NEC Network Queuing System V (NQSV) 利用の手引

[導入編]

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、NEC Network Queuing System V (NQSV) によるジョブ管理システムの導入について説明したものです。

備考

（１）本書は NEC Network Queuing System V (NQSV) R1.00 以降に対応しています。

（２）本書に説明しているすべての機能はプログラムプロダクトであり、以下のプロダクト名およびプロダクト番号に対応しています

プロダクト名	型番
NEC Network Queuing System V (NQSV) /ResourceManager	UWAF00 UWHAF00（サポートパック）
NEC Network Queuing System V (NQSV) /JobServer	UWAG00 UWHAG00（サポートパック）
NEC Network Queuing System V (NQSV) /JobManipulator	UWAH00 UWHAH00（サポートパック）

（３）UNIX は The Open Group の登録商標です。

（４）Intel は、アメリカ合衆国およびその他の国における Intel Corporation の商標です。

（５）OpenStack は、アメリカ合衆国およびその他の国における OpenStack Foundation の商標です。

（６）Red Hat OpenStack Platform は、アメリカ合衆国およびその他の国における Red Hat, Inc. の商標です。

（７）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。

（８）Docker はアメリカ合衆国及びその他の国における Docker, Inc.の商標です。

（９）InfiniBand は、InfiniBand Trade Association の商標またはサービスマークです。

（10）Zabbix はラトビア共和国にある Zabbix LLC の商標です。

（11）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。

本書の対象読者

本書は、システム管理者を対象読者として書かれたものです。
特にシステム管理者がジョブ管理システムの運用設計および構築を行う際には、
本書は有効な説明書となります。
本書は、対象読者がLinuxの一般操作を知っていることを前提として記述されています。

本書の読み進め方

本書は、次の構成となっています。章ごとに対象読者の範囲は異なっており、表の一番右の列にその範囲を示しています。記載された対象読者の後に(*)がついている章については、該当する読者は必ずお読みください。

章	タイトル	内容	対象読者
1	概要	NQSV の概要・構成、製品構成、基本的な用語の説明	システム管理者(*)
2	環境の構築	NQSV の環境を構築するための手順を説明	システム管理者(*)

関連説明書

NEC Network Queuing System V (NQSV)のマニュアルは以下で構成されています。

マニュアル名称	内容
NEC Network Queuing System V (NQSV) 利用の手引［導入編］	システムの全体像および基本的なシステムの構築方法について説明しています。
NEC Network Queuing System V (NQSV) 利用の手引［管理編］	NQSV の管理者が実施する各種設定について説明しています。
NEC Network Queuing System V (NQSV) 利用の手引［操作編］	NQSV の一般利用者が使用する各種機能について説明しています。
NEC Network Queuing System V (NQSV) 利用の手引［リファレンス編］	コマンドリファレンスに関して説明しています。
NEC Network Queuing System V (NQSV) 利用の手引［API 編］	NQSV を操作する C 言語のプログラミングインタフェース（API）について説明しています。
NEC Network Queuing System V (NQSV) 利用の手引［JobManipulator 編］	NQSV のスケジューラコンポーネント JobManipulator に関して説明しています。
NEC Network Queuing System V (NQSV) 利用の手引［アカウンティング・予算管理編］	NQSV のアカウンティング機能に関して説明しています。

表記上の約束

本書では次の表記規則を使用しています。

省略記号 ...　　前述の項目を繰り返すことができることを表しています。ユーザは同様の項目を任意の数だけ入力することができます。

縦棒 |　　オプションまたは必須の選択項目を分割します。

中かっこ {}	１つを選択しなければならない一連パラメータまたはキーワードを表しています。
角かっこ [ ]	省略可能な一連パラメータまたはキーワードを表しています。

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。複数のコアと共用メモリが実装されています。PCI Expressカードであり、x86サーバ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、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)	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。
バッチサーバ（BSV）	バッチサーバホスト上に常駐し、NQSV全体を管理します。
ジョブサーバ（JSV）	実行ホスト上に常駐し、ジョブの実行を管理します。
JobManipulator（JM）	NQSVの提供するスケジューラ機能です。計算リソースの空き容量を管理し、ジョブの開始時刻を割り当てます。
アカウンティングサーバ	アカウント情報の収集・管理、予算管理を行います。
リクエスト	NQSVにおけるユーザジョブの単位であり、１つまたは複数のジョブから構成されたものです。バッチサーバによって管理されます。
ジョブ	ユーザジョブの実行単位であり、ジョブサーバによって管理されます。
論理ホスト	実行ホストの資源を論理的（仮想的）に分割し、ジョブに割り当てたものです
キュー	受け取ったリクエストをプールし、管理する機構です。
BMC	Board Management controllerの略語です。IPMI(Intelligent Platform Management Interface)と呼ばれる業界標準のサーバーマネジメントインタフェースに準拠したサーバー管理を行います。
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側に取り付けるPCIeカードです。
IB	InfiniBandの略語です。
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規格です。
NIC	Network Interface Cardの略語です。他ノードと通信するためのハードウェアです。

目　次

はしがき i
用語定義・略語 vi
目　次 vii
図目次 ix
第 1 章　概要 1
1.1　NQSV とは 1
1.2　NQSV の構成要素 3
1.2.1　ホストとコンポーネント 3
1.2.2　リクエストの投入と実行 7
1.3　製品構成 15
1.3.1　製品パッケージ構成 15
1.3.2　製品パッケージ概要 15
第 2 章　環境の構築 17
2.1　準備 17
2.1.1　クラスタ構成の準備 17
2.1.2　ライセンスサーバの準備 18
2.1.3　NQSV 使用ポート番号の確認 18
2.1.4　yum リポジトリの設定 19
2.2　インストール 23
2.2.1　ライセンスアクセスライブラリ 23
2.2.2　NQSV/ResourceManager 23
2.2.3　NQSV/JobManipulator 25
2.2.4　NQSV/JobServer 25
2.3　設定 25
2.3.1　バッチサーバの設定・起動 25
2.3.2　ユーザ登録 29
2.3.3　クライアント環境設定 30
2.3.4　実行ホストの設定 31
2.3.5　JobManipulator の起動 34
2.3.6　キューの設定 35
2.3.7　アカウンティング・予算管理機能の設定 37
2.3.8　リクエスト実行の確認 41

2.4　アップデート 43
2.5　アンインストール 44
付録 A　発行履歴 45
A.1　発行履歴一覧表 45
A.2　追加・変更点詳細 45
索　引 47

図目次

図 1.2.1-1：小規模ホスト（コンポーネント）構成 3
図 1.2.1-2：大規模ホスト（コンポーネント）構成 4
図 1.2.1-3：アカウント情報の収集・管理 6
図 1.2.2-1：ジョブ投入から実行までの流れ 7
図 1.2.2-2：バッチ／会話リクエスト 11
図 1.2.2-3：パラメトリックリクエスト 11

第1章　　概要

1.1　NQSVとは

NEC Network Queuing System V (NQSV) は、高性能なクラスタ(*1)システムの計算リソースを有効に活用するためのジョブ管理システムです。

NQSV のサポート概要は、以下の通りです。

(1) NQSVは、リソースの最適な割当てを行いジョブの実行を行う機能をサポートします。

(2) ジョブの実行形態としてバッチ処理と会話的利用（会話リクエスト）をサポートします。またバッチ処理においては多重実行機能（パラメトリックリクエスト）もサポートします。

- ジョブの実行形態
  - バッチ
    - 単一実行
    - 多重実行（パラメトリックリクエスト）
  - 会話（会話リクエスト）

(3) NQSVのスケジューラコンポーネント（JobManipulator）は、負荷分散（ロードバランス）やバックフィルによる効率的なリソース管理機能をサポートします。

(4) NQSVはクラスタ単位に、実行ホストの資源を管理し、資源に応じたキューを設定するための機能をサポートします。

(5) 一般利用者／管理者がキューにリクエストを投入・操作するため機能、NQSVの設定や操作、情報表示のための各種コマンドなどをサポートします。

(6) NQSVのアカウントデータの管理、予算の設定、課金レートの設定をサポートします。

(7) NQSVでは、実行ホストを管理し、ホストで発生した障害をバッチサーバへ通知する機能や、NQSVから実行ホストの起動を行うためのエージェントをサポートします。また、電源制御をサポートします。

(8) 独自のクライアントコマンドやスケジューラを作成しNQSVを操作するC言語のプログラミングインタフェース（API）をサポートします。

(9)　その他（オプション）、次の環境を提供します。詳しくは［管理編］を参照してください。

- OpenStack（クラウド環境構築用ソフトウェア群）とDocker（コンテナ型仮想化が可能なソフトウェア）と連携し、実行ホスト内のジョブ実行環境を動的に構成することができます（プロビジョニング環境）。
- バッチサーバ、アカウンティングサーバ、JobManipulatorの各コンポーネントを二重化（冗長化）し、NQSVをダウンさせることなく、継続稼働させることができます（冗長化機能）。

(*1) 1つのバッチサーバを中心に、そこに接続するジョブサーバ群および各種コンポーネントを総称してクラスタと呼びます。

1.2　NQSVの構成要素

1.2.1 ホストとコンポーネント

以下に、ホスト（コンポーネント）構成とホスト構成台数について説明します。

ホスト（コンポーネント）構成

NQSV は、下図のようにバッチサーバを中心に、関連コンポーネントが接続する構成をとります。各コンポーネントの説明は後述します。

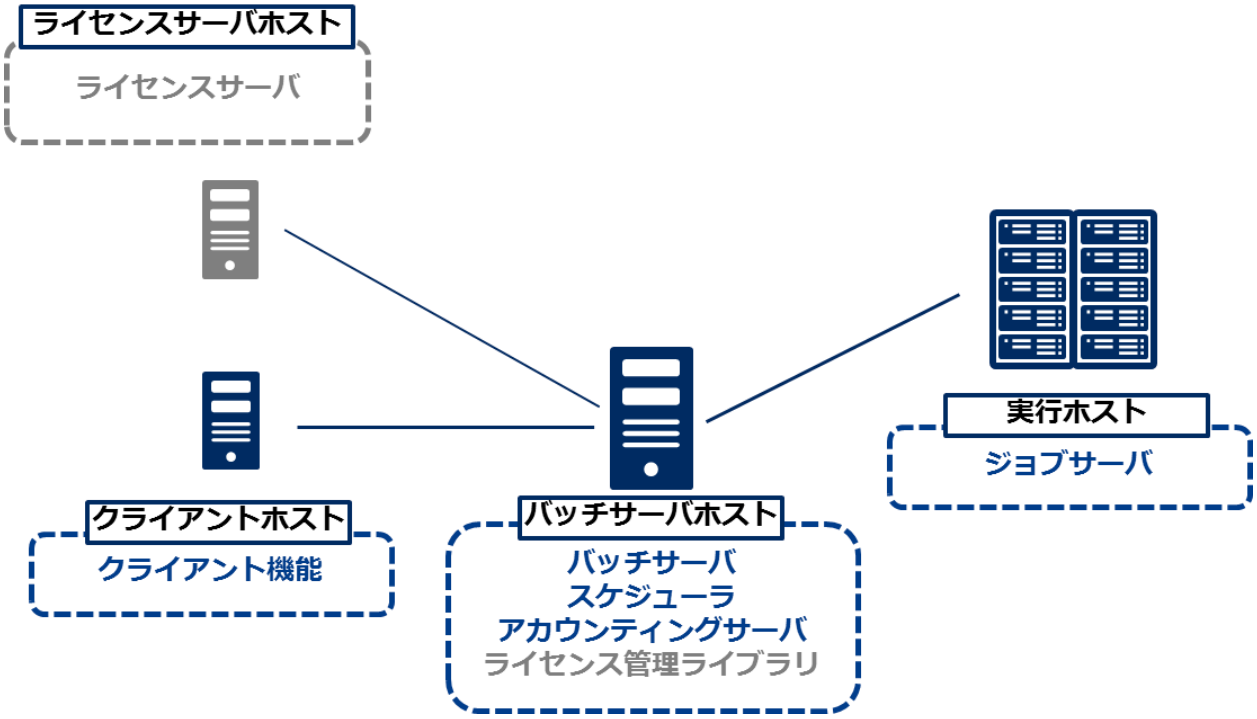

図 1.2.1-1：小規模ホスト（コンポーネント）構成

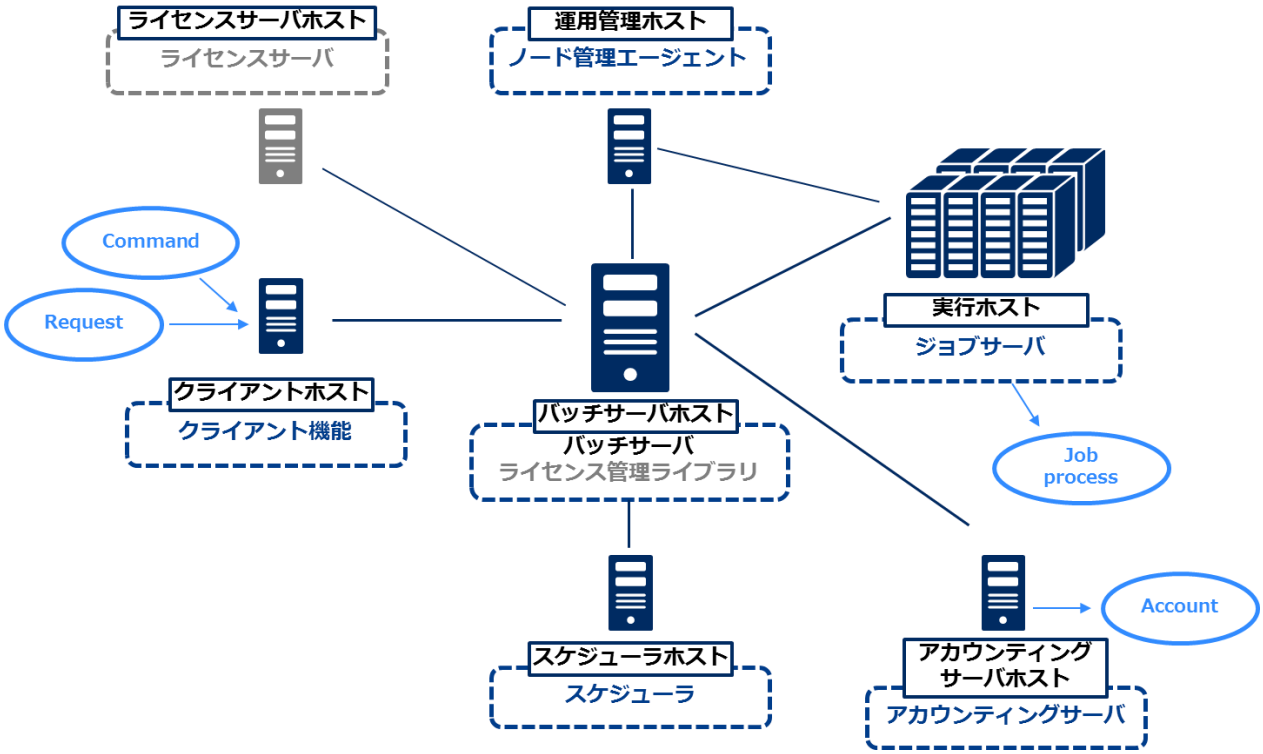

図 1.2.1-2：大規模ホスト（コンポーネント）構成

(※1)ライセンスサーバ：ジョブサーバライセンス、JobManipulator ライセンスの管理サーバ

ホストの種類とその構成台数

NQSV は、以下の種類および、台数のホストを用いて構成します。

バッチサーバホスト

NQSV の中心コンポーネントでありリクエストを管理します。

実行ホスト

ジョブの実行を制御します。

クライアントホスト

リクエストの投入、および、NQSV の管理操作を行います。

スケジューラホスト

スケジューリングの機能を提供します（スケジューラコンポーネントである JobManipulator プロセスを起動します）。

運用管理ホスト

実行ホストのノード障害を検知します。

アカウンティングサーバホスト

アカウンティングサーバはアカウント情報の収集・管理、予算管理を行います。

上記ホストは、NQSV の各機能による論理的な区別です。1 台のマシン上で全てのホストを兼ねることも、また、それぞれ異なったマシン上に各ホストを配置することも可能です。実行ホストと他のホストを異なるマシン上に実現した場合は、実行ホストの計算資源を最大限ジョブに割り当てることができます。

ホスト	台数	備考
バッチサーバホスト	1	バッチサーバは、最大10240ノードの実行ホストを管理可能
実行ホスト	1 - 10240	
クライアントホスト	1 - n	バッチサーバホストに配置することが可能
スケジューラホスト	1 - n	バッチサーバホストに配置することが可能
運用管理ホスト	1 - n	バッチサーバホストに配置することが可能
アカウンティングサーバホスト	1	バッチサーバ、スケジューラ（JobManipulator）と同じホスト上に配置することが可能

構成コンポーネント

NQSV は、以下の種類のコンポーネントで構成されます。

バッチサーバ

バッチサーバ（nqs_bsvd)は、バッチサーバホスト上に常駐し、NQSV 全体を管理します。BSV と呼ぶこともあります。

ジョブサーバ

ジョブサーバ（nqs_shpd）は、実行ホスト上に常駐し、ジョブの実行を管理します。JSV と呼ぶこともあります。

スケジューラ

NQSV ではスケジューラとして、JobManipulator (nqs_jmd)を提供しています。JobManipulator (nqs_jmd)は、通常、バッチホストに常駐します。JM と呼ぶこともあります。

クライアント

リクエストを投入・操作、NQSV の設定や操作、情報表示のための各種コマンド操作などを行います。

アカウンティングサーバ

アカウンティングサーバはアカウント情報の収集・管理を行います。このサーバの存在するマシン上でアカウントデータの参照、予算の設定、課金レートの設定を行います。

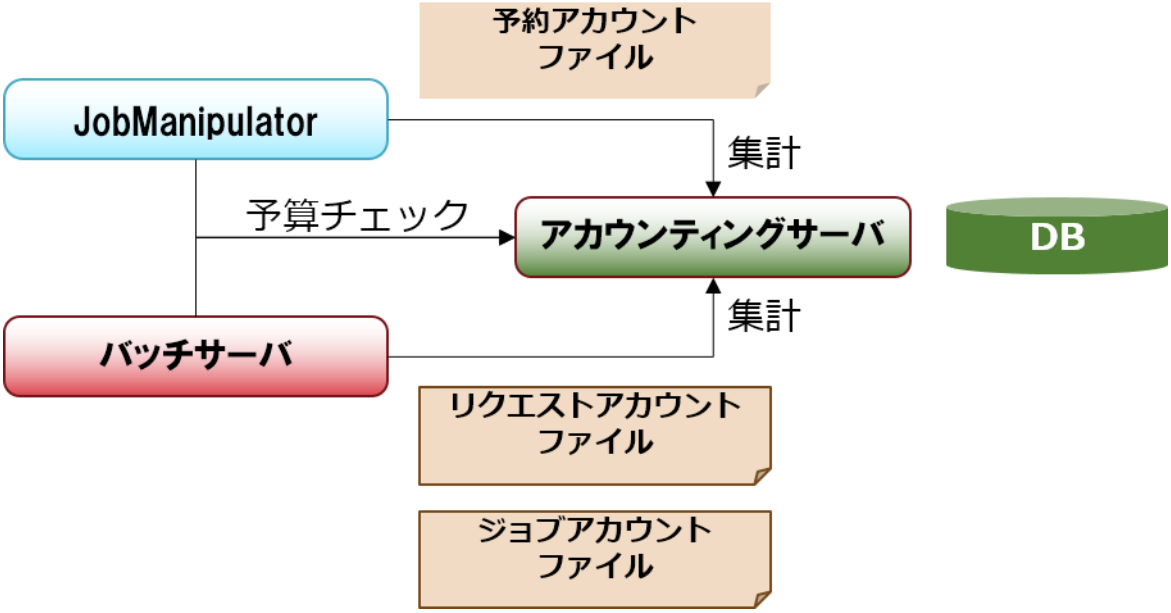

図 1.2.1-3：アカウント情報の収集・管理

運用管理サーバ

障害検知・電源制御を行います。

ノード管理エージェント

ノード管理エージェントは、運用管理ホスト上に常駐し、障害対応の効率化、省電力機能を提供します。

ライセンスサーバ

ライセンスサーバは NQSV のコンポーネントではありませんが、バッチサーバの運用に必要です。各種 SX-Aurora TSUBASA 製品のライセンス情報を一括管理しています。NQSV に関するものでは、ジョブサーバライセンスと JobManipulator ライセンスを管理しています。NQSV で運用するジョブサーバの搭載 CPU 数に対して、以下のライセンスの購入と登録が必要です。

・ジョブサーバライセンス（NEC Network Queuing System V /JobServer）

・JobManipulator ライセンス（NEC Network Queuing System V /JobManipulator）

1.2.2 リクエストの投入と実行

ここでは NQSV を構成する論理的な構成要素を説明します。NQSV の構成の概要は下図の通りです。

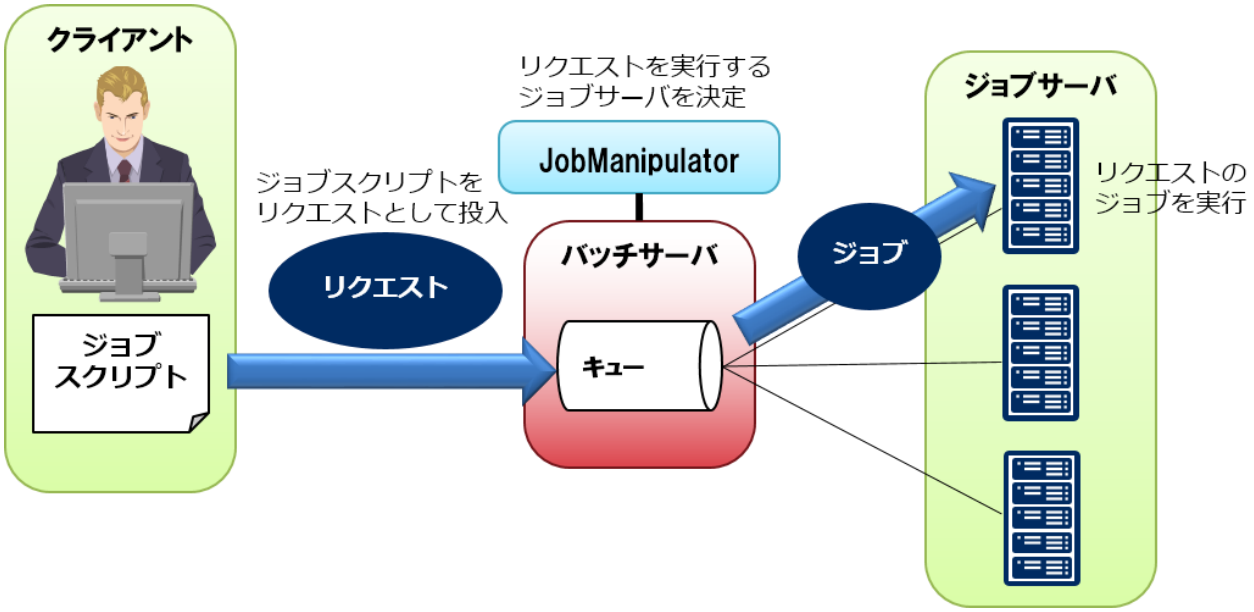

図 1.2.2-1：ジョブ投入から実行までの流れ

キュー

キューとは、受け取ったリクエストをプールし、管理する機構です。リクエストの実行可否を判断し、実行可能であれば順番に実行します。キューには、経過時間やメモリサイズのような資源制限が設定されています。ユーザはリクエストの目的にあったキューを選んで、リクエストを投入します。

リクエスト投入用キュー

リクエストを投入するためのキューは、バッチキュー・会話キュー・転送キューの 3 種類があります。

(1)　バッチキュー

バッチリクエストの実行を制御するキューです。

（[管理編] バッチキュー を参照）

(2)　会話キュー

会話リクエストの実行を制御するキューです。

（[管理編] 会話キュー を参照）

(3)　転送キュー

リクエストの転送を制御するキューです。

（[管理編] 転送キュー を参照）

ファイルステージング用キュー（ネットワークキュー）

ファイルステージングを行うためのキューとして、ネットワークキューがあります。（ファイルステージングの詳細は、[管理編] ファイルステージング を参照）

ネットワークキューとは、キューに投入されたリクエストのステージング対象ファイルをクライアントホストと実行ホスト間で転送するために NQSV が内部的に使用するキューです。

ユーザがこのキューに直接リクエストを投入することはできません。（[管理編] ネットワークキュー を参照）

キューの状態

キューには、投入受付に関する状態と、実行に関する状態があります。

受付については、リクエストの投入を受け付ける ENABLED 状態と、リクエストの投入を受け付けない DISABLED 状態があります。

実行については、リクエストの実行が可能な ACTIVE 状態と、リクエストの実行が不可な INACTIVE 状態があります。

キューの状態は、以下のコマンドで確認することができます。

- qstat(1)コマンド -Q オプション（キューの情報表示）
- qstat(1)コマンド -Qf オプション（キューの詳細情報表示）

（キュー状態の詳細については、[管理編] キューの情報表示、[管理編] キューの状態 を参照）

リクエスト

リクエストは、NQSV におけるユーザジョブの単位であり、バッチサーバによって管理されます。ユーザは、NQSV の機能を利用して、リクエストの投入や管理を行うことができます。

リクエストの種類

ユーザが投入できるリクエストには、以下のような種類があります。

- バッチリクエスト
- パラメトリックリクエスト（バッチリクエストの多重実行形態）
- 会話リクエスト

また、上記の他に、ユーザではなく NQSV が内部的に作成・投入する特殊なリクエストとして、ネットワークリクエストがあります。これはユーザが投入したリクエスト実行時に、必要なファイルをクライアントホストと実行ホスト間で転送するためのもので、バッチサーバが自動的に生成します。

バッチリクエスト

バッチリクエストはプログラムの実行を依頼するリクエストのことです。このリクエストは、NQSV にシェルスクリプトを投入することにより実現されます。また、このスクリプトのコメント部分にバッチリクエスト実行のための設定情報を埋め込むことができ、リクエスト投入時のわずらわしいオプション指定を軽減できます。

リクエストの投入には qsub(1)コマンドを利用します。（バッチリクエストの操作詳細については、［操作編］バッチリクエストの操作 を参照）

また、バッチリクエストの多重実行形態をパラメトリックリクエストと呼びます。パラメトリックリクエストは、一度の投入で、リクエストに指定されたパラメータに従って環境変数や入出力ファイルを変えながら、同一シェルスクリプトを複数回のバッチリクエストとして実行します。（パラメトリックリクエストの操作の詳細については、［操作編］パラメトリックリクエストの操作 を参照）

会話リクエスト

会話リクエストは、会話的に実行ホストを利用するリクエストのことです。会話リクエストには、以下の 2 つのタイプがあります。

- セッション接続タイプ …qlogin(1)コマンドを利用して、rloginのように実行ホストとセッションを接続して会話的に利用します。
- リモート実行タイプ …リモート実行型会話機能（qrsh(1)コマンド）を利用して、rshのようにリモートから実行ホストを利用します。

（リクエスト属性の詳細については、qstat(1)コマンドの-f オプションで表示されます。［操作編］会話リクエストの状態確認 を参照）

リクエストとジョブ

NQSV におけるリクエストとはユーザジョブの管理単位であり、バッチサーバによって管理されます。一方、ジョブとはユーザジョブの実行単位であり、ジョブサーバによって制御されます。（ジョブの詳細については、［操作編］ジョブの操作 を参照してください。）

リクエストはリクエスト ID（リクエスト識別子）によって一意に識別されます。また、ジョブは親リクエストのリクエスト ID とジョブ番号を合わせて表すジョブ ID によって一意に識別されます。

リクエスト ID

バッチサーバ内での通し番号とバッチサーバのホスト名を連結したものです。リクエストの種類にかかわらず、NQSV内で一意なものになります。バッチリクエスト、会話リクエストのリクエストIDは以下の様に表します。

```
<シーケンス番号>.<BSVホスト名>
【例】72.host1
```

パラメトリックリクエストのリクエストIDは以下の様に表します。

- パラメトリックリクエスト

```
<シーケンス番号>[ ].<BSVホスト名>
【例】73[ ].host1
```

- サブリクエスト

```
<シーケンス番号>[<サブリクエスト番号*>].<BSVホスト名>
【例】73[5].host1
```

*サブリクエスト番号：qsub -t で指定した番号範囲を付与

ジョブ ID

ジョブ番号とリクエストIDを合わせたものです。NQSV内で一意なものになります。

```
ジョブ番号：同一リクエストを親に持ったジョブに振られる連番（0以上の整数値）
マスタージョブ：ジョブ番号が0のバッチジョブ
スレーブジョブ：ジョブ番号が1以上のバッチジョブ
```

バッチリクエスト、会話リクエストのジョブは以下の様に表します。

```
<ジョブ番号>:<シーケンス番号>.<BSVホスト名>
【例】0:72.host1
```

パラメトリックリクエストのサブリクエストのジョブは以下の様に表します。

```
<ジョブ番号>:<シーケンス番号>[<サブリクエスト番号>].<BSVホスト名>
【例】0:73[5].host1
```

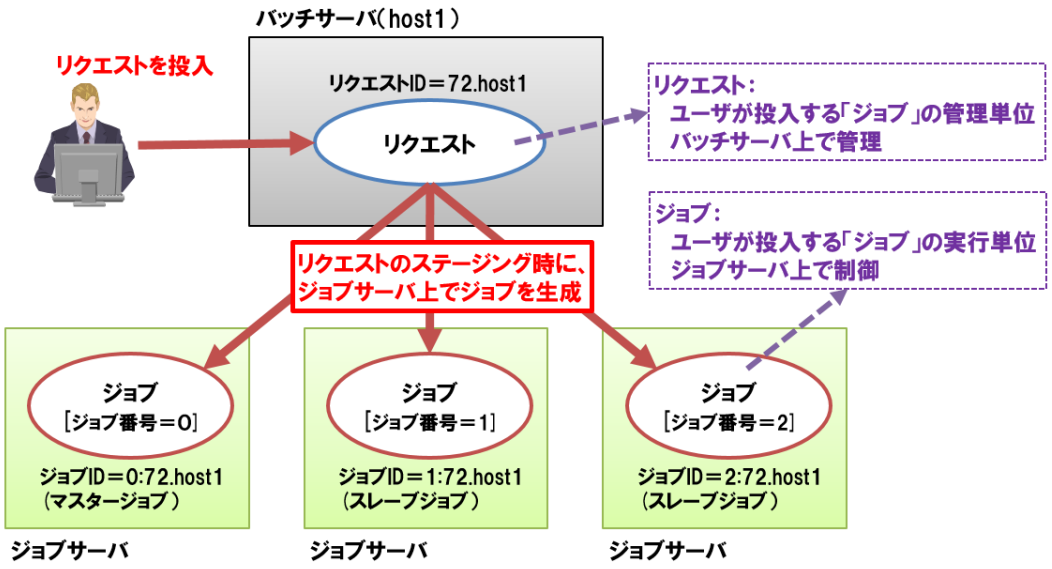

図 1.2.2-2：バッチ／会話リクエスト

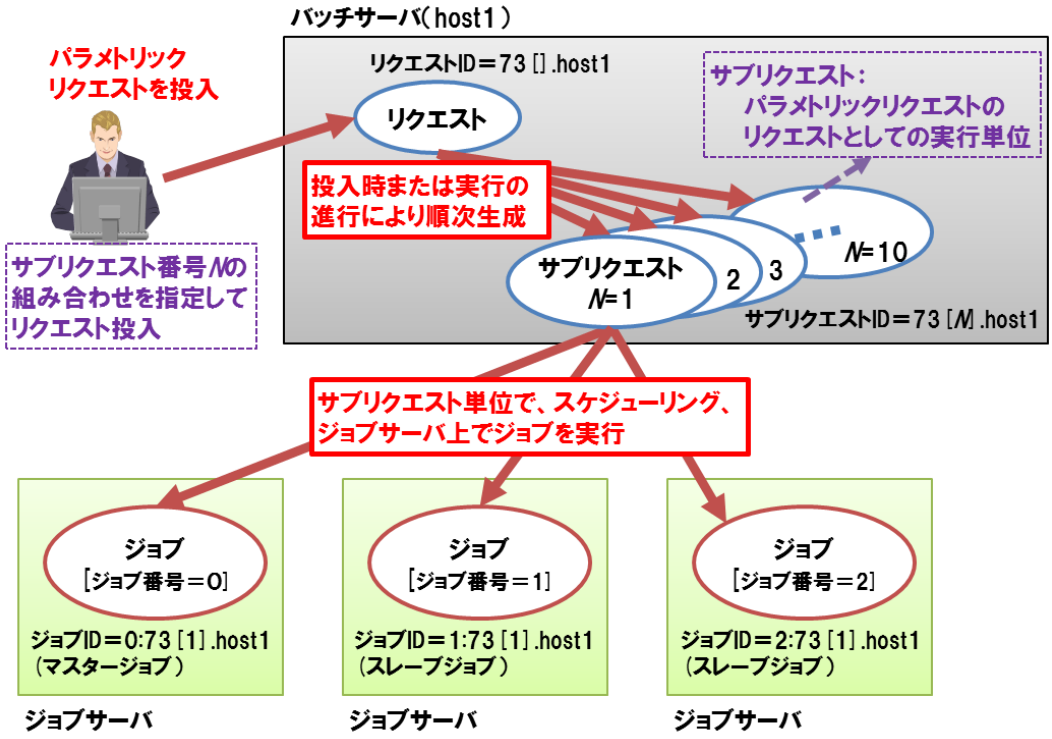

図 1.2.2-3：パラメトリックリクエスト

ジョブと論理ホスト

ジョブは、ジョブサーバ上で実行するプロセスの集合であり、実行ホストにおける実行の単位でもあります。一方、論理ホストは実行ホストの資源を論理的（仮想的）に分割し、ジョブに割り当てたものです。論理ホストはジョブと 1 対 1 で紐づき、ジョブの起動時に生成され、ジョブの実行終了時に消滅します。ジョブと論理ホストは切り離すことのできない関係です。以後の説明で、単に「ジョブ」と記載している場合、ジョブと論理ホストの両方を指します。

※なお、NQSVにおける論理ホストは、ジョブに割り当てる実行ホストの資源を規定するものです。一般的なLinuxの仮想化技術における仮想マシン（Virtual Machine）とは異なるものですので、ご注意ください。

リクエストの資源制限

リクエストにはその属性（パラメータ）として、資源制限値を持ちます。資源制限値は、そのリクエストの動作に必要な資源量を表し、同時に、その資源量以下しか利用できないように NQSV によって制限されます。リクエストの資源制限値は、リクエスト投入コマンド（qsub(1)、qlogin(1)、qrsh(1) コマンド）の-l オプションで指定可能です。（[操作編] 資源制限値の指定、[リファレンス編] ユーザコマンド を参照）

NQSV で利用可能な資源制限は以下のとおりです。

表 1.2-1:資源制限一覧

資源制限値	リクエスト	ジョブ（論理ホスト）	プロセス
経過時間制限値	○	×	×
CPU時間制限値	×	○	○
同時実行CPU台数制限値	×	○	×
同時オープンファイル数制限値	×	×	○
メモリサイズ制限値	×	○	○
データサイズ制限値	×	×	○
スタックサイズ制限値	×	×	○
コアファイルサイズ制限値	×	×	○
ファイルサイズ制限値	×	×	○
仮想メモリサイズ制限値	×	○	○
同時実行GPU台数制限値	×	○	×
VEノード数制限値	×	○	×

リクエストの状態

リクエストは投入されてから消滅するまでの間、さまざまな状態に遷移します。リクエストの状態には以下の種類があります。リクエストの状態遷移の詳細は［API 編］リクエストの状態遷移 を参照してください。

- ARRIVING 状態 (ARI)
  転送キュー経由で転送されてきたリクエストを受信しています。
- TRANSITING 状態 (TRS)
  他のキューへリクエストを転送しています。
- WAITING 状態 (WAT)
  実行時刻に到達するのを待ち合わせています。
- QUEUED 状態 (QUE)
  実行待ちの状態です。スケジュールの対象となります。
- STAGING 状態 (STG)
  リクエストのジョブの生成、およびクライアントホストから実行ホストへのステージング対象ファイルの転送を行っています。
- PRE-RUNNING 状態 (PRR)
  リクエスト内のジョブ実行の前処理を行っています。
  ジョブ実行に必要な情報を各ジョブサーバへ転送し、全てのスレーブジョブを起動した後、最後にマスタージョブを起動します。
  前処理は段階的に行われ、途中でエラーが発生した場合、その時点までの処理を逆順に辿ってキャンセルした後、QUEUEDへ戻ります。
- RUNNING 状態 (RUN)
  リクエスト内のジョブが実行中です。
  MPIジョブの場合は、マスタージョブの終了を検出した時点でPOST-RUNNINGへ遷移します。
  スレーブジョブの終了は状態に影響を与えませんので、仮に全スレーブジョブが終了したとしても、マスタージョブが実行中である限りRUNNING状態が維持されます。
  分散ジョブの場合は、全てのジョブが終了した時点でPOST-RUNNINGへ遷移します。
- POST-RUNNING 状態 (POR)
  リクエスト内のジョブ実行後の後処理を行っています。
- EXITING 状態 (EXT)
  リクエスト内のジョブの標準出力／エラー出力、ステージング対象ファイルを実行ホストからク

ライアントホストへ転送しています。

- HELD 状態 (HLD)
  リクエストはスケジューリング対象から外されています。
  スケジューラからのラン要求またはリスタート要求は受け付けません。
- SUSPENDING 状態 (SUS)
  対象リクエスト内のジョブの停止処理を行い、全ジョブが停止するのを待ち合わせている状態です。
- SUSPENDED 状態 (SUS)
  対象リクエスト内の全てのジョブが停止しています。
- RESUMING 状態 (SUS)
  対象リクエスト内のジョブの実行再開処理を行い、全ジョブが実行を再開するのを待ち合わせている状態です。

リクエストの状態は、qstat(1)コマンドのサマリ表示で STT カラムに表示されます。ただし、SUSPENDING、SUSPENDED、RESUMING の各状態はいずれも“SUS”と表示されます。（[操作編] 基本情報の確認 を参照）

1.3　製品構成

1.3.1 製品パッケージ構成

製品のパッケージ構成、および yum コマンドのグループ名とパッケージファイルの対応は以下の通りです。

製品名	パッケージファイル	グループ名	
		nqsv-batch	nqsv-execution
NEC Network Queuing System V/ ResourceManager	NQSV-ResourceManager	○	-
	NQSV-Client	○	○
	NQSV-API	○	-
NEC Network Queuing System V/ JobServer	NQSV-JobServer	-	○
NEC Network Queuing System V/ JobManipulator	NQSV-JobManipulator	○	-

1.3.2 製品パッケージ概要

(1) NEC Network Queuing System V(NQSV)/ResourceManager

以下の機能を提供します。

- NQSVリクエストの受付、実行管理、リソース管理のバッチサーバ機能
- 障害検知・電源制御のノード管理エージェント機能
- アカウント情報の収集、参照、予算管理のアカウンティングサーバ機能
- ユーザがリクエストを投入・操作するため機能、NQSVの設定や操作、情報表示のための各種コマンドとユーザエージェント機能
- 独自のクライアントコマンドやスケジューラ作成用C言語プログラミングインタフェース（API）

(2) NEC Network Queuing System V(NQSV)/JobServer

ジョブの実行制御、リソース情報収集の機能を提供します。

(3) NEC Network Queuing System V(NQSV)/JobManipulator

JobManipulator は大規模クラスタ運用におけるシングルノードジョブ、マルチノードジョブの混在運用に対応するジョブスケジューラを提供します。

FIFO（fast-in fast-out）を基本とし計算リソース(CPU,メモリ等)の空き容量を管理し、最も早く

実行可能な時刻を割り当てるスケジューリングを実現しています。

[注意]
上記パッケージ以外に、バッチサーバの動作にはライセンスアクセスライブラリのパッケージが必要になります。ライセンスの取得およびライセンスアクセスライブラリに関しては、「HPCソフトウェアライセンス管理説明書」をご参照ください。

第2章　　環境の構築

ここでは、基本的なクラスタ構成での NQSV の構築について説明します。

2.1　準備

2.1.1 クラスタ構成の準備

基本的なクラスタ構成の一例として、以下のように定めます。実際には、構築に合わせて準備してください。

- バッチサーバホスト

ホスト名	IP アドレス	マシン ID
bsv1.example.com	192.168.1.1	10

※図 1.2.1 1：小規模ホスト（コンポーネント）構成のように以下のホストをバッチサーバホストと同じホストに配置。

・スケジューラホスト

・アカウンティングサーバホスト

※運用管理ホストを利用する場合は、バッチサーバホストと同じホストに配置してください。

- 実行ホスト（64ノード）

ホスト名	IP アドレス	ジョブサーバ番号
jsv00.example.com	192.168.1.100	0
jsv01.example.com	192.168.1.101	1
jsv02.example.com	192.168.1.102	2
：	：	：
jsv63.example.com	192.168.1.163	63

※ジョブサーバ番号はバッチサーバ内でユニークとなる 0 から 10239 の番号を割り当てます。

- クライアントホスト

ホスト名	IP アドレス
client1.example.com	192.168.1.10

- ユーザ

NQSV管理者ユーザ	一般ユーザ
root （バッチサーバホスト上のrootユーザを使用）	user1 （クライアントホスト、バッチサーバホスト、実行ホスト上で同一ユーザ名を使用）

- キュー

バッチキュー名	プライオリティ
execque1	10

以降、インストールおよび環境設定は、root 権限で実行してください。qstat(1)コマンドやリクエスト投入は、一般ユーザで実行してください。

2.1.2 ライセンスサーバの準備

まず、構築を行う前に NQSV で運用するジョブサーバの搭載 CPU 数に応じて、ジョブサーバライセンスと、JobManipulator ライセンスを購入の上、ライセンスサーバに登録してください。

構築にあたって、以下のライセンスサーバに関する情報が必要となります。

- ホスト名
- ポート番号

2.1.3 NQSV 使用ポート番号の確認

NQSV は、以下の固定（default）ポート番号を使用しています。構築環境でポート番号が重複する場合は、ポート番号の変更が必要となります。ポート番号の変更方法については［管理編]または［JobManipulator 編]または［アカウンティング・予算管理編]を参照してください。

サーバ	使用（default）ポート番号	備考
バッチサーバ	602	クライアントからの要求受付用ポート番号
クライアント	603	ユーザエージェント
スケジューラ	13000	
アカウンティングサーバ	6542	アカウントデータ受信用ポート番号
	4595	予算要求受付用ポート番号
計算ノード	604	ランチャーデーモン

また、OS のファイアウォール機能が有効になっている場合は、上記のポートを利用できるように各ホスト上で設定が必要です。

2.1.4 yum リポジトリの設定

(1)　オンラインでインターネット上から利用

yum リポジトリが構築済みの場合は、インストール(アップデート)からはじめてください。yum リポジトリの設定のため、以下のコマンドを実行してください。X.X-X は TSUBASA-soft-release パッケージの最新バージョンです。詳細は、「SX-Aurora TSUBASA インストレーションガイド」を参照してください。

```
root# yum install https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-X.X-X.noarch.rpm
```

NQSV の yum リポジトリを有効にするために、リポジトリ設定ファイル（/etc/yum.repos.d/TSUBASA-restricted.repo）を以下のように編集してください。

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

<シリアル No>は、「インターネット配信製品ダウンロードサービス」で取得した「PP・サポートサービス 製品シリアル No.カード」を参照してください。ハイフン(-)を除いて入力してください。

NEC

PP・サポートサービス 製品シリアルNo.カード

品名　：PPSupportPack(NEC Network Queuing System V 8 ライセンス) (1 年間 8H)

型番　：UWHAG00-H111-I

username

シリアル No.　：AAAB-BAAB-BAAA-AAAG

password

(2)　ローカル環境に構築

yum リポジトリが構築済みの場合は、**NQSV 物件の入手**からはじめてください。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。

```
https://sxauroratsubasa.sakura.ne.jp/repos/TSUBASA-soft-release-X.X-X.noarch.rpm
```

TSUBASA-soft-release パッケージのバージョンについては、「SX-Aurora TSUBASA インストレーションガイド」を参照してください。

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。

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

HTTP サーバーが公開するディレクトリ(/var/www/html)配下に、任意のディレクトリを作成してください。この例では、/var/www/html/repos とします。

```
root# mkdir -p /var/www/html/repos
```

リリースパッケージを/var/www/html/repos に置き、以下のコマンドを実行します。

```
root# cd /var/www/html/repos
root# yum install ./TSUBASA-soft-release-X.X-X.noarch.rpm
```

NQSV の yum リポジトリを有効にするために、リポジトリ設定ファイル(/etc/yum.repos.d/TSUBASA-restricted.repo)を編集してください。

```
root# vi /etc/yum.repos.d/TSUBASA-restricted.repo
 [nqsv]
 baseurl=http://address.or.name.of.server/repos/nqsv_elX
 :

 enabled=1 ←0を1に変更
```

yum リポジトリの設定で baseurl にローカルに展開した yum リポジトリの URL(http アドレス)を例のように設定し、enabled に 1 を設定してください。X で表記している部分の OS バージョンについては「SX-Aurora TSUBASA インストレーションガイド」を参照してください。

yum のキャッシュを削除します。

```
root# yum clean all
```

NQSV 物件の入手

NQSV の物件については、SX-Aurora TSUBASA インストレーションガイドの ローカル yum リポジトリの構築、更新 の節を参照して入手してください。なお、有償 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。

以下は、「PP・サポートサービス 製品シリアル No.カード」の例です。ハイフン(-)を除いて入力してください。

NEC

PP・サポートサービス 製品シリアルNo.カード

品名　：PPSupportPack(NEC Network Queuing System V 8 ライセンス) (1 年間 8H)

型番　：UWHAG00-H111-I

username

シリアル No.　：AAAB-BAAB-BAAA-AAAG

password

NQSV の物件を/var/www/html/repos に置き、以下のコマンドを実行します。

```
root# cd /var/www/html/repos
root# unzip nqsv_elX.zip
root# rm *.zip
```

2.2　インストール

NQSV のインストールについて説明します。なお、各パッケージのファイル名については、リリースメモを参照してください。

[注意]

ライセンスサーバが起動し、通信可能な状態で構築作業を行ってください。

構築は、本章に記載の順番でインストールしてください。

2.2.1 ライセンスアクセスライブラリ

バッチサーバの動作にはライセンスアクセスライブラリが必要になるため、ライセンスアクセスライブラリパッケージ（aurlic-lib）を**バッチサーバホスト**にインストールしてください。

ライセンスの取得およびライセンスアクセスライブラリに関しては、「HPC ソフトウェアライセンス管理説明書」をご参照ください。

```
root# yum install aurlic-lib
```

［注意］

ライセンスアクセスライブラリは、バッチサーバがライセンスサーバに接続してライセンスを取得するための各種機能を提供しています。

バッチサーバをインストールする際にライセンスアクセスライブラリ有無のチェックを行うため、バッチサーバをインストールする前にインストールしておく必要があります。

2.2.2 NQSV/ResourceManager

バッチサーバホスト、**アカウンティングサーバホスト**、**運用管理ホスト**において、以下のように NQSV-ResourceManager パッケージをインストールしてください。

```
root# yum install NQSV-ResourceManager
```

以下のパッケージをまとめてバッチサーバホストにインストールする場合は、シェル変数にグループ名を指定してインストールすることも可能です。

- NQSV/ResourceManager(NQSV-ResourceManager, NQSV-Client, NQSV-API)
- NQSV/JobManipulator(NQSV-JobManipulator)

```
root# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nqsv-batch
```

```
root# yum group install nqsv-batch
```

[補足]

物理的には 1 台のマシン上で全てのホストを兼ねることも、また、それぞれ異なったマシン上に各ホストを配置することも可能です。

アカウンティングサーバホスト

リクエスト・ジョブアカウント情報の収集・参照・集計、予算の設定、課金レートの設定等を利用する場合には、アカウンティングサーバの構築が必要です。アカウンティングサーバを構築するアカウンティングサーバホストは、バッチサーバや JobManipulator と同じホストでも別のホストでも構築可能です。NQSV-ResourceManager パッケージをインストールすることでアカウンティング関連のモジュールがインストールされます。

また、NQSV サーバ間のアカウント情報の収集設定や、別ホストのコマンドからアカウンティングサーバにアクセスして情報を表示可能にする設定は、以下の構築を行ってください。

(1) アカウンティングモニタのインストール

バッチサーバおよび JobManipulator を配置するホスト上にはアカウンティングモニタをインストールする必要があります。NQSV-ResourceManager パッケージの中にアカウンティングモニタデーモンとアカウンティングモニタ付属のコマンドが含まれています。

(2) コマンドのインストール

クライアントホストからアカウンティングサーバのアカウンティング情報を参照するためには AUI（Accounting User Interface）の設定が必要です。クライアントホストに NQSV-Client パッケージをインストールしてください。“NEC Network Queuing System V（NQSV）利用の手引［アカウンティング・予算管理編］2.2.2.4 AUI“を参照して設定をしてください。

運用管理ホスト

実行ホストに対する障害対応の効率化、省電力機能を利用する場合は、実行ホストを管理する運用管理ホストの構築が必要です（利用する場合、構築してください）。

実行ホストを管理する運用管理ホスト上に、NQSV-ResourceManager パッケージをインストールしてください。インストール後、ホストで発生した障害のバッチサーバへの通知や、NQSV から実行ホストの起動を行うためのサービスユニット（ノード管理エージェント）を起動してください。

サービスユニット（ノード管理エージェント）の起動については、「管理編」ノード管理エージェントの設定 を参照してください。

クライアントホスト

リクエストの投入、および、NQSV の管理操作を行う**クライアントホスト**上に、NQSV/ResourceManager の中の NQSV-Client パッケージをインストールしてください。

```
root# yum install NQSV-Client
```

2.2.3 NQSV/JobManipulator

バッチサーバホストに NQSV/JobManipulator パッケージをインストールしてください。NQSV/JobManipulator はバッチサーバホスト以外のスケジューラホストにインストールすることも可能です。

```
root# yum install NQSV-JobManipulator
```

2.2.4 NQSV/JobServer

各**実行ホスト**上に、NQSV/JobServer パッケージをインストールしてください。

```
root# yum install NQSV-JobServer
```

以下のパッケージをまとめて実行ホストにインストールする場合は、シェル変数にグループ名を指定してインストールすることも可能です。

- NQSV/ResourceManager(NQSV-Client)
- NQSV/JobServer(NQSV-JobServer)

```
root# /opt/nec/ve/sbin/TSUBASA-groups-remark.sh nqsv-execution
root# yum group install nqsv-execution
```

ジョブサーバをインストールした後、systemctl コマンドを実行して、ランチャーデーモン（nqs_lchd）を起動してください。

```
root# systemctl start nqs-jsv.target
```

※この時、ユニット定義ファイルの内容の編集は必要ありません。

2.3　設定

2.3.1 バッチサーバの設定・起動

(1)　ライセンスの設定

バッチサーバは起動時にライセンスサーバに接続し、使用するライセンス情報を取得します。そのため､接続先ライセンスサーバの情報をあらかじめバッチサーバホスト上に設定する必要があります。

接続先ライセンスサーバのホスト名およびポート番号の情報を、バッチサーバホスト上にある以下のファイルに設定してください。

```
/opt/nec/aur_license/aur_license.conf
```

指定方法は以下の通りです。

```
License_server_host =<ホスト名>
```

ライセンスサーバのホスト名を指定します。

```
License_server_port =<ポート番号>
```

ライセンスサーバの待ち受けポート番号を指定します。

ライセンス取得の詳細については［管理編］2.3 バッチサーバの設定 を参照してください。

(2)　利用するサービスの選択

NQSV/ResourceManager パッケージはバッチサーバに関する様々なサービスユニットが含まれています。これらの中には環境によっては不必要なものもあります。環境に応じて、必要なサービスユニットのみ起動するよう設定してください。

設定のために下記のコマンドを実行します。

```
root# systemctl enable <UnitName>…
```

<UnitName>に指定するユニット名は下記の表を参照してください。

ユニット名	機能の説明
nqs-bsv.service	バッチサーバ
nqs-asv.service	アカウンティングサーバ
nqs-acm.service	アカウンティングモニタ
nqs-nag.service	ノード管理エージェント
nqs-btu.service	冗長化機能の起動デーモン

例えば、バッチサーバを使用する場合は、下記のコマンドを実行してください。

```
root# systemctl enable nqs-bsv.service
```

(3)　マシンIDの登録

バッチサーバを利用可能にするためには、まず、バッチサーバホスト上で、マシン ID の登録を行います。マシン ID とは、NQSV 上でバッチサーバを識別するために使用する 32 ビット長の整数値です。ひとつのバッチサーバに対してマシン ID を一つ割り当てます。（マシン ID の詳細については［管理編］マシン ID の管理 を参照）複数 BSV を運用しない場合、マシン ID は任意の値で問題ありません。

マシン ID の登録は、バッチサーバホスト上で nmapmgr(1M)コマンドを使用します。

```
root# /opt/nec/nqsv/bin/nmapmgr
NMAPMGR>: create
NMAP_SUCCESS:  Successful completion.
NMAPMGR>: add mid 10 bsv1.example.com
NMAP_SUCCESS:  Successful completion.
```

登録した内容は、nmapmgr の show state コマンドによって確認することができます。

```
NMAPMGR>: show state
MachineID  Type   HostName
-------------------------------
10  NQSV  bsv1.example.c
NMAPMGR>: exit
```

(4)　バッチサーバのデータベースの初期化

次に、バッチサーバのデータベースの初期化を行います。データベースの初期化は、バッチサーバホスト上で nqs_bsvd を-i オプション付で起動することで実行します。（バッチサーバデータベースの初期化詳細については ［管理編］バッチサーバデータベースの初期化 を参照）

```
root# /opt/nec/nqsv/sbin/nqs_bsvd -i
BatchServer database was initialized.
```

また、バッチサーバの設定記載用の設定ファイル（/etc/opt/nec/nqsv/nqsd.conf）を必ず作成してください。

```
root# touch /etc/opt/nec/nqsv/nqsd.conf
```

(5)　バッチサーバの起動

データベースの初期化が完了したら、バッチサーバを起動します。

```
root# /opt/nec/nqsv/sbin/nqs_bsvd
NQSV (DEBUG): gmq_attach_qcb: q0000: Attach QCB to queue. (NETWORK name `DefaultNetQue')
NQSV (DEBUG): gma_generate_event: q0000: (QST_CREATE  qid 3.DefaultNetQue)
NQSV (DEBUG): gma_generate_event: q0000: (QSTV_CREATE  qid 3.DefaultNetQue)
NQSV (DEBUG): gmq_attach_qcb: q0001: Attach QCB to queue. (ROUTING name `SysRoutingQue')
NQSV (DEBUG): gma_generate_event: q0001: (QST_CREATE  qid 4.SysRoutingQue)
NQSV (DEBUG): gma_generate_event: q0001: (QSTV_CREATE  qid 4.SysRoutingQue)
NQSV (INFO):  Create request seq# file.
```

バッチサーバが正常に起動した場合、最初に一回のみ上記のメッセージが表示されます。

上記でバッチサーバの起動が確認出来たら、一旦バッチサーバを停止してください。

停止：

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: shutdown
Mgr: exit
```

その後、systemd 経由でバッチサーバを含むサービスを起動してください。systemctl enable で設定したサービスユニットをまとめて起動する場合は下記のコマンドを実行してください。

起動：

```
root# systemctl start nqs-bsv.target
```

詳細は、[管理編] バッチサーバの起動、[管理編] バッチサーバの停止 を参照してください。また、バッチサーバ設定の詳細については ［管理編］ バッチサーバの設定 を参照してください。

2.3.2 ユーザ登録

NQSV 上で NQSV の管理を行うための管理者ユーザ、および、リクエストの投入、ジョブの実行を行うためのユーザをバッチサーバホスト上の以下のファイルに登録します。

ユーザマップファイル（/etc/opt/nec/nqsv/nqs_user.map）

ユーザマップファイルは、ユーザの NQSV へのアクセス権を定義するファイルです。上記、ユーザマップファイルの各行は、以下の設定を行います。

1 列目：NQSV のアクセス権

2 列目：ローカル（バッチサーバホスト）のユーザ名

3 列目：リモート（実行ホスト、クライアントホスト）のユーザの指定

（/etc/opt/nec/nqsv/nqs_user.map の詳細については ［管理編］ユーザの管理 を参照）

例えば、バッチサーバホスト上の root ユーザ、および、クライアント、実行ホスト上で user1 によるリクエストの投入、実行を可能にするために、/etc/opt/nec/nqsv/nqs_user.map ファイルに以下の内容を登録します。

```
PRIV_SCH    root    root:127.0.0.1/32
PRIV_MGR    root    root:192.168.1.1/32
PRIV_NON    root    root:0.0.0.0/0
PRIV_USR    user1   user1:192.168.1.0/24
```

上記の内容は、次の通りです。

2 行目：バッチサーバホスト上の root ユーザが、NQSV の管理者権限でバッチサーバにアクセスするための設定

4 行目：クライアントホストから user1 によって投入されたリクエストを受け付けて、実行ホスト上で user1 によるジョブの実行を行うための設定

2.3.3 クライアント環境設定

クライアント環境の設定について説明します。コマンドと man データは、以下にインストールされております。

- コマンドパス

  OS 共通:

  ユーザコマンド /opt/nec/nqsv/bin

  管理者コマンド /opt/nec/nqsv/sbin

- manパス

  /opt/nec/nqsv/man（English）
  /opt/nec/nqsv/man/ja（Japanese）

（詳細については、［リファレンス編］ユーザコマンド、システム管理コマンド参照）

(1) バッチサーバの管理操作、リクエストの操作、情報参照等の設定

クライアントホスト上で、NQSV の管理操作、リクエストの操作、情報参照等ができるようにするためには、NQSV コマンドが接続するバッチサーバの設定、および、実行結果ファイルの転送のためのデーモン（＝ユーザエージェント）の起動を行う必要があります。

NQSV コマンドが接続するバッチサーバの設定は、クライアントホスト上の /etc/opt/nec/nqsv/api_client.conf ファイルに記述します。

エディタで/etc/opt/nec/nqsv/api_client.conf ファイルを開いて、以下の行を追加してください。

(/etc/opt/nec/nqsv/api_client.conf の詳細については ［管理編］api_client.conf の設定 を参照)

```
batch_server_host   bsv1.example.com
```

また、ユーザエージェントはクライアントホスト上で起動します。その際、以下のように接続するバッチサーバホスト名を引数に指定します。

```
root# /opt/nec/nqsv/sbin/nqs_uagd bsv1.example.com
```

なお、ユーザエージェントの起動/停止は、systemctl stat/stop コマンドで行うこともできます。詳細は、［管理編］ユーザエージェント を参照してください。

(2) JobManipulator の情報参照、管理操作の設定

クライアントホスト上で JobManipulator コマンドを用いて JobManipulator スケジューラの情報表示、および、スケジューラの管理操作をするための設定を行います。

設定はクライアントホスト上の/etc/opt/nec/nqsv/nqs_jmd_cmdapi.conf ファイルで行います。

本ファイルの jm_host に JobManipulator を起動したホストのホスト名を指定してください。

エディタにて/etc/opt/nec/nqsv/nqs_jmd_cmdapi.conf ファイルを開いて、以下の行を追加してください。

```
jm_host bsv.nec.co.jp
```

2.3.4 実行ホストの設定

実行ホストの登録・ジョブサーバ起動

バッチサーバに対して、ジョブを実行するための実行ホストの登録を行います。バッチサーバは、最大 10240 ノードの実行ホストを管理できます。各実行ホスト上では、ジョブの実行制御を行うためのジョブサーバを起動しますが、その際に各ジョブサーバを識別するための番号（ジョブサーバ番号）が必要となります。ジョブサーバ番号はバッチサーバ内でユニークとなる 0 から 10239 の番号を割り当てます。

実行ホストの登録は、qmgr コマンドの attach execution_host サブコマンドにより、実行ホスト名と起動するジョブサーバのジョブサーバ番号を指定して実行します。qmgr の実行は NQSV 管理者権限で行います（前述のユーザ設定例では、バッチサーバホスト上の root ユーザに対して NQSV 管理者権限を設定しています）。

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: attach execution_host host = jsv00.example.com job_server_id = 0
Attach Execution_Host (jsv00.example.com: JSVID = 0).
Mgr: attach execution_host host = jsv01.example.com job_server_id = 1
Attach Execution_Host (jsv01.example.com: JSVID = 1).
Mgr: attach execution_host host = jsv02.example.com job_server_id = 2
Attach Execution_Host (jsv02.example.com: JSVID = 2).
      :
Mgr: attach execution_host host = jsv63.example.com job_server_id = 63
Attach Execution_Host (jsv63.example.com: JSVID = 63).
```

登録した実行ホスト情報は、qstat コマンドの-Et オプション（実行ホストの情報）で確認することができます。

```
$ /opt/nec/nqsv/bin/qstat -Et
ExecutionHost   JSVNO JSV      S OS         Release    Hardware    Load   Cpu STT
--------------- ----- -------- - ---------- ---------- ---------- ----- ----- ---
jsv00.example.c     0 LINKDOWN - --         --         --             -     - INA
jsv01.example.c     1 LINKDOWN - --         --         --             -     - INA
jsv02.example.c     2 LINKDOWN - --         --         --             -     - INA
      :
jsv63.example.c    63 LINKDOWN - --         --         --             -     - INA
```

実行ホストを登録した後、qmgr コマンドの start job_server サブコマンドでジョブサーバを起動します。このとき、qmgr は管理者権限で起動します。

※スケジューラとしてJobManipulatorを利用する場合は、実行ホスト情報を登録するために、全てのジョブサーバを一度起動してください。

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: start job_server all
Start Job_Server all Execution_Host.
```

qmgr コマンドの start job_server サブコマンドでジョブサーバを起動するためには、各実行ホスト上で、ランチャーデーモン(nqs_lchd)が起動されている必要があります。

ジョブサーバの起動は、qstat コマンドの-Et オプション（実行ホストの情報）で確認することができます。

```
$ /opt/nec/nqsv/bin/qstat -Et
ExecutionHost   JSVNO JSV      S OS         Release    Hardware    Load   Cpu STT
--------------- ----- -------- - ---------- ---------- ---------- ----- ----- ---
jsv00.example.c     0 LINKUP   - Linux      4.18.0-147 x86_64     0.0   0.0   ACT
jsv01.example.c     1 LINKUP   - Linux      4.18.0-147 x86_64     0.0   0.0   ACT
jsv02.example.c     2 LINKUP   - Linux      4.18.0-147 x86_64     0.0   0.0   ACT
      :
jsv63.example.c    63 LINKUP   - Linux      4.18.0-147 x86_64     0.0   0.0   ACT
```

なお、ジョブサーバの起動/停止は、systemctl stat/stop コマンドで行うこともできます。詳細は、

［管理編］ジョブサーバの起動、［管理編］ジョブサーバの停止 を参照してください。その他、ジョブサーバ設定の詳細等については ［管理編］実行ホストの管理 を参照してください。

実行ホスト上に openibd がインストールされている環境で、ジョブサーバを systemd によって自動起動する場合は、/usr/local/lib/systemd/system/nqs-jsv.service ファイルに以下の設定を追記してください。

```
[Unit]セクション
After=openibd.service
```

VE および HCA の割当て

SX-Aurora TSUBASA システムにおいては、VE および HCA 割当て機能が利用できます。

VE および HCA の割当て機能では、VE を使用するジョブを、必要な VE 数が搭載されたホスト上で実行する機能を提供します。HCA の正しい割当てを行うため、HCA の構成を記述したデバイスリソース定義ファイル（/etc/opt/nec/nqsv/resource.def）を実行ホスト上に配置・設定してください。詳細については[JobManipulator 編] HCA 割当て機能 を参照してください。また、HCA 障害の検知を行う必要がある場合は[管理編] HCA 障害の検知 を参照してください。

各実行ホストの VE の搭載数は、qstat -Ef コマンドで確認できます。以下は、VE が搭載された実行ホストの qstat -Ef コマンド（実行ホストの情報）での表示イメージです。

```
$ qstat -Ef
Execution Host: host1
    Batch Server = host1.example.com
    Operating System = Linux (Rocky Linux release 8.8 (Green Obsidian))
:
  Vector Engine Information:
    VE Node  0 = Cores:    8 Memory:  48GB Status: ONLINE        OS_Status: ONLINE
    VE Node  1 = Cores:    8 Memory:  48GB Status: ONLINE        OS_Status: ONLINE
:
```

VH の実行ホストがバインドされているキューに、qsub --venode オプション（VE ノード総数）や --venum-lhost（論理ホスト毎の VE ノード数）を指定して投入することで、各ジョブに対して適切な VE ノードが割り当てられ、ジョブを実行することができます。

GPU の割当て

GPGPU 対応機能では、GPU を使用するジョブを、必要な GPU 数が搭載されたホスト上で実行する機能を提供します。本機能は、実行ホスト上に GPU カードが搭載されており、nvidia-smi コマンドがインストールされている場合に、自動的に有効となります。各実行ホストの GPGPU 対応機能が有効であるかどうかは、qstat -Ef コマンド（実行ホストの情報）の出力に GPU 情報が表示されているかどうかで確認できます。

以下は、GPGPU 対応機能が有効である実行ホストの qstat -Ef コマンド（実行ホストの情報）での表示イメージです。

```
$ qstat -Ef
Execution Host: host1
    Batch Server = host1.example.com
    Operating System = Linux
:
    GPU Information:
      Device[0]: GeForce 9800 GT
        TotalGlobalMem = 511 MB
:
```

GPGPU に対応していない実行ホストでは、GPU Information の項目は表示されません。

GPGPU に対応した実行ホストがバインドされているキューに、qsub -l gpunum_job オプションでジョブ毎の同時実行 GPU 台数制限値を設定してリクエストを投入することで、各ジョブに適切な GPU が割り当てられた状態でジョブを実行することができます。

2.3.5 JobManipulator の起動

JobManipulator をインストールしたサーバ上で systemctl コマンドを実行して、JobManipulator（nqs_jmd）を起動してください。なお、スケジューラ ID の既定値は 1 となっています。

```
root# systemctl start nqs-jmd.target
```

初めて JobManipulator を起動したとき、スケジューリング状態は停止状態です。起動後に、smgr(1M)で次のサブコマンドを実施することによりスケジューリングを開始できます。

```
root# smgr -Po
Smgr: start scheduling
Start Scheduling.
```

スケジューラの機能に関しては、［JobManipulator 編］を参照してください。

2.3.6 キューの設定

NQSV 上でリクエストを実行するためには、リクエストを投入するためのキューを作成する必要があります。

(1) バッチキューの作成

バッチキューの作成には、qmgr コマンドの create execution_queue サブコマンドを使用し、作成するキューの名前と、キューのプライオリティを指定して以下のように実行します。（[管理編］バッチキューの生成 を参照）

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: create execution_queue = execque1 pri=10
Queue execque1 created.
```

(2) バッチキューとジョブサーバのバインド

次に、作成したキューに対して、投入されたリクエストを実行するジョブサーバと、実行に際してジョブのスケジューリングを行うスケジューラの設定（バインド）を行います。（[管理編］ジョブサーバ、スケジューラとのバインド を参照）

バッチキューに対するジョブサーバのバインドは、qmgr コマンドの bind execution_queue job_server サブコマンドで、キュー名とバインドするジョブサーバのジョブサーバ番号を指定して行います。

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: bind execution_queue job_server execque1 job_server_id=0
Bound Job_Server_ID (0) to Queue (execque1).
Mgr: bind execution_queue job_server execque1 job_server_id=1
Bound Job_Server_ID (1) to Queue (execque1).
Mgr: bind execution_queue job_server execque1 job_server_id=2
Bound Job_Server_ID (2) to Queue (execque1).
      :
Mgr: bind execution_queue job_server execque1 job_server_id=63
Bound Job_Server_ID (63) to Queue (execque1).
```

バッチキューとジョブサーバのバインドは、qstat コマンドの-Qe オプション（キュー情報）で確認することができます。

```
$ qstat -Qe
```

```
[EXECUTION QUEUE] Batch Server Host: bsv1
==========================================
QueueName    SCH JSVs ENA STS  PRI  TOT ARR WAI QUE PRR RUN POR EXT HLD HOL RST SUS MIG STG CHK
------------ --- ---- ------- ---- ------------------------------------------------------------
Execque1       -   64 DIS INA   10    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
------------ --- ---- ------- ---- ------------------------------------------------------------
 <TOTAL>                            0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
------------ --- ---- ------- ---- ------------------------------------------------------------
```

(3)　バッチキューとスケジューラのバインド

バッチキューにスケジューラをバインドします。バッチキューに対するスケジューラのバインドは、qmgr コマンドの bind execution_queue scheduler サブコマンドによって、キュー名と、バインドするスケジューラのスケジューラ ID を指定して行います。

以下では、スケジューラ ID の既定値が 1 なので、scheduler_id=1 と指定しています。

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: bind execution_queue scheduler execque1 scheduler_id=1
Bound Scheduler_ID (1) to Queue (execque1).
```

バッチキューとスケジューラのバインドは、qstat コマンドの-Qe オプション（キュー情報）で確認することができます。

```
$ qstat -Qe
[EXECUTION QUEUE] Batch Server Host: bsv1
==========================================
QueueName    SCH JSVs ENA STS  PRI  TOT ARR WAI QUE PRR RUN POR EXT HLD HOL RST SUS MIG STG CHK
------------ --- ---- ------- ---- ------------------------------------------------------------
Execque1       1   64 DIS INA   10    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
------------ --- ---- ------- ---- ------------------------------------------------------------
 <TOTAL>                            0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
------------ --- ---- ------- ---- ------------------------------------------------------------
```

(4)　バッチキューの開始

設定したキューを利用可能な状態にするために、投入、および、実行に関するキューの状態を変更します。enable execution_queue サブコマンドにより、指定したキューを投入可能な状態に、start execution_queue サブコマンドによって、指定したキュー内のリクエストが実行開始可能な状態に設

定します。(キュー設定詳細については ［管理編］キューの管理 を参照。)

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: enable execution_queue = execque1
Enable Queue: execque1
Mgr: start execution_queue = execque1
Start Queue: execque1
```

バッチキューの開始は、qstat コマンドの-Qe オプション(キュー情報)で確認することができます。

```
$ qstat -Qe
[EXECUTION QUEUE] Batch Server Host: bsv1
=============================================
QueueName    SCH JSVs ENA STS  PRI  TOT ARR WAI QUE PRR RUN POR EXT HLD HOL RST SUS MIG STG CHK
------------ --- ---- ------- ---- ------------------------------------------------------------
Execque1       1   64 ENA ACT   10    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
------------ --- ---- ------- ---- ------------------------------------------------------------
 <TOTAL>                            0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
------------ --- ---- ------- ---- ------------------------------------------------------------
```

2.3.7 アカウンティング・予算管理機能の設定

NQSV のアカウンティング・予算管理機能を利用する場合は、本節の環境設定を行ってください。詳細は NQSV 利用の手引 [アカウンティング・予算管理編] を参照してください。

アカウンティングサーバの設定

- ユーザマップファイルの配置

アカウンティングサーバでは、NQSV のユーザ権限に応じて各種データの設定や参照ができます。アカウンティングサーバをバッチサーバと異なるホスト上に配置する場合は、バッチサーバホストと同じユーザアカウントが利用できるように設定してください。また、バッチサーバ上のユーザマップファイル"/etc/opt/nec/nqsv/nqs_user.map"をアカウンティングサーバ上にコピーしてください。ユーザマップファイルの形式に関しては、NQSV 利用の手引 [管理編] を参照してください。

- デーモンの自動起動設定

ホスト起動時のアカウンティングサーバデーモンの自動起動設定は以下のとおりです。

```
root# systemctl enable nqs-asv.service
```

- アカウントデータの退避設定

長期間運用を続けると、その間のアカウントデータが蓄積されます。アカウントデータ量が増加すると表示コマンドの動作に時間がかかる場合もあります。そのため、運用に必要なデータを残し、その他のデータを削除、もしくは退避する必要があります。

cron を利用して sceditacct(1M)コマンドを定期的に実行してください。sceditacct(1M)は退避期間を指定できるため、効果的にアカウント DB のデータを削除、退避することが可能です。詳細な設定方法については、NQSV 利用の手引[アカウンティング・予算管理編]の「3.5 アカウントデータの退避」を参照してください。

- 予算管理機能を利用する設定

アカウンティングサーバの設定ファイル（/etc/opt/nec/nqsv/asvd.conf）に次の設定項目を記述してください。予算管理機能を利用しない場合は必要ありません。

```
SBU_CHECK=ON
```

アカウンティングサーバホストにおいて、sbuedit コマンドで課金レートの設定を実施してください。また、budgetedit コマンドで予算の設定を実施してください。

- VEプロセスアカウントを利用する設定

VE プロセスアカウントを利用する場合は、VEOS のプロセスアカウントの出力を有効にしたうえ、NQSV でプロセスアカウントファイルを集計するように設定する必要があります。

VEOS の設定の詳細は、「SX-Aurora TSUBASA インストレーションガイド」の「4.13 Process accounting の設定」を参照してください。

NQSV のプロセスアカウントの設定については、NQSV 利用の手引[アカウンティング・予算管理編]の「3.6 VE プロセスアカウント」を参照してください。

アカウンティングモニタの設定

- デーモンの自動起動設定

ホストの起動時のアカウンティングモニタデーモン自動起動設定は以下のとおりです。

```
root# systemctl enable nqs-acm.service
```

- アカウントファイルのローテート設定

アカウンティングモニタで使用するリクエスト／ジョブ／予約アカウントファイルは、リクエスト

が実行される度にデータが追記されます。そのため、ローテート設定をしていない場合はファイルサイズが増加します。ファイルサイズが増え続けると、アカウンティングサーバにデータを正しく送信できない状況になる場合があります。正常な運用のためにはアカウントファイルの定期的なローテートを実施する必要があります。

はじめに、アカウンティングモニタの設定ファイル（/etc/opt/nec/nqsv/acmd.conf）を運用状況に合わせて適切に設定してください。退避を開始するファイルサイズ(TURN_SIZE)と退避する世代数(TURN_SAVE)を設定してください。既定値は 500MB と 10 世代です。ファイルサイズは必ず 2GB 以下の値に設定してください。

次に、アカウントファイルの定期的なローテート処理を設定してください。cron を使用して scturnacct(1M)コマンドを定期的に起動する設定が必要です。詳細な設定方法については、NQSV 利用の手引[アカウンティング・予算管理編]の「3.5 アカウントデータの退避」を参照してください。

バッチサーバの設定

アカウンティング・予算管理機能を利用するためには、バッチサーバの環境設定も必要です。

- リクエストアカウントとジョブアカウントの設定

NQSV のアカウンティング機能ではリクエストアカウント情報の集計が可能です。リクエストアカウントの集計や予算管理機能を利用するには、リクエストアカウントファイルとジョブアカウントファイルの出力を有効にする設定が必要です。有効にするには qmgr(1M)で次のように設定してください。

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: set batch_server req_account ON
Set Req_Account ON.
Mgr: exit
```

※システムの運用を停止してから設定を変更してください。アカウンティング・予算管理機能を有効にした運用中にリクエストアカウントの出力をオフにしないでください。

- アカウンティングサーバ情報の設定

バッチサーバと JobManipulator がアカウンティングサーバと連携して予算超過をチェックし、リクエスト投入を制限することができます。その機能を有効にするためには、バッチサーバにアカウンティングサーバの情報を設定してください。qmgr(1M)を使用して、アカウンティングサーバのホスト名を設定してください。

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: set batch_server acct_server host = acct-server
Set Acct Server Host.
```

```
Mgr: exit
```

※*acct-server* にアカウンティングサーバのホスト名を指定してください。

- JobManipulatorのリソース予約区間に対する課金の設定

リソース予約区間に対するアカウンティング情報の集計、予算管理機能を使用する場合は、バッチサーバでリソース予約区間のアカウンティング機能を有効にしてください。qmgr(1M)コマンドを使用して設定してください。

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: set batch_server reservation_acct = ON
Set Reservation Accounting
Mgr: exit
```

リソース予約区間に対する課金の設定は、JobManipulator を起動していない状態で実行してください。

- 予算管理機能

NQSV のアカウンティング・予算管理機能は、リクエストの投入時とリソース予約区間の生成時に予算超過をチェックします。予算超過チェックの機能を有効にする場合は、バッチサーバの設定とともにアカウンティングサーバの予算管理機能の設定も有効にする必要があります。

以下のように、qmgr コマンドの"set batch_server nqs_budget_chk"サブコマンドで設定して下さい。

```
root# /opt/nec/nqsv/bin/qmgr -Pm
Mgr: set batch_server nqs_budget_chk = ON
Set NQS Budgetchk
Mgr: exit
```

- 設定内容の確認

アカウンティング・予算管理機能に関するバッチサーバの設定内容は、qstat コマンドの-Bf オプション（バッチサーバの情報）で確認できます。

```
$ /opt/nec/nqsv/bin/qstat -Bf
    Accounting Server Host Name   = localhost
    :
    Budget function               = request,reservation
    Request Accounting            = ON
    :
    Reservation Accounting        = ON
```

```
:
```

デーモンの起動と停止

設定変更後、各デーモンを起動してください。各デーモンの起動、停止方法は以下の通りです。

- アカウンティングサーバの起動・停止

起動：

```
root# systemctl start nqs-asv.service
```

停止：

```
root# systemctl stop nqs-asv.service
```

- アカウンティングモニタの起動・停止

起動：

```
root# systemctl start nqs-acm.service
```

停止：

```
root# systemctl stop nqs-acm.service
```

2.3.8 リクエスト実行の確認

以上の環境設定により、クライアントホストからリクエストの投入ができるようになります。クライアントホスト上の user1 ユーザで qsub(1)コマンドによって以下のようにリクエストを投入します。

```
$ /opt/nec/nqsv/bin/qsub -q execque1 -l elapstim_req=200 --cpunum-lhost=1
uname -a
sleep 100
<Ctrl-d>
Request 0.bsv1.example.com submitted to queue: execque1.
$
```

例では、qsub コマンドの-q オプションで投入先のキュー名を指定し、-l オプションを使用して経過時間制限値 elapstim_req: 200 秒、--cpunum-lhost オプションを使用して同時実行 CPU 台数制限値(論理ホスト) 1）を設定しています。

その後、実行ジョブスクリプトファイルを指定せずリターンキーを押下した場合、qsub コマンドは、

標準入力からのジョブスクリプト入力待ちとなります。

そこで、ジョブとして実行するためのシェルスクリプトイメージを入力し、最後に<Ctrl-d>を押下することにより、リクエストの投入が行われ、標準出力に、投入したリクエストのリクエスト ID が表示されます。

投入したリクエストの状態は、qstat(1)コマンドを使用して確認することができます。

```
$ /opt/nec/nqsv/bin/qstat
RequestID       ReqName  UserName Queue     Pri STT S   Memory      CPU   Elapse R H M Jobs
--------------- -------- -------- -------- ---- --- - -------- -------- -------- - - - ----
0.bsv1.example. STDIN    user1    execque1    0 RUN -    9.34M     0.30       92 Y Y Y    1
$
```

リクエストの実行が終了すると、クライアントホスト上で qsub コマンドを実行した時のカレントディレクトリ上に、結果ファイル（標準出力ファイル、および、標準エラー出力ファイル）が生成されます。結果ファイルのファイル名は、この例の場合、STDIN.o0 および STDIN.e0 となり、STDIN.o0 には、実行ホスト上でジョブとして実行した uname -a の表示結果が格納されています。（リクエスト作成、投入、操作、状態確認等詳細については ［操作編］バッチリクエストの操作 を参照。）

2.4　アップデート

NQSV パッケージのアップデートについて説明します。

yum リポジトリの設定の手順については、2.1.4 yum リポジトリの設定を参照してください。NQSV パッケージをアップデートする際は、アップデート作業の前に、NQSV のすべてのサービスを systemctl stop コマンドで停止してください。設定によって停止方法が異なりますので、サービスの停止に関する詳細は［管理編］ ユニットの管理 を参照してください。

複数の JobManipulator を実行している場合、R1.06 以降のバージョンでは管理方法が変更となります。R1.05 以前のバージョンから R1.06 以降のバージョンへ更新する際、**[JobManipulator 編] 2.5.3 NQSV R1.05 以前から R1.06 以降への更新** を参照してください。

各パッケージのアップデートは、yum コマンドを使用して以下のように行います。NQSV のパッケージファイル名は、1.3.1 製品パッケージ構成 を参照してください。

```
root# yum update <パッケージファイル名>
```

グループ名を指定してインストールした場合は、グループ名を指定してアップデートが可能です。

```
root# yum group update <グループ名>
```

[注意事項]

- 旧バージョンからアップデートする際に、バッチサーバホスト上の設定ファイル（/etc/opt/nec/nqsv/nqsd.conf）がない場合は必ず作成してください。
- 旧バージョンと新バージョンのアカウントデータが混ざるとアカウントが正しく処理できない場合があります。アップデート作業を実施する前に下記の手順でアカウントデータを退避してください。
  1) 実行中ジョブがあれば、ジョブが終わってから 5 分以上待ってから運用を停止します。
  2) アカウンティングモニタの設定ファイル（/etc/opt/nec/nqsv/acmd.conf）の「退避を開始するファイルサイズ(TURN_SIZE)」を一時的に 0 に設定します。
  3) scturnacct(1M)コマンドを実行して古いバージョンのアカウントデータを退避します。
  4) アカウンティングモニタの設定ファイル（/etc/opt/nec/nqsv/acmd.conf）の「退避を開始するファイルサイズ(TURN_SIZE)」の設定をもとに戻します。

2.5　アンインストール

各パッケージのアンインストールは、yum コマンドを使用して以下のように行います。

```
root# yum remove <パッケージファイル名>
```

付録 A　発行履歴

A.1　発行履歴一覧表

2018 年	2 月	初版
2020 年	12 月	第 9 版
2021 年	3 月	第 10 版
2021 年	7 月	第 11 版
2021 年	12 月	第 12 版
2024 年	7 月	第 13 版

A.2　追加・変更点詳細

第 9 版

- 2.3.1 バッチサーバの設定・起動

  設定手順の一部を更新

- 2.4 アップデート

  注意事項を追加

第 10 版

- 2.1.4 yumリポジトリの設定

  パッケージのバージョン表記を修正

- 2.3.1 バッチサーバの設定・起動

  手順を更新

第 11 版

- 2.1.3 NQSV使用ポート番号の確認

  ランチャーデーモンのポート番号を追記

第 12 版

- 2.1.4 SX-Aurora TSUBASAのyumリポジトリを変更

第 13 版

- 2.3.4 実行ホストの設定 のqstatイメージを更新

索　引

B

BMC vi

H

HCA vi

I

IB vi

M

MPI vi

N

NIC vi

V

VE vi
VI クラスター vi

か

会話キュー 7
会話リクエスト 9

き

キュー 7
キューの状態 8

し

冗長化 2
ジョブ ID 10

て

転送キュー 7

に

二重化 2

ね

ネットワークキュー 8

は

バッチキュー 7
バッチリクエスト 9

へ

ベクトルエンジン vi
ベクトルホスト vi

り

リクエスト 9
リクエスト ID 10

NEC Network Queuing System V (NQSV)
利用の手引［導入編］

2024年7月 第13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）

© NEC Corporation 2021
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。