コマンドリファレンス JobCenter

R15.4

■Windows, Windows Server, Microsoft Azure, Microsoft Excel および Internet Explorer は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

■UNIX は、The Open Group が独占的にライセンスしている米国ならびにほかの国における登録商標です。

■HP-UX は、米国 HP Hewlett Packard Group LLC の商標です。

■AIX は、米国 IBM Corporation の商標です。

■Linux は、Linus Torvalds 氏の米国およびその他の国における登録商標または商標です。

■Oracle Linux, Oracle Clusterware および Java は、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。

■Red Hat は、Red Hat,Inc. の米国およびその他の国における登録商標または商標です。

■SUSE は、SUSE LLC の米国およびその他の国における登録商標または商標です。

■NQS は、NASA Ames Research Center のために Sterling Software 社が開発した Network Queuing System です。

■SAP ERP, SAP NetWeaver BW および ABAP は、SAP AG の登録商標または商標です。

■Amazon Web Services は、Amazon.com, Inc. またはその関連会社の米国およびその他の国における商標です。

■iPad および Safari は、米国およびその他の国で登録された Apple Inc. の商標です。

■Docker は、米国およびその他の国で登録された Docker,Inc. の登録商標または商標です。

■その他、本書に記載されているソフトウエア製品およびハードウエア製品の名称は、関係各社の登録商標または商標です。

なお、本書内では、R、TM、cの記号は省略しています。

輸出する際の注意事項

本製品（ソフトウエア）は、外国為替令に定める提供を規制される技術に該当いたしますので、日本国外へ持ち出す際には日本国政府の役務取引許可申請等必要な手続きをお取りください。許可手続き等にあたり特別な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。

はじめに

本書は、JobCenterで用意されているコマンドの利用方法について説明しています。

本書の内容は将来、予告なしに変更する場合があります。あらかじめご了承下さい。

1. マニュアルの読み方

■本バージョンにおける新規機能や変更事項を理解したい場合

→ <リリースメモ>を参照してください。

■JobCenter を新規にインストール、またはバージョンアップされる場合

→ <インストールガイド>を参照してください。

■JobCenter をコンテナ環境で構築、運用をする場合

→ <コンテナガイド>を参照してください。

■JobCenter を初めて利用される場合

→ <クイックスタート編>を参照してください。

■JobCenter の基本的な操作方法を理解したい場合

→ <基本操作ガイド>を参照してください。

■環境の構築や各種機能の設定を理解したい場合

→ <環境構築ガイド>を参照してください。

■JobCenter の操作をコマンドラインから行う場合

→ <コマンドリファレンス>を参照してください。

■JobCenter の運用方法を理解したい場合

→ <運用・構築ガイド>を参照してください。

■運用中のJobCenter を新環境に移行する場合

→ <移行ガイド>を参照してください。

■クラスタ環境で運用中のJobCenter をバージョンアップする場合

→ <クラスタ環境でのバージョンアップ・パッチ適用ガイド>を参照してください。

■その他機能についてお知りになりたい場合

→ 関連マニュアルの内容をお読みいただき、目的のマニュアルを参照してください。

2. コマンドの表記方法

■コマンド中の%InstallDirectory%については、利用するコンポーネントに応じて以下のように読み替えてください。

- JobCenter CL/Win のインストールディレクトリ(デフォルトは、C:\JobCenter\CLXX.YY)
- JobCenter MG/SV のインストールディレクトリ(デフォルトは、C:\JobCenter\SV)
- JobCenter CL/Web のインストールディレクトリ(デフォルトは、C:\jcclweb)
- Definition Helperの解凍後の言語ディレクトリ
- Report Helperの解凍後の言語ディレクトリ
- Analysis Helperの解凍後の言語ディレクトリ

> XX.YYにはバージョン番号が入ります。

■コマンド中の | は、「または」を意味します。

■各オプションは、「-英文字 オプション名($xxx)」となります。

■[ ]付きのオプションは、省略可能です。

■JobCenter CL/Win のコマンドでは、オプションの大文字小文字は区別しません。 たとえば、-Hと-hのどちらを指定してもヘルプが表示されます。

3. 凡例

本書内での凡例を紹介します。

	気をつけて読んでいただきたい内容です。
	本文中の補足説明
	本文中のヒントとなる説明
注	本文中につけた注の説明
__	UNIX版のインストール画面の説明では、__部分(下線部分)はキーボードからの入力を示します。

4. 関連マニュアル

JobCenter に関するマニュアルです。JobCenter メディア内に格納されています。

最新のマニュアルは、JobCenter 製品サイトのダウンロードのページを参照してください。

https://jpn.nec.com/websam/jobcenter/download.html

資料名	概要
JobCenter インストールガイド	JobCenterを新規にインストール、またはバージョンアップする場合の方法について説明しています。
JobCenter クイックスタート編	初めてJobCenterをお使いになる方を対象に、JobCenterの基本的な機能と一通りの操作を説明しています。
JobCenter 基本操作ガイド	JobCenterの基本機能、操作方法について説明しています。
JobCenter 環境構築ガイド	JobCenterを利用するために必要な環境の構築、環境の移行や他製品との連携などの各種設定方法について説明しています。
JobCenter NQS機能利用の手引き	JobCenterの基盤であるNQSの機能をJobCenterから利用する方法について説明しています。
JobCenter 操作・実行ログ機能利用の手引き	JobCenter CL/Winからの操作ログ、ジョブネットワーク実行ログ取得機能および設定方法について説明しています。
JobCenter コマンドリファレンス	GUIと同様にジョブネットワークの投入、実行状況の参照などをコマンドラインから行うために、JobCenterで用意されているコマンドについて説明しています。
JobCenter クラスタ機能利用の手引き	クラスタシステムでJobCenterを操作するための連携方法について説明しています。
JobCenter Helper機能利用の手引き	ExcelをJobCenterの効率的な運用をサポートするJobCenter Definition Helper (定義情報のメンテナンス)、JobCenter Report Helper (帳票作成)、JobCenter Analysis Helper (性能分析)の3つの機能について説明しています。
JobCenter SAP機能利用の手引き	JobCenterをSAPと連携させるための方法について説明しています。
JobCenter WebOTX Batch Server連携機能利用の手引き	JobCenterをWebOTX Batch Serverと連携させるための方法について説明しています。
JobCenter Web機能利用の手引き	Webブラウザ上でジョブ監視を行うことができるJobCenter CL/Webについて説明しています。
JobCenter テキスト定義機能の利用手引き	JobCenterの定義情報をテキストファイルで定義する方法について説明しています。
JobCenter クラスタ環境でのバージョンアップ・パッチ適用ガイド	クラスタ環境で運用しているJobCenterのアップデート、パッチ適用手順を説明しています。
JobCenter 拡張カスタムジョブ部品利用の手引き	拡張カスタムジョブとして提供される各部品の利用方法について説明しています。
JobCenter 運用・構築ガイド	JobCenterの設計、構築、開発、運用について横断的に説明しています。
JobCenter 移行ガイド	運用中のJobCenterを別の新環境に移行する手順について横断的に説明しています。
JobCenter コンテナガイド	JobCenterをコンテナ環境で構築・運用する方法について説明しています。
JobCenter R15.4 リリースメモ	バージョン固有の情報を記載しています。

5. 改版履歴

版数	変更日付	項目	形式	変更内容
1	2020/06/25	新規作成	ー	第1版

目次

はじめに iii
1. マニュアルの読み方 iv
2. コマンドの表記方法 v
3. 凡例 vi
4. 関連マニュアル vii
5. 改版履歴 viii
1. コマンド一覧 1
1.1. サイトが停止状態の場合に一時的なサイト起動を行なうコマンド一覧 5
2. JobCenter CL/Winのコマンド 6
2.1. nscl_open JobCenter CL/Win（GUI画面）の起動と指定したホストへの接続 7
2.1.1. 機能説明 7
2.1.2. オプション 7
2.1.3. 戻り値 8
2.1.4. 注意事項 8
2.2. nscl_term JobCenter CL/Win（GUI画面）の終了 9
2.2.1. 機能説明 9
2.2.2. オプション 9
2.2.3. 戻り値 9
2.3. Nscl_Submit リモートホストのジョブネットワーク投入 10
2.3.1. 機能説明 10
2.3.2. オプション 10
2.3.3. 戻り値 13
2.3.4. 注意事項 14
2.4. Nscl_Refer リモートホストのジョブネットワークの状態を参照 15
2.4.1. 機能説明 15
2.4.2. オプション 15
2.4.3. 戻り値 17
2.4.4. トラッカIDの表示ステータス 17
3. JobCenter MG/SVのコマンド 19
3.1. jnwsummary ジョブネットワーク実行実績を出力 20
3.1.1. 機能説明 20
3.1.2. オプション 20
3.1.3. 戻り値 26
3.1.4. 注意事項 26
3.1.5. 出力形式（CSV形式／可読形式） 26
3.2. jnwschprt ジョブネットワークのカレンダやスケジュール情報を表示 39
3.2.1. 機能説明 39
3.2.2. オプション 39
3.2.3. 戻り値 43
3.2.4. 関連ファイル 43
3.2.5. 注意事項 43
3.2.6. スケジュール情報のCSV形式 44
3.3. jnwop (control) ジョブネットワーク上の各部品を操作 47
3.3.1. 機能説明 47
3.3.2. オプション 48
3.3.3. 戻り値 48
3.3.4. 注意事項 48
3.4. jnwsubmitcmd ジョブネットワークを投入 50
3.4.1. 機能説明 50
3.4.2. オプション 50
3.4.3. 戻り値 53
3.4.4. 注意事項 53
3.5. nqsbg ジョブ内からジョブの終了時に始末されないプロセスを起動 55
3.5.1. 機能説明 55

3.5.2. オプション 55
3.5.3. 戻り値 56
3.5.4. 注意事項 56
3.6. jnwevtmgr イベントの確認と削除 57
3.6.1. 機能説明 57
3.6.2. オプション 58
3.6.3. 戻り値 58
3.6.4. 注意事項 58
3.7. jnwevtcmd イベントを送信 59
3.7.1. 機能説明 59
3.7.2. オプション 59
3.7.3. 戻り値 60
3.7.4. 注意事項 60
3.8. jnwdelete ジョブネットワーク・スケジュールを削除 61
3.8.1. 機能説明 61
3.8.2. オプション 61
3.8.3. 戻り値 62
3.8.4. 注意事項 62
3.9. schctrl スケジュールの有効化/無効化 63
3.9.1. 機能説明 63
3.9.2. オプション 63
3.9.3. サブコマンド 63
3.9.4. 戻り値 66
3.9.5. 注意事項 66
3.10. nqsstart デーモンプロセスを起動 67
3.10.1. 機能説明 67
3.10.2. オプション 67
3.10.3. 戻り値 67
3.10.4. 注意事項 68
3.11. nqsstop デーモンプロセスを停止 69
3.11.1. 機能説明 69
3.11.2. オプション 69
3.11.3. 戻り値 69
3.11.4. 注意事項 69
3.12. nmapmgr ネットワークの構成管理 70
3.12.1. 機能説明 70
3.12.2. 注意事項 71
3.12.3. サブコマンド 71
3.13. qmgr 構成管理および運用管理 75
3.13.1. 機能説明 75
3.13.2. 注意事項 75
3.13.3. 特権 75
3.13.4. ページャ機能（UNIX版） 75
3.13.5. 資源制限 75
3.13.6. サブコマンド 77
3.14. jdh_download 定義情報のダウンロード 115
3.14.1. 機能説明 115
3.14.2. オプション 115
3.14.3. 戻り値 119
3.14.4. 主要メッセージ 119
3.14.5. 注意事項 119
3.15. jdh_upload 定義情報のアップロード 120
3.15.1. 機能説明 120
3.15.2. オプション 120
3.15.3. 戻り値 121
3.15.4. 主要メッセージ 122

3.15.5. 注意事項 122
3.16. jdh_trkget トラッカ情報の取得 123
3.16.1. 機能説明 123
3.16.2. オプション 123
3.16.3. 戻り値 124
3.16.4. 主要メッセージ 124
3.16.5. 注意事項 125
3.17. jpf_stat JPFファイルの情報確認 126
3.17.1. 機能説明 126
3.17.2. オプション 126
3.17.3. 戻り値 126
3.17.4. 実行結果項目(共通) 127
3.18. jc_backup 構成情報のバックアップ 130
3.18.1. 機能説明 130
3.18.2. オプション 130
3.18.3. 戻り値 130
3.18.4. 注意事項 130
3.19. jc_restore 構成情報の復元 131
3.19.1. 機能説明 131
3.19.2. オプション 131
3.19.3. 戻り値 131
3.19.4. 注意事項 131
3.20. jpf_config 構成情報のパラメータを変換 133
3.20.1. 機能説明 133
3.20.2. オプション 133
3.20.3. ルールファイル 133
3.20.4. 戻り値 136
3.20.5. 主要メッセージ 136
3.20.6. 注意事項 137
3.21. jdh_conv JDFファイルとJPFファイルの相互変換 138
3.21.1. 機能説明 138
3.21.2. オプション 138
3.21.3. 戻り値 139
3.21.4. 主要メッセージ一覧 139
3.21.5. 注意事項 140
3.22. sleep Windows版sleepコマンド 141
3.22.1. 機能説明 141
3.22.2. オプション 141
3.22.3. 戻り値 141
3.22.4. 注意事項 141
3.23. spoolconv R12.10.x以前のユーザ定義情報を移行 142
3.23.1. 機能説明 142
3.23.2. オプション 142
3.23.3. 戻り値 142
3.23.4. 注意事項 142
3.24. jc_iedata_conv エクスポートデータからJPFファイルへの変換 143
3.24.1. 機能説明 143
3.24.2. オプション 143
3.24.3. 戻り値 144
3.24.4. 主要メッセージ 144
3.24.5. 注意事項 145
3.25. jc_perf 実行状況の分析 146
3.25.1. 機能概要 146
3.25.2. 機能説明 147
3.25.3. 戻り値 162
3.25.4. 主要メッセージ 162

3.25.5. 注意事項 162
3.26. trfdelete トラッキングファイルの削除 163
3.26.1. 機能説明 163
3.26.2. オプション 163
3.26.3. 戻り値 163
3.26.4. 注意事項 163
3.27. JCSession 単位ジョブから指定したコマンドをActive状態のセッションで実行 164
3.27.1. 機能説明 164
3.27.2. オプション 164
3.27.3. 環境変数 164
3.27.4. 戻り値 165
3.27.5. 使用方法 165
3.27.6. 注意事項 167
3.28. setuserpwd ユーザパスワードの更新 169
3.28.1. 機能説明 169
3.28.2. オプション 169
3.28.3. 出力結果 169
3.28.4. 戻り値 170
3.28.5. 注意事項 170
3.29. paramctrl 暗号化DBへのパラメータ登録 171
3.29.1. 機能説明 171
3.29.2. サブコマンド 171
3.29.3. オプション 171
3.29.4. 戻り値 172
3.29.5. 注意事項 172
3.30. jcres JobCenter MG/SV専用のHTTPデーモン 173
3.30.1. 機能説明 173
3.30.2. オプション 173
3.30.3. 設定ファイル 173
3.30.4. アクセスログ 174
3.30.5. エラーログ 174
3.30.6. API 175
3.30.7. 戻り値 176
3.30.8. 主要メッセージ 176
3.30.9. 注意事項 176
3.31. jnwqctrl 起動トリガ機能が管理するJNW投入予約リストの参照、および削除 178
3.31.1. 機能説明 178
3.31.2. サブコマンド 178
3.31.3. オプション 178
3.31.4. 戻り値 179
3.31.5. 実行結果例 179
3.31.6. 注意事項 181
3.32. jc_usermgr JobCenterのユーザの管理 182
3.32.1. 機能説明 182
3.32.2. サブコマンド 182
3.32.3. オプション 182
3.32.4. 戻り値 183
3.32.5. 主要メッセージ 183
3.32.6. 注意事項 184
3.33. jc_usercfg ユーザ環境の作成、ユーザ環境設定の参照および変更 185
3.33.1. 機能説明 185
3.33.2. サブコマンド 185
3.33.3. オプション 185
3.33.4. 出力結果 186
3.33.5. パラメーター覧 186
3.33.6. 戻り値 197

3.33.7. 主要メッセージ 197
3.33.8. 注意事項 198
3.34. jc_syscfg システム環境設定の参照および変更 199
3.34.1. 機能説明 199
3.34.2. サブコマンド 199
3.34.3. オプション 199
3.34.4. 出力結果 199
3.34.5. パラメーター一覧 200
3.34.6. 戻り値 204
3.34.7. 主要メッセージ 204
3.34.8. 注意事項 205
3.35. jc_permcfg パーミッション設定の参照、および変更 206
3.35.1. 機能説明 206
3.35.2. 戻り値 223
3.35.3. コマンド共通の主要メッセージ 223
3.35.4. 注意事項 223
3.35.5. 制限事項 223
4. クラスタ環境のコマンド 225
4.1. cjcmksite サイトを作成 226
4.1.1. 機能説明 226
4.1.2. オプション 226
4.1.3. 戻り値 226
4.1.4. 注意事項 226
4.2. cjcpw デーモンプロセスの起動と監視、停止 227
4.2.1. 機能説明 227
4.2.2. オプション 228
4.2.3. メンテナンスモード 228
4.2.4. 関連ファイル 228
4.2.5. 戻り値 229
4.2.6. 注意事項 229
4.3. cjcls コマンドを実行したマシン上で稼動しているサイトの一覧を表示 230
4.3.1. 機能説明 230
4.3.2. オプション 231
4.3.3. 戻り値 232
4.3.4. 注意事項 232
4.4. nqspath データベースパス名を表示 233
4.4.1. 機能説明 233
4.4.2. オプション 233
4.4.3. 戻り値 233
4.5. nqsportkpr JobCenter関連サービスのポートを一時的にバインド 234
4.5.1. 機能説明 234
4.5.2. オプション 234
4.5.3. 戻り値 235
4.5.4. 注意事項 235
5. SAP ERPのコマンド 236
5.1. sapclient SAP ERPと連携 237
5.1.1. 機能説明 237
5.1.2. オプション 237
5.1.3. 戻り値 237
5.1.4. 関連ファイル 237
5.2. sapcmd SAP ERP接続パラメータを生成 238
5.2.1. 機能説明 238
5.2.2. オプション 238
5.2.3. 戻り値 238
5.2.4. 関連ファイル 238
6. 設定確認コマンド 239

6.1. jc_check JobCenterの設定環境を確認 240
6.1.1. 機能説明 240
6.1.2. オプション 240
6.1.3. 注意事項 240
6.1.4. 実行結果例（Windows版） 241
6.2. depend_tool JobCenter定義の整合性を確認 243
6.2.1. 機能説明 243
6.2.2. オプション 243
6.2.3. 主要メッセージ 243
6.2.4. 注意事項 245
6.3. jc_ping 指定したサイトに対する通信確認 246
6.3.1. 機能説明 246
6.3.2. オプション 246
6.3.3. 戻り値 247
6.3.4. 注意事項 247
6.3.5. 出力形式（可読形式／JSON形式） 247
6.3.6. 主要メッセージ 250
7. 情報採取コマンド 253
7.1. jc_getinfo JobCenterの障害発生時、原因究明に必要な1次情報を漏れなく採取 254
7.1.1. 機能説明 254
7.1.2. オプション 254
7.1.3. 注意事項 255
7.1.4. 実行結果例（Windows版） 255
7.2. clweb_getinfo CL/Webサーバの障害発生時、原因究明に必要な1次情報を漏れなく採取 257
7.2.1. 機能説明 257
7.2.2. オプション 257
7.2.3. 注意事項 257
8. Analysis Helper機能のコマンド 258
8.1. jdh_trkupdate ローカルのAnalysis Helperのdataフォルダの内容を更新する 259
8.1.1. 機能説明 259
8.1.2. オプション 259
8.1.3. 戻り値 259
8.1.4. 主要メッセージ 259
8.1.5. 実行結果例 259
8.1.6. 注意事項 260

表の一覧

1.1. コマンド一覧 1
1.2. サイトが停止状態の場合に一時的なサイト起動を行なうコマンド一覧 5
2.1. 指定したオプションと画面表示の関係 7
3.1. コマンド終了値と評価結果 25
3.2. 項目形式名 27
3.3. ジョブネットワークの実行実績情報の可読形式フォーマット 37
3.4. 項目形式名 44
3.5. $commandに使用できる文字列 47
3.6. イベントリスト名と内容 57
3.7. イベント一覧出力形式 57
3.8. イベント削除実行後のメッセージ表示 58
3.9. jnwevtcmdコマンドを実行したときのメッセージ表示 59
3.10. jnwdeleteコマンドを実行したときのメッセージ表示 61
3.11. nmapmgrコマンドのデータベース構成要素 70
3.12. 時間制限の指定例 76
3.13. サイズ制限の指定例 76
3.14. Windows版jc_iedata_convコマンドの変換対応表 145
6.1. 可読形式の出力形式 247
6.2. JSON形式のパラメータ名 249

1. コマンド一覧

JobCenterに用意されているコマンドは表1.1「コマンド一覧」のとおりです。

各コマンドとJobCenter CL/Win、JobCenter SV/MGにおける対応について、各製品およびOSごとに「○（対応している）」「×（対応していない）」で示します。

クラスタサイトに対して実行する前に環境変数をあらかじめ設定する必要があるコマンドについては、「NQS_SITE事前設定」の欄に「S」(NQS_SITE事前設定が必要)、「DB」(NQS_SITEDB事前設定が必要)で示してあります。

逆に「！」で示すコマンドは、NQS_SITEやNQS_SITEDBが事前に設定されていると正常に動作しません。

また「－」で示すコマンドはNQS_SITE等の事前の設定が不要、もしくは影響なしとなります。

詳細は注意事項を参照してください。

表1.1 コマンド一覧

コマンド名	機　能	CL/Win対応	MG/SV対応		NQS_SITE事前設定	章番号
			Windows	UNIX		
nscl_open	JobCenter　CL/Win（GUI画面）の起動と指定したホストへの接続を行う。	○	×	×	－	2.1
nscl_term	JobCenter　CL/Win（GUI画面）の終了を行う。	○	×	×	－	2.2
Nscl_Submit	ジョブネットワークの投入を行う。	○	×	×	－	2.3
Nscl_Refer	トラッカ情報の参照を行う。	○	×	×	－	2.4
jnwsummary	ジョブネットワークの実行状況を表示する。	×	○	○	S,DB	3.1
jnwschprt	ジョブネットワークのカレンダ、スケジュール情報を表示する。	×	○	○	S,DB	3.2
jnwop (control)	ジョブネットワーク上の各部品の操作を行う。	×	○	○	S,DB	3.3
jnwsubmitcmd	ユーザが定義したジョブネットワークを投入する。	×	○	○	S,DB	3.4
nqsbg	単位ジョブから、単位ジョブ終了時にシグナルを受信しないバックグラウンドプロセスとしてコマンドを起動する。	×	×	○	－	3.5
jnwevtmgr	イベントの確認・削除を行う。	×	○	○	S,DB	3.6
jnwevtcmd	イベントの送信を行う。	×	○	○	S,DB	3.7
jnwdelete	ジョブネットワーク・スケジュールの削除を行う	×	○	○	S	3.8
schctrl	スケジュールの有効化/無効化	×	○	○	S,DB	3.9
nqsstart	JobCenterのデーモンプロセスを起動する。	×	×	○	－	3.10
nqsstop	JobCenterのデーモンプロセスを停止する。	×	×	○	－	3.11
nmapmgr	JobCenterのネットワークの構成を管理する。	×	○	○	S	3.12

コマンド名	機　能	CL/Win対応	MG/SV対応		NQS_SITE事前設定	章番号
			Windows	UNIX		
qmgr	JobCenterの構成および運用を管理する。	×	○	○	S	3.13
jdh_download	定義情報をダウンロードする。	×	○	○	−	3.14
jdh_upload	定義情報をアップロードする。	×	○	○	−	3.15
jdh_trkget	トラッカ情報を取得する	×	○	○	S,DB	3.16
jpf_stat	JPFファイルの情報を確認する。	×	○	○	−	3.17
jc_backup	構成情報をバックアップする	×	○	○	−	3.18
jc_restore	構成情報を復元する	×	○	○	−	3.19
jpf_config	バックアップした構成情報を基にホスト名やユーザ名等のパラメータを変換した構成情報を作成する	×	○	○	−	3.20
jdh_conv	JDFファイルとJPFファイルの相互変換	×	○	○	−	3.21
sleep	Windows版sleepコマンド	×	○	×	−	3.22
spoolconv	R13.1以前のユーザ定義情報を移行する	×	×	○	S	3.23
jc_iedata_conv	エクスポートデータをJPFファイルに変換する	×	○	○	−	3.24
jc_perf	JobCenterの実行状況について、データを収集し解析する。	×	○	○	S	3.25
trfdelete	トラッキングファイルを削除する。	×	○	○	S,DB	3.26
JCSession	単位ジョブから指定したコマンドをActive状態のセッションで実行する。	×	○	×	−	3.27
setuserpwd	ユーザのパスワードを更新する。	×	○	×	−	3.28
paramctrl	暗号化DBへパラメータを登録する。	×	○	○	S	3.29
jcres	JobCenter　MG/SV専用のHTTPデーモン	×	○	○	−	3.30
jnwqctrl	起動トリガ機能が管理するJNW投入予約リストの参照、および削除を行う。	×	○	○	S	3.31
jc_usermgr	JobCenterのユーザの管理を行う。	×	○	×	−	3.32
jc_usercfg	ユーザ環境の作成、ユーザ環境設定の参照および変更を行う。	×	○	○	S	3.33
jc_syscfg	システム環境設定の参照、および変更を行う。	×	○	○	S	3.34
jc_permcfg	パーミッション設定の参照、および変更を行う。	×	○	○	S	3.35
cjcmksite	クラスタサイトを作成する。	×	○	○	(−)	4.1
cjcpw	ローカルサイト、またはクラスタサイトを起動する。JobCenterの	×	○	○	(−)	4.2

コマンド名	機　能	CL/Win対応	MG/SV対応		NQS_SITE事前設定	章番号
			Windows	UNIX		
	デーモンプロセスが動作していることを監視する。					
cjcls	クラスタサイトの一覧を表示する。	×	○	○	(－)	4.3
nqspath	サイトが使用しているデータベースパス名を表示する。	×	×	○	－	4.4
nqsportkpr	nqsポートを一時的にバインドする。	×	×	○	－	4.5
sapclient	SAP　ERPへのログインを確認する。 auditlevelの設定を行う。	×	○	○	S	5.1
sapcmd	接続パラメータファイルのサンプルファイルを作成する。 パラメータファイルに記述する暗号化したSAPユーザのパスワードを生成する。	×	○	○	S	5.2
jc_check	現在の環境に問題がないか検査する。	×	○	○	!	6.1
depend_tool	定義情報の整合性を確認する。	×	○	○	S	6.2
jc_ping	指定したサイトに対する通信確認	×	○	○	S,DB	6.3
jc_getinfo	障害発生時、原因究明に必要な1次情報を漏れなく採取する。	×	○	○	!	7.1
clweb_getinfo	CL/Webサーバの障害発生時、原因究明に必要な1次情報を漏れなく採取する。	×	×	×	－	7.2
jdh_trkupdate	ローカルのAnalysis　Helperのdataフォルダの内容を更新する。	×	×	×	－	8.1

■上記の一覧表のうち「S」で示すコマンドについては、クラスタサイトを対象にコマンドを実行する直前に環境変数NQS_SITEにサイト名を設定する必要があります。またWindows版JobCenterの場合、「DB」で示すコマンドについては環境変数NQS_SITEDBにサイトのデータベースパスを追加で設定する必要があります。

■「！」で示すコマンドについては、コマンド実行時にNQS_SITEやNQS_SITEDBが事前に設定されていると正常に動作しません。必ず実行前に解除されていることを確認してください。

■NQS_SITEには、cjcmksiteでクラスタサイトを作成する際に指定した$site-nameの値を設定します。

■NQS_SITEDBには、cjcmksiteでクラスタサイトを作成する際に指定した$db_directoryの値をフルパスで設定します。(Windows版のみ)

■UNIX版の場合、思わぬ誤動作を引き起こす原因になりますのでNQS_SITEをユーザの固有の環境変数として.loginや.cshrcに固定で設定しないでください。必ずコマンド実行操作の直前にその都度設定するようにしてください。

■Windows版の場合、思わぬ誤動作を引き起こす原因になりますのでNQS_SITEやNQS_SITEDBをシステム環境変数として固定で設定せず、必ずコマンド実行の直前に設定してください。

■Windows版の各コマンドパスやユーザデータ格納パスは、R12.7以前とは異なりますので注意して下さい。

1.1. サイトが停止状態の場合に一時的なサイト起動を行なうコマンド一覧

以下のコマンドは、対象となるサイトが起動していない状態で実行された場合、DBプロセスの一時的な起動を行います。

daemon.conf に local_daemon=OFF を設定している場合、これらのコマンドをローカルサイトに対して実行すると DBプロセスの一時起動が抑止されるためコマンド実行はエラーとなります。 ローカルサイトに対して実行する前に、local_daemon=SITE, または local_daemon=COMPAT に設定を変更してください。

local_daemon パラメータの詳細については <NQS機能利用の手引き>の「7.3.4 パラメータ」 を参照してください。

表1.2 サイトが停止状態の場合に一時的なサイト起動を行なうコマンド一覧

コマンド名	サイト指定方法	章番号
jdh_trkget	環境変数 NQS_SITE	3.16
jc_backup	-c オプション	3.18
jc_restore	-c オプション	3.19
spoolconv	環境変数 NQS_SITE	3.22
trfdelete	環境変数 NQS_SITE	3.25
jc_check	環境変数 NQS_SITE	6.1

2. JobCenter CL/Winのコマンド

本章ではJobCenter CL/Winがインストールされたマシンにおいて利用可能なコマンドを説明します。

2.1. nscl_open JobCenter CL/Win（GUI画面）の起動と指定したホストへの接続

```
%InstallDirectory%\nscl_open [-server $server] [-user $user [-password $password | -pw $passwordfile | -epw $EncryptPasswordfile]] [-mode { edit | view | ref }] [-d] [-h]
```

2.1.1. 機能説明

■接続確認画面にオプションで指定した項目を設定した状態でJobCenter CL/Win（GUI画面）を起動します。

■本コマンドでは、コマンド引数としてパスワードを直接入力する(-passwordオプション)か、パスワードを記述したファイルを利用する(-pwオプション)か、パスワードファイル作成用コマンドで作成したファイルを利用する(-epwオプション)必要があります。

セキュリティ上、コマンド引数にパスワードを直接入力したり、パスワードを記述したファイルを作成できない場合は、パスワードファイル作成用コマンドを利用してください。パスワードファイル作成用コマンドの詳細は「2.3 Nscl_Submit リモートホストのジョブネットワーク投入」を参照してください。

表2.1 指定したオプションと画面表示の関係

条　件	表　示
-server,-user,-password（-pw,-epw）オプションをすべて指定し、-dオプションを指定しない場合	接続確認画面は表示されずに接続後の画面が表示されます。
-server,-user,-password（-pw,-epw）オプションをすべて指定しない場合	接続先サーバ名、ユーザ名、パスワードをブランクにした接続確認画面が表示されます。
-serverまたはuserオプションのどちらかを指定しない場合	接続先サーバ名またはユーザ名をブランクにした接続確認画面が表示されます。

2.1.2. オプション

-server $server

接続先マシン名(サイト名)を指定します。

-user $user

接続先ユーザ名を指定します。

-password $password

本オプションは -user が指定された場合にのみ有効です。

接続対象ユーザのパスワードに、$passwordで指定したパスワードを指定します。

-pw $passwordfile

本オプションは -user が指定された場合にのみ有効です。

パスワードファイルをフルパスで指定します。

パスワードファイルにユーザのパスワードをそのまま（暗号化せずに）書き込みます。

-epw $EncryptPasswordfile

本オプションは -user が指定された場合にのみ有効です。

パスワードファイル作成コマンド（Nscl_Submitの-mオプション指定）を用いて作成したファイルをフルパスで指定します。詳細は「2.3 Nscl_Submit リモートホストのジョブネットワーク投入」を参照してください。

-mode {edit|view|ref}

接続モードを指定します。

■-mode edit

(通常モード) ジョブネットワークの作成、削除、変更、およびジョブネットワークやジョブの制御が可能です。-modeオプションを省略した場合の既定値です。

■-mode view

(参照モード) ジョブネットワークの作成、削除、変更等はできません。ジョブネットワークやジョブの制御は可能です。

■-mode ref

(Refモード) ジョブネットワークの作成、削除、変更等に加え、ジョブネットワークやジョブの制御も不可となります。

-d

接続確認のダイアログを表示します。

-server,-user,-password(-pw,-epw)のオプションをすべて指定した場合に、本オプションを指定することで接続確認のダイアログ表示を行います。

-h

ヘルプを表示します。他のオプションと併用した場合、他のオプションは無視されます。

2.1.3. 戻り値

戻り値	内　容
0	正常終了です。
1	オプション指定が不正です。
2	-serverオプションにサーバ名が指定されていません。
3	-userオプションにユーザ名が指定されていません。
4	-passwordオプションにパスワードが指定されていません。 -pw (-epw)オプションで指定したパスワードファイルがないか、読み込みエラーが発生しています。
5	-password (-pw、-epw)オプションが -userオプションなしで指定されています。
6	-modeオプションに接続モードが指定されていません。
7	接続モード指定が不正です。

2.1.4. 注意事項

■-server、-user、-password (-pw、-epw)オプションがすべて指定されている場合、引数として指定されているサーバ名、ユーザ名、パスワードの正当性は実際に接続動作を行うまで確認できません。

したがって、引数のいずれかが不正であってもCL/Win(GUI)画面が起動し、CL/Winの画面上で接続エラーとなります。

2.2. nscl_term JobCenter CL/Win（GUI画面）の終了

```
%InstallDirectory%\nscl_term [-server $server [-user $user]] [-h]
```

2.2.1. 機能説明

■nscl_termコマンドを発行したマシンまたはオプションで指定したマシンに接続しているJobCenter CL/Winをすべて終了します。

2.2.2. オプション

オプション指定なし

nscl_termコマンドを発行したマシン上で起動しているJobCenter CL/Winをすべて終了します。

-server $server

指定したマシン(サイト)に接続しているJobCenter CL/Winが終了します。

-user $user

-serverオプションと同時に用い、$serverで指定されたマシンに$userで接続しているJobCenter CL/Winが終了します。

-h

ヘルプを表示します。他のオプションと併用した場合、他のオプションは無視されます。

2.2.3. 戻り値

戻り値	内　容
0	正常終了です。
1	オプション指定が不正です。
2	-serverオプションにサーバ名が指定されていません。
3	-userオプションにユーザ名が指定されていません。

2.3. Nscl_Submit リモートホストのジョブネットワーク投入

■ジョブネットワーク投入コマンド

```
%InstallDirectory%\nscl_submit -s $server -u $user {-pw $passwordfile | -epw $EncryptPasswordfile}
-j $jnwname [-t $YYYYMMDDhhmmss] [-to $SocketTimeoutSeconds] [-c $comment] [-pa $parameter] [-e $key=
$value | -i $inputfile] [-l] [-h]
```

■パスワードファイル作成用コマンド

```
%InstallDirectory%\nscl_submit -m $password -o $outfile [-h]
```

2.3.1. 機能説明

■ローカルまたはリモートホストの任意のユーザの任意のジョブネットワークを投入します。

■成功時にはトラッカIDを、失敗時にはエラーメッセージを返します。

■コマンドはCL/Winのインストールフォルダに格納されています。

■ジョブネットワークの投入には、パスワードを記述したファイルを利用する(-pwオプション)か、パスワードファイル作成用コマンドで作成したファイルを利用する(-epwオプション)必要があります。

セキュリティ上、パスワードを記述したファイルを作成できない場合は、パスワードファイル作成用コマンドを利用してください。

2.3.2. オプション

■ジョブネットワーク投入コマンド

-s $server

接続するマシン名(サイト名)を指定します。

$serverは原則ドメインを付加した形式で指定する必要があります。（サイト名がドメインなしで名前解決できており、その名前でJobCenterがセットアップされている場合はその限りではありません）

接続先のリモートマシンがjccombaseに611以外の番号を割り当てられている場合は、<サイト名:ポート番号>の形式で接続するポート番号を明示的に指定できます。

ポート番号の詳細は<環境構築ガイド>の「2.1 JobCenterで使用するTCPポート」を参照してください。

-u $user

ユーザ名を指定します。ユーザ名に次の文字

!\"#$%&'()*,./:;<=>?@[\\]^`{|}~\t

が使われている場合、不正な文字を含んでいるとしてエラーになります。

JobCenterで使用できるユーザ名の最大長は15バイトです。

-pw $passwordfile

パスワードファイルをフルパスで指定します。

パスワードファイルにユーザのパスワードをそのまま（暗号化せずに）書き込みます。

-epw $EncryptPasswordfile

パスワードファイル作成コマンドを用いて作成したファイルをフルパスで指定します。

- j $jnwname

投入するジョブネットワーク名を指定します。

ジョブネットワーク名の形式は次のとおりです。

- 長さが40バイト以内
- 1文字以上指定
- !\"#$%&'()*,./:;<=>?@[\\]^`{|}~\tは使用できません。

- t $YYYYMMDDhhmmss

投入時刻を指定します。形式は<YYYYMMDDhhmmss>で、localtimeのタイムゾーン（デフォルト：JST-9）に合わせた時刻で指定します。

本オプションを省略した場合や過去の時刻を指定した場合は、デフォルト値として現在時刻を使用します。

時刻の形式は次のとおりです。

- 長さが14バイト（月・日・時・分・秒が1桁のときは、その前に0を付加して2桁で指定）
- 指定が次の範囲内であり、年月日や時刻の組み合わせが正しいこと

  1997≦年≦2037, 1≦月≦12, 1≦日≦31, 時刻は24時間表記

本オプションは <基本操作ガイド>の「3.2.1 即時投入する」の「投入時間」の設定と同じ位置付けとなります。

-to $SocketTimeoutSeconds

JobCenterサーバとの通信タイムアウト値を指定します。

タイムアウトは[0-3599]秒で指定できます。

本オプションを指定しない場合は、既定値の30秒となります。

-c $comment

コメントを指定します。省略した場合はNULLとして扱います。コメント中に改行コードを含むことはできません。

コメント内にスペース文字を含みたい場合は、「-c "comment1 comment2"」のように「"（ダブルクォーテーション）」で囲んでください。囲んでいない場合は、最初の部分（例ではcomment1）のみをコメント文字列として扱います。

なお、「”」自体をエスケープする方法はありませんのでご注意ください。

その他の制限等については <基本操作ガイド>の「3.2.1 即時投入する」 の「コメント」と同様ですので、そちらも参照してください。

最大長は256バイトです。

-pa $parameter

$parameterで指定された文字列が、単位ジョブ実行時の環境変数NSJNW_PARAMとして設定されます。

パラメータ中に改行コードを含むことはできません。

パラメータ内にスペースを含む場合は、-cの場合と同様に「"」で囲んでください。

本オプションは、CL/Winの即時投入操作時のダイアログにおいて「環境変数」-「ジョブパラメータ(NSJNW_PARAM)を設定する」で入力できるパラメータと同等です。

-e $key=$value

単位ジョブ実行時の環境変数として、$key(環境変数名)=$value(環境変数値)を設定します。

複数の環境変数を設定したい場合、本オプションを複数回指定してください。

```
-e key1=value1 -e key2=value2 ...
```

値にスペースを含む文字列を指定したい場合、$key=$value全体をダブルクォーテーションで囲んでください。

```
-e "key1=a b c"
```

本オプションは、CL/Winの即時投入操作時のダイアログにおいて「環境変数」-「複数の環境変数を設定する」で入力できるパラメータと同等です。

環境変数名は最大51バイトで指定してください。

環境変数名には空白、タブ、特殊文字、および =#'" は使用できません。

環境変数値にはタブ、特殊文字、および #'" は使用できません。

本オプションは-iオプションと同時には指定できません。

-i $inputfile

単位ジョブ実行時の環境変数として、$inputfileで指定したファイル内のデータを設定します。$inputfileはkey(環境変数名)=value(環境変数値)の形式で、複数指定する場合は改行して指定してください。

```
key1=value1
key2=value2
...
```

本オプションは、CL/Winの即時投入操作時のダイアログにおいて「環境変数」-「複数の環境変数を設定する」で入力できるパラメータと同等です。

環境変数名は最大51バイトで指定してください。

環境変数名には空白、タブ、特殊文字、および =#'" は使用できません。

環境変数値にはタブ、特殊文字、および #'" は使用できません。

ファイル中の1行は最大8191バイトで指定してください。

本オプションは-eオプションと同時には指定できません。

-l

コマンド成功時に返されるトラッカIDをLocal timeで表示します。省略した場合はトラッカIDをGMTで表示します。トラッカIDに続けて(GMT)または(Local time)が表示されます。

（例）

「ジョブネットワークTEST1を2005年9月15日14:00:00に指定」して投入したときの表示結果は次のようになります。

-lオプションなし：TEST1.20090915050000 (GMT)

-lオプションあり：TEST1.20090915140000 (Local time)

-h

ヘルプを表示します。他のオプションと併用した場合、他のオプションは無視されます。

■パスワードファイル作成用コマンド

-m $password

ユーザのパスワードを指定します。

-o $outfile

作成するパスワードファイルの出力先をフルパスまたは相対パスで指定します。

-h

ヘルプを表示します。他のオプションと併用した場合、他のオプションは無視されます。

2.3.3. 戻り値

戻り値	内　容
0	正常終了です。出力としてトラッカIDを表示します。
1	オプション指定が不正です。
2	-sオプションで指定されたサーバ名が256バイトを超えているか、接続先ポート番号が1～65535の範囲外です。
3	-uオプションのユーザ名に不正なキャラクタが含まれています。
4	パスワードファイルがないか、読み込みエラーが発生しています。
6	ジョブネットワーク名の形式が不正です。
7	時刻の形式が不正、またはコメント長が40バイトを超えています。
10	ネットワーク障害や通信異常、指定されたサーバの名前解決不正、パスワードチェックに失敗しています。 標準エラー出力にサーバからのエラーメッセージを表示します。
11	リモートサーバからのログアウト時にエラーが発生しています。 標準エラー出力にサーバからのエラーメッセージを表示します。

20	指定されたジョブネットワークが存在しないなどサーバ側でエラーを検出しています。 標準エラー出力にサーバからのエラーメッセージを表示します。
22	環境変数設定ファイルの読み込みエラーが発生しています。
30	上記以外のエラーが発生しています。

2.3.4. 注意事項

■ネットワークの問題により指定したリモートホストから返却される通信データが受信できない、またはマシン負荷が高い状況でジョブネットワーク投入に時間がかかる等の理由により、Nscl_Submitがタイムアップして投入結果を得られない場合があります。この場合はリモートホスト側では正常にジョブネットワークが投入されて実行されているにも関わらず、Nscl_Submitコマンド側の戻り値には反映されない状況となります。

そのため、Nscl_Submitによるジョブネットワーク投入の成否は、Nscl_Referでトラッカ情報の有無を確認して判断する必要があります。Nscl_Referには、ジョブネットワーク名と投入時刻から構成されるトラッカIDを指定する必要があるので、Nscl_Submitの-tオプションで、ジョブネットワークの投入時刻をあらかじめ指定しておく必要があります。

（例）

```
Nscl_Submit.exe -s server1 -u user1 -pw passwd.txt -j Jnw1 -t 20121010123456

Nscl_Refer.exe -s server1 -u user1 -pw passwd.txt -tr Jnw1.20121010123456 -l
```

Nscl_Referに-lオプションを指定しない場合、GMTでの時刻指定となりますので注意してください。

■同一ユーザで同一ジョブネットワーク名を指定して本コマンドを複数同時に実行する、あるいは-tオプションで同じ投入時刻を指定して複数回実行したとしても、同一ジョブネットワーク名で複数のトラッカを同一時刻に起動する事は出来ません。タイミング的に最も早くに実行された本コマンドが有効になってトラッカが生成され、その他の本コマンドは実際には何もせず終了することになります。

2.4. Nscl_Refer リモートホストのジョブネットワークの状態を参照

```
%InstallDirectory%nscl_refer -s $server -u $user {-pw $passwordfile | -epw $EncryptPasswordfile} -tr
$trackerID [-l] [-to $SocketTimeoutSeconds] [-h]
```

2.4.1. 機能説明

■ローカルまたはリモートホストの任意のユーザの任意のトラッカ情報を参照します。

■成功時にはトラッカステータスを、失敗時にはエラーメッセージを返します。

■コマンドはCL/Winのインストールフォルダに格納されています。

■本コマンドでは、パスワードを記述したファイルを利用する(-pwオプション)か、パスワードファイル作成用コマンドで作成したファイルを利用する(-epwオプション)必要があります。

セキュリティ上、パスワードを記述したファイルを作成できない場合は、パスワードファイル作成用コマンドを利用してください。パスワードファイル作成用コマンドの詳細は「2.3 Nscl_Submit リモートホストのジョブネットワーク投入」を参照してください。

2.4.2. オプション

-s $server

接続するマシン名(サイト名)を指定します。

$serverは原則ドメインを付加した形式で指定する必要があります。（サイト名がドメインなしで名前解決できており、その名前でJobCenterがセットアップされている場合はその限りではありません）

接続先のリモートマシンがjccombaseに611以外の番号を割り当てられている場合は、<サイト名:ポート番号>の形式で接続するポート番号を明示的に指定できます。

ポート番号の詳細は<環境構築ガイド>の「2.1 JobCenterで使用するTCPポート」を参照してください。

サイト名にIPアドレスを指定することもできます。指定方法は以下のとおりです。

■ポート番号を指定しない場合

<IPアドレス>の形式で指定してください。

■ポート番号を指定する場合

IPv4アドレスの場合は、<IPv4アドレス:ポート番号>の形式で指定してください。

IPv6アドレスの場合は、<[IPv6アドレス]:ポート番号>の形式で指定してください。

-u $user

ユーザ名を指定します。ユーザ名に次の文字

!\"#$%&'()*,./:;<=>?@[\\]^`{|}~\t

が使われている場合、不正な文字を含んでいるとしてエラーになります。

JobCenterで使用できるユーザ名の最大長は15バイトです。

-pw $passwordfile

パスワードファイルをフルパスで指定します。

パスワードファイルにユーザのパスワードをそのまま（暗号化せずに）書き込みます。

-epw $EncryptPasswordfile

パスワードファイル作成コマンド（Nscl_Submitの-mオプション指定）を用いて作成したファイルをフルパスで指定します。詳細は「2.3 Nscl_Submit リモートホストのジョブネットワーク投入」を参照してください。

-tr $trackerID

トラッカを識別するトラッカIDを指定します。

トラッカIDはジョブネットワーク名と投入時刻から構成され、フォーマットは<ジョブネッ

トワーク名.YYYYMMDDhhmmss>になります。

nscl_submitコマンドが返したトラッカIDをそのまま指定してください。

■ジョブネットワーク名の形式は次のとおりです。

- 長さが40バイト以内
- 1文字以上指定
- !\"#$%&'()*,./:;<=>?@[\\]^`{|}~\tを使用できません。

■投入時刻の形式は次のとおりです。-lオプションにより、GMT、Local timeのどちらかになります。

- 長さが14バイト（月・日・時・分・秒が1桁のときは、その前に0を付加して2桁で指定）
- 指定が次の範囲内であり、年月日や時刻の組み合わせが正しいこと。1997≦年≦2037, 1≦月≦12, 1≦日≦31, 時刻は24時間表記

-l

Nscl_Submitコマンド成功時に返されたトラッカIDをLocal timeで指定したい場合に使用します。

本オプションを省略した場合はトラッカIDをGMTで指定します。

（例）

「ジョブネットワークTEST1を2005年9月15日14:00:00に指定」して、Nscl_Submitを-lオプション付き(Local time表記)で投入したときに返されたトラッカID TEST1.20090915140000を、本コマンドの-trに指定する際には次のようになります。

-lオプションなし(GMTで指定)：-tr TEST1.20090915050000

-lオプションあり(Local time指定)：-tr TEST1.20090915140000 -l

-to $SocketTimeoutSeconds

JobCenterサーバとの通信タイムアウト値を指定します。

タイムアウトは[0-3599]秒で指定できます。 本オプションを指定しない場合は30秒となります。

-h ヘルプを表示します。他のオプションと併用した場合、他のオプションは無視されます。

2.4.3. 戻り値

戻り値	内　容
0	正常終了です。標準出力に指定されたトラッカIDのステータスを表示します。 トラッカIDのステータス表示はCL/Winのトラッカ一覧表示と同一の内容となります。ただし[アーカイブ済]かどうかと、サブジョブネットワークの状態までは表示しません。 ステータス表示の詳細は「2.4.4 トラッカIDの表示ステータス」を参照してください。
1	オプション指定が不正です。
2	-sオプションで指定されたサーバ名が256バイトを超えているか、接続先ポート番号が1～65535の範囲外です。
3	-uオプションのユーザ名に不正なキャラクタが含まれています。
4	パスワードファイルがないか、読み込みエラーが発生しています。
5	トラッカIDの形式が不正です。
6	ジョブネットワーク名の形式が不正です。
7	時刻の形式が不正です。
10	ネットワーク障害や通信異常、指定されたサーバの名前解決不正、パスワードチェックに失敗しています。 標準エラー出力にサーバからのエラーメッセージを表示します。
11	リモートサーバからのログアウト時にエラーが発生しています。 標準エラー出力にサーバからのエラーメッセージを表示します。
20	トラッカデータの読み込みに失敗しました。
21	指定されたトラッカIDが存在しません。
30	上記以外のエラーが発生しています。

2.4.4. トラッカIDの表示ステータス

トラッカIDの表示ステータス

「状態」で示す文字列が標準出力に表示されます。

「ラベル」はjnwsummaryで同じトラッカを指定して表示した際にJNW　　STATUSに表示される文字列で、Nscl_Referの「状態」との対応を示します。

状　態	ラベル
予定	WAIT
予定(確定)	PREPARE
ｴﾗｰ停止	STOP(ERROR)
ﾀﾞｲｱﾛｸﾞ	DIALOG
応答待ち	HOLD

実行中異常[待合]	RUN[ERROR][WAIT]
実行中異常	RUN(ERROR)
実行中[警告][待合]	RUN[WARNING][WAIT]
実行中[警告]	RUN[WARNING]
実行中[待合]	RUN[WAIT]
実行中	RUN
中断	ABORT
分岐ｽｷｯﾌﾟ済	SKIPPED
ｽｷｯﾌﾟ済	SKIPPED
中断済	KILLED
異常終了	DONE(ERROR)
警告終了	DONE[WARNING]
ｽｷｯﾌﾟ済[同時実行]	SKIPPED[RUNLIM]
正常終了	DONE
実行中<再実行>	RUN(RERUN)
実行中異常<再実行>	RUN(ERROR)(RERUN)
正常終了<再実行>	DONE(RERUN)
異常終了<再実行>	DONE(ERROR)(RERUN)
不明	UNKNOWN
[ｽｷｯﾌﾟ]	[SKIP]
[保留]	[HOLD]
[ｱｰｶｲﾌﾞ済]	[ARCHIVED]

3. JobCenter MG/SVのコマンド

本章ではJobCenter MG/SVがインストールされたマシンで利用可能なコマンドを説明します。

3.1. jnwsummary ジョブネットワーク実行実績を出力

■UNIX版

```
/usr/lib/nqs/gui/bin/jnwsummary [-o $output] [-u $user | -a] [-c $component] [-j $JNWsearch
[,$JNWsearch ...]] [-p $print_format [,$print_format ...] | -t $testp] [-h]
```

■Windows版

```
%InstallDirectory%\bin\jnwsummary [-o $output] [-u $user | -a] [-c $component] [-j $JNWsearch
[,$JNWsearch...]] [-p $print_format [,$print_format...] | -t $testp] [-h]
```

3.1.1. 機能説明

■ローカルもしくはクラスタサイトについて、アクセス権のある任意のユーザのトラッカ情報(実行実績)を検索、取得し、トラッカ情報の表示もしくは状態の検査を行います。

■コマンド全般に対して有効な一般オプションと、コマンドの動作を切り替える機能オプションがあります。

■表示および検査の対象は、一般オプションの-j（ジョブネットワーク検索オプション）で指定します。

■対象トラッカの実績情報の詳細表示は、機能オプションの-pオプションで指定します（詳細表示モード）。

■トラッカおよび関連ジョブの実行時間や終了状態についてレポートします。

■対象トラッカの状態（実行中や終了など）の検査は、機能オプションの-tオプションで指定します（検査モード）。

■-p, -t両オプションを省略した場合、対象トラッカの一覧の表示のみを行います（一覧表示モード）。

3.1.2. オプション

1. 一般オプション

jnwsummaryコマンド全般に対して有効なオプションです。

-o $output

実績情報の出力を指定したファイル名に出力します。

省略した場合、標準出力に出力します。

-u $user | -a

トラッカ情報を検索するユーザを指定します。

■-u $user

$userで指定したユーザのトラッカを対象とします。ただし、当該ユーザのデータにアクセスするための権限が必要です。

■-a

すべてのユーザを対象とします。

■-uと-aを同時に指定

-aを優先します。

■-uと-aを同時に省略

コマンドを実行したユーザのトラッカのみを対象とします。

-c $component

本コマンドの対象になるジョブネットワーク（JNW）やサブジョブネットワーク（sub　JNW）および部品を直接指定します。-p, -tオプション指定時のみ有効です。

部品指定方法は、“JNW:subJNW...subJNW/部品名”で指定します。対象部品がジョブネットワークやサブジョブネットワークの場合は「/」以降を省略します。また、サブジョブネットワークを持たない場合は“JNW/部品名”で指定します。

（例）

-c “ABC00:ABC01/JOB1”：ABC00中のABC01という名前のサブジョブネットワークの中にあるJOB1という部品

-c “ABC00:ABC01”：ABC00中のABC01という名前のサブジョブネットワーク

-c “ABC00/JOB1”：ABC00という名前のジョブネットワーク中のJOB1という部品

-j $JNWsearch

ジョブネットワーク検索オプションです。

サブオプションと組み合わせることで、検索対象とするトラッカを明示的に指定できます。指定した条件に該当するジョブネットワークが見つからない場合は、エラーになります。

省略した場合、同一ジョブネットワーク名のトラッカが複数あったときには、最新のトラッカだけが検索対象となります（-j lastが指定された状態と同じになります）。

-tオプションを指定した場合、検索結果が唯一のジョブネットワークにならないときにエラーになります。本オプションは、サブオプションと組み合わせて指定することで、コマンドの出力形式を制御できます。

■-jサブオプション

-jオプションは、次の従属するサブオプションを1つ以上指定する必要があります。「,」で区切ることで複数のサブオプションを指定できます。

-j name=$JNWname

検索対象とするトラッカ（ジョブネットワーク名）を$JNWnameに指定します。ジョブネットワーク名にはルートジョブネットワーク名を指定します。

省略した場合、すべてのジョブネットワーク名が検索対象となります。

trkサブオプションと併せて使用することはできません。

-j run

現在実行中のトラッカのみを対象として検索します。

省略した場合、すべてのステータスのジョブネットワークが検索対象となります。

trk, prep, lastサブオプションと併せて使用することはできません。

-j last

検索対象のジョブネットワークのうち、最新の実行結果だけを検索対象とします。

省略した場合、対象のジョブネットワークのすべての実行結果について検索されます。

trk, prep, run, date, todayサブオプションと併せて使用することはできません。

-j date=[YYYYMMDD[-yyyymmdd] | [YYYY/][M]M/DD[-[yyyy/][m]m/dd]]

指定した期間に実行を開始したすべてのジョブネットワークを検索します。YYYY(/)MM(/)DDとyyyy(/)mm(/)ddには、検索期間の始めと終わりを西暦年月日で指定します。

終わりの「-yyyymmdd」および「-[yyyy/][m]m/dd」の指定を省略した場合、指定期間はYYYY年MM月DD日の1日間のみになります。

（例）

-j date=20060925-20061001西暦2006年9月25日～10月1日の7日間

-j date=20061001西暦2006年10月1日の1日間

省略した場合、すべての期間を検索対象とします。

trk, last, todayサブオプションと併せて使用することはできません。

-j today

コマンド実行当日に実行を開始したすべてのジョブネットワークについて検索します。

省略した場合、すべての期間を検索対象とします。

trk, last, dateサブオプションと併せて使用することはできません。

-j prep

検索対象に予定状態のトラッカを含めます。

省略した場合、予定状態のトラッカは検索対象とされません。

run, lastサブオプションと併せて使用することはできません。

-j trk=JNWname.YYYYMMDDHHmmSS

ジョブネットワークID（トラッカ名）を直接指定します。

YYYYMMDDHHmmSSは、ジョブネットワークの投入（予定）をGMTで表したものです。月・日・時・分・秒が1桁のときは、その前に0を付加します。

jnwsubmitcmdでジョブネットワークを投入した場合は、投入時の標準出力に表示されます。

prepサブオプションと組み合わせて使用できます。

name, run, last, date, todayサブオプションと併せて使用することはできません。また、-aオプションと併せて使用することはできません。

-h

ヘルプを表示します。他のオプションと併用した場合、他のオプションは無視されます。

2. 機能オプション

jnwsummaryコマンドを使用する場合は、次のオプションを指定できます。省略した場合（デフォルト）は一覧表示モードとして動作します。

デフォルト　（一覧表示モード）

対象トラッカの現在の状況を一覧表示します。

-pオプション　（詳細表示モード）

検索対象のトラッカ情報を詳細出力します。

本オプションは、サブオプションと組み合わせて指定することで、コマンドの出力形式を制御できます。

■-pサブオプション

-pオプションは、次の従属するサブオプションを1つ以上指定する必要があります。「,」で区切ることで複数のサブオプションを指定できます。

-p format=csv | rd

出力形式をCSV形式もしくは可読形式（readable）のどちらかで指定します。

- -p format=csv

  CSV形式で出力します。

- -p format=rd

  可読形式で出力します。

省略した場合、可読形式として出力されます。

出力形式の詳細については、後述の「出力形式（CSV形式／可読形式）」を参照してください。

formatサブオプション以外の-pサブオプションは、formatサブオプションで指定する出力形式によって出力する情報が異なります。

［CSV形式（format=csv）の場合］

-p style={0|1}

出力情報指定を行います。指定方法は次のとおりです。

- 0

  個別部品情報を出力しません。トラッカのサマリーのみ表示します。

- 1

  全情報の出力を行います。

省略した場合、style=0として動作します。

-p ujonly

サマリーとジョブネットワーク部品情報を省略し、個別部品情報のみを出力します。

style=1の場合のみ有効です。

-p erronly

エラーが発生している部品のみの個別部品情報を出力します。

style=1の場合のみ有効です。

-p notitle

タイトル行の出力を抑制します。

［可読形式（format=rd）の場合］

-p style={0|1|2}

出力情報指定を行います。指定方法は次のとおりです。

- 0

  個別部品情報を出力しません。トラッカのサマリーのみ表示します。

- 1

  一部の個別部品情報を省略します。

- 2

  全情報の出力を行います。

省略した場合、style=0として動作します。

-p ujonly

ジョブネットワーク部品情報を省略し、サマリーと個別部品情報のみを出力します。

style=1（or 2）の場合のみ有効です。

-p erronly

エラーが発生している部品のみの個別部品情報を出力します。

style=1（or 2）の場合のみ有効です。

-p output

単位ジョブ、ERPジョブ、WOBSジョブ、カスタムジョブのSTDOUT(spool),STDERR(joblog)、BIジョブのIPKG STATUS、PCジョブのCHAIN LOG,CHAIN PROCESSを出力します。

style=1（or 2）の場合のみ有効です。

-tオプション　（検査モード）

検索対象のトラッカの状態を指定された検査式（パターン）で評価します。評価結果は、本コマンドの終了値で判断できます。

コマンド終了値と評価結果は次のとおりです。

表3.1 コマンド終了値と評価結果

コマンド終了値	評価結果
0	検査式が真であった。
1	検査式が偽であった。
2	対象部品が存在しない。
3	その他エラーが発生した。

検索結果が唯一のトラッカにならない場合にはエラーになります。その場合は、-jオプションで対象のトラッカを明示的に指定してください。

■-tサブオプション

状態検査オプションtestpは-tのサブオプションであり、独自の検査式を必ず指定します。

-t testp="検査式"

"検査式"には、ジョブネットワークやそのジョブ部品の状態を評価する式を記述します。式は「逆ポーランド」形式で記述します。

式は「パターン（pattern）」と「オペレータ（operator）」からなります。パターンを2つ続けて記述する場合は、「/」で区切ります。

- 検査式のパターン

  式="pattern operator"
  式="pattern / 式 operator"
  式="pattern"

オペレータは「+」（または（or））、「*」（かつ（and））、「!」（否定（not））です。それぞれ評価値に対して作用します。

パターンは指定した部品に対して評価を行うことで真または偽の評価値を持ちます。

- 部品の属性による検査の場合の評価値

  hold　　部品に「保留」が指定されている場合に真
  skip　　部品に「skip」が指定されている場合に真、条件分岐によりskipになっている場合も真

■ 部品の状態による検査の場合の評価値

部品は必ず次のいずれかの状態になっています。

wait	部品がwait状態の場合に真
submit	部品がsubmit状態の場合に真
run	部品がrun状態の場合に真
done	部品がdone状態の場合に真（エラー終了も含みます）

■ その他付属的な属性による検査の場合の評価値

error	該当部品でエラーが発生している状態の場合に真（同時に部品はdone状態）
stopped	該当部品でエラーが発生し、ジョブネットワークの実行が停止している状態の場合に真（同時に部品はdone状態であり、かつerror状態）
complete	該当部品はdone状態であり、かつerrorではない状態の場合に真
archived	属するジョブネットワークはアーカイブ済みであり、再実行できません（同時に部品はdone状態）

（例）

testp="error!done*"：実行が完了しており、かつエラーではない。

testp="complete"：実行が完了しており、かつエラーではない。

testp="hold/skip+wait*"：wait状態でかつhold状態またはskip状態

3.1.3. 戻り値

戻り値	内　容
0	正常終了です（-tオプション使用時を除く）。
0以外	異常終了です。エラー内容のメッセージを標準エラーに出力します。

3.1.4. 注意事項

■単位ジョブパラメータにおいて［結果］タブの標準出力先を変更している場合は、結果を参照できません。

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITEにサイト名を設定する必要があります。(Windowsの場合NQS_SITEDBにサイトのデータベース名の設定も必要)

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

■R13.1以降のバージョンでは、JobCenterが停止している状態で本コマンドを使用することはできません。

3.1.5. 出力形式（CSV形式／可読形式）

1. CSV形式（-p format=csv）

jnwsummaryでジョブネットワーク実行実績情報をCSV形式で出力した場合のフォーマットについて説明します。

jnwsummaryのデータ出力は、次のような異なる形式から構成されます。

■ジョブネットワークサマリー

■ジョブネットワーク部品情報

■単位ジョブ部品情報

■ERPジョブ部品情報

■BIジョブ部品情報

■PCジョブ部品情報

■WOBSジョブ部品情報

カスタムジョブ部品は、単位ジョブ部品として扱われますので注意してください。

［項目形式］

ジョブネットワークサマリーは全体で一度だけ出力されます。それ以外の項目は出現した部品の項目に応じて出現回数が決定されます。

CSVファイルではそれぞれの項目種別ごとに連続して出力されます。このため単位ジョブなどの部品は、必ずいずれかのジョブネットワークに属しています。ただし、このような構造の判断はCSVの行の並びではなく、カラムに記述された所属ジョブネットワーク名から行う必要があります。

これらの項目はコマンドのオプションにより出力を選択できます。

それぞれの項目種別を最初に出力するときに、次のようなフォーマットになります。

```
#TYPE=項目形式名
各項目名（例：jnwname,jnwdate,…）
各項目の値（例：rootJNW,19991220133000,…）
```

表3.2 項目形式名

項目形式名	説　明
SUMMARY	ジョブネットワークサマリー
JNW	ジョブネットワーク部品情報
UNITJOB	単位ジョブ部品情報
ERPJOB	ERPジョブ部品情報
BIJOB	BIジョブ部品情報
PCJOB	PCジョブ部品情報
WOBSJOB	WOBSジョブ部品情報

また、異なる項目形式種別を出力するときには空白行が1行挿入されます。タイトル行はコマンドのオプションにより出力を制限できます。

項目形式ごとに、次に示す項目名がカンマ区切りで出力されます。各項目のデータはカラム単位で出力されます。

情報のない項目は空データが設定されます。

■#TYPE=SUMMARY

項目名	説　明
jnwname	ジョブネットワーク名
jnwdate	ジョブネットワーク投入日付（GMT）（YYYYMMDDHHmmSS）
user	ユーザ
machine	実行マシン
starttime	実績開始時刻

	時刻のフォーマット共通（YYYYY/MM/DD HH:mm:SS）
endtime	実績終了時刻
amounttime	実行時間（秒単位）数値
expstarttime	予定開始時刻
expendtime	予定終了時刻
status	現在のステータス ステータスとしては以下を出力します。 ■ Windowsの場合 prepare　予定状態かつ確定の状態 wait　予定状態かつ未確定の状態 wait_hold　予定状態かつ保留状態 wait_skip　予定状態かつスキップ状態 submit　投入中の状態 run　実行中の状態かつ正常状態 run_warn　実行中かつ警告状態 run_error　実行中かつ異常状態 run_dialog　ダイアログ待ち合わせ中の状態 done　正常終了、または警告終了の状態 done_skip　スキップによる中断済みの状態 error　異常終了の状態 errorstop　異常終了かつ停止状態 ■ UNIXの場合 wait　予定状態 run　投入中の状態、実行中の状態、ダイアログ待ち合わせ中の状態、または異常終了かつ停止状態 done　正常終了の状態、警告終了の状態、異常終了の状態、またはスキップによる中断済みの状態
totalcomponent	全部品の数
totalwait	全部品中未実行の数
totalwaithold	全部品中未実行でholdの数
totalwaitskip	全部品中未実行でskipする予定の数
totalrun	全部品中実行中の数
totaldone	全部品中実行済みの数
totaldoneskip	全部品中実行済みでskip済みの数
totalerror	全部品中でerrorの発生した数
ujcomponent	全単位ジョブの数
ujwait	全単位ジョブ中未実行の数
ujwaithold	全単位ジョブ中未実行でholdの数
ujwaitskip	全単位ジョブ中未実行でskipする予定の数
ujrun	全単位ジョブ中実行中の数

ujdone	全単位ジョブ中実行済みの数
ujdoneskip	全単位ジョブ中実行済みでskip済みの数
ujerror	全単位ジョブ中でerrorの発生した数
jnwcomponent	全サブジョブネットワークの数
jnwwait	全サブジョブネットワーク中未実行の数
jnwwaithold	全サブジョブネットワーク中未実行でholdの数
jnwwaitskip	全サブジョブネットワーク中未実行でskipする予定の数
jnwrun	全サブジョブネットワーク中実行中の数
jnwdone	全サブジョブネットワーク中実行済みの数
jnwdoneskip	全サブジョブネットワーク中実行済みでskip済みの数
jnwerror	全サブジョブネットワーク中でerrorの発生した数
ERPcomponent	全ERPジョブの数
ERPwait	全ERPジョブ中未実行の数
ERPwaithold	全ERPジョブ中未実行でhold の数
ERPwaitskip	全ERPジョブ中未実行でskip する予定の数
ERPrun	全ERPジョブ中実行中の数
ERPdone	全ERPジョブ中実行済みの数
ERPdoneskip	全ERPジョブ中実行済みでskip済みの数
ERPerror	全ERPジョブ中でerrorの発生した数
BIcomponent	全BIジョブの数
BIwait	全BIジョブ中未実行の数
BIwaithold	全BIジョブ中未実行でhold の数
BIwaitskip	全BIジョブ中未実行でskip する予定の数
BIrun	全BIジョブ中実行中の数
BIdone	全BIジョブ中実行済みの数
BIdoneskip	全BIジョブ中実行済みでskip済みの数
BIerror	全BIジョブ中でerrorの発生した数
othercomponent	その他部品の数
otherwait	その他部品中未実行の数
otherwaithold	その他部品中未実行でholdの数
otherwaitskip	その他部品中未実行でskipする予定の数
otherrun	その他部品中実行中の数
otherdone	その他部品中実行済みの数
otherdoneskip	その他部品中実行済みでskip済みの数
othererror	その他部品中でerrorの発生した数

■#TYPE=JNW（ジョブネットワーク）

部品の個数に、ジョブネットワーク中のサブジョブネットワーク分を含みません。

項目名	説　明
type	“JNW”固定
parentjnw	所属ジョブネットワーク名（JNW:subJNW...）

jnwdate	ジョブネットワーク投入日付（GMT）（YYYYMMDDHHmmSS）
name	ジョブネットワーク名
starttime	実績開始時刻 時刻のフォーマット共通（YYYYY/MM/DD HH:mm:SS）
endtime	実績終了時刻
amounttime	実行時間（秒単位）
expstarttime	予定開始時刻
expendtime	予定終了時刻
status	現在のステータス ステータスとしては以下を出力します。 ■ Windowsの場合 prepare　予定状態かつ確定の状態 wait　予定状態かつ未確定の状態 wait_hold　予定状態かつ保留状態 wait_skip　予定状態かつスキップ状態 submit　投入中の状態 run　実行中の状態かつ正常状態 run_warn　実行中かつ警告状態 run_error　実行中かつ異常状態 run_dialog　ダイアログ待ち合わせ中の状態 done　正常終了、または警告終了の状態 done_skip　スキップによる中断済みの状態 error　異常終了の状態 errorstop　異常終了かつ停止状態 ■ UNIXの場合 wait　予定状態 run　投入中の状態、実行中の状態、ダイアログ待ち合わせ中の状態、または異常終了かつ停止状態 done　正常終了の状態、警告終了の状態、異常終了の状態、またはスキップによる中断済みの状態
totalcomponent	全部品の数
totalwait	全部品中未実行の数
totalwaithold	全部品中未実行でholdの数
totalwaitskip	全部品中未実行でskipする予定の数
totalrun	全部品中実行中の数
totaldone	全部品中実行済みの数
totaldoneskip	全部品中実行済みでskip済みの数
totalerror	全部品中でerrorの発生した数
ujcomponent	全単位ジョブの数
ujwait	全単位ジョブ中未実行の数

ujwaithold	全単位ジョブ中未実行でholdの数
ujwaitskip	全単位ジョブ中未実行でskipする予定の数
ujrun	全単位ジョブ中実行中の数
ujdone	全単位ジョブ中実行済みの数
ujdoneskip	全単位ジョブ中実行済みでskip済みの数
ujerror	全単位ジョブ中でerrorの発生した数
jnwcomponent	全サブジョブネットワークの数
jnwwait	全サブジョブネットワーク中未実行の数
jnwwaithold	全サブジョブネットワーク中未実行でholdの数
jnwwaitskip	全サブジョブネットワーク中未実行でskipする予定の数
jnwrun	全サブジョブネットワーク中実行中の数
jnwdone	全サブジョブネットワーク中実行済みの数
jnwdoneskip	全サブジョブネットワーク中実行済みでskip済みの数
jnwerror	全サブジョブネットワーク中でerrorの発生した数
ERPcomponent	全ERPジョブの数
ERPwait	全ERPジョブ中未実行の数
ERPwaithold	全ERPジョブ中未実行でholdの数
ERPwaitskip	全ERPジョブ中未実行でskipする予定の数
ERPrun	全ERPジョブ中実行中の数
ERPdone	全ERPジョブ中実行済みの数
ERPdoneskip	全ERPジョブ中実行済みでskip済みの数
ERPerror	全ERPジョブ中でerrorの発生した数
BIcomponent	全BIジョブの数
BIwait	全BIジョブ中未実行の数
BIwaithold	全BIジョブ中未実行でholdの数
BIwaitskip	全BIジョブ中未実行でskipする予定の数
BIrun	全BIジョブ中実行中の数
BIdone	全BIジョブ中実行済みの数
BIdoneskip	全BIジョブ中実行済みでskip済みの数
BIerror	全BIジョブ中でerrorの発生した数
othercomponent	その他部品の数
otherwait	その他部品中未実行の数
otherwaithold	その他部品中未実行でholdの数
otherwaitskip	その他部品中未実行でskipする予定の数
otherrun	その他部品中実行中の数
otherdone	その他部品中実行済みの数
otherdoneskip	その他部品中実行済みでskip済みの数
othererror	その他部品中でerrorの発生した数

■#TYPE=UNITJOB（単位ジョブ部品情報）

項目名	説　明

type	“UNITJOB”固定
parentjnw	所属ジョブネットワーク名（JNW:subJNW...）
jnwdate	ジョブネットワーク投入日付（GMT）（YYYYMMDDHHmmSS）
name	単位ジョブ部品名
starttime	実績開始時刻 時刻のフォーマット共通（YYYYY/MM/DD HH:mm:SS）
endtime	実績終了時刻
amounttime	実行時間（秒単位）
expstarttime	予定開始時刻
expendtime	予定終了時刻
status	現在のステータス ステータスとしては以下を出力します。 ■ Windowsの場合 wait　未実行の状態 wait_hold　未実行の状態かつ保留状態 wait_skip　未実行の状態かつスキップ状態 submit　投入中の状態 run　実行中の状態 run_warn　警告終了の状態 done　正常終了の状態 done_skip　スキップにより終了した状態 error　異常終了の状態 errorstop　異常終了かつ停止状態 ■ UNIXの場合 wait　未実行の状態 submit　投入中の状態 run　実行中の状態 done　正常終了の状態、警告終了の状態、またはスキップにより終了した状態 errorpass　異常終了の状態 errorstop　異常終了かつ停止状態
exitcode	ジョブの終了値（内部エラーの場合は、値を設定しない）
errcode	エラーコード
requestid	リクエストID
machine	ジョブ実行マシン名

■#TYPE=ERPJOB（ERPジョブ部品情報）

項目名	説　明
type	“ERPJOB”固定
parentjnw	所属ジョブネットワーク名（JNW:subJNW...）

jnwdate	ジョブネットワーク投入日付（GMT）（YYYYMMDDHHmmSS）
name	ERPジョブ部品名
starttime	実績開始時刻 時刻のフォーマット共通（YYYYY/MM/DD HH:mm:SS）
endtime	実績終了時刻
amounttime	実行時間（秒単位）
expstarttime	予定開始時刻
expendtime	予定終了時刻
status	現在のステータス ステータスとしては以下を出力します。 ■ Windowsの場合 wait　未実行の状態 wait_hold　未実行の状態かつ保留状態 wait_skip　未実行の状態かつスキップ状態 submit　投入中の状態 run　実行中の状態 done　正常終了の状態 done_skip　スキップにより終了した状態 error　異常終了の状態 errorstop　異常終了かつ停止状態 ■ UNIXの場合 wait　未実行の状態 submit　投入中の状態 run　実行中の状態 done　正常終了の状態、またはスキップにより終了した状態 errorpass　異常終了の状態 errorstop　異常終了かつ停止状態
machine	ジョブ実行マシン名

■#TYPE=BIJOB（BIジョブ部品情報）

項目名	説　明
type	“BIJOB”固定
parentjnw	所属ジョブネットワーク名（JNW:subJNW...）
jnwdate	ジョブネットワーク投入日付（GMT）（YYYYMMDDHHmmSS）
name	BIジョブ部品名
starttime	実績開始時刻 時刻のフォーマット共通（YYYYY/MM/DD HH:mm:SS）
endtime	実績終了時刻
amounttime	実行時間（秒単位）

expstarttime	予定開始時刻
expendtime	予定終了時刻
status	現在のステータス ステータスとしては以下を出力します。 ■ Windowsの場合 wait　未実行の状態 wait_hold　未実行の状態かつ保留状態 wait_skip　未実行の状態かつスキップ状態 submit　投入中の状態 run　実行中の状態 done　正常終了の状態 done_skip　スキップにより終了した状態 error　異常終了の状態 errorstop　異常終了かつ停止状態 ■ UNIXの場合 wait　未実行の状態 submit　投入中の状態 run　実行中の状態 done　正常終了の状態、またはスキップにより終了した状態 errorpass　異常終了の状態 errorstop　異常終了かつ停止状態

■#TYPE=PCJOB（PCジョブ部品情報）

項目名	説　明
type	“PCJOB”固定
parentjnw	所属ジョブネットワーク名（JNW:subJNW...）
jnwdate	ジョブネットワーク投入日付（GMT）（YYYYMMDDHHmmSS）
name	PCジョブ部品名
starttime	実績開始時刻 時刻のフォーマット共通（YYYYY/MM/DD HH:mm:SS）
endtime	実績終了時刻
amounttime	実行時間（秒単位）
expstarttime	予定開始時刻
expendtime	予定終了時刻
status	現在のステータス ステータスとしては以下を出力します。 ■ Windowsの場合 wait　未実行の状態 wait_hold　未実行の状態かつ保留状態

	wait_skip　未実行の状態かつスキップ状態 submit　投入中の状態 run　実行中の状態 done　正常終了の状態 done_skip　スキップにより終了した状態 error　異常終了の状態 errorstop　異常終了かつ停止状態 ▪ UNIXの場合 wait　未実行の状態 submit　投入中の状態 run　実行中の状態 done　正常終了の状態、またはスキップにより終了した状態 errorpass　異常終了の状態 errorstop　異常終了かつ停止状態

■#TYPE=WOBSJOB（WOBSジョブ部品情報）

項目名	説　明
type	“WOBSJOB”固定
parentjnw	所属ジョブネットワーク名（JNW:subJNW...）
jnwdate	ジョブネットワーク投入日付（GMT）（YYYYMMDDHHmmSS）
name	WOBSジョブ部品名
starttime	実績開始時刻 時刻のフォーマット共通（YYYYY/MM/DD HH:mm:SS）
endtime	実績終了時刻
amounttime	実行時間（秒単位）
expstarttime	予定開始時刻
expendtime	予定終了時刻
status	現在のステータス ステータスとしては以下を出力します。 ▪ Windowsの場合 wait　未実行の状態 wait_hold　未実行の状態かつ保留状態 wait_skip　未実行の状態かつスキップ状態 submit　投入中の状態 run　実行中の状態 run_warn　警告終了の状態 done　正常終了の状態 done_skip　スキップにより終了した状態 error　異常終了の状態 errorstop　異常終了かつ停止状態

	■ UNIXの場合 wait　未実行の状態 submit　投入中の状態 run　実行中の状態 done　正常終了の状態、警告終了の状態、またはスキップにより終了した状態 errorpass　異常終了の状態 errorstop　異常終了かつ停止状態
exitcode	ジョブの終了値
errcode	エラーコード

［出力例］

「単位ジョブ、ERPジョブをそれぞれ１つずつ含むサブジョブネットワークsubJNW」と「単位ジョブ、ERPジョブ、BIジョブをそれぞれ１つずつ含むジョブネットワークrootJNW」の全出力のサンプルは、次のようになります。

```
#TYPE=SUMMARY
jnwname,jnwdate,user,machine,starttime,endtime,amounttime,expstarttime,expendti
me,status,totalcomponent,totalwait,totalwaithold,totalwaitskip,totalrun,totaldo
ne,totaldoneskip,totalerror,ujcomponent,ujwait,ujwaithold,ujwaitskip,ujrun,ujdo
ne,ujdoneskip,ujerror,jnwcomponent,jnwwait,jnwwaithold,jnwwaitskip,jnwrun,jnwdo
ne,jnwdoneskip,jnwerror,ERPcomponent,ERPwait,ERPwaithold,ERPwaitskip,ERPrun,ERP
done,ERPdoneskip,ERPerror,BIcomponent,BIwait,BIwaithold,BIwaitskip,BIrun,BIdone
,BIdoneskip,BIerror,othercomponent,otherwait,otherwaithold,otherwaitskip,otherr
un,otherdone,otherdoneskip,othererror
rootJNW,20060818050507,npxjc,ORGAN,2006/08/18 14:05:08,,0,2006/08/18 14:05:07,2
006/08/18 14:05:07,run,7,1,1,0,3,3,0,0,2,0,0,0,0,2,0,0,2,0,0,0,2,0,0,0,2,0,0,0,
1,1,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0

#TYPE=JNW
type,parentjnw,jnwdate,name,starttime,endtime,amounttime,expstarttime,expendtim
e,status,totalcomponent,totalwait,totalwaithold,totalwaitskip,totalrun,totaldon
e,totaldoneskip,totalerror,ujcomponent,ujwait,ujwaithold,ujwaitskip,ujrun,ujdon
e,ujdoneskip,ujerror,jnwcomponent,jnwwait,jnwwaithold,jnwwaitskip,jnwrun,jnwdon
e,jnwdoneskip,jnwerror,ERPcomponent,ERPwait,ERPwaithold,ERPwaitskip,ERPrun,ERPd
one,ERPdoneskip,ERPerror,BIcomponent,BIwait,BIwaithold,BIwaitskip,BIrun,BIdone,
BIdoneskip,BIerror,othercomponent,otherwait,otherwaithold,otherwaitskip,otherru
n,otherdone,otherdoneskip,othererror
JNW,rootJNW,20060818050507,subJNW,2006/08/18 14:05:08,,0,2006/08/18 14:05:07,20
06/08/18 14:05:07,run,2,0,0,0,1,1,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
JNW,"",20060818050507,rootJNW,2006/08/18 14:05:08,,0,2006/08/18 14:05:07,2006/0
8/18 14:05:07,run,4,1,1,0,1,2,0,0,1,0,0,0,0,1,0,0,1,0,0,0,1,0,0,0,1,0,0,0,0,1,0
,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0

#TYPE=UNITJOB
type,parentjnw,jnwdate,name,starttime,endtime,amounttime,expstarttime,expendtim
e,status,exitcode,errcode,requestid,machine
UNITJOB,rootJNW:subJNW,20060818050507,JOB1,2006/08/18 14:05:12,2006/08/18 14:05
:13,1,2006/08/18 14:05:07,2006/08/18 14:05:07,done,0,0,3.ORGAN,ORGAN
UNITJOB,rootJNW,20060818050507,JOB1,2006/08/18 14:05:10,2006/08/18 14:05:10,0,2
006/08/18 14:05:07,2006/08/18 14:05:07,done,0,0,2.ORGAN,ORGAN
```

```
#TYPE=ERPJOB
type,parentjnw,jnwdate,name,starttime,endtime,amounttime,expstarttime,expendtim
e,status,machine
ERPJOB,rootJNW:subJNW,20060818050507,JOB2,2006/08/18 13:33:24,,0,2006/08/18 14:
05:07,2006/08/18 14:05:07,run,-
ERPJOB,rootJNW,20060818050507,JOB2,2006/08/18 13:33:22,2006/08/18 13:33:23,1,20
06/08/18 14:05:07,2006/08/18 14:05:07,done,-

#TYPE=BIJOB
type,parentjnw,jnwdate,name,starttime,endtime,amounttime,expstarttime,expendtim
e,status,
BIJOB,rootJNW,20060818050507,JOB3,,,0,2006/08/18 14:05:07,2006/08/18 14:05:07,w
ait_hold,
```

2. 可読形式（readable）

jnwsummaryでジョブネットワークの実行実績情報を可読形式で出力した場合のフォーマットについて説明します。

次のような情報がフォーマットされて出力されます。該当する項目に情報がない場合「-」で印字されます。

表3.3 ジョブネットワークの実行実績情報の可読形式フォーマット

項目名	説　明
トラッカID (TRACKER-ID)	ジョブネットワークを識別するトラッカID
ユーザ名 (USER)	ジョブネットワークを投入したユーザ名
実績開始時刻(START TIME)	実際に処理を開始した時間
実績終了時刻(END TIME)	実際に処理を終了した時間
予定開始時刻(START TIME(EXP.))	ジョブネットワーク投入時刻から予想される開始時間
予定終了時刻(END TIME(EXP.))	ジョブネットワーク投入時刻から予想される終了時間
実行時間(秒単位)(AMOUNTTIME)	実績の開始から終了までの時間
ステータス(STATUS)	現在の該当部品のステータス
部品数(COUNT)	ジョブネットワークに定義された全部品数 サブジョブネットワークの配下の部品も計算されます。 各部品および状態ごとに個数を計算した情報も同時に出力します。 対象とする部品は次のとおりです。 ■全情報(ALL) ■単位ジョブ(UJ) ■ERPジョブ(ERP) ■BIジョブ(BI) ■サブジョブネットワーク(JNW) ■PCジョブ(PC) ■WOBSジョブ(WOBSJ)

	■カスタムジョブ(CSTMJ) ■その他(OTHER) 上記部品について次の統計を出力します。 ■全個数(TOTAL) ■未実行(WAIT) ■未実行のうちholdのもの(WAIT(hold)) ■未実行のうちskipする予定のもの(WAIT(skip)) ■実行中(RUN) ■実行済み(DONE) ■実行済みのうちskip済みのもの(DONE(skip)) ■errorが発生しているもの(ERROR)
終了コード(EXIT CODE)	単位ジョブ(シェル)の終了値またはエラーコード
実行マシン名(EXEC MAC)	ジョブが実行されたマシン名
リクエストID(REQUEST-ID)	単位ジョブを処理したリクエストID
標準出力(STDOUT)	特別に指定した場合のみ出力
標準エラー出力(STDERR)	特別に指定した場合のみ出力

3.2. jnwschprt ジョブネットワークのカレンダやスケジュール情報を表示

■UNIX版

```
/usr/lib/nqs/gui/bin/jnwschprt [-l注1] [-I注2] [-o $output] [-u $user | -a] [-d $date] [-p $starttime]

/usr/lib/nqs/gui/bin/jnwschprt -s [-I注2] [-o $output] [-f {csv | rd}] [-u $user] [-c | +c] [-C | +C]
-j $jnwname -d $date [-p $starttime]

/usr/lib/nqs/gui/bin/jnwschprt -s [-I注2] [-o $output] {-f {csv | rd}} [-u $user] [-c | +c] [-C | +C]
-i $jnwname.YYYYMMDDHHmmSS
```

注1小文字のエル
注2大文字のアイ

■Windows版

```
%InstallDirectory%\bin\jnwschprt [-l注1] [-I注2] [-o $output] [-u $user | -a] [-d $date] [-p $starttime]

%InstallDirectory%\bin\jnwschprt -s [-I注2] [-o $output] [-f {csv | rd}] [-u $user] [-c | +c] [-C | +C]
-j $jnwname -d $date [-p $starttime]

%InstallDirectory%\bin\jnwschprt -s [-I注2] [-o $output] [-f {csv | rd}] [-u $user] [-c | +c] [-C |
+C] -i $jnwname.YYYYMMDDHHmmSS
```

注1小文字のエル
注2大文字のアイ

3.2.1. 機能説明

■コマンドを実行するローカルもしくはクラスタサイトについて、アクセス権のある任意のユーザの任意のジョブネットワークについて、スケジュールされている予定トラッカ情報の表示を行います。

■第一の形式では、日付を指定して、その日に実行が予定されている予定トラッカの一覧を表示します。日付を指定しない場合はコマンド実行当日の一覧を表示します。

■第二の形式では、指定したジョブネットワークについて指定した日付に異なる予定時刻の複数の予定トラッカが存在する場合、最も古い時刻の予定トラッカについて、指定日におけるカレンダ分岐を加味した単位ジョブの情報を表示します。ただし指定日に実行中の複数のトラッカが存在していてまだ実行前の単位ジョブを含んでいる場合は、実行中のトラッカのうち最も新しい投入日付のトラッカが選択されて単位ジョブの情報を表示します。

■第三の形式では、指定したトラッカIDについて実行(予定)日におけるカレンダ分岐を加味した単位ジョブの一覧を表示します。トラッカIDは過去の日時のものも指定可能です。

3.2.2. オプション

-l(小文字のエル)

ジョブネットワークの一覧表示を行います。

-s

指定したジョブネットワークのジョブのスケジュール情報を出力します。

-I(大文字のアイ)

コンフィグレーションファイルの読み込みを禁止します。

-o $output

実行結果をoutputで指定したファイルへ出力します。本オプションを省略した場合、標準出力に出力します。

-u $user | -a

ジョブネットワークを検索するユーザを指定します。

■-u $user

userで指定したユーザ名に対してジョブネットワークの検索を行います。ただし、ファイルにアクセス権がない場合は実行できません。

■-aと-uの両方を同時に省略

コマンドを実行したユーザのジョブネットワークについて検索を行います。

■-a

ジョブネットワークの検索を全ユーザに対して実行します。ただし、ファイルにアクセス権がない場合は実行できません。

-j $jnwname

検索対象とするジョブネットワーク名を$jnwnameに指定します。ジョブネットワーク名にはルートジョブネットワーク名を指定します。

-d $date

ジョブネットワークの検索範囲またはスケジュール表示の日付をdateに指定します。

次のフォーマットで指定します。

-d [YYYY/]MM/DD

-p $starttime

指定した日付が開始される時刻を指定します。

省略した場合、日付はLocal timeのAM0:00から開始されます。

バッチジョブ業務に併せた日次処理の一覧を正しく得るために使用します。

次のフォーマットで指定します。

-p HH:MM

-f csv | rd

出力をCSV形式または可読形式（readable）で出力します。

■-f csv

CSVで出力します。

■-f rd

可読形式で出力します。

省略した場合、可読形式になります。

-c | +c

情報出力に、直前の先行ジョブまたは待ち合わせ部品の情報を出力します。

■-c

情報を出力します。

■+c

その出力を抑止します。

省略した場合、出力になります。

-C | +C

スケジュール情報出力に、すべての先行ジョブまたは待ち合わせ部品の情報を出力します。

■-C

情報を出力します。

■+C

その出力を抑止します。

省略した場合、出力を抑止します。

-i $jnwname.YYYYMMDDHHmmSS

トラッカを識別するトラッカIDを指定します。

トラッカIDはジョブネットワーク名と投入時刻から構成され、投入時刻の西暦年月日時分秒をGMT表記で指定します。

本コマンドを第一の形式で実行して表示されたトラッカIDをこのオプションに指定する使用方法を推奨します。

（例）

-i jnw1.20090501130500

ジョブネットワークjnw1の、西暦2009年5月1日22時5分投入(予定)のトラッカID

3.2.2.1. コンフィグレーションファイル

頻繁に指定するオプションをコンフィグレーションファイルに記述することでデフォルト値として使用できます。この場合でもコマンドラインにオプションを指定することで設定した値を変更できます。

コンフィグレーションファイルは、以下のインストールディレクトリ配下、または各ユーザのデータディレクトリ配下にjnwschprt.fというファイル名で作成します。

■ローカルサイト/クラスタサイト内共通コンフィグレーションファイル

UNIX	/usr/lib/nqs/gui/bin/jnwschprt.f
Windows	%InstallDirectory%\spool\conf\JNWSCHPRT.F

■各ユーザデータディレクトリ配下のコンフィグレーションファイル

UNIX	/usr/spool/nqs/gui/<ユーザ名>/jnwschprt.f

Windows	%InstallDirectory%\spool\users\<ユーザ名>\JNWSCHPRT.F

クラスタ環境の場合は /usr/spool や %InstallDirectory% の部分を <クラスタDBパス>と読み替えてください。

ファイルには、コマンドラインで指定するオプションをそのまま記述します(コマンド名そのものは不要です)。なお、シェルが作用する特殊文字（環境変数やクオート文字）は解釈せずそのままの状態でjnwschprtコマンドに渡されますので注意してください。

（設定例）次のような情報を出力する場合の記述例です。

■ジョブネットワーク一覧で、日付の区切りをAM8:00とする。

■全ユーザのジョブネットワークを検索して出力する。

■スケジュール情報では全先行ジョブを出力する。

```
% cat /usr/lib/nqs/gui/bin/jnwschprt.f
-s -p 8:00 -a -C +c
```

なお、コマンドラインと上記の各jnwschprt.f内に同じ種類のオプションが指定されて、かつそれぞれ異なるパラメータが指定された場合は、コマンドライン指定を最優先で解釈します。異なる種類のオプションについては、全て解釈の対象となります。

（設定例）各ファイルに異なるオプションが設定されている状態でコマンドを実行した例です。

```
% cat /usr/lib/nqs/gui/bin/jnwschprt.f
-a -p 9:00
% cat /usr/spool/nqs/gui/<ユーザ名>/jnwschprt.f
-s -p 8:00
% /usr/lib/nqs/gui/bin/jnwschprt -C -p 7:00
```

この場合、重なっている -p オプション指定はコマンドラインが優先されるため、

```
% /usr/lib/nqs/gui/bin/jnwschprt -a -s -C -p 7:00
```

を実行した場合と同じに解釈されます。

もしコマンドラインに -p オプション指定が無い場合は、インストールディレクトリ配下の共通コンフィグレーションファイルが優先されますので、

```
% /usr/lib/nqs/gui/bin/jnwschprt -a -s -C -p 7:00
```

を実行したものと解釈されます。

3.2.2.2. ジョブネットワーク一覧情報

次のような情報を出力します。

■検索した日付（検索開始時刻-検索終了時刻）

■ユーザ名

■ジョブネットワーク名

■投入予定日時

■ジョブネットワーク-ID名（JNW名.YYYYMMDDHHmmSS）

全ユーザ検索を行ったときに、ファイルにアクセスできないユーザがあった場合は、その旨を報告します。

3.2.2.3. スケジュール一覧情報

出力にはCSV形式と可読形式（readable）があります。CSV形式の詳細については後述の「3.2.6 スケジュール情報のCSV形式」を参照してください。

スケジュール一覧情報には、ジョブネットワークについての情報（サマリー）と個々の単位ジョブ、ERPジョブ、BIジョブ、PCジョブ、WOBSジョブ、カスタムジョブのスケジュール情報を次のように出力します。

■ジョブネットワークサマリー

- ジョブネットワーク名
- ユーザ名
- 予定開始時間
- トラッカ名（JNW.YYYYMMDDHHmmSS）
- ジョブ部品個数（単位ジョブ個数＋ERPジョブ個数＋BIジョブ個数＋PCジョブ個数＋WOBSジョブ個数＋カスタムジョブ個数）
- カレンダ分岐でスキップされたジョブ部品個数

■スケジュール情報

- 部品名
- 所属ジョブネットワーク
- 事前保留および事前スキップの有無
- 直前の先行ジョブまたは待ち合わせ部品
- すべての先行ジョブまたは待ち合わせ部品

一部の情報は、オプションにより出力が抑制される場合があります。

3.2.2.4. カレンダ分岐の扱い

スケジュール情報の一覧を出力するとき、指定した日付条件に従ってカレンダ分岐の判断を行います。判断の結果実行されないジョブについては、スケジュール情報の出力の対象にはしません。

3.2.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.2.4. 関連ファイル

jnwschprt.f

3.2.5. 注意事項

■スケジュール情報表示を行うときに、過去のエントリーを表示することも可能です。ただし、ジョブネットワークフローの情報は必ず現在の定義情報を参照するため、実際に処理されたスケジュール情報とは異なることがあります。

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITEにサイト名を設定する必要があります。(Windowsの場合NQS_SITEDBにサイトのデータベースパスの設定も必要)

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

■Windows版では、実行済みのジョブネットワークについても出力の対象となりますが、UNIX版では実行予定のジョブネットワークのみ出力の対象となります。

■R13.1以降のバージョンでは、JobCenterが停止している状態で本コマンドを使用す ることはできません。

■環境変数LANGをJobCenterのセットアップ言語と合わせる必要があります。

3.2.6. スケジュール情報のCSV形式

jnwschprtのスケジュール情報をCSV形式で出力した場合の詳細なフォーマットについて説明します。

［項目形式］

スケジュール情報は、次の2つの異なる形式からなります。

■ジョブネットワークサマリー

出力の先頭で一度だけ出力されます。

■ジョブ部品（単位ジョブまたはERPジョブまたはBIジョブ）のスケジュール情報

それぞれの部品について繰り返し出力されます。

表3.4 項目形式名

項目形式名	説　明
SUMMARY	ジョブネットワークサマリー
SCHEDULE	ジョブ部品のスケジュール情報

■#TYPE=SUMMARYサマリーの出力は次のようなフォーマットになります。

```
#TYPE=SUMMARY
jnw,user,start,jnwid,njob,ncskip
値,...
```

- サマリーの出力のそれぞれの項目について説明します。

項目名	説　明
jnw	ジョブネットワーク名
user	ユーザ名
start	ジョブネットワークの予定開始時刻 Local timeで次のように表します。 “YYYY/MM/DD HH:mm:SS”
jnwid	JobCenterで管理されているID ジョブネットワーク名.YYYYMMDDHHmmSS （YYYYMMDDHHmmSSは投入日時をGMTで表したもの）
njob	このジョブネットワークで実行されるジョブ個数

	カレンダ分岐でスキップされる個数は含みません。
ncskip	カレンダ分岐でスキップされたジョブ個数

■#TYPE=SCHEDULE（ジョブ部品のスケジュール情報）

スケジュール情報の出力は次のようなフォーマットになります。

スケジュール行はサマリー行のあとに1行の空白をあけて、続けて出力されます。(initskipの項目は、JobCenter R12.5.4以降で出力されます)

```
#TYPE=SCHEDULE
index,name,parentjnw,inithold,initskip
SCHEDULE, 値
PREV, ジョブ名...
ALLPREV, ジョブ名...
```

1,2行目はタイトル行で1度だけ出力されます。

3,4,5行目は個々のジョブ部品についてそれぞれ出力されます。ただし、コマンドのオプションにより出力が抑止されることがあります。

3,4,5行目の最初の項目（SCHEDULE, PREV, ALLPREV）は固定です。これをキーにしてその行が何を記述しているか判断できます。

それぞれの項目について説明します。

- 3行目：SCHEDULE

項目名	説　明
index	固定の文字列で“SCHEDULE”が設定されます。
name	ジョブ部品名
parentjnw	ジョブ部品が属しているジョブネットワーク名です。サブジョブネットワークの場合は次のように記述されます。“rootJNW[:subJNW1...]:subJNWx”
inithold	事前保留属性の有無を表します。属性の設定がある場合は1、ない場合は0になります。
initskip	事前スキップ属性の有無を表します。属性の設定がある場合は1、ない場合は0になります。 この項目は、JobCenter R12.5.4以降で出力されます。

- 4行目：PREV

  そのジョブ部品の先行ジョブおよび待ち合わせ部品のうち直前のものだけを表示します。各カラムに先行する部品1つを表示します。

  - 最大50 カラム分の先行部品の情報を1 行に出力します。
  - 先行部品が50 を超える場合は、2 行に分けてPREV 行を出力します。

  部品を次のように表しています。

  - 単位ジョブまたはERP ジョブ：

    "J rootJNW[:subJNW...]/ ジョブ名"

  - 時刻待ち合わせ部品（待ち合わせ時刻 HH:MM）：

    "T YYYY/MM/DD HH:MM"

GMT　で「YYYY：西暦年、MM：月、DD：日、HH：時、MM：分」のように表します。月・日・時・分が1 桁のときは、その前に0 を付加してください。

- ファイル待ち合わせ部品

  "F ファイル名"

■ 5行目：ALLPREV

そのジョブ部品のすべての先行ジョブおよび待ち合わせ部品を表示します。

PREV行とほぼ同一ですが、すべての先行ジョブについて表示する点だけが異なります。

3.3. jnwop (control) ジョブネットワーク上の各部品を操作

■UNIX版

```
/usr/lib/nqs/gui/bin/jnwop control [-u $user] [-t $YYYYMMDDHHmmSS] [-tid $machine:$user:
$JNWname.YYYYMMDDHHmmSS(GMT).nstrk] [-oldest] $command $JNWname [$JOBname]
```

■Windows版

```
%InstallDirectory%\bin\jnwop control [-u $user] [-t $YYYYMMDDHHmmSS] [-tid $machine:$user:
$JNWname.YYYYMMDDHHmmSS(GMT).nstrk] [-oldest] $command $JNWname [$JOBname]
```

3.3.1. 機能説明

■実行中（アーカイブ待ちを含む）、または実行予定のジョブネットワークについて、それに存在する単位ジョブまたはジョブネットワーク、サブジョブネットワークの操作を行います。可能な操作は、次のとおりです。

保留、保留解除、スキップ、スキップ解除、再実行、強制停止、強制起動、エラーの場所から再実行、エラーの場所をSKIP、正常終了扱いに変更、トラッカの削除、トラッカのアーカイブ

■$commandには次の表の文字列を使用できます。

表3.5 $commandに使用できる文字列

文字列	内　容
hold	指定した部品を「保留」します。すでに保留されていた場合は無視します。
release	指定した部品の「保留」を解除します。すでに保留が解除されていた場合は無視します。
skip	指定した部品を「スキップ」します。すでにスキップされていた場合は無視します。
noskip	指定した部品の「スキップ」を解除します。すでにスキップが解除されていた場合は無視します。
rerun	指定した部品を「再実行」します。
estop	指定した部品を「強制停止」します。
forcestart	指定した部品を「強制起動」します。
delete	指定したトラッカ情報を「削除」します。
before	指定したトラッカがエラー停止状態のとき、エラー停止している単位ジョブから再実行します。フローの中でその単位ジョブ以降にあるものも再実行されます。
after	指定したトラッカがエラー停止状態のとき、エラー停止している単位ジョブをスキップして次の単位ジョブから再実行します。フローの中でその単位ジョブ以降にあるものも再実行されます。
statusdone	指定した部品が異常終了、警告終了のとき、その部品を「正常終了扱い」に変更します。
archive	指定したトラッカが正常終了、異常終了、警告終了のとき、トラッカをアーカイブします。 但し、トラッカの投入時刻から１分以上経過している必要があります。

■$JNWnameには操作対象となるジョブネットワーク名を指定します。対象の部品がサブジョブネットワークの場合、「:」でジョブネットワーク名を区切って階層を表現します。

（例）

"rootJNW:subJNW"

■$JOBnameには操作対象となる単位ジョブ名を指定します。

省略した場合、$JNWnameで指定したジョブネットワークまたはサブジョブネットワークが対象になります。

3.3.2. オプション

-u $user

操作対象とするジョブネットワークを所有しているユーザ名を指定します。指定しない場合、本コマンドを実行するユーザが所有するジョブネットワークが操作対象となります。

ジョブネットワーク投入時に使用される環境変数については、「3.3.4 注意事項」を参照してください。

本オプションを使用できるのはJobCenter管理者のみです。

-t $YYYYMMDDHHmmSS

対象となるトラッカを指定するために使用します。

ジョブネットワークの投入時刻はLocal timeで次のように表します（トラッカ名の一部で使用されているGMTとは異なります）。

形式は（YYYYMMDDHHmmSS）です。

省略した場合、一番最近に投入したトラッカが対象となります。

-tid $machine:$user:$JNWname.YYYYMMDDHHmmSS(GMT).nstrk

対象となるトラッカをGMTで指定するために使用します。

次のように指定します。

マシン名:ユーザ名:ジョブネットワーク名.投入時刻（YYYYMMDDHHmmSS（GMT））.nstrk

-oldest

部品の所属するジョブネットワークのトラッカのステータスが「実行中」または「予定(確定)」の中で、現在時刻の前後72時間以内で投入予定時刻が最も古いトラッカを操作対象とします。

投入予定時刻を変更した場合、変更前の投入予定時刻（トラッカIDの投入時刻）で操作対象を判断します。

本オプションは-tまたは-tidオプションと同時には指定できません。

3.3.3. 戻り値

戻り値	内　容
0	正常終了です。
0以外	異常終了です。エラー内容のメッセージを標準エラーに出力します。

3.3.4. 注意事項

■UNIXでnsumsmgr(またはroot)でjnwopコマンドを-uオプション付で使用すると、「Failed: mkdir () failure…」というメッセージが表示される場合がありますが、操作が完了したかどうかは戻り値によって判断してください。CL/Winにより一度でもnsumsmgr(またはroot)でログインすると、このメッセージは表示されなくなります。

■Windowsでjnwopコマンドを使用する場合は、管理者権限が必要です。

■hold/release, skip/noskip等を実行する際、操作が不能であれば無視します。

■実際には存在しないトラッカを-tまたは-tidで指定して操作を行おうとした場合、操作自体は失敗しますが、その指定したトラッカが「予定(確定)」状態で指定した日時で作成されますので注意してください。(作成された場合はdeleteで直ちに消去してください)

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITEにサイト名を設定する必要があります。(Windowsの場合NQS_SITEDBにサイトのデータベースパスの設定も必要)

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

■jnwopコマンドを実行することで「予定」状態のトラッカを「予定(確定)」状態にした場合(具体的には、$commandにstatusdone, archive以外を指定して実行した場合)、そのトラッカの使用する環境変数は以下の通りです。

- コマンド実行ユーザ自身が所有するジョブネットワークを対象として実行した場合

  コマンド実行時の環境変数がそのまま使用されます。特に環境変数
  TZ、LOGNAME、HOME、LANG、NQS_SITEの設定に注意して正しい値を設定してください。

- 他ユーザのジョブネットワークを対象として実行した場合

  ジョブネットワークがスケジュール投入される時と同じ環境変数が使用されます。

3.4. jnwsubmitcmd ジョブネットワークを投入

■UNIX版

```
/usr/lib/nqs/gui/bin/jnwsubmitcmd [-p $parameter] [-e $key=$value | -i $inputfile] [-c $comment] [[-b] -t $YYYYMMDDHHmmSS(localtime) | $HH:MM(localtime)] [-d] [-h] [-s] [-r $remotehostname] [-u $user] [-w|-k|-f] $JNWname
```

■Windows版

```
%InstallDirectory%\bin\jnwsubmitcmd [-p $parameter] [-e $key=$value | -i $inputfile] [-c $comment] [[-b] -t $YYYYMMDDHHmmSS(localtime) | $HH:MM(localtime)] [-d] [-h] [-s] [-r $remotehostname] [-u $user] [-w|-k|-f] $JNWname
```

3.4.1. 機能説明

■$JNWnameで指定したジョブネットワークを投入し、トラッカを生成します。$JNWnameには、コマンドを使用するユーザの環境で定義したジョブネットワークを指定します。

■正常にジョブネットワーク投入が完了した場合、次の形式で生成したトラッカ名が出力されます。

マシン名:ユーザ名:ジョブネットワーク名.YYYYMMDDHHmmSS.nstrk

YYYYMMDDHHmmSSは、GMTで表した投入日時です。

■エラーが発生した場合には、標準エラーにメッセージが出力されます。

3.4.2. オプション

-p $parameter

$parameterで指定された文字列が、単位ジョブ実行時の環境変数NSJNW_PARAMとして設定されます。

本オプションは、CL/Winの即時投入操作時のダイアログにおいて「環境変数」-「ジョブパラメータ(NSJNW_PARAM)を設定する」で入力できるパラメータと同等です。

-e $key=$value

単位ジョブ実行時の環境変数として、$key(環境変数名)=$value(環境変数値)を設定します。

複数の環境変数を設定したい場合、本オプションを複数回指定してください。

```
-e key1=value1 -e key2=value2 ...
```

値にスペースを含む文字列を指定したい場合、$key=$value全体をダブルクォーテーションで囲んでください。

```
-e "key1=a b c"
```

本オプションは、CL/Winの即時投入操作時のダイアログにおいて「環境変数」-「複数の環境変数を設定する」で入力できるパラメータと同等です。

環境変数名は最大51バイトで指定してください。

環境変数名には空白、タブ、特殊文字、および =#'" は使用できません。

環境変数値にはタブ、特殊文字、および #'" は使用できません。

本オプションは-iオプションと同時には指定できません。

-i $inputfile

単位ジョブ実行時の環境変数として、$inputfileで指定したファイル内のデータを設定します。$inputfileはkey(環境変数名)=value(環境変数値)の形式で、複数指定する場合は改行して指定してください。

```
key1=value1
key2=value2
...
```

マルチバイト文字を含む場合、ファイル内の文字コードは以下のようにしてください。

■UNIX版: JobCenterのセットアップ言語に合わせてください。

■Windows版: SJISで作成してください。

本オプションは、CL/Winの即時投入操作時のダイアログにおいて「環境変数」-「複数の環境変数を設定する」で入力できるパラメータと同等です。

環境変数名は最大51バイトで指定してください。

環境変数名には空白、タブ、特殊文字、および =#'" は使用できません。

環境変数値にはタブ、特殊文字、および #'" は使用できません。

ファイル中の1行は最大8191バイトで指定してください。

本オプションは-eオプションと同時には指定できません。

-c $comment

トラッカで表示されるコメントです。

CL/Winの即時投入操作時のダイアログで入力できるコメントと同等です。

最大長はUNIX版は128バイト、Windows版は40バイトです。

-t $YYYYMMDDHHmmSS(localtime) | $HH:MM(localtime)

■-t $YYYYMMDDHHmmSS

ジョブネットワークの投入時刻（YYYYMMDDHHmmSS Local time）を指定します。

CL/Winの即時投入操作時のダイアログで入力できる時刻と同等です。

■-t $HH:MM

ジョブネットワークの投入時刻（HH:MM Local time（時：分））を指定します。

年月日（YYYYMMDD）には、コマンド実行時のLocal timeの年月日が設定されます。

秒（SS）には、00秒が設定されます。

投入時刻の時（HH）には、0～23の整数が指定できます。

投入時刻の分（MM）には、0～59の整数が指定できます。

月・日・時・分・秒が1桁のときは、その前に0を自動で付加します。

-b

-tオプションで指定した投入時刻が現在時刻より過去でも、その投入時刻でのJNW投入を許可します。

本オプションを指定しないで投入時刻が現在時刻より過去の場合は、投入時刻が現在時刻に補正されてJNW投入されます。

-d

$JNWnameと同名のジョブネットワークについて、同一投入日の他のトラッカ（予定を含む）が存在した場合、投入を行いません。

トラッカ名の“YYYYMMDD”部分が同一の場合、同一投入日と判断します。

本オプションは日次ジョブの二重投入を防止するために使用します。

-h

ジョブネットワークの同時実行数制限の設定値以上のジョブネットワークが投入され待ち状態になっている場合に、$JNWnameで指定したジョブネットワークを最優先で実行します。

本オプションの指定をされたジョブネットワークが複数ある場合は、投入時刻順で実行します。

-s

$JNWnameと同名のジョブネットワークについて、同一投入時刻（YYYYMMDDHHmmSS）の他のトラッカ（予定を含む）が存在した場合には、時間をずらして1秒後に投入します。

■投入時刻の指定

環境変数JNW_SUBMITTIMEに投入時刻（YYYYMMDDHHmmSS（GMT））を指定できます。

-tオプションと同時に使用した場合、-tオプションの時刻が優先となります。この値は、トラッカ名に反映されます。

-r $remotehostname

リモートサーバ上に存在するJNWを投入したい場合に指定します。本オプションを指定する場合、後述する-uオプションを併せて指定する必要があります。本オプションを指定しない場合、ローカルサーバ上のJNWが対象となります。

本オプションを指定する際は、必ずNQS_SITE環境変数を設定しない状態で実行してください。NQS_SITE環境変数が設定されていると、存在しないパスへのアクセスに失敗する等のエラーにより実行できない場合があります。

本オプションは、Windows版の場合、JobCenter管理者のみが利用できます。

-u $user

投入するJNWを所有しているユーザ名を指定します。指定しない場合、jnwsubmitcmdコマンドを実行したユーザの所有するJNWを投入します。

JNW投入時に使用される環境変数は、以下の通りです。

■コマンド実行ユーザ自身が作成したJNWを投入する場合

コマンド実行時の環境変数がそのまま使用されます。

■他ユーザのJNWを投入する場合、または、-rによりリモートサーバ上のJNWを投入する場合

他ユーザまたはリモートサーバ上で、JNWがスケジュール投入される時と同じ環境変数が使用されます。

例えば、JNWが存在するサーバのOSにシステム環境変数を追加設定した直後に本オプションを指定してJNWを投入しても、追加設定した環境変数は反映されません。(スケジュール実行時はユーザのログイン環境ではないため)

そのユーザ環境に環境変数を反映させるためには、JNWの所有ユーザでCL/WinによりJNWが存在するサーバにログインして、最低1回スケジュールのダミールールの作成保存を行ってJobCenterに環境変数の変更内容を反映させてください。ダミールールやダミースケジュールは保存後に削除して構いません。

本オプションは、Windows版の場合、JobCenter管理者のみが利用できます。

-w

JNWパラメータに設定されている事前設定の内容を無視して、JNWを保留状態で投入します。

本オプションは、Windows版はR12.8.1から利用可能です。UNIX版はR12.7から利用可能です。

-k

JNWパラメータに設定されている事前設定の内容を無視して、JNWをスキップ状態で投入します。

本オプションは、Windows版はR12.8.1から利用可能です。UNIX版はR12.7から利用可能です。

-f

JNWパラメータに設定されている事前設定の内容を無視して、JNWを即時実行状態で投入します。

本オプションは、Windows版はR12.8.1から利用可能です。UNIX版はR12.7から利用可能です。

3.4.3. 戻り値

戻り値	内　容
0	正常にジョブネットワーク投入が完了しています。
1	ジョブネットワークの投入には失敗しましたが再度実行すれば成功する可能性があります。
3	オプションの指定ミスなどコマンドの使用方法に問題があります。
4	環境変数設定ファイルの読み込みエラーが発生しています。
その他	JobCenter内部に何らかの問題が発生しています。

3.4.4. 注意事項

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITEにサイト名を設定する必要があります。(Windowsの場合NQS_SITEDBにサイトのデータベースパスの設定も必要)

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

■本コマンド実行時の環境変数が、ジョブネットワーク実行時の環境変数として引き継がれます(-r、-uオプション指定時は除く)。特に環境変数TZ、LOGNAME、HOME、LANG、NQS_SITEの設定に注意して正しい値を設定してください。

■UNIX版の場合、コマンド実行時の環境変数LANGはJobCenterのセットアップ言語に合わせるようにしてください。Windows版の場合は、環境変数LANGではなくインストール時に指定したセットアップ言語が利用されるので、特に注意する必要はありません。

また、-rオプションによりリモートサーバ上のJNWを指定する場合、EUCとSJISの混在は可能ですが、一方がUNICODE環境でもう一方が異なる場合はエラーとなりますので注意してください。

■単位ジョブから本コマンドを使用してその他のジョブネットワークを起動する場合、本コマンドを実行する前に環境変数　STANDARD_ERRORとSTANDARD_OUTPUT　をクリアする必要があります。クリアしない場合、本コマンドで起動されたジョブネットワークの単位ジョブの出力結果、エラー出力結果がトラッカに記録されません。

■同一ユーザで同一ジョブネットワーク名を指定して本コマンドを複数同時に実行する、あるいは-tオプションで同じ投入時刻を指定して複数回実行したとしても、同一ジョブネットワーク名で複数のトラッカを同一時刻に起動する事は出来ません。タイミング的に最も早くに実行された本コマンドが有効になってトラッカが生成され、その他の本コマンドは実際には何もせず終了することになります。

そのような場合は -sオプションを指定するようにしてください。

3.5. nqsbg ジョブ内からジョブの終了時に始末されないプロセスを起動

■UNIX版

```
/usr/lib/nqs/nqsbg [-w] [ -n $subcommand ] $maincommand
```

■Windows版

(無し)

3.5.1. 機能説明

■nqsbgコマンドは、単位ジョブスクリプトから単位ジョブ終了時に上記シグナル（TERM/KILL）を受信しないバックグラウンドプロセスとして、$maincommandで指定したユーザコマンドを起動する機能を提供します。

JobCenterは通常、ジョブリクエストから起動したコマンドプロセスの終了を待ち合わせます。そのコマンドプロセスが終了してジョブリクエストの処理を終了する時に、コマンドプロセスから起動された同一プロセスグループに属する子プロセスに対してシグナル（TERM/KILL）を送信し、ジョブリクエストから起動されたプロセスがジョブの終了後に残らないよう制御します。

例えばメールを送信するプロセスをバックグラウンドで起動して戻るユーザコマンドや、デーモンプロセスのような常駐プロセスを単位ジョブスクリプトに記述して起動した場合、単位ジョブとしてトラッカ上正常に終了した時点で実際には起動したはずのバックグラウンドプロセスが終了させられてる、という状態を本コマンドにより回避できます。

■$maincommandに指定したコマンドをバックグラウンドプロセスとして起動します。

■起動されたプロセスはジョブから制御が切り離されるため、このプロセスの終了状態をJobCenterから監視することはできません。また、JobCenterを終了しても起動されたプロセスは終了しません。

■本コマンドはUNIX版のみの実装となります。(Windows版については、OSのstartコマンド等を利用してください)

3.5.2. オプション

(オプションなし：通常の使用方法)

nqsbgはサブプロセスを起動して直ちに終了します。

起動されたサブプロセスはさらに$maincommandで指定したユーザコマンドを実行します。そのため、nqsbgのプロセスIDと実際にバックグラウンドで起動されるプロセスのプロセスIDは異なります。

-w

サブプロセスを起動しません。

nqsbgは端末とプロセスグループを切り離したあと、実際にバックグラウンドで起動するサブプロセスを起動せずに、直接コマンドを起動します。この場合nqsbgのプロセスIDと起動されるコマンドのプロセスIDが同じになりますが、プロセス自体はバックグラウンドになりません。

-n $subcommand

$subcommandに指定した文字列を、起動コマンド（$maincommand）のargv[0]に指定して利用します。psなどで表示されるプロセス名を変更する必要がある場合に利用します。

3.5.3. 戻り値

戻り値	内　容
0	戻り値1以外の場合です。 ■$maincommandで指定したコマンドファイルが実際に存在せず起動できなかった場合でも、nqsbg自体の戻り値としては0となりますのでご注意ください。 ■$maincommand　で指定したコマンドが異常終了した場合でも、nqsbgは$maincommandを起動するとすぐプロセスグループを切り離すため、nqsbg自体の戻り値としては0となりますのでご注意ください。
1	プロセスのプロセスグループを切り離すことができません。 サブプロセス／コマンドプロセスが生成できません。

-wを指定して直接コマンドを起動した場合は、起動されたプログラムの終了コードを返します。

3.5.4. 注意事項

■nqsbgで起動されるプロセスについては、標準出力と標準エラー出力はいずれも/dev/nullにリダイレクトするように設定してください。

/dev/nullへのリダイレクトを設定していない場合、本コマンドを介して起動したプロセスの標準出力/標準エラー出力に書き込まれる内容はジョブのトラッカデータを構成するファイルに記録されます。　そのためトラッカデータの肥大や、データ参照時等にエラーを誘発する可能性が有ります。

3.6. jnwevtmgr イベントの確認と削除

■UNIX版

```
/usr/lib/nqs/gui/bin/jnwevtmgr {-l | -d $hostname:$eventid }
```

■Windows版

```
%InstallDirectory%\bin\jnwevtmgr {-l | -d $hostname:$eventid }
```

3.6.1. 機能説明

■イベント送受信部品について現在の状況を表示します。または保持されているイベントを削除します。「他人のトラッカの操作」の権限が必要なCL/Winでの操作と異なり、本コマンドは他ユーザのトラッカ中のイベントも表示または削除できます。

■イベント確認

イベント一覧の出力形式は次のようになります。

```
***イベントリスト名***

インデックス トラッカ名.YYYYMMDDHHmmSS 部品名 ホスト名 イベントID タイムアウト
```

各項目の詳細については、次の表3.6「イベントリスト名と内容」、表3.7「イベント一覧出力形式」のとおりです。

表3.6 イベントリスト名と内容

イベントリスト名	内　容
EventWaitList	受信待ちをしている受信部品一覧
EventSendList	受信確認で待ち合わせている送信部品一覧
EventHoldList	保持しているイベント一覧

表3.7 イベント一覧出力形式

名　称	イベント一覧出力形式
インデックス	インデックス用の数字が出力されます。
トラッカ名	部品が所属しているトラッカ名.時刻（GMT）が出力されます。保持しているイベント一覧の場合は送信を行った部品が所属しているトラッカ名が出力されます。
部品名	部品名です。保持しているイベント一覧の場合は送信を行った部品名が出力されます。
ホスト名	部品に指定してある送信元または送信先ホスト名です。
イベントID	部品に指定してあるイベントIDです。
タイムアウト	タイムアウト時刻です。無限の場合は「-」で表示されます。

jnwevtcmdで送信した部品名に関しては、トラッカ名にPIDjnwevtcmdのプロセス番号.jnwevtcmdを、実行した時刻（ローカルタイム）、部品名にはjnwevtcmdと表示されます。

■イベント削除

イベント削除実行後のメッセージ表示は次のとおりです。

表3.8 イベント削除実行後のメッセージ表示

メッセージ表示	内　容
Invalid eventhostname or eventide.	指定されたホスト名またはイベントIDが不正です。
Invalid eventide.	イベントIDが不正です。
Not found target event.	指定されたターゲットが見つかりませんでした。
Success.	正しく削除できました。

3.6.2. オプション

-l

イベント一覧を表示します。

-d $hostname:$eventid

保持されているイベントの送信元ホスト名、イベントIDを指定し削除を行います。

$hostname：イベント送信元

$eventid：イベントID

3.6.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.6.4. 注意事項

■イベントの削除ができるのは、イベントを保持しているマシン（イベント送信で送信先のマシン）上になります。

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITEにサイト名を設定する必要があります。(Windowsの場合NQS_SITEDBにサイトのデータベースパスの設定も必要)

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

3.7. jnwevtcmd イベントを送信

■UNIX版

```
/usr/lib/nqs/gui/bin/jnwevtcmd -h $hostname -i $eventid [ -m $msg ] [ -l {off | on { -t $YYYYMMDDHHmmSS
| -c $YYYYMMDDHHmmSS | -a [+]$HH:MM } }  ] [-s]
```

■Windows版

```
%InstallDirectory%\bin\jnwevtcmd -h $hostname -i $eventid [ -m $msg ] [ -l {off | on { -t
$YYYYMMDDHHmmSS | -c $YYYYMMDDHHmmSS | -a [+]$HH:MM } }  ] [-s]
```

3.7.1. 機能説明

■任意のユーザでジョブネットワークのイベント受信部品に対して任意のタイミングでイベントを送信できます。-sオプション指定時を除き、指定したイベントIDでイベントを待ち受ける全てのイベント待ち合わせ部品に対してイベントを送信できます。

■jnwevtcmdコマンドを実行したときのメッセージ表示内容は次のとおりです。

表3.9 jnwevtcmdコマンドを実行したときのメッセージ表示

表　示	内　容
FOUND	相手先で受信されました。
NOTFOUND	相手先で受信されませんでした。
FAILED	相手先でエラーが発生しました。
***** is too long	指定している*****パラメータが長すぎます。
Need parameter. (-* *****)	*****のパラメータが必要です。
Invalid Option	オプションの指定が間違っています。
Lack of Option	オプションが不足しています。
Invalid format	指定しているパラメータフォーマットが不正です。

3.7.2. オプション

-h $hostname

送信先ホスト名を指定します。

-i $eventid

イベントIDを指定します。

-m $msg

メッセージを指定します。

-l {off | on}

イベント保持の有無を指定します。

-t $YYYYMMDDHHmmSS | -c $YYYYMMDDHHmmSS | -a [+]$HH:MM

イベント保持がonの場合の保持期限を指定します。

■-t $YYYYMMDDHHmmSS

GMTで次のように表します。

YYYY：西暦年、MM：月、DD：日、HH：時、mm：分、SS：秒

月・日・時・分・秒が1桁のときは、その前に0を付加してください。

■-c $YYYYMMDDHHmmSS

Local timeで次のように表します。

YYYY：西暦年、MM：月、DD：日、HH：時、mm：分、SS：秒

月・日・時・分・秒が1桁のときは、その前に0を付加してください。

■-a [+]$HH:MM

絶対時刻、相対時刻を指定します。最大99:59まで指定可能です。

＋：相対時刻指定（コマンド発行した時刻を基準に計算します）

HH：時、MM：分

-s

同じイベントIDでイベント待ち状態となっている部品のうち、開始時刻が最も早い部品にのみイベントを通知します。

3.7.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。
2	socket通信を試みましたがconnectできませんでした。
255	通信異常終了です。

3.7.4. 注意事項

■イベント送信コマンドでは、受信待ち合わせを行うことはできません。

■同一ユーザまたは異なるユーザが同じ$eventidを使用して、同一マシンから同一送信先ホストに複数回イベントを送信した場合、送信先ホスト上では最後に送信されたイベントで上書きされます。

■クラスタサイトから本コマンドでイベントを送信する場合、事前に環境変数NQS_SITEにサイト名を設定する必要があります。(Windowsの場合NQS_SITEDBにサイトのデータベースパスの設定も必要)

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

3.8. jnwdelete ジョブネットワーク・スケジュールを削除

■UNIX版

```
/usr/lib/nqs/gui/bin/jnwdelete [-u $user] -j $jnw

/usr/lib/nqs/gui/bin/jnwdelete [-u $user] [-f] -g $jnwgroup

/usr/lib/nqs/gui/bin/jnwdelete [-u $user] -s $schedule
```

■Windows版

```
%InstallDirectory%\bin\jnwdelete [-u $user] -j $jnw

%InstallDirectory%\bin\jnwdelete [-u $user] [-f] -g $jnwgroup

%InstallDirectory%\bin\jnwdelete [-u $user] -s $schedule
```

3.8.1. 機能説明

■ジョブネットワークまたはスケジュールを削除するコマンドです。

■第一の形式ではジョブネットワークを、第二の形式ではジョブネットワークグループを、第三の形式ではスケジュールをそれぞれ削除することができます。また、-j,-g,-sはそれぞれ同時には指定できず、いずれかを指定する必要があります。

■jnwdeleteコマンドを実行したときのメッセージ表示内容は次のとおりです。

表3.10 jnwdeleteコマンドを実行したときのメッセージ表示

表　示	内　容
DELETE: user=$user, JNW=$jnw	指定したジョブネットワークの削除に成功しました。
DELETE: user=$user, Group=$jnwgroup	指定したジョブネットワークグループの削除に成功しました。
DELETE: user=$user, Schedule=$schedule	指定したスケジュールの削除に成功しました。(存在しないスケジュールを指定した場合にも出力されます)
Specified object not exists.	指定したジョブネットワークが存在しません。
LANG is not equal to the language used by JobCenter	環境変数LANGがJobCenterのセットアップ言語と一致していません。
cannot get the database path	環境変数NQS_SITEで指定しているサイト名が間違っています。
Permission denied.	対象ユーザの定義を削除する権限がありません。
invalid option	オプションの指定が間違っています。

メッセージの先頭に表示されるエラーコードは記載していません。

■本コマンドは削除対象が存在しない場合は成功とみします。

3.8.2. オプション

-u $user

削除したい定義をもつユーザ名を指定します。指定しない場合、コマンド実行ユーザのジョブネットワーク、スケジュールが対象になります。

-j $jnw

指定したジョブネットワークを削除します。

-g $jnwgroup

指定したジョブネットワークグループを削除します。グループ名は最上位グループから以下のように指定します。

（例）user1のGrp1中に存在するGrp1を通常削除したい場合

jnwdelete -u user1 -g .Grp1.Grp2

$jnwgroup指定時の先頭の「.」はRootグループを意味します。

-f

ジョブネットワークグループ削除時の強制削除/通常削除を指定します。-fを指定しない場合は通常削除、-fを指定した場合は強制削除になります。

-s $schedule

指定したスケジュールを削除します。

3.8.3. 戻り値

戻り値	内　容
0	正常終了
1	異常終了時に返却され、標準エラー出力にそのエラーが出力されます。
2	コマンドオプションが間違っているか、実行環境に問題がある場合に返却され、標準エラー出力にそのエラーが出力されます。

3.8.4. 注意事項

■本コマンドで削除を行うためには、コマンド実行ユーザに対して、JobCenterのパーミッション設定における「ジョブネットワークの作成/変更/削除」の権限が与えられている必要があります。パーミッション設定の詳細については<環境構築ガイド>の10章 「ユーザ権限（パーミッション設定）」 を参照してください。

■CL/Winで編集中のジョブネットワークは削除できませんが、スケジュールは削除可能になっていますので注意してください。

■環境変数LOGNAMEが本コマンドの実行ユーザと一致している必要があります。suコマンドでユーザ変更を行う場合にはsu -を使用してください。

■環境変数LANGをJobCenterのセットアップ言語と合わせる必要があります。

■クラスタサイトに対して実行する場合、事前に環境変数NQS_SITEにサイト名を設定する必要があります。

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

3.9. schctrl スケジュールの有効化/無効化

■UNIX版

```
/usr/lib/nqs/gui/bin/schctrl [-u $user] $subcommand { $schedule... | $group...  | all }
```

■Windows版

```
%InstallDirectory%\bin\schctrl [-u $user] $subcommand { $schedule... | $group... | all }
```

3.9.1. 機能説明

■本コマンドは、スケジュールの有効化/無効化及びスケジュールのグルーピング操作を行います。

■登録済みスケジュールの運用を開始させたり、稼動中のスケジュールを停止させることができます。また、スケジュールをグルーピングすることで有効化/無効化を一元的に管理ができます。

■グルーピングされたスケジュール及びグルーピングされたグループを操作する際には、コロンを区切り文字とした以下の形式で指定します。

指定対象	形　式
スケジュール指定	$group:$schedule 例）GROUP01:GROUP02:GROUP03:GROUP04:Schedule01
グループ指定	$group:$group 例）GORUP01:GROUP 一部サブコマンドでは、Rootグループを指定する指定子「:」が使用できます。

■スケジュールグループ名の最大指定文字数は40バイト、スケジュール名の最大指定文字数は128バイトとなります。グループは最大５階層まで作成可能であり、各グループには親子関係があります。

■親グループに対して行った操作は子グループに対しても影響します。なお、同一グループ内に同名の子グループを重複して作成することはできません。

3.9.2. オプション

-u $user

スケジュールを操作するユーザを指定します。

省略した場合、コマンド実行したユーザのスケジュールを操作します。

本オプション使用可能なユーザは次の通りです。

■UNIX版では、rootユーザのみとなります。

■Windows版では、JobCenter管理者のみとなります。

3.9.3. サブコマンド

すべてのコマンドは 1 行で入力する必要があります。

1. subcommand一覧

文字列	内　容
on sched	指定したスケジュールを有効化します。すでに有効化されていた場合は無視します。

on group	指定したグループに所属する全スケジュールを有効化します。すでに有効化されていた場合は無視します。
on all	全スケジュールを有効化します。すでに有効化されていた場合は無視します。
off sched	指定したスケジュールを無効化します。すでに無効化されていた場合は無視します。
off group	指定したグループに所属する全スケジュールを無効化します。すでに無効化されていた場合は無視します。
off all	全スケジュールを無効化します。すでに無効化されていた場合は無視します。
create	グループを作成します。
delete	グループを削除します。
rename	グループの名前を変更します。
regist	指定したスケジュールをグループに登録します。
release	指定したスケジュールをグループから除外します。
move sched	指定したスケジュールをグループに移動します。
move group	指定したグループをグループに移動します。
show sched	指定したスケジュールの有効/無効状態を表示します。
show group	指定したグループに所属する全スケジュール及び全グループをtree構造で表示します。 各スケジュールの有効/無効状態を表示します。
show all	全スケジュール及び全グループをtree構造で表示します。 各スケジュールの有効/無効状態を表示します。

2. コマンド形式

schctrl on sched $schedule...

$scheduleを有効化します。

$scheduleは複数指定することが可能です。（最大数：100）

schctrl on group $group...

$groupに所属する全スケジュールを有効化します。

$groupは複数指定することが可能です。（最大数：100）

$groupにRootグループ指定「:」が使用できます。

schctrl on all

ユーザの全スケジュールを有効化します。

schctrl off sched $schedule...

$scheduleを無効化します。

$scheduleは複数指定することが可能です。（最大数：100）

schctrl off group $group...

$groupに所属する全スケジュールを無効化します。

$groupは複数指定することが可能です。（最大数：100）

$groupにRootグループ指定「:」が使用できます。

schctrl off all

ユーザの全スケジュールを無効化します。

schctrl create $group...

新規グループを作成します。

同一グループ内に同名のグループを作成することはできません。グループは上限値である５階層を超えて作成することはできません。

$groupは複数指定することが可能です。（最大数：100）

schctrl delete $group...

グループを削除します。削除対象グループにスケジュールや子グループが所属していた場合、当該スケジュールは無所属となり子グループは削除されます。この際、無所属となったスケジュールは有効/無効状態を保ちます。

$groupは複数指定することが可能です。（最大数：100）

schctrl rename $group $group

変更対象$group の名前を 変更後$group に変更します。コマンドパラメータの末尾には変更後グループ名を指定してください。

同一グループ内に同名のグループが存在する場合は変更することはできません。

schctrl regist $schedule... $group

無所属スケジュール$scheduleを$groupに登録します。コマンドパラメータの末尾には登録先グループ名を指定してください。

$scheduleは複数指定することが可能です。（最大数：100）

schctrl release $schedule...

$scheduleを所属グループから除外します。

除外したスケジュールは有効/無効状態を保ったまま無所属になります。

$scheduleは複数指定することが可能です。（最大数：100）

schctrl move sched $schedule... $group

$scheduleを$groupへ移動します。コマンドパラメータの末尾には登録先グループを指定してください。

移動したスケジュールは有効/無効状態を保ったまま指定グループに所属します。

$scheduleは複数指定することが可能です。（最大数：100）

$groupにRootグループ指定「:」が使用できます。

schctrl move group $group... $group

$groupを移動先$groupへ移動します。コマンドパラメータの末尾には移動先グループ名を指定してください。なお、移動後にグループ階層が上限値（５階層）を超える場合には移動できません。

移動したグループは有効/無効状態を保ったまま移動先グループに所属します。

移動元$groupは複数指定することが可能です。（最大数：100）

移動先$groupにRootグループ指定「:」が使用できます。

schctrl show sched $schedule...

$scheduleの有効/無効状態を次の形式で表示します。

[SCHEDULE] $schedule：<on|off>

$scheduleは複数指定することが可能です。（最大数：100）

schctrl show group $group...

$groupに所属する全スケジュール及び全グループをtree構造で表示します。

各スケジュールの有効/無効状態を表示します。

$groupは複数指定することが可能です。（最大数：100）

$groupにRootグループ指定「:」が使用できます。

schctrl show all

全スケジュール及び全グループをtree構造で表示します。

各スケジュールの有効/無効状態を表示します。

3.9.4. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.9.5. 注意事項

■本コマンドでは特殊文字「!"#$%&'()*,/:;<=>?@[\]^`{|}~」を無効とみなします。

■スケジュールを無効化しても、すでにスケジュール投入されたトラッカは強制終了されず、処理を継続します。また、保留操作等によって、予定状態から予定確定状態になったトラッカに関連するスケジュールの無効化操作を行っても、予定確定状態のトラッカを削除しません。

■CL/Win上からスケジュールパラメータの変更を行った場合でも、本コマンドで設定したスケジュールの有効/無効状態は保持されます。

■スケジュールの有効/無効状態及びスケジュールのグルーピングは、CL/Winの［スケジュール一覧］における操作と同等です。

■本コマンドでスケジュールを有効化/無効化しても、CL/Winのトラッカ一覧表示の自動更新では状態変更が反映されません。CL/Winからスケジュールの状態変更を確認する際はトラッカ一覧表示で[最新表示(R)　F5]を実行してください。

■R13.1以降のバージョンでは、JobCenterが停止している状態で本コマンドを使用す ることはできません。

3.10. nqsstart デーモンプロセスを起動

■UNIX版

```
/usr/lib/nqs/nqsstart [-f] [$sitename]
```

■Windows版

（無し）

3.10.1. 機能説明

■JobCenterを起動します。なおJobCenter　デーモンの起動はrootアカウント（スーパーユーザ）しか行えません。本コマンドはUNIX版のみの実装となります。

■本コマンドは、JobCenter デーモン（/usr/lib/nqs/nqsdaemon）と、イベント送信等に関連するデーモンプロセスを起動します。

■本コマンドは、デーモンの起動設定ファイル（/usr/lib/nqs/rc/daemon.conf）を読み込み、デーモン起動におけるオプションを読み込みます。

3.10.2. オプション

$sitename

サイトモードで起動するローカルサイト名またはクラスタサイト名を指定します。

-f

JobCenter デーモン（/usr/lib/nqs/nqsdaemon）と、イベント送信等に関連するデーモンプロセスを起動します。

前回起動時にJobCenter　　が正常に終了しなかった、あるいは終了方法が正しく無かった状況では、nqsstart ［$sitename］ のコマンドがエラーとなる場合があります。その際は起動しようとするホスト、またはサイトでデーモンが起動していない事を確認した後、-fのオプションを付加して強制起動してください。

ホスト、またはサイトでデーモンが起動していないことの確認方法

1. ps –efコマンドでjnwxxx，nqsxxx，NQSxxxなどの名前でgrepしてJobCenterの各デーモンのPIDおよびPPIDを確認します。
2. cjclsコマンドでクラスタサイト上のnqsdaemon　のPIDを確認します。（なお、SHUT　と表示される場合は、cjclsコマンドの出力結果の同じ行に表示されたサイトにおいてJobCenter　が起動していないことを表しています。）
3. ホスト上のデーモンを確認する場合は、2の結果のnqsdaemonのPIDおよびPPIDが同nqsdaemonのPIDと一致するデーモン群を、1の結果より除いたものの存在を確認してください。サイト上のデーモンを確認する場合は、2の結果のnqsdaemonのPIDおよびPPIDが同nqsdaemonのPIDと一致するデーモン群の存在を1の結果から確認してください。

3.10.3. 戻り値

戻り値	内　容
0	正常終了です。

1	異常終了です。

3.10.4. 注意事項

■JobCenter　は通常システム立ち上げ時に起動されますが、システムの管理者が本コマンドを利用して起動させることも可能です。ただし、そのような場合、コマンドの標準出力をファイル等にリダイレクションしておかないと、コンソール、または端末に各デーモンの出力する情報が表示されるようになります。

■本コマンドを使用せずに、JobCenter　デーモンを直接起動した場合、依存関係のある必要な常駐プロセスの幾つかが起動されないため、一部の機能が使用出来なくなる場合があります。

■ローカルサイトの起動時は$site-nameは指定する必要はありません。またクラスタサイト起動については、通常の運用ではcjcpwを使用するようにしてください。

■nqsstartを起動する時点の、rootアカウントのumask値に注意してください。例えばumask=002が設定された状態でnqsstartでJobCenterを起動すると、JobCenter内部では処理を行う常駐プロセス他全てにそのumask値が引き継がれますので、JobCenterが直接管理するデータや作成ファイルのパーミッション、単位ジョブスクリプトから実行した任意のユーザ作成ファイルにもumask=002が適用されるなど、想定と異なる状態になる場合があります。

そのためumaskを厳密に管理したい場合は、単位ジョブスクリプトの先頭でumaskを明示的に設定する、あるいは単位ジョブパラメータの「スクリプト実行用シェルコマンド」に/bin/cshを明示的に指定して各ユーザの.cshrcにumaskを設定する、などによりコントロールする必要があります。

3.11. nqsstop デーモンプロセスを停止

■UNIX版

```
/usr/lib/nqs/nqsstop [$sitename]
```

■Windows版

（無し）

3.11.1. 機能説明

■JobCenterを停止します。なおJobCenterデーモンの停止はrootアカウント（スーパーユーザ）しか行えません。本コマンドはUNIX版のみの実装となります。

■本コマンドは、JobCenterデーモン及び関連するデーモンプロセスを停止します。

■JobCenterは、システムのshutdown処理により自動的に停止します。システム停止を行わず任意のタイミングでJobCenterを停止する場合に本コマンドを使用してください。

3.11.2. オプション

$site-name サイトモードで起動したデーモンを停止する場合に、起動時と同じローカルサイト名またはクラスタサイト名を指定します。

3.11.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.11.4. 注意事項

■本コマンドを使用せずにqmgrのshutdownコマンドやkillコマンドによりデーモンを停止した場合、一部のJobCenter関連プロセスが正常に終了しません。

そのため、JobCenterデーモン再起動の時点でそれらのプロセスが残っていた場合、正しく動作しなくなる場合があります。

■本コマンド実行後に「jnwengine: shutdown.」のメッセージが表示される時と表示されずに終了する時があります。

ターミナルにメッセージが表示されるまでの間に、メッセージ出力元プロセスがタイミングにより早めに終了してしまうと表示されない場合がありますが、内部処理に違いはなく、特に問題はありません。

3.12. nmapmgr ネットワークの構成管理

■UNIX版

```
/usr/bin/nmapmgr
```

■Windows版

```
%InstallDirectory%\bin\qcmd\nmapmgr
```

3.12.1. 機能説明

■nmapmgr コマンドは、JobCenter が使用するネットワークデータベースを構築・設定します。このネットワークデータベースは次の 6 つの基本的な要素を含みます。

表3.11 nmapmgrコマンドのデータベース構成要素

表　示	内　容
mid	JobCenter ネットワークを構成しているマシン間で一意になるように付けられたマシンID番号です。JobCenterはこのmidを特定のマシンを識別するために使用します。この値の最小値は1です。この値の最大値は2^{31} - 1です。
principal-name	JobCenterが所属するネットワーク中で一意なホスト名です。その名前でネットワークにアクセスします。
alias	ネットワーク内のマシンのprincipal-nameの別名です。aliasはローカルJobCenterホストにのみ理解されます。(ネットワーク上の通信相手としては解釈されません)。本来のマシン名ではなく業務上の役割で識別したい場合等に使用します。aliasは255文字以内の長さの文字列にすることができます。
uid	ユーザIDです。リモートホスト上のユーザIDをローカルホストの任意のユーザIDにマッピングできます。
gid	グループIDです。リモートホスト上のグループIDをローカルホストの任意のグループIDにマッピングできます。
type	サポートされているJobCenterのタイプです。現在以下のタイプが用意されています。 ■ nec UNIX、Linux上で稼動するJobCenter、またはNEC用に機能拡張されたNQS ■ necnt Windows OS上で稼働するJobCenter ■ cos COSMIC 版 NQS (連携するマシンがACOSの場合)

■本コマンドによりネットワーク構成を変更する操作は、UNIXの場合は、rootアカウント（スーパーユーザ）、Windowsの場合は、JobCenter管理者しか行えません。

■それ以外のユーザはshowやgetサブコマンドのみ実行できます。

■nmapmgrコマンドを実行すると、プロンプト「NMAPMGR>:」が表示されます。この状態で、以下に述べるサブコマンドが入力できます。シェルやコマンドプロンプトに戻るときにはexitサブコマンドまたはquitサブコマンドを使用します。

3.12.2. 注意事項

■クラスタサイトについて設定を行う場合、事前に環境変数NQS_SITEにサイト名を設定する必要があります。

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

3.12.3. サブコマンド

説明の中で大文字により示されている部分は、コマンドを入力する際の省略形を示しています。たとえば、CHange Name であれば ch n と省略できます。ただし、省略形c n ではCReate コマンドと区別できないためエラーとなります。

実際のコマンド記述は大文字、小文字のどちらでもかまいませんが、マシン名などは大文字小文字が区別されます。

サブコマンド文字列長は1行256バイト(UNIX)または254バイト(Windows)まで指定可能です。

すべてのコマンドは 1 行で入力する必要があり、行継続文字"\"は使用できません。

Add Mid $mid $principal-name

$mid と $principal-name（プリンシパル名）の組み合わせでネットワーク構成にマシン名の情報を追加します。

$principal-name は、その mid に対応するマシンを一意に識別するために (JobCenterセットアップの際に) 決まる名前です。

Add Name $alias $to-mid

$to-mid のマシンIDで識別されるマシンの $alias (別名)を追加します。

$alias は本サブコマンドでもCL/Winでも 1つの mid にいくつでも割り当てられますが、CL/Winからはプリンシパル名を残して別名のみ削除することはできません。

別名のみ削除するには Delete Name サブコマンドを使ってください。

Add Uid $from-mid $from-uid $to-uid

$from-mid のマシンIDで識別されるリモートマシン上の $from-uid で示されるユーザアカウントと、ローカルマシン上の $to-uid で示されるユーザアカウントのユーザマッピング設定を追加します。

CL/Winでのユーザマッピング操作は $from_mid で示すマシン(ジョブリクエストの転送元)に接続して行いますが、本サブコマンドを実行してユーザマッピングを設定する場合は、マッピング先のマシン上(ジョブリクエストの転送先)で行うことに注意してください。

リクエスト転送元のマシンに本サブコマンドでユーザマッピングを設定しても解釈されません。

また、転送元のユーザIDと転送先のユーザIDは n:1 のマッピング関係を持つことに注意してください。1:n のマッピング関係はジョブリクエストの実行ユーザが一意に決まらないため、設定できません。

CHange Name $mid $principal-name

$mid のマシンIDで識別されるマシンの $principal-name (プリンシパル名) を変更します。

CReate

ネットワーク構成データベースをクリアします。

本サブコマンドを実行すると、以下のような確認メッセージが表示されます。

```
Are you sure to initialize the Network mapping database? (y/[n]) :
```

ネットワーク構成データベースをクリアする場合はyまたはYを、キャンセルする場合はnまたはNを入力してください。

本サブコマンドを実行すると、自マシンを含めたすべてのネットワーク構成データベースをクリアします。したがって緊急時以外は使用しないでください。

データベースをクリアした場合、そのままではJobCenterの起動/停止ができなくなりますので注意してください。

もし誤ってクリアしてしまった場合は、Add Mid サブコマンドと SEt Local Name サブコマンドにより、クリア前に設定してあった自身のマシン名とマシンIDを自マシンに再登録・再認識させてください。その後 qmgr の set machine_group サブコマンドで自分自身のマシン名をマシングループに登録すれば、JobCenterの起動が可能になります。

Delete DEFUid $from-mid

$from-mid のマシンIDで識別されるリモートマシン上のユーザアカウントに対する、デフォルトのユーザマッピング設定を削除します。

SEt DEFUid サブコマンドも参照してください。

Delete Gid $from-mid $from-gid

$from-mid のマシンIDで識別されるリモートマシン上の $from-gid で示されるグループIDに関するマッピング関係を削除します。

Delete Mid $mid

構成データベースから $mid のマシンIDで識別されるマシン情報を削除します。

Add Mid サブコマンドも参照してください。

Delete Name $alias

構成データベースから $alias (別名)を削除します。

Add Name サブコマンドも参照してください。

Delete Uid $from-mid $from-uid

$from-mid のマシンIDで識別されるリモートマシン上の $from-uid で示されるユーザアカウントに関するユーザマッピング設定を削除します。

Add Uid サブコマンドも参照してください。

Exit

nmapmgr コマンドを終了します。

Get Gid $from-mid $from-gid

$from-mid のマシンIDで識別されるマシン上の $from-gid で示されるグループIDにマッピングしている、ローカルマシン上のグループ ID を表示します。

Get Mid $name

$name にプリンシパル名または別名を指定して、そのマシンに割り当てられているマシンIDを表示します。

Get Name $mid

$mid のマシンIDで識別されるマシンのプリンシパル名を表示します。

登録されているマシン名がプリンシパル名か別名か区別が付かない場合に、Get Mid サブコマンドと組み合わせてプリンシパル名を識別する際に使用します。

Get Uid $from-mid $from-uid

$from-mid のマシンIDで識別されるのマシン上の $from-uid で示されるユーザIDにマッピングしている、ローカルマシン上のユーザ ID を表示します。

Help

サブコマンド一覧を表示します。

Quit

nmapmgr コマンドを終了します。

SEt DEFUid $from-mid $def-uid

$from-mid のマシンIDで識別されるリモートマシン上のユーザアカウントに対する、デフォルトのユーザマッピング設定として$def-uidで示されるユーザIDを設定します。

SEt Local Name $localname

自マシンのJobCenterに設定されているプリンシパル名を $localname に変更します。

他マシンではなく自マシンのプリンシパル名を CHange Name サブコマンドで変更した場合は、必ず本サブコマンドにより自身のプリンシパル名を再登録してください。もし再登録しない場合、正常動作しなくなる可能性があります。

SEt Mail_address $mid $address

$mid のマシンIDで識別されるマシンのメールアドレスを $address に設定します。

SEt Type $mid $type

$mid のマシンIDで識別されるマシンに $type で指定するJobCenter のマシンタイプを設定します。(通常、CL/Winからマシンアイコンを追加する場合は、自動的に適切なマシンタイプが選択されます)

$type には次のいずれかを指定できます。

■nec

$midのマシンがUNIXの場合に指定します。

■necnt

$midのマシンがWindowsの場合に指定します。

■cos

$midのマシンがACOSの場合に指定します。

ACOSマシンと連携する場合はCL/Winからのマシン登録はできません。Add　Nameサブコマンドと本サブコマンドでマシン情報を登録してください。

SHow State [$name]

$name にプリンシパル名または別名を指定して、そのマシンのマッピング状態やマシンID、マシンタイプを表示します。

$name を省略した場合はすべてのマシンの情報を表示します。

3.13. qmgr 構成管理および運用管理

■UNIX版

```
/usr/bin/qmgr
```

■Windows版

```
%InstallDirectory%\bin\qcmd\qmgr
```

3.13.1. 機能説明

qmgrは、システム管理者とシステム操作員が、ローカルマシン上のJobCenterシステムのNQS関連の制御のために使用するプログラムです。

qmgrコマンドを起動すると、qmgrはプロンプト「Mgr:」を返し、その後に後述するサブコマンドを入力できます。 qmgrコマンドを終了するには exit コマンドを使います。

なおWindows版については、設定に関わらずセットアップ時にJobCenter管理者として指定したユーザしかqmgrコマンドを起動できません。

3.13.2. 注意事項

■クラスタサイトについて設定を行う場合、事前に環境変数NQS_SITEにサイト名を設定する必要があります。

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

3.13.3. 特権

qmgrサブシステムには特権のタイプが2種類用意されています。

■JobCenter管理者(show　managersサブコマンドで:m付きで表示されるユーザ)は、すべてのサブコマンドの使用が許されています。

■JobCenter操作員(show　managersサブコマンドでで:o付きで表示されるユーザ)は、キュー・フォーム・デバイス・目的地キュー・管理者の設定・作成・削除を除く一部の制限されたサブコマンドのみ使用できます。

以降の各サブコマンドの説明に、必要とされる特権を明記してあります。

なお特権のないユーザは、helpとshowサブコマンドのみ使用できます。

3.13.4. ページャ機能（UNIX版）

環境変数PAGERにページングを行うためのプログラムを指定できます。このとき、helpサブコマンドとshowサブコマンドの出力が、そのプログラムの標準入力として使用されます。たとえば、

```
PAGER=/usr/bin/more
export PAGER
```

のように設定することで、helpの出力がページングされて表示されます。

また、指定されたプログラムが実行できないときは、そのまま出力されます。

3.13.5. 資源制限

setおよびmodifyサブコマンドで指定する制限値の形式は以下のとおりです。

3.13.5.1. 時間制限

時間に関する制限は次の形式で指定します。

```
[[hours:]minutes:]seconds[.fraction]
```

無制限の場合は“unlimited”という文字列を指定します。

表3.12 時間制限の指定例

指定例	内　容
1234:58:21.29	1234時間58分21.29秒
59:01	59分1秒
12345	12345秒
121.1	121.1秒

3.13.5.2. サイズ制限

サイズに関する制限は次の形式で指定します。

```
integer[.fraction][units]
```

units に指定できる単位は次のとおりです。

単位記述	単　位
B	バイト
kb	キロバイト
mb	メガバイト
gb	ギガバイト

unitsを指定しなかった場合はバイトと解釈されます。無制限の場合は、“unlimited”という文字列を指定します。

制限値はマシンにとって都合のよい値に変換される場合があります。

表3.13 サイズ制限の指定例

指定例	内　容
1234	1234バイト
1234kb	1234キロバイト
1234.5gb	1234.5ギガバイト

3.13.5.3. ナイス値

ナイス値は直接数値を指定します。リクエストのすべての制限値がキューの対応する制限値を超えていない場合にのみ、そのリクエストはキューにつながれます。リクエストに無制限値がある場合は、キューの対応する制限値もまた無制限でなければなりません。リクエストに指定されていない制限値については、キューの対応する制限値がリクエストの制限値となります。

1度リクエストをキューにつないだ後は、キューの制限値が変更されても、リクエストの制限値は変更されません。

SVR4.2MP を採用している UNIX では、プロセスごとのメモリサイズ制限はプロセスごとのスタックセグメントサイズ制限より大きくなければなりません。スタックセグメント制限はデフォルトで 256MB です。メモリサイズ制限はそれよりも大きくする必要があります。

リクエストを実行するシェルも、資源制限を受けますので注意してください。

3.13.6. サブコマンド

説明の中で大文字により示されている部分は、サブコマンドを入力する際の省略形を示しています。たとえば、ADd DEStination と表記してあれば ad des と省略できます。ただし省略形 ad de では ADd DEVice サブコマンドと区別できないためエラーとなります。

実際のサブコマンド記述は大文字、小文字のどちらでもかまいませんが、キューやユーザ名などオブジェクトの名称は大文字小文字が区別されます。…は、その直前 [] 内の部分を任意個繰り返して指定できることを示しています。　サブコマンド文字列長は1行256バイト(UNIX)または254バイト(Windows)まで指定可能です。サブコマンドを複数行にわたって入力する場合は、改行の前に行継続文字"\"を入力します。

リクエストIDはリクエストの投入時にNQS内で一意に割り当てられるIDで、シーケンス番号と投入マシン名で付与されます。シーケンス番号部分だけを指定した場合、マシン名部分はローカルサイトとして解釈されます。

ABort Queue $queue [$seconds]

$queue に指定するキュー内で現在実行している、すべてのリクエストが以下のようにアボートされます。

まず、そのキューで現在実行中リクエストの各プロセスにSIGTERMシグナルが送られます。次に $seconds に指定する秒数が経過した後、SIGKILLシグナルが、指定されたキュー内で実行している各リクエストの残存プロセスに送られます。

$seconds には 0 から600までの値を指定できます。指定しなかった場合のデフォルトは 60 (秒)になります。

このコマンドによってアボートされるすべてのリクエストはキュー上から削除され、そのリクエストの結果ファイルが返されます。

> $queue にネットワークキューを指定する場合
>
> 指定するキュー上で　RUNNING中であるすべてのネットワークリクエストが削除されます。その際、削除されたネットワークリクエストが転送しようとしていた結果ファイルは、実行マシン上のリクエストの所有者のホームディレクトリに置かれます。

JobCenter 操作員特権が必要です。

ADd Queues = ($queue[,$queue]...) $complex

$queue で指定する 1つまたは複数のキューを $complex で指定するバッチキューコンプレックス (複合体) に加えます。

$complex に指定するバッチキューコンプレックスは Create Complex サブコマンドであらかじめ作成しておく必要があります。

JobCenter 管理者特権が必要です。

ADd DEStination = $destination $queue $position

ADd DEStination = ($destination[,$destination]...) $queue $position

$queue　で指定するパイプキューの有効な目的地(転送先)として　$destination　で指定された目的地を加えます。$destinationが複数ある場合、設定されたリストの先頭から順番に目的地への転送が試みられます。

$position　には、これから指定する目的地をすでにそのパイプキューに指定されている目的地の前に置くか後に置くかを次の形式で指定します。指定しない場合はすでにパイプキューに指定されている目的地リストの末尾に追加します。

■after $queue_name

$queue_nameで指定された目的地の後に新しい目的地を置きます。

■before $queue_name

$queue_nameで指定された目的地の前に新しい目的地を置きます。

■first

新しい目的地をすべての目的地の先頭に置きます。

■last

新しい目的地をすべての目的地の末尾に置きます。

JobCenter管理者特権が必要です。

ADd DEVice = $device $queue

ADd DEVice = ($device [,$device] ...) $queue

(R12.7よりこの機能はサポートしておりません)

$queue で指定するキューの資源として $device で指定するデバイスを加えます。 指定されたデバイスはすでに存在していなければなりません (Create DEVICE サブコマンド参照)。

JobCenter 管理者特権が必要です。

ADd Forms $form-name [$form-name]...

有効なフォームリストに $form-name で指定するフォームを加えます。

JobCenter 管理者特権が必要です。

ADd Groups = $group $queue

ADd Groups =($group[,$group]...) $queue

$group で指定するグループを、$queue で指定するキューアクセス許可リストに加えます。それ以降、そのグループに属するユーザは $queue のキューに対してリクエストを投入できるようになります。

グループの指定方法として次の2つの形式があります。

■$group-name

■[$group-id]

たとえば $group-name が “group1”、 $group-id が“2001”の場合、“group1”か “[2001]”のいずれかの形式で指定します。

キューアクセス制限機能は SEt NO_Access サブコマンドで有効、SEt UNrestricted_access サブコマンドで無効となります。一度無効化した場合、キューアクセス許可リストは初期化されます。

なおOSからグループ定義が削除されてもアクセス許可リストには同期しません。その場合は[$group-id] の形式を指定して DElete Groups サブコマンドで削除してください。

Windows の場合は、JobCenter利用者は JobCenterグループに参加必須のため、本機能でグループを設定してもアクセス許可リストには表示されません。

JobCenter 管理者特権が必要です。

ADd Managers $manager [$manager]...

特権を認められた JobCenter 管理者のリストに、$manager で指定するアカウントを加えます。

$managerの指定方法として次の2つの形式があります。

■$local_account_name:m または $local_account_name:o

■[$local_user_id]:m または [$local_user_id]:o

たとえば $ local_account_name が “user1”、 $local_user_id が“1001”の場合、“user1:m”または“[1001]:m”のいずれかの形式で指定します。

■アカウント名または[ユーザid]に:m を付与

すべての qmgr コマンドを使うことができるJobCenter 管理者特権についての追加操作を意味します。

■アカウント名または[ユーザid]に:o を付与

一部のコマンドを使うことができるいわゆるJobCenter 操作員特権についての追加操作を意味します。

特権についての詳細は上記「3.13.3 特権」の項を参照してください。

UNIXの場合、nsumsmgrとrootアカウント(スーパーユーザ) はあらかじめJobCenter管理者特権付きで登録されています。

Windowsの場合、JobCenter管理者があらかじめJobCenter管理者特権付きで登録されています。

JobCenter 管理者特権が必要です。

ADd Users = $user $queue

ADd Users =($user [,$user]...) $queue

$user で指定するユーザを $queue で指定するキューアクセス許可リストに加えます。それ以降、そのユーザは $queue のキューに対してリクエストを投入できるようになります。

$userの指定方法として次の2つの形式があります。

■$user-name

■[$user-id]

たとえば $user-name が“user1”、$user-id が“1001”の場合、“user1”か“[1001]”のいずれかの形式で指定します。

キューアクセス制限機能は SEt NO_Access サブコマンドで有効、SEt UNrestricted_access サブコマンドで無効となります。なお一度無効化した場合、キューアクセス許可リストは初期化されます。

Windowsの場合は、JobCenterグループに参加していて、かつ[サーバの環境設定]のユーザ一覧に登録されているユーザアカウントのみ、キューアクセス許可リストに加えることができます。

なおOSからユーザ定義が削除された場合、または Windowsの [サーバの環境設定] からユーザがクリアされた場合でも、アクセス許可リストには同期しません。その場合は[$user-id]の形式を指定してDElete Usersサブコマンドで削除してください。

JobCenter 管理者特権が必要です。

Create Batch_queue $queue PRiority=$p [PIpeonly] [Run_limit= $n]

$queue で指定する名前のバッチキューをキュープライオリティ $p で作成します。

$p は 0 が最も低く、63 が最も高い優先度となります。キュープライオリティについては<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

PIpeonly　を指定する場合、キューに登録されるリクエストはパイプキューから受け渡されるリクエストに限定されます。

$n には個々のバッチキュー内で同時に実行できるリクエストの数を指定します。Run_limitを指定しない場合のデフォルトは 1 です。

Run_limitで指定できる値の上限はデフォルトでUNIX版では 220、Windows版では 190 です。SEt RUn_limit サブコマンドでリクエスト同時実行数制限を設定している場合は、上限はその値と同じとなります。

JobCenter 管理者特権が必要です。

Create Complex =($queue[,$queue]...) $complex

$queue で指定するバッチキューの集合からなるキューコンプレックスを作成します。

キューコンプレックスとしてJobCenter がグループ化したバッチキューの組は、全体としての run_limit を持つことができます。SEt COMplex 系サブコマンドを参照してください。

JobCenter 管理者特権が必要です。

Create DEVICE $device FOrms=$forms FUIlname=$filename Server =($server)

(R12.7よりこの機能はサポートしておりません)

$forms で指定するフォームに属するJobCenter デバイスを定義し、 $server で指定するサーバと関係づけます。

$server にはプログラムバイナリの絶対パス名といくつかの引き数 (オプションなど) を指定します。

$filename は、デバイス (スペシャルファイル) の絶対パス名であり、たとえばUNIX版では /dev/device のように指定します。

JobCenter 管理者特権が必要です。

Create DEVICE_queue $queue PRiority=$p [Device=$device] [Device =($device [,$device]...)][PIpeonly]

(R12.7よりこの機能はサポートしておりません)

$queue で指定する名前のデバイスキューを $p で指定するキュープライオリティで作成します。

$p は 0 が最も低く、63 が最も高い優先度となります。キュープライオリティについては<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

PIpeonly　が指定する場合、このデバイスキューに登録されるリクエストは他のパイプキューから受け渡されるリクエストに限定されます。

$device にはこのキューをサービスする 1つもしくはそれ以上の JobCenter デバイスのリストを指定します。

JobCenter 管理者特権が必要です。

Create Network_queue $queue Destination=$dest-name Priority =$p [Server = ($server)] [Run_limit = $n]

$dest-name で指定するホストを転送先とするネットワークキュー $queue を $p で指定するキュープライオリティで作成します。(Windows版では未サポート)。

ネットワークキューはリクエスト実行結果ファイルの転送用のキューです。もし転送先ホストに対応するネットワークキューがなければ、デフォルト・ネットワークキュー (Default-NetQue) が用いられます(UNIXのみ、Windowsには存在しません)。

1つのホストについて複数のネットワークキューが作成できますが、使われるのはキュープライオリティの最も高いキューだけで、その他のキューは無視されます。 1つのネットワークキューに複数のホストを指定することはできません。また、キュー作成後のホストの変更はできません。

$dest-name の指定方法として次の2つの形式があります。

■$machine-name

■[$machine-id]

たとえば $machine-name が“machine1”、 $machine-id が“100”の場合、“machine1”か“[100]”のいずれかの形式で指定します。

$p は 0 が最も低く、63 が最も高い優先度となります。キュープライオリティについては<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

$server には、このキューで使用するサーバプログラム名を指定します。指定しなかった場合は SEt NETWORK Client サブコマンドで指定するサーバプログラムを用います。また、キュー作成後にサーバプログラムを変更するときには、SEt NETWORK_client サブコマンドを使用します。詳細は<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

$n には、個々のネットワークキューで同時に転送できるリクエスト数を指定します。Run_limitを指定しない場合のデフォルトは 1 です。

Run_limitで指定できる値の上限はデフォルトで 220 です。SEt RUn_limitサブコマンドでリクエスト同時実行数制限を設定している場合は、上限はその値と同じとなります。

JobCenter 管理者特権が必要です。

Create Pipe_queue $queue PRiority=$p SErver=($server) [Destination = $destination] | [Destination = ($destination[,$destination]...)] [PIpeonly] [Run_limit = $n][STaywait][Check]

$queue で指定する名前のパイプキューを Priority の$p に指定するキュープライオリティで作成し、それを $server で指定するサーバと関係づけます。

$p は 0 が最も低く、63 が最も高い優先度となります。キュープライオリティについては<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

$server には転送プログラムの絶対パス名とオプションなどを指定します(ただしR12.8以降のWindows版では指定不要)。指定方法の詳細は、以下を参照してください。

<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」

<NQS機能利用の手引き>の「6.7 負荷分散環境」

$destination には、このパイプキューから送信する 1つもしくはそれ以上の目的地キューのリストを指定します。

PIpeonly を指定する場合、このパイプキューに登録されるリクエストは他のパイプキューから受け渡されるリクエストに限定されます。

$n には、個々のパイプキューで同時に転送できるリクエスト数を指定します。Run_limitを指定しない場合のデフォルトは 1 です。

Run_limitで指定できる値の上限はデフォルトでUNIX版では 220、Windows版では 190 です。SEt RUn_limit サブコマンドでリクエスト同時実行数制限を設定している場合は、上限はその値と同じとなります。

STaywait を指定する場合は時間指定つき（qsub の -a オプション）のリクエストをパイプキュー上でウェイトさせます。

Check を指定する場合はチェック機能が働き、パイプキューに登録する前に転送できる目的地があるか調べます。ただし、この場合目的地キューはローカルマシン上のキューに限ります。目的地が見つからない場合、リクエストはキューイングされずエラーになります。

JobCenter 管理者特権が必要です。

DElete Complex $complex

$complexで指定するキューコンプレックスを削除します。

JobCenter 管理者特権が必要です。

DElete DEStination =$destination $queue

DElete DEStination =($destination[,$destination]...) $queue

$queue　で指定するパイプキューのリクエスト転送目的キューリストから、$destination　で指定する目的地キューを削除します。

ただし、指定されたキューから削除する目的地キューへ転送処理中のリクエストについては,、そのまま処理を続行します。

パイプキューのすべての目的地キューがこの方法で削除された場合は、パイプキューは停止状態と同様、リクエストをキューイングしたまま転送動作を行わない状態になります。

JobCenter 管理者特権が必要です。

DElete DEVICE $device

(R12.7よりこの機能はサポートしておりません)

$device　　で指定するデバイスを削除します。ただし、デバイスは使用不可状態でなければ削除できません(DIsable Device コマンド参照)。

JobCenter 管理者特権が必要です。

DElete DEVICE =$device $queue

DElete DEVICE =($device [,$device]...) $queue

(R12.7よりこの機能はサポートしておりません)

$queue で指定するデバイスキューの使用JobCenter デバイスリストから $device で指定するデバイスを削除します。指定された JobCenter デバイス上で実行中のリクエストについては、そのまま処理を続行します。

デバイスキューの使用JobCenter デバイスリストからすべてのエントリが削除されると、デバイスキューは停止状態と同様の状態になります。

JobCenter 管理者特権が必要です。

DElete Forms $form_name [$form_name]...

$form_name で指定するフォームを有効なフォームのリストから削除します。

JobCenter 管理者特権が必要です。

DElete Groups = $group $queue

DElete Groups =($group[,$group]...) $queue

$group で指定されたグループが、 $queue で指定されたキューアクセス許可リストから削除されます。グループの指定の方法は以下の2つの方法があります。

■$group-name

■[$group-id]

たとえば $group-name が“group1”、 $group-id が“2001”の場合、“group1”か “[2001]”のどちらかを指定します。

JobCenter 管理者特権が必要です。

DElete Managers $manager [$manager]...

$manager で指定するアカウントをJobCenter 管理者のリストから削除します。

$managerの指定方法として次の2つの形式があります。

■$local_account_name:m または $local_account_name:o

■[$local_user_id]:m または [$local_user_id]:o

たとえば $ local_account_name が "user1"、 $local_user_id が"1001"の場合、"user1:m"または"[1001]:m"のいずれかの形式で指定します。

■アカウント名または[ユーザid]に:m を付けた場合

すべての qmgr コマンドを使うことができるJobCenter 管理者特権についての削除操作を意味します。

■アカウント名または[ユーザid]に:o を付けた場合

一部のコマンドを使うことができるいわゆるJobCenter 操作員特権についての削除操作を意味します。

特権についての詳細は上記「3.13.3 特権」の項を参照してください。

UNIXの場合、nsumsmgrとrootアカウント(スーパーユーザ) はJobCenter管理者リストから削除することはできません。

Windowsの場合、JobCenter管理者アカウントは JobCenter管理者リストから削除することはできません。

JobCenter 管理者特権が必要です。

DElete Queue $queue

$queue で指定するキューを削除します。

キューを削除するには、そのキューにリクエストがキューイングされておらず、かつそのキューが使用不可状態 (DIsable Queue 参照) であることが必要です。

デフォルトネットワークキュー (DefaultNetQue) は、このサブコマンドで削除することはできません。(UNIXのみ、Windowsには存在しません)

削除するキューが他のパイプキューの転送先に指定されていた場合でも、その転送先リストには本サブコマンドの削除処理内容は反映されず、リストにそのまま残りますので注意してください。

JobCenter 管理者特権が必要です。

DElete Request $request_id

$request_id で指定するリクエストを削除します。 指定されたバッチリクエストが結果ファイル出力中であった場合、そのバッチリクエストを親とする、すべてのネットワークリクエストを削除します。その削除されたネットワークリクエストが転送しようとしていた結果ファイルは、リクエストを実行していたマシン上のリクエストの所有者のホームディレクトリに置かれます。

ただし本サブコマンドでネットワークリクエストを直接指定することはできません。

JobCenter 操作員特権が必要です。

DElete User = $user $queue

DElete User =($user[,$user]...) $queue

$user で指定するユーザを $queue で指定するキューのアクセス許可リストから削除します。 ユーザの指定方法として次の2つの形式があります。

■$user-name

■[$user-id]

たとえば $user-name が“user1”、 $user-id が“1001”の場合、“user1”か“[1001]”のいずれかの形式で指定します。

JobCenter 管理者特権が必要です。

DIsable Device $device

(R12.7よりこの機能はサポートしておりません)

$device で指定するデバイスを使用不可状態にします。

デバイスキューの使用 JobCenter デバイスリストのエントリのすべての JobCenter デバイスが使用不可状態になると、そのデバイスキューは停止状態と同様の状態になります。

JobCenter 操作員特権が必要です。

DIsable Group = $group

DIsable Group =($group[,$group]...)

$group で指定するグループを JobCenter 使用不可にします。

それ以降、そのグループに属するユーザはJobCenter のどのキューに対してもリクエストを投入できなくなります。

グループの指定方法として次の2つの形式があります。

■$group-name

■[$group-id]

たとえば $group-name が“group1”、 $group-id が“2001”の場合、“group1”か “[2001]”のいずれかの形式で指定します。

JobCenter 管理者特権が必要です。

DIsable Queue $queue

$queue で指定するキューを使用不可状態にします。

STOp Queueサブコマンドと異なり、使用不可状態のキューには新たなリクエストを投入できません。ただしそのキュー上ですでに実行状態にあるリクエストについては処理は続行されます。

指定するキューがすでに使用不可状態の場合は、何も起こりません。

JobCenter 操作員特権が必要です。

DIsable User = $user

DIsable User =($user[,$user]...)

$user で指定するユーザを JobCenter 使用不可にします。

それ以降、そのユーザは JobCenter のどのキューに対してもリクエストを投入できなくなります。

ユーザの指定方法として次の2つの形式があります。

■$user-name

■[$user-id]

たとえば $user-name が“user1”、 $user-id が“1001”の場合、“user1”か“[1001]”のいずれかの形式で指定します。

JobCenter 管理者特権が必要です。

ENable Device $device

(R12.7よりこの機能はサポートしておりません)

$device で指定するデバイスを使用可能状態にします。

JobCenter 操作員特権が必要です。

ENable Group = $group

ENable Group =($group[,$group]...)

$group で指定するグループの JobCenter 使用不可能状態を解除します。

JobCenter使用不可能状態を解除されたグループに属するユーザは、それ以降 JobCenter キューにリクエストを投入できるようになります。

グループの指定方法として次の2つの形式があります。

■$group-name

■[$group-id]

たとえば $group-name が“group1”、 $group-id が“2001”の場合、“group1”か“[2001]”のいずれかの形式で指定します。

JobCenter 管理者特権が必要です。

ENable Queue $queue

$queue で指定するキューを使用可能状態にします。

キューを使用可能状態にすると、そのキューにリクエストを投入できるようになります。

指定したキューがすでに使用可能状態の場合は、何も起こりません。

JobCenter 操作員特権が必要です。

ENable User = $user

ENable User =($user[,$user]...)

$user で指定するユーザのJobCenter 使用不可能状態を解除します。

JobCenter 使用不可能状態を解除されたユーザはそれ以降、JobCenter キューにリクエストを投入できるようになります。

ユーザの指定方法として次の2つの形式があります。

■$user-name

■[$user-id]

たとえば $user-name が“user1”、 $user-id が“1001”の場合、“user1”か“[1001]”のいずれかの形式で指定します。

JobCenter 管理者特権が必要です。

EXit

qmgr コマンドを終了します。

HElp [$command]

help 情報が得られます。

引き数なしで HElp コマンドを使用すると、使用できるコマンドの一覧が表示されます。

$command 引き数を付けて HElp コマンドを使用すると、指定されたコマンドについての情報が表示されます。

コマンドの指定はそれがユニークである限りは部分文字列を指定してもかまいません。

$command 引き数を詳細に指定すれば、受け取る情報もより詳しく表示されます。

HOld Request $request_id

$request_id で指定するリクエストをホールド状態にします。

このサブコマンドによってホールドされたリクエストは、qrls(1)コマンドではリリースできません。RELease Request でリリースしてください。

リクエストをホールド状態にしたままでJobCenter をシャットダウンした場合、その状態は次回立ち上げ時にも引き継がれます。

qrls(1)コマンドはUNIX版のみサポートしています。

JobCenter 操作員特権が必要です。

Lock Local_daemon

JobCenter NQSデーモンのテキスト・セグメントとデータ・セグメントをメモリにロックします。 plock(2) の PROCLOCK オプションを参照してください。

ロックを除去するには後述の Unlock Local_daemon を実行する必要があります。

JobCenter 操作員特権が必要です。

MODify Request DAta_limit = ($value) $request_id

$request_id によって指定するリクエストのプロセスごとのデータセグメントサイズ制限値を $value に変更します。上記「3.13.5 資源制限」の項を参照してください。

ただし指定するリクエストがすでに実行中の場合は変更できません。

バッチキュー上では、システムでプロセスごとのデータセグメントサイズの制限機能がサポートされている場合のみ変更が可能です。

JobCenter 操作員特権が必要です。

MODify Request MAIL_End $mode $request_id

$request_id によって指定するリクエストの実行終了時メール送信モードを $mode で指定する形にします。

ただし指定するリクエストがすでに実行中の場合は変更できません。

$mode の指定方法は次のとおりです。

■on

メールを送信する

■off

メールを送信しない

JobCenter 操作員特権が必要です。

MODify Request MAIL_Start $mode $request_id

$request_id によって指定するリクエストの実行開始時メール送信モードを $mode で指定する形にします。

ただし指定するリクエストがすでに実行中の場合は変更できません。

$mode の指定方法は次のとおりです。

■on

メールを送信する

■off

メールを送信しない

JobCenter 操作員特権が必要です。

MODify Request MAIL_User $user $request_id

$request_id によって指定するリクエストに関連するメールの送信相手を $user で指定するユーザに変更します。

ただし指定するリクエストがすでに実行中の場合は変更できません。

JobCenter 操作員特権が必要です。

MODify Request Nice_value $value $request_id

$request_id によって指定するリクエストの nice 制限値を $value に変更します。上記「3.13.5 資源制限」の項を参照してください。

ただし指定するリクエストがすでに実行中の場合は変更できません。

JobCenter 操作員特権が必要です。

MODify Request PPCOresize_limit = ($value) $request_id

$request_id によって指定するリクエストのプロセスごとのコアファイルサイズ制限値を $value に変更します。上記「3.13.5 資源制限」の項を参照してください。

ただし指定するリクエストがすでに実行中の場合は変更できません。

バッチキュー上では、システムでプロセスごとのコアファイルサイズの制限機能がサポートされている場合のみ変更が可能です。

JobCenter 操作員特権が必要です。

MODify Request PPCPu_limit = ($value) $request_id

$request_id によって指定するリクエストのプロセスごとの CPU 時間制限値を $valueに変更します。上記「3.13.5 資源制限」の項を参照してください。

ただし指定するリクエストがすでに実行中の場合は変更できません。

バッチキュー上では、システムでプロセスごとの CPU 時間制限機能がサポートされている場合のみ変更が可能です。

JobCenter 操作員特権が必要です。

MODify Request PPMemory_limit = ($value) $request_id

$request_id によって指定するリクエストのプロセスごとのメモリサイズ制限値を$value に変更します。上記「3.13.5 資源制限」の項を参照してください。

ただし指定するリクエストがすでに実行中の場合は変更できません。

システムでプロセスごとのメモリサイズの制限機能がサポートされている場合のみ変更が可能です。

JobCenter 操作員特権が必要です。

MODify Request PPPermfile_limit = ($value) $request_id

$request_id によって指定するリクエストのプロセスごとの永久ファイルサイズ制限値を $value に変更します。上記「3.13.5 資源制限」の項を参照してください。

ただし指定するリクエストがすでに実行中の場合は変更できません。

システムでプロセスごとの永久ファイルサイズの制限機能がサポートされている場合のみ変更が可能です。

JobCenter 操作員特権が必要です。

MODify Request PRIority $priority $request_id

$request_id によって指定するリクエストのリクエストプライオリティを $priority に変更します。

リクエストプライオリティの変更により、キュー内のリクエストの処理順序を変更できます。

ただし指定するリクエストがすでに実行中の場合は変更できません。

JobCenter 操作員特権が必要です。

MODify Request Restart $mode $request_id

$request_id によって指定するリクエストの再実行可否モードを $mode で指定するモードに変更します。

ただし指定するリクエストがすでに実行中の場合は変更できません。

$mode の指定方法は次のとおりです。

■on

再実行可

■off

再実行不可

JobCenter 操作員特権が必要です。

MODify Request SHell = ($shellname) $request_id

$request_id によって指定するリクエストのリクエスト実行シェルを $shellnameに変更します。

ただし指定するリクエストがすでに実行中の場合は変更できません。

UNIXの場合は “/bin/sh” や “/bin/csh” 等のOSで使用可能なシェルプログラムを絶対パスで指定します。(なおユーザプログラムを指定した場合の正常動作は保証しておりません)

Windowsの場合はcmd.exeの絶対パス指定以外の実行プログラム指定はサポートしておりません。

JobCenter 操作員特権が必要です。

MODify Request STAck_limit = ($value) $request_id

$request_id によって指定するリクエストのスタックセグメントサイズ制限値を $valueに変更します。上記「3.13.5 資源制限」の項を参照してください。

ただし指定するリクエストがすでに実行中の場合は変更できません。

JobCenter 操作員特権が必要です。

MODify Request STDERR $filename $request_id

$request_id によって指定するリクエストの標準エラー出力結果ファイルを $filename で指定するファイルに変更します。

ただし指定するリクエストがすでに実行中の場合は変更できません。

JobCenter 操作員特権が必要です。

MODify Request STDERR_Spool $mode $request_id

$request_id によって指定するリクエストの標準エラー出力結果ファイルの転送モードを $mode で指定するモードに変更します。

ただし指定するリクエストがすでに実行中の場合は変更できません。

$mode の指定方法は次のとおりです。

■on

スプールモード

■off

非スプールモード

JobCenter 操作員特権が必要です。

MODify Request STDOUT $filename $request_id

$request_id によって指定するリクエストの標準出力結果ファイルを $filename で指定するファイルに変更します。

ただし指定するリクエストがすでに実行中の場合は変更できません。

JobCenter 操作員特権が必要です。

MODify Request STDOUT_Spool $mode $request_id

$request_id によって指定するリクエストの標準出力結果ファイルの転送モードを $mode で指定するモードに変更します。

ただし指定するリクエストがすでに実行中の場合は変更できません。

$mode の指定方法は次のとおりです。

■on

スプールモード

■off

非スプールモード

JobCenter 操作員特権が必要です。

MOVe Queue $queue1 $queue2

$queue1 で指定するキューに現在あるすべてのリクエストを $queue2 で指定するキューに移動します。

ただしすでに実行中のリクエストは移動できません。

JobCenter 操作員特権が必要です。

MOVe Request $request_id [$request_id ...] $queue

$request_id で指定するリクエストを $queue で指定するキューへ移動します。

ただしすでに実行中のリクエストは移動できません。

本サブコマンドでは、複数の $request_id を指定する場合は空白で区切ります。

JobCenter 操作員特権が必要です。

Purge Queue $queue

$queueで指定するキュー内のすべてのキューイング状態のリクエストを削除します。

ただしすでに実行中のリクエストは削除されません。

JobCenter 操作員特権が必要です。

RELease Request $request_id

$request_id で指定するリクエストのホールド状態を解除します。

ホールド状態以外のリクエストを指定すると警告が発せられますが、リクエストの状態には影響しません。

JobCenter 操作員特権が必要です。

REMove Queue =($queue[,$queue]...) $complex

$queue で指定するキューを $complex で指定するキューコンプレックスから除去します。

JobCenter 管理者特権が必要です。

RESEt Log_file

JobCenterログファイルを初期化します。

それまでのログファイルは、SEt LOG_file サブコマンドでNO_Backup が指定されていない限り、ログファイル名に .bak を付与して切り替えます。

JobCenter 管理者特権が必要です。

RESEt USERTOKen

初回ジョブ実行時にキャッシュ保存していたユーザトークンをクリアします。

クリア処理後には、次回初めてジョブを実行した時点でユーザトークンを再取得し、キャッシュ保存します。

本サブコマンドはWindows版のみサポートしています。

JobCenter 管理者特権が必要です。

RESTart Request $request_id

$request_id で指定する実行中リクエストの実行を中止し、キューに再登録します。

ただし再実行不可の属性が指定されたリクエストは再登録されません。

通常は他の実行待ちリクエストの最後に登録されますが、リクエストプライオリティが他よりも高い場合は、それに応じた優先的な位置に登録されます。

再度リクエストの実行を開始する際は、スクリプトの最初から実行しなおされます。

JobCenter 操作員特権が必要です。

RESUme Request $request_id

$request_id によって指定するリクエストの実行を再開します。

指定できるのは SUSPEND 状態のリクエストに限ります。

JobCenter 操作員特権が必要です。

SEt ALl_queue Intra_queue_scheduling_type $type

システム内のすべてのバッチキューのキュー内リクエストスケジューリング方式を $typeで指定する方式に設定します。

$typeの指定方法は次の通りです。

■type0

従来のスケジューリング方式で、同一リクエストプライオリティをもつリクエスト間では投入順序により実行順序が決められます。

■type1

同一リクエストプライオリティをもつリクエスト間では、投入順序に関係なく、それぞれのユーザのリクエストを順番に実行します。1人のユーザはあらかじめ連続して実行できるリクエストの数が決められており、それ以上リクエストを投入していた場合は、ほかのユーザのリクエストが実行されるまで順番待ちをします。

これにより、1人のユーザが投入したリクエストだけが集中的に実行されることを防ぎます。1人のユーザが連続して実行できるリクエストの数は、set continuous_scheduling_numberサブコマンドで設定されます。

JobCenter 管理者特権が必要です。

SEt CHeck $queue

$queue で指定するパイプキューの事前チェック機能を設定します。

JobCenter 管理者特権が必要です。

SEt COMplex Group_run_limit = $run_limit $complex

1つのグループについて、$complex で指定するキューコンプレックスで同時に実行できるリクエストの数を設定します。

UNIX版のみサポートしています。

このサブコマンドでキューコンプレックスに制限を設定すると、そのキューコンプレックス内では、1つのグループに属する各ユーザのリクエストの合計が $run_limit で指定する数を超えて同時に実行されることがなくなります。

もし $run_limit で指定された数を超えてリクエストを起動しようとした場合、そのリクエストは queued 状態にされます。

$run_limit には 1 から 上限値 までの値を指定できます。上限値はデフォルトで 220 です。SEt RUn_limit サブコマンドでリクエスト同時実行数制限を設定している場合は、上限値はその値と同じとなります。

$run_limitのデフォルトは 1 です。

JobCenter 操作員特権が必要です。

SEt COMplex Run_limit = $run_limit $complex

$complex で指定する JobCenter キューコンプレックスの同時実行数制限を設定します。

$run_limit はキューコンプレックス内で同時に実行することが許されるリクエストの最大数を指定します。

$run_limit には 1 から 上限値 までの値を指定できます。上限値はデフォルトでUNIX版では 220、Windows版では190 です。SEt RUn_limitサブコマンドでリクエスト同時実行数制限を設定している場合は、上限値はその値と同じとなります。

JobCenter 操作員特権が必要です。

SEt COMplex User_run_limit = $run_limit $complex

1人のユーザが　$complex　で指定するキューコンプレックスで同時に実行できるリクエストの数を設定します。

UNIX版のみサポートしています。

このサブコマンドでキューコンプレックスに制限を設定すると、そのキューコンプレックス内では、1人のユーザのリクエスト数が $run_limit で指定する数を超えて同時に実行されることがなくなります。

もし $run_limit で指定された数を超えてリクエストを起動しようとした場合、そのリクエストは queued 状態にされます。

$run_limit には 1 から 上限値 までの値を指定できます。上限値はデフォルトで 220 です。SEt RUn_limit サブコマンドでリクエスト同時実行数制限を設定している場合は、上限値はその値と同じとなります。

$run_limitのデフォルトは 1 です。

JobCenter 操作員特権が必要です。

SEt CONtinuous_scheduling_number $number $queue

$queue で指定するキューにおいて、1人のユーザが連続して実行できるリクエストの数を $number に指定します。

$numberには 0 から 50 までの値を指定できます。デフォルトは 0 (無制限) です。

本機能は、バッチキューに対してのみ有効です。

JobCenter 管理者特権が必要です。また、Windows版では利用できません。

SEt CORefile_limit = ($limit) $queue

$queue で指定するバッチキューにプロセスごとのコアファイル制御値を設定します。

この値は投入されるリクエストに設定されたプロセスごとのコアファイルサイズ制御値と比較され、その値がキューに設定された値より大きい場合は、リクエストの登録が拒否されます。

またリクエストにプロセスごとのコアファイル制限値が特に設定されていない場合は、キューに設定した値をリクエストの制限値として使用します。

すでにキューに登録されているリクエストには、新しく設定される値は何も影響を与えません。

$limit の構文については、上記「3.13.5 資源制限」の項を参照してください。

本サブコマンドはシステムでプロセスごとのコアファイル制御値の設定がサポートされている場合のみ有効です。

JobCenter 管理者特権が必要です。

SEt CUStomjob Batch_queue $queue

$queueで指定するバッチキューにカスタムキュー属性を設定します。カスタムキュー属性は、拡張カスタムジョブを実行するバッチキューに設定する必要があります。

JobCenter管理者特権が必要です。

拡張カスタムジョブ部品をカスタムキュー属性の付与されていないバッチキューに投入した場合、カスタムジョブの実行はエラーになります。

カスタムキュー属性を付与したバッチキューに、通常のカスタムジョブ部品および単位ジョブ部品を実行することはできません。これらの部品をカスタムキューに投入した場合、カスタムキューは[STOPPED]状態になります。

SEt DAta_limit = ($limit) $queue

$queue で指定するバッチキューにプロセスごとのデータセグメントサイズ制御値を設定します。

この値は投入されるリクエストに設定されたプロセスごとのデータセグメントサイズ制御値と比較され、その値がキューに設定された値より大きい場合は、リクエストの登録が拒否されます。

またリクエストにプロセスごとのデータセグメント制限値が特に設定されていない場合は、キューに設定した値をリクエストの制限値として使用します。

すでにキューに登録されているリクエストには、新しく設定される値は何も影響を与えません。

$limit の構文については、上記「3.13.5 資源制限」の項を参照してください。

本サブコマンドはシステムでプロセスごとのデータセグメントサイズ制御値の設定がサポートされている場合のみ有効です。

JobCenter 管理者特権が必要です。

SEt DEBug $level

デバッグレベルを設定します。 $level には次の値が指定できます。

■0

デバックなし

■1

最小デバックレベル

■2以上

最大デバックレベル

JobCenter管理者特権が必要です。

SEt DEFault Batch_request Priority $priority

バッチリクエストの起動順序を決める既定リクエストプライオリティを変更します。既定リクエストプライオリティを変更しない場合のデフォルトは31です。

ユーザがqsub(1)　コマンドでリクエストプライオリティ指定を省略した場合の既定値として使用される値です。

なお、qsub(1)コマンドはUNIX版のみサポートしています。

$priorityは 0 が最も低く、63 が最も高い優先度となります。キュープライオリティについては<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

JobCenter 管理者特権が必要です。

SEt DEFault Batch_request Queue $queue

$queue で指定するキューを既定バッチキューとして設定します。 ユーザがqsub(1) コマンドで投入キュー名を省略した場合の既定値として使用します。

デフォルトはNONEです。

なお、qsub(1) コマンドはUNIX版のみサポートしています。

JobCenter 管理者特権が必要です。

SEt DEFault DEStination_retry Time $retry_time_in_seconds

1回のリクエスト転送のためのリトライ時間$retry_time_in_secondsを定義します。

JobCenterは1回のリクエスト転送において、このリトライ時間の秒数が経過するまでリクエスト転送先とのコネクション開設を再試行します。

リトライ時間が経過すると、この転送に失敗したとみなして Destination_retry Waitの時間待ち合わせたのち、再度転送を試みます。

$retry_time_in_secondsには 0 から 100000000 までの値を指定します。デフォルトは 16 (秒)です。

JobCenter 管理者特権が必要です。

SEt DEFault DEStination_retry Wait $retry_interval_in_seconds

リクエスト転送のための待ち時間を定義します。

リクエスト転送先がデータを受け取れなかった場合、この待ち時間の分だけ待ち合わせてからリトライを試みます。

$retry_interval_in_secondsには 1 から 10000000 までの値を指定します。デフォルトは 300 (秒) です。

JobCenter 管理者特権が必要です。

SEt DEFault DEVice_request Priority $priority

(R12.7よりこの機能はサポートしておりません)

デバイスリクエストの既定リクエストプライオリティを設定します。これは qpr(1) コマンドでリクエストプライオリティを省略したときに使用される値です。

なお、qpr (1) コマンドはUNIX版のみサポートしています。

JobCenter 管理者特権が必要です。

SEt DEFault Expire Time $expire_time_in_seconds

リクエストの実行終了後に、そのリクエストの状態情報をNQSデータベース内に保持しておく時間を秒単位で指定します。

$expire_time_in_secondsには 0 から 100000000 までの値を指定できます。デフォルトは 259200 (秒)です。

JobCenter 管理者特権が必要です。

SEt DEFault Network_retry TIME $retry_time_in_seconds

リクエスト転送および実行結果のネットワークリクエスト返却以外のコネクション開設のための、最大リトライ間隔を定義します。

JobCenter はこの値と等しくなるまでリトライ間隔を1→2→4→…と増やしながらコネクションの開設のリトライを行います。

$retry_time_in_secondsには 0 から 100000000 までの値を指定できます。デフォルトは 16 (秒)です。

JobCenter 管理者特権が必要です。

SEt DEFault Network_retry TIME_Out $time_out_time_in_seconds

ネットワークのコネクション先の応答がない場合に、 socket ストリームの切断と見なすまでの時間を設定します。

$time_out_time_in_secondsには 1 から 100000 までの値を指定できます。デフォルトは 300 (秒)です。

JobCenter 管理者特権が必要です。

SEt DEFault Network_retry Wait $retry_interval_in_seconds

リクエスト転送および実行結果のネットワークリクエスト返却以外のコネクションのための待ち時間を定義します。ネットワークに障害が起こった場合、 この待ち時間の分だけ待ち合わせてからリトライを試みます。

$retry_interval_in_secondsには 0 から 10000000 までの値を指定できます。0 が指定された場合、リトライは行いません。デフォルトは0 (秒)です。

JobCenter 管理者特権が必要です。

SEt DEFault PRINT_Request Forms $form-name

(R12.7よりこの機能はサポートしておりません)

$form-nameで指定するフォームを既定出力フォームとして設定します。これはqpr(1) コマンドでフォームパラメータを省略したときに使用されるフォームです。

デフォルトは(設定なし)です。

なお、qpr (1) コマンドはUNIX版のみサポートしています。

JobCenter 管理者特権が必要です。

SEt DEFault PRINT_Request Queue $queue

(R12.7よりこの機能はサポートしておりません)

$queue で指定するキューを既定出力キューとして設定します。これは qpr(1) コマンドでキューパラメータを省略したときに使用されるキューです。

デフォルトはNONEです。

なお、qpr (1) コマンドはUNIX版のみサポートしています。

JobCenter 管理者特権が必要です。

SEt DEFault STAGe_retry Time $retry_time_in_seconds

ネットワークリクエストが存在できる期間を秒単位で設定します。

$retry_time_in_secondsには 0 から 100000000 までの値を指定できます。デフォルトは 259200 (秒)です。

ネットワークリクエストは、親であるバッチリクエストが結果ファイル出力状態 (EXITING) になってから、このサブコマンドで指定された時間が経過するまでの間リトライを繰り返します。

存在期間を過ぎると、もしネットワークリクエストがリトライ可能な原因で結果ファイル出力に失敗してもリトライされません。その場合は結果ファイルを実行マシン上のリクエスト実行ユーザのホームディレクトリ (UNIX) または %InstallDirectory%\spool\work (Windows)に保存し、ユーザにメールで知らせます。

JobCenter 管理者特権が必要です。

SEt DEStination = $destination $queue

SEt DEStination = ($destination[,$destination]...) $queue

$destination で指定する 1つまたはそれ以上の目的地キューを $queue で指定するパイプキューに関連づけます。

$destinationの数そのものに制限はありませんが、サブコマンド1行のバイト長に制限がありますので、それを超えた場合の目的地キューの追加はADd DEStinationサブコマンドで行ってください。

JobCenter 管理者特権が必要です。

SEt DEVICE = $device $queue

SEt DEVICE =($device [,$device]...) $queue

(R12.7よりこの機能はサポートしておりません)

$queue で指定するキューに $device で指定する 1つもしくはそれ以上のデバイスを関連づけます。

JobCenter 管理者特権が必要です。

SEt DEFault STAGe_retry Wait $retry_interval_in_seconds

ネットワークリクエストのリトライの間隔を秒単位で設定します。

$retry_interval_in_seconds には 0 から 10000000 までの値を指定できます。デフォルトは 300 (秒)です。

ネットワークリクエストがリトライ可能な理由で結果ファイル出力に失敗した場合、このサブコマンドで指定する時間だけ待ってからリトライを行います。この間リクエストはリトライ待ち状態 (WAITING) になります。また、このリクエストが投入されているネットワークキューは次回のリトライが 始まるまで新しいネットワークリクエストのスケジューリングを行いません。

JobCenter 管理者特権が必要です。

SEt DEVICE_server = ($server) $device

(R12.7よりこの機能はサポートしておりません)

$device で指定するデバイスと $server で指定するサーバを関連づけます。

$server には、必要な引き数を後に付けたサーバプログラムの絶対パス名を指定します。詳細は<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

JobCenter 管理者特権が必要です。

SEt ERRor_log [Size=($logfile_size)] [Backup=$backup_number]

NQS関連のエラーログの設定を変更します。

■$logfile_size

エラーログのファイルサイズを指定します。指定可能な範囲は65536(64kb)～4194304(4mb)です。デフォルトの値は1048576(1mb)です。

単位を省略した場合はbyteとなりますが、kb、mbの単位を指定することもできます。

4MBを指定する場合の指定例: size=(4mb)

■$backup_number

エラーログのバックアップ世代数を指定します。指定可能な範囲は1～10です。デフォルトの値は10です。

JobCenter管理者特権が必要です。

SEt EVent_spool Size =($size)

NQSqwatch を使用して待ち合わせる内部イベントの最大スプーリングサイズを $size で指定する値に設定します。

UNIX版のみサポートしています。

イベントのスプーリングファイルは、イベントのエントリごとに作成されます。

本設定以前に作成されたイベント通知用ファイルのサイズは、このサブコマンドの影響を受けません。また、実際に作成されるイベント通知用ファイルのサイズはシステムに依存したページサイズ単位まで拡張されます。

$size には 0 から 10485760 までの値を指定できます。デフォルトは 2097152 (バイト)です。サイズ指定方法については上記「3.13.5 資源制限」の項を参照してください。

JobCenter 管理者特権が必要です。

SEt EVent_spool Time =($time)

NQSqwatch を使用して待ち合わせる内部イベントの最大スプーリング時間を $time で指定する時間に設定します。

本設定以前に作成されたイベント通知ファイルに対する待ち合わせは本設定の影響を受けません。また、イベント受信プロセスが指定時間内にイベントを受け取らなかった場合、該当するイベント通知ファイルに登録されたイベントはすべて破棄され、イベント通知ファイルは削除されます。

$timeは次のいずれかの形式で設定します。指定方法の詳細については上記「3.13.5 資源制限」の項を参照してください。

■HH:MM:SS.MS

■HH:MM:SS

■MM:SS.MS

■MM:SS

■SS.MS

■SS

■.MS

ただし.MSの形式(ミリ秒指定)は本サブコマンドでは意味を持ちません。

$time には秒数の場合で 0 から 4294967295 までの値を指定できます。デフォルトは 3600 (秒)です。

JobCenter 管理者特権が必要です。

SEt Forms $form-name [$form-name]...

$form-name で指定するフォームを設定します。

このコマンドを実行すると、以前に設定されていたフォームリストは削除され新しいリストが設定されます。なお、フォームリストへの追加は ADd Forms サブコマンドで行います。

JobCenter 管理者特権が必要です。

SEt Forms = $form-name $device

(R12.7よりこの機能はサポートしておりません)

$device で指定するデバイスに $form-name で指定するフォームを設定します。

JobCenter 操作員特権が必要です。

SEt FRee_destination Pipe_queue $queue

$queue で指定するパイプキューを自由転送先パイプキューとして使用することを設定します。

リクエストを新規に投入する場合、ユーザは任意の転送先キューを投入時オプションとして指定できます。自由転送先パイプキューはリクエストに記述されたオプションを読み取り、その転送先キューへリクエストを転送します。

その転送先へリクエストが転送できなかった場合、またはリクエストに転送先がオプションで指定されなかった場合は、この自由転送先パイプキューに指定された転送先が使用されます。

リクエストに指定する転送先オプションは、ほかのキューへ転送された時点で無効になります。また投入するパイプキュー自体を転送先に指定することはできません。

JobCenter 管理者特権が必要です。

SEt GLobal Batch_request_limit $limit

システム全体のバッチキューにおけるリクエスト同時実行可能数の上限を設定します。

各バッチキューの同時実行可能リクエスト数に余裕があっても、システム全体で実行しているバッチリクエスト数が本サブコマンドで設定した値になると、それ以上バッチリクエストは実行されません。

$limit にシステム内のすべてのバッチキューの同時実行リクエスト数の合計より大きい値を指定した場合、本設定は意味をなさなくなります。

JobCenter 管理者特権が必要です。

■13.1以前の場合

$limitにはUNIX版では1から1000まで、Windows版では1 から 100 までの値を指定できます。デフォルトは 100 です。

■13.2以降の場合

$limitには1 から 1000 までの値を指定できます。デフォルトは 1000 です。

SEt GLobal GROUP_Run_limit $run_limit

1つのグループがシステム内で同時に実行可能なリクエスト数の上限を設定します。

UNIX版のみサポートしています。

1つのグループについて、本サブコマンドで設定した数のリクエストしか実行されなくなります。もし設定数を超えてリクエストを起動しようとすると、そのリクエストは queued 状態にされます。

$run_limit には 0 から 32767 までの値を指定できます。デフォルトは 0 (無制限) です。

JobCenter 操作員特権が必要です。

SEt GLobal GROUP_Submit_limit $limit

1つのグループがシステム内に投入可能なリクエスト数の上限を設定します。

UNIX版のみサポートしています。

投入できるリクエスト上限数とは running, queued, suspending, waiting, holding, exiting の状態にある各リクエストと、チェックポイントが取得されてリスタートファイルのみ残っているリクエストの合計の上限数です。

1つのグループはそれぞれ本サブコマンドで設定した数のリクエストしか投入できなくなります。もし指定された数を超えてリクエストを投入するとエラーとなり、メッセージが出力されます。

$limit には 0 から 32767までの値を指定できます。デフォルトは 0 (無制限) です。

JobCenter 管理者特権が必要です。

SEt GLobal Network_request_limit $limit

システム全体で実行可能なネットワークリクエスト数の上限を設定します。

UNIX版のみサポートしています。

各ネットワークキューの同時実行可能リクエスト数に余裕があっても、各ネットワークキューで実行しているリクエスト数の総数が本サブコマンドで設定した値に達すると、それ以上リクエストは同時に実行されません。

JobCenter 管理者特権が必要です。

■13.1以前の場合

$limitには1から1000までの値を指定できます。デフォルトは 50 です。

■13.2以降の場合

$limitには1から1000までの値を指定できます。デフォルトは 1000 です。

ただし $limit でシステム内の すべてのネットワークキューの同時実行リクエスト数の合計より大きい値を指定した場合、本設定は意味をなさなくなります。

SEt GLobal Pipe_request_limit $limit

システム全体のパイプキューにおけるリクエスト同時転送可能数の上限を設定します。

各パイプキューの同時転送可能リクエスト数に余裕があっても、各パイプキューで実行しているリクエスト数の総数が本サブコマンドで設定した値に達すると、それ以上リクエストは同時に転送されません。

JobCenter 管理者特権が必要です。

■13.1以前の場合

このパラメータはUNIX版では1から1000までの値を指定できます。デフォルトは 100 です。Windows版では1 から 50 までの値を指定できます。デフォルトは 50 です。

■13.2以降の場合

このパラメータは1から1000までの値を指定できます。デフォルトは 1000 です。

ただし $limit でシステム内のすべてのパイプキューの同時実行リクエスト数の合計より大きい値を指定した場合、本設定は意味をなさなくなります。

SEt GLobal USER_Run_limit $run_limit

1人のユーザがシステム内で同時に実行可能なリクエスト数の上限を設定します。

UNIX版のみサポートしています。

1人のユーザについて、本サブコマンドで設定した数のリクエストしか実行されなくなります。もし設定数を超えてリクエストを起動しようとすると、そのリクエストは queued 状態にされます。

$run_limit には 0 から 32767 までの値を指定できます。デフォルトは 0 (無制限) です。

JobCenter 操作員特権が必要です。

SEt GLobal USER_Submit_limit $limit

1人のユーザがシステム内に投入可能なリクエスト数の上限を設定します。

UNIX版のみサポートしています。

投入できるリクエスト上限数とは、running, queued, suspending, waiting, holding, exitingの状態にある各リクエストと、チェックポイントが取得されてリスタートファイルのみ残っているリクエストの合計の上限数です。

1人のユーザはそれぞれ本サブコマンドで設定した数のリクエストしか投入できなくなります。もし指定された数を超えてリクエストを投入するとエラーとなり、メッセージが出力されます。

$limit には 0 から 32767までの値を指定できます。デフォルトは 0 (無制限) です。

JobCenter 管理者特権が必要です。

SEt GRoup Runlimit = $limit $queue

$queue で指定するキューに対するグループごとのリクエスト同時実行数を設定します。

同一グループに属するユーザは、各ユーザが指定キューで実行しているリクエスト数の総数が本コマンドで設定した値に達すると、それ以上リクエストは同時に実行されません。

$limit には 0 から 50 までの値を指定できます。デフォルトは 0 (無制限) です。

JobCenter 管理者特権が必要です。

SEt INTEr_queue_scheduling_type $type

キュー間でのリクエストスケジューリング方式を $type で指定する方式に設定します。バッチキューに対してのみ有効です。

$typeの指定方法は次の通りです。

■type0

従来のスケジューリング方式で、同一キュープライオリティをもつキュー間では各キューのスケジュール対象になっているリクエストの投入順序により実行順序が決められます。

■type1

同一キュープライオリティをもつキュー間では、投入順序に関係なく、それぞれのキューから順番にリクエストが実行されます。これにより、1つのキューに投入されているリクエストだけが集中的に実行されることを回避します。

デフォルトは type0 です。

JobCenter 管理者特権が必要です。

SEt INTRa_queue_scheduling_type $type $queue

$queue で指定するキュー内でのリクエストスケジューリング方式を $type で指定する方式に設定します。バッチキューに対してのみ有効です。

$typeの指定方法は次の通りです。

■type0

従来のスケジューリング方式で、同一リクエストプライオリティをもつリクエスト間では投入順序により実行順序が決められます。

■type1

同一リクエストプライオリティをもつリクエスト間では、投入順序に関係なく、それぞれのユーザのリクエストを順番に実行します。

1人のユーザはあらかじめ連続して実行できるリクエストの数が決められており、それ以上リクエストを投入していた場合は、ほかのユーザのリクエストが実行されるまで順番待ちをします。これにより、1人のユーザが投入したリクエストだけが集中的に実行されることを防ぎます。

1人のユーザが連続して実行できるリクエストの数は、SEt CONtinuous_scheduling_number サブコマンドで設定されます。

デフォルトは type0 です。

JobCenter管理者特権が必要です。

SEt LIfetime $life_time

システム内のパイプキューリクエストのライフタイムを時間単位で設定します。(分単位や秒単位の設定はできません)

ライフタイムを過ぎたパイプキューリクエストは削除されて投入元にエラーを返します。

$life_time には 0 から 100000 までの値を指定できます。デフォルトは 0 (無制限) です。

JobCenter 管理者特権が必要です。

SEt LOAd_balance Batch_queue $queue [keep_request_limit = $num] [Deliver_wait = $time]

$queue で指定するバッチキューを負荷分散用のバッチキューとして使用することを設定します。

$num にはこのバッチキューのリクエスト保有数制限を指定します。

$time にはデマンドデリバリ方式の負荷分散で運用する際のリクエスト到着待ち時間を指定します。

$num にはUNIX版では 0 から 6000 までの値を指定できます。Windows版では 0 から 2500 までの値を指定できます。デフォルトは 0 です。

$time には 0 から 9999 までの値を指定できます。デフォルトは 30 (秒)です。

上記はいずれも 0 は無制限を意味しませんので注意してください。

JobCenter 管理者特権が必要です。

SEt LOAd_balance Pipe_queue $queue [Reserve_run_limit = $num Destination_retry_wait = $time]

$queue で指定するパイプキューを負荷分散用のパイプキューとして使用することを設定します。

$num には、デマンドデリバリ用に確保しておくリクエストの同時転送数です。これはそのパイプキューに設定された同時転送数の値（run_limit － 1）以下でなければなりません。条件を満たさない場合、リクエストは転送処理されなくなります。

$time には負荷分散スケジューリング待ちの場合にパイプキュー上で WAIT 状態で待ち合わせる時間を設定します。WAIT 状態で設定した時間までバッチキューへ転送されなかった場合は、再び転送を試みます。

$num には 0 から 上限値 までの値を指定できます。上限値はデフォルトでUNIX版では 220、Windows版では190 です。SEt RUn_limitサブコマンドでリクエスト同時実行数制限を設定している場合は、上限値はその値と同じとなります。

$numのデフォルトは 0 です。

$time には 0 から 32400 までの値を指定できます。デフォルトは 3600 (秒)です。

上記はいずれも 0 は無制限を意味しませんので注意してください。

この設定は、パイプキューのほかのいくつかの属性と同時に使用できません。

JobCenter 管理者特権が必要です。

Reserve_run_limit または Destination_retry_wait を設定する場合には、必ずその両方のパラメータを同時に指定してください。

片方だけ指定した場合、指定されなかった方は「既定値が指定された」と解釈して再設定されます。

SEt LOG_file $file_name [Size = ($logfile_size) [NO_Backup]]

JobCenter メッセージのためのログファイル名を絶対パスで指定します。

$logfile_size には、ログファイルのサイズに上限値を設定します。ファイルサイズが上限値に達するとローテーションを行い、以前のログファイルはバックアップファイルに保存されます。上限値が設定されない場合はサイズは制限されません。

NO_Backup が同時に指定された場合には、バックアップファイルは作成されません。

$logfile_size には 数値としては 1 から 99999999 までの値を指定できます。上記「3.13.5 資源制限」の項を参照してください。ただし内部に設定される値としては 1024 バイトから 2147483647 バイトまでの範囲になるよう指定してください。

サイズやバックアップの設定を解除するには、 Size や NO_Backup を指定せずに再びこのコマンドを実行してください。

JobCenter 管理者特権が必要です。

SEt MAChine_group = ()

SEt MAChine_group = ($scheduler_machine[,$machine]...)

マシングループの設定、解除を行います。設定はコマンドの後に、1つまたはそれ以上のマシン名を指定します。このときマシングループの先頭に指定されたマシンがそのグループのスケジューラマシンとして認識されます。(なおマシン台数に制限はありませんが、通常8マシン程度を推奨しています)

マシングループの設定を解除するには、グループメンバを指定せずに再びこのコマンドを実行してください。

ただし最低限自マシンがマシングループに設定されていないとJobCenterが正常に動作しなくなりますので、自マシンのみ指定して直ちに設定しなおすよう注意してください。

JobCenter 管理者特権が必要です。

SEt MAIl $user_id

JobCenter システムが送信するメールの送信者を設定します。

JobCenter 管理者特権が必要です。

SEt MANager $manager [$manager]...

$manager で指定するアカウントをJobCenter 管理者リストに設定します。

$managerの指定方法として次の2つの形式があります。

■$local_account_name:m または $local_account_name:o

■[$local_user_id]:m または [$local_user_id]:o

たとえば $ local_account_name が “user1”、 $local_user_id が“1001”の場合、“user1:m”または“[1001]:m”のいずれかの形式で指定します。

■アカウント名または[ユーザid]に:m を付与

すべての qmgr コマンドを使うことができるJobCenter 管理者特権についての設定操作を意味します。

■アカウント名または[ユーザid]に:o を付与

一部のコマンドを使うことができるいわゆるJobCenter 操作員特権についての設定操作を意味します。

特権についての詳細は上記「3.13.3 特権」の項を参照してください。

UNIXの場合、nsumsmgrとrootアカウント(スーパーユーザ) はあらかじめJobCenter管理者特権付きで登録されています。Windowsの場合JobCenter管理者があらかじめJobCenter管理者特権付きで登録されています。

JobCenter 管理者特権が必要です。

SEt MAPping_mode $mode JobCenter ネットワークマッピングモードを設定します。マッピングモードによって、リモートホストのユーザから転送されたリクエストについて、ローカルホスト上のジョブ実行ユーザを決定する方法が異なります。

$mode は type1, type2, type3 のいずれかの文字列、または 0, 1, 2 の数字で指定します。

■0 または TYPE1

リモートホスト上のユーザ名と同じ名前のユーザにマッピングします。

■1 または TYPE2

ユーザ名に関係なく、nmapmgrに設定したユーザにマッピングします。

■2 または TYPE3

nmapmgrに設定したユーザ、さもなくばユーザ名の一致するユーザにマッピングします。

JobCenter 管理者特権が必要です。

SEt MAXimum Copies $copies

$copies をプリントの最大複写数として設定します。デフォルトは 2 です。

JobCenter 管理者特権が必要です。

SEt MAXimum IDc_connection $number

ホスト間通信で使用する IDC の最大同時接続数を設定します。

UNIX版のみサポートしています。

マシングループ内の各マシンとの通信を行う際のオーバヘッドを減らすために、スケジューラマシンが他のマシンと切断せずに保持しておくコネクションの上限数です。

デマンドデリバリによる負荷分散機能を利用する際もこのIDCによる通信が行われますが、この場合ジョブを分散するホスト数よりもこの設定値が小さい場合、回線の設定、破棄が頻発することになるため全体のジョブスループットが低下する可能性がありますので注意してください。

$number には 1 から 1024 までの値を指定できます。デフォルトは 32 です。

JobCenter 管理者特権が必要です。

SEt MAXimum Open_retries $retries

(R12.7よりこの機能はサポートしておりません)

$retries で指定する数をデバイスのオープンに失敗したときの最大再試行数として設定します。デフォルトは 2 回です。

JobCenter 管理者特権が必要です。

SEt MAXimum Print_size $size

$size で指定する数をJobCenter プリントファイルの最大バイト数として設定します。

$size には 1 から 100000000 までの値を指定できます。デフォルトは 1000000 (バイト)です。

JobCenter 管理者特権が必要です。

SEt MAXimum Request_priority $limit

qsub(1) コマンドで -p オプションに指定できるリクエストプライオリティを 0 から$limit までに制限します。制限値以上のプライオリティが指定された場合は、そのリクエストの投入が拒否されます。

SEt DEFault Batch_request Priority サブコマンドで設定した既定リクエストプライオリティと同じか、より高い値を指定する必要があります。

$limit は 0 が最も低く、63 が最も高い優先度となります。リクエストプライオリティについては<NQS機能利用の手引き>の「2.1.1 バッチリクエスト」を参照してください。デフォルトは 0 (無制限)です。

qsub(1) コマンドはUNIX版のみサポートしています。

JobCenter 管理者特権が必要です。

SEt NETWORK Client = ($client)

リクエストの実行結果転送のために使用されるサーバプログラム名を設定します。

Create Network_queue サブコマンドで Server の 指定がなかった場合、このサブコマンドで指定したプログラムが用いられます。

$client には、サーバプログラムの絶対パス名に必要な引数を加えたパラメータを指定します。詳細は<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

JobCenter 管理者特権が必要です。

本サブコマンドは SEt NETWORK_client サブコマンドと名称が似ているのでご注意ください。

SEt NETWORK_client = ($client) $queue

$queue で指定するネットワークキューのサーバプログラム名を $client に変更します。

$client には、ネットワークキューのサーバプログラムの絶対パス名を指定します。詳細は<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

JobCenter 管理者特権が必要です。

本サブコマンドは SEt NETWORK Client サブコマンドと名称が似ているのでご注意ください。

SEt NIce_limit = $nice_value $queue

$queue で指定するバッチキューに nice 値制限（$nice_value）を設定します。

すでにキューに登録されているリクエストには、新しい設定値は何も影響を与えません。

この値は投入されるリクエストに設定された nice 値と比較され、その値がバッチキューに設定された $nice_value より小さい (より優先度が高い) 場合はリクエストの登録が拒否されます。

またリクエストにnice 値が特に設定されていない場合は、キューに設定した $nice_value をリクエストの nice 値として使用します。

$nice_value には正または負の整数を指定します。上記「3.13.5 資源制限」の項を参照してください。

JobCenter 管理者特権が必要です。

SEt NO_Access $queue

$queue で指定するキューのユーザ、グループごとの利用制限を有効にします。

Add User、ADd Group サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

Set NO_Check $queue

$queue で指定するパイプキューの事前チェック機能を解除します。

Create Pipe_queue、SEt CHeck サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

SEt NO_CUStomjob Batch_queue $queue

$queueで指定するバッチキューのカスタムキュー属性を解除します。

SEt CUStomjob Batch_queue サブコマンドも参照してください。

JobCenter管理者特権が必要です。

SEt NO_Default Batch_request Queue

既定バッチリクエストキューの設定を解除します。

SEt DEFault Batch_request Queue サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

SEt NO_Default PRINT_Request Forms

既定プリンタリクエストフォームの設定を解除します。

SEt DEFault PRINT_Request Forms サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

SEt NO_Default Print_request Queue

既定プリンタリクエストキューの設定を解除します。

SEt DEFault PRINT_Request Queue サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

SEt NO_Free_destination Pipe_queue $queue

$queue で指定するパイプキューの自由転送先パイプキューとしての属性を解除します。

SEt FRee_destination Pipe_queue サブコマンドも参照してください。

JobCenter 操作員特権が必要です。

SEt NO_Load_balance Batch_queue $queue

$queue で指定する負荷分散用のバッチキューの設定を解除します。

SEt LOAd_balance Batch_queue サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

SEt NO_Load_balance Pipe_queue $queue

$queue で指定する負荷分散用のパイプキューの設定を解除します。

SEt LOAd_balance Pipe_queue サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

Set NO_Staywait $queue

$queue で指定するパイプキューの staywait 機能を解除します。

Create Pipe_queue 、SEt STAYwait サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

Set NO_Transparent Pipe_queue $queue

$queue で指定する透過型パイプキューの設定を解除します。

SEt Transparent Pipe_queue サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

SEt Open_wait $interval

(R12.7よりこの機能はサポートしておりません)

デバイスのオープンが失敗したときに待つ間隔 (秒数) を指定します。

$interval には 1 から 1024 までの値を指定できます。デフォルトは 5 (秒)です。

JobCenter 管理者特権が必要です。

SEt PER_Process Cpu_limit = ($limit) $queue

$queue で指定するバッチキューにプロセスごとの CPU 時間制限値を設定します。

本サブコマンドは、システムでプロセスごとの CPU 時間制限値の設定がサポートされている場合のみ有効です。

この値は投入されるリクエストに設定されたプロセスごとの CPU 時間制限値と比較され、その値が、キューに設定された値より大きい場合はリクエストの登録が拒否されます。またリクエストにプロセスごとの CPU 時間制限値が特に設定されていない場合は、キューに設定した値をリクエストの制限値として使用します。

すでにキューに登録されているリクエストには、新しく設定される値は何も影響を与えません。

$limit の構文については、上記「3.13.5 資源制限」の項を参照してください。

JobCenter 管理者特権が必要です。

SEt PER_Process Memory_limit = ($limit) $queue

$queue で指定するバッチキューにプロセスごとのメモリサイズ制限値を設定します。

本サブコマンドは、システムでプロセスごとのメモリサイズ制限値の設定がサポートされている場合のみ有効です。

この値は投入されるリクエストに設定されたプロセスごとのメモリサイズ制限値と比較され、その値が、キューに設定された値より大きい場合はリクエストの登録が拒否されます。またリクエストにプロセスごとのメモリサイズ制限値が特に設定されていない場合は、キューに設定した値をリクエストの制限値として使用します。

すでにキューに登録されているリクエストには、新しく設定される値は何も影響を与えません。

$limitの構文については、上記「3.13.5 資源制限」の項を参照してください。

JobCenter 管理者特権が必要です。

SEt PER_Process Permfile_limit = ($limit) $queue

$queue で指定するバッチキューにプロセスごとの永久ファイルサイズ制限値を設定します。

本サブコマンドは、システムでプロセスごとの永久ファイルサイズ制限値の設定がサポートされている場合のみ有効です。

この値は投入されるリクエストに設定されたプロセスごとの永久ファイルサイズ制限値と比較され、その値が、キューに設定された値より大きい場合はリクエストの登録が拒否されます。またリクエストにプロセスごとの永久ファイルサイズ制限値が特に設定されていない場合は、キューに設定した値をリクエストの制限値として使用します。

すでにキューに登録されているリクエストには、新しく設定される値は何も影響を与えません。

$limit の構文については、上記「3.13.5 資源制限」の項を参照してください。

JobCenter 管理者特権が必要です。

SEt PIpe_client = ($client) $queue

$queue で指定するパイプキューと $client で指定する転送プログラムを関係づけます。

$client　には、転送プログラムの絶対パス名と、必要な引き数を付加したパラメータを指定します。指定方法の詳細は<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」および「6.7 負荷分散環境」を参照してください。

JobCenter 管理者特権が必要です。

SEt PRiority = $priority $queue

$queueで指定するキューに にプライオリティを設定します。

$priority は 0 が最も低く、63 が最も高い優先度となります。キュープライオリティについては<NQS機能利用の手引き>の「5.3 JobCenterキューの属性定義」を参照してください。

JobCenter 管理者特権が必要です。

SEt Queue Reboot_mode = $val $queue

$queueで指定するキューに 再起動時の動作モードを設定します。

$valは次のいずれかの形式を指定します。

■RESTART

■STOP

■PURGE

■MIGRATION_STOP

■MIGRATION_PURGE

指定方法の詳細は<NQS機能利用の手引き>の「6.1.3.7 再起動属性」を参照してください。

JobCenter 管理者特権が必要です。

ERP・BIジョブは、キューを利用しないでジョブの投入を行っているため、キュー起動時の属性（reboot_mode）を設定してもその影響を受けません。

なお、起動時の設定を"STOP"に設定しジョブの実行を停止させたい場合、対処策として各ERP・BIジョブの直前にダミーの単位ジョブを配置してください。

SEt RUn_limit = $run_limit $queue

$queue で指定するバッチまたはパイプキューのリクエスト同時実行数制限を $run_limit で指定する数に変更します。

$run_limit には 1 から 上限値 までの値を指定できます。上限値は、GLobal Batch_request_limit、GLobal Pipe_request_limit 及び GLobal Network_request_limitの合計値です。

デフォルトの状態では、上限値はUNIX版で 200、Windows版で 190 となります。

JobCenter 操作員特権が必要です。

SEt SHell_strategy FIxed = ($shell)

バッチリクエストを実行するシェルプログラム選択方式を、指定した $shell 固定にします。

UNIXの場合は "/bin/sh" や "/bin/csh" 等のOSで使用可能なシェルプログラムを絶対パスで指定します。(なおユーザプログラムを指定した場合の正常動作は保証しておりません)

Windowsの場合は本指定がデフォルトであり、cmd.exeの絶対パス指定以外の実行プログラム指定はサポートしておりません。

JobCenter 管理者特権が必要です。

SEt SHell_strategy FRee

バッチリクエストを実行するシェルプログラム選択方式をFREEにします。

バッチリクエストを実行する際に、まずリクエストのユーザのログインシェルが起動されます。次にそのログインシェルが、バッチリクエストの内容から適切なシェルを選択し、そのシェルがバッチリクエストを実行します。

あたかも、インタラクティブな処理と同様な形態でバッチリクエストが実行されます。

UNIXの場合は本指定がデフォルトです。Windowsの場合は本指定をしても解釈されず、cmd.exe固定で動作は変更しません。

JobCenter 管理者特権が必要です。

SEt SHell_strategy Login

UNIXの場合、バッチリクエストを実行するシェルをユーザのログインシェルにします。 ログインシェルは、**getent passwd** で指定されているシェルが選択されます。

Windowsの場合は本指定をしても解釈されず、cmd.exe固定で動作は変更しません。

JobCenter 管理者特権が必要です。

SEt STACk_limit = ($limit) $queue

$queue で指定するバッチキューにスタックセグメントサイズ制限値を設定します。

本サブコマンドはシステムでサポートされている場合のみ有効です。

この値は投入されるリクエストに設定されたスタックセグメント制限値と比較され、その値が、キューに設定された値より大きい場合はリクエストの登録が拒否されます。またリクエストにスタックセグメントサイズ制限値が特に設定されていない場合は、キューに設定した値をリクエストのスタックセグメントサイズ制限値として使用します。

すでにキューに登録されているリクエストには、新しく設定される値は何も影響を与えません。

$limit の構文については、上記「3.13.5 資源制限」の項を参照してください。

JobCenter 管理者特権が必要です。

SEt STAYwait $queue

$queue で指定するパイプキューに staywait 機能を設定します。

Create Pipe_queue、SEt NO_STAYwait サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

Set Transparent Pipe_queue $queue

$queue で指定するパイプキューを透過型パイプキューとして使用することを設定します。

この設定は、上記のstaywait機能などパイプキューいくつかの属性と同時に使用できません。

Set NO_Transparent Pipe_queue サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

SEt UNrestricted_access $queue

$queue で指定するキューのアクセス制限を無効とし、すべてのユーザを利用可能とします。

SEt NO_Access サブコマンドも参照してください。

JobCenter 管理者特権が必要です。

SEt USer Runlimit = $limit $queue

$queue で指定するキューに対するユーザのリクエスト同時実行数を設定します。

1人のユーザは、指定されたキュー上で同時に実行できるリクエスト数を　$limit　の値に制限されます。指定キューで実行しているリクエスト数の総数が本コマンドで設定した値になるとそれ以上そのユーザのリクエストは同時実行されません。

$limit には 0 から 50 までの値を指定できます。デフォルトは 0 (無制限) です。

JobCenter 管理者特権が必要です。

SHOw All

デバイス、フォーム、制限、管理者、パラメータ、キューについての情報の要旨を表示します。

詳細については以降のコマンドを参照してください。

SHOw Complex [$complex-name]

すべてのキュー複合体の状態を表示します。

$complex-name を指定した場合、状態表示はそのキュー複合体に限定して表示します。

SHOw Device [$device]

(R12.7よりこの機能はサポートしておりません)

すべてのJobCenter デバイスの状態を表示します。

$device を指定した場合は、そのデバイスについての情報のみ表示されます。

SHOw Forms

有効なフォームのリストを表示します。

SHOw Limits_supported

システムでサポートされているJobCenter 資源制限のリストを表示します。

SHOw LOng Complex [$complex-name]

すべてのキュー複合体の状態を拡張形式で表示します。

$complex-name を指定した場合は、そのキュー複合体に限定して表示します。

SHOw LOng Queue [$queue [$user]]

すべての JobCenter キューの状態を拡張形式で表示します。

$queue を指定した場合は、そのキューに限定して表示します。

$user を指定した場合は、そのユーザに属しているリクエストに限定して表示します。

R13.2以降では、JobCenter停止中にキューのリクエストの情報を表示することはできません。

SHOw MAChine_group

設定されたマシングループを表示します。

SHOw MANagers

管理者特権または操作員特権を与えられた JobCenter 管理者のリストを表示します。

SHOw Parameters

システム全体に関わるJobCenter パラメータのリストを表示します。

SHOw Queue [$queue [$user]]

すべてのキューの状態を表示します。

$queue を指定した場合、状態表示はそのキューに限定して表示します。

$user を指定した場合、キュー上のリクエストのうち、そのユーザが投入したリクエストに限定して状態を表示します。

R13.2以降では、JobCenter停止中にキューのリクエストの情報を表示することはできません。

SHUtdown [$seconds]

JobCenter をシャットダウンします。

各々の実行中のリクエストの各プロセスにまず SIGTERM シグナルが送られて、$seconds で指定した実時間が経過した後に SIGKILL シグナルが各々のリクエストの残存するプロセスに送られます。

$seconds には 0 から 600 までの値を指定できます。デフォルトは 60 (秒) です。

JobCenter 操作員特権が必要です。

JobCenterの正式な停止コマンドはnqsstopであり、本サブコマンドではありません。 本サブコマンドはNQSの常駐プロセスの停止と、それに依存する常駐プロセスのうちいくつかの部分的な停止を伴います。本サブコマンドはシステム全体が不正な状態に落ちいった場合などの緊急時以外は使用しないようご注意ください。

STArt Queue $queue

$queue で指定するキューを開始します。

本サブコマンドで $queue を開始すると、そのキューが使用可能状態(enable)の場合はリクエストの処理が再開します。

指定したキューがすでに開始されている場合は何も起こりません。

JobCenter 操作員特権が必要です。

STArt All Queue

システム上のすべてのキューを開始します。

本サブコマンドですべてのキューを開始すると、使用可能状態(enable)のキューについてはリクエストの処理が再開します。

すでに開始されているキューについては何も起こりません。

JobCenter 操作員特権が必要です。

STOp Queue $queue

$queue で指定するキューを停止します。

本サブコマンドで $queue を停止すると、そのキュー上にあるリクエストについて実行開始せず、処理が抑制されます。ただしそのキュー上ですでに実行状態にあるリクエストについては処理は続行されます。

DIsable Queue サブコマンドと異なり、停止したキューに新しくリクエストを投入することは可能ですが、キューイングされるだけで実行開始せず処理が抑制されます。

JobCenter 操作員特権が必要です。

STOp All Queue

システム上のすべてのキューを停止します。

すべてのキュー上にあるリクエストについて実行開始せず、処理が抑制されます。ただし、すでに実行状態にあるリクエストについては処理は続行されます。

DIsable All Queue サブコマンドと異なり、停止したキューに新しくリクエストを投入することは可能ですが、キューイングされるだけで実行開始せず処理が抑制されます。

JobCenter 操作員特権が必要です。

SUspend Request $request_id

$request_id で指定するリクエストの実行を一時中断します。

リクエストはRUNNING 状態のものに限ります。このコマンドで中断したリクエストはJobCenter のユーザコマンド qrsm では実行再開はできません。RESUme Request サブコマンドで再開してください。

qrsmコマンドはUNIX版のみサポートしています。

JobCenter 操作員特権が必要です。

Unlock Local_daemon

メモリにプロセスロックしている JobCenter NQSデーモンのプロセスロックを除去します。

plock(2)の UNLOCK オプションを参照してください。

JobCenter 操作員特権が必要です。

3.14. jdh_download 定義情報のダウンロード

■UNIX版

```
/usr/lib/nqs/gui/bin/jdh_download [def] [-h $hostname[:$port]] [-e [$execuser]] [-u $user | -a] [-p $password] [-t $target] [-g[j] $jnwgroup | -j $jnw] [-gs $schgroup | -s $sch] [-gd $customgroup | -d $customname] [-c $wcalname] [-l $triggername] [-r $rel_target] [-o $filename | -o $dirname ] [-w $second] [-f]
/usr/lib/nqs/gui/bin/jdh_download conf [-h $hostname[:$port]] [-u $user] [-p $password] [-o $filename] [-w $second]
```

■Windows版

```
%InstallDirectory%\bin\jdh_download [def] [-h $hostname[:$port]] [-e [$execuser]] [-u $user | -a] [-p $password] [-t $target] [-g[j] $jnwgroup | -j $jnw] [-gs $schgroup | -s $sch] [-gd $customgroup | -d $customname] [-c $wcalname] [-l $triggername] [-r $rel_target] [-o $filename | -o $dirname ] [-w $second] [-f]
%InstallDirectory%\bin\jdh_download conf [-h $hostname[:$port]] [-u $user] [-p $password] [-o $filename] [-w $second]
```

3.14.1. 機能説明

JobCenter MG/SVから定義情報、または構成情報をダウンロードして、JPFファイルを作成します。JobCenter CL/Winのダウンロード操作と同等の機能をコマンドラインから実行することができます。コマンド名の後にサブコマンドが指定でき、サブコマンドに何も指定しないか「def」を指定すると定義情報のダウンロードになります。サブコマンドに「conf」を指定すると構成情報のダウンロードになります。

3.14.2. オプション

-h $hostname[:$port]

$hostnameは、ダウンロード先のJobCenter MG/SVのホスト(サイト)名、またはIPアドレスを指定します。

$portは、ダウンロード先のJobCenter MG/SVの接続ポート番号(jccombaseのTCPポート番号)を指定できます。

■ポート番号を指定する際の注意事項

以下の条件をともに満たす環境下で実行する場合のみ、ポート番号を指定してください。

- JobCenter MG/SV、およびJobCenter CL/Winが未インストールの環境下においてDefinition Helper同梱の本コマンドを使用する場合
- ダウンロード先のJobCenter MG/SVの接続ポート番号(jccombaseのTCPポート番号)をデフォルト値(611)以外に変更している場合

$hostnameにIPアドレスを指定することもできます。指定方法は以下のとおりです。

■ポート番号を指定しない場合

<IPアドレス>の形式で指定してください。

■ポート番号を指定する場合

IPv4アドレスの場合は、<IPv4アドレス:ポート番号>の形式で指定してください。

IPv6アドレスの場合は、<[IPv6アドレス]:ポート番号>の形式で指定してください。

本オプションを指定しない場合、コマンド実行マシンからダウンロードします。

-u $user | -a

ダウンロードするユーザを指定します。

■-u $user

$userで指定したユーザを対象とします。

■-a

全ユーザを対象とします。

-aを指定した時は-rオプションで jscfが指定された場合と同様の動作となります。

本オプションを指定しない場合、コマンド実行ユーザでログインします。

-e $execuser

ログイン先ユーザを指定します。

本オプションを指定しない場合は以下の動作となります。

■-uでユーザが指定されている。

-uで指定したユーザでログインします。

■上記以外

コマンドを実行しているユーザ名でログインします。

-p $password

ログイン先ユーザのパスワードを指定します。

本オプションを指定しない場合、パスワードプロンプトが表示されます。

-t $target

ダウンロードする定義情報の種類を選択します。種類の選択には、以下の文字を指定してください。

j：ジョブネットワーク
s：スケジュール
c：カレンダ
d：カスタムジョブ定義
l：起動トリガ

本オプションを指定しない場合、「j」を指定したとみなします。

-g[j] $jnwgroup | -j $jnw

ダウンロードするジョブネットワークグループ、ジョブネットワークを指定します。

■-g[j] $jnwgroup

$jnwgroupで指定したグループを対象とします。

グループの階層指定は「.」を区切り文字とします。グループパスはルート階層から指定してください(例:.daily_job.job_bat01)

-g、-gjどちらとも同じ動作を行います。-gは過去との互換性のために存在します。

■-j $jnw

$jnwで指定したジョブネットワークを対象とします。

-tオプションでjを指定している状態で、本オプションを指定しない場合、全ジョブネットワーク定義情報をダウンロードします。

本オプションは-aオプションと同時には指定できません。

-r $rel_target

関連するサブジョブネットワーク、カレンダ分岐の参照スケジュール、カレンダ、監視対象テキストログもダウンロードするかどうか指定します。

j: 関連するサブジョブネットワークもダウンロードします

s: 関連するカレンダ分岐の参照スケジュールもダウンロードします。

c: 関連するカレンダ分岐の参照スケジュールで選択したカレンダもダウンロードします。

f: 全ての監視対象テキストログもダウンロードします。

m: 関連する監視対象テキストログもダウンロードします。

-gs $schgroup | -s $sch

ダウンロードするスケジュールグループ、スケジュールを指定します。

■-gs $schgroup

$schgroupで指定したグループを対象とします。

グループの階層指定は「.」を区切り文字とします。グループパスはルート階層から指定してください(例:.daily_sch.sch01)

■-s $sch

$schで指定したスケジュールを対象とします。

-tオプションでsを指定している状態で、本オプションを指定しない場合、全スケジュール定義情報をダウンロードします。

本オプションは-aオプションと同時には指定できません。

本オプションを指定する場合、必ず-tオプションでsを指定してください。

-l $triggername

$triggernameで指定した起動トリガをダウンロードします。

-tオプションでlを指定している状態で、本オプションを指定しない場合、全起動トリガ定義情報をダウンロードします。

本オプションは-aオプションと同時には指定できません。

本オプションを指定する場合、必ず-tオプションでlを指定してください。

-gd $customgroup | -d $customname

ダウンロードするカスタムジョブテンプレートグループ、カスタムジョブテンプレートを指定します。

■-gd $customgroup

$customgroupで指定したグループを対象とします。

グループの階層指定は「.」を区切り文字とします。グループパスはルート階層から指定してください(例:.User.NewGroup1)

■-d $customname

$customnameで指定したカスタムジョブテンプレートを対象とします。

-tオプションでdを指定している状態で、本オプションを指定しない場合、全カスタムジョブテンプレート定義情報をダウンロードします。

本オプションは-aオプションと同時には指定できません。

本オプションを指定する場合、必ず-tオプションでdを指定してください。

-c $wcalname

$wcalnameで指定したカレンダをダウンロードします。

-tオプションでcを指定している状態で、本オプションを指定しない場合、全カレンダ定義情報をダウンロードします。

本オプションを指定する場合、必ず-tオプションでcを指定してください。

-o $filename | -o $dirname

ダウンロードした定義情報の出力先JPF名を指定します。全ユーザのダウンロードを指定した時はディレクトリ名となり、ディレクトリ配下に各ユーザ名のjpfファイルが作成されます。

本オプションを指定しない場合、「jc_def_<YYYYMMDDhhmmss形式の実行日時>.jpf」というファイル名で出力します。全ユーザのダウンロードの場合は、「jc_def_<YYYYMMDDhhmmss形式の実行日時>」というディレクトリを作成します。

全ユーザのダウンロード時には.jpfの名前がつくディレクトリ名は指定できません。.jpfがつく名前を指定した場合は.jpfの部分が削除された名前でディレクトリ名が作成されます。

-w $second

Comagnet通信のタイムアウト時間（秒）を指定します。タイムアウト値（秒）は30から86400までの整数で入力してください。 本オプションを指定しない場合、デフォルト値の600秒（10分）がタイムアウト値となります。

-f CL/Winでダウンロード対象の定義を編集中の場合でもダウンロードします。 本オプションを指定しない場合、CL/Winでダウンロード対象の定義を編集中の場合にはダウンロードに失敗します。

3.14.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.14.4. 主要メッセージ

メッセージ	内容
Invalid host format error.	接続先のホスト名、またはポート番号のフォーマットが正しくありません。 <サイト名>:<ポート番号>の形式になっているか確認してください。
jccombase port number should be from 1 to 65535.	接続先ポート番号は、1から65535の範囲で指定してください。
invalid timeout. [30 - 86400]	Comagnet通信のタイムアウト時間(秒)は、30から86400の範囲で指定してください。
Authentication failure. Check username or password.	ユーザ名もしくはパスワードが正しくありません。ログインユーザ情報が正しいか確認してください。
connection failure. hostname=($hostname) port=($port)	指定ホストに接続できません。接続先を確認してください。
Invalid first group path. ("." required)	グループ指定する際にはルート階層として先頭に「.」をつけてください。
no such a group path.	指定されたグループは存在しません。

3.14.5. 注意事項

■ホスト(サイト)名を指定する場合は、必ず名前解決が行えるようにOSのネットワーク環境を設定してください。

■ダウンロードする定義情報の量によっては、ダウンロード先のJobCenter MG/SVの搭載メモリ以上のメモリを確保しようとするため、メモリ不足によるエラーが発生する場合があります。ダウンロード時に必要なメモリ使用量の概算算出方法については、<環境構築ガイド>の「20.7　アップロード・ダウンロード時のメモリ使用量概算算出方法」を参照してください。

3.15. jdh_upload 定義情報のアップロード

■UNIX版

```
/usr/lib/nqs/gui/bin/jdh_upload [-h $hostname[:$port]] [-u $user] [-p $password] [-c] [-r $rulefile]
[-f] [-i] [-w $second] {$jpf_file|-a $jpf_dir}
```

■Windows版

```
%InstallDirectory%\bin\jdh_upload [-h $hostname[:$port]] [-u $user] [-p $password] [-c] [-r
$rulefile] [-f] [-i] [-w $second] {$jpf_file|-a $jpf_dir}
```

3.15.1. 機能説明

JobCenter MG/SVへ定義情報をアップロードします。JobCenter CL/Winのアップロード操作と同等の機能をコマンドラインから実行できます。

3.15.2. オプション

-h $hostname[:$port]

$hostnameは、アップロード先のJobCenter MG/SVのホスト(サイト)名またはIPアドレスを指定します。

$portは、アップロード先のJobCenter MG/SVの接続ポート番号(jccombaseのTCPポート番号)を指定できます。

■ポート番号を指定する際の注意事項

以下の条件をともに満たす環境下で実行する場合のみ、ポート番号を指定してください。

- JobCenter MG/SV、およびJobCenter CL/Winが未インストールの環境下においてDefinition Helper同梱の本コマンドを使用する場合
- アップロード先のJobCenter MG/SVの接続ポート番号(jccombaseのTCPポート番号)をデフォルト値(611)以外に変更している場合

$hostnameにIPアドレスを指定することもできます。指定方法は以下のとおりです。

■ポート番号を指定しない場合

<IPアドレス>の形式で指定してください。

■ポート番号を指定する場合

IPv4アドレスの場合は、<IPv4アドレス:ポート番号>の形式で指定してください。

IPv6アドレスの場合は、<[IPv6アドレス]:ポート番号>の形式で指定してください。

本オプションを指定しない場合、コマンド実行マシンへアップロードします。

-u $user

接続するJobCenter MG/SVログイン先ユーザを指定します。

本オプションを指定しない場合、コマンド実行ユーザでログインします。

-p $password

ログイン先ユーザのパスワードを指定します。

本オプションを指定しない場合、パスワードプロンプトが表示されます。

-w $second

Comagnet通信のタイムアウト時間（秒）を指定します。タイムアウト値（秒）は30から86400までの整数で入力してください。

本オプションを指定しない場合、デフォルト値の600秒（10分）がタイムアウト値となります。

-c

チェックモードで動作します。チェックモードでは、定義情報の依存関係確認のみを行い、サーバ上の定義情報を更新しません。

-r $rulefile

ルールファイルに記載されたホスト名の変換フォーマットにしたがって、変換を行います。

-f

キュー情報のチェックを省略します。

-i

拡張アイコンデータをアップロード対象に含めます。拡張カスタムジョブテンプレートをアップロードする場合は、本オプションを設定してください。

$jpf_file

アップロードする定義情報(JPFファイル)を指定します。本パラメータか-aオプションのディレクトリパラメータどちらかを必ず設定してください。

-a $jpf_dir

指定されたディレクトリ($jpf_dir)内のjpfファイルをすべてアップロードします。全ユーザダウンロードで生成したディレクトリを指定してください。

3.15.2.1. ルールファイル

ホスト名の変換定義を以下のように記述します。

変換元ホスト名1 変換先ホスト名1

変換元ホスト名2 変換先ホスト名2

・

・

3.15.3. 戻り値

戻り値	内　容
0	正常終了です。

1	異常終了です。

3.15.4. 主要メッセージ

メッセージ	内容
Invalid host format error.	接続先のホスト名、またはポート番号のフォーマットが正しくありません。 <サイト名>:<ポート番号>の形式になっているか確認してください。
jccombase port number should be from 1 to 65535.	接続先ポート番号は、1から65535の範囲で指定してください。
invalid timeout. [30 - 86400]	Comagnet通信のタイムアウト時間(秒)は、30から86400の範囲で指定してください。
Authentication failure. Check username or password.	ユーザ名もしくはパスワードが正しくありません。ログインユーザ情報が正しいか確認してください。
connection failure. hostname=($hostname) port=($port)	指定ホストに接続できません。接続先を確認してください。
cannot access "filename": No such file	指定したJPFファイル名は存在しません。
Dependency Error	パラメータの依存関係が正しくないためアップロードできません。例えば、スケジュールに関連付けしたカレンダがアップロードする定義情報やJobCenter MG/SV上の定義情報に存在しない場合、依存関係エラーとなります。 詳細については<基本操作ガイド>の「12.4　依存関係エラー」を参照してください。

3.15.5. 注意事項

■ホスト(サイト)名を指定する場合は、必ず名前解決が行えるようにOSのネットワーク環境を設定してください。

■アップロードする定義情報の量によっては、アップロード先のJobCenter MG/SVの搭載メモリ以上のメモリを確保しようとするため、メモリ不足によるエラーが発生する場合があります。アップロード時に必要なメモリ使用量の概算算出方法については、<環境構築ガイド>の「20.7　アップロード・ダウンロード時のメモリ使用量概算算出方法」を参照してください。

■巨大なJPFファイルのアップロードを行った場合、通信タイムアウトが発生する可能性があります。タイムアウトが発生してもサーバ側で定義の更新は継続して行われており、その間は定義の編集を行うことが出来ませんのでご注意ください。

3.16. jdh_trkget トラッカ情報の取得

■UNIX版

```
/usr/lib/nqs/gui/bin/jdh_trkget [-u $user] [-t arc] [-o $output] $YYYYMMDD-YYYYMMDD

/usr/lib/nqs/gui/bin/jdh_trkget [-u $user] -t act [-n $tracker-name] [-i $tracker-id] [-o $output]
[$YYYYMMDD-YYYYMMDD]

/usr/lib/nqs/gui/bin/jdh_trkget [-u $user] -t all [-n $tracker-name] [-i $tracker-id] [-o $output]
$YYYYMMDD-YYYYMMDD
```

■Windows版

```
%InstallDirectory%\bin\jdh_trkget [-u $user] [-t arc] [-o $output] $YYYYMMDD-YYYYMMDD

%InstallDirectory%\bin\jdh_trkget [-u $user] -t act [-n $tracker-name] [-i $tracker-id] [-o $output]
[$YYYYMMDD-YYYYMMDD]

%InstallDirectory%\bin\jdh_trkget [-u $user] -t all [-n $tracker-name] [-i $tracker-id] [-o $output]
$YYYYMMDD-YYYYMMDD
```

3.16.1. 機能説明

指定したオプションと合致するトラッカ情報を収集し、JPFファイルを作成します。

トラッカ情報とは、未アーカイブのトラッカ情報と、トラッカアーカイブファイルを指します。

本コマンドはrootユーザ(Windowsの場合はJobCenter管理者)で実行してください。

3.16.2. オプション

-u $user

トラッカ情報を取得するユーザを指定します。

本オプションを指定しない場合、取得する対象ユーザを全ユーザとします。

-t $target

取得対象のトラッカ情報を指定します。指定可能な文字列は以下になります。

■arc

取得対象をトラッカアーカイブファイルのみとします。

■act

取得対象を未アーカイブのトラッカ情報のみとします。

■all

取得対象を全てのトラッカ情報(未アーカイブのトラッカ情報、およびトラッカアーカイブファイル)とします。

本オプションを指定しない場合、トラッカアーカイブファイルのみを採取対象とします。

-n $tracker-name

取得対象のトラッカ名を指定します。

本オプションを指定しない場合、トラッカ名による取得対象の制限は行いません。

本オプションは、未アーカイブのトラッカ情報を取得対象とした場合のみ有効です。-t arcでは、指定できません。

-i $tracker-id

取得対象のトラッカIDを指定します。指定したトラッカIDは、取得期間の影響を受けません。

本オプションを指定しない場合、トラッカIDを指定したトラッカの取得は行いません。

(例) NewJnw.20121221095106

本オプションは、未アーカイブのトラッカ情報を取得対象とした場合のみ有効です。-t arcでは、指定できません。

-o $output

出力ファイル名を指定します。省略した場合はデフォルトの名前(trkget_YYYYMMDDhhmmss.jpf)で出力します。

$YYYYMMDD-YYYYMMDD

トラッカアーカイブファイルを取得する期間を指定します。期間は開始日と終了日を8桁の年月日(YYYYMMDD[ローカルタイム])でハイフンで繋いで指定します。

(例) 20110702-20110912

トラッカアーカイブファイルは、指定した期間に一部でも含まれる場合は取得します。

本オプションは、-t actのみ省略可能です。

3.16.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.16.4. 主要メッセージ

メッセージ	内容
get trkdata(archive) user(user-count).	取得前情報の表示です。 取得対象がトラッカアーカイブファイルのみで、取得対象のユーザ数をuser-countに表示します。
get trkdata(active) user(user-count) tracker(tracker-count).	取得前情報の表示です。 取得対象が未アーカイブのトラッカ情報のみで、取得対象のユーザ数をuser-count、取得する未アーカイブのトラッカ情報数をtracker-countに表示します。
get trkdata(all) user(user-count) tracker[active](tracker-count).	取得前情報の表示です。 取得対象が全てのトラッカ情報で、取得対象のユーザ数をuser-count、取得する未アーカイブのトラッカ情報数をtracker-countに表示します。

メッセージ	内容
Only root user can execute this command.	rootユーザ(Windowsの場合はJobCenter管理者)でコマンドを実行してください。
Too few arguments.	コマンドの引数が不足しています。トラッカアーカイブファイルの取得期間(YYYYMMDD-YYYYMMDD)の引数を指定してください。
no trkdata in YYYYMMDD-YYYYMMDD	指定された期間(YYYYMMDD-YYYYMMDD)にトラッカアーカイブファイルは存在しませんでした。
Invalid time format(YYYYMMDD-YYYYMMDD)	指定された期間のフォーマットが間違っています。YYYYMMDD-YYYYMMDDのフォーマットで指定してください。
Invalid time format(start>end)	指定された取得期間の終了日が開始日より前になっています。終了日を開始日より後にしてください。
Invalid target(target)	指定された取得対象(target)が間違っています。-tで指定できる取得対象を指定してください。
No such a user (user=username)	指定されたユーザは存在しません。指定したユーザ名を確認し、指定ユーザがCL/Winでログイン可能か確認してください。
failed to get tracker[active] user=(user-count) tracker(tracker-count).	取得後情報の表示です。 未アーカイブのトラッカ情報において、取得できなかった情報がありました。 取得できなかったユーザ数をuser-count、取得できなかった未アーカイブのトラッカ情報数をtracker-countに表示します。 jpf_statコマンドで採取した情報を確認し、必要なトラッカ情報がなければ、本コマンドを再度実行してください。
failed to get tracker[archive] user=(user-count) tracker(tracker-count).	取得後情報の表示です。 取得できなかったトラッカアーカイブファイルがありました。　取得できなかったユーザ数をuser-count、取得できなかったトラッカアーカイブ数をtracker-countに表示します。 jpf_statコマンドで採取した情報を確認し、必要なトラッカ情報がなければ、本コマンドを再度実行してください。

3.16.5. 注意事項

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITE、NQS_SITEDBを設定する必要があります。

詳細は1章 「コマンド一覧」 の注意事項を参照してください。

■環境変数LANGについて

UNIX版の場合、コマンド実行時の環境変数LANGはJobCenterのセットアップ言語に合わせてください。

Windows版の場合は、環境変数LANGではなくインストール時に指定したセットアップ言語が利用されるので、特に注意する必要はありません

3.17. jpf_stat JPFファイルの情報確認

■UNIX版

```
/usr/lib/nqs/gui/bin/jpf_stat [-v | -h] $jpf_file

/usr/lib/nqs/gui/bin/jpf_stat -h -a $jpf_folder
```

■Windows版

```
%InstallDirectory%\bin\jpf_stat [-v | -h] $jpf_file

%InstallDirectory%\bin\jpf_stat -h -a $jpf_folder
```

3.17.1. 機能説明

JPFファイルの情報を表示します。出力項目はJPFファイルの種類によって異なります。

3.17.2. オプション

-v

詳細情報を表示します。

-h

JPFファイルに含まれる定義のホスト名変換対象のパラメータに設定されているホスト名の一覧を表示します。

-a

$jpf_folder配下のJPFファイルを対象とする場合に指定します。

本オプションを指定する場合、必ず-hオプションを指定してください。

$jpf_file

JPFファイルを指定します。

本パラメータは-aオプション以外を指定した場合に必須項目となります。

$jpf_folder

JPFファイルが保存されているディレクトリを指定します。

本パラメータは-aオプションを指定した場合に必須項目となります。

3.17.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.17.4. 実行結果項目(共通)

メッセージ	内容
Format type	JPFファイルのバージョン
Created JobCenter version	JPFファイルを作成したJobCenter MG/SVのバージョン
Creation time	JPFファイルの作成日時
Created platform	JPFファイルを作成したOS種類

コマンド実行結果例(JobCenter定義情報のJPFファイルの詳細情報を表示の場合)

表示を一部加工してあります。また「/* */」内は内容説明のためのコメントです。「...」は後半の内容の省略を表しています。

```
./jpf_stat.exe -v jc_def.jpf
=== JPF file status ===
Format type: Definition 7.0
Created JobCenter version: 15.1.0
Creation time: 2017:05:08 14:32:28
Created platform: Windows

=== Each type of information ===

*** JobNetworks(3) *** /* 3はファイルに含まれている定義の個数です */
日次業務1 (Update) /* 定義名とアップロード時に選択される動作を表示しています */
月次業務1 (Update)
月次業務2 (Update)
*** Schedules(3) ***
日次スケジュール (Update)
月次スケジュール (Update)
特殊処理 (Update)
*** Triggers(TRG)(2) ***
アプリケーションAのエラーログ監視 (Update)
アプリケーションBの警告ログ監視 (Update)
*** Triggers(TARGET_TEXTLOG)(3) ***
アプリケーションAのログ (Update)
アプリケーションBのログ (Update)
アプリケーションCのログ (Update)
*** Calendars(1) ***
社内カレンダー (Update)
*** CustomJob Templates(4) ***
ProcessB (Update)
ProcessC (Update)
ProcessE (Update)
EndProcess (Update)
```

コマンド実行結果例(JobCenter定義情報のJPFファイルのホスト名変換対象のホスト名の一覧を表示の場合)

```
./jpf_stat.exe -h jc_def.jpf
=== JPF file hostlist ===
hostname1,hostname2,hostname3
```

コマンド実行結果例(JobCenter構成情報のJPFファイルの場合)

```
./jpf_stat -v jc_conf.jpf

=== JPF file status ===
Format type: Configuration 3.0
Created JobCenter version: 13.1
Creation time: 2012:07:03 12:29:57
Created platform: Linux

=== Each type of information ===

** Configuration Status **

Backup mode: Local

*** Host infomation *** /* ホスト情報*/
hostname: aaa  /* ホスト名 */
mid: 123       /* マシン名 */

*** Queues *** /* キュー状態 */
guilb_def (type=pipe, enable=true, running=true) /* キュー名とキューの状態 */
guinw (type=pipe, enable=true, running=true)
guitp_1 (type=pipe, enable=true, running=true)
guibs_1 (type=batch, enable=true, running=true)

*** Machine List *** /* マシン一覧 */
aaa
bbb

*** Maching groups *** /* マシングループ */
aaa

*** Spool configuration files *** /* ローカル/クラスタサイトの設定ファイル */
jobmsg.conf
umsconf.f
trkcolor.f
...

*** Configuration backup users *** /* ユーザ環境設定をバックアップしているユーザ */
nsumsmgr

*** Machine backup files *** /* インストール環境の設定ファイル */
comagent.sh
destconf.f
jnwcaster.sh
...

*** Backup OS infomation *** /* バックアップしたOSの情報 */
xxxxxx


*** OS Account Infomation *** /* バックアップしたときのユーザ情報 */
root (uid=0)
bin (uid=1)
...
```

コマンド実行結果例(JobCenter実績情報(トラッカ)のJPFファイルの場合)

```
./jpf_stat trkget.jpf
=== JPF file status ===
Format type: Tracker 3.0
Created JobCenter version: 13.1.2
Creation time: 2012:12:21 19:46:28
Created platform: Linux

=== Each type of information ===

** Tracker Status **

====== User (npxjc) ======
-- Tracker [archive] --
  trkarc20121019-20121019
  trkarc20121113-20121113
  trkarc20121119-20121119
  trkarc20121219-20121219
  trkarc20121220-20121220
  trkarc20121221-20121221
-- Tracker [active] --
count : 3

====== User (nsumsmgr) ======
-- Tracker [archive] --
  trkarc20121221-20121221
-- Tracker [active] --
count : 1

====== User (root) ======
-- Tracker [archive] --
  trkarc20121221-20121221
-- Tracker [active] --
count : 1
```

3.18. jc_backup 構成情報のバックアップ

■UNIX版

```
/usr/lib/nqs/gui/bin/jc_backup conf [-c $clusterdb] [-o $output]
```

■Windows版

```
%InstallDirectory%\bin\jc_backup conf [-c $clusterdb] [-o $output]
```

3.18.1. 機能説明

JobCenterの構成情報をバックアップします。　本コマンドは、UNIX版ではrootユーザ、Windows版ではJobCenter管理者で実行してください。 JobCenterが運用状態でも停止状態でも利用できます。 詳細については<環境構築ガイド>の17章 「JobCenter環境のバックアップ・復元」 を参照してください。

3.18.2. オプション

-c $clusterdb

クラスタ構成情報をバックアップする場合に、JobCenterのクラスタDBパスを指定します。　指定しない場合はローカル構成情報をバックアップします。

-o $output

出力ファイル名を指定します。 指定しない場合は、以下のファイル名で出力します。

ローカル構成情報の場合：jc_conf_local_YYYYMMDDhhmmss.jpf

クラスタ構成情報の場合：jc_conf_cluster_YYYYMMDDhhmmss.jpf

3.18.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.18.4. 注意事項

■本コマンドは、OSの管理者権限を持つユーザで実行してください。

■本コマンドは、cmd.exeの右クリックメニューから「管理者として実行」を選択して起動し、コマンドラインから実行してください。

■R15.3より前のバージョンでは、バックアップ・復元機能が想定している使用法は、バックアップした構成情報を元のサーバ（または、物理的には異なるが同一プラットフォームの同名ホスト）へ復元することです。　構成情報をテンプレート的に他のマシンへ配布して、部分的に変更して使用するような用途はサポートしていません。

■R15.3以降のバージョンでは、特定の条件の環境において、jpf_configコマンドによりバックアップした構成情報を基にルールファイルの変換ルールに従いホスト名やユーザ名等のパラメータを変換した構成情報を作成しリストアできる様になりました。

詳細については、<移行ガイド>の3章 「移行先作業」 を参照してください。

3.19. jc_restore 構成情報の復元

■UNIX版

```
/usr/lib/nqs/gui/bin/jc_restore conf [-c $clusterdb] $jpf_file
```

■Windows版

```
%InstallDirectory%\bin\jc_restore conf [-c $clusterdb] $jpf_file
```

3.19.1. 機能説明

JobCenterの構成情報を復元します。 本コマンドは、UNIX版ではrootユーザ、Windows版ではJobCenter管理者で実行してください。 JobCenterが停止状態のとき利用できます。 詳細については<環境構築ガイド>の17章 「JobCenter環境のバックアップ・復元」 を参照してください。

3.19.2. オプション

-c $clusterdb

クラスタ構成情報を復元する場合に、JobCenterのクラスタDBパスを指定します。 指定しない場合は、ローカル構成情報として復元します。

jpf_file

構成情報のJPFファイル（jc_backupコマンドで作成したバックアップファイル、または、jpf_configコマンドで変換した構成情報のファイル）を指定します。 本パラメータは必須項目です。

3.19.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.19.4. 注意事項

■本コマンドは、OSの管理者権限を持つユーザで実行してください。

■本コマンドは、cmd.exeの右クリックメニューから「管理者として実行」を選択して起動し、コマンドラインから実行してください。

■JobCenterで使用する以下のユーザが、復元環境にも存在する必要があります。

- jc_backupコマンドで作成した構成情報のバックアップファイルの場合には、バックアップ時にJobCenterで使用していたユーザ
- jpf_configコマンドで変換した構成情報のファイルの場合には、変換後の構成情報のJobCenterで使用するユーザ

構成情報のJPFファイルのユーザの情報は、jpf_statコマンドで確認できますので、必要なユーザを復元環境に登録してください。このとき、ユーザ名だけでなく、UIDも構成情報のJPFファイルの内容と一致させる必要があります。

UNIX環境のユーザの追加/変更については、OSのコマンドでできます。

Windows環境のユーザの追加/変更については、<環境構築ガイド>の「13.4 ユーザの設定」を参照してください。

jpf_statコマンドの詳細については、「3.17 jpf_stat JPFファイルの情報確認」を参照してください。

■R15.3より前のバージョンでは、バックアップ・復元機能が想定している使用法は、バックアップした構成情報を元のサーバ（または、物理的には異なるが同一プラットフォームの同名ホスト）へ復元することです。 構成情報をテンプレート的に他のマシンへ配布して、部分的に変更して使用するような用途はサポートしていません。

■R15.3以降のバージョンでは、特定の条件の環境において、jpf_configコマンドによりバックアップした構成情報を基にルールファイルの変換ルールに従いホスト名やユーザ名等のパラメータを変換した構成情報を作成しリストアできる様になりました。

詳細については、<移行ガイド>の3章 「移行先作業」 を参照してください。

3.20. jpf_config 構成情報のパラメータを変換

■UNIX版

```
/usr/lib/nqs/gui/bin/jpf_config update -f $rulefile [-o $output] $jpf_file
```

■Windows版

```
%InstallDirectory%\bin\jpf_config update -f $rulefile [-o $output] $jpf_file
```

3.20.1. 機能説明

バックアップした構成情報を基にルールファイルの変換ルールに従いホスト名やユーザ名等のパラメータを変換した構成情報を作成します。対象となる構成情報については、<環境構築ガイド>の「17.2.2　　バックアップ・復元対象の構成情報」を参照してください。本コマンドは、UNIX版ではrootユーザ、Windows版ではJobCenter管理者で実行してください。

3.20.2. オプション

-f $rulefile

ホスト名やユーザ名等のパラメータの変換ルールを記載したルールファイルを指定します。本パラメータは必須項目です。

-o $output

ルールファイルに従いパラメータを変換した構成情報の出力ファイル名を指定します。指定しない場合は、以下のファイル名で出力します。

ローカル構成情報の場合：jc_conf_local_YYYYMMDDhhmmss.jpf

クラスタ構成情報の場合：jc_conf_cluster_YYYYMMDDhhmmss.jpf

■同名のファイル名が既に存在する場合は上書きします。

■$jpf_fileと同じファイルは指定できません。

$jpf_file

構成情報のバックアップファイル（jc_backupコマンドで作成したJPFファイル）を指定します。本パラメータは必須項目です。

3.20.3. ルールファイル

ルールファイルはテキスト形式で、変換ルールで指定できるパラメータは以下です。

■ホスト名

■IPアドレス

■構成情報をバックアップしたサイトのユーザ名

■マシンID

■ユーザID

■グループID

3.20.3.1. ホスト名の変換ルール

ホスト名の変換ルールは項目間を半角スペース(複数可)またはタブ区切りで以下のように記載します。

```
hostname 変換元ホスト名1 変換先ホスト名1

hostname 変換元ホスト名2 変換先ホスト名2

・

・
```

■ホスト名は255バイトまで入力することができます。また、空白、タブ、半角カタカナ、全角文字、特殊文字、及び「!」「"」「#」「$」「%」「&」「'」「(」「)」「*」「,」「/」「;」「<」「=」「>」「?」「@」「[」「\」「]」「^」「`」「{」「|」「}」「~」は使用できません。

■ホスト名のFQDNとショートネームを区別します。FQDNとショートネームの両方を使用している場合には、FQDNとショートネームの変換ルールを記載してください。

■UNIXではホスト名の大文字・小文字を区別します。

3.20.3.2. IPアドレスの変換ルール

IPアドレスの変換ルールは項目間を半角スペース(複数可)またはタブ区切りで以下のように記載します。

```
ipaddress 変換元IPアドレス1 変換先IPアドレス1

ipaddress 変換元IPアドレス2 変換先IPアドレス2

・

・
```

■IPアドレスの変換は、IPアドレスの文字列が一致したものを変換します。よって、IPv6アドレスの場合にバックアップした構成情報のJPFファイルに保存されている表記文字列と一致する表記文字列でルールファイルに記載する必要があります。

■UNIXではIPアドレスの大文字・小文字を区別します。

3.20.3.3. 構成情報をバックアップしたサイトのユーザ名の変換ルール

構成情報をバックアップしたサイトのユーザ名の変換ルールは項目間を半角スペース(複数可)またはタブ区切りで以下のように記載します。

```
username 変換元ユーザ名1 変換先ユーザ名1

username 変換元ユーザ名2 変換先ユーザ名2

・

・
```

■ユーザ名は15バイトまで入力することができます。また、空白、タブ、特殊文字、及び「!」「"」「#」「$」「%」「&」「'」「(」「)」「*」「,」「.」「/」「:」「;」「<」「=」「>」「?」「@」「[」「\」「]」「^」「`」「{」「|」「}」「~」は使用できません。

■UNIXではユーザ名の大文字・小文字を区別します。

3.20.3.4. マシンIDの変換ルール

マシンIDの変換ルールは項目間を半角スペース(複数可)またはタブ区切りで以下のように記載します。

```
mid 変換元マシンID1 変換先マシンID1

mid 変換元マシンID2 変換先マシンID2

・

・
```

3.20.3.5. ユーザIDの変換ルール

ユーザIDの変換ルールはマシン一覧に登録されているマシンの変換元のマシンID別に項目間を半角スペース(複数可)またはタブ区切りで以下のように記載します。

```
uid 変換元マシンID1 変換元ユーザID1 変換先ユーザID1

uid 変換元マシンID1 変換元ユーザID2 変換先ユーザID2

uid 変換元マシンID2 変換元ユーザID1 変換先ユーザID1

・

・
```

3.20.3.6. グループIDの変換ルール

グループIDの変換ルールはマシン一覧に登録されているマシンの変換元のマシンID別に項目間を半角スペース(複数可)またはタブ区切りで以下のように記載します。

```
gid 変換元マシンID1 変換元グループID1 変換先グループID1

gid 変換元マシンID1 変換元グループID2 変換先グループID2

gid 変換元マシンID2 変換元グループID1 変換先グループID1

・

・
```

3.20.3.7. ルールファイルの記載例

ルールファイルの記載例を以下に示します。

```
hostname host1 newhost01
hostname host1.local newhost01.local
hostname host2 newhost02
hostname host2.local newhost02.local
ipaddress 192.168.0.1 192.168.100.1
ipaddress 192.168.0.2 192.168.100.2
mid 77001 1
mid 77002 2
username test01 user01
username test02 user02
uid 77001 501 1001
```

```
uid 77001 502 1002
uid 77002 501 1001
uid 77002 502 1002
gid 77001 301 601
gid 77001 302 602
gid 77002 301 701
gid 77002 302 702
```

3.20.4. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.20.5. 主要メッセージ

メッセージ	内容
update successfully.(output=filename)	構成情報の変換に成功しました。 変換後の構成情報のJPFファイルをfilenameに表示します。
Only JobCenter admins(adminuser) can execute this command.	Windows版ではJobCenter管理者で実行してください。 adminuserにはJobCenter管理者ユーザを表示します。
Only root user can execute this command.	UNIX版ではrootユーザで実行してください。
Invalid subcommand. Usage:jpf_config update -f $rulefile [-o $output] $jpf_file	jpf_configコマンドのサブコマンドが正しくありません。 サブコマンドにはupdateを指定してください。
Usage:jpf_config update -f $rulefile [-o $output] $jpf_file	コマンドの引数が正しくありません。
The output file cannot be the same as backup file.	-oオプションで指定した構成情報の出力ファイルと指定した構成情報のバックアップファイルが同じです。
ERROR: No such rulefile "filename".	指定したルールファイルが存在しません。 filenameには指定したルールファイルを表示します。
ERROR: Fail to read rulefile: Invalid key. (file=filename, key=paramname)	ルールファイルの変換ルールのパラメータ名が正しくありません。 filenameには指定したルールファイルを表示します。 paramnameにはエラーとなった正しくないパラメータ名を表示します。
ERROR: Fail to read rulefile: Invalid rule format. (file=filename, line=linenumber)	ルールファイルの変換ルールの書式が正しくありません。 filenameには指定したルールファイルを表示します。 linenumberにはエラーとなったルールファイルの行番号を表示します。
ERROR: Fail to Check rules:詳しいエラー	ルールファイルの変換ルールのパラメータの設定値が正しくありません。

メッセージ	内容
ERROR: platform does not support.	構成情報をバックアップしたプラットフォームとは異なるプラットフォームで構成情報の変換は実行できません。

3.20.6. 注意事項

■R15.3より前のバージョンのJobCenter(MG/SV)でバックアップした構成情報のJPFファイルの変換はサポート外となります。

■構成情報をバックアップしたプラットホームとは異なるプラットフォームでは構成情報の変換は実行できません。(jpf_configコマンド実行時にエラーとなります)

		jpf_configコマンドを実行するプラットホーム			
		Linux(x64)	HP-UX	AIX	Windows(x64)
構成情報をバックアップしたプラットホーム	Linux(x64)	○	×	×	×
	HP-UX	×	○	×	×
	AIX	×	×	○	×
	Windows(x64)	×	×	×	○

■異なるプラットフォームの構成情報のJPFファイルには変換できません。(LinuxからHP-UX等)

■ローカルサイトからクラスタサイト、クラスタサイトからローカルサイトへの構成情報のJPFファイルには変換できません。

■JobCenter(MG/SV)のセットアップ言語の変換はできません。

■対象の構成情報のうち、以下の設定ファイルについては本コマンドの変換対象外となります。

UNIX版	saprfc.ini
	jcdbs.conf
Windows版	saprfc.ini
	jcdb.conf

■ルールファイルには以下の制限があります。

- コメントを記述することはできません。
- 変換元の値が同じ場合、先(ルールファイルの行頭に近い方)に記載した変換ルールが使用されます。(後の記載は無視されます)

3.21. jdh_conv JDFファイルとJPFファイルの相互変換

■UNIX版

```
/usr/lib/nqs/gui/bin/jdh_conv [-w $version] [-c] [-o $output] [$JDFFile ...]

/usr/lib/nqs/gui/bin/jdh_conv -x [-o $output] [$JPFFile]
```

■Windows版

```
%InstallDirectory%\bin\jdh_conv [-w $version] [-c] [-o $output] [$JDFFile ...]

%InstallDirectory%\bin\jdh_conv -x [-o $output] [$JPFFile]
```

3.21.1. 機能説明

JDFファイルから定義情報のJPFファイルを生成を行います。-xオプションを指定した場合はJPFファイルからJDFへの変換を行います。

3.21.2. オプション

オプション	説明
-w $version	生成するJPFのバージョンを"x.x"の形式で指定します。現在は"1.0","2.0","3.0","4.0","5.0","6.0","7.0"が指定可能です。省略した場合はjdh_convが対応している最新のバージョンで生成します。 ■12.8.2～12.10のバージョンでは互換性のチェックは行われていません。 ■7.0より前のバージョンを指定した場合、起動トリガ定義、および監視対象テキストログは生成するJPFファイルに含まれません。
-c	JPFファイルの作成は行わず、JDFファイルのチェックのみ行います。 12.8.2～12.10のバージョンではチェックは部分的です。
-o $output	出力先を指定します。 本オプションを指定しない場合、「jc_def_<YYYYMMDDhhmmss形式の実行日時>.jpf」というファイル名で出力します。-xオプション指定時のデフォルトの出力先は「jpf_YYYYMMDDHHMMSS」になります。
$JDFFile ...	JPFに変換したいJDFファイルを指定します。customjob_templateタイプ、jnwタイプ、shcedタイプ、trgタイプ、wkcalタイプは全て指定する必要はなく定義を作成したものだけ指定可能です。また、オペレーションファイル(operation.jdf)のみで指定することも可能です。 (例1) $jdh_conv customjob_template.jdf jnw.jdf sched.jdf trg.jdf wkcal.jdf operation.jdf (例2) $jdh_conv jnw.jdf operation.jdf (例3) $jdh_conv operation.jdf jdh_convコマンドで変換を行う際、オペレーションを定義したオペレーションファイル（operation.jdf）は必ず指定してください。

オプション	説明
-x	JPFからJDFへの変換を行います。結果の構成は以下のとおりです。 `jpf/` ` |-- def/` ` |  |-- customjob_template.jdf //customjob_templateタイプのJDF` ` |  |-- jnw.jdf //jnwタイプのJDF` ` |  |-- sched.jdf //schedタイプのJDF` ` |  |-- trg.jdf //trgタイプのJDF` ` |  `-- wkcal.jdf //wkcalタイプのJDF` ` `-- operation/` `  `-- operation.jdf //operationタイプのJDF` 各JDFファイルは、定義がない場合は存在しません。
$JPFFile	JDFに変換したいJPFファイルを指定します。

3.21.3. 戻り値

戻り値	内容
0	正常終了です。
1	異常終了です。

3.21.4. 主要メッセージ一覧

メッセージ	内容
JPF file "jc_def_20111028162232.jpf" conversion successfully.	JDFファイルからJPFファイルの作成に成功しました。作成ファイル名は"jc_def_20111028162232.jpf"です。
Output JDFs to "jpf_20111028162342/jpf".	JPFファイルからJDFファイルへの変換に成功しました。出力先は"jpf_20111028162342/jpf"です。
Convert check successfully.	JDFのチェックが成功しました。
ERROR: test.jdf:2: Invalid line format. "param="	このエラーメッセージは行のJDFフォーマットが間違っている場合の一般的なメッセージです。該当ファイルの行を確認してJDFフォーマットに従っているか確認してください。上記は「test.jdfファイルの2行目のフォーマットが不正です。2行目の内容は"param="です。」と言う意味になります。
ERROR: test.jdf: No body section.	JDFファイルにボディセクションが存在しません。JDFファイルには必ずボディセクションが必要なので指定してください。具体的には「%%%」セパレータが必要になります。
ERROR: test.jdf:1: No "version" key in header section.	ヘッダセクションにversionキーが存在しません。JDFファイルではヘッダセクションにversionキーは必ず指定してください。
ERROR: test.jdf:1: No "type" key in header section.	ヘッダセクションにtypeキーが存在しません。JDFファイルではヘッダセクションにtypeキーは必ず指定してください。
ERROR: test.jdf:4: No "id" key	オブジェクトに必須なキーが存在しません。必須なキーを指定してください。この場合は「test.jdfファイルの4行目付近のオブジェクトに必要なidキーが存在しません」という意味になります。

メッセージ	内容
ERROR: test.jdf:5: Unknown object type(soso)	未定義のキーが指定されています。この場合は「soso」という存在しないキーが指定されています。
ERROR: test.jdf:4: pre_parts (1) not found for obj 3	JNWフローの依存関係エラーです。この場合は「部品のid=3の前部品に指定されているid=1の部品が存在しません。」という依存関係エラーになります。
ERROR: test.jdf:9: Duplicate object id (3)	部品のidが重複しています。この場合は「test.jdfファイルの9行目でid=9が重複しています」と言う意味になります。
WARNING: test.jdf: JDF of Launch Trigger was ignored. This JDF format is available in version 7.0 or later.	JDFのバージョンが7.0より前のバージョンが指定されたため、起動トリガ定義、および監視対象テキストログはJPFファイルに含まれません。
WARNING: test.jdf: Operation of Launch Trigger was ignored. This type is available in version 7.0 or later.	JDFのバージョンが7.0より前のバージョンが指定されたため、起動トリガ定義、および監視対象テキストログのアップロード時の操作が無視されました。

3.21.5. 注意事項

■12.8.2～12.10に同梱されているjdh_convは現在一般ユーザには非公開機能であり、以下の制限事項があります。本コマンドを使用の際には制限事項について理解した上でご使用ください。

- JDFのフォーマットのエラーチェックが十分ではありません。使用の際には、JDFのフォーマットが正確に記載されているかを確認して、注意してご使用ください。
- JPFバージョンの互換性のチェックは実装されていません。互換性に関してはJDFファイル作成時に十分確認して作成してください。
- JDFのフォーマット、JPFバージョンの互換性に関しては、<テキスト定義機能の利用手引き>を参照してください。

■不正なJDFファイルから作成したJPFファイルをサーバにアップロードした場合、予期せぬ不具合が生じる可能性があります。

3.22. sleep Windows版sleepコマンド

■Windows版

```
%InstallDirectory%\bin\utils\sleep.exe  [$second]
```

3.22.1. 機能説明

sleepを行うコマンドをWindowsで提供するコマンドです。　本コマンドで指定した秒数だけ動作を停止します。

3.22.2. オプション

$second 動作を停止する秒数（整数）を指定します。 0から2147483647の範囲で指定します。

3.22.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.22.4. 注意事項

■指定する秒数は整数で入力してください。（マイナスの値を入れるとエラーとなります）

２つ以上の引数を指定しないでください。（こちらも、エラーとなります）

3.23. spoolconv R12.10.x以前のユーザ定義情報を移行

■UNIX版

```
/usr/lib/nqs/gui/bin/spoolconv [-c $clusterdb]
```

■Windows版

（無し）

3.23.1. 機能説明

R12.10.x 以前の環境で構築したユーザ定義をR13.1へ移行します。UNIX版では本コマンドを使用して移行を行う必要が有りますが、Windows版では自動で移行が行われます。

3.23.2. オプション

-c $ clusterdb

クラスタサイトの定義を移行する場合に、JobCenterのクラスタDBパスを指定します。指定しない場合は、ローカルサイトを対象として移行します。本オプションを指定する場合は、環境変数NQS_SITEの設定が必要です。

3.23.3. 戻り値

戻り値	内 容
0	正常終了です。
1	異常終了です。

3.23.4. 注意事項

■本コマンドはrootユーザで実行してください。

3.24. jc_iedata_conv エクスポートデータからJPFファイルへの変換

■UNIX版

```
/usr/lib/nqs/gui/bin/jc_iedata_conv [-o $output] [-r $rulefile] $exportdata_dir
```

■Windows版

```
%InstallDirectory%\bin\jc_iedata_conv.exe  [-o $output] [-r $rulefile] $exportdata_dir
```

3.24.1. 機能説明

エクスポートデータからJPF形式に変換します。変換時にルールファイルを指定することで、ホスト名変換を行うことができます。

以下にUNIX版の実行例を示します。

```
$ /usr/lib/nqs/gui/bin/jc_iedata_conv -r rulefile -o /tmp/jpf /usr/spool/nqs/users/nsumsmgr/
exportfile/JNW_jcserver_201304161001001
convert successfully.(output=/tmp/jpf)
```

3.24.2. オプション

-o $output

変換結果の出力先ディレクトリを指定します。本オプションを省略した場合、出力先ディレクトリは、コマンドを実行したカレントディレクトリ配下のjc_YYYYMMDDhhmmssというディレクトリになります。

YYYYMMDDhhmmssは、コマンド実行時の日時になります。

-r $rulefile

ホスト名変換用のルールファイル名を指定します。本オプションを省略した場合、ホスト名変換は行いません。

ホスト名変換の対象パラメータは、以下になります。

■ジョブネットワークのデフォルト投入キューのホスト名部分

■単位ジョブ、WOBSジョブ、カスタムジョブの投入キューのホスト名部分

■イベント送信部品の送信先ホスト名

■イベント受信部品の正常終了条件、および異常終了条件の送信元ホスト名

$exportdata_dir

変換対象のエクスポートデータディレクトリを指定します。

3.24.2.1. ルールファイル

ホスト名の変換定義を半角スペース区切りで以下のように記述します。

変換元ホスト名1 変換先ホスト名1

変換元ホスト名2 変換先ホスト名2

・

.

■変換先ホスト名の最大バイト数は、63バイトです。

■ホスト名にタブ、半角カタカナ、全角文字および以下の文字は使用できません。また、ホスト名の先頭文字に数字は使用できません。

! " # $ % & ' ( ) * , / : ; < = + > ? @ [\ ] ^ ` { | } ~

3.24.3. 戻り値

戻り値	内 容
0	正常終了です。
1	異常終了です。

3.24.4. 主要メッセージ

メッセージ	内容
Too many arguments.	引数の数が多いです。引数の指定数が正しいか確認してください。
Too few arguments.	引数の数が少ないです。引数の指定数が正しいか確認してください。
No such data_dir path "JNW_jcserver_20130415091900".	指定したディレクトリは存在しません。エクスポートデータを格納したディレクトリパスが正しいか確認してください。
Invalid export data dir "JNW_jcserver_20130415091900".	指定したディレクトリ配下のエクスポートデータは不正です。 指定したディレクトリ配下に以下の２ファイルが存在することを確認してください。 ■エクスポートデータファイル 例：JNW_jcserver_20130415091900.zip ■属性定義ファイル 例：JNW_jcserver_20130415091900.param
Invalid rule format(line=1)	指定したルールファイルは不正です。ルールファイル内の１行目のホスト名変換の指定方法が正しいか確認してください。
Invalid character(source machine name).	指定したルールファイル内の変換元ホスト名が不正です。指定したルールファイル内の変換元ホスト名に使用できない文字が含まれていないか確認してください。
Invalid character(destination machine name).	指定したルールファイル内の変換先ホスト名が不正です。指定したルールファイル内の変換先ホスト名に使用できない文字が含まれていないか確認してください。
No such rulefile path "rulefile".	指定したルールファイルは存在しません。ルールファイルパスが正しいか確認してください。
Cannot convert ExportData.(DataLang=<エクスポートデータの文字コード>, TermLang=<端末の文字コード>)	変換サポート外のエクスポートデータが指定されたため、変換を中止しました。

メッセージ	内容
	本メッセージは、Windows版のみのメッセージになります。 Windows版の変換サポート可否の詳細は、注意事項を参照してください。
convert successfully. (output=jc_20130415202745)	エクスポートデータからJPFファイルの変換に成功しました。 JPFファイルは、jc_20130415202745ディレクトリ配下にユーザごとのJPFファイルが作成されます。

3.24.5. 注意事項

■エクスポートデータ内で以下のような問題がある場合、JPFファイルに変換はできますが、 JPFファイルのアップロードに失敗します。定義の移行時は、依存関係のある定義を含めてエクスポートしてください。

サブジョブネットワーク部品として参照するジョブネットワークの定義がエクスポートデータ内に存在しない

カレンダ分岐部品として参照するスケジュールの定義がエクスポートデータ内に存在しない

スケジュール起動を行うスケジュールの定義がエクスポートデータ内に存在しない

スケジュールから参照する稼働日カレンダの定義がエクスポートデータ内に存在しない

■カレンダ情報の変換結果を格納するユーザは以下の通りです。定義移行時は、変換結果の出力先ディレクトリをアップロード対象の ディレクトリとして指定し、全ユーザを対象にアップロードを実行してください。

UNIX版の場合、nsumsmgrのJPFファイルに格納されます。

Windows版の場合、JobCenter管理者のJPFファイルに格納されます。

■Windows版のエクスポートデータ変換のサポート可否について

Windows版の場合、変換対象のエクスポートデータの文字コードによって、変換を中止するパターンがあります。

エクスポートデータの変換対応表は、以下の通りです。

○:変換可能、×:変換不可能

表3.14 Windows版jc_iedata_convコマンドの変換対応表

エクスポートデータの文字コード	日本語OS環境(SJIS)	中国語OS環境(GB18030)	英語OS環境
ASCII(C)	○	○	○
SJIS	○	×	×
GB18030	×	○	×
EUC	×	×	×
UTF-8	×	×	×

3.25. jc_perf 実行状況の分析

■UNIX版

```
/usr/lib/nqs/gui/bin/jc_perf config [-s $server] [$key=$val]

/usr/lib/nqs/gui/bin/jc_perf stat [-q] $data_file

/usr/lib/nqs/gui/bin/jc_perf view { qreq $qtype $qname | rtrk | rjob [-a $jobtype]  } [-s $server]
[-f {csv | rd}] [-v {0 | 1 | 2}] [-n] [-d $date] [-p $period] [-i $interval] [-e $extdir]
```

■Windows版

```
%InstallDirectory%\bin\jc_perf config [-s $server] [$key=$val]

%InstallDirectory%\bin\jc_perf stat [-q] $data_file

%InstallDirectory%\bin\jc_perf view { qreq $qtype $qname | rtrk | rjob [-a $jobtype]  } [-s $server]
[-f {csv | rd}] [-v {0 | 1 | 2}] [-n] [-d $date] [-p $period] [-i $interval] [-e $extdir]
```

3.25.1. 機能概要

JobCenterの実行状況について、データを収集し解析を行います。

本コマンドにより、データ収集のON/OFFの切り替え、設定の確認、及び、データの解析を行うことができます。収集できるデータは以下の3種類です。

■トラッカの実行状況（rtrkデータ）

ジョブネットワークを実行開始する場合、フローや各部品のデータはメモリ上に展開され、実行完了後にアーカイブされるとメモリから破棄されます。従って、同時に実行させるジョブネットワークが多い場合には、JobCenterのメモリ使用量は増加することになります。

通常は事前のサイジングにより、ピーク時間帯やそのときのメモリ使用量を見積りますが、実際には、運用開始後の定義追加等により、当初の見積りを超えてしまう場合があります。この場合、メモリ使用量やCPU使用率などのリソース監視についてはOSの標準機能等により確認可能ですが、どの時間にどの程度のジョブネットワークが稼働中だったかについては、その実行結果(トラッカ)を確認する必要があります。

本データを収集することで、このようなトラッカの実行状況を確認することが可能になります。

■フロー上で投入可能状態となっているジョブ数（rjobデータ）

ジョブネットワークが実行開始されてフローがジョブ部品に到達した場合、その投入処理が行われますが、投入可能な状態（前段の部品が実行完了している状態）となっているジョブ部品が複数存在する場合には、まったくの同時に処理が行われるわけではなく、内部的には最初に投入可能状態となったジョブ部品から順次投入処理が行われることになります。

従って、並列度の高い分岐に到達して同時に多数のジョブ部品の投入処理を行わなければならない場合や、大量のジョブネットワークを同時に稼働させているために、同様に多数のジョブ部品の投入処理が発生している場合には、前段の部品が実行完了しているにもかかわらず後続のジョブ部品がなかなか始まらない（WAITからSUBMITに遷移しない）という状況が発生することがあります。

単位時間あたりに投入可能なジョブ数は対象サーバのスペックによりある程度決まってしまうため、それを超える量が同時に発生した場合には遅延が発生することになります。このようなフロー上の部品と部品の間が開いてしまう遅延現象は、もちろんメモリやCPU等のリソース枯渇によっても発生しますが、リソースが十分にある状況下でも前述のような理由により発生する場合があります。

本データを収集することで、このような遅延現象を発生させる要因となる、投入可能状態となっているジョブ数を確認することができます。

■NQSにキューイングされたリクエストの滞留状況（qreqデータ）

フローが単位ジョブ部品に到達するとその投入処理が行われ、指定されたキューにキューイングされますが、キュー内ではいくつかの状態を遷移し、キューの同時実行可能数に空きがあれば実行が開始されることになります。フロー上で対象のジョブがSUBMITと表示されていて、なかなか実行が始まらない(RUN状態に変わらない)場合、投入先のキューの同時実行可能数に空きがなく、キューの中で順番待ちをしていることが考えられます。

本データを収集することで、このようなキューの滞留状況を確認することができます。

3.25.2. 機能説明

jc_perfコマンドには3つのコマンドモードがあります。ここではそれぞれのコマンドモードの使い方とオプションについて説明します。

3.25.2.1. 設定の確認・変更（config）

■概要

第1引数にconfigを指定することで、データ収集機能の設定の確認と変更を行うことができます。

■オプション

-s $server

リモートサーバを指定します。指定しない場合はローカルサーバが対象となります。

設定確認・変更を行いたいサーバがリモートのサーバであれば本オプションを指定し、現在ログインしているローカルサーバが対象であれば本オプションを指定しないようにしてください。

なお、ローカルサーバ上にローカルサイトとクラスタサイトが存在する場合には、環境変数NQS_SITEによって区別されます。

$key=$val

指定したパラメータ($key)を指定した値($val)に変更します。指定可能なパラメータ名と値は以下の通りです。

パラメータ名($key)	設定可能な値($val)	初期値	説明
qreq	onまたはoff	off	qreqデータを収集するかどうかを指定します。
rtrk	onまたはoff	off	rtrkデータを収集するかどうかを指定します。
rjob	onまたはoff	off	rjobデータを収集するかどうかを指定します。
file_size	64～1024	128	データ保存先となるファイル(データファイル)の、1ファイルの最大サイズをMB単位で指定します。
file_lifetime	1～365	90	データファイルの保存期間を日で指定します。

$key=$valを指定した場合は設定変更が行われますが、指定しない場合には設定確認のみとなります。

configモードによる設定の確認・変更を行う場合、対象のJobCenterが起動している必要があります。

■出力例

configモードを指定した場合、標準出力に以下のような出力が行われます。

```
server: jobmanager.nec.com
```

```
collect(qreq): off
collect(rtrk): off
collect(rjob): off
file_size: 128 MB
file_lifetime: 90 days
```

各行の出力は$keyで指定するパラメータに対応しています。

■データファイルについて

収集されたデータは以下のファイルに保存されます。

格納先ディレクトリ

UNIX版	ローカルサイト	/usr/spool/nqs/perf/<YYYYMMDD>
	クラスタサイト	<クラスタDBパス>/perf/<YYYYMMDD>
Windows版	ローカルサイト	%InstallDirectory\spool\perf\%<YYYYMMDD>
	クラスタサイト	<クラスタDBパス>\perf\<YYYYMMDD>

ファイル名

qreqデータ	qreq_<n>.dat
rtrkデータ	rtrk_<n>.dat
rjobデータ	rjob_<n>.dat

格納先ディレクトリの<YYYYMMDD>は日付を表しており、1日単位でディレクトリ管理されています。このディレクトリ配下に各種データファイルが作成されますが、ファイル名の<n>は1以上の整数であり、1日のうちで1つのデータファイルのサイズがfile_sizeパラメータの値を超えるような場合に2,3,...と作成されます。

また、データファイルはプラットフォーム非依存のバイナリ形式で保存されており、データを参照・解析するにはviewモードを利用する必要があります。viewモードの詳細については「3.25.2.3 データの参照・解析（view）」を参照してください。

データファイルを他サーバへコピーする場合には、必ず日付ディレクトリ単位でコピーしてください。

3.25.2.2. データファイルの確認（stat）

■概要

第1引数にstatを指定することで、保存されたデータのファイルのプロパティ情報を確認することができます。

■オプション

-q

データファイルに含まれているキュー名を一覧表示します。$data_fileにqreqデータを指定した場合のみ有効です。

viewモードでデータを解析する場合、キュー種別とキュー名が必要ですが、それが不明な場合にはこの-qオプションで事前に確認してください。

$data_file

プロパティ情報を参照したいデータファイルを指定します。本パラメータは必須項目です。

■出力例

statモードで実行した場合、標準出力に以下のような出力が行われます。(-qオプションを指定した場合の例)

```
=== Data file status ===
Server: jobmanager.nec.com
JobCenter version: 13.2
Platform: Linux
Data type: qreq
Data version: 100
Data size: 44
Number of data: 1637
Creation time: 2013/09/20 01:03:12
File date: 20130920
File id: 1

=== Queue list ===
--- Pipe queue ---
guilb_def
guinw
guitp_1
pipe1
pipe2
--- Batch queue ---
batch1
batch2
guibs_1
--- Network queue ---
DefaultNetQue
```

Data file statusの出力項目は以下の通りです。

Server	データ収集元のサーバ名です。
JobCenter version	JobCenterのバージョンです。
Platform	プラットフォーム種別です。HPUX,AIX,Linux,Windowsのいずれかになります。
Data type	データ種別です。qreq,rtrk,rjobのいずれかになります。
Data version	データのバージョンです。
Data size	1データのサイズ(byte)です。
Number of data	このデータファイルに含まれるデータ数です。
Creation time	データファイルの作成時刻です。
File date	データファイルの日付です。格納先ディレクトリの<YYYYMMDD>に一致します。
File id	データファイルのファイルIDです。データファイル名の<n>に一致します。

Queue listには対象データファイルに含まれるキュー一覧が出力されます。

3.25.2.3. データの参照・解析（view）

■概要

第1引数にviewを指定することで、収集されたデータの解析をすることができます。

第2引数には必ずデータ種別を表す以下のいずれかの識別子を指定する必要があります。

qreq	qreqデータの解析を行います。

	qreqを指定する場合、続けて解析対象となるキュー種別($qtype)とキュー名($qname)を指定する必要があります。 $qtypeには以下のいずれかを指定します。 ■ pipe: パイプキュー ■ batch: バッチキュー ■ network: ネットワークキュー $qnameにはキュー名を指定します。キュー名の形式は<キュー名>@<ホスト名>となっていますが@以降は指定せずにキュー名のみを指定します。 また、statモードでデータファイルに含まれるキュー一覧を確認することも可能です。この方法については「3.25.2.1 設定の確認・変更（config）」を参照してください。
rtrk	rtrkデータの解析を行います。
rjob	rjobデータの解析を行います。 デフォルトの解析対象は単位ジョブのみですが、-a　$jobtypeを指定することで追加のジョブ種別を指定することができます。 $jobtypeには以下を指定することができます。 ■ sap: SAP系ジョブ部品(ERPジョブ、BIジョブ、PCジョブ)

■その他のオプション

-s $server

リモートサーバを指定します。指定しない場合はローカルサーバが対象となります。

解析を行いたいサーバがリモートのサーバであれば本オプションを指定し、現在ログインしているローカルサーバが対象であれば本オプションを指定しないようにしてください。

なお、ローカルサーバ上にローカルサイトとクラスタサイトが存在する場合には、環境変数NQS_SITEによって区別されます。

-eオプションと同時には指定できません。

-f

解析結果の出力形式を指定します。以下のいずれかを指定します。

■ rd: 可読形式で出力します

■ csv: CSV形式で出力します

本オプションを指定しない場合可読形式での出力となります。

それぞれの出力レベルでの出力フォーマットについては「3.25.2.4　viewモードの出力フォーマット」を参照してください。

-v

解析結果の出力レベルを指定します。以下のいずれかを指定します。

- 0: 対象データのIN/OUT数
- 1: ステータス毎の時間平均値
- 2: 0及び1で出力される全情報、及び、時間毎の最大値/最小値

本オプションを指定しない場合、0指定と同じ出力となります。

それぞれの出力レベルでの出力フォーマットについては「3.25.2.4　viewモードの出力フォーマット」を参照してください。

レベル2の場合、出力する情報量が多いため可読形式での表示は行えません。レベル2を指定する場合、必ず-fオプションでCSV形式を指定してください。

-n

解析結果の出力時に、タイトル行(ヘッダ行)を出力させたくない場合に指定します。

-d $date

解析したいデータの日時をYYYYMMDDの形式で指定します。

例えば、2013年9月20日のデータを解析したい場合、20130920を指定します。

本オプションを指定しない場合、当日のデータが対象となります。

-p $period

解析したいデータの期間をhhmm-hhmmの形式で指定します。指定可能範囲は0000-2400です。

例えば、12:00-18:00のデータを解析したい場合、1200-1800と指定します。この場合、12:00:00.000から17:59:59.999までに記録されたデータが対象となります。(18:00:00.000ちょうどのデータは含まれません)

当日のデータを解析する場合、現在時刻以降のデータ出力は行われませんので、出力結果が$periodで指定した期間よりも短くなる場合があります。

本オプションを指定しない場合、指定日の全データが対象となります。

-e $extdir

対象サーバ上のデータではなく、別のサーバからコピーしたデータを解析する場合に指定します。

$extdirで指定するディレクトリは日付ディレクトリ（YYYYMMDD）の1つ上のディレクトリを指定してください。データファイルの格納ディレクトリについては「データファイルについて」を参照してください。

-sオプションと同時には指定できません。

3.25.2.4. viewモードの出力フォーマット

viewモードの出力フォーマットについて説明します。

以降の説明では、全て-iオプション(interval)を60分で指定した場合の出力例として説明します。

■qreq、レベル0

可読形式

```
DATE                    IN      OUT
==================================
2013/09/20 00:00        48       48
2013/09/20 01:00        31       31
2013/09/20 02:00         0        0
2013/09/20 03:00        11       10
2013/09/20 04:00         8        9
2013/09/20 05:00         0        0
...
```

CSV形式

```
DATE,IN,OUT
2013/09/20 00:00,48,48
2013/09/20 01:00,31,31
2013/09/20 02:00,0,0
2013/09/20 03:00,11,10
2013/09/20 04:00,8,9
2013/09/20 05:00,0,0
...
```

IN/OUTの見方は以下の通りです。

IN	その時間帯に、対象キューにエンキューされた(キューに入った)数です。
OUT	その時間帯、対象キューからデキューされた(キューから出た)数です。

■qreq、レベル1

キューの種別によってリクエストの取りうるステータスが異なるため、出力フォーマットも異なります。

それぞれの数値は、リクエストがそのステータスで存在していた時間平均を表しています。TOTALはその合計値となっています。

従って、ステータスごとの値が1.0を超えているようであれば、その時間帯は平均して1個以上のリクエストがそのステータスで存在したことを意味しています。

それぞれのステータスの意味については<NQS機能利用の手引き>の「2.1.5　リクエストの状態」を参照してください。

可読形式(パイプキュー)

```
DATE                TOTAL   DEPART    ROUTE   QUEUED     WAIT     HOLD   ARRIVE
================================================================================
2013/09/20 00:00      0.1      0.0      0.1      0.0      0.0      0.0      0.0
2013/09/20 01:00      0.0      0.0      0.0      0.0      0.0      0.0      0.0
2013/09/20 02:00      0.1      0.0      0.1      0.0      0.0      0.0      0.0
2013/09/20 03:00      0.8      0.0      0.8      0.0      0.0      0.0      0.0
2013/09/20 04:00      0.1      0.0      0.1      0.0      0.0      0.0      0.0
2013/09/20 05:00      0.0      0.0      0.0      0.0      0.0      0.0      0.0
...
```

CSV形式(パイプキュー)

```
DATE,TOTAL,DEPART,ROUTE,QUEUED,WAIT,HOLD,ARRIVE
2013/09/20 00:00,0.1,0.0,0.1,0.0,0.0,0.0,0.0
2013/09/20 01:00,0.0,0.0,0.0,0.0,0.0,0.0,0.0
2013/09/20 02:00,0.1,0.0,0.1,0.0,0.0,0.0,0.0
2013/09/20 03:00,0.8,0.0,0.8,0.0,0.0,0.0,0.0
2013/09/20 04:00,0.1,0.0,0.1,0.0,0.0,0.0,0.0
2013/09/20 05:00,0.0,0.0,0.0,0.0,0.0,0.0,0.0
...
```

パイプキューは他のキューへ転送することが主目的であり、転送処理中のステータスはROUTEかDEPARTとなります。従って、この値が大きい場合には転送処理に時間がかかっているリクエストが多いことを意味しますので注意が必要です。

可読形式(バッチキュー)

```
DATE               TOTAL    EXIT     RUN SUSPND  STAGE QUEUED    WAIT    HOLD ARRIVE
================================================================================
2013/09/20 00:00     0.2     0.0     0.0     0.0     0.0     0.1     0.0     0.0     0.1
2013/09/20 01:00    20.4     0.0     8.0     0.0     0.0    12.3     0.0     0.0     0.1
2013/09/20 02:00    20.0     0.0     8.0     0.0     0.0    11.8     0.0     0.0     0.2
2013/09/20 03:00    17.9     0.1     8.0     0.0     0.0     9.8     0.0     0.0     0.0
2013/09/20 04:00     5.1     0.0     4.9     0.0     0.0     0.1     0.0     0.0     0.1
2013/09/20 05:00     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
...
```

表示の都合上、表示幅を調整しているため、実際の出力の幅とは異なります。

CSV形式(バッチキュー)

```
DATE,TOTAL,EXIT,RUN,SUSPEND,STAGE,QUEUED,WAIT,HOLD,ARRIVE
2013/09/20 00:00,0.2,0.0,0.0,0.0,0.0,0.1,0.0,0.0,0.1
2013/09/20 01:00,20.4,0.0,8.0,0.0,0.0,2.3,0.0,0.0,0.1
2013/09/20 02:00,20.0,0.0,8.0,0.0,0.0,1.8,0.0,0.0,0.2
2013/09/20 03:00,17.9,0.1,8.0,0.0,0.0,0.8,0.0,0.0,0.0
2013/09/20 04:00,5.1,0.0,4.9,0.0,0.0,0.1,0.0,0.0,0.1
2013/09/20 05:00,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
...
```

バッチキューの目的はキューイングされているリクエスト(ジョブ)を実行することであり、実行中のリクエストはRUNとなります。RUNとなるためにはキューの同時実行可能数に空きがある必要があり、空きがない場合にはQUEUEDで待たされることになります。従って、QUEUEDの値が大きい場合には対象キューでリクエストが詰まっていることを意味しますので注意が必要です。

例えば上の出力例において対象キューの同時実行可能数が8だとすると、1～3時台はRUNが8.0となっており、常に上限いっぱいのリクエストがRUN状態となっていることがわかります。またその際、QUEUEDで多数のリクエストが待ち合わせさせられていることがわかります。

可読形式(ネットワークキュー)

```
DATE                TOTAL      RUN   QUEUED     WAIT     HOLD   ARRIVE
=====================================================================
2013/09/20 00:00      0.0      0.0      0.0      0.0      0.0      0.0
2013/09/20 01:00      0.3      0.3      0.0      0.0      0.0      0.0
2013/09/20 02:00      0.3      0.3      0.0      0.0      0.0      0.0
2013/09/20 03:00      0.2      0.2      0.0      0.0      0.0      0.0
2013/09/20 04:00      0.1      0.1      0.0      0.0      0.0      0.0
2013/09/20 05:00      0.0      0.0      0.0      0.0      0.0      0.0
...
```

CSV形式(ネットワークキュー)

```
DATE,TOTAL,RUN,QUEUED,WAIT,HOLD,ARRIVE
2013/09/20 00:00,0.0,0.0,0.0,0.0,0.0,0.0
2013/09/20 01:00,0.3,0.3,0.0,0.0,0.0,0.0
2013/09/20 02:00,0.3,0.3,0.0,0.0,0.0,0.0
2013/09/20 03:00,0.2,0.2,0.0,0.0,0.0,0.0
2013/09/20 04:00,0.1,0.1,0.0,0.0,0.0,0.0
2013/09/20 05:00,0.0,0.0,0.0,0.0,0.0,0.0
...
```

ネットワークキューの目的は実行完了したリクエスト(ジョブ)の結果を転送元へ返却することであり、結果返却処理中の場合にはRUNとなります。同時実行可能数を超えている場合にはQUEUEDで待ち合わせが発生します。

ネットワークキューはUNIX版だけのキュー種別であり、Windows版には存在しません。

■qreq、レベル2

レベル0とレベル1の情報に加え、各時間帯の最小値(MIN)と最大値(MAX)が出力されます。より詳細な情報を確認したい場合に指定してください。

以下に出力フォーマットのみ記載します。

CSV形式(パイプキュー)

```
DATE,IN,OUT,TOTAL(AVE),TOTAL(MIN),TOTAL(MAX),DEPART(AVE),DEPART(MIN),
DEPART(MAX),ROUTE(AVE),ROUTE(MIN),ROUTE(MAX),QUEUED(AVE),QUEUED(MIN),
QUEUED(MAX),WAIT(AVE),WAIT(MIN),WAIT(MAX),HOLD(AVE),HOLD(MIN),HOLD(MAX),
ARRIVE(AVE),ARRIVE(MIN),ARRIVE(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
...
```

CSV形式(バッチキュー)

```
DATE,IN,OUT,TOTAL(AVE),TOTAL(MIN),TOTAL(MAX),EXIT(AVE),EXIT(MIN),EXIT(MAX),
RUN(AVE),RUN(MIN),RUN(MAX),SUSPEND(AVE),SUSPEND(MIN),SUSPEND(MAX),STAGE(AVE),
STAGE(MIN),STAGE(MAX),QUEUED(AVE),QUEUED(MIN),QUEUED(MAX),WAIT(AVE),WAIT(MIN),
WAIT(MAX),HOLD(AVE),HOLD(MIN),HOLD(MAX),ARRIVE(AVE),ARRIVE(MIN),ARRIVE(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
...
```

CSV形式(ネットワークキュー)

```
DATE,IN,OUT,TOTAL(AVE),TOTAL(MIN),TOTAL(MAX),RUN(AVE),RUN(MIN),RUN(MAX),QUEUED(AVE),
QUEUED(MIN),QUEUED(MAX),WAIT(AVE),WAIT(MIN),WAIT(MAX),HOLD(AVE),HOLD(MIN),HOLD(MAX),
ARRIVE(AVE),ARRIVE(MIN),ARRIVE(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
...
```

表示の都合上、タイトル行に改行が入っていますが、実際には1行で出力されます。

■rtrk、レベル0

可読形式

```
DATE                    IN      OUT
==================================
2013/09/20 00:00        10        8
2013/09/20 01:00        12       11
2013/09/20 02:00        14       15
2013/09/20 03:00        14       14
2013/09/20 04:00        10       10
2013/09/20 05:00        10       11
...
```

CSV形式

```
DATE,IN,OUT
2013/09/20 00:00,10,8
2013/09/20 01:00,12,11
2013/09/20 02:00,14,15
2013/09/20 03:00,14,14
2013/09/20 04:00,10,10
2013/09/20 05:00,10,11
...
```

IN/OUTの見方は以下の通りです。

IN	その時間帯にメモリに読み込まれたトラッカ数です。以下のいずれかの場合にメモリに読み込まれます。 ■ 予定状態のトラッカが予定開始時刻を迎えた場合 ■ 予定状態のトラッカ対して何らかの操作を行い、予定[確定]状態となった場合 ■ ジョブネットワークを即時投入した場合
OUT	その時間帯にメモリから解放されたトラッカ数です。以下のいずれかの場合にメモリから解放されます。 ■ 未アーカイブのトラッカがアーカイブされた場合 ■ 未アーカイブのトラッカを削除した場合

■rtrk、レベル1

可読形式

```
DATE                TOTAL  PREPARE  RUNCTRL      RUN RERUNWAI
============================================================
2013/09/20 00:00      0.0      0.0      0.0      0.0      0.0
2013/09/20 01:00      6.5      0.2      0.0      2.3      4.0
2013/09/20 02:00     19.8      1.1      4.8      9.1      4.8
2013/09/20 03:00     17.0      0.0      0.0      5.8     11.2
2013/09/20 04:00      9.0      0.0      0.0      3.9      5.1
2013/09/20 05:00      0.0      0.0      0.0      0.0      0.0
...
```

CSV形式

```
DATE,TOTAL,PREPARE,RUNCTRL,RUN,RERUNWAI
2013/09/20 00:00,0.0,0.0,0.0,0.0,0.0
2013/09/20 01:00,6.5,0.2,0.0,2.3,4.0
2013/09/20 02:00,19.8,1.1,4.8,9.1,4.8
2013/09/20 03:00,17.0,0.0,0.0,5.8,11.2
2013/09/20 04:00,9.0,0.0,0.0,3.9,5.1
2013/09/20 05:00,0.0,0.0,0.0,0.0,0.0
...
```

各ステータスの見方は以下の通りです。

PREPARE	予定[確定]状態となっているトラッカ数の時間平均値です。
RUNCTRL	ジョブネットワークパラメータの同時起動可能数の制限により、予定[確定]状態として待ち合わせているトラッカ数の時間平均値です。
RUN	実行中となっているトラッカ数の時間平均値です。
RERUNWAI	実行は完了していますがまだアーカイブされていないトラッカ数の時間平均値です。

■rtrk、レベル2

レベル0とレベル1の情報に加え、各時間帯の最小値(MIN)と最大値(MAX)が出力されます。より詳細な情報を確認したい場合に指定してください。

以下に出力フォーマットのみ記載します。

CSV形式

```
DATE,IN,OUT,TOTAL(AVE),TOTAL(MIN),TOTAL(MAX),PREPARE(AVE),PREPARE(MIN),
PREPARE(MAX),RUNCTRL(AVE),RUNCTRL(MIN),RUNCTRL(MAX),RUN(AVE),RUN(MIN),
RUN(MAX),RERUNWAI(AVE),RERUNWAI(MIN),RERUNWAI(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0.0,0,0,0.0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0.0,0,0,0.0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0.0,0,0,0.0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0.0,0,0,0.0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0.0,0,0,0.0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0.0,0,0,0.0,0,0
...
```

表示の都合上、タイトル行に改行が入っていますが、実際には1行で出力されます。

■rjob、レベル0

可読形式

```
                     UNIT     UNIT
DATE                   IN      OUT
==================================
2013/09/20 00:00      121      121
2013/09/20 01:00      374      374
2013/09/20 02:00      811      811
2013/09/20 03:00      374      374
2013/09/20 04:00      374      374
2013/09/20 05:00      370      370
...
```

CSV形式

```
DATE,UJ_IN,UJ_OUT
2013/09/20 00:00,121,121
2013/09/20 01:00,374,374
2013/09/20 02:00,811,811
2013/09/20 03:00,374,374
2013/09/20 04:00,374,374
2013/09/20 05:00,370,370
...
```

IN/OUTの見方は以下の通りです。

IN	その時間帯に投入可能状態となった(フローが到達した)ジョブ部品数です。
OUT	その時間帯に投入処理が完了したジョブ部品数です。 OUTが記録されても実行が開始されているわけではありませんので注意してください。実際にジョブが実行されるのは、投入先のキューの空き状況に依存します。 キューの状態はqreqデータで確認することができます。

上記出力例は-aオプションを指定しない単位ジョブだけの出力例であり、-aオプションを指定した場合にはさらに指定したジョブ部品のフィールドが出力されます。

■rjob、レベル1

可読形式

```
                     UNIT     UNIT     UNIT
DATE                TOTAL     WAIT      RUN
==========================================
2013/09/20 00:00      0.0      0.1      0.0
2013/09/20 01:00      4.5      4.5      0.0
2013/09/20 02:00     12.0     12.0      0.0
2013/09/20 03:00     10.0     10.0      0.0
2013/09/20 04:00     46.2     46.1      0.1
2013/09/20 05:00      0.0      0.0      0.0
...
```

CSV形式

```
DATE,UJ_TOTAL,UJ_WAIT,UJ_RUN
2013/09/20 00:00,0.0,0.1,0.0
2013/09/20 01:00,4.5,4.5,0.0
2013/09/20 02:00,12.0,12.0,0.0
2013/09/20 03:00,10.0,10.0,0.0
2013/09/20 04:00,46.2,46.1,0.1
2013/09/20 05:00,0.0,0.0,0.0
...
```

WAITとRUNは投入可能状態となったジョブ部品の、JobCenter内部での処理状態を表しています。TOTALはその合計値です。

WAIT	投入可能状態となったジョブ部品のうち、まだ投入処理が始まっていないジョブ部品数の時間平均値です。
RUN	投入可能状態となったジョブ部品のうち、投入処理が始まったジョブ部品数の時間平均値です。

つまり、フローがジョブ部品に到達するとこのrjobデータのINとして計上され、その際の最初の状態がWAITとなります。その後、実際に投入処理が始まるとRUNに変わり、投入処理が完了するとrjobデータのOUTとして計上されます。

このレベル1のデータにおいて、RUNとなっている時間、すなわち投入処理にかかっている時間はほぼ一瞬であり、実際にはWAIT状態として計上されることがほとんどです。従って、TOTAL値にのみ注目してください。

ここでのTOTAL値が大きい場合、投入可能状態となっているジョブが未処理のまま溜まっていることを意味するため、その場合、前段の部品が実行完了しているにも関わらず後続のジョブ部品がなかなか始まらない（SUBMIT状態に遷移しない）という遅延が発生します。このような状況が頻発し、かつ、発生している遅延時間が運用として許容できない場合には、ジョブの実行が集中しないようスケジューリング設定の見直しが必要になります。

上記出力例は-aオプションを指定しない単位ジョブだけの出力例であり、-aオプションを指定した場合にはさらに指定したジョブ部品のフィールドが出力されます。

■rjob、レベル2

レベル0とレベル1の情報に加え、各時間帯の最小値(MIN)と最大値(MAX)が出力されます。より詳細な情報を確認したい場合に指定してください。

以下に出力フォーマットのみ記載します。

CSV形式

```
DATE,UJ_IN,UJ_OUT,UJ_TOTAL(AVE),UJ_TOTAL(MIN),UJ_TOTAL(MAX),UJ_WAIT(AVE),
UJ_WAIT(MIN),UJ_WAIT(MAX),UJ_RUN(AVE),UJ_RUN(MIN),UJ_RUN(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
...
```

表示の都合上、タイトル行に改行が入っていますが、実際には1行で出力されます。

上記出力例は-aオプションを指定しない単位ジョブだけの出力例であり、-aオプションを指定した場合にはさらに指定したジョブ部品のフィールドが出力されます。

3.25.3. 戻り値

戻り値	内 容
0	正常終了です。
1	異常終了です。

3.25.4. 主要メッセージ

メッセージ	内容
Only manager user can execute this command.	JobCenter管理者ユーザ以外で実行しています。JobCenter管理者ユーザで実行してください。
Too many arguments.	引数の数が多いです。引数の指定数が正しいか確認してください。
Too few arguments.	引数の数が少ないです。引数の指定数が正しいか確認してください。
Data file or data not found.	データファイルが存在しないか、データファイル中に解析対象となるデータが存在しません。
Invalid command mode	コマンドモードが不正です。config,stat,viewのいずれかを指定する必要があります。
Invalid data type	指定したデータ種別が不正です。qreq,rtrk,rjobのいずれかを指定する必要があります。

3.25.5. 注意事項

■本コマンドはJobCenter管理者ユーザで実行する必要があります。

■本コマンドによりデータ収集機能をONにした場合、実行状況を記録したデータが蓄積されるため、そのデータの保存期間とディスクの空き容量に注意してください。

3.26. trfdelete トラッキングファイルの削除

■UNIX版

```
/usr/lib/nqs/trfdelete [-a | -o $orig_mid -s $del_seqno]
```

■Windows版

```
%InstallDirectory%\bin\trfdelete [-a | -o $orig_mid -s $del_seqno]
```

3.26.1. 機能説明

JobCenterのバージョンアップ時や障害発生時の対応でトラッキングファイルを削除する場合に使用するコマンドです。　本コマンドは、UNIX版ではrootユーザ、Windows版ではJobCenter管理者で実行してください。JobCenterが停止している状態で利用してください。

3.26.2. オプション

-a

JobCenterのバージョンアップ時や障害発生時の対応でトラッキングデータを全削除する場合に使用します。JobCenterが停止している状態で実行する必要があります。

オプションを省略した場合の既定値の動作になります。

-o $orig_mid -s $del_seqno

指定したmid（MGのマシンID）、seqno（シーケンス番号）のトラッキングデータ１件を削除します。JobCenterが起動している状態で実行する必要があります。

3.26.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

3.26.4. 注意事項

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITE、NQS_SITEDBを設定する必要があります。

■正常に動作しているリクエストのトラッキングデータを削除した場合、リクエストの実行は正常に継続することができなくなります。JobCenterのバージョンアップ時や、障害発生時の対応でトラッキングファイルの削除を指示された場合以外では使用しないでください。

■UNIX版のクラスタサイトの運用系のバージョンアップ時のNQS関連データをリストア後のクラスタサイトに対するtrfdeleteコマンドの実行については、下記のディレクトリ配下にO0000～O0029,S0000～S0029のディレクトリが存在しない時には実行する必要はありません。

UNIX版	<クラスタDBパス>/lock/tracking

3.27. JCSession 単位ジョブから指定したコマンドをActive状態のセッションで実行

■Windows版

```
%InstallDirectory%\bin\utils\JCSession.exe [-l] [-i $sessionid] $command [$arguments...]
```

3.27.1. 機能説明

単位ジョブから指定したコマンドをActive状態のセッションで実行します。

このコマンドを使用することにより、Windows Server 2008以降の環境でNotePadなどのGUIアプリケーションやExcelのマクロを設定したジョブを正常に実行することができます。

本コマンドを使用するには、単位ジョブの投入キューにLSAキューを指定する必要があります。LSAキューの詳細は <NQS機能利用の手引き>の「6.1.3.8 LSAキュー属性」 を参照してください。

3.27.2. オプション

-l

指定したコマンド内部でジョブオブジェクトを利用している場合、本オプションを指定してください。本オプションを指定しない場合、そのコマンド内部におけるジョブオブジェクトの処理が失敗します。

詳しくは「3.27.6 注意事項」を参照してください。

-i $sessionid

指定したコマンドを実行するセッションIDです。

指定したセッションIDがActive状態でない場合はエラーとなります。

本オプションを省略した場合は、環境変数JC_JOBRUNSESSIONの値を使用します。

本オプションも環境変数JC_JOBRUNSESSIONも設定されていない場合、Active状態のセッションが自動的に選択されるため、明示的に特定のセッションを指定してジョブを実行したい場合には、必ず本オプションを指定するか、環境変数JC_JOBRUNSESSSIONを指定してください。

$command $arguments...

実行するコマンドおよび引数です。

$commandの値は259バイト以内で指定してください。

3.27.3. 環境変数

本コマンドは以下の環境変数を使用します。

環境変数名	説明
JC_JOBRUNSESSION	指定したコマンドを実行するセッションIDです。-iオプションを省略した場合、この値を使用します。

	指定したセッションIDがActive状態でない場合はエラーとなります。
JC_USE_USERENV	指定したコマンドの環境変数の継承方法を設定します。以下の値が利用可能です。 ■JobCenter JobCenterの環境変数を使用します。 ■User 指定したコマンドを実行するセッションユーザの環境変数を使用します。 ■Both JobCenterの環境変数および指定したコマンドを実行するセッションユーザの環境変数の両方を使用します。 本環境変数を設定していない場合のデフォルト値です。 それぞれ同じ環境変数が定義されている場合は、セッションユーザの環境変数が優先されます。

3.27.4. 戻り値

JCSessionコマンドの戻り値は以下のとおりです。

戻り値	内　容
0	正常終了です。
1	異常終了です。

JCSessionコマンドが正常に実行される場合、$commandで指定したコマンドが実行されるので、コマンド終了時に得られる戻り値は$commandの戻り値になります。

3.27.5. 使用方法

Administratorユーザのセッションで NotePadを起動する場合を例にして、JCSessionコマンドの使用方法を説明します。

1. Administratorユーザのセッション IDの確認

   あらかじめ、NotePadを画面上に表示したいセッションのIDを確認します。

   対象となるジョブ実行サーバにログオンし、Ctrl + Shift + Esc キーを押してタスクマネージャを起動し、[ユーザー]タブを表示します。ID列の値がセッションIDです。ここではAdministratorユーザのセッションIDが"1"であることが確認できます。

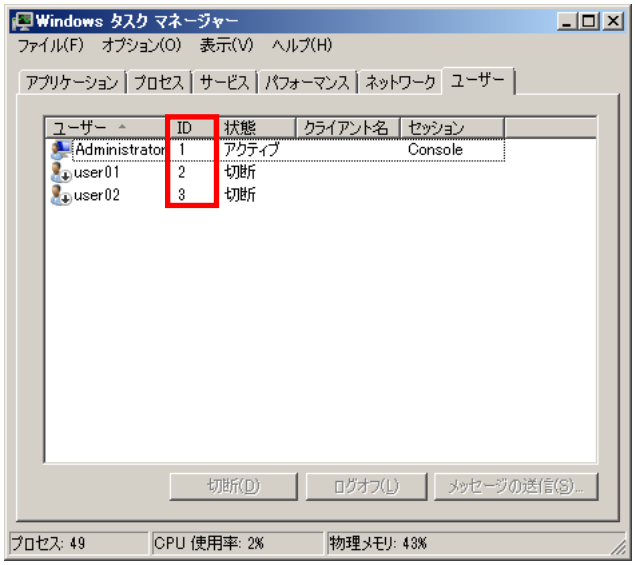

図3.1 セッションIDの確認

■セッションIDはWindowsのログオン/ログアウトで値が変更されるため、対象のユーザを常にログオン状態にしておく必要があります。

■ID列が表示されていない場合は、以下の手順を行ってください。

- Windows Server 2008の場合

  a. タスクマネージャの[ユーザー]タブを表示します。

  b. メニューバーの[表示]→[列の選択]をクリックします。

  c. IDのチェックボックスにチェックを入れます。

- Windows Server 2012の場合

  a. タスクマネージャの[ユーザー]タブを表示します。

  b. 列のタイトルを右クリックし、IDを選択します。

2. ジョブネットワークの作成

   ジョブネットワークを作成します。作成方法については<基本操作ガイド>の「3.1 ジョブネットワークを作成する」を参照して下さい。

3. 単位ジョブの配置

   手順2で作成したジョブネットワークのフローに単位ジョブを配置します。配置方法については<基本操作ガイド>の「4.1 部品オブジェクトを配置する」を参照して下さい。

4. 単位ジョブのスクリプトの設定

   単位ジョブのスクリプトを開き、以下のコマンドを記入します。ここで、-iオプションに手順1で確認したセッションIDである"1"を指定します。

```
%InstallDirectory%\bin\utils\JCSession.exe -i 1 notepad.exe
```

5. 単位ジョブの投入キューにLSAキューを設定

単位ジョブパラメータの設定画面を開き、[実行設定]タブの[投入キュー]にある[参照]ボタンをクリックします。

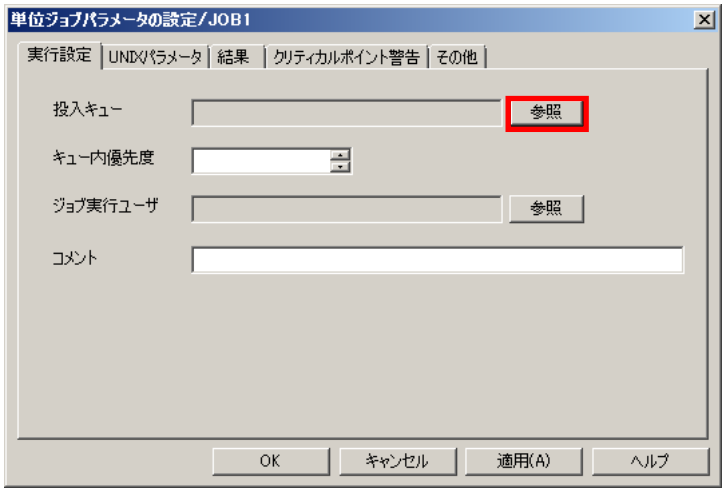

図3.2 単位ジョブパラメータの設定画面

投入先キューの設定画面が表示されます。LSAキューを選択し、[OK]ボタンをクリックします。

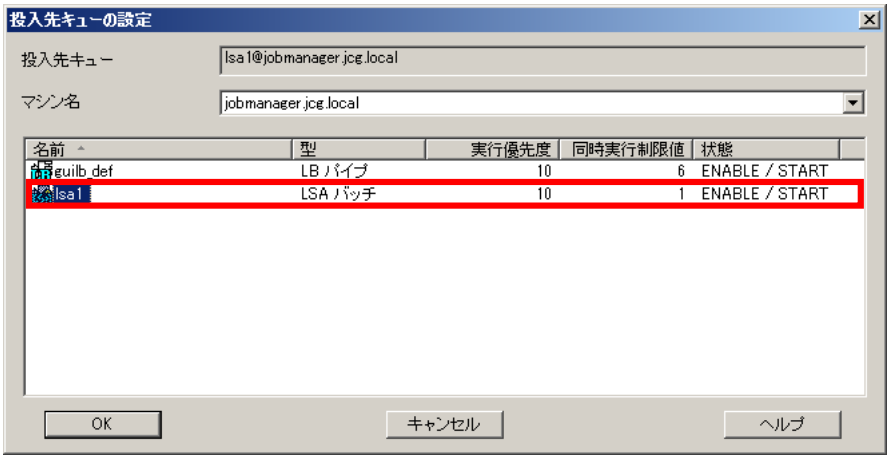

図3.3 投入先キューの設定画面

単位ジョブパラメータの設定画面で[OK]ボタンをクリックし、設定を終了してください。

6. ジョブネットワークの実行

作成したジョブネットワークを実行します。実行方法については<基本操作ガイド>の「3.2　ジョブネットワークの実行」を参照して下さい。

AdministratorユーザのセッションにNotePadが起動されます。NotePadが起動している間は単位ジョブが実行中の状態になります。NotePadを終了すると単位ジョブも終了します。

3.27.6. 注意事項

■本コマンドはコマンドプロンプトから直接実行することはできません。

■JobCenterが「起動(cjcpw)」で起動している場合、本コマンドを使用することはできません。「起動(サービス)」で起動して下さい。

起動方法については<環境構築ガイド>の「13.3.2 サイトの起動と停止」を参照して下さい。

■ユーザアカウント制御が有効である場合にユーザアカウント制御が表示されるような$commandをJCSessionコマンドを実行したいときは、以下のように「cmd.exe /c」経由で実行する必要があります。

```
%InstallDirectory%\bin\utils\JCSession.exe cmd.exe /c $command
```

なお「cmd.exe　/c」を経由しないで実行した場合、エラーとなって標準エラー出力に以下のようなメッセージが出力されます。

```
CreateProcessAsUser ERROR, ec=740.
```

■JCSessionコマンドでは、Windowsにおけるジョブオブジェクト(プロセスをグループ化したもの)を利用しています。JCSession経由で起動するアプリケーションにはジョブオブジェクトが割り当てられ、CL/Winから強制停止(またはスキップ)した場合に、JCSession経由で起動された一連のプロセスを強制的に終了するために利用しています。

ただし、1つのプロセスに異なるジョブオブジェクトを割り当てることはできないため、JCSessionコマンド経由で実行するアプリケーション内部で独自にジョブオブジェクトを割り当てる処理が存在する場合、その処理が失敗することになります。これを回避するには、-Iオプションを指定します。この場合、JCSessionコマンドがプロセスを起動する際に、ジョブオブジェクトを割り当てないようになります。

ただし、-Iオプションを指定したJCSessionの単位ジョブを強制停止(またはスキップ)した場合、単位ジョブ自体は強制停止(またはスキップ)されますが、JCSessionコマンド経由で起動されたプロセスは実行されたままとなりますので、注意してください。

■1つの単位ジョブスクリプト内部で実行されるJCSessionについて、JCSessionから起動されたアプリケーションがジョブの強制停止(またはスキップ)によって一緒に停止されるのは累積1000回までのJCSessionの呼び出しまでとなります。1001回目のJCSessionについては、ジョブの強制停止(またはスキップ)によって該当プロセス群の停止は行われません。

3.28. setuserpwd ユーザパスワードの更新

■Windows版

```
%InstallDirectory%\bin\setuserpwd -c

%InstallDirectory%\bin\setuserpwd $user $password
```

3.28.1. 機能説明

Windows版JobCenter MG/SVにおいて、指定したユーザのパスワードを更新します。

Windows版JobCenter　MG/SVでは、指定されたユーザの権限でジョブを実行する際、バックグラウンドでOSにログオン認証する必要があるため、そのOSパスワードを事前にJobCenter　MG/SVに登録する必要があります。これはJobCenterへのユーザ登録時に行われますが、OSのパスワードを変更した場合にはその変更内容で再度登録する必要があります。

パスワードの登録は、本コマンドで更新する他、サーバの環境設定からも更新可能です。サーバの環境設定から行う方法については、<環境構築ガイド>の「13.4 ユーザの設定」を参照してください。

3.28.2. オプション

-c

パスワードの登録状況を標準出力に表示します。

$user

更新対象のユーザ名を指定します。

$password

更新対象となるユーザのパスワードを指定します。

パスワードに以下の文字が含まれる場合、それぞれ以下の文字でエスケープが必要です。

エスケープが必要な文字	エスケープ文字	コマンド実行例
^	^	setuserpwd.exe user1 te^^st^<123^>45
&		
|		
<		
>		
"	\	setuserpwd.exe user1 test\"123\"45

3.28.3. 出力結果

-c指定時、標準出力に出力される内容は以下の通りです。

```
 UserName                 HostName                 PasswordStatus
-----------------------------------------------------------------------
 user1                    testdomain.local         OK
 user2                    testdomain.local         NG
 testuser                 jobmanager               OK
 Administrator            testdomain.local         OK
```

項目	内容
UserName	ユーザ名
HostName	対象ユーザの所属するドメイン名が表示されます。 ローカルユーザの場合は自ホスト名が表示されます。
PasswordStatus	対象ユーザのパスワード登録状況が表示されます。 OK: 正しいパスワードが登録されています NG: パスワードが間違っています NOTSET: 未登録

その他の主要メッセージは以下の通りです。

メッセージ	内容
Updating JobCenter administrator password... OK	JobCenter管理者のパスワードの更新に成功しました。
Set user password OK!	一般ユーザのパスワードの更新に成功しました。
The password is incorrect.(ERRORCODE=xxx)	入力されたパスワードによる認証に失敗しました。(エラーコード=xxx)
The account does not belong to JobCenter group.	対象ユーザはJobCenterグループに所属していません。
Error! The parameter is incorrect.	引数が間違っています。

3.28.4. 戻り値

■-c指定時

戻り値	内　容
0	全ユーザのPasswordStatusがOK
1	PasswordStatusがNGとなっているユーザが1人以上存在
2	コマンド異常終了

■パスワード更新時

戻り値	内　容
0	正常終了
2	異常終了

3.28.5. 注意事項

■JobCenter管理者（アカウント）のパスワードを変更した場合は、JobCenter　MG/SVの再起動を行う必要があります。

3.29. paramctrl 暗号化DBへのパラメータ登録

■UNIX版

```
/usr/lib/nqs/gui/tools/paramctrl list

/usr/lib/nqs/gui/tools/paramctrl [ -f | -c ] set $key $value

/usr/lib/nqs/gui/tools/paramctrl [-f] del $key
```

■Windows版

```
%InstallDirectory%\tools\paramctrl.bat list

%InstallDirectory%\tools\paramctrl.bat [ -f | -c ] set $key $value

%InstallDirectory%\tools\paramctrl.bat [-f] del $key
```

3.29.1. 機能説明

■本コマンドは、拡張カスタムジョブで使用するパラメータをJobCenter内のDBに暗号化して格納する機能を提供します。

■設定した暗号化パラメータは、拡張カスタムジョブのカスタムパラメータでkeyの形式で指定することができます。パスワード等の秘匿パラメータを利用する場合に、本コマンドを使用することで暗号化して保存することができます。

■カスタムパラメータにおけるパラメータの展開は実行時に行われるため、スクリプトプレビュー機能ではkeyのまま表示されます。

3.29.2. サブコマンド

サブコマンド	機能
list	設定されているキーの一覧を表示します。
set	暗号化DB上にキー($key)と値($value)の組を登録します。 $keyは半角英字の大文字小文字を区別します。
del	暗号化DB上に登録されているキー($key)を削除します。 $keyは半角英字の大文字小文字を区別します。

3.29.3. オプション

-f

本オプションを指定した場合、既存のキーに対して確認を行わずに上書き(set)または削除(del)を行います。

本オプションおよび後述の-cオプションのどちらも指定しない場合、以下のような確認メッセージが表示されます。

■set指定時

```
Entry of "$key" already exists. Do you want to overwrite it? (y/n)
```

■del指定時

```
Do you want to delete key ? [ $key ] (y/n)
```

続行する場合はyを選択します。nを選択した場合、処理はキャンセルされます。

-c

setサブコマンドのオプションです。

本オプションを指定した場合、$keyと同名のキーが既に存在した時に確認を行わず登録処理をキャンセルします。

登録処理がキャンセルされた場合、以下のようなメッセージが表示されコマンドは正常終了します。

```
key already exists : [ $key ].
```

-cオプションと-fオプションを同時に指定することはできません。

同時に指定された場合、以下のメッセージが表示されコマンドは異常終了します。

```
<-f> and <-c> cannot exist at the same time.
```

3.29.4. 戻り値

戻り値	内容
0	正常終了です。
1	異常終了です。

3.29.5. 注意事項

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITE、NQS_SITEDBを設定する必要があります。

■暗号化パラメータは、バックアップ・復元機能のバックアップ対象外データです。暗号化パラメータを復元する場合は、本コマンドにより同じ値を再設定する必要があります。

3.30. jcres JobCenter MG/SV専用のHTTPデーモン

■UNIX版

```
/usr/lib/nqs/gui/tools/jcres

/usr/lib/nqs/gui/tools/jcres -e

/usr/lib/nqs/gui/tools/jcres -v
```

■Windows版

```
%InstallDirectory%\tools\jcres.bat

%InstallDirectory%\tools\jcres.bat -e

%InstallDirectory%\tools\jcres.bat -v
```

3.30.1. 機能説明

本コマンドは、JobCenter　MG/SV専用のHTTPデーモンjcresの起動及び停止を行います。jcresのAPIを用いて、ブラウザからMG-SV間の疎通確認を行うことができます。

jcresの使用方法については、<環境構築ガイド>の18章 「jcresの使用方法」 を参照してください。

3.30.2. オプション

-e jcresを停止します。
-v バージョン情報を表示します。

3.30.3. 設定ファイル

jcresの動作を設定することができます。デフォルトの動作を変更したい場合に設定してください。

設定ファイルは以下のディレクトリに格納されています。

UNIX版	/usr/lib/nqs/rc/jcres.conf
Windows版	%InstallDirectory%\etc\jcres.conf

テキストエディタで以下のフォーマットに従って設定してください。

```
パラメータ=値
```

設定可能なパラメータは以下のとおりです。

パラメータ	説明	範囲	初期値
LOGDIR	アクセスログ及びエラーログを保存するディレクトリを絶対パスで指定します。	絶対パス	■UNIX版 /usr/spool/nqs/log ■Windows版 %InstallDirectory%\spool\log\error
PORT	ポート番号を指定します。	1 ～ 65535	50080

パラメータ	説明	範囲	初期値
IPADDRESS	IPアドレスを指定します。	IPv4アドレス	0.0.0.0

設定ファイルのサンプルは以下のとおりです。

```
LOGDIR=/usr/spool/nqs/log
PORT=50080
IPADDRESS=0.0.0.0
```

■設定ファイルは、jcresを停止してから編集してください。jcresが起動したままの状態で設定ファイルを編集した場合、jcresを正常に停止できなくなる可能性があります。

■設定ファイルの読み込みに失敗した場合は、各パラメータは初期値で動作します。

■パラメータが未設定の場合や不正なパラメータを指定した場合、そのパラメータは初期値で動作します。

■LOGDIRを設定する場合は、事前に対象のディレクトリを作成しておいてください。存在しないディレクトリやファイルを作成できないディレクトリを指定した場合、初期値で動作します。

3.30.4. アクセスログ

受信したHTTPリクエストのアクセスログが記録されます。

アクセスログは以下のディレクトリに格納されます。

UNIX版	/usr/spool/nqs/log/jcres_access.log
Windows版	%InstallDirectory%\spool\log\error\jcres_access.log

格納先のディレクトリは設定ファイルで変更することができます。設定ファイルの詳細は「3.30.3 設定ファイル」を参照してください。

アクセスログのフォーマットは以下のとおりです。

```
送信元IPアドレス [YYYY/MM/DD hh:mm:ss] HTTPメソッド URL レスポンスステータス
```

■送信元IPアドレス

HTTPリクエストの送信元IPアドレスです。

■[YYYY/MM/DD hh:mm:ss]

HTTPリクエストの受信日時です。

■HTTPメソッド

HTTPメソッド名です。

■URL

リクエストのURLです。

■レスポンスステータス

クライアントに返却したレスポンスコードです。

3.30.5. エラーログ

障害が発生した場合の原因究明に必要な情報が記録されます。

エラーログは以下のディレクトリに格納されます。

UNIX版	/usr/spool/nqs/log/jcres_error.log
Windows版	%InstallDirectory%\spool\log\error\jcres_error.log

格納先のディレクトリは設定ファイルで変更することができます。設定ファイルの詳細は「3.30.3 設定ファイル」を参照してください。

なお、エラーログのフォーマットは公開しておりませんので、あらかじめご了承ください。

3.30.6. API

jcresで使用可能なAPIです。以下のフォーマットでHTTPリクエストを発行してください。

```
http://<jcresのIPアドレス>:<jcresのポート番号>/<APIのURL>
```

> 結果はapplication/octet-stream形式で返却されますので、ブラウザによってはファイルをダウンロードしてテキストエディタで表示する必要があります。

3.30.6.1. ping

■機能説明

jcresの疎通確認を行います。

jcresが起動している場合、文字列"pong"を返却します。

■URL

```
/ping
```

■パラメータ

なし

■HTTPメソッド

GET

3.30.6.2. jc_ping

■機能説明

jc_pingコマンドで送信元サイトから送信先サイトへの疎通確認を行います。

送信元サイトはjcresを起動しているマシンのローカルサイトまたはクラスタサイトを指定することができます。別マシンのサイトを指定することはできません。別マシンのサイトを送信元サイトに指定したい場合は、送信先サイトが存在するマシンのjcresに対して本APIのHTTPリクエストを発行してください。

jc_pingコマンドの実行に成功した場合、jc_pingコマンドの可読形式の結果が返却されます。

jc_pingコマンドについては「6.3 jc_ping 指定したサイトに対する通信確認」を参照してください。

■URL

- 送信元サイトがローカルサイトの場合

```
/jc_ping?dhost=<送信先サイト>&af=<アドレスファミリ>&ctw=<タイムアウト時間>
```

- 送信元サイトがクラスタサイトの場合

```
/sites/<送信元サイト>/jc_ping?dhost=<送信先サイト>&af=<アドレスファミリ>&ctw=<タイムアウト時間>
```

■パラメータ

- 送信元サイト

  送信元サイト名です。

  送信元サイトがクラスタサイトの場合に指定します。JobCenterで登録している正式なサイト名を指定してください。

- dhost=<送信先サイト>

  送信先サイト名です。jc_pingコマンドの$destination_hostに相当します。

  JobCenterで登録している正式なサイト名を指定してください。

- af=<アドレスファミリ>

  疎通確認を行うアドレスファミリです。jc_pingコマンドの-pオプションに相当します。

  IPv4アドレスの場合は"ipv4"、IPv6アドレスの場合は"ipv6"を指定してください。省略した場合はIPv4アドレスで疎通確認を行います。

- ctw=<タイムアウト時間>

  各デーモンごとのconnect確認のタイムアウト時間(秒)です。jc_pingコマンドの-wオプションに相当します。

  0～600の整数を指定してください。省略した場合は30秒となります。

■HTTPメソッド

GET

3.30.7. 戻り値

戻り値	内容
0	正常終了です。
1	異常終了です。

3.30.8. 主要メッセージ

メッセージ	内容
error: Only root user can execute this command.	UNIX版ではrootユーザで実行してください。
error: Only JobCenter admin can execute this command.	Windows版ではJobCenter管理者で実行してください。
failed to bind. (エラーの詳細情報)	バインドに失敗しました。
failed to stop server. (エラーの詳細情報)	jcresの停止に失敗しました。

3.30.9. 注意事項

■本コマンドは、UNIX版ではrootユーザ、Windows版ではJobCenter管理者で実行してください。

■IPv6アドレスはサポートしていません。

■jcresを複数起動しないでください。

3.31. jnwqctrl 起動トリガ機能が管理するJNW投入予約リストの参照、および削除

■UNIX版

- JNW投入予約リストの参照

```
/usr/lib/nqs/gui/tools/jnwqctrl show -a

/usr/lib/nqs/gui/tools/jnwqctrl show -u $user  [-t $trg ] [-l $line  [-r]]

/usr/lib/nqs/gui/tools/jnwqctrl show -h
```

- JNW投入予約リストの削除

```
/usr/lib/nqs/gui/tools/jnwqctrl delete -a

/usr/lib/nqs/gui/tools/jnwqctrl delete -u $user  [-t $trg ]

/usr/lib/nqs/gui/tools/jnwqctrl delete -h
```

■Windows版

- JNW投入予約リストの参照

```
%InstallDirectory%\tools\jnwqctrl show -a

%InstallDirectory%\tools\jnwqctrl show -u $user  [-t $trg ] [-l $line  [-r]]

%InstallDirectory%\tools\jnwqctrl show -h
```

- JNW投入予約リストの削除

```
%InstallDirectory%\tools\jnwqctrl delete -a

%InstallDirectory%\tools\jnwqctrl delete -u $user  [-t $trg ]

%InstallDirectory%\tools\jnwqctrl delete -h
```

3.31.1. 機能説明

本コマンドは、起動トリガ機能が管理するJNW投入予約リストの参照、および削除する機能を提供します。

JNW投入予約リストは、起動トリガ機能から投入するJNWの投入データを管理しています。

起動トリガ機能から大量のJNW投入が行われる場合(例えば、メッセージラッシュなど)、本コマンドで状況の確認、およびJNW投入のキャンセルを行うことができます。

3.31.2. サブコマンド

サブコマンド	機能
show	起動トリガ機能が管理する投入予約リストを参照します。
delete	起動トリガ機能が管理する投入予約リスト内のJNW投入データの一部、または全てを削除します。

3.31.3. オプション

-a, --all

JNW投入予約リスト内の全てのJNW投入データを操作対象とします。

-u, --user

JNW投入予約リスト内のJNW投入データにおいて、$userで指定したユーザ名と一致するJNW投入データを操作対象とします。

本オプションは-aオプションと同時に指定できません。

-t, --target

JNW投入予約リスト内のJNW投入データにおいて、$trgで指定した起動トリガ名と一致するJNW投入データを操作対象とします。

本オプションは-aオプションと同時に指定できません。

-l, --line

JNW投入予約リスト内の投入予約の追加が古い順から$line件のJNW投入データを操作対象とします。

-uオプション、-tオプションを指定している場合、-uオプションと-tオプションによる操作対象の絞込み後に適用されます。

$lineは、最大1000件まで指定可能です。

-r, --reverse

-lオプションを指定した場合、操作対象とする順番を投入予約の追加が新しい順(直近で投入予約された順)に変更します。

-h, --help

コマンドのヘルプメッセージが表示されます。

3.31.4. 戻り値

戻り値	内容
0	正常終了です。
1	異常終了です。

3.31.5. 実行結果例

■JNW投入予約リストの参照

- テキストログ監視の出力形式
  - 出力形式は、以下の通りです。

```
ユーザ名,起動トリガ名,JNW名,環境変数1,環境変数2,...
```

環境変数1,環境変数2,...に対応する環境変数は、以下の環境変数の順番で出力されます。

以下の環境変数については、<基本操作ガイド>の「7.2.1.4　　起動条件を設定する」を参照してください。

- NSJNW_TRG_LOG_MATCH_LINE

- NSJNW_TRG_LOG_MATCH_0
- NSJNW_TRG_LOG_MATCH_1
- NSJNW_TRG_LOG_MATCH_2
- NSJNW_TRG_LOG_MATCH_3
- NSJNW_TRG_LOG_MATCH_4
- NSJNW_TRG_LOG_MATCH_5
- NSJNW_TRG_LOG_MATCH_6
- NSJNW_TRG_LOG_MATCH_7
- NSJNW_TRG_LOG_MATCH_8
- NSJNW_TRG_LOG_MATCH_9

環境変数が未設定の場合は、空文字列となります。

- 出力例は、以下の通りです。

```
testuser,ログ監視1,エラー対応JNW,2017/03/28 10:00:00 (error) error message.,2017/03/28
 10:00:00,error,,,,,,,,
testuser,ログ監視2,警告対応JNW,2017/03/28 10:00:00 (warn) warning message.,2017/03/28
 10:00:00,warn,,,,,,,,
```

■JNW投入予約リストの削除

指定したオプションに応じた削除条件を表示した後、削除を実行するか確認を行います。

表示を一部加工してあります。また、「/* */」内は内容説明のためのコメントです。

```
Remove the records matching the following conditions:
  User = testuser        /* -uオプション指定時のみ表示 */
  Target = trgname       /* -tオプション指定時のみ表示 */
  All records            /* -aオプション指定時のみ表示 */

remove?(y/n)
```

削除確認に対してyで応答後、処理の経過、および結果概要が表示されます。

結果概要は、ユーザごとに起動トリガ定義別に削除したJNW投入データの件数を表示します。

```
process: 1000 records, delete: 1000 records
process: 2000 records, delete: 2000 records
process: 3000 records, delete: 3000 records
process: 3967 records, delete: 3967 records
```

```
--------------- delete result ---------------
user=testuser1
  trg=trgname10, deleted=2910
  trg=trgname11, deleted=792
user=testuser2
  trg=trgname20, deleted=265
```

3.31.6. 注意事項

■本コマンドは、UNIX版ではrootユーザ、Windows版ではJobCenter管理者で実行してください。

■本コマンドは、JobCenterが起動している場合のみ利用可能です。

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITEを設定する必要があります。

3.32. jc_usermgr JobCenterのユーザの管理

■Windows版

- OS上のユーザをJobCenterのユーザとして登録

```
%InstallDirectory%\bin\jc_usermgr.exe add -u [$domainname\]$username -p $password
```

3.32.1. 機能説明

本コマンドは、JobCenterのユーザの管理を行う機能を提供します。

3.32.2. サブコマンド

サブコマンド	機能
add	OS上に存在しているユーザをJobCenterのユーザとして登録します。 JobCenterのユーザとして登録することで、たとえば次のようなことができるようになります。 ■単位ジョブパラメータのジョブ実行ユーザとして選択 ■ユーザマッピングのユーザ一覧として選択 ■本サブコマンドでJobCenterのユーザとして登録を行っただけでは、ユーザ環境設定やパーミッション設定を行えません。JobCenterへのログイン、またはjc_usercfgコマンドのinitサブコマンドを実行する必要があります。 jc_usercfgコマンドの詳細は「3.33 jc_usercfg ユーザ環境の作成、ユーザ環境設定の参照および変更」を参照してください。 ■JobCenterを正常に動作させるためにはJobCenterのユーザに対して、適切な権限を付与する必要があります。 必要な権限については<リリースメモ>の「3.3.5 必要な権限」を参照してください。

3.32.3. オプション

-u [$domainname\]$username

ユーザ名を指定します。ドメインユーザの場合はドメイン名も指定します。

ユーザ名、ドメイン名の制限は次のとおりです。

1.ユーザ名、ドメイン名ともに15バイト以内で指定してください。

2.ユーザ名にはスペース、タブ文字、および以下の記号は使用できません。

!"#$%&'()*,./:;<=>?@[\]^`{|}~

3.ドメイン名にはスペース、タブ文字、および以下の記号は使用できません。

/\:*?"<>|

4.コマンドを実行するサーバがドメインコントローラの場合、ドメイン名の指定は不要です。ドメイン名が指定された場合、ドメイン名部分は無視します。

5.ユーザ名、ドメイン名は大文字・小文字を区別しません。

-p $password

-uオプションで指定したユーザのパスワードを指定します。

パスワードに以下の文字が含まれる場合、それぞれ次の文字でエスケープが必要です。

エスケープが必要な文字	エスケープ文字	コマンド実行例
^	^	setuserpwd.exe user1 te^^st^<123^>45
&		
|		
<		
>		
"	\	setuserpwd.exe user1 test\"123\"45

3.32.4. 戻り値

戻り値	内容
0	正常終了です。
1	異常終了です。

3.32.5. 主要メッセージ

メッセージ	内容
Successfully added ユーザ名 to JobCenter group.	ユーザの登録に成功しました。
WARN: The specified user has already existed in the JobCenter group.	指定したユーザは既にJobCenterのユーザとして登録されています。
ERROR: Too few arguments.	引数の数が少ないです。
ERROR: Too many arguments.	引数の数が多いです。
ERROR: Invalid subcommand(サブコマンド).	無効なサブコマンドが指定されています。
ERROR: Please input user name.	ユーザ名を入力してください。
ERROR: Please input password.	パスワードを入力してください。
ERROR: User name can't more than 15 bytes.	ユーザ名には15バイトより大きい値は指定できません。
ERROR: User name can't use space, tab and the characters as shown below. The forbidden characters: !"#$%&'()*,./:;<=>?@[\]^`{|}~	ユーザ名に禁止文字が入力されています。
ERROR: Username is invalid.	ユーザ名が不正です。
ERROR: User does not exist in OS.	指定されたユーザはOSに存在しません。
ERROR: Please execute jc_usermgr.exe with the user in Administrators group.	Administratorsグループのユーザで実行してください。
ERROR: Failed to get JobCenter Administrator.	JobCenter管理者の取得に失敗しました。

3.32.6. 注意事項

■本コマンドは、OSの管理者権限を持つユーザで実行してください。

■本コマンドは、cmd.exeの右クリックメニューから「管理者として実行」を選択して起動し、コマンドラインから実行してください。

■次のような動作を行った場合、JobCenterが正常に動作しなくなる可能性があります。

- <リリースメモ>の「5.2.1　注意事項」の「使用不可ユーザ名について」に記載のユーザをJobCenterユーザとして登録する。
- ドメインユーザではなくローカルユーザをJobCenter管理者としてセットアップした後、ドメインユーザをJobCenterユーザとして登録する。
- 次のようなユーザの両方をJobCenterユーザとして登録する。
  - 同じアカウント名のローカルユーザとドメインユーザ
  - 同じアカウント名のドメインユーザと別ドメインのユーザ

3.33. jc_usercfg ユーザ環境の作成、ユーザ環境設定の参照および変更

■UNIX版

- ユーザ環境の作成

```
/usr/lib/nqs/gui/bin/jc_usercfg init [-u $username ]
```

- ユーザ環境設定の参照

```
/usr/lib/nqs/gui/bin/jc_usercfg getcfg [-u $username ]
```

- ユーザ環境設定の変更

```
/usr/lib/nqs/gui/bin/jc_usercfg setcfg [-u $username ] $paramname $paramvalue
```

■Windows版

- ユーザ環境の作成

```
%InstallDirectory%\bin\jc_usercfg.exe init [-u $username ]
```

- ユーザ環境設定の参照

```
%InstallDirectory%\bin\jc_usercfg.exe getcfg [-u $username ]
```

- ユーザ環境設定の変更

```
%InstallDirectory%\bin\jc_usercfg.exe setcfg [-u $username ] $paramname $paramvalue
```

3.33.1. 機能説明

本コマンドは、初回ログイン時のユーザ環境の作成を行う機能ならびにユーザ環境設定の参照、変更を行う機能を提供します。

3.33.2. サブコマンド

サブコマンド	機能
init	対象のユーザをJobCenterで使用できるように、CL/Winの初回接続時に行われるユーザ環境の作成を行います。 ユーザ名を省略した場合は実行ユーザを対象とします。
getcfg	対象のユーザのユーザ環境設定を参照します。 ユーザ名を省略した場合は実行ユーザを対象とします。
setcfg	対象のユーザのユーザ環境設定を変更します。 ユーザ名を省略した場合は実行ユーザを対象とします。

3.33.3. オプション

-u $username

対応のユーザ名を指定します。

このオプションはJobCenter管理者のみが利用できます。

> JobCenterとして使用できるユーザ名には制限があります。詳細は <リリースメモ>の5章「注意事項・制限事項」を参照してください。

3.33.4. 出力結果

getcfgサブコマンドの出力結果は以下の通りです。

```
==========================basic config==========================
def_queue                                guilb_def
error_action                             continue
overtime_action                          continue
erp_auto_start                           on
flow_zoom                                100

==========================tracker config========================
trk_filter                               off
trk_status_filter                        waitting,running,terminated
show_skipped_subjnw(branch skipped)      off
skipped_waitjnw_status(branch skipped)   hide
show_skipped_waitjnw(branch skipped)     on
show_arch_trk                            on
latest_trk_mode                          off
latest_trk_target                        all
latest_trk_period                        10(day)
disp_date_format                         mid(mm/dd hh:mm)
disp_start_time                          00:00
hierarchical_mode                        auto
trk_disp_period                          1(day)
trk_show_hierarchy                       on
trk_hierarchical_level                   all
trk_display_mode                         tree
auto_refresh                             on
refresh_interval                         30(second)
refresh_after_ctrl                       on

==========================archive config========================
enable_trk_archive                       on
arch_file_name                           trkarc
retention_interval                       1(day)
disable_cross_month                      on
auto_del_arch_file                       on
retention_file_num                       90
arch_stdout                              on
arch_stderr                              on
arch_log                                 on
arch_env                                 on
arch_timing_mode                         relative
relative_time                            10(minute)
absolute_time                            00:00
include_errstop_trk                      off
```

3.33.5. パラメータ一覧

ユーザ環境設定で参照・変更できるパラメータは以下となります。

パラメータ	説明
def_queue	ジョブネットワークに属する各ジョブを投入するキューの既定値を設定できます。 <キュー名>@<マシン名>の形式で設定を行います。 <キュー名>および<マシン名>の制限は以下の通りです。 ■キュー名 ▪ 最大バイト数は15バイトです。 ▪ 半角英数字、アンダーラインのみ指定可能です。また、最初の文字を数字にすることはできません。 ■マシン名 ▪ 最大バイト数は63バイトです。 ▪ 数値のみの名前は使用できません。 ▪ スペース、タブ、改行および以下の文字は使用できません。 ! " # $ % & ' ( ) * , / : ; < = + > ? @ [\ ] ^ ` { | } ~ ユーザ環境作成時はデフォルト値として、自マシンの「guilb_def」が設定されます。
error_action	エラー発生時のジョブネットワークのデフォルト動作を設定できます。 ■stop エラー発生時にジョブネットワークを停止します。 ■continue エラー発生時にジョブネットワークを停止しません。 ■abort エラーが発生時にジョブネットワークの実行を中断します。 ユーザ環境作成時はデフォルト値として「continue」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
overtime_action	ジョブネットワークまたは単位ジョブの実行が終了予定時刻を超過した場合の動作を設定できます。 ■continue ジョブネットワークまたは単位ジョブの実行を継続します。 ■stop ジョブネットワークまたは単位ジョブの実行を停止します。 ■skip ジョブネットワークまたは単位ジョブの実行をスキップします。

パラメータ	説明
	ユーザ環境作成時はデフォルト値として「continue」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
erp_auto_start	ERPジョブを自動スタートするかを設定できます。 ■on JobCenterにより自動的に実行状態になります。 ■off 対象のERPジョブを投入されるとSAP　ERPシステムへのジョブ登録処理が行われます。ただしスタート処理は行われず、[SUBMIT]状態となったまま実行操作の待ち状態となります。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
flow_zoom	ジョブネットワークのフローの表示倍率におけるデフォルト値を設定できます。 表示倍率は10～100の範囲で設定できます。 ユーザ環境作成時はデフォルト値として「100」が設定されます。
trk_filter	トラッカの表示条件の設定を使用するか設定できます。 ■on 表示条件の設定を使用します。 ■off 表示条件の設定を使用しません。 ユーザ環境作成時はデフォルト値として「off」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
trk_status_filter	トラッカの表示条件として、表示するトラッカのステータスの条件を設定できます。 以下の中から条件を選択します。条件を複数選択したい場合は「,」で区切って指定します。 (例)　waiting、runningおよびterminatedを設定したい場合は「waitting,running,terminated」を指定します。 ■waitting まだ実行が開始されていないトラッカのみを表示します。 ■running 実行が開始されているトラッカのみを表示します。 ■terminated すでに実行が終了しているトラッカのみを表示します。

パラメータ	説明
	■error 異常が発生したトラッカのみを表示します。 ■warning 警告が発生したトラッカのみを表示します。 ■stop エラー停止したトラッカのみを表示します。 ■skip スキップ状態のトラッカのみを表示します。 ■hold 保留状態のトラッカのみを表示します。 ■wait 待ち合わせ状態のトラッカのみを表示します。 ユーザ環境作成時はデフォルト値として 「waitting,running,terminated」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
show_skipped_subjnw	トラッカの表示条件として、分岐スキップ済サブJNWを表示するかを設定できます。 ■on 分岐スキップ済サブJNWを表示します。 ■off 分岐スキップ済サブJNWを表示しません。 ユーザ環境作成時はデフォルト値として「off」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
skipped_waitjnw_status	トラッカの表示条件として、予定状態のトラッカについて実際の処理分岐（カレンダ分岐）を判定するかしないかを設定できます。 「skipped_waitjnw_status」を「show」に設定した上で 「show_skipped_waitjnw」を「on」に設定することで、予定状態のトラッカについてカレンダ分岐により実行されないサブJNWを判定し、分岐スキップ済みとして表示できます。 ■show 予定状態のトラッカについてカレンダ分岐により実行されないサブJNWを判定します。 ■hide

パラメータ	説明
	予定状態のトラッカについてカレンダ分岐により実行されないサブJNWを判定しません。 ユーザ環境作成時はデフォルト値として「hide」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
show_skipped_waitjnw	「skipped_waitjnw_status」が「show」となっている時に予定状態のトラッカについてカレンダ分岐により実行されないサブJNWを表示するかを設定できます。 ■on 予定状態のトラッカについてカレンダ分岐により実行されないサブJNWを表示します。 ■off 予定状態のトラッカについてカレンダ分岐により実行されないサブJNWを表示しません。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
show_arch_trk	アーカイブ済みのトラッカを表示するかしないかを設定できます。 ■on アーカイブ済みのトラッカも表示します。 ■off アーカイブ済みのトラッカは表示しません。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
latest_trk_mode	トラッカ一覧画面を全表示モードで表示するか、最新トラッカ表示モードで表示するかを設定できます。 ■on 最新トラッカ表示モードで表示します。 ■off 全表示モードで表示します。 ユーザ環境作成時はデフォルト値として「off」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
latest_trk_target	「latest_trk_mode」を「on」に設定したとき、最新トラッカ表示モードで表示するトラッカの条件を設定できます。 ■all 過去全てのトラッカを検索して、最新のトラッカのみを表示します。

パラメータ	説明
	■period 「latest_trk_period」で設定された期間内のトラッカを検索して、最新のトラッカのみを表示します。 ユーザ環境作成時はデフォルト値として「all」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
latest_trk_period	「latest_trk_target」を「period」に設定した時に、トラッカを検索する期間を設定できます。 期間は1～9999の範囲で設定できます。 ユーザ環境作成時はデフォルト値として「10」が設定されます。
disp_date_format	トラッカリストの表示をテキストモードにした場合に、表示される予定および実績開始終了時間の表示形式のデフォルト値を設定できます。 ■short 「hh:mm」形式で時分を表示します。 ■mid 「MM/DD hh:mm」形式で月日時分を表示します。 ■long 「YYYY/MM/DD hh:mm:ss」形式で年月日時分秒を表示します。 ユーザ環境作成時はデフォルト値として「mid」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
disp_start_time	トラッカリストの表示を開始する先頭の時刻を設定できます。 時刻は0～23(時)の範囲での整数で設定できます。 ユーザ環境作成時はデフォルト値として「0」が設定されます。
hierarchical_mode	サブジョブネットワークのトラッカの階層を自動的に表示するかどうかのデフォルト値を設定できます。 ■select 選択した状態を保持します。 ■auto 自動的に表示します。 ■non_auto 自動的に表示しません。 ユーザ環境作成時はデフォルト値として「auto」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
trk_disp_period	トラッカリストの表示期間のデフォルト値を設定できます。

パラメータ	説明
	表示期間は1～59分、1～23時、1～7日、1～5週、1～3ヶ月の範囲で設定できます。分指定の場合は「min」、時間指定の場合は「h」、日指定の場合は「d」、週指定の場合は「w」、月指定の場合は「m」をつけて指定します。 (例) 12時間の場合は「12h」、1ヶ月の場合は「1m」 ユーザ環境作成時はデフォルト値として「1(day)」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
trk_show_hierarchy	トラッカの表示形式を階層表示として表示するか設定できます。 ■on 常にルートジョブネットワークから階層表示します。 ■off 指定した条件に合うトラッカのみを表示します。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
trk_hierarchical_level	トラッカ表示モードに表示されるジョブネットワークの階層レベルを設定できます。 階層レベルには「all」または、1～40の範囲の数値で設定できます。 ■all すべてのジョブネットワークを表示します。 ■1～40 設定した階層までのジョブネットワークを表示します。 ユーザ環境作成時はデフォルト値として「all」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
trk_display_mode	トラッカ表示モードのデフォルト値を設定できます。 ■tree トラッカをツリー構造で階層的に表示します。 ■time 予定開始時間の昇順に並べ替えて表示します。 ■sort マシングループに属するすべてのマシンまたは該当マシンごとに、予定開始時間の昇順で並べ替えて表示します。 ユーザ環境作成時はデフォルト値として「tree」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。

パラメータ	説明
auto_refresh	トラッカやリクエストに関するウィンドウの更新を自動で行うか設定できます。「on」に設定した場合の更新間隔は「refresh_interval」の設定に従います。 ■on 一定間隔で表示更新を行います。 ■off 自動更新を行いません。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
refresh_interval	「auto_refresh」を「on」に設定したときに表示更新を行う間隔を設定できます。 間隔は10～3600秒または1～60分の範囲で設定できます。秒指定の場合は「s」、分指定の場合は「min」をつけて指定します。 (例) 40秒の場合は「40s」、30分の場合は「30min」 ユーザ環境作成時はデフォルト値として「30(second)」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
refresh_after_ctrl	トラッカ操作後にトラッカ一覧画面の全件更新を行うか設定できます。 ■on トラッカ操作後にトラッカ一覧画面の全件更新を行います。 ■off トラッカ操作後にトラッカ一覧画面の全件更新を行いません。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
enable_trk_archive	トラッカ情報をトラッカアーカイブファイルに保存するかどうかを設定できます。 ■on 保存します。 ■off 保存しません。(アーカイブ待ち時間経過後にトラッカは削除されます) ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
arch_file_name	トラッカ情報を保存するトラッカアーカイブファイル名を設定できます。

パラメータ	説明
	トラッカアーカイブファイル名の制限は以下の通りです。 ■2バイト文字および以下の文字は使用できません。 「+」「'」「#」「,」「;」「*」「?」「"」「<」「>」「|」 ■192バイト以内で指定してください。 ■アーカイブファイルへのフルパスがOS毎のパスの長さ制限に収まっている必要がありますのでご注意ください。 アーカイブファイルへのフルパスの長さは以下のとおりです。 JobCenterのインストールディレクトリまたはサイトデータベースのパス長 + アーカイブファイル名の長さ + 53バイト ユーザ環境作成時はデフォルト値として「trkarc」が設定されます。
retention_interval	アーカイブファイルの保存間隔を設定できます。 保存間隔は1～365日、1～53週、1～12ヶ月の範囲で設定できます。日指定の場合は「d」、週指定の場合は「w」、月指定の場合は「m」をつけて指定します。 (例) 10日の場合は「10d」、1ヶ月の場合は「1m」 「retention_interval」と「retention_file_num」の設定値はそれらの合計が1年間より小さい値になるよう設定する必要があります。 ユーザ環境作成時はデフォルト値として「1(day)」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
disable_cross_month	保存間隔の設定で、単位を期間（週）に設定した場合に、アーカイブファイルの保存間隔が月を跨がないか設定できます。 ■on 月を跨ぎません。 ■off 月を跨ぎます。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
auto_del_arch_file	保存期間を経過したアーカイブファイルを自動削除するかどうかを設定できます。 ■on 自動削除します。 ■off 自動削除しません。 ユーザ環境作成時はデフォルト値として「on」が設定されます。

パラメータ	説明
	setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
retention_file_num	保存期間を経過したアーカイブファイルを保存する個数を設定できます。 保存期間の個数は0～90の範囲で設定できます。 ユーザ環境作成時はデフォルト値として「90」が設定されます。 「retention_interval」と「retention_file_num」の設定値はそれらの合計が1年間より小さい値になるよう設定する必要があります。
arch_stdout	各ジョブの標準出力の内容をトラッカアーカイブファイルに保存するかどうかを設定できます。 ■on 保存します。 ■off 保存しません。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
arch_stderr	各ジョブの標準エラー出力の内容をトラッカアーカイブファイルに保存するかどうかを設定できます。 ■on 保存します。 ■off 保存しません。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
arch_log	ジョブネットワーク実行時のログをトラッカアーカイブファイルに保存するかどうかを設定できます。 ■on 保存します。 ■off 保存しません。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
arch_env	ジョブネットワーク実行時の環境変数をトラッカアーカイブファイルに保存するかどうかを設定できます。 ■on

パラメータ	説明
	保存します。 ■off 保存しません。 ユーザ環境作成時はデフォルト値として「on」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
arch_timing_mode	トラッカの削除、または、アーカイブするタイミングを設定できます。 ■relative ジョブネットワークの実行が完了してから、待ち時間で設定した時間ほど経過したらトラッカの削除、または、アーカイブを行います。 ■absolute ジョブネットワークの実行が完了してから、次の指定した時刻に到達した時にトラッカの削除、または、アーカイブを行います。 ユーザ環境作成時はデフォルト値として「relative」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
relative_time	「arch_timing_mode」を「relative」に設定したときのジョブネットワークの実行が完了してから、トラッカを削除、または、アーカイブするまでの待ち時間を設定できます。 待ち時間は0～2678400秒、0～44640分、0～744時間、0～31日の範囲で設定できます。秒指定の場合は「s」、分指定の場合は「min」、時間指定の場合は「h」、日指定の場合は「d」をつけて指定します。 (例) 5分の場合は「5min」、1日の場合は「1d」 ユーザ環境作成時はデフォルト値として「10(minute)」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
absolute_time	「arch_timing_mode」を「absolute」に設定したときのトラッカを削除、または、アーカイブする時刻を設定できます。 時刻は00:00~23:59の範囲で設定できます。 ユーザ環境作成時はデフォルト値として「00:00」が設定されます。
include_errstop_trk	エラー停止状態のトラッカをトラッカ削除/アーカイブのタイミングの設定に従い、削除、または、アーカイブするかを設定できます。 ■on エラー停止状態のトラッカをトラッカ削除/アーカイブのタイミングの設定に従い、削除、または、アーカイブします。 ■off エラー停止状態のトラッカをトラッカ削除/アーカイブのタイミングの設定に従い、削除、または、アーカイブしません。

パラメータ	説明
	ユーザ環境作成時はデフォルト値として「off」が設定されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。

3.33.6. 戻り値

戻り値	内容
0	正常終了です。
1	異常終了です。

3.33.7. 主要メッセージ

■コマンド共通のメッセージ

メッセージ	内容
ERROR: Too few arguments.	引数の数が少ないです。
ERROR: Invalid subcommand($subcommand).	無効なサブコマンドが指定されています。
ERROR: It is only JobCenter manager that can use '-u' option.	JobCenter管理者以外のユーザが-uオプションを使用することはできません。
ERROR: User name can't more than 15 bytes.	ユーザ名の長さの制限を超えています。ユーザ名は15byteまでにする必要があります。
ERROR: Invalid character.	ユーザ名に使用できない文字が含まれています。
ERROR: Invalid site.	環境変数NQS_SITEで指定されたサイト名に問題があります。

■initサブコマンド固有のメッセージ

メッセージ	内容
Successfully initialized $username.	ユーザ環境の作成に成功しました。
WARN: User is already initialized.	対象のユーザのユーザ環境は既に作成されています。
ERROR: User does not exist (user=$username).	対象のユーザが存在しません。

■getcfgサブコマンド固有のメッセージ

メッセージ	内容
ERROR: You are not initialized user.	コマンドの実行ユーザのユーザ環境が作成されていません。
ERROR: User does not exist (user=$username).	対象のユーザが存在しません。

■setcfgサブコマンド固有のメッセージ

メッセージ	内容
ERROR: You are not initialized user.	コマンドの実行ユーザのユーザ環境が作成されていません。
ERROR: User does not exist (user=$username).	対象のユーザが存在しません。
ERROR: Set userconfig failure("No "$paramname" key.")	指定されたパラメータ名が誤っています。
ERROR: Set userconfig failure("Invalid parameter value.")	指定されたパラメータの値が不正です。
ERROR: Get lock failure.	他のユーザが対象のユーザのユーザ環境設定を編集しているため、ユーザ環境の設定に失敗しました。

3.33.8. 注意事項

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITEを設定する必要があります。

詳細は 1章 「コマンド一覧」 の注意事項を参照してください。

■getcfgおよびsetcfgサブコマンドは、コマンド自体の実行ユーザおよび対象とするユーザの両方がユーザ環境を作成したユーザの必要があります。ユーザ環境の作成は、JobCenterへのログインまたはinitサブコマンドで実施してください。

■Windows版の場合はJobCenter管理者についてもインストール時にユーザ環境の作成が行われません。そのためgetcfgおよびsetcfgを使用したい場合は一度、JobCenterへのログインまたはinitサブコマンドでユーザ環境の作成を行う必要が有ります。

■setcfgサブコマンドで設定できる各パラメータの詳細については、<環境構築ガイド>の「6.2　ユーザ環境の設定を変更する」を参照してください。

■setcfgサブコマンドで「def_queue」を設定するとき、指定したキュー名が存在するかはチェックされませんのでご注意ください。

3.34. jc_syscfg システム環境設定の参照および変更

■UNIX版

- システム環境設定の参照

```
/usr/lib/nqs/gui/bin/jc_syscfg getcfg [ -p $paramname | -c $category ]
```

- システム環境設定の変更

```
/usr/lib/nqs/gui/bin/jc_syscfg setcfg $paramname $paramvalue
```

■Windows版

- システム環境設定の参照

```
%InstallDirectory%\bin\jc_syscfg.exe getcfg [ -p $paramname | -c $category ]
```

- システム環境設定の変更

```
%InstallDirectory%\bin\jc_syscfg.exe setcfg $paramname $paramvalue
```

3.34.1. 機能説明

本コマンドは、システム環境設定の参照、変更を行う機能を提供します。

3.34.2. サブコマンド

サブコマンド	機能
getcfg	システム環境設定を参照します。
setcfg	$paramnameで指定したシステム環境設定を$paramvalueで指定した値に変更します。

3.34.3. オプション

-p $paramname 参照するシステム環境設定のパラメータ名を指定します。本オプションを省略した場合、全てのシステム環境設定を対象とします。 指定できるパラメータは「3.34.5 パラメーター一覧」を参照してください。
-c $category 参照するシステム環境設定のカテゴリを指定します。本オプションを省略した場合、全てのシステム環境設定を対象とします。 指定できるカテゴリは「3.34.5 パラメーター一覧」を参照してください。

3.34.4. 出力結果

getcfgサブコマンドの出力結果は以下の通りです。

```
==========================time config==========================
time.customized             off
time.mode                   all
time.shift                  00:00
time.zone                   off

==========================op&exec log==========================
op.apply                    off
```

```
op.period                   1(year)
op.baknum                   0
op.size                     1(mb)
exec.apply                  off
exec.period                 1(year)
exec.baknum                 0
exec.size                   1(mb)

==========================smtp server==========================
smtp.mode                   each
smtp.hostname
smtp.port                   25
smtp.auth                   off
smtp.auth.cram.md5          off
smtp.auth.login             off
smtp.auth.plain             off
smtp.userid
smtp.password
```

3.34.5. パラメータ一覧

システム環境設定で参照・変更できるパラメータ、及びカテゴリは以下となります。

カテゴリ	パラメータ	説明
time	time.customized	補正時刻の使用有無を表します。 ■on 補正時刻を使用します。 ■off 補正時刻を使用しません。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
	time.mode	補正時刻の設定方法を表します。 ■all サイト全体に設定された補正時刻が適用されます。 ■each ジョブネットワーク毎に補正時刻が設定できます。 time.modeパラメータを「each」に設定した場合、time.shiftパラメータの値が「00:00」に変更されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
	time.shift	サイト全体に補正時刻を適用する場合の補正時刻を表します。 補正時刻は、time.modeパラメータが「all」の場合に設定が可能で、0時0分からの相対時刻指定で、-23:59～23:59の範囲で設定を行うことができます。
	time.zone	カレンダへのタイムゾーン設定機能の使用有無を表します。 ■on

カテゴリ	パラメータ	説明
		カレンダへのタイムゾーン設定機能を使用します。 ■off カレンダへのタイムゾーン設定機能を使用しません。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
log	op.apply	操作ログの出力有無を表します。 ■on 操作ログを出力します。 ■off 操作ログを出力しません。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
	op.period	操作ログの保存期間を表します。 保存期間は、1～365日間、または1～5年間の範囲で設定を行うことができます。日指定の場合は「d」を、年指定の場合は「y」を付けて指定します。 (例) 30日間の場合は「30d」、5年間の場合は「5y」 保持期間を「365d」で設定した場合、getcfgサブコマンドでの出力は「1(year)」として表示されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
	op.baknum	操作ログのバックアップファイルの数を表します。 バックアップファイルの数は、0～9の範囲で設定を行うことができます。
	op.size	操作ログの1ファイルの最大サイズを表します。 1ファイルの最大サイズは、1～1024KB、または1～1024MBの範囲で設定を行うことができます。KB指定の場合は「kb」を、MB指定の場合は「mb」を付けて指定します。 (例) 100KBの場合は「100kb」、512MBの場合は「512mb」 1ファイルの最大サイズを「1024kb」で設定した場合、getcfgサブコマンドでの出力は「1(mb)」として表示されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
	exec.apply	実行ログの出力有無を表します。 ■on 実行ログを出力します。 ■off 実行ログを出力しません。

カテゴリ	パラメータ	説明
		setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
	exec.period	実行ログの保存期間を表します。 保存期間は、1～365日間、または1～5年間の範囲で設定を行うことができます。日指定の場合は「d」を、年指定の場合は「y」を付けて指定します。 (例) 30日間の場合は「30d」、5年間の場合は「5y」 保持期間を「365d」で設定した場合、getcfgサブコマンドでの出力は「1(year)」として表示されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
	exec.baknum	実行ログのバックアップファイルの数を表します。 バックアップファイルの数は、0～9の範囲で設定を行うことができます。
	exec.size	実行ログの1ファイルの最大サイズを表します。 1ファイルの最大サイズは、1～1024KB、または1～1024MBの範囲で設定を行うことができます。KB指定の場合は「kb」を、MB指定の場合は「mb」を付けて指定します。 (例) 100KBの場合は「100kb」、512MBの場合は「512mb」 1ファイルの最大サイズを「1024kb」で設定した場合、getcfgサブコマンドでの出力は「1(mb)」として表示されます。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
smtp	smtp.mode	SMTPサーバの指定方法を表します。 ■all サイト全体に設定されたSMTPサーバが適用されます。 ■each ジョブネットワーク毎にSMTPサーバが設定できます。 smtp.modeパラメータを「all」に設定する場合、smtp.hostname、smtp.portパラメータを設定する必要があります。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
	smtp.hostname	SMTPサーバ名を表します。 SMTPサーバ名を設定する場合は、SMTPサーバのホスト名、またはIPアドレスを指定します。SMTPサーバ名は256バイトまで入力することができます。また、空白、タブ、半角カタカナ、全角文字、及び「!」「"」「#」「$」「%」「&」「'」「(」「)」「*」「,」「/」「;」「<」「=」「>」「?」「@」「[」「\」「]」「^」「`」「{」「|」「}」「~」は使用できません。
	smtp.port	SMTPサーバのポート番号を表します。 ポート番号は、1～65535の範囲で設定を行うことができます。

カテゴリ	パラメータ	説明
	smtp.auth	SMTPユーザ認証機能の利用有無を表します。 ■on SMTPユーザ認証機能を利用します。 ■off SMTPユーザ認証機能を利用しません。 smtp.authパラメータを「on」に設定する場合、smtp.userid、smtp.passwordパラメータの設定、及びsmtp.auth.cram.md5、smtp.auth.login、smtp.auth.plainパラメータのいずれかが「on」に設定されている必要があります。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。
	smtp.auth.cram.md5	SMTPユーザ認証機能のユーザ認証方式として、CRAM-MD5の利用有無を表します。 ■on ユーザ認証方式にCRAM-MD5を利用します。 ■off ユーザ認証方式にCRAM-MD5を利用しません。 smtp.authパラメータが「on」に設定されている状態で、smtp.auth.cram.md5パラメータを「off」に設定する場合、smtp.auth.login、smtp.auth.plainパラメータのいずれかが「on」に設定されている必要があります。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。 ユーザ認証方式を複数設定した場合は、CRAM-MD5→LOGIN→PLAINの優先順位で、順次SMTPサーバへの認証を試みます。
	smtp.auth.login	SMTPユーザ認証機能のユーザ認証方式として、LOGINの利用有無を表します。 ■on ユーザ認証方式にLOGINを利用します。 ■off ユーザ認証方式にLOGINを利用しません。 smtp.authパラメータが「on」に設定されている状態で、smtp.auth.loginパラメータを「off」に設定する場合は、smtp.auth.cram.md5、smtp.auth.plainパラメータのいずれかが「on」に設定されている必要があります。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。 ユーザ認証方式を複数設定した場合は、CRAM-MD5→LOGIN→PLAINの優先順位で、順次SMTPサーバへの認証を試みます。

カテゴリ	パラメータ	説明
	smtp.auth.plain	SMTPユーザ認証機能のユーザ認証方式として、PLAINの利用有無を表します。 ■on ユーザ認証方式にPLAINを利用します。 ■off ユーザ認証方式にPLAINを利用しません。 smtp.authパラメータが「on」に設定されている状態で、smtp.auth.plainパラメータを「off」に設定する場合は、smtp.auth.cram.md5、smtp.auth.loginパラメータのいずれかが「on」に設定されている必要があります。 setcfgサブコマンドで設定する場合、大文字・小文字は区別しません。 ユーザ認証方式を複数設定した場合は、CRAM-MD5→LOGIN→PLAINの優先順位で、順次SMTPサーバへの認証を試みます。
	smtp.userid	SMTPユーザ認証機能で使用するユーザIDを表します。 ユーザIDは255バイトまで入力することができます。また、タブ、半角カタカナ、全角文字、特殊文字、及び「"」「#」「'」は使用できません。
	smtp.password	SMTPユーザ認証機能で使用するユーザIDに対応するパスワードを表します。パスワードが設定されている場合、マスクした値で表示されます。 パスワードは64バイトまで入力することができます。

3.34.6. 戻り値

戻り値	内容
0	正常終了です。
1	異常終了です。

3.34.7. 主要メッセージ

■コマンド共通のメッセージ

メッセージ	内容
ERROR: Too few arguments.	コマンドの引数が不足しています。
ERROR: Too many arguments.	コマンドの引数が多いです。
ERROR: Invalid subcommand(サブコマンド).	無効なサブコマンドが指定されています。
ERROR: This user is not registed.	コマンドを実行しているユーザはJobCenterグループに所属していません。
ERROR: User does not exist.	コマンドを実行しているユーザのユーザ環境が作成されていません。
ERROR: Invalid site.	環境変数NQS_SITEで指定したサイトが不正です。

■getcfgサブコマンド固有のメッセージ

メッセージ	内容
ERROR: The specified parameter not found.	指定されたパラメータは存在しません。

メッセージ	内容
ERROR: The specified category not found.	指定されたカテゴリは存在しません。
ERROR: Paramname can't more than 40 bytes.	パラメータ名には40バイトより大きい値は指定できません。
ERROR: Get sysconfig failure("エラーメッセージ")	システム環境設定の取得に失敗しました。

■setcfgサブコマンド固有のメッセージ

メッセージ	内容
Successfully set system config.	システム環境設定の設定に成功しました。
ERROR: Set system config failure("No "パラメータ名" key.")	指定したパラメータが存在しないため、システム環境設定の設定に失敗しました。
ERROR: Set system config failure("Invalid parameter value.")	指定したパラメータの値が不正なため、システム環境設定の設定に失敗しました。
ERROR: Set system config failure("The required paramaters aren't proper.")	依存関係のあるパラメータの値が不適切であるため、システム環境設定の設定に失敗しました。
ERROR: Set system config failure("Can't set the time.shift paramater when the time.mode is "each".")	time.modeパラメータが「each」であるため、time.shiftパラメータの設定に失敗しました。
ERROR: Set system config failure("エラーメッセージ")	システム環境設定の設定に失敗しました。
ERROR: Only manager can execute this command.	JobCenter管理者でのみ実行できるサブコマンドです。
ERROR: Get lock failure.	システム環境設定が編集中のため、設定に失敗しました。

3.34.8. 注意事項

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITEを設定する必要があります。

詳細は 1章 「コマンド一覧」 の注意事項を参照してください。

■本コマンドは、JobCenterへのログイン、またはjc_usercfgコマンドのinitサブコマンドでユーザの環境を作成したユーザで実行する必要があります。

jc_usercfgコマンドの詳細は 「3.33 jc_usercfg ユーザ環境の作成、ユーザ環境設定の参照および変更」を参照してください。

■setcfgサブコマンドは、JobCenter管理者で実行する必要があります。

3.35. jc_permcfg パーミッション設定の参照、および変更

■UNIX版

```
/usr/lib/nqs/gui/bin/jc_permcfg $subcommand $options...
```

■Windows版

```
%InstallDirectory%\bin\jc_permcfg.exe $subcommand $options...
```

3.35.1. 機能説明

本コマンドは、パーミッション設定の参照、変更を行う機能を提供します。

本コマンドのサブコマンドは以下の通りです。

サブコマンド	機能
getperm	権限グループのパーミッション情報を出力します。
creategrp	権限グループを新規作成します。
delgrp	権限グループを削除します。
copygrp	権限グループを複製します。
setperm	権限グループのアクセス権限を設定します。
assignuser	ユーザの所属する権限グループを変更します。
addaccuser	権限グループに他ユーザへのアクセス許可を追加します。
delaccuser	権限グループから他ユーザへのアクセス許可を削除します。

それぞれのサブコマンドの使い方とオプションについては以降の節で説明します。

■デフォルト権限グループについて

本コマンドでは、デフォルトの権限グループは以下の名前で入出力します。(CL/Winの画面上では末尾にアスタリスク「*」が付いて表示されますが、本コマンドではアットマーク「@」で始まる文字列で入出力します。)

権限グループ名	本コマンドで表示・入力する名称
JobCenter管理者	@JobCenterManager
ジョブネットワーク開発者	@JnwDeveloper
ジョブネットワーク運用者	@JnwOperator
実行監視者	@ExecSurveillant
一般ユーザ	@NormalUser

権限グループ名の制限は以下の通りです。

- 権限グループ名の最大バイト数は、40バイトです。
- 空白、タブ、特殊文字および以下の文字は使用できません。

  ! " # $ % ' ( ) * , . / : ; < = > ? @ [\ ] ^ ` { | } ~

- デフォルトの権限グループを指定する場合を除き、権限グループ名にアットマーク「@」は使用できません。

3.35.1.1. 権限グループのパーミッション情報の参照（getperm）

■コマンドラインインタフェース

```
jc_permcfg getperm [-g $groupname ]
```

■概要

権限グループのパーミッション情報を出力します。

■オプション

-g $groupname

パーミッション情報を出力したい権限グループ名を指定します。

本オプション省略時は、すべての権限グループのパーミッション情報を出力します。

■出力フォーマット

権限グループのパーミッション情報の出力フォーマットを説明します。

```
<権限グループ名>
  Member
    <所属ユーザ一覧>
  Authority
    <アクセス権限一覧>
  Permitted User
    <アクセス許可ユーザ一覧>
```

セクション	説明
Member	権限グループに所属しているユーザ名の一覧がカンマ区切りで表示されます。
Authority	各アクセス権限の有無が「権限文字列 <on/off>」の形式で表示されます。 権限文字列とそれに対応するアクセス権限は以下の通りです。

権限文字列	アクセス権限
own.jnw.read	ユーザ自身のジョブネットワークの一覧と参照
own.jnw.write	ユーザ自身のジョブネットワークの作成/変更/削除
own.jnw.execute	ユーザ自身のジョブネットワークの実行
own.trk.read	ユーザ自身のトラッカの一覧と参照
own.trk.operation	ユーザ自身のトラッカの操作
other.jnw.read	他ユーザのジョブネットワークの一覧と参照
other.jnw.write	他ユーザのジョブネットワークの作成/変更/削除
other.jnw.execute	他ユーザのジョブネットワークの実行
other.trk.read	他ユーザのトラッカの一覧と参照
other.trk.operation	他ユーザのトラッカの操作

Permitted User	権限グループからアクセス許可のあるユーザ名の一覧がカンマ区切りで表示されます。

■実行例

以下にUNIX版の実行例を示します。

- 指定した権限グループの情報を出力

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root
```

- すべての権限グループの情報を出力

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm
@JobCenterManager
  Member
    nsumsmgr,root
  Authority
    own.jnw.read         on
    own.jnw.write        on
    own.jnw.execute      on
    own.trk.read         on
    own.trk.operation    on
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root

@JnwDeveloper
  Member

  Authority
    own.jnw.read         on
    own.jnw.write        on
    own.jnw.execute      on
    own.trk.read         on
    own.trk.operation    on
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root

@JnwOperator
```

```
  Member

  Authority
    own.jnw.read         on
    own.jnw.write        off
    own.jnw.execute      on
    own.trk.read         on
    own.trk.operation    on
    other.jnw.read       on
    other.jnw.write      off
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root

@ExecSurveillant
  Member

  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         on
    own.trk.operation    off
    other.jnw.read       off
    other.jnw.write      off
    other.jnw.execute    off
    other.trk.read       on
    other.trk.operation  off
  Permitted User
    jnwmonitor,nsumsmgr,root

他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root

@NormalUser
  Member

  Authority
    own.jnw.read         on
    own.jnw.write        on
```

```
    own.jnw.execute      on
    own.trk.read         on
    own.trk.operation    on
    other.jnw.read       off
    other.jnw.write      off
    other.jnw.execute    off
    other.trk.read       off
    other.trk.operation  off
  Permitted User
```

■主要メッセージ

「3.35.3 コマンド共通の主要メッセージ」を参照してください。

3.35.1.2. 権限グループの新規作成（creategrp）

■コマンドラインインタフェース

```
jc_permcfg creategrp $groupname
```

■概要

権限グループの新規作成を行います。

本サブコマンドで作成した権限グループはすべての権限が無効になっているので、「3.35.1.5　権限グループのアクセス権限を変更（setperm）」、「3.35.1.7　権限グループに他ユーザへのアクセス許可を追加（addaccuser）」で権限を設定してください。

■オプション

```
$groupname

新規作成する権限グループ名を指定します。
```

既に存在する権限グループと同じ名前を指定することはできません。

■実行例

以下にUNIX版の実行例を示します。

```
# /usr/lib/nqs/gui/bin/jc_permcfg creategrp 新規権限グループ
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 新規権限グループ
新規権限グループ
  Member

  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       off
    other.jnw.write      off
    other.jnw.execute    off
    other.trk.read       off
    other.trk.operation  off
```

```
  Permitted User
```

■主要メッセージ

本サブコマンド固有のメッセージを説明します。その他のメッセージは「3.35.3　コマンド共通の主要メッセージ」を参照してください。

メッセージ	内容
Successfully set the permission.	権限グループの作成に成功しました。
ERROR: The specified group already exists.	指定した名前の権限グループは既に存在します。

3.35.1.3. 権限グループの削除（delgrp）

■コマンドラインインタフェース

```
jc_permcfg delgrp $groupname
```

■概要

権限グループの削除を行います。

■オプション

$groupname

削除する権限グループ名を指定します。

デフォルトの権限グループは削除できません。

■実行例

以下にUNIX版の実行例を示します。

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root
# /usr/lib/nqs/gui/bin/jc_permcfg delgrp 他ユーザ監視グループ
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
ERROR: The specified group could not be found.
```

■主要メッセージ

本サブコマンド固有のメッセージを説明します。その他のメッセージは「3.35.3　コマンド共通の主要メッセージ」を参照してください。

メッセージ	内容
Successfully set the permission.	権限グループの削除に成功しました。
ERROR: Can't specify the group name to be a built-in group.	デフォルトの権限グループは削除できません。

3.35.1.4. 権限グループの複製（copygrp）

■コマンドラインインタフェース

```
jc_permcfg copygrp $srcgroup $dstgroup
```

■概要

既に存在する権限グループと同じ設定内容を持つ権限グループを作成します。

■オプション

$srcgroup

複製元となる権限グループ名を指定します。

$dstgroup

複製先の権限グループ名を指定します。

$srcgroupにJobCenter管理者グループ「@JobCenterManager」を指定することはできません。

$dstgroupに既に存在する権限グループと同じ名前を指定することはできません。

■実行例

以下にUNIX版の実行例を示します。

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root
# /usr/lib/nqs/gui/bin/jc_permcfg copygrp 他ユーザ監視グループ コピーグループ
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g コピーグループ
コピーグループ
```

```
Member

Authority
  own.jnw.read          off
  own.jnw.write         off
  own.jnw.execute       off
  own.trk.read          off
  own.trk.operation     off
  other.jnw.read        on
  other.jnw.write       on
  other.jnw.execute     on
  other.trk.read        on
  other.trk.operation   on
Permitted User
  jnwmonitor,nsumsmgr,root
```

■主要メッセージ

本サブコマンド固有のメッセージを説明します。その他のメッセージは「3.35.3　コマンド共通の主要メッセージ」を参照してください。

メッセージ	内容
Successfully set the permission.	権限グループの複製に成功しました。
ERROR: The specified srcgroup could not be found.	複製元に指定した名前の権限グループが見つかりません。
ERROR: The specified dstgroup already exists.	複製先に指定した名前の権限グループは既に存在します。
ERROR: Can't copy from JobCenter manager group.	JobCenter管理者グループを複製することはできません。

3.35.1.5. 権限グループのアクセス権限を変更（setperm）

■コマンドラインインタフェース

```
jc_permcfg setperm $groupname $mode
```

■概要

権限グループのアクセス権限を変更します。

■オプション

$groupname

アクセス権限を変更する権限グループ名を指定します。

$mode

アクセス権限を文字列で指定します。指定方法は、「アクセス権限の指定方法」を参照してください。

デフォルトの権限グループのアクセス権限を変更することはできません。

■アクセス権限の指定方法

対象の権限グループに付与したいアクセス権限の権限文字列を指定します。複数の権限文字列を指定する場合はカンマ区切りで指定します。指定できる権限文字列は以下の通りです。

権限文字列	アクセス権限名
own.jnw.read	ユーザ自身のジョブネットワークの一覧と参照
own.jnw.write	ユーザ自身のジョブネットワークの作成/変更/削除
own.jnw.execute	ユーザ自身のジョブネットワークの実行
own.trk.read	ユーザ自身のトラッカの一覧と参照
own.trk.operation	ユーザ自身のトラッカの操作
other.jnw.read	他ユーザのジョブネットワークの一覧と参照
other.jnw.write	他ユーザのジョブネットワークの作成/変更/削除
other.jnw.execute	他ユーザのジョブネットワークの実行
other.trk.read	他ユーザのトラッカの一覧と参照
other.trk.operation	他ユーザのトラッカの操作

一部のアクセス権限には依存関係があるものが存在し、それらを設定場合には前提となるアクセス権限が必要となります。本コマンドでアクセス権限を設定する場合、前提となるアクセス権限が明示的に指定されていなくても、自動的に設定されます。

権限文字列	前提となる(同時に付与される)アクセス権限の権限文字列
own.jnw.write	own.jnw.read
own.jnw.execute	own.trk.read own.trk.operation
own.trk.operation	own.trk.read
own.jnw.write	own.jnw.read
other.jnw.execute	other.trk.read other.trk.operation
other.trk.operation	other.trk.read

権限文字列にallを使うことで、複数のアクセス権限を同時に付与できます。

権限文字列	付与されるアクセス権限の権限文字列
all	own.jnw.read own.jnw.write own.jnw.execute own.trk.read own.trk.operation other.jnw.read other.jnw.write other.jnw.execute other.trk.read other.trk.operation
own.all	own.jnw.read

権限文字列	付与されるアクセス権限の権限文字列
	own.jnw.write own.jnw.execute own.trk.read own.trk.operation
own.jnw.all	own.jnw.read own.jnw.write own.jnw.execute own.trk.read own.trk.operation 依存関係があるため、own.trkのアクセス権限も同時に付与されます。
own.trk.all	own.trk.read own.trk.operation
other.all	other.jnw.read other.jnw.write other.jnw.execute other.trk.read other.trk.operation
other.jnw.all	other.jnw.read other.jnw.write other.jnw.execute other.trk.read other.trk.operation 依存関係があるため、other.trkのアクセス権限も同時に付与されます。
other.trk.all	other.trk.read other.trk.operation

権限文字列に空文字「""」を指定した場合、権限グループの全てのアクセス権限を無効にします。

■実行例

以下にUNIX版の実行例を示します。

```
# /usr/lib/nqs/gui/bin/jc_permcfg creategrp 新規権限グループ
```

```
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg setperm 新規権限グループ own.all,other.trk.all
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 新規権限グループ
新規権限グループ
  Member

  Authority
    own.jnw.read         on
    own.jnw.write        on
    own.jnw.execute      on
    own.trk.read         on
    own.trk.operation    on
    other.jnw.read       off
    other.jnw.write      off
    other.jnw.execute    off
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root
# /usr/lib/nqs/gui/bin/jc_permcfg setperm 新規権限グループ ""
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 新規権限グループ
新規権限グループ
  Member

  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       off
    other.jnw.write      off
    other.jnw.execute    off
    other.trk.read       off
    other.trk.operation  off
  Permitted User
```

■主要メッセージ

本サブコマンド固有のメッセージを説明します。その他のメッセージは「3.35.3　コマンド共通の主要メッセージ」を参照してください。

メッセージ	内容
Successfully set the permission.	アクセス権限の変更に成功しました。
ERROR: Invaild permission mode.	不正な権限文字列が指定されました。
WARN: The permission mode has no change.	変更前と変更後のアクセス権限に違いはありません。
ERROR: Can't specify the group name to be a built-in group.	デフォルトの権限グループのアクセス権限は変更できません。

3.35.1.6. ユーザの所属する権限グループを変更（assignuser）

■コマンドラインインタフェース

```
jc_permcfg assignuser $groupname $user1 [ $user2 $user3 ... $userN ]
```

■概要

ユーザの所属する権限グループを変更します。

■オプション

$groupname

ユーザを所属させる権限グループ名を指定します。

$user1 [$user2 $user3 ... $userN]

$groupnameで指定した権限グループに所属させるユーザ名を1個以上指定します。

一度に指定できるユーザの最大数は100です。

JobCenter管理者の所属権限グループを変更することはできません。

JobCenter管理者グループに一般ユーザを所属させることはできません。

■実行例

以下にUNIX版の実行例を示します。

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g @NormalUser
@NormalUser
  Member
    jnwmonitor,user01,user02
  Authority
    own.jnw.read          on
    own.jnw.write         on
    own.jnw.execute       on
    own.trk.read          on
    own.trk.operation     on
    other.jnw.read        off
    other.jnw.write       off
    other.jnw.execute     off
    other.trk.read        off
    other.trk.operation   off
  Permitted User

# /usr/lib/nqs/gui/bin/jc_permcfg assignuser 他ユーザ監視グループ jnwmonitor
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read          off
    own.jnw.write         off
    own.jnw.execute       off
    own.trk.read          off
    own.trk.operation     off
    other.jnw.read        on
    other.jnw.write       on
```

```
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root,user01,user02
# /usr/lib/nqs/gui/bin/jc_permcfg assignuser @JnwDeveloper user01 user02
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g @JnwDeveloper
@JnwDeveloper
  Member
    user01,user02
  Authority
    own.jnw.read         on
    own.jnw.write        on
    own.jnw.execute      on
    own.trk.read         on
    own.trk.operation    on
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root,user01,user02
```

■主要メッセージ

本サブコマンド固有のメッセージを説明します。その他のメッセージは「3.35.3　コマンド共通の主要メッセージ」を参照してください。

メッセージ	内容
Successfully set the permission.	所属する権限グループの変更に成功しました。
ERROR: Can't change the JobCenter Manager group's user.	所属先の権限グループにJobCenter管理者グループを指定することはできません。
WARN: The user already exists(username).	ユーザは既に指定された権限グループに所属しています。
ERROR: Can't modify the JobCenter manager(username) to group(groupname).	JobCenter管理者の所属する権限グループを変更することはできません。

3.35.1.7. 権限グループに他ユーザへのアクセス許可を追加（addaccuser）

■コマンドラインインタフェース

```
jc_permcfg addaccuser $groupname $user1 [ $user2 $user3 ... $userN ]

jc_permcfg addaccuser $groupname -a
```

■概要

権限グループに他ユーザへのアクセス許可を追加します。

■オプション

$groupname

アクセス許可を追加する権限グループ名を指定します。

$user1 [$user2 $user3 ... $userN]

$groupnameで指定した権限グループからアクセス許可するユーザ名を1個以上指定します。

一度に指定できるユーザの最大数は100です。

-a

$groupnameで指定した権限グループのアクセス許可ユーザに全ユーザを追加します。

JobCenter管理者グループのアクセス許可ユーザを変更することはできません。

他ユーザに対する権限を一切持たない権限グループに対しては、アクセス許可ユーザの変更は行えません。そのため、あらかじめ対象の権限グループに対して、他ユーザに対するいずれかの権限が設定されている必要があります。

■実行例

以下にUNIX版の実行例を示します。

- 指定したユーザへのアクセス許可を追加

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User

# /usr/lib/nqs/gui/bin/jc_permcfg addaccuser 他ユーザ監視グループ nsumsmgr
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
```

```
    nsumsmgr
```

- すべてのユーザへのアクセス許可を追加

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    nsumsmgr
# /usr/lib/nqs/gui/bin/jc_permcfg addaccuser 他ユーザ監視グループ -a
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root,user01,user02
```

■主要メッセージ

本サブコマンド固有のメッセージを説明します。その他のメッセージは「3.35.3　コマンド共通の主要メッセージ」を参照してください。

メッセージ	内容
Successfully set the permission.	アクセス許可の追加に成功しました。
WARN: Duplicate username(username).	引数に同じユーザ名を複数回指定しています。
WARN: The user already exists(username).	指定されたユーザは既にアクセス許可されています。
ERROR: No permission.	権限グループに他ユーザに対するアクセス権限がないため、アクセス許可の追加ができません。

3.35.1.8. 権限グループから他ユーザへのアクセス許可を削除（delaccuser）

■コマンドラインインタフェース

```
jc_permcfg delaccuser $groupname $user1 [ $user2 $user3 ... $userN ]

jc_permcfg delaccuser $groupname -a
```

■概要

権限グループから他ユーザへのアクセス許可を削除します。

■オプション

$groupname

アクセス許可を削除する権限グループ名を指定します。

$user1 [$user2 $user3 ... $userN]

$groupnameで指定した権限グループからアクセス許可を削除するユーザ名を1個以上指定します。

一度に指定できるユーザの最大数は100です。

-a

$groupnameで指定した権限グループのアクセス許可ユーザから全ユーザを削除します。

JobCenter管理者グループのアクセス許可ユーザを変更することはできません。

他ユーザに対する権限を一切持たない権限グループに対しては、アクセス許可ユーザの変更は行えません。そのため、あらかじめ対象の権限グループに対して、他ユーザに対するいずれかの権限が設定されている必要があります。

■実行例

以下にUNIX版の実行例を示します。

- 指定したユーザへのアクセス許可を削除

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read          off
    own.jnw.write         off
    own.jnw.execute       off
    own.trk.read          off
    own.trk.operation     off
    other.jnw.read        on
    other.jnw.write       on
    other.jnw.execute     on
    other.trk.read        on
    other.trk.operation   on
  Permitted User
    jnwmonitor,nsumsmgr,root,user01,user02
# /usr/lib/nqs/gui/bin/jc_permcfg delaccuser 他ユーザ監視グループ user01 user02
```

```
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root
```

- ユーザへのアクセス許可をすべて削除

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
    jnwmonitor,nsumsmgr,root
# /usr/lib/nqs/gui/bin/jc_permcfg delaccuser 他ユーザ監視グループ -a
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g 他ユーザ監視グループ
他ユーザ監視グループ
  Member
    jnwmonitor
  Authority
    own.jnw.read         off
    own.jnw.write        off
    own.jnw.execute      off
    own.trk.read         off
    own.trk.operation    off
    other.jnw.read       on
    other.jnw.write      on
    other.jnw.execute    on
    other.trk.read       on
    other.trk.operation  on
  Permitted User
```

■主要メッセージ

本サブコマンド固有のメッセージを説明します。その他のメッセージは「3.35.3　コマンド共通の主要メッセージ」を参照してください。

メッセージ	内容
Successfully set the permission.	アクセス許可の削除に成功しました。
WARN: Duplicate username(username).	引数に同じユーザ名を複数回指定しています。
WARN: The user could not be found(username).	指定されたユーザがアクセス許可ユーザの中から見つかりません。
ERROR: No permission.	権限グループに他ユーザに対するアクセス権限がないため、アクセス許可の削除ができません。

3.35.2. 戻り値

戻り値	内容
0	正常終了です。
1	異常終了です。

3.35.3. コマンド共通の主要メッセージ

メッセージ	内容
ERROR: Too few arguments.	コマンドの引数が不足しています。
ERROR: Invalid subcommand(subcommand).	無効なサブコマンドが指定されています。
ERROR: The user name could not be found(username).	指定されたユーザが見つかりません。
ERROR: The specified group could not be found.	指定された権限グループが見つかりません。
ERROR: Group name can't more than 40 bytes.	40バイトを超える権限グループ名が指定されました。
ERROR: Group name has invalid character.	権限グループ名に使用できない文字が指定されました。
ERROR: Invalid group name.	権限グループ名に空文字列が指定されました。
ERROR: Invalid site.	環境変数NQS_SITEの値が不正です。
ERROR: Only manager can execute this command.	一般ユーザで実行できない操作です。

3.35.4. 注意事項

■getpermサブコマンドはJobCenterのユーザ環境を作成したユーザで実行する必要があります。ユーザ環境の作成は、JobCenterへのログイン、またはjc_usercfgのinitサブコマンドで実施してください。

その他のサブコマンドはJobCenter管理者で実行する必要があります。

■クラスタサイトについてコマンドを実行する場合、事前に環境変数NQS_SITEを設定する必要があります。

詳細は 1章 「コマンド一覧」 の注意事項を参照してください。

■Windows版JobCenterで権限グループを指定する場合、大文字小文字は区別されません。

3.35.5. 制限事項

対象のJobCenter　MGにCL/Winで一度もログインされていないユーザについては、パーミッション設定の対象とすることができません。そのため、パーミッション設定を行うには、あらかじめ対象ユーザでJobCenter MGにログインを行ってください。この初回ログインによりユーザ環境がMG上に生成され、対象ユーザに対するパーミッション設定を行うことができるようになります。

または、CL/Winからログインを行わなくとも、jc_usercfgのinitサブコマンドで同等のことを行うことができます。詳細については「3.33 jc_usercfg ユーザ環境の作成、ユーザ環境設定の参照および変更」を参照してください。

4. クラスタ環境のコマンド

クラスタ環境において、クラスタサイトを作成する、データベースを再構築する、あるいはデーモンプロセスの起動および監視、停止を行うなどの機能を持つコマンドです。

4.1. cjcmksite サイトを作成

■UNIX版

```
/usr/lib/nqs/cluster/cjcmksite $site-name $nqs-host-id $db-directory
```

■Windows版

```
%InstallDirectory%\bin\cluster\cjcmksite $site-name $nqs-host-id $db-directory
```

4.1.1. 機能説明

■クラスタサイトを作成します。サイトはパッケージごとに作成されるJobCenterの実行環境です。

■$site-nameには、サイト名を指定します。サイト名には使用するリロケータブルIPアドレスに対応するホスト名（ネットワークオフィシャルなフルドメイン名）を指定します。エイリアス名(別名)は指定できません。

■$nqs-host-idには、JobCenterで使用するマシンIDを指定します。

マシンIDはシステムでユニークなID（数字）をつける必要があります。

■$db-directoryには、JobCenterで使用するデータベースディレクトリ名を指定します。

これはパッケージの共有ディスク上のディレクトリを指定します。指定したディレクトリにJobCenterのデータベースが作成されます。

■本コマンドは、UNIX版ではrootユーザ、Windows版ではJobCenter管理者で実行してください。

4.1.2. オプション

なし

4.1.3. 戻り値

戻り値	内容
0	正常終了です（cjcmksite:complete.のメッセージを表示）。
1	異常終了です。

4.1.4. 注意事項

■クラスタサイトは、リロケータブルIPアドレスを特定してバインドするため、ルーティングの設定によっては、接続できない場合があります。ローカルでサイトを指定したときも同様となります。

■cjcmksiteで誤ったサイト名を指定するなどで構築に失敗した場合は、$db_directory配下を一旦削除した上で再度cjcmksiteで構築しなおしてください。

■追加するサイトのDBパスが既存の他サイトのDBパスと重ならないようにしてください。

4.2. cjcpw デーモンプロセスの起動と監視、停止

■UNIX版

クラスタ環境の場合

```
/usr/lib/nqs/cluster/cjcpw [-v] [-c] $site-name $db-directory

/usr/lib/nqs/cluster/cjcpw -stop $site-name
```

ローカル環境の場合

```
/usr/lib/nqs/cluster/cjcpw [-v] [-c] -local

/usr/lib/nqs/cluster/cjcpw -stop -local
```

■Windows版

クラスタ環境の場合

```
%InstallDirectory%\bin\cluster\cjcpw [ [ -service [-f] ] | [ [-u JobCenter管理者] [-c] ] ] $site-name $db-directory

%InstallDirectory%\bin\cluster\cjcpw -stop $site-name
```

ローカル環境の場合

```
%InstallDirectory%\bin\cluster\cjcpw [ [ -service [-f] ] | [ [-u JobCenter管理者] [-c] ] ] -local

%InstallDirectory%\bin\cluster\cjcpw -stop -local
```

4.2.1. 機能説明

■クラスタ環境の場合

- 第一の形式は$site-nameで指定したクラスタサイトを起動します。起動後はクラスタサイトの常駐プロセスが動作していることを監視します(-cオプションが指定されていない場合)。もし必要な常駐プロセスが何らかの理由により終了した場合、その時点で他の常駐プロセスの終了とクラスタサイトの停止を試みます。
- 第二の形式は、クラスタサイトの停止を行います。クラスタサイトの常駐プロセスを順次停止し、その停止完了を待ち合わせた上で自身を停止します。
- UNIX環境の場合、本コマンドはクラスタサイト起動に必要なシンボリックリンク等が存在しない場合は自動的に形成します。
- 第二の形式は$site-nameで指定したクラスタサイトの停止を行います。クラスタサイトの常駐プロセスを順次停止し、その停止完了を待ち合わせた上で自身を停止します。通常はクラスタ管理ソフトのクラスタパッケージ停止スクリプトに記述して、そこから実行されるようにします。
- $site-nameには、サイト名を指定します。
- $db-directoryには、JobCenterで使用するデータベースディレクトリ名を指定します。

■ローカル環境の場合

- 第一の形式はローカルサイトを起動します。起動後は、ローカルサイトの常駐プロセスが動作していることを監視します。もし必要な常駐プロセスが何らかの理由により終了した場合、その時点で他の常駐プロセスの終了を試みます。

なお、Windows版のcjcpwでローカルサイトの起動を行った場合、JobCenterのサービスであるjcserviceは起動せず、cjcpwがトッププロセスになります。

- 第二の形式は、ローカルサイトの停止を行います。ローカルサイトの常駐プロセスを順次停止し、その停止完了を待ち合わせた上で自身を停止します。

4.2.2. オプション

-c

cjcpwはデーモンプロセスを監視しません。必要なデーモンプロセスを起動して常駐したのを確認した後、自身は終了します。

-v

動作状況を標準エラー出力に表示します。

-stop

サイトの停止を行います。

JobCenter常駐プロセスの終了を待ち合わせるのは、サイト起動時に-cを付けずに実行して常駐しているcjcpwの方です。本オプション付きで実行したcjcpwは常駐プロセスの終了を待ち合わせませんので、注意してください。

-u

JobCenter管理者を指定します。

-service [-f]

サイトをローカルシステムアカウントで、Windowsのサービスとして起動します。

初めてWindowsのサービスとして起動する場合には、以下のメッセージにより、Windowsのサービスへの登録確認が行われます。

Would you like to register this site as a Windows Service?[y/n](default:n)

-fオプションを指定した場合には、Windowsのサービスへの登録確認は行われません。

4.2.3. メンテナンスモード

daemon.confファイル中にmaintenance=ONの記述があった場合、cjcpwから起動されるnqsstartコマンドはデーモン類の起動をしません。このときcjcpwはデーモンの起動異常を検出しません。

管理者はデーモンを起動しないまま、サイトに対するメンテナンスを行うことができます。

同様にnqsstopによってデーモンを停止した場合もcjcpwは異常を検出しません。この場合にはnqsstartによってデーモンを再起動することで、再びcjcpwによるプロセス監視を有効にすることができます。

4.2.4. 関連ファイル

/usr/lib/nqs/nqsstart

/usr/lib/nqs/nqsstop

/usr/lib/nqs/rc/daemon.conf

/usr/spool/nqs/daemon.conf

4.2.5. 戻り値

戻り値	内容
0	正常終了です。
1	異常終了です。

4.2.6. 注意事項

■cjcpwを-cオプション無しで起動すると、cjcpw自身は実行中のままになり、プロンプトが返ってこなくなります。そのため、クラスタサイトまたはローカルサイトを停止したい場合は別の端末からcjcpw –stopを実行して下さい。ただし、このcjcpw -stopコマンド自身は、実行中のままになっている起動時のcjcpwとは非同期に処理を行います。

■NQSのキューで実行中のジョブリクエストについてもプロセス停止や状態保存などの処理を行うため、元々の-cオプション無しで起動したcjcpwプロセスはクラスタサイト全体の終了を待ち合わせる時間が長くかかる場合があります。

4.3. cjcls コマンドを実行したマシン上で稼動しているサイトの一覧を表示

■UNIX版

```
/usr/lib/nqs/cluster/cjcls [-a|-p]
```

■Windows版

```
%InstallDirectory%\bin\cluster\cjcls [-a|-l|-p]
```

4.3.1. 機能説明

■UNIX版のコマンド実行結果の例と説明を以下に示します。

[UNIX版の実行結果例]

```
----------------------------------------------------------------------
 SITE-NAME                 DAEMON-PID    DB-PATH(LINK)
----------------------------------------------------------------------
 testsite                  18616         /mnt/jobdb
                                         (/usr/spool/nqs/0AC0120A)
 othersite                 18672         /mnt/other-jobdb
                                         (/usr/spool/nqs/0AC0120B)
----------------------------------------------------------------------
```

SITE-NAMEの列にサイト名、DAEMON-PIDの列にnqsdaemonのプロセスIDが表示されます(プロセスが起動していない場合はSHUTと表示されます)。　DB-PATH(LINK)の列にはデータベースディレクトリと、データベースディレクトリへのシンボリックリンク(括弧内のパス)が2行で表示されます。

このコマンドは任意のユーザで実行可能です。

■Windows版のコマンド実行結果の例と説明を以下に示します。

[Windows版の実行結果例]

```
-------------------------------------------------------------------------------
 SITE-NAME                      PID    DB-PATH                        STATUS
-------------------------------------------------------------------------------
 testsite                       4284   D:\testsite                    RUN
 othersite                      4356   E:\othersite                   RUN
-------------------------------------------------------------------------------
```

SITE-NAMEの列にサイト名、PIDの列にcjcpwのプロセスID(プロセスが起動していない場合はSHUTと表示されます)、DB-PATHの列にデータベースディレクトリ、STATUSの列にサイトの状態が表示されます。ただしオプションでローカルサイトの情報を表示した場合、ローカルサイトをサービスとして起動していると、ローカルサイトのPIDにはjcserviceのプロセスIDが表示されます。

STATUSの列に表示されるステータスとサイトの状態は以下の通りです。

ステータス	サイトの状態
RUN	サイトが起動しています。
RUN(SERV)	ローカルサイトがサービスとして起動しています。
NW	非監視モード(cjcpwで-cオプションで起動した場合)でサイトが起動しています。
SHUT	サイトが停止しています。

BROKEN	サイトの必要な常駐プロセスで起動していないものがあります(起動中の可能性もあります)。
DISCON	データベースにアクセスできない状態です。

このコマンドは任意のユーザで実行可能です。

4.3.2. オプション

-a

ローカルサイトも含めてプロセス監視の状況を表示します。UNIX版の場合はこのオプションを指定すると実行結果にCJCPWの列が追加され、監視状況(ON/OFF)が表示されます。

-l

本オプションはWindows版のみ有効です。

ローカルサイトも含めてプロセス監視の状況をロング形式で表示します。ロング形式では途中で途切れてしまうような長い項目も全て表示します。

-p

ローカルサイトも含めて起動中のサイト名、プロセス名、PIDを表示します。Windows版の場合、-pと-Pは同じ出力です。

[-pの実行結果例(UNIX版)]

```
-----------------------------------------------------------------------
 SITE-NAME                 PROCESS-NAME                 PID
-----------------------------------------------------------------------
 (localsite)               jcdbs                        6341
                           logdaemon                    6351
                           nqsdaemon                    6352
                           netdaemon                    6354
                           jnwcaster                    6358
                           sclaunchd                    6364
                           jnwengine                    6362
                           jnwlauncher                  6372
                           comagent                     6371
-----------------------------------------------------------------------
 clustersite               jcdbs                        6438
                           logdaemon                    6448
                           nqsdaemon                    6449
                           netdaemon                    6451
                           jnwcaster                    6455
                           sclaunchd                    6461
                           jnwengine                    6459
                           jnwlauncher                  6472
                           comagent                     6468
-----------------------------------------------------------------------
```

[-pの実行結果例(Windows版)]

```
---------------------------------------------------------------------------
 SITE-NAME                      PROCESS-NAME                   PID
---------------------------------------------------------------------------
 (local)                        cjc                            1412
```

```
                                comagent                        3104
                                jcdbs                           1960
                                jnwengine                       3052
                                jnwlauncher                     3062
                                logdaemon                       2444
                                netdaemon                       2840
                                nqsdaemon                       2404
                                sclaunchd                       3092
-----------------------------------------------------------------------------
 ClusterSite                    cjc                             6684
                                comagent                        724
                                jcdbs                           936
                                jnwengine                       1444
                                jnwlauncher                     4062
                                logdaemon                       5400
                                netdaemon                       4960
                                nqsdaemon                       5788
                                sclaunchd                       5924
-----------------------------------------------------------------------------
```

Windows版の-pの実行結果において"cjc"で表示されているプロセス名はWindows内で"jcservice(サービス起動時)"や"cjcpw(cjcpwコマンド起動時)"のイメージ名（プロセス名）で管理されており、そのイメージ名に紐付いたPIDが表示されます。

4.3.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

4.3.4. 注意事項

■cjcmksiteコマンドを用いて不要なサイトを構築した場合、cjclsに不要なサイトの表示が出力されます。UNIX版の場合、この状態はDB-PATH(LINK)の列に表示されているシンボリックリンクを削除することで解消します。

■本コマンドを実行しても、そのノードマシンで一度もクラスタサイトを実行していない場合は、何も情報を出力しません。

4.4. nqspath データベースパス名を表示

■UNIX版

```
/usr/lib/nqs/nqspath [$site-name]
```

■Windows版

（無し）

4.4.1. 機能説明

■そのサイトが使用しているデータベースパス名（標準スプールパス名）を表示します。（ただし改行コードは付与されません）

■$site-nameには、サイト名を指定します。存在しないサイト名を指定した場合は異常終了します。

■$site-nameを省略した場合、環境変数NQS_SITEの値が使用されます。環境変数NQS_SITEが設定されていない場合、ローカルサイトのデータベースのパス名が表示されます。

■任意のユーザで実行できます。

4.4.2. オプション

なし

4.4.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

4.5. nqsportkpr JobCenter関連サービスのポートを一時的にバインド

■UNIX版

```
/usr/lib/nqs/nqsportkpr [-d] [-s service1,service2...] [-i IP1,IP2...] [$hostname]
```

■Windows版

(無し)

4.5.1. 機能説明

■IP1,IP2...に指定したIPアドレス、或いは$hostnameから取得するIPアドレスでservice1,service2...に指定したJobCenter関連サービスのポートを一時的にバインドして、他のプロセスがバインドされたポートを使用しないよう、一時的に占有します。JobCenter関連サービスのポートは/etc/servicesに記述されているポートを使用します（デフォルト nqs：607/tcp jccombase：611/tcp）。

(実行例)

ローカルサイト	/usr/lib/nqs/nqsportkpr -s nqs,jccombase localhost
クラスタサイト	/usr/lib/nqs/nqsportkpr -i 192.168.32.34 testsite1
バインド停止	/usr/lib/nqs/nqsportkpr -d testsite1

■nqsportkprによるポートのバインドは、JobCenterのデーモン（netdaemon）起動時に自動的に解除されます。

■本コマンドはrootユーザのみ実行可能です。

4.5.2. オプション

-d

指定したアドレスに対するnqsportkprによるバインドを停止します。

-s service1,service2...

指定したサービスのポートに対するバインドを実施します。最大6個までです。省略した場合、nqsとjccombaseサービスのポートに対するバインドとなります。

-i IP1,IP2...

指定したIPアドレスに対するバインドを実施します。最大6個までです。省略した場合、$hostnameから取得するIPアドレスに対するバインドとなります。

$hostname

指定したサイトに対するバインドを実施します。ローカルサイトの場合、localhostで指定、または省略してください。クラスタサイトの場合、必ずクラスタサイト名を指定してください。

■-iのオプションのみを指定した場合は、ローカルサイトに対して、-iのオプションで指定したIPアドレスに対するバインドとなります。

■$hostnameのオプションのみを指定した場合は、$hostnameで指定したサイト名に対して、$hostnameから取得するIPアドレスに対するバインドとなります。

■-iのオプションと$hostnameのオプションを両方指定した場合は、$hostnameで指定したサイト名に対して、-iのオプションで指定したIPアドレスに対するバインドとなります。

■-iのオプションと$hostnameのオプションを両方省略した場合は、ローカルサイトのINADDR_ANYアドレスに対するバインドとなります。

4.5.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

4.5.4. 注意事項

■JobCenterをインストールすると自動的にOSのrcスクリプト(nqs.pre)として本コマンドを実行するようにセットアップされますが、自動的にセットアップされるのはローカルサイト分のみとなります。

■クラスタサイト起動前に本コマンドでサービスのポートを占有したい場合は、別途クラスタパッケージのスクリプト内で、リロケータブルIPアドレスが有効になった直後に本コマンドを実行しておく必要があります。

5. SAP ERPのコマンド

JobCenterからSAP ERPシステムへの連携を可能とするERP Optionを利用する際に、使用するコマンドです。

5.1. sapclient SAP ERPと連携

■UNIX版

```
/usr/lib/nqs/sap/sapclient $command [パラメータ...]
```

■Windows版

```
%InstallDirectory%\bin\sapclient $command [パラメータ...]
```

5.1.1. 機能説明

■jnwengineが、ERPジョブ部品を実行、監視するときに使用します。また、ユーザがERPへのログインを確認する場合にも使用します。SAP ERPが記録するXMIログのオーディットレベルの設定を行う場合にも使用します。

$commandには、ユーザが直接使用するサブコマンドを指定します。ユーザが直接使用するサブコマンドは次のとおりです。

■サブコマンド

conntest destination

SAP ERPへの接続テストを行います。

destinationには、設定ファイルdestconf.fに定義されている接続先パラメータセット名を指定します。

destconf.f と saprfc.iniに必要なパラメータを設定してから実行してください。

auditlevel level

オーディットレベルを設定します。

levelには、0～3の値を指定します。

5.1.2. オプション

なし

5.1.3. 戻り値

contestが成功した場合、"connected successful"と表示します。

エラーの場合はその旨を表示します。

5.1.4. 関連ファイル

/usr/lib/nqs/sap/destconf.f
/usr/spool/nqs/gui/<ユーザ名>/sapopt.f
/usr/spool/nqs/saprfc.ini

クラスタ環境の場合は /usr/spool の部分を<クラスタDBパス>と読み替えてください。

5.2. sapcmd SAP ERP接続パラメータを生成

■UNIX版

```
/usr/lib/nqs/sap/sapcmd -ms

/usr/lib/nqs/sap/sapcmd -mp -U $sap_user -P $password
```

■Windows版

```
%InstallDirectory%\bin\sapcmd -ms

%InstallDirectory%\bin\sapcmd -mp -U $sap_user -P $password
```

5.2.1. 機能説明

■第一の形式は、接続パラメータファイルのサンプルファイルを作成します。

■第二の形式は、パラメータファイルに記述する、暗号化したSAPユーザのパスワードを生成します。

5.2.2. オプション

-ms

接続パラメータファイルのサンプルファイルを作成します。

-mp -U $sap_user -P $password

パラメータファイルに記述する、暗号化したSAPユーザのパスワードを生成します。

パスワードは次の形式で生成されます。

$sap_user： ユーザ名

$password： パスワード

5.2.3. 戻り値

第二の形式は、暗号化したSAPユーザのパスワードを表示します。

5.2.4. 関連ファイル

destconf.f.sapmle ... connopt.fのサンプル

6. 設定確認コマンド

JobCenterはネットワークの設定やユーザのデータディレクトリの各種設定に問題があると、起動や動作に支障をきたすことがあります。たとえば、システムのネットワーク設定の変更時やデータディレクトリの移行時などに、設定変更のミスや漏れがあった場合、JobCenterが正しく動作しない場合があります。

設定確認コマンドはJobCenterの実行環境が変動した際、現在の環境に問題がないかどうかを検査します。

6.1. jc_check JobCenterの設定環境を確認

■UNIX版

```
/usr/lib/nqs/check/jc_check [-v] [-l [-b] [-u $user] [$server]]
```

■Windows版

```
%InstallDirectory%\bin\check\jc_check\jc_check [-v] [-l [-b] [-u $user] [$server]]
```

6.1.1. 機能説明

■JobCenterの設定環境の確認を行います。

■ネットワークの設定状況、ユーザのデータディレクトリの状況などを検査し、JobCenterが動作する環境として問題がないかどうかを確認します。

■$serverにはホスト名を指定します。指定されたホスト名に対して検査を実行し、結果を出力します。本オプションは-lが指定された場合にのみ有効です。

6.1.2. オプション

-v

Verbose modeで実行します。

-l

すべてのユーザ、すべてのホスト名（サイト名）に対して、すべての検査を実行し、結果を出力します。

-b

本オプションは-lが指定された場合にのみ有効です。

ジョブネットワーク定義ファイルの確認とキューの詳細情報の取得が省略されます。キュー情報は一覧情報のみが取得されます。

「-l」オプション指定時には全ユーザの定義ファイルを検査するため、定義ファイル数が多い場合には処理に非常に時間がかかる場合があります。

この処理時間を短縮したい場合には本オプションを指定してください。

-u $user

本オプションは-lが指定された場合にのみ有効です。

データディレクトリの検査部分に関して、$userに指定されたユーザのデータディレクトリに対して検査を実行し、結果を出力します（デフォルトではJobCenterに登録されているすべてのユーザのデータディレクトリについて検査を行います）。

ネットワーク設定の検査部分には影響を及ぼしません。

6.1.3. 注意事項

■本コマンドはクラスタサイト利用の際でも、必ず環境変数NQS_SITEやNQS_SITEDBが設定されていない状態で実行してください。

NQS_SITEやNQS_SITEDBを設定したまま実行すると、正常に情報が採取できません。

■jc_checkコマンドはUNIX版ではroot、Windows版では管理者権限で実行してください。

コマンドを実行すると次の項目を検査し、[OK]/[NG]で結果を表示します。

■resolv.defのフォーマットチェックの結果(Windows版のみの機能、UNIX版ではresolv.defを定義してもチェックを行いません)

■ネットワークの設定状況（ホスト名またはサイト名とIPアドレスの名前解決の状況）

■JobCenterの各ユーザのデータディレクトリの設定状況（ディレクトリおよびファイルのオーナ名の確認、重要パラメータの設定状況の確認）

■各ホスト（サイト）上でのマシングループの登録状況

■各ホスト（サイト）に登録されているキューの一覧および詳細情報

■各ホスト（サイト）のNQS全体に関係するグローバルパラメータの設定値

[NG]であった場合、その原因もあわせて表示されます（Windows版とUNIX版では検査項目、出力が若干異なります）。

なお、ユーザ数やジョブネットワーク数が非常に多い場合、実行終了まで時間がかかることがあります。

■-lオプション指定時は、環境によっては非常に多くの情報が端末に出力されます。その場合は、出力結果をファイルにリダイレクトして参照してください。

■-bオプションは、ユーザの定義情報が極めて多く、処理に非常に時間がかかる場合にのみ指定するようにしてください。-bオプションによる情報採取の省略によって、ユーザの定義情報やキューの詳細情報は省略されます。

ただしネットワーク設定や環境設定などのJobCenterシステムやプロセスなどの障害の一次解析に必要な情報は取得されます。基本的にはすべての情報を採取していただくのが望ましいので、極力-bオプションを指定せずに情報採取を行ってください。

■クラスタ環境で1台のマシンに対して複数のホスト名（サイト名）が定義されている場合、$serverを設定してもネットワーク設定の検査部分には影響を及ぼしません。

6.1.4. 実行結果例（Windows版）

Windows版での実行結果の例を示します。

この例では、次の2点において問題が発生していることが分かります。

［実行結果例］

```
#### jc_check command output ####
#### 2013/01/07 16:27:30 REV. *** #####
Host:hostA
-------------------------------------------------------------------
resolv.def's format:            [OK]

resolv.def:
The resolv.def file is empty.

Windows (OS):
hostA.example.com       (192.168.1.144) [OK]

Windows (Wrapper):
hostA.example.com       (192.168.1.144) [OK]
```

```
-----------------------------------------------------------------
Check local site:
hostA.example.com       hostA.example.com[OK]
-----------------------------------------------------------------
Users(ALL)
-----------------------------------------------------------------
nsagent               (1001)    PASSWORD:[NG](NO PASSWORD)注1
Administrator         (0)       PASSWORD:[OK]
                                PRIVILEGE:[NG]注2
                                        SeBatchLogonRight

localsite:
        nsagent(1001):
                        JNW:[--]
                        SCH:[--]
        Administrator(0):
                        JNW:[--]
                        SCH:[--]
-----------------------------------------------------------------
NQS_site=hostA.example.com
-----------------------------------------------------------------
 (ここには “depend_tool” コマンドの実行結果が出力されます)
-----------------------------------------------------------------
NQS_site=hostA.example.com
-----------------------------------------------------------------
NMAP LIST
-----------------------------------------------------------------
 (ここには “nmapmgr show state” コマンドの実行結果が出力されます)
-----------------------------------------------------------------
MACHINE GROUP
-----------------------------------------------------------------
 (ここには “qmgr show machine_group” コマンドの実行結果が出力されます)
-----------------------------------------------------------------
QUEUES AND REQUESTS
-----------------------------------------------------------------
 (ここには “qmgr show long q” コマンドの実行結果が出力されます)
-----------------------------------------------------------------
PARAMETERS
-----------------------------------------------------------------
 (ここには “qmgr show parameter” コマンドの実行結果が出力されます)
-----------------------------------------------------------------
#### OK=4, NG=2 ####
#### normal end of list ####
```

注1　"nsagent"ユーザでパスワードが設定されてないか無効になっている(CL/Win未接続、またはパスワード失効)

注2　"Administrator"ユーザ(uid=0で表示されるJobCenter管理者)で権限(SeBatchLogonRight)の不足が発生している

6.2. depend_tool JobCenter定義の整合性を確認

■UNIX版

```
/usr/lib/nqs/check/depend_tool [-r] [-u $user]
```

■Windows版

```
%InstallDirectory%\bin\check\depend_tool\depend_tool [-r] [-u $user]
```

6.2.1. 機能説明

■JobCenterが管理している定義のファイルと、その関連情報を格納した内部DB上のデータの整合性確認を行います。

■ジョブネットワーク定義、スケジュール定義、起動トリガ定義、監視対象テキストログ、カレンダ定義、カスタムジョブ定義を検査し、JobCenterが動作する環境として問題がないかどうかを確認します。

■本コマンドはUNIX版ではroot、Windows版では管理者権限で実行してください。

6.2.2. オプション

-u $user

ユーザ定義データの検査部分に関して、$userに指定されたユーザに対して検査を実行し、結果を出力します（デフォルトではJobCenterに登録されているすべてのユーザのデータディレクトリについて検査を行います）。

-r

定義修復モードで実行します。

定義の依存関係および、定義ファイルとDB間のデータ整合性の確認を行い、データの不整合が存在する場合は、データの修復を選択する事ができます。

本オプションは、障害発生時の対応で指示された場合に使用するようにしてください。

6.2.3. 主要メッセージ

■定義の検査時のメッセージ

メッセージ	内容
Dependencies are valid	依存関係の不整合は検出されませんでした。
Missing subjnw ["JNW_NAME"]	サブJNW部品として参照するJNW "JNW_NAME"の定義ファイルが存在しないか、DBに登録されていません。
Missing calbranch ["SCH_NAME"]	カレンダ分岐部品として参照するスケジュール"SCH_NAME"の定義ファイルが存在しないか、DBに登録されていません。
Missing schedule ["SCH_NAME"]	スケジュール起動を行うスケジュール"SCH_NAME"の定義ファイルが存在しないか、DBに登録されていません。
Missing calendar ["CAL_NAME"]	スケジュールから参照する稼動日カレンダ"CAL_NAME"の定義ファイルが存在しないか、DBに登録されていません。
Missing jobnetwork ["JNW_NAME"]	起動トリガで投入するJNW "JNW_NAME"の定義ファイルが存在しないか、DBに登録されていません。

メッセージ	内容
Missing target_textlog ["TARGET_TEXTLOG_NAME"]	起動トリガの監視対象として参照する監視対象テキストログ "TARGET_TEXTLOG_NAME"の定義ファイルが存在しないか、DBに登録されていません。
Invalid entries are registered in DB(subjnw). ["JNW_NAME"]	サブJNW部品として参照するJNW "JNW_NAME"の定義ファイルは存在しませんが、DBに登録されています。
Invalid entries are registered in DB(calbranch). ["SCH_NAME"]	カレンダ分岐部品として参照するスケジュール "SCH_NAME"の定義ファイルは存在しませんが、DBに登録されています。
Invalid entries are registered in DB(schedule). ["SCH_NAME"]	スケジュール起動を行うスケジュール "SCH_NAME"の定義ファイルは存在しませんが、DBに登録されています。
Invalid entries are registered in DB(calendar). ["CAL_NAME"]	スケジュールから参照する稼働日カレンダ "CAL_NAME"の定義ファイルは存在しませんが、DBに登録されています。
Invalid entries are registered in DB(jobnetwork). ["JNW_NAME"]	起動トリガで投入するJNW "JNW_NAME"の定義ファイルは存在しませんが、DBに登録されています。
Invalid entries are registered in DB(target_textlog). ["TARGET_TEXTLOG_NAME"]	起動トリガの監視対象として参照する監視対象テキストログ "TARGET_TEXTLOG_NAME"の定義ファイルは存在しませんが、DBに登録されています。
Doesn't exist or not registered in DB.	定義ファイルが存在しないか、DBに登録されていません。
Invalid entries are registered in DB.	定義ファイルの存在しない定義が、DBに登録されています。

■定義の修復時のメッセージ

メッセージ	内容
DB was successfully repaired ($OBJ_TYPE["OBJ_NAME"])	DB内のデータの修復に成功しました。 $OBJ_TYPEは修復対象定義、"OBJ_NAME"は修復対象定義名になります。 $OBJ_TYPEに表示される定義は、以下の通りです。 ■ JNW：ジョブネットワーク ■ SCHED：スケジュール ■ TRG：起動トリガ ■ TARGET_TEXTLOG：監視対象テキストログ ■ WKCAL：稼働日カレンダ ■ CUSTOMJOB：カスタムジョブ
failed to repair DB ($OBJ_TYPE["OBJ_NAME"])	DB内のデータの修復に失敗しました。 $OBJ_TYPEは修復対象定義、"OBJ_NAME"は修復対象定義名になります。 $OBJ_TYPEに表示される定義は、以下の通りです。 ■ JNW：ジョブネットワーク

メッセージ	内容
	■ SCHED：スケジュール ■ TRG：起動トリガ ■ TARGET_TEXTLOG：監視対象テキストログ ■ WKCAL：稼働日カレンダ ■ CUSTOMJOB：カスタムジョブ
Createted empty definition ($OBJ_TYPE["OBJ_NAME"])	依存関係のある定義の修復のため、依存関係先の定義を新規作成しました。 $OBJ_TYPEは新規作成した定義、"OBJ_NAME"は新規作成した定義名になります。 $OBJ_TYPEに表示される定義は、以下の通りです。 ■ JNW：ジョブネットワーク ■ SCHED：スケジュール ■ TARGET_TEXTLOG：監視対象テキストログ ■ WKCAL：稼働日カレンダ ■ CUSTOMJOB：カスタムジョブ

6.2.4. 注意事項

■依存関係のある定義の修復により依存関係先の定義が新規作成された場合、新規作成された定義を再度設定する必要があります。

■本コマンドをクラスタサイトを対象に利用する場合は、必ず環境変数NQS_SITEが設定されている状態で実行してください。

6.3. jc_ping 指定したサイトに対する通信確認

■UNIX版

```
/usr/lib/nqs/check/jc_ping [-h] [-f {rd | json}] [-p {ipv4 | ipv6}] [-w $timeout] $destination_host
```

■Windows版

```
%InstallDirectory%\bin\check\jc_ping [-h] [-f {rd | json}] [-p {ipv4 | ipv6}] [-w $timeout]
$destination_host
```

6.3.1. 機能説明

■ローカルサイトもしくはクラスタサイトを送信元のサイトとし、$destination_hostに指定した送信先のサイトのnqs,jccombase,jceventのポートに対して順に通信確認を行います。

■送信元のサイトは、環境変数:NQS_SITEで設定したサイトとなります。(環境変数:NQS_SITEを設定していない場合には、ローカルサイトとなります。)

6.3.2. オプション

-h

ヘルプを表示します。他のオプションと併用した場合、他のオプションは無視されます。

-f

実行結果の出力形式を指定します。以下のいずれかを指定します。

■rd: 可読形式で出力します

■json: JSON形式で出力します

本オプションを指定しない場合、可読形式での出力となります。

それぞれの出力形式のフォーマットについては「6.3.5 出力形式（可読形式／JSON形式）」を参照してください。

-p

通信確認を行うプロトコルを指定します。以下のいずれかを指定します。

■ipv4: IPv4アドレスでの通信確認を行います

■ipv6: IPv6アドレスでの通信確認を行います

本オプションを指定しない場合、IPv4アドレスでの通信確認を行います。

-w $timeout

各ポートに対する接続処理のタイムアウト時間(秒)を指定します。タイムアウト時間（秒）は0から600までの整数で入力してください。

本オプションを指定しない場合、デフォルト値として30秒がタイムアウト時間(秒)となります。

$destination_host

送信先のサイト名を指定します。

6.3.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。

コマンドの戻り値は、通信確認の継続が不可能な場合のみ異常終了となります。

6.3.4. 注意事項

■本コマンドは、送信元のサイトから送信先のサイトに対する通信確認を行うコマンドです。MG-SV間の双方向の通信確認をする場合には、それぞれのサイトで本コマンドを実行する必要があります。

■UNIX環境にてローカルサイトをサイトモードで起動している環境の場合、送信元のサイトがローカルサイトでも環境変数:NQS_SITEを設定して本コマンドを実行する必要があります。

■Windows版に於いて、Winsockの初期化でエラーが発生した場合に限り、オプションによる出力フォーマットの指定に関わらず出力形式がJSON形式となります。

■送信元のサイトのマシン一覧に送信先のサイトが追加されている必要があります。

■送信先のサイトのマシン一覧に送信元のサイトが追加されている必要があります。

■本コマンドはWindows版では、JobCenter管理者またはJobCenterグループに所属しているアカウントで実行してくだい。JobCenter管理者、JobCenterグループに所属しているアカウント以外のアカウントで実行する場合にはcmd.exeの右クリックメニューから「管理者として実行」を選択して起動したコマンドプロンプトで本コマンドを実行してください。

6.3.5. 出力形式（可読形式／JSON形式）

6.3.5.1. 可読形式

可読形式のファーマットは次のようになります。

```
results :
 destination_host = 送信先サイト名
 destination_ip   = 送信先サイトのIPアドレス
 source_host      = 送信元サイト名
 source_ip        = 送信元サイトのIPアドレス

 connect :
  nqs       : [nqsのポートの通信確認結果]
    nqsのポートの通信確認のメッセージ
  jccombase : [jccombaseのポートの通信確認結果]
    jccombaseのポートの通信確認のメッセージ
  jcevent   : [jceventのポートの通信確認結果]
    jceventのポートの通信確認のメッセージ

通信確認結果のメッセージ
```

可読形式の各名称の説明は以下の通りです。

表6.1 可読形式の出力形式

名称	説　明
送信先サイト名	送信先のサイト名が表示されます。

送信先サイトのIPアドレス	送信先のサイトのIPアドレスが表示されます。
送信元サイト名	送信元のサイト名が表示されます。
送信元サイトのIPアドレス	送信元のサイトのIPアドレスが表示されます。 送信元のサイト名の名前解決で複数のIPアドレスが取得された場合には、カンマ区切りでIPアドレスが表示されます。 本コマンドで使用するIPアドレスは先頭に"*"が付与され、常に先頭に表示されます。
nqsのポートの通信確認結果	nqsのポートの通信確認が正常の場合にはOKが、エラーの場合にはNGが表示されます。
nqsのポートの通信確認のメッセージ	エラー時はエラーメッセージが表示されます。(複数行出力される場合有り) wildcard=onの環境で、nqsのポートの通信確認が正常時の場合には、"wildcard=on"が表示されます。 wildcard=offの環境で、nqsのポートの通信確認が正常時の場合には、メッセージは何も表示されません。
jccombaseのポートの通信確認結果	jccombaseのポートの通信確認が正常の場合にはOKが、エラーの場合にはNGが表示されます。
jccombaseのポートの通信確認のメッセージ	エラー時はエラーメッセージが表示されます。(複数行出力される場合有り) 正常時の場合には、メッセージは何も表示されません。
jceventのポートの通信確認結果	jceventのポートの通信確認が正常の場合にはOKが、エラーの場合にはNGが表示されます。
jceventのポートの通信確認のメッセージ	エラー時はエラーメッセージが表示されます。(複数行出力される場合有り) 正常時の場合には、メッセージは何も表示されません。
通信確認結果のメッセージ	エラー時はエラーメッセージが表示されます。(複数行出力される場合有り) 全ての通信確認が正常に行われた場合には、「$destination_host is alive.」が表示されます。

[通信確認がすべて正常時の出力例]

```
results :
 destination_host = hostB
 destination_ip   = 192.168.0.2
 source_host      = hostA
 source_ip        = 192.168.0.1

 connect :
  nqs       : [OK]
  jccombase : [OK]
  jcevent   : [OK]

hostB is alive.
```

[各ポートの通信確認がエラーの場合の出力例]

```
results :
 destination_host = hostB
 destination_ip   = 192.168.0.2
 source_host      = hostA
 source_ip        = 192.168.0.1
```

```
connect :
 nqs       : [NG]
   NQS net daemon is not present at transaction peer. Retry later.
 jccombase : [NG]
   20388:cl.c(98): cl_open_timeout:jcsock_connect_challenge() failure("No connected socket.
 errno=111")
 jcevent   : [NG]
   No connected socket. errno=111

Communication Failure.
```

6.3.5.2. JSON形式

JSON形式のファーマットは次のようになります。

```
{
  "status_code":0,
  "result_message":"hostB is alive."
  "results":{
    "source_host":"hostA",
    "source_ip":"192.168.0.1",
    "destination_host":"hostB",
    "destination_ip":"192.168.0.2",
    "connect":{
      "nqs":{
        "status_code":0,
        "result_message":""
      },
      "jccombase":{
        "status_code":0,
        "result_message":""
      },
      "jcevent":{
        "status_code":0,
        "result_message":""
      }
    }
  }
}
```

JSON形式の各パラメータの説明は以下の通りです。

表6.2 JSON形式のパラメータ名

パラメータ名	説　明
status_code	確認結果が設定されます。 通信確認が正常の場合には0、それ以外は1が設定されます。
result_message	確認結果のメッセージが設定されます。 エラー時はエラーメッセージが設定されます。 全ての通信確認が正常に行われた場合には、「$destination_host is alive.」が設定されます。 wildcard=onの環境で、nqsのポートの通信確認が正常時の場合には、nqsのresult_messageのみ"wildcard=on"が設定されます。

results	通信確認の内容が設定されます。 通信確認前にエラーが発生した場合には空となります。
source_host	送信元のサイト名が設定されます。
source_ip	送信元のサイトのIPアドレスが設定されます。 送信元のサイト名の名前解決で複数のIPアドレスが取得された場合には、カンマ区切りでIPアドレスが設定されます。 本コマンドで使用するIPアドレスは先頭に"*"が付与され、常に先頭に設定されます。
destination_host	送信先のサイト名が設定されます。
destination_ip	送信先のサイトのIPアドレスが設定されます。
connect	各ポートの通信確認の内容が設定されます。
nqs	nqsのポートの通信確認の内容が設定されます。
jccombase	jccombaseのポートの通信確認の内容が設定されます。
jcevent	jceventのポートの通信確認の内容が設定されます。

6.3.6. 主要メッセージ

6.3.6.1. 通信確認結果の主要メッセージ

メッセージ	内容
$destination_host is alive.	全てのポートの通信確認が正常です。
Communication Failure.	一部のポートの通信確認がNGとなっています。NGとなっているポートの通信結果のエラーメッセージを確認してください。
Usage: jc_ping [-f rd | json] [-p ipv4 | ipv6] [-w $timeout] $destination_host	引数が正しくありません。引数が正しいか確認してください。
Invalid format. [rd | json]	-fオプションにrd,json以外が指定されました。-fオプションの指定を正しくしてください。
Invalid address family. [ipv4 | ipv6]	-pオプションにipv4,ipv6以外が指定されました。-pオプションの指定を正しくしてください。
Invalid timeout. [0 - 600]	-wオプションに0から600の数値以外が指定されました。-wオプションの指定を正しくしてください。
Too many arguments.	引数の数が多いです。引数の指定数が正しいか確認してください。
Too few arguments.	引数の数が足りません。引数の指定数が正しいか確認してください。
Failed to get source hostname.	送信元のサイト名の取得に失敗しました。マシン名が正しいか、または環境変数:NQS_SITEに設定したサイト名が正しいか確認してください。
Failed to get source ipaddress.	送信元のサイト名のIPアドレスの取得に失敗しました。サイト名とIPアドレスの名前解決を確認してください。
Failed to get destination ipaddress.	送信先のサイト名のIPアドレスの取得に失敗しました。サイト名とIPアドレスの名前解決を確認してください。
Failed to create process.(errno=エラー番号)	JobCenter管理者またはJobCenterグループに所属しているアカウントで実行されていない可能性があります。またはcmd.exeの右クリックメニューから「管理

メッセージ	内容
	者として実行」を選択して起動したコマンドプロンプトから実行されていない可能性があります。

6.3.6.2. nqsのポートの通信確認の主要メッセージ

メッセージ	内容
wildcard=on	送信元のサイトのdaemon.confに「NQSDAEMON_OPT=-x wildcard=ON」が設定されている環境で通信確認が正常に行われました。
Unable to chdir() to the NQS root directory.(errno=エラー番号)	NQSのデータディレクトリへの移動に失敗しました。送信元のサイト名が正しいか、送信元のサイトのJobCenterサイトデータベースが正しく作成されているか確認してください。
failed to getaddrinfo: node=$hostname service=$servicename (エラー番号情報).	名前解決またはポート番号の取得に失敗しました。OSのエラー番号に対応したエラーを解決してください。
failed to jc_getaddrinfo: node=$hostname service=$servicename ai_family=$family(エラー番号情報)	名前解決またはポート番号の取得に失敗しました。OSのエラー番号に対応したエラーを解決してください。
failed to NSgetaddrinfo: parameter error (service=$servicename errno=エラー番号)	ポート番号の取得に失敗しました。OSのエラー番号に対応したエラーを解決してください。
Unable to get Machine ID.(host=$hostname, ret=マシンID取得関数の返り値)	該当のサイト名のマシンIDの取得に失敗しました。送信元のサイトのマシン一覧に該当のサイト名が登録されているか確認してください。
Machine-id conflict between client and peer at transaction peer. Seek staff support.	送信元のサイトのマシン一覧に登録されている送信先のサイトのマシンIDが、送信先のサイトのマシンIDと一致していません。マシンIDが一致しているか確認してください。
Client machine-id unknown at transaction peer. Seek staff support.	送信先のサイトのマシン一覧に送信元のサイトが登録されていません。送信先のサイトのマシン一覧に送信元のサイトが登録されているか確認してください。
Connect(2) or read(2) timeout at local host. Retry later.	送信先のサイトへのconnect処理またはreadの処理がタイムアウトにより終了しました。送信先のサイト名が正しいか、送信先のサイトのJobCenterが起動しているか、ネットワークが正しいか確認してください。
NQS net daemon is not present at transaction peer. Retry later.	送信先のサイトのNQSのポートに接続ができません。送信先のサイトのJobCenterが起動しているか、ネットワークが正しいか確認してください。
Local network database error at local host. Seek staff support.(エラーの詳細メッセージ)	送信元のサイトのIPアドレスの取得に失敗しました。サイト名とIPアドレスの名前解決を確認してください。
No network port available at local host. Retry later.(エラーの詳細メッセージ)	利用できるネットワークポートがありませんでした。再実行すると解決する場合があります。

6.3.6.3. jccombaseのポートの通信確認の主要メッセージ

メッセージ	内容
get_jccombase_port() failure("エラー番号情報"(エラー番号)	ポート番号の取得に失敗しました。OSのエラー番号に対応したエラーを解決してください。
jcsock_getaddrinfo() failure("failed to getaddrinfo: node=$hostname service=$servicename (エラー番号情報).")	名前解決またはポート番号の取得に失敗しました。OSのエラー番号に対応したエラーを解決してください。

メッセージ	内容
NSgetaddrinfo() failure(failed to NSgetaddrinfo: parameter error (service=$servicename errno=エラー番号)) servername=$servicename	名前解決またはポート番号の取得に失敗しました。OSのエラー番号に対応したエラーを解決してください。
jcsock_connect_challenge() failure("connect timeouted.(timeout = <$timeout>sec)")	送信先のサイトへのconnect処理がタイムアウトにより終了しました。送信先のサイト名が正しいか、送信先のサイトのJobCenterが起動しているか、ネットワークが正しいか確認してください。
skt_ConnectChallenge() failure(connect timeouted.(timeout = <$timeout>sec))	送信先のサイトへのconnect処理がタイムアウトにより終了しました。送信先のサイト名が正しいか、送信先のサイトのJobCenterが起動しているか、ネットワークが正しいか確認してください。
jcsock_connect_challenge() failure("connect error. errno=エラー番号")	送信先のサイトのjccombaseのポートに接続ができません。送信先のサイトのJobCenterが起動しているかネットワークが正しいか確認または、OSのエラー番号に対応したエラーを解決してください。
skt_ConnectChallenge() failure(connect error. errno=エラー番号)	送信先のサイトのjccombaseのポートに接続ができません。送信先のサイトのJobCenterが起動しているかネットワークが正しいか確認または、OSのエラー番号に対応したエラーを解決してください。
skt_Read:read() connection close	応答待ちでconnectionが切断されました。送信先のサイトのJobCenterのバージョンがR12.4以前の可能性があります。

6.3.6.4. jceventのポートの通信確認の主要メッセージ

メッセージ	内容
failed to getaddrinfo: node=$hostname service=$servicename (エラー番号情報).	名前解決またはポート番号の取得に失敗しました。OSのエラー番号に対応したエラーを解決してください。
failed to jc_getaddrinfo: node=$hostname service=$servicename ai_family=$family(エラー番号情報)	名前解決またはポート番号の取得に失敗しました。OSのエラー番号に対応したエラーを解決してください。
failed to NSgetaddrinfo: parameter error (service=$servicename errno=エラー番号)	ポート番号の取得に失敗しました。OSのエラー番号に対応したエラーを解決してください。
connect timeouted.(timeout = <$timeout>sec)	送信先のサイトへのconnect処理がタイムアウトにより終了しました。送信先のサイト名が正しいか、送信先のサイトのJobCenterが起動しているか、ネットワークが正しいか確認してください。
connect error. errno=エラー番号	送信先のサイトのjceventのポートに接続ができません。送信先のサイトのJobCenterが起動しているかネットワークが正しいか確認または、OSのエラー番号に対応したエラーを解決してください。
No connected socket. errno=エラー番号	送信先のサイトのjceventのポートに接続ができません。送信先のサイトのJobCenterが起動しているかネットワークが正しいか確認または、OSのエラー番号に対応したエラーを解決してください。

7. 情報採取コマンド

JobCenterの障害発生時、原因究明に必要な1次情報を漏れなく採取します。

7.1. jc_getinfo JobCenterの障害発生時、原因究明に必要な1次情報を漏れなく採取

■UNIX版

```
/usr/lib/nqs/check/jc_getinfo [-b] [-d $output] [-e]
```

■Windows版

```
%InstallDirectory%\bin\check\jc_getinfo [-b | -p] [-d $output] [-m {0 | 1 | 2 | 3 | 4}] [-e] [-w]
```

7.1.1. 機能説明

■JobCenterの障害発生時、本コマンドを実行することによって、原因究明に必要な情報（各種設定情報、ログ情報）が自動的に採取されます。

■採取された情報は、-dオプションで指定したディレクトリに既定のファイル名で格納されます(-dオプションを指定しない場合は、カレントディレクトリにそれぞれ格納されます)。

■UNIX版では、上記ディレクトリの直下に"jcdata_<MMDDhhmm>_<hostname>.tar.Z"(Linux以外)、"jcdata_<MMDDhhmm>_<hostname>.tar.gz"(Linux)が作成されます。

■Windows版では、上記ディレクトリ直下に作成された"jcdata"ディレクトリの配下に採取された情報が既定のファイル名、ディレクトリ構造に従い格納されます。

7.1.2. オプション

-b

ジョブネットワーク定義ファイルの確認とJobCenterのユーザ定義情報のファイル一覧、キューの詳細情報の取得が省略されます。キュー情報は一覧情報のみが取得されます。

jc_getinfoコマンドは全ユーザの定義ファイルを検査するため、定義ファイル数が多い場合には、処理に非常に時間がかかる場合があります。

この処理時間を短縮したい場合には本オプションを指定してください。

-p

本オプションはWindows版のみ有効です。

ユーザのパスワードチェックをJobCenter管理者として指定したユーザのみ実施します。

本オプションは、障害発生時の対応で指示された場合に使用するようにしてください。

-d $output

採取された情報の格納先を指定します。

-m {0 | 1 | 2 | 3 | 4}

本オプションはWindows版のみ有効です。

採取するWindows OSイベントログの種類を指定するeventModeオプションです。

本オプションを指定しない場合は、システムイベントログとアプリケーションイベントログを採取します。

■0 全イベントログを採取

■1 システムイベントログのみ採取

■2 アプリケーションイベントログのみ採取

■3 セキュリティイベントログのみ採取

■4 システムイベントログとアプリケーションイベントログを採取

-e

定義情報のメタデータやトラッカの実行状態を管理するデータベースの取得が省略されます。

本オプションは、障害発生時の対応で指示された場合に使用するようにしてください。

-w

本オプションはWindows版のみ有効です。

Windowsエラー報告(ワトソンログ)の採取を省略します。本オプションは、サポートセンターの指示による利用以外は、使用しないでください。

7.1.3. 注意事項

■本コマンドはクラスタサイト利用の際でも、必ず環境変数NQS_SITEやNQS_SITEDBが設定されていない状態で実行してください。

NQS_SITEやNQS_SITEDBを設定したまま実行すると、正常に情報が採取できません。

■本コマンドはUNIX版ではrootユーザ、Windows版ではJobCenter管理者で実行してください。

■Windows　Server　2008以降の環境で実行する場合、cmd.exeの右クリックメニューから「管理者として実行」を選択して起動し、コマンドラインから実行してください。

■-bオプションは、ユーザの定義情報が極めて多く、処理に非常に時間がかかる場合にのみ指定するようにしてください。-bオプションによる情報採取の省略によって、ユーザの定義情報やキューの詳細情報は省略されます。

ただし、ネットワーク設定や環境設定などのJobCenterシステムやプロセスなどの障害の一次解析に必要な情報は取得されます。基本的にはすべての情報を採取していただくのが望ましいので、極力-bオプションを指定せずに情報採取を行ってください。

7.1.4. 実行結果例（Windows版）

Windows版での実行結果例は図7.1「Windows版でのjc_getinfoコマンド実行結果例」のようになります。

この例では、-dオプションを指定しなかったため、実行したディレクトリの配下に“jcdata”ディレクトリが作成されています。

採取される情報、作成されるディレクトリとファイルの数および種類はJobCenterの運用状況によっても変化します。

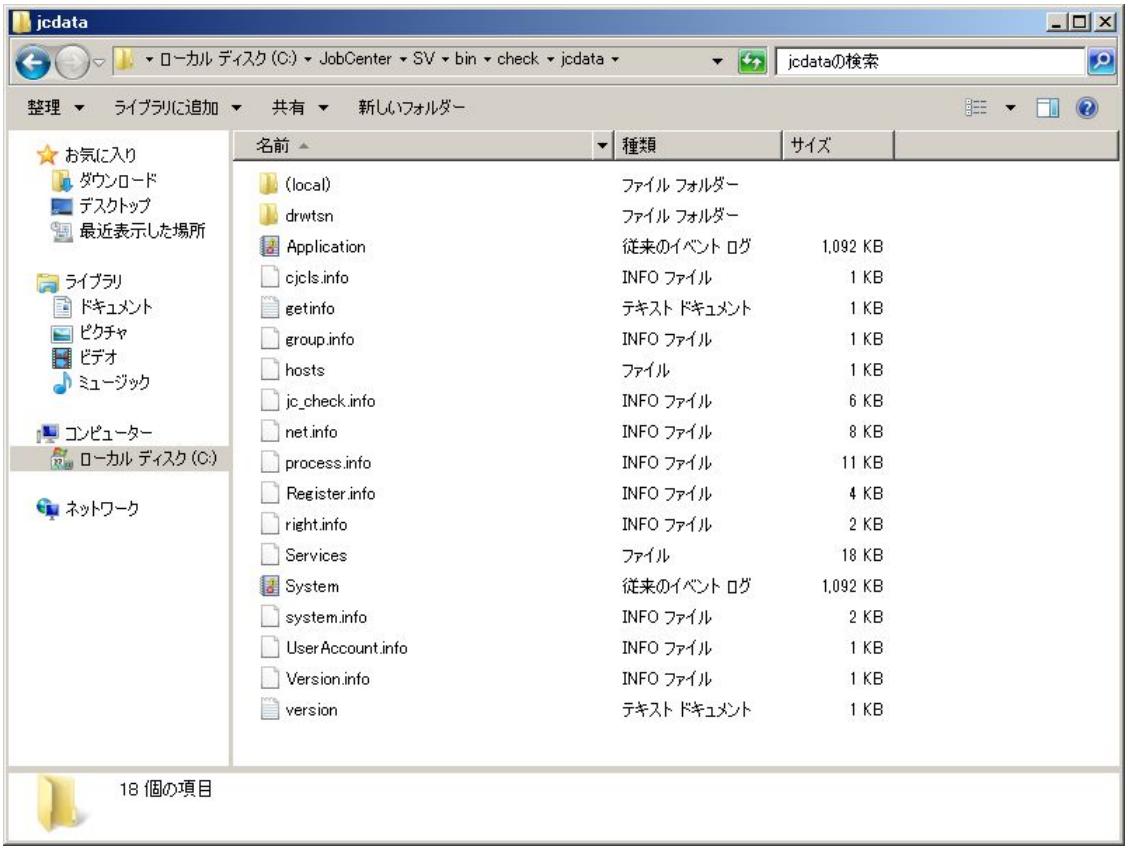

図7.1 Windows版でのjc_getinfoコマンド実行結果例

7.2. clweb_getinfo CL/Webサーバの障害発生時、原因究明に必要な1次情報を漏れなく採取

■UNIX版

```
%InstallDirectory%/script/clweb_getinfo/clweb_getinfo.sh [-d $output]
```

■Windows版

```
%InstallDirectory%\script\clweb_getinfo\clweb_getinfo [-d $output]
```

7.2.1. 機能説明

■CL/Webサーバの障害発生時、本コマンドを実行することによって、原因究明に必要な情報（各種設定情報、ログ情報）が自動的に採取されます。

■採取された情報は、-dオプションで指定したディレクトリに既定のファイル名で格納されます(-dオプションを指定しない場合は、カレントディレクトリに格納されます)。

■上記ディレクトリの直下に"clwebinfo_<YYYYMMDDhhmmss>.zip"が作成されます。

7.2.2. オプション

-d $output

採取された情報の格納先を指定します。

7.2.3. 注意事項

■本コマンドはUNIX版ではrootユーザ、Windows版ではJobCenter管理者で実行してください。

8. Analysis Helper機能のコマンド

JobCenter Analysis Helper機能を利用する際に、使用するコマンドです。

8.1. jdh_trkupdate ローカルのAnalysis Helperのdataフォルダの内容を更新する

```
%InstallDirectory%\bin\jdh_trkupdate [-f] [-u $user] $jpf_file
```

8.1.1. 機能説明

■jdh_trkgetコマンドで収集したJPFファイルから必要なデータを抽出し、Analysis Helperで利用できる形式に変換します。変換後のデータはAnalysis Helperのインストールフォルダ配下のdataフォルダに格納されます。

■$jpf_fileにはjdh_trkgetコマンドで収集したJPFファイルを指定します。

8.1.2. オプション

-f

dataフォルダの内容に関わらず、更新処理を行います。省略した場合はdataフォルダ内にある変換済みのデータをチェックして、更新が必要なものに対してのみ処理を行います。

-u $user

$userに指定されたユーザのデータのみ更新処理を行います。省略した場合は全ユーザのデータの更新処理を行います。

8.1.3. 戻り値

戻り値	内　容
0	正常終了です。
1	異常終了です。エラー内容のメッセージを標準エラーに出力します。

8.1.4. 主要メッセージ

メッセージ	内容
All tracker data update successfully!	データの更新処理が正常終了しました。
Not necessary to be updated.	データ更新の必要はありませんでした。

8.1.5. 実行結果例

本コマンドの実行結果の例を示します。

```
D:\NECJCpkg-jah>bin\jdh_trkupdate trkget_sample.jpf
extracting...
checking...
(1/14) user=root file=trkarc20110730-20110730 updating...
(2/14) user=root file=trkarc20110731-20110731 updating...
(3/14) user=root file=trkarc20110801-20110801 updating...
(4/14) user=root file=trkarc20110802-20110802 updating...
(5/14) user=root file=trkarc20110803-20110803 updating...
(6/14) user=root file=trkarc20110804-20110804 updating...
(7/14) user=root file=trkarc20110805-20110805 updating...
(8/14) user=user1 file=trkarc20110730-20110730 updating...
(9/14) user=user1 file=trkarc20110731-20110731 updating...
(10/14) user=user1 file=trkarc20110801-20110801 updating...
```

```
(11/14) user=user1 file=trkarc20110802-20110802 updating...
(12/14) user=user1 file=trkarc20110803-20110803 updating...
(13/14) user=user1 file=trkarc20110804-20110804 updating...
(14/14) user=user1 file=trkarc20110805-20110805 updating...
All tracker data update successfully!
```

8.1.6. 注意事項

■JPFファイルは中にトラッカのアーカイブファイルを圧縮して保持しており、本コマンドの実行時に最初に解凍作業を行います。そのため、アーカイブファイルのサイズ分だけ一時的にローカルコンピュータのディスクを消費することになります。本コマンド実行時に必要なディスク容量については、<Helper機能利用の手引き>の「4.6 トラッカデータの見積り」を参照してください。

■データの整合性を保つため、Analysis　HelperのExcel上でトラッカデータの読み込み中に本コマンドを実行するとdataフォルダの更新処理は行われずにエラーとなります。本コマンドは、Excel上でトラッカデータの読み込み処理を行っていない状態で実行してください。

発行年月 Jun 2020
NEC Corporation　2020