ジョブサーバーIDが100番のジョブサーバーをbqキューからアンバインドする場合は、以下のコマンドを実行します。

```
$ qmgr -Pm
Mgr: unbind execution_queue job_server bq job_server_id = 100
```

※会話キューの場合はunbind interactive_queueサブコマンドを使用してください。

(2) 上記操作の後、対象ジョブサーバー上にジョブが存在しないことをNQSV/Clientをインストールしているホスト上で、qstat(1)コマンドの -Jオプションで確認します。

```
$ qstat -J -Pm
```

qstat -Jの実行結果のJSVNO欄に(1)でアンバインドしたジョブサーバーのジョブサーバーIDが表示されないことを確認してください。

対象ジョブサーバー上にジョブが存在した場合は、運用方針に従って以下のいずれかの操作を行ってください。

- 対象ジョブの実行終了を待つ
- 対象ジョブをqrerunによって別の実行ホストで再実行する
- 対象ジョブをqdelで削除する

(3) 対象VH上でsystemctlを使って対象VH上のジョブサーバーおよびランチャーを停止します。

```
systemctl stop nqs-jsv.target
```

6.1.2　対象 VH を運用監視対象から外す

対象の VH に対して、Zabbix 等の運用監視ソフトウェアを使用した監視を行っている場合は、対象の VH の監視をメンテナンスモードにしてください。

6.2　ScaTeFS Clientのアンインストール(選択)

ScaTeFS Client を使用している場合は、以下の手順で、ScaTeFS Client サービスを停止した後に ScaTeFS Client をアンインストールしてください。

(1) アンインストール対象VHでScaTeFSのファイルシステムをNFSでエクスポートしている場合、そのファイルシステムをマウントしている全てのNFSクライアントからそのファイルシステムをアンマウントしてください。アンマウントが完了したら(2)へ進んでください。アンインストール対象VHでScaTeFSのファイルシステムをNFSでエクスポートしていない場合は、(3)に進んでください。

(2) アンインストール対象VHでnfs-serverサービスを停止します。

```
# systemctl stop nfs-server
```

(3) アンインストール対象VHでScaTeFSのファイルシステムをSambaで共有として公開している場合、その共有を使用している全てのWindows等のCIFSクライアントでその共有へのアクセスを止めてください。アクセスの停止が完了したら(4)へ進んでください。

アンインストール対象VHでScaTeFSのファイルシステムをSambaで共有として公開していない場合は、(5)に進んでください。

(4) アンインストール対象VHでsmbサービスとnmbサービスを停止します。

```
# systemctl stop smb
# systemctl stop nmb
```

(5) アンインストール対象VHでScaTeFSのファイルシステムをアンマウントします。

```
# umount -a -t scatefs
```

(6) アンインストール対象VHでscatefs-clientサービスを停止します。

```
# systemctl stop scatefs-client
```

(7)　現在インストールされているScaTeFSのパッケージグループのIDを確認します(下線部分)。

```
# yum group list installed -v 'scatefs-client-*'
(...)
インストール済みグループ:
   ScaTeFS Client for SX-Aurora TSUBASA and Mellanox OFED 4.9-0.1.7.0 (scatefs-client-tsubasa)
完了
```

パッケージグループのIDを指定して、アンインストールを行います。

以下はパッケージグループのIDが "scatefs-client-tsubasa" の例です。

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh scatefs-client-tsubasa
# yum group remove scatefs-client-tsubasa
```

⚠ 注意

- /opt/nec/ve/sbin/TSUBASA-groups-remark.sh が存在しない場合、TSUBASA-groups-remark.shの実行はスキップしてください。

- yum group remove実行時に「グループから削除するパッケージがありません」というエラーになった場合は、以下のようにyum groups mark removeで削除を行ってください。

```
# yum group remove scatefs-client-tsubasa
No environment named scatefs-client-tsubasa exists
Maybe run: yum groups mark remove (see man yum)
グループから削除するパッケージがありません
# yum groups mark remove scatefs-client-tsubasa
Marked remove: scatefs-client-tsubasa
```

- PPサポート契約がなくパッケージグループ指定が利用できない場合、以下のようにScaTeFS Clientのパッケージを直接アンインストールしてください。

```
# yum remove '*scatefs-client-*'
```

(8)　ScaTeFS Clientのパッケージがすべてアンインストールされていることを確認します。

```
# rpm -qa | grep scatefs-client
(何も表示されない)
```

もしまだパッケージが残っている場合は、以下のように直接アンインストールしてください。

```
# rpm -qa | grep scatefs-client
scatefs-client-xxxxx
# yum remove scatefs-client-xxxxx
```

6.3　アンインストールの実行

全ての VE プログラムが動作していないことを確認し、サービス停止スクリプトを実行してください

```
# /opt/nec/ve/sbin/terminate-all-ve-services
```

- Auroraソフトウェアのアンインストール

```
# TSUBASA_GROUPS="ve-devel ve-infiniband nec-sdk-devel nec-mpi-devel nqsv-execution ve-runtime
nec-sdk-runtime nec-mpi-runtime ve-frontend"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group remove $TSUBASA_GROUPS
# yum remove TSUBASA-soft-release-ve[13]
```

NEC から提供されたパッケージが、まだインストールされていないか確認します。インストールされている場合は、yum remove コマンドを実行してアンインストールしてください。ただし、SX-Aurora TSUBASA 向けソフトウェア以外の NEC 製ソフトウェアをインストールしている場合は、そのソフトウェアはアンインストール不要です。

```
# rpm -qa --qf '%{VENDOR} %{NAME}-%{VERSION}-%{RELEASE}.%{ARCH}\n'| grep NEC
```

付録 A　マルチインスタンス対応のソフトウェアのインストール

コンパイラ

- 特定バージョンのコンパイラのインストール

特定のバージョンのコンパイラを yum コマンドで以下のようにインストールします。X.X.X はインストールしたいコンパイラのバージョン番号です。

実行・コンパイル環境

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-compiler-devel-X.X.X
# yum group install nec-compiler-devel-X.X.X
```

実行環境

実行環境は最新の物件が過去物件を包含しているため、特定のバージョンのインストールは不要です。「5.11 アップデートの実行」のアップデート手順に従って最新物件の適用を行ってください。

フロントエンド

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-compiler-frontend-X.X.X
# yum group install nec-compiler-frontend-X.X.X
```

インストールしたコンパイラは、/opt/nec/ve/bin/nfort-X.X.X (X.X.X はバージョン)で起動できます。/opt/nec/ve/bin/nfort のパスで起動されるコンパイラは変更されません。/opt/nec/ve/bin/nfort のパスで起動されるコンパイラを変更する手順は次節を参照してください。

- /opt/nec/ve/bin のコンパイラの変更

yum コマンドで/opt/nec/ve/bin/nfort のパスで起動されるコンパイラを変更します。(ncc、nc++も変更されます。)

実行・コンパイル環境

```
# yum remove nec-nfort-inst
# yum remove nec-nc++-inst
# yum install nec-nfort-inst-X.X.X-X.noarch
# yum install nec-nc++-inst-X.X.X-X.noarch
```

- /opt/nec/ve/bin のコンパイラを変更せずにアップデート

以下の例のようにシェル変数 TSUBASA_GROUPS を指定します。これにより/opt/nec/bin/nfort で起動されるコンパイラを変更せずにアップデートできます。(ncc、nc++も同様です。)

- 実行・コンパイル環境の例

  全ての VE プログラムが動作していないことを確認し、サービス停止スクリプトを実行してください。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
```

  Auroraソフトウェアをアップデートします。

```
# TSUBASA_GROUPS="ve-devel ve-infiniband nec-sdk-devel-nodefault nec-mpi-devel nqsv-execution"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

  サービス再開スクリプトを実行してください。

```
# /opt/nec/ve/sbin/start-all-ve-services
```

- 実行環境の例

  全ての VE プログラムが動作していないことを確認し、サービス停止スクリプトを実行してください。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
```

  Auroraソフトウェアをアップデートします。

```
# TSUBASA_GROUPS="ve-runtime ve-infiniband nec-sdk-runtime nec-mpi-runtime nqsv-execution"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

  サービス再開スクリプトを実行してください。

```
# /opt/nec/ve/sbin/start-all-ve-services
```

- フロントエンドの例

```
# TSUBASA_GROUPS="ve-frontend nec-sdk-frontend-nodefault nec-mpi-frontend"
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh $TSUBASA_GROUPS
# yum group install $TSUBASA_GROUPS
```

アップデートしたコンパイラは、/opt/nec/ve/bin/nfort-X.X.X (X.X.X はバージョン)で起動できます。以下は、バージョン 2.4.1 のコンパイラを起動するときの例です。

```
$ /opt/nec/ve/bin/nfort-2.4.1 t.f90
```

MPI

特定のバージョンの MPI を yum コマンドで以下のようにインストールします。"X-X-X"はインストールしたい MPI のバージョン番号 X.X.X に対応します。インストール前に以下の注意をご確認ください。

実行・コンパイル環境

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-mpi-devel-X-X-X
# yum group install nec-mpi-devel-X-X-X
```

実行環境

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-mpi-runtime-X-X-X
# yum group install nec-mpi-runtime-X-X-X
```

フロントエンド

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-mpi-frontend-X-X-X
# yum group install nec-mpi-frontend-X-X-X
```

⚠ 注意

バージョン指定のインストール時、/opt/nec/ve/bin に配置される mpirun のようなコマンドは更新されません。/opt/nec/ve/bin に配置するコマンドを更新する場合、以下のコマンドで nec-mpi-runtime パッケージを指定バージョンに更新してください。

```
# yum install nec-mpi-runtime-X.X.X
```

/opt/nec/ve/binに配置するコマンドを含めて最新バージョンに更新する場合は、バージョンを指定しないインストールを行ってください。

バージョンを指定してインストールする場合、nec-mpi-runtime-common パッケージが必要になります。作業の開始前に以下のコマンドで nec-mpi-runtime-common パッケージのインストールを完了させてください。

```
# yum install nec-mpi-runtime-common
```

Numeric Library Collection

特定のバージョンの Numeric Library Collection を yum コマンドで以下のようにインストールします。X.X.X はインストールしたい Numeric Library Collection のバージョン番号です。

実行・コンパイル環境

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-nlc-devel-X.X.X
# yum group install nec-nlc-devel-X.X.X
```

実行環境

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-nlc-runtime-X.X.X
# yum group install nec-nlc-runtime-X.X.X
```

フロントエンド

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-nlc-frontend-X.X.X
# yum group install nec-nlc-frontend-X.X.X
```

NLCPy

特定のバージョンの NLCPy を yum コマンドで以下のようにインストールします。X.X.X はインストールしたい NLCPy のバージョン番号です。

実行・コンパイル環境

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-nlcpy-devel-X.X.X
# yum group install nec-nlcpy-devel-X.X.X
```

実行環境

```
# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nec-nlcpy-runtime-X.X.X
# yum group install nec-nlcpy-runtime-X.X.X
```

付録 B　インストールされるパッケージ

SX-Aurora ソフトウェアパッケージ一覧

表 13 インストールパッケージ一覧

ソフトウェアコンポーネント	パッケージファイル	入手方法 (製品名) (注)
VEOS	autoconf-ve automake-ve coreutils-ve gdb-ve glibc-ve1 glibc-ve1-devel glibc-ve3 glibc-ve3-devel glibc-ve-wrapper kheaders-ve1 kheaders-ve3 libgcc-ve1-static libgcc-ve3-static libsysve-ve1 libsysve-ve1-devel libsysve-ve3 libsysve-ve3-devel libthread_db-ve1 libthread_db-ve3 libtool-ve libved procps-ng-ve psacct-ve psmisc-ve strace-ve sysstat-ve time-ve util-linux-ve ve_drv-kmod vedebuginfo velayout venumainfo veoffload-aveo veoffload-aveo-devel veoffload-aveorun-ve1 veoffload-aveorun-ve1-devel veoffload-aveorun-ve3 veoffload-aveorun-ve3-devel veoffload-veda veoffload-veda-devel veoffload-veda-ve1 veoffload-veda-ve3 veos veos-devel veos-headers veosinfo veosinfo3	A

ソフトウェアコンポーネント	パッケージファイル	入手方法 (製品名) (注)
	veos-libveptrace vesysinit vesysinit-udev vp-kmod veswap libvhcall-fortran-ve1 libvhcall-fortran-ve1-devel libvhcall-fortran-ve3 libvhcall-fortran-ve3-devel libsmartcols-ve cmake-ve veosctl ve-command-name-compat-devel ve-command-name-compat-front-end ve-command-name-compat-runtime	
InfiniBand for SX-Aurora TSUBASA	libveib-mofedX.X kmod-ve_peermem-mofedX.X rdma-core-ve1-mofedX.X libibverbs-ve1-mofedX.X libibumad-ve1-mofedX.X sharp-ve1-X.X.X libvesharp-X.X.X rdma-core-ve3-mofedX.X libibverbs-ve3-mofedX.X libibumad-ve3-mofedX.X sharp-ve3-X.X.X	A
MMM	ftmon ftmon-ve3 mmm mmm-config mmm-analysis mmm-analysis-ve3 mmm-msl mmm-msl-ve3 rtmon rtmon-ve3 ve-firmware	A
VMC Firmware	vmcfw vmcfw-ve3	A
ライセンスアクセスライブラリ	aurlic-lib	A
ScaTeFS Client	kmod-scatefs-client-modules-mofedX.X scatefs-client-libscatefsib_ve scatefs-client-mount-utils scatefs-client-rcli-utils scatefs-client-sys-utils scatefs-client-usr-utils	B(ScaTeFS/Client)
NEC MPI	nec-mpi-devel-X-X-X nec-mpi-devel-ve3-X-X-X nec-mpi-libs-X-X-X nec-mpi-libs-ve3-X-X-X nec-mpi-utils-X-X-X nec-mpi-utils-ve3-X-X-X nec-mpi-runtime-X-X-X nec-mpi-runtime-ve3-X-X-X	A、B両方で入手可能

ソフトウェアコンポーネント	パッケージファイル	入手方法 (製品名) (注)
	nec-mpi-runtime nec-mpi-runtime-common nec-mpi-runtime-pbs	
チューニングツール	nec-veperf-bin nec-veperf-ve1-devel nec-veperf-ve3-devel nec-veperf-ve1-libs nec-veperf-ve3-libs nec-ftraceviewer	A、B両方で入手可能
NEC Parallel Debugger	nec-paralleldebugger	B
NQSV/JobServer	NQSV-JobServer	B(NQSV/JobServer)
NQSV/Client	NQSV-Client	B(NQSV/Resource Manager)
Numeric Library Collection	nec-asl-ve-X.X.X nec-aslfftw-ve-X.X.X nec-blas-ve-X.X.X nec-heterosolver-ve-X.X.X nec-lapack-ve-X.X.X nec-nlc-base-X.X.X nec-nlc-doc-X.X.X nec-sblas-ve-X.X.X nec-sca-ve-X.X.X nec-scalapack-ve-X.X.X nec-asl-ve-devel-X.X.X nec-aslfftw-ve-devel-X.X.X nec-blas-ve-devel-X.X.X nec-heterosolver-ve-devel-X.X.X nec-lapack-ve-devel-X.X.X nec-sblas-ve-devel-X.X.X nec-sca-ve-devel-X.X.X nec-scalapack-ve-devel-X.X.X	A、B両方で入手可能
NLCPy	nec-python36-nlcpy-ve-X.X.X nec-python36-numpy-X.X.X-Y.Y.Y	A、B両方で入手可能
mpi4py-ve	nec-python36-mpi4py-ve-X.X.X	A、B両方で入手可能
binutils	binutils-ve	A、B両方で入手可能
C/C++ compiler	nec-nc++-X.X.X nec-nc++-inst nec-nc++-doc-X.X.X nec-nc++-shared-X.X.X nec-nc++-shared-devel-X.X.X nec-nc++-shared-inst-X nec-nc++-compat (*) nec-nc++-compat-shared nec-nc++-compat-shared-devel nec-nc++-compat-shared-inst	A、B両方で入手可能 (*)はB(SDK-SE)のみ
Fortran compiler	nec-nfort-X.X.X nec-nfort-inst nec-nfort-doc-X.X.X nec-nfort-shared-X.X.X nec-nfort-shared-devel-X.X.X nec-nfort-shared-inst-X nec-nfort-runtime	A、B両方で入手可能
NEC Compat C++ 標準ライブラリ for Vector Engine	nec-nc++-compat nec-nc++-compat-shared nec-nc++-compat-shared-devel nec-nc++-compat-shared-inst	B

注）A：NECの無償yumリポジトリからyumコマンドを使ってインストール

B：NECのアクセス制限付きyumリポジトリからyumコマンドを使ってインストール

PPサポートの契約をしていない場合は、「インターネット配信製品ダウンロードサービス」で入手したパッケージをインストール

パッケージグループに含まれるパッケージ

3.2.4 節に記載したパッケージグループでインストール・アップデート対象となるパッケージファイルは以下の通りです。

表 14 パッケージグループ/InfiniBand for SX-Aurora TSUBASA

パッケージファイル	グループ名
	ve-infiniband
libibverbs-ve1-mofedX.X	○
rdma-core-ve1-mofedX.X	○
libveib-mofedX.X	○
ve-memory-mapping	○
kmod-ve_peermem-mofedX.X	○
libibumad-ve1-mofedX.X	○
sharp-ve1-X.X.X	○
libvesharp-X.X.X	○
libibverbs-ve3-mofedX.X	○
rdma-core-ve3-mofedX.X	○
libibumad-ve3-mofedX.X	○
sharp-ve3-X.X.X	○

表 15 パッケージグループ/VE Application

パッケージファイル	グループ名		
	ve-devel	ve-runtime	ve-frontend
aurlic-lib	○	○	○
aurlic-server	○	○	○
autoconf-ve	○	-	○
automake-ve	○	-	○
coreutils-ve	○	○	-
ftmon	○	○	-
ftmon-ve3	○	○	-
gdb-ve	○	○	○
glibc-ve1	○	○	○
glibc-ve1-devel	○	-	○
glibc-ve3	○	○	○
glibc-ve3-devel	○	-	○
glibc-ve-wrapper	○	○	○
kheaders-ve1	○	-	○
kheaders-ve3	○	-	○
libgcc-ve1-static	○	-	○
libgcc-ve3-static	○	-	○
libsysve-ve1	○	○	○
libsysve-ve1-devel	○	-	○
libsysve-ve3	○	○	○
libsysve-ve3-devel	○	-	○

パッケージファイル	グループ名		
	ve-devel	ve-runtime	ve-frontend
libthread_db-ve1	○	○	○
libthread_db-ve3	○	○	○
libtool-ve	○	-	○
libved	○	○	○
mmm	○	○	-
mmm-analysis	○	○	-
mmm-analysis-ve3	○	○	-
mmm-config	○	○	-
mmm-msl	○	○	-
mmm-msl-ve3	○	○	-
procps-ng-ve	○	○	-
psacct-ve	○	○	-
psmisc-ve	○	○	-
rtmon	○	○	-
rtmon-ve3	○	○	-
strace-ve	○	○	-
sysstat-ve	○	○	-
time-ve	○	○	-
util-linux-ve	○	○	-
ve_drv-kmod	○	○	-
vedebuginfo	○	-	○
ve-firmware	○	○	-
velayout	○	○	○
venumainfo	○	○	-
veoffload-aveo	○	○	-
veoffload-aveo-devel	○	-	○
veoffload-aveorun-ve1	○	○	○
veoffload-aveorun-ve1-devel	○	-	○
veoffload-aveorun-ve3	○	○	○
veoffload-aveorun-ve3-devel	○	-	○
veoffload-veda	○	○	○
veoffload-veda-devel	○	-	○
veoffload-veda-ve1	○	○	○
veoffload-veda-ve3	○	○	○
veos	○	○	-
veos-devel	○	-	○
veos-headers	○	-	○

パッケージファイル	グループ名		
	ve-devel	ve-runtime	ve-frontend
veosinfo	○	○	-
veosinfo3	○	○	-
veos-libveptrace	○	○	○
vesysinit	○	○	-
vesysinit-udev	○	○	-
vmcfw	○	○	-
vmcfw-ve3	○	○	-
vp-kmod	○	○	-
veswap	○	○	-
libvhcall-fortran-ve1	○	○	○
libvhcall-fortran-ve1-devel	○	-	○
libvhcall-fortran-ve3	○	○	○
libvhcall-fortran-ve3-devel	○	-	○
libsmartcols-ve	○	○	-
cmake-ve	○	-	○
veosctl	○	○	-

注）VH上で、VEを使用するプログラムの実行とコンパイルを行う場合はve-devel、実行のみを行う場合はve-runtime、フロントエンドマシンにはve-frontendを使用します。

表 16 パッケージグループ/SDK (MPI 以外)

パッケージファイル	グループ名				
	nec-sdk-runtime	nec-sdk-devel (SDK-CE)	nec-sdk-frontend (SDK-CE)	nec-sdk-devel (SDK-SE/ Compat C++)	nec-sdk-frontend (SDK-SE/ Compat C++)
binutils-ve	○	○	○	○	○
nec-aslfftw-ve-X.X.X	○	○	○	○	○
nec-aslfftw-ve-devel-X.X.X	-	○	○	○	○
nec-asl-ve-X.X.X	○	○	○	○	○
nec-asl-ve-devel-X.X.X	-	○	○	○	○
nec-blas-ve-X.X.X	○	○	○	○	○
nec-blas-ve-devel-X.X.X	-	○	○	○	○
nec-ftraceviewer	○(*2)	○	○	○	○
nec-heterosolver-ve-X.X.X	○	○	○	○	○
nec-heterosolver-ve-devel-X.X.X	-	○	○	○	○
nec-lapack-ve-X.X.X	○	○	○	○	○
nec-lapack-ve-devel-X.X.X	-	○	○	○	○
nec-python36-nlcpy-ve-X.X.X	○	○	-	○	○
nec-python36-numpy-	○	○	-	○	○

パッケージファイル	グループ名				
	nec-sdk-runtime	nec-sdk-devel (SDK-CE)	nec-sdk-frontend (SDK-CE)	nec-sdk-devel (SDK-SE/ Compat C++)	nec-sdk-frontend (SDK-SE/ Compat C++)
X.X.X-Y.Y.Y					
nec-python36-mpi4py-ve-X.X.X	○	○	-	○	-
nec-nc++-X.X.X	-	○	○	○	○
nec-nc++-doc-X.X.X	-	○	○	○	○
nec-nc++-inst	-	○	○	○	○
nec-nc++-shared-X.X.X	○	○	○	○	○
nec-nc++-shared-devel-X.X.X	-	○	○	○	○
nec-nc++-shared-inst-X	○	○	○	○	○
nec-nc++-compat	-	-	-	○	○
nec-nc++-compat-shared	○	○	○	○	○
nec-nc++-compat-shared-devel	-	○	○	○	○
nec-nc++-compat-shared-inst	○	○	○	○	○
nec-nfort-X.X.X	-	○	○	○	○
nec-nfort-doc-X.X.X	-	○	○	○	○
nec-nfort-inst	-	○	○	○	○
nec-nfort-runtime	○	○	○	○	○
nec-nfort-shared-X.X.X	○	○	○	○	○
nec-nfort-shared-devel-X.X.X	-	○	○	○	○
nec-nfort-shared-inst-X	○	○	○	○	○
nec-nlc-base-X.X.X	○	○	○	○	○
nec-nlc-doc-X.X.X	-	○	○	○	○
nec-paralleldebugger	○(*2)	-	-	○	○
nec-sblas-ve-X.X.X	○	○	○	○	○
nec-sblas-ve-devel-X.X.X	-	○	○	○	○
nec-scalapack-ve-X.X.X	○	○	○	○	○
nec-scalapack-ve-devel-X.X.X	-	○	○	○	○
nec-sca-ve-X.X.X	○	○	○	○	○
nec-sca-ve-devel-X.X.X	-	○	○	○	○
nec-veperf-bin	○(*2)	○	○	○	○
nec-veperf-ve1-devel	-	○	○	○	○
nec-veperf-ve3-devel	-	○	○	○	○
nec-veperf-ve1-libs	○	○	○	○	○
nec-veperf-ve3-libs	○	○	○	○	○

(*1) VH 上で、VE を使用するプログラムの実行とコンパイルを行う場合は nec-sdk-devel、実行のみを行う場合は nec-sdk-runtime、フロントエンドマシンには nec-sdk-frontend を使用します。

(*2) リポジトリ "SDK-SE/Compat C++" の nec-sdk-runtime グループにのみ含まれます。

表 17 パッケージグループ/SDK (MPI)

パッケージファイル	グループ名		
	nec-mpi-runtime	nec-mpi-devel	nec-mpi-frontend
nec-mpi-devel-X-X-X	-	○	○
nec-mpi-devel-ve3-X-X-X	-	○	○
nec-mpi-libs-X-X-X	○	○	○
nec-mpi-libs-ve3-X-X-X	○	○	○
nec-mpi-utils-X-X-X	○	○	○
nec-mpi-utils-ve3-X-X-X	○	○	○
nec-mpi-rutnime-X-X-X	○	○	○
nec-mpi-rutnime-ve3-X-X-X	○	○	○
nec-mpi-runtime	○	○	○
nec-mpi-runtime-common	○	○	○
nec-mpi-runtime-pbs	○	○	○

付録 C　ネットワーク構成

SX-Aurora TSUBASAシステムの運用において、複数のVHにより並列計算を行うよう環境を構築する場合は、以下のネットワーク構成が必要になります。

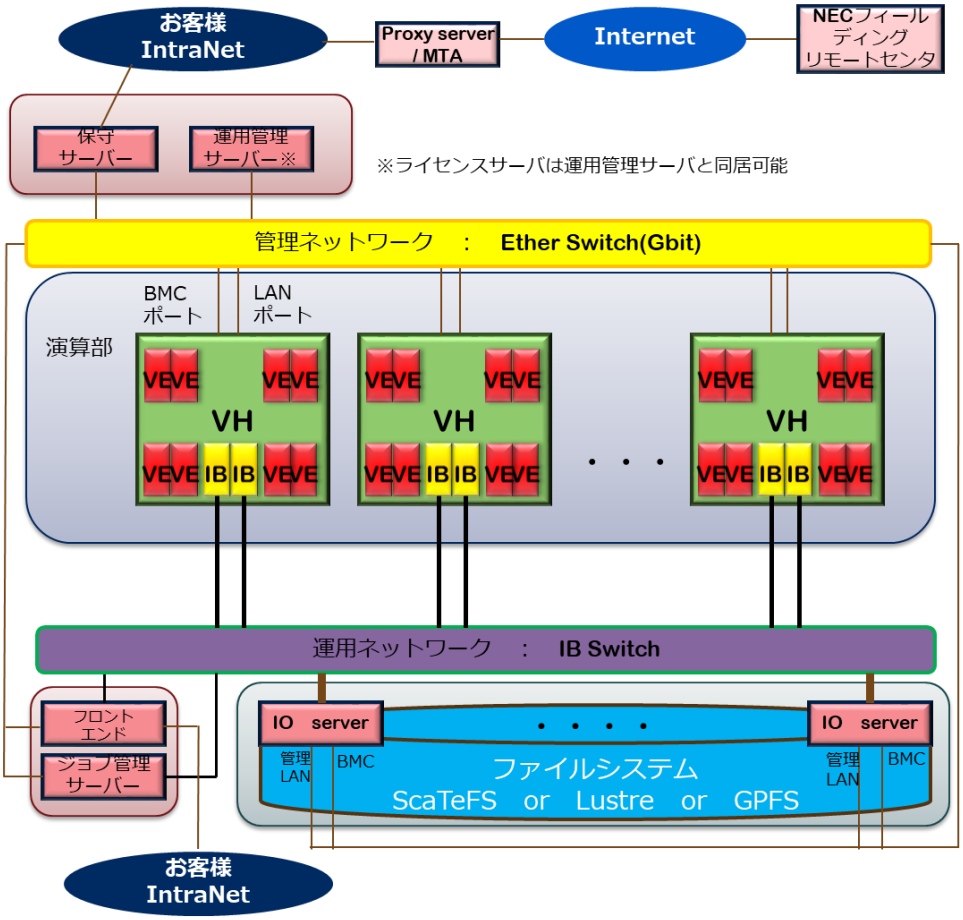


図 10 ネットワーク構成

運用ネットワーク

運用ネットワークは以下の 3 つの機能を有します。

- MPI通信用ネットワーク（InfiniBandを使用する場合）
- IOネットワーク(ファイルシステム用ネットワーク)
- 運用管理ネットワーク(ジョブ制御、転送用。InfiniBandを使用する場合は、IP over IBインタフェースを使用)

運用ネットワークは、InfiniBand プロトコルで接続される広帯域ネットワークです。本ネットワーク

に接続されるサーバーには InfiniBand カード(HCA)の搭載が必要です。各サーバーは InfiniBand スイッチに専用のケーブルで接続され、ネットワークを構築します。物理的な分割はできませんが、QoS 設定により機能ごとに最大帯域を割り当てることができます。本ネットワークは VI クラスターを構築する際に必須となります。なお、シングルシステム(VH 単体)で使用する場合は、運用ネットワークを Ethernet で構成することもできます。

本ネットワークへの接続が想定されるサーバー、機器は以下のとおりです。ただし、これらの機器に限らず、お客様の要求に応じて、柔軟なネットワーク構成をとることができます。

- 演算部(複数台のVIにより構成されるシステムの中核部)
- フロントエンドマシン
- ジョブ管理サーバー
- IOサーバー(ファイルシステムと接続)

管理ネットワーク

管理ネットワークは、以下の 2 つの機能を有します。

- 運用管理ネットワーク(システム運用、演算リソース監視用)
- 保守管理ネットワーク(ハードウェア監視用)

管理ネットワークは、Ethernet で接続されるネットワークです。ギガビットイーサー(GbE)を推奨します。VH は標準で搭載されている LAN ポートより接続可能です。各サーバーは Ethernet スイッチにケーブルで接続され、ネットワークを構築します。また、VH の BMC ポートを管理ネットワークに接続することを強く推奨します。本ガイドの説明は、VH の Ethernet インタフェースの 1 つが、管理ネットワークに接続されていることを前提としています。

本ネットワークへの接続が想定されるサーバー、機器は以下のとおりです。ただし、これらの機器に限らず、お客様の要求に応じて、柔軟なネットワーク構成をとることができます。

- 演算部(複数台のVIにより構成されるシステムの中核部)
- 運用管理サーバー（ライセンスサーバーを含む）

- 保守サーバー
- yumリポジトリサーバー
- フロントエンドマシン
- ジョブ管理サーバー
- IOサーバー、およびストレージの管理インタフェース

付録 D　glibc 環境への移行のお願い

2019 年 1 月リリースの SX-Aurora TSUBASA ソフトウェアより、glibc (the GNU C Library)が VE 上の正式な C ライブラリとなり、旧 C ライブラリである musl-libc のサポートは 2019 年 3 月末で終了しました。

SX-Aurora TSUBASA ソフトウェアをご利用中のお客様は、以下の通り glibc 環境への移行とプログラムの再コンパイルをお願いします。

(1) glibcに対応したSX-Aurora TSUBASAソフトウェアにアップデートしてください。

(2) glibc環境でプログラムを再コンパイルして下さい。

Musl-libc は、今後メンテナンスはされませんので、musl-libc 関連パッケージを以下のコマンドでアンインストールしてください。

```
# yum -y remove musl-libc-ve musl-libc-ve-devel libsysve-musl libsysve-musl-devel veos-musl-headers
```

glibc を用いてコンパイルされたバイナリは、musl-libc を用いてコンパイルされたバイナリと互換性がありません。バイナリにリンクされた C ライブラリを区別したい場合には、以下のように"ve-libc-check"スクリプトを実行して下さい。このスクリプトは、全ての VE バイナリ("a.out", ".o", ".a", ".so")に対して利用できます。

```
$ /opt/nec/ve/bin/ve-libc-check ./a.out
  This is compiled with musl-libc: /home/userxxx/a.out
```

上記のようなメッセージが表示される場合、"a.out"は musl-libc を用いてビルドされています。何もメッセージが表示されない場合は musl-libc を用いていません。

⚠ 注意

- ".s"ファイルから生成されたオブジェクトファイルを"ve-libc-check"で確認することはできません。".s"ファイルを用いている場合には、musl-libcでコンパイルしたバイナリとglibcでコンパイルしたバイナリを混在させないよう、ご注意をお願いいたします。
- glibcでコンパイル・リンクしたVEプログラムが、musl-libcでビルドしたライブラリを動的にリンクあるいはロードする場合には、"ve-libc-check"で確認することはできません。ご利用のライブラリは、glibcを用いて再作成して頂くようお願いいたします。

付録 E　HugePages の手動設定方法

(1) HugePages 数の計算

HugePages 自動設定ツールを利用しない場合、必要な HugePages のページ数（表 18 をもとに計算した値の合計値）を設定してください。

例えば、8 コアの VE 1 枚を使用では、VEOS は 32、NEC MPI は 128 の HugePages を使用します。そのため、必要な HugePages のページ数は、それらの値を合計した 160 となります。

なお、HugePages は 1 ページ当たり 2MB のメモリが割り当てられます。

表 18 必要な HugePages のページ数

Package	The number of huge pages
VEOS (Partial Proces Swapping: VHメモリへのスワッピング)	[スワップアウトするメモリサイズの合計] / [2M] (*1)
VEOS (Partial Process Swapping: ファイルへのスワッピング)	VE枚数 × 32
VEOS(その他)	VEのcore数 × VE枚数 × 4 (*2)
NEC MPI	VEのcore数 × VE枚数 × 16 (*2)
InfiniBand 搭載ノード	+ VHの論理core数 × 8 (*2) (*3)
InfiniBand 非搭載ノード	+ VHの論理core数 × 32 (*2) (*3)
ScaTeFS Client (2VEサーバー)	VEのcore数 × VE枚数 × 32 (*2)
ScaTeFS Client (4VEサーバー)	VEのcore数 × VE枚数 × 64 (*2)

(*1) 例えば、8VEを搭載している装置で、VEあたり最大で40GBのメモリをVHメモリにスワップアウトする場合、163,840(=8×40×1024/2)ページのHugePagesが必要になります。

(*2) Partial Proces Swapping が有効な場合、さらに「×２」してください。

(*3) "NEC MPI"の値に追加で設定してください。

NQSV のソケットスケジューリングを利用する場合は、NEC Network Queuing System V (NQSV) 利用の手引［管理編］18 章ソケットスケジューリング 18.1.3 メモリアロケーションポリシー を必ず参照してください。

HugePages 自動設定ツールは VE アーキテクチャに基づき一定の HugePages 数を確保する場合があります。その場合、表 18 に基づいて算出した HugePages 数がツールによって算出した HugePages 数よりも少なくなる可能性があります。ツールを使わない場合は 表 18 によって算出したページ数を使用してください。

(2) HugePages の設定

この項では、計算した値を HugePages として設定します。

① 現在のHugePagesのページ数の確認

以下のコマンドで、現在のHugePagesのページ数を確認します。もしゼロではない場合、すでに本節の設定を実施済みか /etc/sysctl.conf を確認してください。

SX-Aurora TSUBASAソフトウェア以外のソフトウェア用にHugePagesの設定を行っている場合は、そのソフトウェアが必要とするHugePagesのページ数を(1)の結果に加算してください。

```
# sysctl vm.nr_hugepages
vm.nr_hugepages = 0
```

② 算出したHugePagesのページ数の設定と反映

ファイル/etc/sysctl.confにvm.nr_hugepages行を追加し、(1)で算出したページ数を設定します。すでにvm.nr_hugepages行がある場合、算出したページ数で値を更新します。

```
# vi /etc/sysctl.conf
vm.nr_hugepages = 384
```

以下のコマンドで設定を反映します。

```
# sysctl --system
```

③ 設定したHugePagesのページ数の確認

以下のコマンドで設定が反映されていることを確認します。本例の場合、「vm.nr_hugepages = 384」が出力されれば、設定は完了です。

```
# sysctl vm.nr_hugepages
vm.nr_hugepages = 384
```

④ HugePages設定コマンドの無効化

HugePages設定コマンド が値を設定しないように /etc/opt/nec/ve/veos/ve-hugepages.conf で SKIP_SETTINGを有効にします。

```
# vi /etc/opt/nec/ve/veos/ve-hugepages.conf
SKIP_SETTING=YES
```

⑤　サービスの再起動

各種サービスがHuge Pagesを使用するようにするため、サービスを再起動します。

```
# /opt/nec/ve/sbin/terminate-all-ve-services
# /opt/nec/ve/sbin/start-all-ve-services
```

⑥　VEの状態確認

vecmd コマンドで VE カードが ONLINE であることを確認します。VE が ONLINE になるまでには数分程度かかることがあります。

```
# /opt/nec/ve/bin/vecmd state get
Vector Engine MMM-Command v1.0.0
Command:
state -N 0 get
--------------------------------------------------------------------------------
VE0 [03:00.0] [ ONLINE ] Last Modif:2017/11/29 10:18:00
--------------------------------------------------------------------------------
Result: Success
```

⚠ 注意

"UNINITIALIZED"や"<OFFLINE>"が表示される場合は、しばらくお待ちください。

付録 F　発行履歴

発行履歴一覧表

2018 年	2 月	初版
2023 年	3 月	48 版
2023 年	6 月	49 版
2023 年	9 月	50 版
2023 年	10 月	51 版
2024 年	2 月	52 版
2024 年	10 月	53 版
2025 年	2 月	54 版
2025 年	5 月	55 版
2025 年	7 月	56 版

追加・変更点詳細

52 版

- 商標を更新
- アクセス制限付きyumリポジトリ設定ファイルに関する記述追加
- インストールおよびアップデートにおいて、VE番号定義ファイルの確認・設定を追加

53 版

- RHEL/CentOS 7.9及びRHEL/Rocky Linux 8.6対応保守停止に関連する修正
- RHEL/Rocky Linux 8.10のサポートに関する記述追加
- MLNX_OFED 23.10-3.2.2.0のサポートに関する記述追加
- HCAデバイス名の固定化に関する記述追加
- VMC Firmwareアップデート手順を変更
- HCAファームウェア更新の手順を追加
- ベクトルエンジンのタイプ別のインストールパッケージに関する記述追加

54 版

- モデルB403-8に関する記述追加

55 版

- RHEL/Rocky Linux 8.8対応保守停止に関連する修正

56 版

- 英語版と版数を合わせるための修正(内容変更なし)

索　引

B

binutils 89

C

C/C++ compiler 40, 89

F

Fortran compiler 40, 89

H

HCA iii

I

IB iii

M

MMM 88
MPI iii

N

NEC MPI 40, 45, 88
NLCPy 89
NQSV/Client 63, 78, 79, 89
NQSV/JobServer 89
Numeric Library Collection 89

O

OpenMP iii

T

TSUBASA 用 InfiniBand 88

V

VE iii
VH iii
VI iii
VMC iii
VMC Firmware 88

え

エッジモデル iii

お

オンサイトモデル iii

て

データセンターモデル iii

の

ノード iii

へ

ベクトルアイランド iii
ベクトルエンジン iii
ベクトルホスト iii

ら

ライセンスアクセスライブラリ 88

SX-Aurora TSUBASA システムソフトウェア

SX-Aurora TSUBASA
インストレーションガイド

2025年　　7月 56版

日本電気株式会社

東京都港区芝五丁目７番１号

TEL(03)3454-1111（大代表）

© 2018 NEC Corporation
日本電気株式会社の許可なく複製・改変などを行うことはできません。
本書の内容に関しては将来予告なしに変更することがあります。