環境構築ガイド JobCenter

R16.1

■Windows, Windows Server, Microsoft Azure, Microsoft Excel, Internet Explorer および Microsoft Edge 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 およびその他の AWS 商標は、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, iPadOS 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。

■iOS は、Apple Inc. のOS名称です。IOS は、Cisco Systems, 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または登録商標であり、ライセンスに基づき使用されています。

■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。

■Firefox は、Mozilla Foundation の米国およびその他の国における商標または登録商標です。

■UiPath は、UiPath 社の米国およびその他の国における商標です。

■Box, boxロゴは、Box, Inc. の米国およびその他の国における商標または登録商標です。

■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。

本マニュアルでは、製品名およびサービス名を次のように略称表記しています。

略称	製品名・サービス名
Office	Microsoft Office
Excel	Microsoft Excel
Azure	Microsoft Azure
Internet Explorer	Internet Explorer 11
Firefox	Mozilla Firefox
AWS	Amazon Web Services
EC2	Amazon Elastic Compute Cloud
EBS	Amazon Elastic Block Store
S3	Amazon Simple Storage Service
ELB	Elastic Load Balancing
CloudFormation, CF	AWS CloudFormation
CloudWatch, CW	Amazon CloudWatch
RDS	Amazon Relational Database Service
Glue	AWS Glue
Lambda	AWS Lambda
EKS	Amazon Elastic Kubernetes Service
ECS	Amazon Elastic Container Service
STS	AWS Security Token Service
CloudWatch Logs	Amazon CloudWatch Logs
SNS	Amazon Simple Notification Service

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　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。なお、本書内に記載されている画面例と実際の画面とは異なることがありますので注意してください。

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. 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。

3. 用語集

本マニュアルで使用する用語を以下で説明します。

用語	説明
ユーザ権限	JobCenterを利用するユーザに対して付与するJobCenterの権限の総称。
権限グループ	JobCenterを利用するユーザの目的に応じた特定の権限を持つグループ。デフォルトでは、ジョブネットワーク開発者といった権限グループがあります。
LDAP	ディレクトリサービスを制御するためのプロトコル。Lightweight Directory Access Protoclの略称
LDAPサーバ	LDAPで制御できるディレクトリサービスを提供するサーバ。
LDAPサーバの権限グループ	Windows版JobCenterのLDAPサーバ連携の設定後、JobCenter上の権限グループと連携しているLDAPサーバ上のグループのこと。

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 R16.1 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	2022/04/11	新規作成	－	第1版
2	2022/07/15	修正	－	R16.1.1リリースに伴い版改訂
3	2022/09/26	修正	－	R16.1.2リリースに伴い版改訂
4	2022/12/02	修正	－	表記ゆれの修正 「20.1　JobCenterプロセスの監視方法（Windowsの場合）」にjl_notifierの説明を追記
5	2023/10/10	修正	－	R16.1.3リリースに伴い版改訂

目次

はじめに iv
1. マニュアルの読み方 v
2. 凡例 vi
3. 用語集 vii
4. 関連マニュアル viii
5. 改版履歴 ix
1. JobCenter環境構築 1
1.1. JobCenterの製品構成 2
1.2. JobCenterの各製品とOSとの対応 3
2. ネットワーク環境構築 4
2.1. JobCenterで使用するTCPポート 5
2.1.1. NQS 6
2.1.2. jccombase（JobCenterの独自プロトコル） 6
2.1.3. jccombase-over-ssl（JobCenterの独自プロトコル） 7
2.1.4. jcevent（JobCenterの独自プロトコル） 7
2.1.5. jnwengine（JobCenterの独自プロトコル・Windows版のみ） 8
2.1.6. jcdbs（Windows版のみ） 8
2.1.7. jcwebserver（Windows版とLinux版のみ） 8
2.1.8. https 8
2.2. ネットワーク接続を構築する際の注意事項 10
2.2.1. UNIX版JobCenterとWindows版JobCenterを接続する場合 10
2.2.2. UNIX版JobCenterとSUPER-UX NQSを接続する場合 10
2.3. Windows環境における名前解決方法 11
2.3.1. Windowsがサポートする機能による名前解決 11
2.3.2. resolv.defを利用した名前解決 11
2.4. UNIX環境における名前解決方法 14
2.4.1. OSがサポートする機能による名前解決 14
2.4.2. resolv.defを利用した名前解決 14
2.5. 複数NIC使用時のJobCenterの設定について 16
3. JobCenterネットワーク環境構築 17
3.1. ネットワーク上にある他マシンのマシンIDを登録する 18
3.1.1. 標準リモートマシン構成とマシングループのメリット・デメリット 18
3.1.2. 標準リモートマシン構成 19
3.1.3. マシングループ構成（JobCenter CL/Winからの登録） 21
3.1.4. マシングループ構成（コマンドラインからの登録） 23
3.2. ユーザの関連付けを行う（ユーザマッピング） 24
3.2.1. CL/Win（ビューア）からユーザマッピングを行う 24
3.2.2. コマンドラインからユーザマッピングを行う 26
3.3. JobCenterネットワーク環境を構築する際の注意事項 28
3.3.1. 管理可能台数 28
3.3.2. マシングループ構成 28
3.3.3. UNIX版サーバのマシングループへの追加 28
3.3.4. Windows版サーバのジョブ実行ユーザの追加 28
3.3.5. WAN環境での利用 28
3.3.6. 管理者パスワード・ユーザパスワード 29
3.3.7. 登録可能なユーザ数 29
4. キューの作成 30
4.1. デフォルトで作成されるキュー 31
4.2. デフォルトのリクエスト転送順 32
4.3. 自由なキュー構成を構築する 33
4.4. キューの利用可能ユーザを制限する 34
4.5. 管理マシンの詳細を設定する 36
5. JobCenter起動時の設定を変更する 37
5.1. デーモン設定ファイル（daemon.conf）の格納場所 38

5.1.1. Windowsの場合 38
5.1.2. UNIXの場合 38
5.2. デーモン設定ファイルの使用可能パラメータ 39
5.2.1. Windows/UNIX共通 39
5.2.2. Windowsの場合 49
5.2.3. UNIXの場合 50
5.3. 複数IPアドレス使用時のdaemon.conf設定について 53
5.3.1. クラスタ構成／複数NIC使用時のdaemon.conf設定について 53
5.3.2. IPv6アドレスで通信を待ち受ける場合のdaemon.conf設定について 56
5.4. デーモン設定ファイルのフォーマット 58
5.5. JobCenterの起動時ライセンスチェックについて 61
5.6. Windows版JobCenterの動作設定について 62
5.6.1. ジョブ実行時にユーザプロファイルを読み込む 62
5.6.2. ログオン動作のタイプを変更する 62
5.6.3. サイト設定ファイル（site.conf）を作成する 62
5.7. jcwebserverの動作設定について 64
5.7.1. jcwebserver設定ファイル（jcwebserver.conf）の格納場所 64
5.7.2. jcwebserver設定ファイルの使用可能パラメータ 64
6. JobCenter部品パラメータおよび環境の設定変更 68
6.1. JobCenter部品のデフォルトパラメータの設定を定義する 70
6.1.1. ジョブネットワーク 70
6.1.2. 単位ジョブ 70
6.1.3. ERPジョブ 70
6.1.4. BIジョブ 70
6.1.5. PCジョブ 70
6.1.6. WOBSジョブ 70
6.1.7. ダイアログ 71
6.1.8. イベント送信 71
6.1.9. イベント受信 71
6.1.10. 既定値をロード 71
6.2. ユーザ環境の設定を変更する 72
6.3. トラッカ表示の設定を変更する 75
6.4. アーカイブの設定を変更する 80
6.5. システム環境の設定を変更する 86
6.6. 色の設定を変更する 89
6.7. 操作・実行ログ 91
6.8. SMTPサーバ 93
7. 稼働日カレンダの設定 94
7.1. 稼働日カレンダを設定する 95
7.2. 稼働日カレンダを配布する 100
8. 異なる言語間における接続設定 101
9. 日本語環境での文字コード変換 103
9.1. 文字コード変換の設定の必要性を判断する 104
9.1.1. 文字コード変換の設定の必要性を判断する(UNICODE環境混在なし) 104
9.1.2. 文字コード変換の設定の必要性を判断する(UNICODE環境混在あり) 104
9.2. 文字コード変換を設定する 106
9.2.1. UNIX版JobCenterの文字コード変換を設定する 106
9.2.2. Windows版JobCenterの文字コード変換を設定する 107
10. ユーザ権限（パーミッション設定） 109
10.1. アクセス権限の種類 110
10.2. デフォルトの権限グループ 112
10.3. 権限グループを設定する 114
10.4. 権限グループの設定例 115
10.4.1. シナリオ1 115
10.4.2. シナリオ2 116
10.5. 権限設定の注意事項 119

10.5.1. 権限グループに所属させるユーザやアクセス先のユーザ 119
10.5.2. スケジュールによる投入やコマンドによる操作 119
10.5.3. ログインユーザで行う必要がある操作 119
11. ディレクトリサービス連携によるユーザー管理 120
11.1. LDAPサーバ連携(Windows) 121
11.1.1. LDAPサーバ連携の設定 122
11.1.2. LDAPサーバ連携の設定例 124
11.1.3. LDAPサーバ連携の注意事項 128
11.1.4. LDAPサーバ連携の制限事項 129
11.2. LDAPサーバ連携（Linux) 130
11.2.1. ディレクトリサービス連携の動作概要 130
11.2.2. ディレクトリサービス連携の設定 130
11.2.3. JobCenter ログインユーザとしてのLDAP連携設定 133
11.2.4. JobCenter ジョブ実行ユーザとしてのLDAP連携設定 133
11.2.5. 注意事項、制限事項 134
12. イベント連携 135
12.1. イベント連携の概要 136
12.2. UNIX版のイベント連携 138
12.2.1. イベントログファイルへの出力方法 138
12.2.2. イベント設定ファイル 138
12.2.3. イベント定義ファイル 140
12.2.4. イベント一覧 143
12.3. Windows版のイベント連携 149
12.3.1. イベントログファイルへの出力方法 149
12.3.2. ESMPRO/統合ビューア 151
12.3.3. Micro Focus Operations Manager software 173
12.3.4. イベント定義ファイル 176
12.3.5. イベントの種類 177
13. サーバの環境設定(Windows版) 181
13.1. サーバの環境設定の起動 182
13.2. バージョン情報の確認 183
13.3. サイトの設定 184
13.3.1. サイトのプロパティ 185
13.3.2. サイトの起動と停止 189
13.3.3. ジョブの実行設定 191
13.4. ユーザの設定 194
13.4.1. ユーザのプロパティ 195
13.4.2. ユーザの追加 197
13.4.3. ユーザの削除 198
13.5. プロパティの設定 200
13.5.1. 総合設定のプロパティ 200
14. 環境移行 202
14.1. 環境移行の種類 203
14.2. 異なるマシンへユーザ定義データを移行する 205
14.2.1. 定義データのダウンロード 206
14.2.2. 定義データ内のホスト名変更（移行先ホスト名が異なる場合） 206
14.2.3. 定義データのアップロード 208
14.2.4. 補足:アップロード・ダウンロード機能がないバージョンからのデータ移行について 209
14.3. ホスト名を変更する 210
14.3.1. ホスト名の変更 211
14.3.2. マシンIDの変更作業 214
14.4. ユーザのUIDを変更する 218
14.4.1. 事前確認作業 218
14.4.2. ユーザのUID変更作業 219
14.5. 本番環境での確認作業 222
15. ジョブ実行時の環境変数の取り扱い 223

15.1. UNIX版JobCenterの環境変数 224
15.1.1. JobCenter MG側の環境変数 224
15.1.2. JobCenter SV側の環境変数 224
15.1.3. 環境変数の設定方法 225
15.1.4. MGとSVとのLANGが異なる場合の注意事項 228
15.1.5. 環境変数TZに関する注意事項(Linux、AIX版) 229
15.2. Windows版JobCenterの環境変数 231
15.2.1. JobCenter MG側の環境変数 231
15.2.2. JobCenter SV側の環境変数 231
15.2.3. 環境変数の設定方法 233
15.3. ジョブ投入時に独自に設定される環境変数 238
15.3.1. UNIX版 238
15.3.2. Windows版 239
15.4. 環境変数「NQS_SITE」、「NQS_SITEDB」の設定が必要なモジュール 241
16. 日本以外のタイムゾーンで利用する 242
16.1. JobCenterセットアップ後に必要な設定 243
16.2. カレンダへのタイムゾーン設定機能 244
16.2.1. 動作イメージ 244
16.2.2. カレンダへのタイムゾーン設定機能を有効にする 245
16.2.3. カレンダにタイムゾーンの設定を行う 246
16.2.4. スケジュールにタイムゾーンの設定を行ったカレンダを指定する 247
16.2.5. ジョブネットワークや部品オブジェクトへの影響 247
16.2.6. 夏時間の設定を行う場合の動作 248
16.2.7. タイムゾーンに応じた監視を行う 250
16.2.8. 注意事項 251
17. JobCenter環境のバックアップ・復元 252
17.1. バックアップ・復元に関する諸注意 253
17.1.1. 復元時の影響 253
17.1.2. 運用上の注意 253
17.2. JobCenterの構成情報をバックアップ・復元する 254
17.2.1. 構成情報のバックアップ・復元機能 254
17.2.2. バックアップ・復元対象の構成情報 258
17.2.3. 注意事項 261
17.3. イメージバックアップを利用したJobCenter環境のバックアップ・復元 262
17.3.1. イメージバックアップの作成 262
17.3.2. イメージバックアップからの復元 262
17.4. 復元後の整合性確認 264
17.4.1. MG-SV構成のMG(リクエスト投入元サーバー)を復元する場合 264
17.4.2. MG-SV構成のSV(リクエスト投入先サーバー)を復元する場合 266
17.4.3. イベント送信部品、イベント受信部品による連携を行っている環境を復元する場合 267
18. jcresの使用方法 268
18.1. 起動方法 269
18.2. 停止方法 270
18.3. 設定ファイル 271
18.4. ブラウザからMG-SV間の疎通確認を行う 272
18.4.1. 使用方法 272
18.4.2. トラブルシュート 274
18.5. 自動起動・停止の設定方法 278
18.5.1. Red Hat Enterprise Linux 6 278
18.5.2. Red Hat Enterprise Linux 7 279
18.5.3. Windows Server 2012 R2 282
19. WebAPI機能 292
19.1. ライセンスについて 293
19.2. WebAPI導入のイメージ 294
19.3. WebAPIのリクエスト内容 295
19.4. WebAPI一覧 296

19.4.1. WebAPI version 2 機能一覧 296
19.5. WebAPIの認証 297
19.5.1. APIアクセスキー認証 297
19.6. WebAPI version 2 298
19.6.1. 利用方法 298
19.6.2. APIアクセスキーリソース 305
19.6.3. ジョブネットワークリソース 310
19.6.4. トラッカリソース 313
19.6.5. WebAPIのエラー一覧 325
20. JobCenterのプロセスを監視する 327
20.1. JobCenterプロセスの監視方法（Windowsの場合） 328
20.2. JobCenterプロセスの監視方法（UNIXの場合） 330
20.2.1. CJCPW起動のOSサービス化 331
21. SystemManager GでJobCenterの障害情報を監視する 334
21.1. 機能概要 335
21.2. 構築手順 336
21.3. 障害情報一覧 346
21.3.1. 監視するメッセージの表の見方 347
21.3.2. キューがSTOP状態 349
21.3.3. MG→SVのTCP通信が失敗する 351
21.3.4. SVのOSまたはJobCenterが停止している 356
21.3.5. SV→MGのTCP通信が失敗する 358
21.3.6. NQSの発信ポートが不足している 360
21.3.7. 環境変数が大きすぎる 364
21.3.8. ユーザマッピングができていない 366
21.3.9. ユーザマッピング先のユーザが存在しない 368
21.3.10. MGにてSVの名前解決が出来ない 370
21.3.11. SVにてMGの名前解決が出来ない 372
21.3.12. MG→SV転送時にリクエストIDの重複が発生した 374
21.3.13. 同時起動可能数を超えた 376
21.3.14. ジョブ実行ユーザに指定したユーザが存在しない 378
21.3.15. 2GBを超えるトラッカのアーカイブに失敗 379
22. SystemManager GでJobCenterのメトリクスを収集・分析する 381
22.1. 機能概要 382
22.2. 構築手順 383
22.2.1. 性能値取得設定ファイル(getcustomperflist.properties)の設定 384
22.2.2. カスタム性能監視の設定 387
22.2.3. 性能グラフの表示 396
22.2.4. 分析／レポート機能 398
22.3. メトリクス一覧 405
22.3.1. 性能値取得設定ファイルのパラメータ 406
22.3.2. 単位ジョブ実行数 409
22.3.3. ジョブネットワーク実行数 410
22.3.4. 投入待ちジョブ部品数 412
22.3.5. ジョブネットワーク開始時間差 413
22.3.6. 未アーカイブトラッカ全体サイズ 414
22.3.7. アーカイブ済みトラッカ全体サイズ 417
22.3.8. 最大トラッカサイズ 421
22.3.9. キュー使用率 424
22.3.10. リクエスト増減数 426
23. システム利用資源 428
23.1. UNIX版JobCenterのプロセスが使用する資源 429
23.1.1. nqsdaemon（リクエスト実行） 430
23.1.2. jnwengine（ジョブネットワーク制御実行） 430
23.1.3. sclaunchd（スケジュール実行） 430
23.1.4. comagent（通信用モジュール） 431

23.1.5. jnwlauncher（起動トリガの各デーモンの管理） 431
23.1.6. jl_submit（起動トリガのジョブネットワーク投入の管理） 431
23.1.7. jl_logmonitor（起動トリガのテキストログ監視実行） 432
23.1.8. jcwebserver（WebAPI通信用モジュール） 432
23.1.9. その他 432
23.2. メモリ使用量概算算出方法（UNIX版） 434
23.2.1. jnwengineが必要とするメモリ使用量 434
23.2.2. jcdbsが必要とするメモリ使用量 436
23.3. DISK使用容量の概算算出方法（UNIX版） 437
23.3.1. ジョブネットワークが必要とするDISK使用容量 437
23.3.2. トラッカアーカイブのDISK使用容量概算算出方法 439
23.3.3. ジョブリクエストのNQS実行時のDISK使用容量概算算出方法 439
23.3.4. jcdbsのDISK使用容量概算算出方法 439
23.4. Windows版JobCenterのプロセスが使用する資源 441
23.4.1. nqsdaemon（リクエスト実行） 442
23.4.2. logdaemon（nqsdaemonとnetdaemonのためのログを記述） 442
23.4.3. netdaemon（リモートから転送されるリクエスト受信処理） 442
23.4.4. jnwengine（ジョブネットワーク制御実行） 442
23.4.5. qwb（NQS中のジョブの状態変化監視とjnwengineへの通知） 443
23.4.6. trkrdr（jnwengineのコマンド待ち合わせとジョブ情報取得） 443
23.4.7. sclaunchd（スケジュール実行） 443
23.4.8. comagent（通信用モジュール） 443
23.4.9. jnwlauncher（起動トリガの各デーモンの管理） 444
23.4.10. jl_submit（起動トリガのジョブネットワーク投入の管理） 444
23.4.11. jl_logmonitor（起動トリガのテキストログ監視実行） 444
23.4.12. jcwebserver（WebAPI通信用モジュール） 444
23.5. メモリ使用量概算算出方法（Windows版） 446
23.5.1. jnwengineが必要とするメモリ使用量 446
23.5.2. jcdbsが必要とするメモリ使用量 448
23.6. DISK使用容量の概算算出方法（Windows版） 450
23.6.1. ジョブネットワークが必要とするDISK使用容量 450
23.6.2. トラッカアーカイブのDISK使用容量概算算出方法 452
23.6.3. ジョブリクエストのNQS実行時のDISK使用容量概算算出方法 452
23.6.4. jcdbsのDISK使用容量概算算出方法 453
23.7. 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 454
23.7.1. Windows版JobCenter MG/SVのアップロード・ダウンロード時のメモリ使用量の概算算出方法 454
23.7.2. UNIX版JobCenter MG/SVのアップロード・ダウンロード時のメモリ使用量の概算算出方法 455
24. OS要塞化 457
25. トラブルシューティング 458
25.1. トラブルシューティングQ&A 459
25.2. エラーログファイルの設定を変更する 462
25.2.1. Windows版 462
25.2.2. UNIX版 466
25.2.3. NQS関連のログ 467
25.2.4. jcwebserver関連のログ 468
25.3. JobCenter MG/SV の障害発生時、原因究明に必要な情報を採取する 470
25.3.1. JobCenter CL/Winで採取する場合 470
25.3.2. jc_getinfoコマンドで採取する場合 475

表の一覧

1.1. JobCenterの製品構成 2
4.1. デフォルトで作成されるキューの初期設定一覧 31
5.1. site.confのパラメータの設定範囲 63
5.2. jcwebserver設定ファイルのパラメーター一覧 64
8.1. 多言語チェックボックスの有無と他言語間での接続の関係 102
9.1. JobCenterネットワーク構成の組み合わせと文字コード変換の設定の必要性 104
9.2. JobCenterネットワーク構成の組み合わせと文字コード変換の設定の必要性 105
10.1. 権限グループで設定できるアクセス権限の一覧 110
10.2. デフォルトで用意されている権限グループとアクセス権限 112
10.3. デフォルトで用意されている権限グループの用途 112
11.1. LDAPサーバ設定項目 123
11.2. JobCenter側の権限グループとLDAPサーバ側の権限グループの対応表 123
11.3. LDAPサーバ連携の設定例 124
11.4. ユーザの権限グループ 125
11.5. LDAPサーバ連携のリトライ設定 127
12.1. イベント連携機能がサポートするイベント連携種別 137
12.2. イベント一覧 144
12.3. [エラー停止時のイベントを送信]の設定と動作 159
12.4. イベント一覧(ソース名「jnwexe」) 160
12.5. イベント一覧(ソース名「NetShepherd」) 173
12.6. イベント名とイベントの意味一覧 177
13.1. サイトの列項目 184
13.2. サイトの状態 184
13.3. ユーザの列項目 194
13.4. パスワードチェックの説明 195
15.1. UNIX版の環境変数一覧 238
15.2. UNIX版の環境変数一覧（ジョブネットワークを投入した場合） 238
15.3. Windows版の環境変数一覧 239
15.4. Windows版の環境変数一覧（ジョブネットワークを投入した場合） 239
17.1. バックアップ対象の構成情報 258
18.1. MGの環境 274
18.2. SVの環境 275
19.1. WebAPI version 2 機能一覧 296
19.2. パラメータフォーマット一覧 298
19.3. リクエスト形式の構成内容 301
19.4. リクエストヘッダ 302
19.5. クエリパラメータ 302
19.6. レスポンスヘッダ 303
19.7. レスポンスボディのパラメータ（単一） 303
19.8. レスポンスボディのパラメータ（複数） 303
19.9. レスポンスボディのパラメータ(失敗時) 304
19.10. リクエストパラメーター一覧 306
19.11. レスポンスボディのパラメーター一覧 307
19.12. リクエストパラメーター一覧 308
19.13. リクエストパラメーター一覧 318
19.14. リクエストパラメーター一覧 320
19.15. リクエストパラメーター一覧 322
19.16. エラー一覧 325
20.1. JobCenter常駐プロセス一覧（Windows） 328
20.2. cjcpwコマンド 330
20.3. JobCenter常駐プロセス一覧（UNIX） 330
21.1. アプリケーションログ監視のパラメータ 338
21.2. 監視対象ログ定義のパラメータ 340
21.3. フィルタ定義のパラメータ 342

21.4. 監視可能な障害情報一覧 346
22.1. カスタム性能監視のパラメータ 389
22.2. 監視項目選択のパラメータ 390
22.3. カウンタダイアログのパラメータ 392
22.4. 閾値設定ダイアログのパラメータ 394
22.5. グラフ設定ダイアログ(性能グラフ)のパラメータ 401
22.6. 収集・分析可能な性能値一覧 405
22.7. 性能値取得設定ファイルの共通パラメータ 406
22.8. 性能値取得設定ファイルの個別パラメータ 406
22.9. 単位ジョブ実行数の性能値取得設定ファイルの設定値 409
22.10. ジョブネットワーク実行数の性能値取得設定ファイルの設定値 410
22.11. 投入待ちジョブ部品数の性能値取得設定ファイルの設定値 412
22.12. ジョブネットワーク開始時間差の性能値取得設定ファイルの設定値 413
22.13. 未アーカイブトラッカ全体サイズの性能値取得設定ファイルの設定値 415
22.14. 未アーカイブトラッカ全体サイズと最大トラッカサイズを同時に取得する場合の設定値 415
22.15. アーカイブ済みトラッカ全体サイズの性能値取得設定ファイルの設定値 418
22.16. 最大トラッカサイズの性能値取得設定ファイルの設定値 422
22.17. 未アーカイブトラッカ全体サイズと最大トラッカサイズを同時に取得する場合の設定値 422
22.18. キュー使用率の値について 424
22.19. キュー使用率の性能値取得設定ファイルの設定値 424
22.20. リクエスト増減数の値について 426
22.21. リクエスト増減数の性能値取得設定ファイルの設定値 426
23.1. 本文中に出てくるカーネルパラメータやシェル制限の意味 429
23.2. カーネルパラメータやシェル制限と計算式一覧 429
23.3. 本文中に出てくるパラメータの意味 441
23.4. カーネルパラメータと計算式一覧 441
23.5. 各定義のアップロード時のメモリ使用量 454
23.6. 各定義のダウンロード時のメモリ使用量 455
23.7. 各定義のアップロード時のメモリ使用量 455
23.8. 各定義のダウンロード時のメモリ使用量 456
25.1. エラーログの名称とファイルパス 463
25.2. 詳細ログの名称とファイルパス 464
25.3. ログファイル名 467
25.4. ログファイル名 469
25.5. [詳細設定]画面の設定 473

1. JobCenter環境構築

JobCenterはマルチプラットフォーム、マルチベンダによるネットワーク分散環境でのジョブの効率的な自動運用を実現するジョブ管理製品です。

JobCenterは、次に説明する製品やOSとの組み合わせでシステムを構築します。

1.1. JobCenterの製品構成

JobCenterの標準的な製品構成は、マネージャ機能（JobCenter MG）、サーバ機能（JobCenter SV）およびビューワ機能（JobCenter CL/Win）です。SAP ERP連携はオプションになります。

表1.1 JobCenterの製品構成

機　　能	名　　称	説　　明
マネージャ機能	JobCenter MG	ジョブ実行環境構築、状態監視を行います。 監視GUIは、JobCenter CL/Winを使用します。
サーバ機能	JobCenter SV	NQSをベースにしたジョブ実行機能を提供します。
ビューワ機能	JobCenter CL/Win	JobCenter MG, JobCenter SVに接続するWindows上のビューワです。
SAP ERP連携機能	JobCenter for ERP Option	SAP ERPシステム、BIシステムへのジョブ投入を行います。

JobCenterの製品構成イメージは図1.1「JobCenterの製品構成イメージ」のとおりです。Windows環境では、JobCenter MG, JobCenter SVおよびJobCenter CL/Winを同一のマシンにインストールして、小規模なシステムも構成できます。

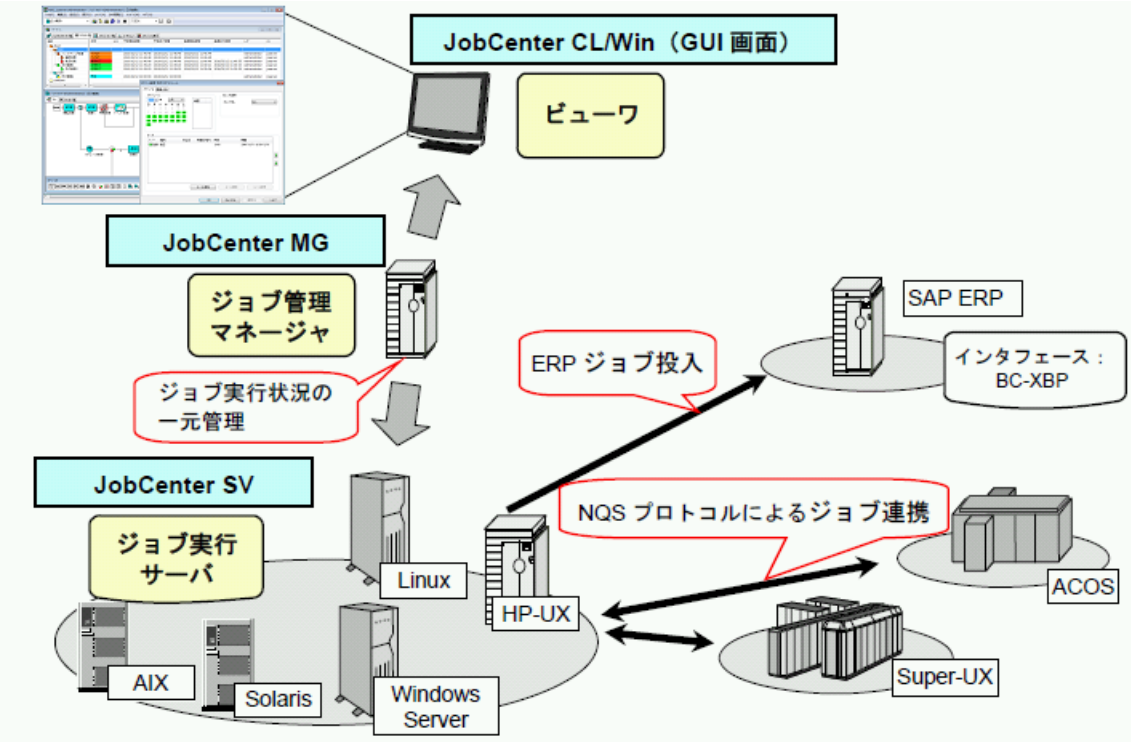

図1.1 JobCenterの製品構成イメージ

1.2. JobCenterの各製品とOSとの対応

JobCenterの各製品と対応OSにつきましては、<リリースメモ>の3章 「動作環境」 をご覧ください。

2. ネットワーク環境構築

JobCenterのネットワークを構築するために、サーバ間のネットワーク環境を構築します。

JobCenterネットワーク環境は、JobCenterのサーバ間通信およびサーバ内通信には、TCPポート番号を指定します。

JobCenterはUNIX版とWindows版、SUPER-UX NQSと接続することが可能です。

JobCenter起動時に、常駐プロセスがそれぞれ下記のポート番号についてbind()を試みます。

もし他のアプリケーション等がすでにそのポート番号を使用していた場合、もしくは何らかの理由によりJobCenterプロセスがすでに常駐していてポート番号が占有されていた場合は、JobCenterの起動に失敗しますので注意してください。

Windowsでrsh.exeを利用する場合の注意事項

JobCenterとrsh.exeを利用する環境の場合、ポートの衝突が発生し、rsh.exeが無応答になる場合があります。その状況を回避するにはrsh.exeをもう一つ実行してください。rsh.exeが降順に空きポートを確認して動作します。

WindowsにおけるNQSの使用ポートについての注意事項

Windows OSの実装上、TIME_WAITで切断待ちになる時間が比較的長めになっています。 頻繁に接続・切断を繰り返すPPが他に存在したり、NQSジョブリクエストを大量に集中的に投入する状況では、Windows　　OSの空きポートが不足して、結果としてJobCenterの処理が遅延する場合があります。　　そのような状況下では、次のようにWindowsのレジストリTcpTimedWaitDelayを追加して、30秒～60秒程度で接続ポートを解放するよう調整してください。

キー名	HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\TcpTimedWaitDelay
値の種類	REG_DWORD
有効な範囲	30 ～ 300 (10進表記・秒)
デフォルト値	0x78 (10進表記では120)

参考URL

■TCP/IP の再送タイムアウトの最大値を変更する方法

https://support.microsoft.com/ja-jp/help/170359

2.1. JobCenterで使用するTCPポート

JobCenterのサーバ間のネットワークのプロトコルには、伝送制御プロトコル／インターネットプロトコル（TCP/IP）を使用します。MGとSV双方でお互いにTCP/IPとホスト名の解決が正常に動作するように設定してください。

JobCenterのサーバ間通信およびサーバ内通信では、JobCenterのセットアップ時に指定したTCPポート番号を複数使用します。なお既定値と異なるポート番号を使いたい場合は、同一システムを構成する全てのMGとSVで同じ番号を使用するように設定してください。

R13.2以前とR14.1以降で設定する方法が異なります。

■R13.2以前

以下のファイルに記載されている情報を修正してください。（インストール時に指定した情報は既に書き込まれています）

```
\WINDOWS\system32\drivers\etc\SERVICES
```

■R14.1以降

以下２つのファイル両方に記載されている情報を修正してください。（インストール時に指定した情報は既に書き込まれています）

```
\WINDOWS\system32\drivers\etc\SERVICES
＜jobcenterインストールフォルダ＞\etc\services
```

jccombase(611)やjccombase-over-ssl(23116)のポートを変更する場合、CL/Winのセットアップ時にポートをそのポートへ変更する必要があります。

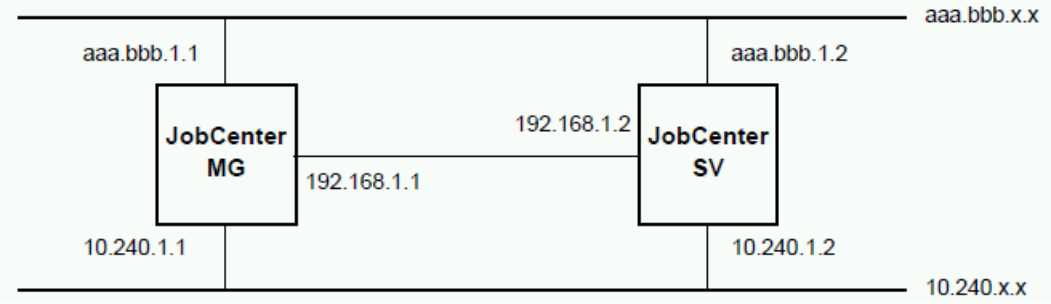

図2.1 IPアドレスの指定例

図の例ではaaa.bbb.1.1～aaa.bbb.1.2はグローバルアドレス、10.240.1.1～10.240.1.2および192.168.1.1, 192.168.1.2はプライベートアドレスです。MGからSVへのジョブ転送とSVからMGへの結果返却は同じネットワークを経由するよう、ネットワークのルーティングテーブルを適切に設定する必要があります。

次に、JobCenter MG/SVおよびJobCenter CL/Winで使用する8種類のプロトコルとTCPポート番号について説明します。FireWall等のフィルタリングルール設定の参考にしてください。

ポート番号の表記	意　味
n/tcp → m/tcp	ソースポートnからデスティネーションポートmについてTCPコネクションを張ります。TCPコネクションは双方向のデータ通信に用いられます。

2.1.1. NQS

MG⇔SV間、SV⇔SV間で、ジョブの制御（単位ジョブリクエストの転送、結果取得）を行う際、使用するプロトコルです。

NQSでは、「ジョブの転送」「結果ファイル転送」「SV⇒MG状態通知」で双方向の通信が行われるため、使用するポートは下記の表の通りになります。

MGからSVにジョブリクエストを転送する場合、およびSVからMGにジョブ実行結果を返却する場合は、データ転送が終了すると直ちにコネクションを切断します。

■NQSプロトコルが使用するポート

JobCenter MG	コネクションを張る方向	JobCenter SV
512～1023/tcp[注1]	→	607/tcp
607/tcp	←[注2]	512～1023/tcp[注1]
512～1023/tcp	→	607/tcp[注3]

注1　これらのポートは、通常「Well-Knownポート」と呼ばれています。NQSではtcpポートのうち512番から1023番で未使用のものをソースポートとして選択して使用します。

IPパケットのsource　IPアドレスについては「5.3.1　クラスタ構成／複数NIC使用時のdaemon.conf設定について」を参照してください。(FireWallでsource　IPアドレスでアクセス制限をかける際には注意してください)

注2　ジョブリクエスト転送および結果の返却とは別に、SV⇒MG状態通知(ジョブ実行状況通知)のコネクションを常時1本維持します。

このコネクションはkeepalive動作を行いませんので、MGとSV間のネットワーク上にルータやFireWallが存在すると、無通信状態を検出したルータが片側のみセッション切断を行い、ハーフオープンセッション状態が発生してTCP/IP通信上の問題が発生する場合があります。それを回避するためには

- MGからSVに定期的に"exit"だけを記述した空ジョブを投入するようスケジュールを設定する(強制的にSV⇒MGの状態通知の通信を行わせる)
- MGのマシングループにSVを参加させる(マシングループ内のSVがMGに状態通知の通信を定期的に行う)

などの設定を行うようにしてください。

なお、自分自身に対してもこのコネクションを常時1本張るように動作します。

注3　マシングループ構成になっているスケジューラマシンに、CL/Winで管理者ユーザで接続すると本セッションが作成されます。

CL/Winでスケジューラに管理者で接続し、SV上のトラッカデータに差分が生じない場合、本セッションで無通信となってしまうため、FWやルータ等で通信がリセットされてしまう可能性があります。ユーザ環境設定画面で自動更新をOFFにする対策や各SV上にダミーのJNW定義を作り、一定時間毎にJNWが投入されるか投入されたJNWのステータスが更新されるようにする等の対策を行うようにしてください。

2.1.2. jccombase（JobCenterの独自プロトコル）

CL/WinからJobCenterを操作する際に使用するプロトコルです。その他Nscl_Submit、Nscl_Refer、jnwsubmitcmd -r、jdh_upload、jdh_download、jnwdelete(Windows版のみ) の各コマンドを実行すると、このプロトコルを使用して目的のリモートホストにコマンドのデータを送信します。

また、MG⇔SV間でキューの制御やマシン一覧の管理などJobCenterの管理操作を行う場合にも使用します。ジョブの実行制御には直接関係しません。

CL/WinからSVへの操作要求～SVからCL/Winへの結果転送のたびに新しいコネクションが張られ、データ転送が終了すると直ちに切断します。

■jccombaseプロトコルが使用するポート

JobCenter CL/Win Nscl_Submit等のリモート実行系コマンド	コネクションを張る方向	JobCenter MG/SV
1024～/tcp[注4]	→	611/tcp

JobCenter MG	コネクションを張る方向	JobCenter SV
1024～/tcp[注4]	→	611/tcp

注4　エフェメラルポートをソースポートとして選択して使用します。エフェメラルポートの空きが無くなると接続できなくなりますのでご注意ください。なおMG/SVからCL/Winに対してコネクションを張ることはありません。

なお、本プロトコルのIPパケットのsource IPとしては、destination IPと通信可能なセグメントに対応したIPアドレスを用います。(クラスタサイトについても同様のため、FireWallでsource　IPアドレスでアクセス制限をかける際には注意してください。詳細は「5.3.1　クラスタ構成／複数NIC使用時のdaemon.conf設定について」の各注意事項を参照してください)

2.1.3. jccombase-over-ssl（JobCenterの独自プロトコル）

CL/WinからJobCenterを操作する際の通信を暗号化したプロトコルです。その他、Nscl_Submit、Nscl_Refer、 の各コマンドを暗号化通信を指定して実行すると、このプロトコルを使用して目的のリモートホストにコマンドのデータを送信します。

CL/WinからSVへの操作要求～SVからCL/Winへの結果転送のたびに新しいコネクションが張られ、データ転送が終了すると直ちに切断します。

サイトに証明書と秘密鍵ファイルが設定されていない場合には、本プロトコルは使用されません。

■jccombase-over-sslプロトコルが使用するポート

JobCenter CL/Win Nscl_Submit等のリモート実行系コマンド	コネクションを張る方向	JobCenter MG/SV
1024～/tcp[注5]	→	23116/tcp

注5　エフェメラルポートをソースポートとして選択して使用します。エフェメラルポートの空きが無くなると接続できなくなりますのでご注意ください。なおMG/SVからCL/Winに対してコネクションを張ることはありません。

2.1.4. jcevent（JobCenterの独自プロトコル）

JobCenterのイベント送受信部品による連携機能が使用するプロトコルです。イベント送信部品からイベント送信が行われるたびに新しいコネクションが張られ、データ転送が終了すると直ちに切断します。

ただし特定の条件の下では、イベント受信側から送信元に対して通知を行うためにコネクションが張られる場合があります。

その他、ファイル待ち合わせ部品で「他のJobCenterサーバ上のファイルを待つ」をONにすると、リモートマシン上のMG/SVにこのプロトコルを使用するコネクションが張られます。このコネクションはリモートファイルを待つ動作が終了するまで維持します。

■jceventプロトコルが使用するポート

JobCenter MG/SV	コネクションを張る方向	JobCenter MG/SV
1024～/tcp[注6]	→	10012/tcp
10012/tcp	←[注7]	1024～/tcp[注6]

注6　エフェメラルポートをソースポートとして選択して使用します。エフェメラルポートの空きが無くなると接続できなくなりますのでご注意ください。

なお、本プロトコルのIPパケットのsource IPとしては、destination IPと通信可能なセグメントに対応したIPアドレスを用います。(クラスタサイトについても同様のため、FireWallでsource　IPアドレスでアクセス制限をかける際には注意してください。詳細は「5.3.1　クラスタ構成／複数NIC使用時のdaemon.conf設定について」の各注意事項を参照してください)

注7　イベント送信側に対して、イベント受信側からコネクションを張って状態通知を行うのは次の場合です。

- イベント送信部品に受信確認ONが設定されていて、イベント受信部品が後から起動されて受信された。
- 送信されたイベントが、後から同じイベントで上書きされた。
- 送信されたイベントにより受信側で保持できるイベント件数を超過して、古い受信イベントが破棄された。(JNWENGINE_OPT=-uが設定されていない場合。詳細は「5.2.1 Windows/UNIX共通」のJNWENGINE_OPTの項目を参照してください)

2.1.5. jnwengine（JobCenterの独自プロトコル・Windows版のみ）

ホスト/サイト内部のプロセス間通信で使用するプロトコルです。ポート番号609/tcpを使用します。

2.1.6. jcdbs（Windows版のみ）

ホスト/サイト内部のプロセス間通信で使用するプロトコルです。エフェメラルポートをソースポートとしてポート番号23131/tcpとの通信を行います。エフェメラルポートの空きが無くなると接続できなくなりますのでご注意ください。

2.1.7. jcwebserver（Windows版とLinux版のみ）

WebブラウザからWebAPIによりJobCenterを操作する際に使用するプロトコルです。

■jcwebserverプロトコルが使用するポート

Webブラウザ	コネクションを張る方向	JobCenter MG/SV
n/tcp[注8]	→	23180/tcp

注8　Webブラウザが使用するポート番号については、Webブラウザのベンダにご確認ください。

2.1.8. https

CL/WebサーバとWebブラウザ間の通信で使用するプロトコルです。

■httpsプロトコルが使用するポート

Webブラウザ	コネクションを張る方向	JobCenter CL/Web
n/tcp[注9]	→	443/tcp

注9　Webブラウザが使用するポート番号については、Webブラウザのベンダにご確認ください。

上記の他に、CL/Webサーバの内部通信のためにエフェメラルポートをソースポートとして選択して使用します。エフェメラルポートの空きが無くなると接続できなくなりますのでご注意ください。

2.2. ネットワーク接続を構築する際の注意事項

UNIX版JobCenterとWindows版JobCenterまたはSUPER-UX NQSとの接続について説明します。

なおJobCenter　CL/Win(GUI画面）からではなくnmapmgrコマンドでマシン登録を行う場合は、次のように設定してください。

■UNIX系マシン上のnmapmgrの設定では、Windows版JobCenterのNQS　TYPEを「necnt」で登録してください。

■Windowsマシン上のnmapmgrの設定では、UNIX版JobCenterのNQS TYPEを「nec」で登録してください。

■OLF/JB-SVやOLF/JB-CLでACOSマシンとNQS連携する場合は、ACOSマシンのNQS　TYPEを「cos」で登録してください。

- nmapmgrコマンドでマシン登録を行う

```
# nmapmgr ↵
NMAPMGR:> add mid <マシンID> <マシン名>
```

- nmapmgrコマンドでJobCenterのNQS TYPEを登録する

```
# nmapmgr ↵
NMAPMGR:> set type <マシンID> cos
```

詳細については<NQS機能利用の手引き>の「6.5.2 リモートマシン定義」、<コマンドリファレンス>の「3.12 nmapmgr ネットワークの構成管理」を参照してください。

2.2.1. UNIX版JobCenterとWindows版JobCenterを接続する場合

次のような制限がありますので注意してください。詳細については<リリースメモ>を参照してください。

■シェルスクリプトの代わりにバッチファイル形式で記述する必要があります。

■日本語処理コード体系が通常のUNIXとは異なります。文字コード変換が必要な場合があります。詳細については9章 「日本語環境での文字コード変換」 を参照してください。

■GUI専用のためqstatなど情報表示系のコマンドで情報を表示できません。

■qsubなどで指定する結果ファイルのパス名において、ドライブ名(A:など）を使用できます。ただし1文字のマシン名は、ドライブ名として解釈します。

2.2.2. UNIX版JobCenterとSUPER-UX NQSを接続する場合

SUPER-UX NQSは、弊社スーパーコンピュータ「SXシリーズ」上のUNIXで使用できるNQSです。機能の詳細な説明はSUPER-UX NQSに付属する「NQS利用の手引」を参照してください。

JobCenterからジョブの投入を行うときに、SUPER-UXの機能を使用するための、いくつかのオプションが指定できます。またqstatなどのコマンドを用いてSUPER-UX上のジョブの状態を調べることができます。詳細については<NQS機能利用の手引き>の4章 「JobCenter ユーザコマンド一覧」 を参照してください。

接続にあたっては、nmapmgrコマンドで指定するマシンタイプを「nec」のまま使用するようにしてください。詳細については<コマンドリファレンス>の「3.12 nmapmgr ネットワークの構成管理」を参照してください。

2.3. Windows環境における名前解決方法

Windowsの問題やネットワーク環境により、名前解決が正しくできずJobCenterが正常に動作しない場合があります。

上記のような問題を解決する方法として、以下の方法があります。

■Windowsがサポートする機能を利用した名前解決

■resolv.defを利用した名前解決

2.3.1. Windowsがサポートする機能による名前解決

Windowsがサポートする以下のような方法で正引き・逆引きともに正しく行えるように設定してください。

■DNSによる名前解決

■hostsファイル(<Windowsシステムディレクトリ>\drivers\etc\hosts)による名前解決

2.3.2. resolv.defを利用した名前解決

Windowsがサポートする機能による名前解決を行なっても問題が解決しない場合、または以下のような状況で名前解決の問題が発生した場合は JobCenterの名前解決機能(resolv.defファイル)を利用してください。

■名前解決が公開用のLAN側のIPアドレス設定で行われる状況で、非公開の内部用LANでJobCenterを運用したい場合など、環境によりDNSやhostsファイルの設定を変更することが困難な場合

■WSFC環境でJobCenterを運用する場合

■複数NIC環境でJobCenterを運用する場合

■IPアドレスからコンピュータ名の逆引きができない環境において、CL/Winの接続、及び画面遷移などの動作が遅い場合

■MG/SVのローカルサイトについては、JobCenterをインストールする際に自動でresolv.defファイルに登録されます。詳細は<インストールガイド>の「2.4.4 Windows版（通常インストール)」を参照してください。

■複数NIC環境の場合は、resolv.defの設定に加えてリモートのJobCenter　SVからのジョブ投入時に投入元のIPアドレスからホスト名の逆引きを行わないようにする必要があります。

詳細については「5.3.1　クラスタ構成／複数NIC使用時のdaemon.conf設定について」を参照してください。

resolv.defファイルの格納場所は、以下になります。

MG/SV	<インストールディレクトリ>\etc\resolv.def
CL/Win	<インストールディレクトリ>\resolv.def

JobCenterが停止している状態で、resolv.defの修正を行ってください。

resolv.defファイルの記述フォーマットは次のとおりです。

```
<SV1-パブリックLANのIPアドレス> <SV1-エイリアス> <SV1-マシン名>
<SV2-パブリックLANのIPアドレス> <SV2-エイリアス> <SV2-マシン名>
```

```
<SV3-パブリックLANのIPアドレス> <SV3-エイリアス> <SV3-マシン名>
 :
 :
<SVn-パブリックLANのIPアドレス> <SVn-エイリアス> <SVn-マシン名>
<リロケータブルIPアドレス> <クラスタサイト-エイリアス> <クラスタサイト-マシン名>
```

■resolv.defファイルに記述するマシン名はエイリアス名ではないマシン名を記述してください。エイリアス名ではないマシン名は以下に記載するようなホスト名またはFQDNとなります。

- Windows環境におけるエイリアス名ではないマシン名

  インストール時に「ホスト名とFQDNの設定」および「マシン一覧への登録名の設定」にて設定したホスト名、またはFQDNとなります。

- UNIX環境におけるエイリアス名ではないマシン名

  セットアップ時に自動的に設定されます。システムでFQDNが設定されている場合、FQDNが優先され、FQDNが設定されていない場合はホスト名となります。

また、クラスタ環境におけるマシン名は、クラスタ用のサイト作成時に設定したサイト名となります。

■<IPアドレス>、<エイリアス>、<マシン名>等、各項目の間は1バイト空白、またはタブで区切ります。

■マシン名がFQDNの場合、FQDNと併せてホスト名をエイリアスとして記載する必要があります。マシン名がホスト名の場合、エイリアスを記載する必要はありません。

例

マシン名がFQDNの場合

```
192.168.0.1 hostA hostA.example.com
```

マシン名がホスト名の場合

```
192.168.0.1 hostA
```

■同じIPアドレスの行を複数記述することはできません。

■ネットワーク上で名前解決できない不正な設定を記述しても、無視されます。必ず名前解決可能な組み合わせで記述してください。

resolv.defファイルはhostsファイルのようにコメントアウトを利用することができません。

NG例

```
192.168.0.1 hostA hostA.example.com #DBサーバ
```

resolv.defファイルが存在する場合、JobCenterSV(/MG)はDNSやhostsファイルよりも優先的にresolv.defファイルの設定内容を参照します。JobCenterSV(/MG)が正引きまたは逆引きしたときにどのマシンでも同じように正しく名前解決を行えるようにresolv.defファイルに設定してください。

以下、複数のLANにマシンが接続している状況で、JobCenterがその一方のLANを利用するようにしたい場合のresolv.defファイルやdaemon.confファイルの設定例を示します。

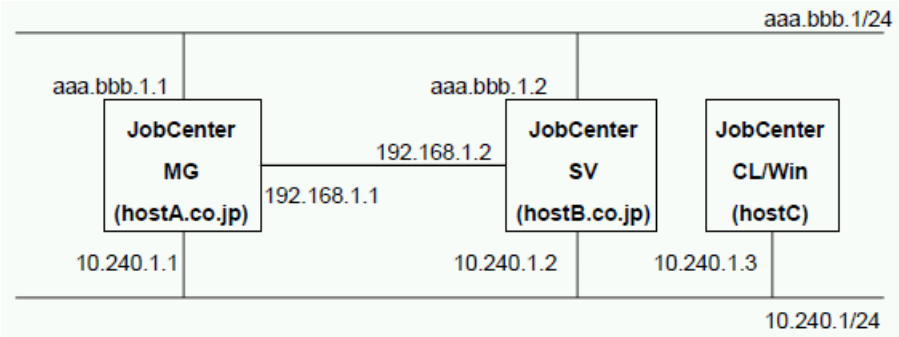

図2.2 IPアドレスの指定例

図中のaaa.bbb.1.1～aaa.bbb.x.xはグローバルアドレスを示します。

10.240.1.1～10.240.x.xおよび192.168.1.1, 192.168.1.2は、プライベートアドレスを示します。

MG、SV各マシンの<インストールディレクトリ>\etc\resolv.def設定例

```
10.240.1.1   hostA   hostA.co.jp
10.240.1.2   hostB   hostB.co.jp
10.240.1.3   hostC
```

MGマシンの<インストールディレクトリ>\etc\daemon.conf設定例

```
ipaddress=10.240.1.1
ipcheck=OFF
```

SVマシンの<インストールディレクトリ>\etc\daemon.conf設定例

```
ipaddress=10.240.1.2
ipcheck=OFF
```

<インストールディレクトリ>はJobCenterのインストールディレクトリです。デフォルトはC:\JobCenter\SV（CL/WinのデフォルトはC:\JobCenter\CLxx.yy　※xx.yyはバージョン番号）になります。

Windows版ではIPv6アドレスのみの環境はサポートしていません。IPv4アドレスも必要となります。詳細については「5.3.2.1 JobCenterのIPv6運用環境の前提条件」を参照してください。

2.4. UNIX環境における名前解決方法

UNIXの問題やネットワーク環境により、名前解決が正しくできずJobCenterが正常に動作しない場合があります。

上記のような問題を解決する方法として、以下の方法があります。

■OSがサポートする機能を利用した名前解決

■resolv.defを利用した名前解決

2.4.1. OSがサポートする機能による名前解決

OSがサポートする以下のような方法で正引き・逆引きともに正しく行えるように設定してください。

■DNSによる名前解決

■hostsファイル(/etc/hosts)による名前解決

■nsswitch.confファイル(/etc/nsswitch.conf)による名前解決の参照順設定

2.4.2. resolv.defを利用した名前解決

OSがサポートする機能による名前解決を行なっても問題が解決しない場合、または名前解決に時間がかかる場合は JobCenterの名前解決機能(resolv.defファイル)を利用してください。

■複数NIC環境の場合は、resolv.defの設定に加えてリモートのJobCenter　SVからのジョブ投入時に投入元のIPアドレスからホスト名の逆引きを行わないようにする必要があります。

詳細については「5.3.1　クラスタ構成／複数NIC使用時のdaemon.conf設定について」を参照してください。

resolv.defファイルの格納場所は、以下になります。

ローカルサイト・クラスタサイト共通	/usr/lib/nqs/rc/resolv.def

resolv.defファイルの記述フォーマットは次のとおりです。

```
<SV1-IPv4アドレス> <SV1-マシン名> <SV1-エイリアス>
<SV2-IPv4アドレス> <SV2-マシン名> <SV2-エイリアス>
<SV3-IPv4アドレス> <SV3-マシン名> <SV3-エイリアス>
 :
 :
<SVn-IPv4アドレス> <SVn-マシン名> <SVn-エイリアス>
<リロケータブルIPv4アドレス> <クラスタサイト-マシン名> <クラスタサイト-エイリアス>
```

■resolv.defファイルに記述するマシン名はエイリアス名ではないマシン名を記述してください。

■resolv.defファイルに記述されたマシン名の大文字と小文字は区別しません。

■<IPv4アドレス>、<マシン名>、<エイリアス>等、各項目の間は1バイト空白、またはタブで区切ります。

■マシン名がFQDNの場合、FQDNと併せてホスト名をエイリアスとして記載する必要があります。マシン名がホスト名の場合、エイリアスを記載する必要はありません。

■同じIPアドレスの行を複数記述することはできません。

■ネットワーク上で名前解決できない不正な設定を記述しても、無視されます。必ず名前解決可能な組み合わせで記述してください。

■クラスタサイトのジョブ実行時にローカルサイトの名前解決を行うため、クラスタサイトのみ利用する環境であっても、resolv.defにシステムのホスト名を登録することを推奨します。

■HP-UXではパッケージ管理コマンド(swinstall、swremove、swlist)はシステムのホスト名の名前解決ができない場合エラーとなるため、以下の操作を行う際は/etc/hostsやDNSで名前解決する必要があります。

・パッケージの操作

・構成情報のバックアップ

・jc_getinfoコマンド

resolv.defファイルが存在する場合、JobCenterSV(/MG)はDNSやhostsファイルよりも優先的にresolv.defファイルの設定内容を参照します。JobCenterSV(/MG)が正引きまたは逆引きしたときにどのマシンでも同じように正しく名前解決を行えるようにresolv.defファイルに設定してください。

Windows版のresolv.defと以下の箇所が異なります。

■IPv6アドレスは指定できません。

■行の途中に「#」を指定すると、以降の文字列はコメントになります。

■1行に記載できる最大長は、8191バイトです。最大長を超える行は無視されます。

■インストール時にresolv.defファイルを格納しません。本機能を利用する場合はお客様自身でファイルを作成する必要があります。ファイルは以下の権限で作成してください。

・アクセス権：0644(-rw-r-r--)

・所有者：root

・所有グループ：bin

■名前解決の際に常にresolv.defを参照します。このため、resolv.defの修正内容はJobCenterを再起動しなくても即座に反映されます。

■マシン名の記載順序が異なります。UNIX版ではIPv4アドレスの次に記載します。

2.5. 複数NIC使用時のJobCenterの設定について

複数NIC環境でJobCenterを使用する場合、JobCenterがバインドするIPアドレスを指定する必要があります。daemon.confを設定することでJobCenterが使用するIPアドレスを指定できます。

> 詳しい設定方法は「5.3.1 クラスタ構成／複数NIC使用時のdaemon.conf設定について」を参照してください。
>
> Windows環境の場合はresolv.defファイルによる名前解決指定が必要になる場合があります。「2.3 Windows環境における名前解決方法」も参照してください。

以下では、複数NIC環境でJobCenterを使用する場合(または、ローカルサイトとクラスタサイトを同一マシン上で同時に使用する場合)の動作仕様と注意点を記載します。

■source IPについて

JobCenterにおいて、TCP/IP通信のsource IPは以下のように割り当てられます。

JobCenterのプロトコル	source IP
NQS(607/tcp)	自サイト名を正引きして得られたIPアドレス[注1]
jccombase(611/tcp)	INADDR_ANY
jcevent(10012/tcp)	INADDR_ANY

[注1]上記はデフォルト設定であり、INADDR_ANYの通信に変更することが可能(後述)

■source IPがINADDR_ANYの場合の動作について

TCP/IPのsource IPには、destination IPと通信可能なセグメントに対応したIPアドレスが用いられます。どのIPアドレスが用いられるかの優先順位はOSにより決定されます。

なお、これは複数NIC環境に限りません。同一マシン上で動作するクラスタサイトとローカルサイトの通信で、source IPがローカルサイトのIPアドレスになる場合があることを示します。

■MG-SVが異なるセグメントに所属している場合の設定について

MG-SVが異なるセグメントに所属している場合、NQSについては自サイト名を正引きして得られたIPアドレスをsource IPとして通信しようとするため、正しく通信できない場合があります。これを解決するには、daemon.confに「NQSDAEMON_OPT=-x wildcard=ON」の設定を行う必要があります。この設定を行った場合、NQSの通信もINADDR_ANYになります。

またこの場合には、destination IP側のMG/SVのdaemon.confに「ipcheck=OFF」の設定も合わせて行ってください。

> daemon.confの設定については「5.2 デーモン設定ファイルの使用可能パラメータ」を参照してください。

■ファイアーウォール等の通信制限について

source IPをINADDR_ANYとすることで異なるセグメント間の連携も可能となりますが、通信可能なIPアドレスが複数ある場合、どのIPアドレスを用いるかはNICの優先順位によって決定されるため、ファイアーウォール等の通信制限を行う場合には注意してください。

特に、クラスタ環境においてローカルサイトとクラスタサイトが同じセグメントに所属している場合、NICの優先順によっては、クラスタサイトの通信がフローティングIPアドレスではなくローカルサイトのIPアドレスから送信される場合があります。このような場合は、ファイアーウォールの例外設定において、ローカルサイトのIPアドレスについて通信を許可するようにしてください。

3. JobCenterネットワーク環境構築

JobCenterネットワークの環境構築により、ネットワーク上の複数のマシンのJobCenterを集中して管理したり、キューやリクエストなどの状態監視、運用操作が行えます。

パイプキューを使ってジョブを転送しますので、ネットワーク上にあるジョブの転送先となる他マシンのマシンIDの定義を行ったあと、ジョブ投入側のユーザとジョブ実行側のユーザの関連付けを行う必要があります。必要となる設定は次のとおりです。

■ネットワーク上にある他マシンのマシンIDを登録する

■ユーザの関連付けを行う（ユーザマッピング）

これらの環境構築はJobCenter管理者で行います。

この設定を行わずにネットワーク経由でジョブの投入を行うと次のようなエラーが表示されます。

```
Client machine-id is unknown at transaction peer;
No account authorization at transaction peer;
```

マシン環境を変更したときは、マシンIDの設定およびユーザマッピングが正しく再設定されているか必ず確認してください。

R14.2以降では、マシン一覧に追加したマシンに対してJobCenterで使用する各ポートへの通信が正常に行えるかを確認するコマンドが実装されました。環境構築後や、リクエストの投入が正常に行われない時などの場合にこのコマンドで、各ポートへの通信が正常に行えるかを確認してください。詳細については、 <コマンドリファレンス>の「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。

3.1. ネットワーク上にある他マシンのマシンIDを登録する

ネットワーク上にあるジョブの転送先となる他マシンのマシンIDを登録します。なお、自マシンのマシンIDはインストール時に登録されています。

ネットワーク上にある他マシンのマシンIDを自マシンに登録する方法には次の3通りがあります。

■標準リモートマシン構成

■マシングループ構成（JobCenter CL/Winからの登録）

■マシングループ構成（コマンドラインからの登録）

3.1.1. 標準リモートマシン構成とマシングループのメリット・デメリット

標準リモートマシン構成は、マシンアイコン追加とユーザマッピングを設定する基本的な構成です。マシングループでは、システム全体を集中管理するためのより便利な機能を利用できます。マシングループについての説明は、 <NQS機能利用の手引き>の「6.7.6 マシングループ/ スケジューラマシン」を参照してください。

標準リモートマシン構成とマシングループには、それぞれ次のようなメリットとデメリットがあります。通常のリモートジョブリクエストの実行においては、設定が容易な標準リモートマシン構成を推奨していますが、次の点に留意して適切な構成を選択してください。

	標準リモートマシン構成	マシングループ構成
設定変更時作業量	小○	大×
ネットワーク負荷	低○	高×
情報一覧性	低×	高○
稼働日カレンダ同期	手動×	自動○
デマンドデリバリ方式による負荷分散	不可×	可○
その他	下記参照	下記参照

■設定変更時作業量

マシングループ構成では、マシングループに参加している各SVにCL/WinでJobCenter管理者アカウントにより接続しても、常にEUIモードでの接続となるため、キューの構成変更やその他の環境設定変更ができなくなります。設定変更するためには一旦マシングループから離脱させる必要があります。

■ネットワーク負荷

標準リモートマシン構成では、マシン間の定常的な通信は発生しないため、ネットワークに負荷をかけません。

一方、マシングループ構成では、マシングループに参加している各SVとの定期的な状態参照の通信が発生し、トラッカ一覧@全マシンでは全てのマシンのトラッカデータの転送を行うため、ジョブ実行状況収集時の負荷が標準リモートマシン構成よりも重くなります。

また、マシングループに所属させるメンバマシンの数は、おおむね8台が推奨値です。10数台以上を管理する場合は、通信量の増大による負荷に留意する必要があります。（必要に応じてマシングループを分割して設定してください）

■情報一覧性

標準リモートマシン構成では、他の各マシンのキューやトラッカを参照する場合は、各マシンアイコンごとに個別にウィンドウを開いて参照する方法のみとなります。

一方、マシングループ構成では、MGマシンの[トラッカ一覧@全マシン]ウィンドウで、グループ傘下の全マシンの全ユーザのトラッカ情報を参照できます。

■稼働日カレンダ同期

標準リモートマシン構成では、他マシンとの稼働日カレンダの同期は、マシンアイコンごとに個別に「稼働日カレンダの更新」を実行する必要があります。

一方、マシングループ構成では、稼働日カレンダをスケジュールマシン上で更新すると、グループ内の各メンバマシンに自動配布されますので、他のマシン上にジョブネットワークを置いて稼働日カレンダ参照スケジュールにより起動する運用の場合のカレンダ管理が一元化されます。

■デマンドデリバリ方式による負荷分散

標準リモートマシン構成では、デマンドデリバリ方式による負荷分散はできません。

一方、マシングループ構成では、複数マシンの複数のキューを指定した、デマンドデリバリ方式による負荷分散環境が構築できます。

■その他

標準リモートマシン構成では、マシンアイコンの追加だけではリモート操作は行えません。最低限、JobCenter管理者アカウント同士のユーザマッピングを手動で設定する必要があります。

一方、マシングループ構成では、マシングループ内のすべてのマシンが起動していることを前提に動作しますので、例えばマシングループからマシンを削除する際に、グループ内の一部のマシンでJobCenterが起動していないと、正常に削除できない場合があります。

3.1.2. 標準リモートマシン構成

標準リモートマシン構成で他マシンのマシンIDを登録する方法は、次のとおりです。

マシン一覧に表示されるマシンアイコンについて、同一マシンが「ホスト名のみ」と「FQDN」の2通りでアイコンが2個表示される場合があります。これはセットアップやマシングループへのマシン追加の際に、FQDNで認識されるマシンについては自動的にホスト名のみの名前を別名として設定するためです。

別名はマネージャフレームのマシン一覧表示で運用上の役割で識別したい場合や、nmapmgrやqmgrサブコマンドにおける利便性向上のために利用することができます。(ただし有効範囲は自マシン(サイト)内だけです。通信上の名前解決には使用できません)

別名が不要な場合は、CL/Winからではなくnmapmgrコマンドで削除できます。(CL/Winからは、別名だけを削除することはできません)

<コマンドリファレンス>の「3.12　nmapmgr　ネットワークの構成管理」の「サブコマンド」の「Delete Name $alias」を参照してください。

■マシン一覧へマシンを追加する

1. 管理者権限のアカウント（UMS　Mode）でログインし、マネージャフレームの［マシン一覧］を表示します。

2. ［マシン一覧］ウィンドウ上で右クリックをしたときのポップアップメニューから［新規作成］を選択するか、メニューバーの［ファイル］－［新規作成］を選択します。

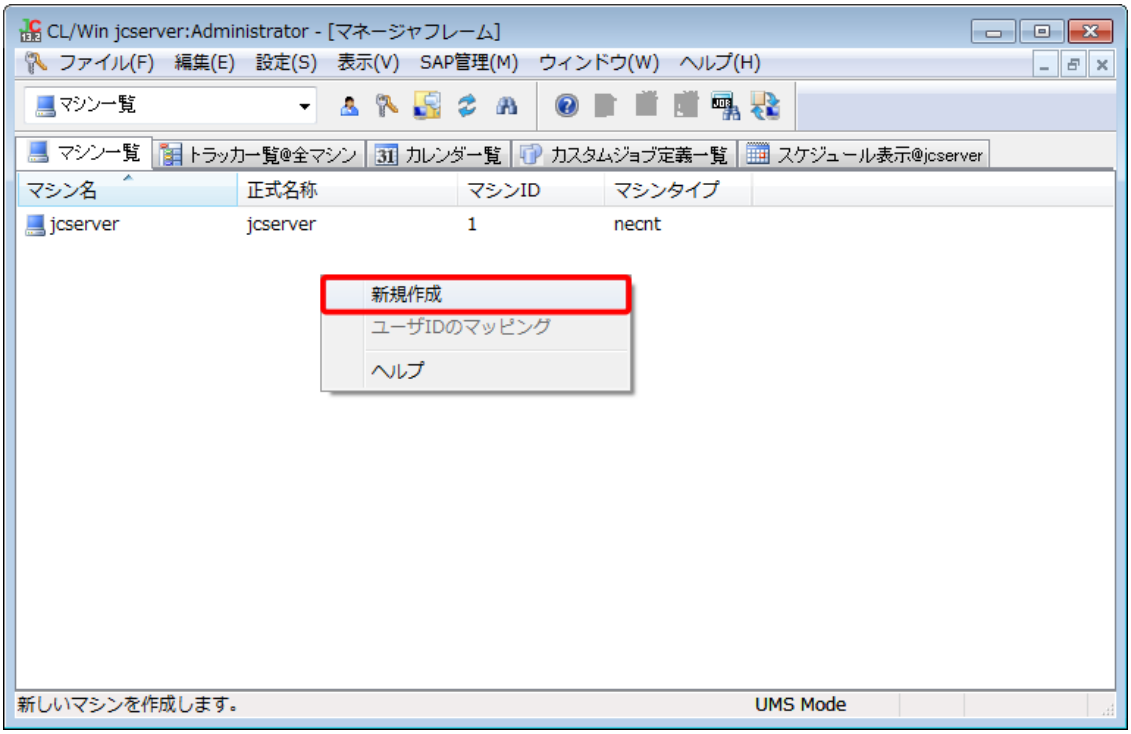

図3.1 マシン一覧へのマシン追加画面例（標準リモートマシン構成機能）

3. ［マシンの追加］ダイアログが表示されますので、追加するマシンのホスト名を入力します。

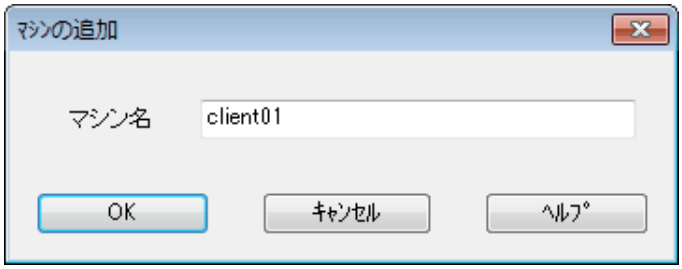

図3.2 ［マシンの追加］ダイアログ画面例

4. 設定後、［OK］ボタンをクリックします。

5. マネージャフレームの［マシン一覧］ウィンドウに追加されたマシンが表示されます。

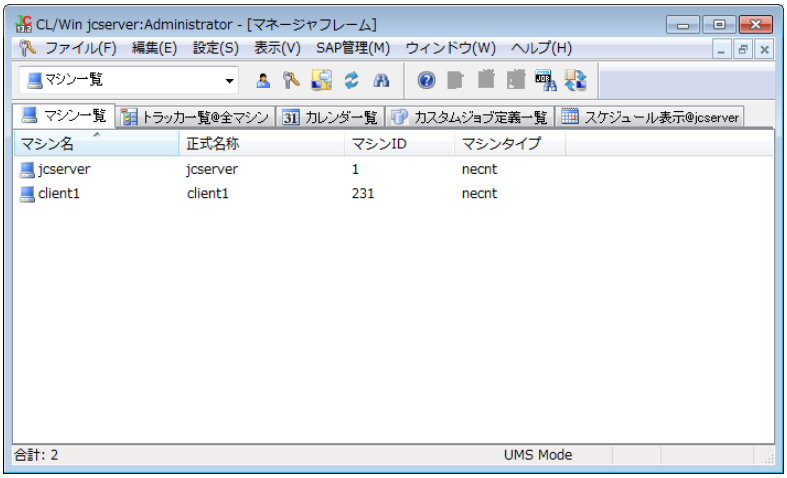

図3.3 マシン追加後の［マシン一覧］画面例（標準リモートマシン構成機能）

■マシン一覧からマシンを削除する

1. マシンを追加するときの操作と同様に、［マシン一覧］ウィンドウを表示します。

2. 削除したいマシンにあわせて右クリックから［削除］を選択するか、あるいは削除したいマシンを選択し、メニューバーから［ファイル］－［削除］を選択します。

3. ［マシン一覧］からマシンが削除されます。マシンが削除されると同時に、そのマシンのすべての別名が削除されます。

マシングループに追加されたマシンをマシン一覧ウィンドウから削除した場合、マシングループからも自動的に削除されますので注意してください。

3.1.3. マシングループ構成（JobCenter CL/Winからの登録）

ネットワーク上にある他マシンをマシングループに追加、削除します。マシン一覧に存在しないマシンをマシングループに追加すると、マシン一覧にもマシンIDが自動的に設定され、JobCenter管理者同士のユーザマッピングも自動的に設定されます。

■マシンをマシングループに追加する場合、そのマシンを複数のマシングループに参加させることはできません。

■すでに他のマシングループに参加しているマシンを指定してマシングループに参加させると、そのマシンは以前のマシングループからはずれてしまいますが、元のマシングループのスケジューラマシンにはその変更が反映されません。そのため、すでにマシングループに参加しているマシンを他のマシングループに参加させたい場合は、既存のスケジューラマシンのマシングループ設定から一旦削除してから、新しいマシングループに追加するよう注意してください。

■マシングループに参加しているマシンをマシン一覧から削除すると、自動的にマシングループからも削除されます。

■マシングループに追加したマシンのホスト名やクラスタサイト名変更を行う場合は、事前にスケジューラマシンのマシングループとマシン一覧から削除してから変更してください。

■マシングループに他マシンを追加する

1. スケジューラマシンにするマシンに管理者権限のアカウント（UMS　Mode）でログインし、メニューバーの［設定］－［マシングループ］を選択します。

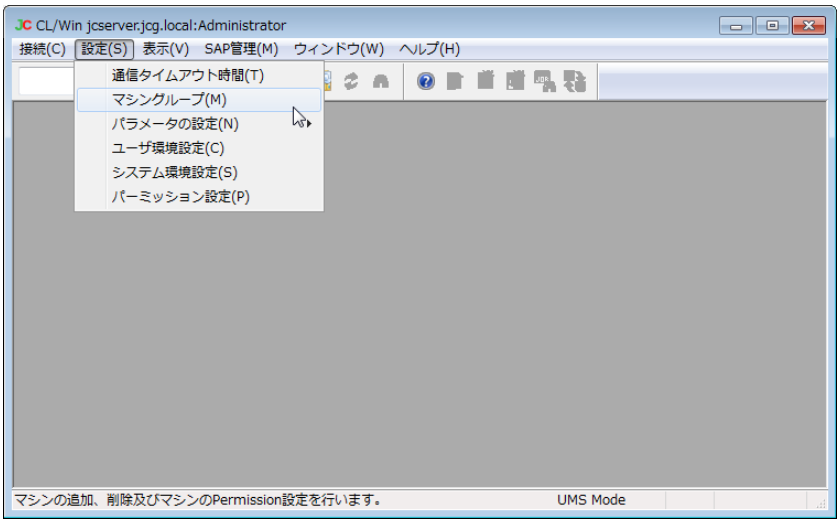

図3.4 メニューバー［設定］－［マシングループ］選択画面例

2. ［マシングループ］ダイアログが開きますので［追加］ボタンをクリックしてください。

図3.5 ［マシングループ］ダイアログ画面例

3. ［マシンの追加］ダイアログが開きますので、マシングループに追加したいマシンの正しいホスト名を入力します。（ホスト名は必ず名前解決の設定で正引きまたは逆引きできるよう設定してください。）

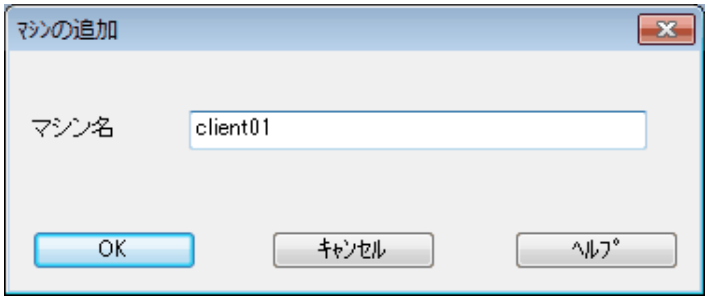

図3.6 ［マシンの追加］ダイアログ画面例

4. 設定後、［OK］ボタンをクリックします。

5. ［マシングループ］ダイアログにマシンが追加されます。

図3.7 マシン追加後の［マシングループダイアログ］画面例

■マシングループから他マシンを削除する

1. マシンを追加するときの操作と同様に、［マシングループ］ダイアログを表示します。
2. 削除したいマシンを選択し、［削除］ボタンをクリックします。
3. ［確認］ウィンドウが開きますので、［OK］ボタンをクリックします。
4. ［マシングループ］ダイアログからマシンが削除されます。
5. マシングループからマシンを削除しても、マシン一覧から削除されてないことを確認します。

マシングループに追加されたマシンをマシン一覧ウィンドウから削除した場合、マシングループからも自動的に削除されますので注意してください。

3.1.4. マシングループ構成（コマンドラインからの登録）

コマンドラインからマシングループを設定する場合は、nmapmgrコマンドとqmgrコマンドを使用して設定します。(特定のマシンのみ指定した追加、削除ではなく、マシングループ再定義となります)

両コマンドの詳細については、<NQS機能利用の手引き>の「6.5.2　リモートマシン定義」、<コマンドリファレンス>の「3.12 nmapmgr ネットワークの構成管理」、<コマンドリファレンス>の「3.13 qmgr 構成管理および運用管理」をそれぞれ参照してください。

1. nmapmgrコマンドでサブコマンドadd mid <マシンID> <マシン名>を実行する

   事前に追加対象マシンをマシン一覧に登録します。この操作はマシングループに追加するすべてのマシンについて行います。

   UNIXとWindows間で異なるプラットフォームのマシンを登録する際は、nmapmgrコマンドのSEt　Typeサブコマンドでそれぞれ NQS TYPEの適切な設定が必要になります。

2. qmgrコマンドでサブコマンドSEt MAChine_group=（<スケジューラマシン名> [, <マシン名> …]）を実行する

   対象マシンをマシングループに登録します。このとき、マシングループの先頭に指定されたマシンがスケジューラマシンとなります。また、必ず自分自身をグループ内に含むようにします。

   qmgrコマンドの引数はSEt　MAChine_group=部分を含めて1行256バイト(UNIX)または254バイト(Windows)以内で指定する必要があります。

   マシン名にはあらかじめ設定しておいたエイリアス名(別名)も指定できます。解除は自分だけを含むグループを再設定します。

   設定後の確認は、qmgrコマンドのSHOw MAChine_groupサブコマンドで行います。

3.2. ユーザの関連付けを行う（ユーザマッピング）

ネットワーク上にある他マシンのマシンIDの定義を行ったあと、ジョブ投入側のユーザとジョブ実行側のユーザの関連付けを行う必要があります。これをユーザマッピングと呼びます。

ユーザマッピングの設定には、GUI画面からとコマンドラインからとの2つの方法があります。なお、「3.1.3 マシングループ構成（JobCenter CL/Winからの登録）」 の方法でマシングループに追加した他マシンに対しては、JobCenter管理者アカウント同士のユーザマッピングが自動的に設定されます。

■投入側マシンの1ユーザIDを実行側マシンの複数ユーザIDにマッピングすることはできません(1:nのマッピング)。これはジョブ実行ユーザが一意に決まらないためです。投入側マシンの複数ユーザIDを実行側マシンの1ユーザIDにマッピングすることは可能です。(n:1のマッピング)。

■クラスタサイトを構成する各ノードマシン間で同じユーザ名を共有していても、ユーザIDが異なる場合はフェイルオーバ時にユーザマッピングが整合しなくなり、正常に動作しない場合があります。クラスタノード間ではユーザIDも揃えるよう注意してください。

3.2.1. CL/Win（ビューア）からユーザマッピングを行う

JobCenter CL/Win（GUI画面）からユーザマッピングを行う場合（標準リモートマシン構成）、ジョブリクエストを投入する側のマシン/サイトにCL/Winで接続して行います。

1. マネージャフレームの［マシン一覧］を表示します。

2. ［マシン一覧］のウィンドウ上で右クリックをしたときのポップアップメニューから［ユーザIDのマッピング］を選択するか、メニューバーの［設定］－［ユーザIDのマッピング］を選択します。

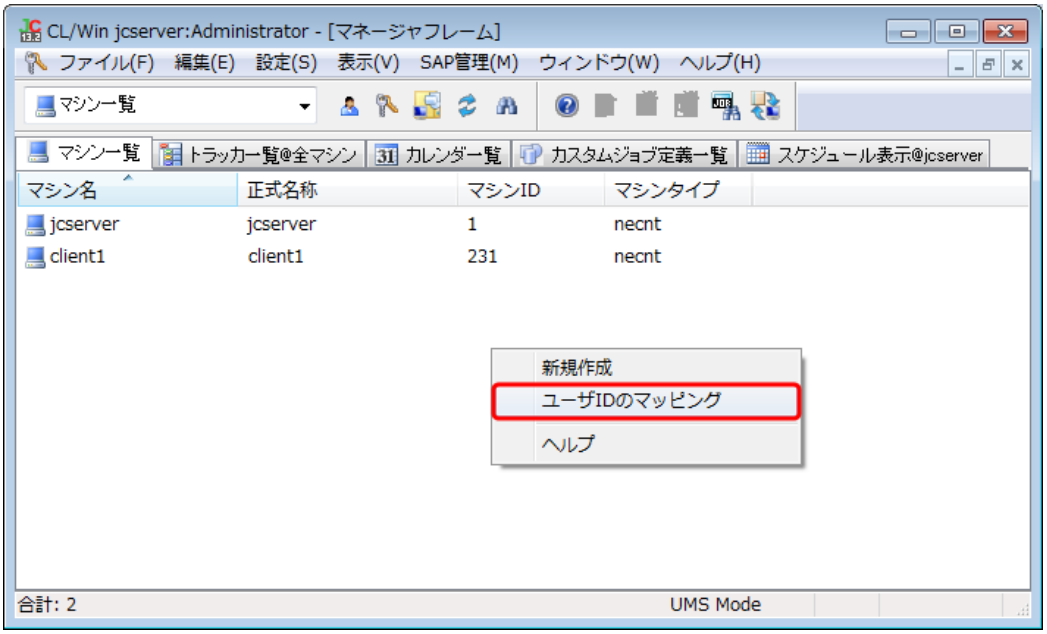

図3.8 ユーザIDのマッピングの設定画面例（標準リモートマシン構成機能）

3. ［ユーザマッピングの設定］ダイアログが表示されます。転送元マシンのマシンとユーザおよび転送先マシンのマシンとユーザを一覧から選択します。

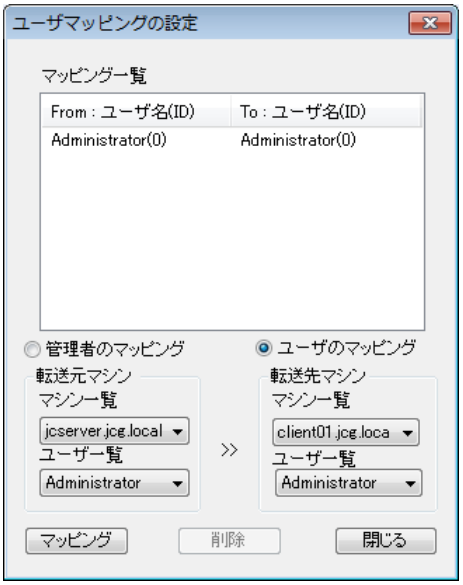

図3.9 ［ユーザマッピング］ダイアログ画面例

> 転送先マシンがWindowsの場合、ユーザ一覧にはそのWindows版JobCenter MGまたはSVに登録済みのユーザか、CL/Winで接続したことがあるユーザのみが表示されます。転送先マシンがUNIXの場合はOSに登録されている全てのユーザが表示されます。
>
> ユーザ登録の詳細については「13.4 ユーザの設定」を参照してください。

4. 設定後、［マッピング］ボタンをクリックします。

［ユーザマッピング］ダイアログの設定内容は次のとおりです。

■マッピング一覧

関連付けられた転送元マシンと転送先マシンのユーザマッピングの情報の一覧です。

From:ユーザ名(ID)	転送元マシンのユーザ名（ID）
To:ユーザ名(ID)	転送先マシンのユーザ名（ID）

■管理者のマッピング／ユーザのマッピング

マッピングの対象種別を選択します。管理者のマッピングかユーザのマッピングのいずれかを選択します。

管理者のマッピング	関連付ける転送元マシンの管理者ユーザと転送先マシンサーバの管理者ユーザを、暗黙のうちに選択します。 ▪ UNIXサーバの管理者ユーザはnsumsmgr権限もしくはroot権限です。 ▪ Windowsサーバの管理者ユーザはJobCenter管理者アカウントになります。
ユーザのマッピング	関連付けるマネージャのユーザアカウントとサーバのユーザをそれぞれ選択します。

■転送元マシン

ユーザマッピングする転送元のマシンとユーザを選択します。

マシン一覧	転送元のマシンをマシン一覧から選択します。
ユーザ一覧	転送元のマシンのユーザをユーザ一覧から選択します。

■転送先マシン

ユーザマッピングする転送先のマシンとユーザを選択します。

マシン一覧	転送先のマシンをマシン一覧から選択します。
ユーザ一覧	転送先のマシンのユーザをユーザ一覧から選択します。

■［マッピング］ボタン

ユーザマッピングを開始します。マッピングが正常終了するとマッピング一覧にマシンが追加されます。

■［削除］ボタン

マッピング一覧にて選択されたユーザのマッピングを削除します。

■［閉じる］ボタン

［ユーザマッピングの設定］ダイアログを閉じます。

ユーザマッピングを行った際にエラーが出力された場合

ユーザマッピング元、ユーザマッピング先それぞれのマシンに接続したCL/Winのマネージャフレームで、［マシン一覧］画面において表示されている相手のマシンを選択します。

1. 右クリックしたときのポップアップメニューから［マシンID一覧の変更］を選択し、［MID一覧の変更］ウィンドウにそれぞれのマシンIDが正しく登録されているか確認してください。
2. 登録されていない場合には、［MID一覧の変更］ウィンドウの［更新］ボタンをクリックし、マシンが登録されることを確認してください。
3. もし上記がうまくいかない場合は名前解決が双方で一致していない可能性がありますので、ネットワークの設定状況をよく確認してください。

3.2.2. コマンドラインからユーザマッピングを行う

ユーザマッピングのモード(Mapping mode)には次の3パターンあります。

モード	特徴
TYPE1	ユーザ名を用いたマッピング rshと同じ認証の仕組みにより、ジョブリクエスト投入側マシンと実行側マシンのユーザ名そのもので識別するため、実行側マシン上の.rhostsに投入側マシンのホスト名(やユーザ名)の設定が必要
TYPE2	JobCenter独自のマッピング uidによりユーザを識別するため、CL/Winまたはコマンドにより事前にユーザマッピング設定が必要
TYPE3	TYPE1とTYPE2を併用。TYPE1とTYPE2どちらかのマッピングが設定されていれば良い。JobCenter独自のマッピングの方が優先する

それぞれのマッピングモードについての詳細は<NQS機能利用の手引き>の「6.5.1　JobCenterネットワーク環境の概要」の「ユーザに関するネットワーク環境」を参照してください。

ここでは「3.2.1 CL/Win（ビューア）からユーザマッピングを行う」でCL/Winにより設定する場合と同様の設定、つまりTYPE2またはTYPE3で使用するJobCenter独自のマッピングを、コマンドラインから行う方法について説明します。

JobCenter独自のマッピングは、投入側マシンからジョブリクエストを受付ける際に、

■どのマシン/サイトから、

■どのuidのユーザのジョブリクエストを、

■自マシン上のどのuidのユーザでプロセスを実行するか

についての情報を、「ジョブリクエストを受付けて実行する側のマシン/サイト」に設定するものです。ジョブリクエスト投入側に設定するのではないことに注意してください。

CL/Winによる設定ではジョブリクエスト投入側のマシン/サイトに接続して設定しますが、実際の設定値は上記の通り実行側マシン/サイト上にマッピング設定内容が伝達されて格納されます。

JobCenter独自のマッピングは、ジョブリクエスト実行側マシンでroot(UNIX)もしくはJobCenter管理者アカウント(Windows)によりnmapmgrコマンドを起動し、サブコマンドを実行して設定します。

1. まず、マッピングを取りたいジョブリクエスト投入側のマシン/サイトのマシンIDを調べます。

```
# nmapmgr
NMAPMGR:> show state
HOST NAME: host2.co.jp  HOST ID: 120
NQS TYPE: EXTENDED TYPE OF NEC-NT
MAIL ADDRESS: not set
USER MAPPING

GROUP MAPPING
```

「HOST NAME: host2.co.jp」に並んで「HOST ID: 200」で示されているのがマシンIDです。

2. 次に、投入側マシンと実行側マシン間でマッピングを取りたいユーザのuidを**id**コマンド(UNIX)もしくはサーバの環境設定(Windows)で調べた上で、次のようにマッピングを設定します。

```
NMAPMGR>: add uid <投入側マシンID> <投入側マシン上のユーザID> <実行側ローカルマシン上のユーザID>
```

Windows版では、JobCenter管理者については実際のuidに関わらずユーザIDには全て0が割り当てられて扱われますので注意してください。

3. 問題なく設定されたことを確認します。下記の例ではhost2.co.jpのuid=200のユーザを、実行側である自マシン上のuid=300のユーザとマッピングしています。

```
# nmapmgr
NMAPMGR:> show state
HOST NAME: host2.co.jp  HOST ID: 120
NQS TYPE: EXTENDED TYPE OF NEC-NT
MAIL ADDRESS: not set
USER MAPPING
200 -> 300
GROUP MAPPING
```

show state引数のマシン名に別名を指定しても、情報は表示されます。

3.3. JobCenterネットワーク環境を構築する際の注意事項

3.3.1. 管理可能台数

1台のマシンで管理可能な最大マシン台数は8台が目安です。

9台以上のマシンを管理することも可能です。ただし、ネットワークの負荷が高くなり、レスポンス性能が低下します。

大規模な構成にする場合は、別々のマシンでそれぞれ複数のマシングループを設定、管理することで負荷を分散させてください。

3.3.2. マシングループ構成

マシングループを構成しているマシン同士は同一ネットワーク上に存在する必要があります。また、全てのマシン同士が名前解決の設定で正引きまたは逆引きできるよう設定を行っている必要があります。

resolv.defファイルを用いたJobCenterの名前解決を行っているWindowsマシンが存在するマシングループにUNIXマシンを追加しようとした場合、UNIXマシンでWindowsマシンの名前解決の設定を行っていないとマシングループの作成に失敗します。

3.3.3. UNIX版サーバのマシングループへの追加

UNIX版JobCenter SVをマシングループに追加する場合に、タイミングにより追加後にマシングループの取得に失敗する場合があります。

このような場合一旦マシングループ追加のダイアログを閉じて再度マシングループ追加のダイアログを開いてください。マシングループにすでに追加されている場合には、再度追加する必要はありません。

3.3.4. Windows版サーバのジョブ実行ユーザの追加

Windows版JobCenter SVにおいてジョブ実行ユーザ（ジョブのプロセスを起動したいユーザアカウント）を追加するには、以下のいずれかの作業が必要です。

■「サーバの環境設定」の「ユーザの追加」でユーザを登録する

「サーバの環境設定」を起動し「ユーザの追加」でユーザを登録してください。登録を行うには、ユーザ名、パスワードを入力して、［OK］ボタンをクリックしてください。詳細は「13.4　ユーザの設定」を参照してください。

■jc_usermgrコマンドのaddサブコマンドを実行する

jc_usermgrコマンドのaddサブコマンドを実行してください。詳細は<コマンドリファレンス>の「3.32 jc_usermgr JobCenterのユーザの管理」のaddサブコマンドの説明を参照してください。

> UNIX版JobCenterは上記作業は不要です。OSに存在する全てのユーザをジョブ実行ユーザとしてそのまま利用可能です。

3.3.5. WAN環境での利用

WAN環境で利用する場合には通信コストなどをよくご検討のうえご利用ください。

JobCenter MG機能を使用してネットワーク上のJobCenter SVを管理している場合、ジョブを動かしていない場合でも最低10分に1回の通信が行われます。

3.3.6. 管理者パスワード・ユーザパスワード

JobCenter管理者またはジョブ実行ユーザのパスワードを変更したときに必要になる処置について説明します。

■UNIXの場合

必要な処置はありません。OSアカウントが有効な状態を維持してください。

■Windowsの場合

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で新しいパスワードを再設定し、JobCenter MG/SVを再起動してください。

- ファイル待ち合わせ部品でリモートマシンの待ち合わせ対象をUNC表記で指定していない場合は、再起動ではなくqmgrコマンドの「reset usertoken」サブコマンドの実行でも対応可能です。

また、ジョブ実行ユーザのパスワードを変更した場合は、次のいずれかの処置を行ってください。

- 新しいパスワードで、JobCenter CL/WinからJobCenter MG/SVに接続し直す。
- JobCenter MG/SVで新しいパスワードを再設定する。

パスワードを変更したユーザで実行するジョブ(スクリプト)の中に、リモートマシンへアクセスする処理(リモートマシンへのファイルのコピー等)がある場合は、JobCenter MG/SVを再起動してください。

JobCenter MG/SVでのパスワードの設定方法は、「13.4 ユーザの設定」を参照してください。

> 複数のJobCenter MG/SVを利用している環境でドメインユーザのパスワードを変更した場合、
>
> - そのユーザをJobCenter管理者としている
>
> または
>
> - そのユーザをジョブ実行ユーザとしている
>
> に該当するJobCenter MG/SVは、複数存在する可能性があります。各JobCenter MG/SVが上記に該当するかどうかを確認し、該当するものについて漏れなく必要な処置を行ってください。

3.3.7. 登録可能なユーザ数

JobCenterグループに登録できるユーザ数の目安は1000ユーザまでです。

1000ユーザを超えて登録した場合は、レスポンス性能の低下などJobCenterが正常に動作しない可能性があります。

4. キューの作成

JobCenterを使用するために必要な環境を作成するために、キューの作成が必要となります。

JobCenterは、バッチ処理を行うためのシステムです。バッチ処理とは、リクエストを受け付けてキューイングし、順番に処理する機能です。

JobCenterはリクエストを受け付けると、一旦キューに登録し、順番に処理していきます。このキューは、システムの中に複数定義できます。それぞれのキューには、同時実行可能数、投入可能ユーザ、資源制限、キュー間実行優先順位などの属性を定義できます。

マシン上にジョブネットワークを実行するデフォルトのキューを作成する方法および新規にキューを作成する方法を説明します。

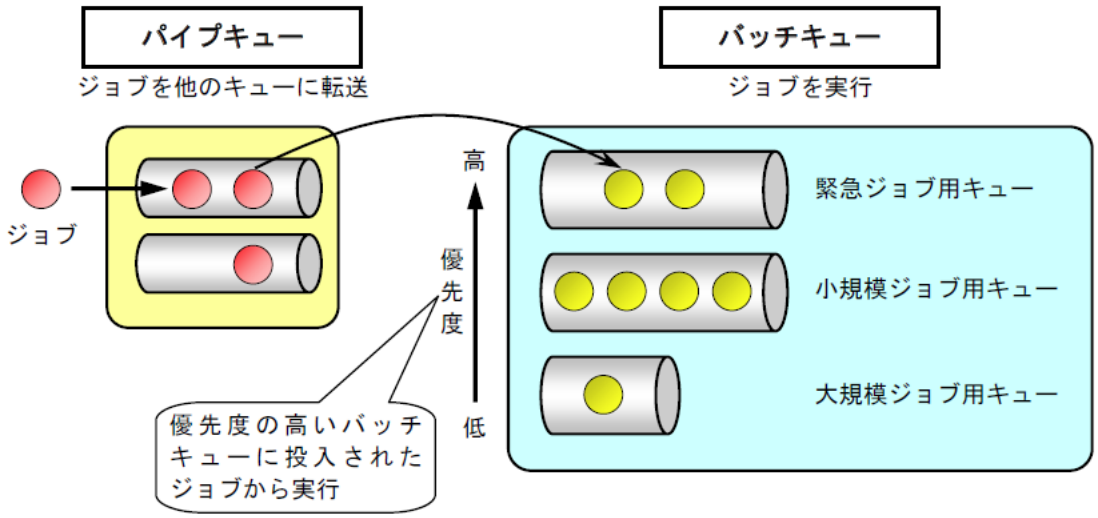

図4.1 バッチ処理のイメージ

Windows OSでアカウントのパスワードを変更した場合、「サーバの環境設定」の「ユーザ」画面でパスワードの状態をチェックしてください。

表示されているユーザについて［OK］になっていない場合は、CL/Winでそのユーザで新しいパスワードを指定してMG/SVに接続し直すか、「ユーザ」画面でパスワードを登録し直して［OK］にしてください。

［OK］になっていないままジョブリクエストを実行しようと、認証失敗によりバッチキューが停止してジョブが実行できなくなります。

(UNIXの場合はパスワード再設定は不要ですが、OSのアカウントが有効な状態を常に維持してください)

4.1. デフォルトで作成されるキュー

通常、JobCenterをインストール・セットアップした際にはデフォルトで以下のキューが自動作成されます。デフォルトで作成されるキューは 図4.2「デフォルトで作成されるキュー」および表4.1「デフォルトで作成されるキューの初期設定一覧」のとおりです。

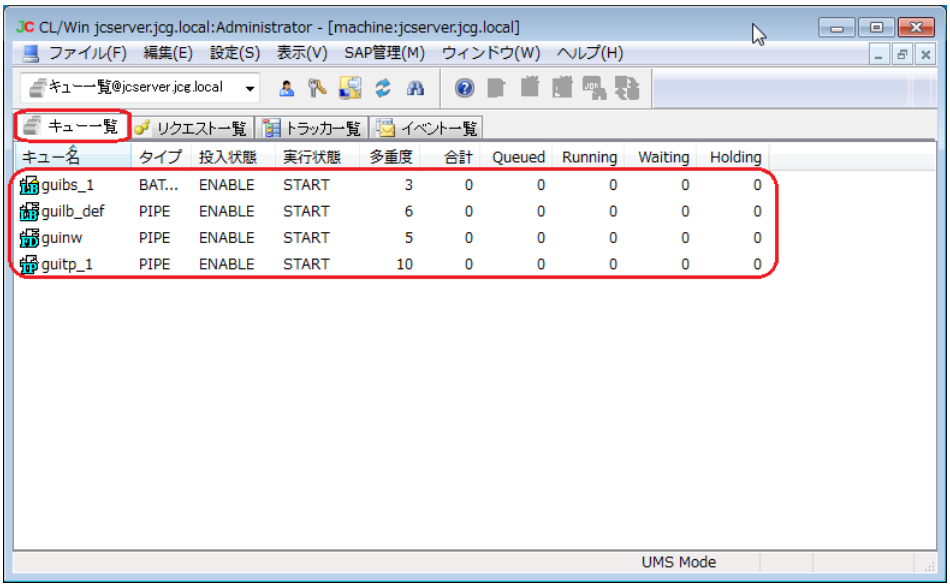

図4.2 デフォルトで作成されるキュー

表4.1 デフォルトで作成されるキューの初期設定一覧

キュー名	属　性	役　割	初期設定値
guibs_1	負荷分散バッチキュー	デマンドデリバリ方式で負荷分散パイプキューからジョブリクエストを受け取って実行します。	優先度 10 多重度　1
guinw	パイプキュー	自由転送キューです。他のマシンにジョブリクエストを転送する場合に使用します。	優先度 10 多重度　5
guitp_1	透過型パイプキュー	高速かつ低負荷で、ジョブリクエストをローカルのバッチキューに転送します。	優先度 10 多重度 10 転送先 guibs_1
guilb_def	負荷分散パイプキュー	負荷分散を行うときに有効です。初期設定ではデフォルトの投入先キューとして設定されています。	優先度 10 多重度　6 転送先 guitp_1
DefaultNetQue (UNIX版のみ)	ネットワークキュー	ジョブリクエスト実行結果の返却転送用キューでqmgrでのみ参照・制御可能です。(CL/Winでは表示不可。このキューが停止すると単位ジョブが実行中のままになりますので注意してください)	多重度 1000

デフォルトで作成されるキューの転送順については、「4.2　デフォルトのリクエスト転送順」を参照してください。

キューの詳しい役割については、<NQS機能利用の手引き>の「2.2 キュー」を参照してください。

4.2. デフォルトのリクエスト転送順

初期設定ではジョブネットワークを実行すると、フローに従ってデフォルトの投入先キューであるguilb_defパイプキューに順番に単位ジョブが投入されます。

その後、リクエストは次の順番で転送されます。

guilb_def 負荷分散パイプキュー	→	guitp_1 透過型パイプキュー	→	guibs_1 負荷分散バッチキュー

初期設定ではバッチキューguibs_1の多重度の値が1に設定されていますので、同時に複数のジョブリクエスト実行ができず、1つずつ順番に処理されます。

そのため、後から投入されたジョブリクエストは実行待ちになります。必要に応じてguibs_1の多重度の値を2以上に調節してください。

4.3. 自由なキュー構成を構築する

JobCenter CL/Winでは、標準的なキュー構成（デフォルトのキュー）を定めることでユーザの操作を軽減していますが、自由なキュー構成をユーザが自分で構築することも可能です。キュー構成を構築するためにはCL/Winから管理者アカウント（UMSモード）でログインしている必要があります。

追加可能なキューは次の2種類です。キューは合わせて500個まで作成可能です。

■バッチキュー（バッチリクエストを投入・実行するためのキュー）

■パイプキュー（ほかのキューにリクエストを転送するためのキュー）

バッチキューの作成とパラメータ設定およびパイプキューの作成、パラメータ設定、転送先設定については、<基本操作ガイド>の10章「ネットワークキューイングシステム（NQS）の利用方法」を参照してください。 また、各設定項目の詳細については<NQS機能利用の手引き>を参照してください。

4.4. キューの利用可能ユーザを制限する

管理者アカウントでログインした場合、ユーザに対して任意のキューの利用を許可したり、禁止したりできます。

キューの利用可能ユーザを制限する方法は、次のとおりです。

1. 管理者権限のアカウント（UMS　Mode）でログインし、マネージャフレームの［マシン一覧］を表示します。

2. ユーザを制御するキューを選択し、右クリックしたときのポップアップメニューから［キューユーザ］を選択します。

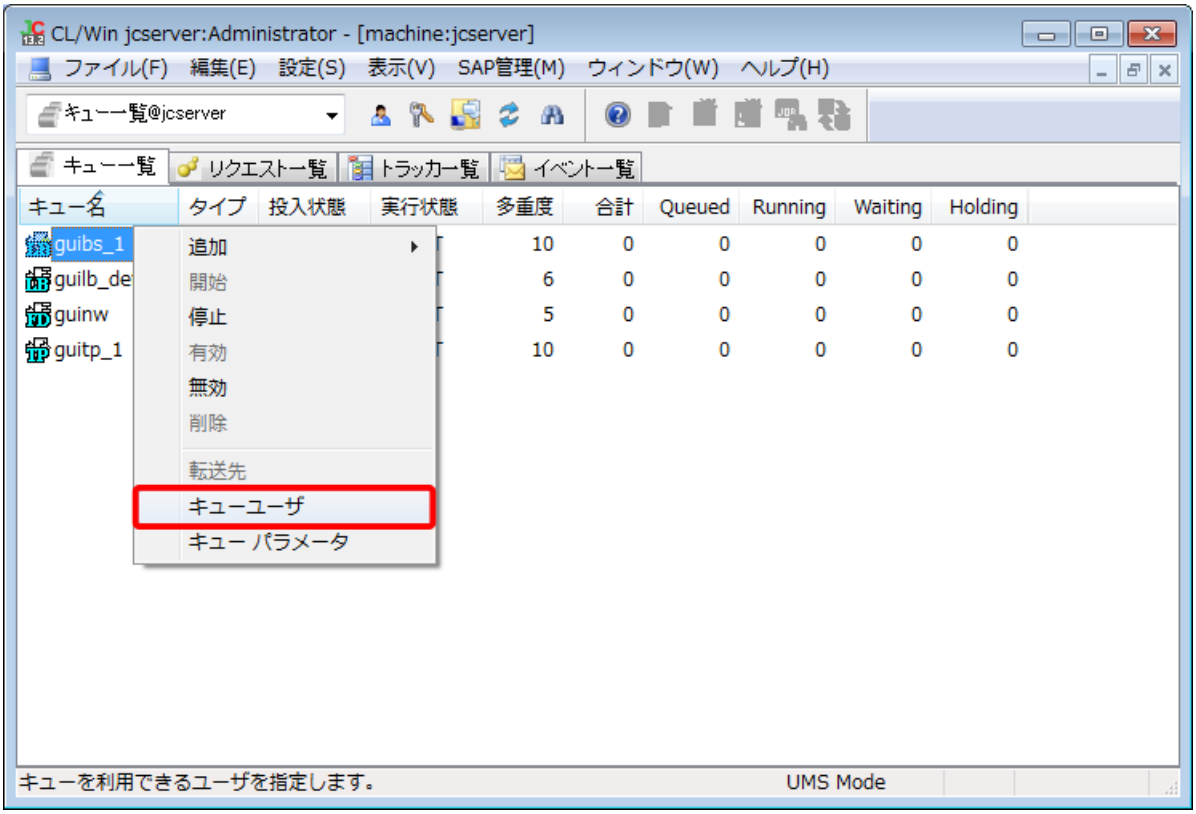

図4.3 ［キューユーザ］メニュー選択画面例

3. ユーザを制御するキューを選択し、右クリックしたときのポップアップメニューから[キューユーザ]を選択します。「ユーザ一覧」のリストには選択されたマシンに登録されているすべてのユーザが表示されます。

   デフォルトでは「すべてのユーザ」にチェックが入っています。この状態ではすべてのユーザが、そのキューを利用できます。

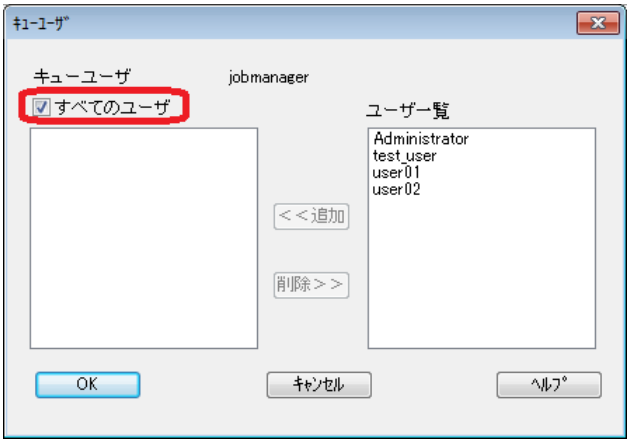

図4.4 ［キューユーザ］ダイアログ画面例

4. キューの利用可能ユーザを追加する場合は、「すべてのユーザ」のチェックをはずします。

5. 「ユーザ一覧」のリストからキューの利用を可能にするユーザやグループを選択し、［<<追加］ボタンをクリックします。また、キューの利用可能ユーザを削除する場合は、削除したいユーザを選択したあと、［削除>>］ボタンをクリックします。

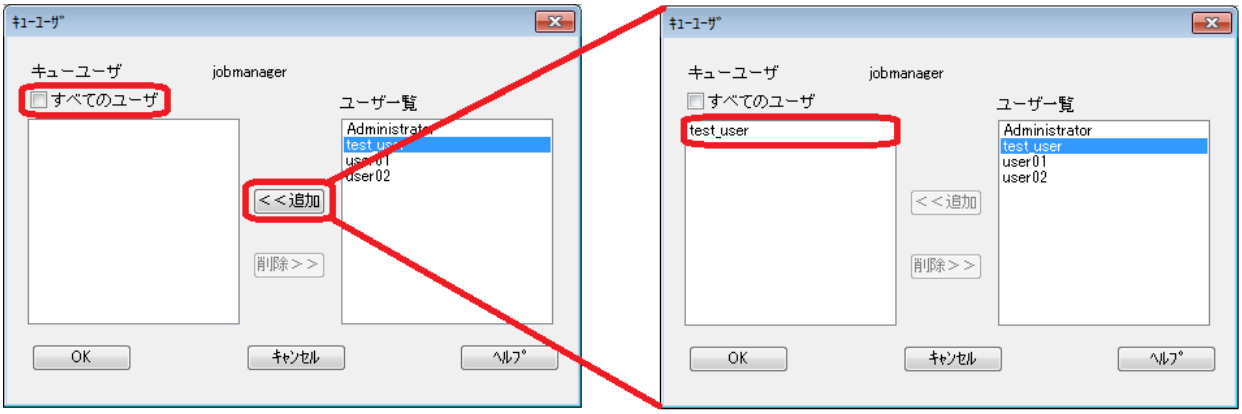

図4.5 キューの利用を可能にするユーザやグループの追加画面例

6. 設定後、［OK］ボタンをクリックします。

4.5. 管理マシンの詳細を設定する

マシンパラメータを編集し、管理マシンの詳細を設定します。

1. NQSフレームの［キュー一覧］においてマシンを選択し、メニューバー［設定］の［NQSパラメータ］を選択すると、現在マシンに設定されているパラメータを表示します。

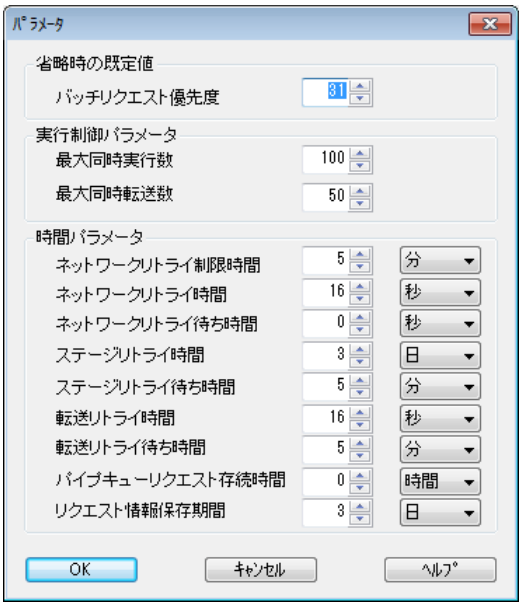

図4.6 NQSの［パラメータ］表示例

2. 設定後、［OK］ボタンをクリックするとマシンパラメータは修正値に更新されます。

設定項目の詳細については、<NQS機能利用の手引き>の「6.3　JobCenter環境パラメータの設定」を参照してください。

また、環境パラメータを定義、変更するためのコマンドはqmgrコマンドのサブコマンドとしてそれぞれ用意されています。qmgrコマンドのサブコマンドの詳細については<コマンドリファレンス>の「3.13　qmgr　構成管理および運用管理」の説明を参照してください。

次に同時実行可能なバッチリクエスト数の変更例を記載します。

（例）同時実行可能バッチリクエスト数を15に変更する

```
# qmgr ↵
Mgr: set global batch_request_limit 15
```

5. JobCenter起動時の設定を変更する

JobCenter起動時にデーモン設定ファイル（daemon.conf）を読み込むことにより、起動時の設定を変更できます。この設定によりJobCenter停止中のスケジュールを無効にするなどの運用が可能になり、障害復旧時などの運用性が向上します。

なおdaemon.confの設定変更を行った場合、次にJobCenterを再起動するまで変更内容は反映されませんのでご注意ください。

5.1. デーモン設定ファイル（daemon.conf）の格納場所

デーモン設定ファイル（daemon.conf）の格納場所は、環境により異なります。

5.1.1. Windowsの場合

デーモン設定ファイルを作成する場合は、次の場所に格納します。

JobCenter起動時にローカル環境・クラスタ環境それぞれのサイトごとに設置されたファイルを読み込みます。

ローカルサイト	%InstallDirectory%\etcディレクトリ配下
クラスタサイト	%JobCenterDatabaseDirectory%\etcディレクトリ配下

上記の %InstallDirectory% は JobCenter のインストールディレクトリです。デフォルトは C:\JobCenter\SV になります。

また %JobCenterDatabaseDirectory% は、cjcmksite でクラスタサイトを構築する際に指定した共有ディスク上のデータベースディレクトリです。

5.1.2. UNIXの場合

デーモン設定ファイルを作成する場合は、次の場所に格納します。

JobCenter起動時に、まずデフォルトで存在する共通ファイル、次に各サイトごとに設置されたファイルがあればそれを読み込みます。後に読み込まれるファイルのパラメータ記述内容が優先されます。

共通	/usr/lib/nqs/rc/daemon.conf (ローカルサイト、クラスタサイトの各デーモン設定ファイルよりも先に、デフォルトで存在する本ファイルが読み込まれます)
ローカルサイト	/usr/spool/nqs/daemon.conf
クラスタサイト	<JobCenterDBパス>/nqs/daemon.conf

共通のdaemon.confファイルにipaddressを記述すると、ローカルとクラスタでネットワーク資源の競合が起きることになり、正常に動作しません。

共通のdaemon.confには、ipaddressを記述しないでください。

5.2. デーモン設定ファイルの使用可能パラメータ

5.2.1. Windows/UNIX共通

1. SCLAUNCHD_FORGET

   SCLAUNCHD_FORGET=<秒> のようにデーモン設定ファイルに指定することで、スケジュール実行を管理するsclaunchdデーモンが起動したときに、<秒>以前に起動するようにスケジュール設定されていたジョブネットワークを起動しないように設定できます。

   <秒>には1～2147483647の範囲で指定します。

   たとえば、再起動の時点で予定日時より1日（86400秒）以上経過したスケジュールが設定されているジョブネットワークを起動しないようにするには、次のように指定します。

```
SCLAUNCHD_FORGET=86400
```

2. NQSDAEMON_OPT

   次のようにデーモン設定ファイルに指定すると、JobCenterの起動時にすべてのキューをSTOPした状態で起動できます。

```
NQSDAEMON_OPT=-s
```

   また、-xオプションと次のようにサブオプションを組み合わせて指定することで、JobCenterの起動設定を変更できます。

```
NQSDAEMON_OPT=-x trkappend=ON
```

サブオプション	概　要
trkappend=ON	コンティニュー部品などにより単位ジョブが再実行された場合に、標準出力および標準エラー出力の内容を当該単位ジョブの前回実行結果に追記します。
trkappend=OFF	コンティニュー部品などにより単位ジョブが再実行された場合に、標準出力および標準エラー出力の内容を当該単位ジョブの前回実行結果に上書きします。 本サブオプションが指定されない場合のデフォルト値です。
wildcard=ON	NQSのジョブリクエスト転送におけるTCP/IPのsource IPとして、destination IPと通信可能なセグメントに対応したIPアドレスを選んで用います。 これによりマシンが複数セグメントへのI/Fを持つ場合に、ジョブリクエスト投入元と投入先の間で異なる経路(ルーティング)でNQSの通信を行おうとしてリモート投入がエラーになる状態を回避できます。 ⚠ 本設定をNQSジョブリクエストを投入する側のdaemon.confに設定すると同時に、ジョブリクエストを投入される側のSVのdaemon.confにipcheck=OFFを必ず設定してください。 (複数のIPアドレスによる通信待ち受け設定やipcheck=OFF設定については「5.3.1 クラスタ構成／複数NIC使用時のdaemon.conf設定について」を参照してください)
wildcard=OFF	NQSのジョブリクエスト転送におけるTCP/IPのsource　IPとして、下記のIPアドレスを用います。 ■UNIXの場合 ローカルサイト (local_daemon=COMPAT)：destination IPと通信可能なセグメントに対応したIPアドレス

<table>
<tr><td></td><td>
<table>
<tr><td>ローカルサイト
(local_daemon=SITE)</td><td>ホスト名を正引きして得られたIPアドレス</td></tr>
<tr><td>クラスタサイト</td><td>サイト名を正引きして得られたIPアドレス</td></tr>
</table>

(local_daemonパラメータについては<NQS機能利用の手引き>の「7.3 デーモン起動オプション」を参照してください)

■Windowsの場合
<table>
<tr><td>ローカルサイト</td><td>ホスト名を正引きして得られたIPアドレス</td></tr>
<tr><td>クラスタサイト</td><td>サイト名を正引きして得られたIPアドレス</td></tr>
</table>

本サブオプションが指定されない場合のデフォルト値です。</td></tr>
<tr><td>idc_keepalive=間隔(秒)</td><td>マシングループを組んでいる環境またはMG-SV環境では、NQS間でセッションが張られたままとなります。

本サブオプションを指定すると、そのセッションに対して指定した間隔ごとにダミーのパケットを送信する様になります。

これにより、ルーターやFireWallで無通信状態のセッションを自動削除する設定がされているネットワーク環境にて、このセッションが無通信状態により削除されることを回避します。

間隔に指定できる範囲は60～86400の整数値で、60未満の場合には60で、86400より大きい場合には86400で動作します。

間隔に指定する値は、ルーターやFireWallの無通信状態のタイムアウト値より小さい値を指定してください。

本サブオプションを指定する場合には、MG、SVの両方に設定してください。

本サブオプションが指定されない場合には、一定間隔毎のダミーのパケットを送信しません。</td></tr>
<tr><td>stage_retry_

failed_action=

copytotempdir</td><td>SVで実行されたジョブの標準出力/標準エラー出力をMGへ転送できなかった場合、以下の場所に結果を格納したファイルが保存されます。

■UNIXの場合
<table>
<tr><td>標準出力</td><td><ジョブ実行ユーザのホームディレクトリ>/ジョブ名.o%seqno%</td></tr>
<tr><td>標準エラー出力</td><td><ジョブ実行ユーザのホームディレクトリ>/ジョブ名.e%seqno%</td></tr>
</table>

■Windowsの場合
<table>
<tr><td>標準出力</td><td><JobCenterDBパス>\spool\work\ジョブ名.o%seqno%</td></tr>
<tr><td>標準エラー出力</td><td><JobCenterDBパス>\spool\work\ジョブ名.e%seqno%</td></tr>
</table>

本サブオプションが指定されない場合のデフォルト値です。

■上記の %seqno% はリクエストIDのシーケンス番号部分の数値です。

■上記の <JobCenterDBパス> は ローカルサイトの場合、JobCenterのインストールディレクトリです。デフォルトはC:\JobCenter\SV になります。

クラスタサイトの場合、クラスタサイトを構築する際に指定した共有ディスク上のデータベースディレクトリです。</td></tr>
</table>

stage_retry_ failed_action= delete	SVで実行されたジョブの標準出力/標準エラー出力をMGへ転送できなかった場合、結果を格納したファイルを保存しません。

3. JNWENGINE_OPT

JNWENGINE_OPTに、次に示すオプションをデーモン設定ファイルにハイフン[-]付きで指定することで、JobCenterの主にジョブネットワークの各部品についての動作を変更できます。

オプションを複数指定する場合は、次のようにスペースで区切って指定します。

```
JNWENGINE_OPT=-os -ot -m 1000
```

オプション	概　要
oe	このオプションを指定するとタイムアウト時のEXITコードを異常とします。 -otオプションが設定されている場合のみ有効となります。
ot	このオプションを指定すると、ファイル待ち合わせ部品、イベント送受信部品がタイムアウトしたときにエラーとしません。
os	このオプションを指定すると、実行中の部品に対してスキップ操作を行ったときに、該当部品の状態がSKIP+ERRORとなります。 また、本オプション未指定時では、ジョブネットワークトラッカを再実行すると再実行前にSKIP+ERRORであった部品がSKIP+DONEに変わりますが、本オプション指定時は、SKIP+ERRORのままとなります。
s	このオプションを指定すると既存ジョブネットワーク情報の再構築を行いません。 既存ジョブネットワーク情報の再構築とは、失われたジョブネットワーク情報をアーカイブ前のトラッカデータをもとに、ジョブ実行中にデーモンの再起動を行ったときなどに行う処理です。
Ef	ファイル待ち合わせ部品でエラーが発生した場合、ジョブネットワークを自動停止します。 なお、ジョブネットワークのパラメータ「エラー時の自動停止」を「停止する」に設定しておく必要があります。
Es	イベント送信部品でエラーが発生した場合、ジョブネットワークを自動停止します。 なお、ジョブネットワークのパラメータ「エラー時の自動停止」を「停止する」に設定しておく必要があります。
Er	イベント受信部品でエラーが発生した場合、ジョブネットワークを自動停止します。 なお、ジョブネットワークのパラメータ「エラー時の自動停止」を「停止する」に設定しておく必要があります。
Tf	ファイル待ち合わせ部品の更新基準時刻とタイムアウトの相対時間の計算方法として、基準をジョブネットワークの開始時刻からファイル待ち合わせ部品の開始時刻に変更します。 またTfオプションを指定し、相対時刻指定のファイル待ち合わせ部品が待ち合わせ動作中にJobCenterが再起動した場合、その待ち合わせ部品は再起動時刻からの相対時間で再計算されます。
Tt	時刻待ち合わせ部品の時刻の相対時間の計算方法として、基準をジョブネットワークの開始時刻から時刻待ち合わせ部品の開始時刻に変更します。

	ただし、本オプションを指定している場合でも、時刻待ち合わせ部品から表示される実際待ち時刻ダイアログ中の時刻は、本オプションを指定していない場合と同じくジョブネットワークの開始時刻を基準としたものになります。これはJobCenter CL/Win上での表示上の問題であり、時刻待ち合わせ部品の開始時刻を基準として正しく動作します。正しい時刻はジョブネットワークのログで確認できます。 またTtオプションを指定し、相対時刻指定の時刻待ち合わせ部品が待ち合わせ動作中にJobCenterが再起動した場合、その待ち合わせ部品は再起動時刻からの相対時間で再計算されます。
Ts	イベント送信部品の受信確認タイムアウトとイベント有効期間タイムアウトの相対時間の計算方法として、基準をジョブネットワークの開始時刻からイベント送信部品の開始時刻に変更します。
Tr	イベント受信部品の受信タイムアウトの相対時間の計算方法として、基準をジョブネットワークの開始時刻からイベント受信部品の開始時刻に変更します。
n 回数	イベント送信(jnwetvcmdコマンドも含む)が通信状態等によりタイムアウトしないよう通信リトライを行う回数です。 設定範囲は1～2147483647の整数値で、範囲外の場合のデフォルト値は0回でリトライしません。
i 間隔	nオプションで指定したイベント送信リトライを行う際のリトライ間隔です。 設定範囲は1～2147483647の整数値で、範囲外の場合のデフォルト値は30秒です。
m 件数	イベント送信部品から送信されて自マシンに届いたイベントのうち、保持できるイベント件数を変更します。 設定範囲は500～5000件で、このオプションがない場合は、デフォルト値500件を使用します。
u	保留可能イベント数の諸元を超えた場合に、イベント上書きを禁止し、イベント送信を異常終了させます。
p 回数	ジョブの転送処理中にGUIからスキップや強制停止を行った場合や、クリティカルポイント警告および予想実行時間の超過が発生したときの警告動作のリトライ回数を指定します。 設定範囲は1～2147483647の整数値で、範囲外の場合のデフォルト値は10回です。
r 間隔	ジョブの転送処理中にGUIからスキップや強制停止を行った場合や、クリティカルポイント警告および予想実行時間の超過が発生したときの警告動作のリトライ間隔を指定します。 設定範囲は1～2147483647の整数値で、範囲外の場合のデフォルト値は8秒です。
R 回数	(Windowsのみサポート)ファイル待ち合わせ部品の対象がネットワークドライブ上のファイルの場合、通信状態等によりファイル待ち最終判定がタイムアウトしないよう通信リトライを行う回数です。 設定範囲は1～2147483647の整数値で、範囲外の場合のデフォルト値は10回です。
I 間隔	(Windowsのみサポート)Rオプションのファイル待ちリトライを行う際のリトライ間隔です。 設定範囲は5～60の整数値で、範囲外の場合のデフォルト値は5秒です。
It	このオプションを指定すると、トラッカのログに出力されるログの時刻がミリ秒まで出力されます。 このオプションが指定されない場合は、ログの時刻は秒までの出力となります。
F	ファイル待ち合わせ部品において、通過条件が「存在する」または「更新」の場合に、対象ファイルが他プロセスにオープンされていないこと、または、一定時間更新されていないことを確認します。本オプションが指定されない場合(デフォルト)は、ファイル作成途中で

あってもディスク上にファイルのエントリが作成された時点で通過条件を満たしたと判断されます。

本オプションは以下の2つのサブオプションを指定します。

close_check=ON	待ち合わせ対象ファイルが別のプロセスによりオープンされていないことを確認します。
update_check_value=間隔(秒)	ファイルの最終更新時刻が一定時間更新されていないことを確認します。間隔は秒で指定します。

ファイル作成中、ファイルがオープンされたままとなる場合にはclose_check=ONの指定のみ行えば問題ありません。そうではなく、オープンとクローズが繰り返されるような場合はupdate_check_valueを指定し、最終更新時刻から一定時間ファイルが更新されていないことを以てファイル作成が完了したと見なすようにしてください。これらの両方を指定することも可能です。

以下は両方を指定する場合の例です。

JNWENGINE_OPT=-F close_check=ON,update_check_value=60

ファイル待ち合わせ部品において「他のJobCenterサーバ上のファイルを待つ」の設定を行ってリモートサーバのファイルを待ち合わせる場合、待ち合わせ対象のファイルが存在するサーバ上のdaemon.confに本オプションを設定する必要があります。

UNIX/Linuxのclose_checkは、システムコマンドであるfuserを利用して行います。そのため、fuserが以下にインストールされている必要があります。

■Linux: /sbin/fuser

■Linux以外: /usr/sbin/fuser

fuserで検知できないプロセスについては、close_checkは有効に働きませんのでご注意ください。

Windowsのclose_checkでは、非共有モードで対象ファイルをオープンし、オープンできた場合はすぐにクローズして、他にオープンしているプロセスは存在しないと判断します。このチェックにおけるオープンからクローズまでの間は、他のプロセスはアクセスできません。そのため、ファイル作成がオープンとクローズを繰り返すような場合、ファイルアクセスがエラーとなることを考慮してリトライを行う等の対処が必要です。これが難しい場合は、update_check_valueによる最終更新時刻の確認のみ行うようにしてください。

4. JNWENGINE_RMTFCHECK_RTYTIMEOUT

JNWENGINE_RMTFCHECK_RTYTIMEOUT=<秒> のようにデーモン設定ファイルに指定すると、ファイル待ち合わせ部品が「他のJobCenterサーバ上のファイルを待つ」を実行する際に、リモートマシンとの通信障害(パケット通信のタイムアウト含む)発生時に異常終了と判断するまでのリトライ時間(秒)を設定できます。

■<秒>は1～2147483647の範囲で指定します。

■0や負の値の場合には無効な設定となり、デフォルト動作と同じように900秒間連続で通信障害が発生したらファイル待ち合わせ部品を異常終了します。

リモートファイルの待ち合わせ処理で、通信障害が1時間連続で発生した場合に異常終了させる場合は、次のように指定します。

```
JNWENGINE_RMTFCHECK_RTYTIMEOUT=3600
```

5. JNWENGINE_RMTFCHECK_INTERVAL

JNWENGINE_RMTFCHECK_INTERVAL=<秒>　のようにデーモン設定ファイルに指定すると、ファイル待ち合わせ部品が「他のJobCenterサーバ上のファイルを待つ」を実行する際に、リモートマシン上で待ち合わせファイルの状態をチェックする間隔を設定できます。

■<秒>は1～2147483647の範囲で指定します。

■0や負の値の場合には無効な設定となり、デフォルト動作と同じように20秒間隔でリモートマシン上の待ち合わせファイルをチェックします。

サーバ負荷軽減を考慮してリモートファイルの待ち合わせ処理を2分毎にチェックするには、次のように指定します。

```
JNWENGINE_RMTFCHECK_INTERVAL=120
```

ファイルチェックのインターバルを短時間にした状態で複数のリモートファイル待ち合わせ処理を同時実行した場合、JobCenterへの負荷が非常に大きくなります。本パラメータを設定される場合には、マシンスペックを十分考慮された上で運用してください。

6. JNWENGINE_RMTFCHECK_RUNLIMIT

JNWENGINE_RMTFCHECK_RUNLIMIT=<同時実行数>　のようにデーモン設定ファイルに指定すると、ファイル待ち合わせ部品が「他のJobCenterサーバ上のファイルを待つ」を実行する際に、同時に実行できるリモートファイル待ち合わせ処理の数を設定できます。

■<同時実行数>はOSに設定された1プロセスあたりのファイルディスクリプター制限値および、JobCenterの制限値400の範囲内で指定します。

■0や負の値の場合には無効な設定となり、デフォルト動作と同じように30個のファイル待ち合わせ部品を実行上限としてリモートファイルの待ち合わせ処理を行います。

ファイル待ち合わせ部品で、リモートファイルの待ち合わせの処理が最大45個同時に行われるようにする場合は、次のように指定します。

```
JNWENGINE_RMTFCHECK_RUNLIMIT=45
```

大量のリモートファイル待ち合わせ処理を同時実行した場合、JobCenterへの負荷が非常に大きくなります。本パラメータを設定する場合は、マシンスペックを十分考慮して運用してください。

また、ファイル待ち合わせ部品のリモートファイル待ち合わせ処理が実行上限数（デフォルト30）を超えて行われた場合、リモートファイル待ち合わせ処理は実行待ち状態となります。

同時実行数に空きができるまで待ち合わせてからリモートファイル待ち合わせ処理が実行されます。

7. ipaddress

ipaddressに次の形式(複数のIPアドレスをカンマ（,）区切りで記述)で記述する事により、複数のIPアドレスで通信を待ち受けることができます。

記述できるIPアドレスはIPv4アドレス、IPv6アドレスそれぞれ5つまでです。

```
ipaddress=IPv4アドレス1,IPv4アドレス2,IPv4アドレス3,IPv4アドレス4,IPv4アドレス5,IPv6アドレス1,IPv6アドレス2,IPv6アドレス3,IPv6アドレス4,IPv6アドレス5
```

8. bindmode

bindmodeに次のオプションを設定する事で、IPv4アドレスのみ、IPv6アドレスのみ、IPv4アドレス・IPv6アドレスの両方で通信を待ち受けることができます。

未設定の場合には、IPv4アドレスがある場合にはIPv4アドレスで、IPv6アドレスのみの場合にはIPv6アドレスで通信を待ち受けます。(IPv4アドレスのみが優先されます)

```
bindmode=all
```

オプション	概要
ipv4	IPv4アドレスのみで通信を待ち受けます
ipv6	IPv6アドレスのみで通信を待ち受けます
all	IPv4アドレス・IPv6アドレスの両方で通信を待ち受けます

daemon.confにipaddressのパラメータの設定がある場合には、bindmodeのパラメータは無効となります。

9. ipcheck

ipcheck=OFF（値は大文字で指定）のようにデーモン設定ファイルに指定すると、ジョブ投入元サーバ(MG)のIPアドレスからホスト名の逆引きを行い、投入元(MG)のチェックを行なわなくなります。

未設定の場合には、投入元(MG)のIPアドレスからホスト名の逆引きを行い、投入元(MG)のチェックを行います。

```
ipcheck=OFF
```

10. JL_LOGMONITOR_FORGET

```
JL_LOGMONITOR_FORGET=ON
```

上記をデーモン設定ファイルに指定することで、JobCenterが停止している間に起動トリガのテキストログ監視で監視対象となっているテキストログファイルに追加されたログをJobCenter起動時に監視しません。

起動トリガのテキストログ監視の詳細は、<基本操作ガイド>の7章 「スケジュール以外の契機でジョブネットワークを投入する方法」 を参照してください。

未設定および「JL_LOGMONITOR_FORGET=OFF」の場合には、JobCenterが停止している間に起動トリガのテキストログ監視で監視対象となっているテキストログファイルに追加されたログをJobCenter起動時に監視します。

JobCenterが停止している間に起動トリガのテキストログ監視で監視対象となっているテキストログファイルに大量のログが出力されていた場合、JobCenterの起動時に大きな負荷がかかる可能性があります。

11. JL_LOGMONITOR_INTERVAL

JL_LOGMONITOR_INTERVAL=<秒>　のようにデーモン設定ファイルに指定すると、起動トリガのテキストログ監視で監視対象となっているテキストログファイルに追加されたログを監視する間隔を設定できます。

起動トリガのテキストログ監視の詳細は、<基本操作ガイド>の7章 「スケジュール以外の契機でジョブネットワークを投入する方法」 を参照してください。

■<秒>は1～2147483647の範囲で指定します。

■0や負の値の場合には無効な設定となり、デフォルト動作と同じように10秒間隔で監視します。

12. JL_LOGMONITOR_MAXLINE

JL_LOGMONITOR_MAXLINE=<行数> のようにデーモン設定ファイルに指定すると、起動トリガのテキストログ監視で監視対象となっているテキストログファイルの１回の監視で読み込む追加されたログの最大行数を設定できます。最大行数を超えたログ部分については次回の監視時に持ち越されます。

起動トリガのテキストログ監視の詳細は、<基本操作ガイド>の7章 「スケジュール以外の契機でジョブネットワークを投入する方法」 を参照してください。

■<行数>は1～2147483647の範囲で指定します。

■0や負の値の場合には無効な設定となり、デフォルト動作と同じように最大10000行読み込みます。

13. JNWLAUNCHER_BOOT_TIMEOUT

JNWLAUNCHER_BOOT_TIMEOUT=<秒>　のようにデーモン設定ファイルに指定すると、jnwlauncher(起動トリガの各デーモンを管理するデーモン)の起動時のタイムアウト時間を設定できます。

JobCenterを再起動すると未アーカイブのトラッカの再構築が行われ、jnwlauncherはこの処理の完了を待ち合わせて起動します。JNWLAUNCHER_BOOT_TIMEOUTはこの待ち合わせのタイムアウト時間となります。

■<秒>は1～2147483647の範囲で指定します。

■0や負の値の場合には無効な設定となり、デフォルト動作と同じように120秒のタイムアウト時間が設定されます。

14. COMAGENT_OPT

操作ログ、実行ログを作成する設定の場合、名前解決に時間がかかる環境においてCL/Winの操作レスポンスが悪化することがあります。

次のようにデーモン設定ファイルに指定すると、操作ログ、実行ログ作成の際に接続元マシン名をマシン名に変換せずIPアドレスで記録します。

未設定の場合には、接続元マシン名を逆引きの名前解決を行ってマシン名に変換します。

```
COMAGENT_OPT=-i
```

15. COMAGENT_OTHR_SVR_TIMEOUT

COMAGENT_OTHR_SVR_TIMEOUT=<秒> のようにデーモン設定ファイルに指定すると、comagent(通信用デーモン)のソケット通信のタイムアウト時間を設定できます。

CL/Winからサーバへ接続する際のソケットの待ち時間を長くしたい場合に、デフォルト値の30秒よりも長い値を設定し、JobCenterを再起動してください。

■<秒>は30～2147483647の範囲で指定します。

■範囲外の値の場合には無効な設定となり、デフォルト動作と同じように30秒のタイムアウト時間が設定されます。

16. COMAGENT_SSLCERT

CL/Winとサーバの通信を暗号化するための、TLS証明書ファイルを設定します。

設定方法は、以下の3通りがあります。

■証明書の絶対パスを指定

例

```
COMAGENT_SSLCERT=/usr/local/ssl/server.crt
```

■証明書のファイル名だけを指定

例

```
COMAGENT_SSLCERT=tls.crt
```

証明書ファイルの格納先は以下の通りです。

- Windowsの場合

ローカルサイト	%InstallDirectory%\etcディレクトリ配下
クラスタサイト	%JobCenterDatabaseDirectory%\etcディレクトリ配下

- UNIXの場合

ローカルサイト	/usr/spool/nqsディレクトリ配下
クラスタサイト	<JobCenterDBパス>/nqsディレクトリ配下

■設定を省略

証明書ファイルのファイル名は、「ssl_cert」です。

証明書ファイルの格納先は、以下の通りです。

- Windowsの場合

ローカルサイト	%InstallDirectory%\etc\ssl_cert
クラスタサイト	%JobCenterDatabaseDirectory%\etc\ssl_cert

- UNIXの場合

ローカルサイト	/usr/spool/nqs/ssl_cert
クラスタサイト	<JobCenterDBパス>/nqs/ssl_cert

■証明書ファイルを配置しない場合、サーバ側で通信の暗号化用のプロトコル（jccombase-over-ssl）を使用しない状態となります。そのため、CL/Winで「保護された接続」を選択して接続することはできません。

■Windowsの場合、TLS証明書のパス、または、ファイル名にスペースを使用できません。

17. COMAGENT_SSLKEY

CL/Winとサーバの通信を暗号化するための、TLS秘密鍵ファイルを設定します。

設定方法は、以下の3通りがあります。

■秘密鍵ファイルの絶対パスを指定

例

```
COMAGENT_SSLKEY=/usr/local/ssl/server.key
```

■秘密鍵のファイル名だけを指定

例

```
COMAGENT_SSLKEY=tls.key
```

秘密鍵ファイルの格納先は以下の通りです。

- Windowsの場合

ローカルサイト	%InstallDirectory%\etcディレクトリ配下
クラスタサイト	%JobCenterDatabaseDirectory%\etcディレクトリ配下

- UNIXの場合

ローカルサイト	/usr/spool/nqsディレクトリ配下
クラスタサイト	<JobCenterDBパス>/nqsディレクトリ配下

■設定を省略

秘密鍵ファイルのファイル名は、「ssl_key」です。

秘密鍵ファイルの格納先は、以下の通りです。

- Windowsの場合

ローカルサイト	%InstallDirectory%\etc\ssl_key
クラスタサイト	%JobCenterDatabaseDirectory%\etc\ssl_key

- UNIXの場合

ローカルサイト	/usr/spool/nqs/ssl_key
クラスタサイト	<JobCenterDBパス>/nqs/ssl_key

■秘密鍵ファイルを配置しない場合、サーバ側で通信の暗号化用のプロトコル（jccombase-over-ssl）を使用しない状態となります。そのため、CL/Winで「保護された接続」を選択して接続することはできません。

■Windowsの場合、秘密鍵ファイルのパス、または、ファイル名にスペースを使用できません。

18. jcwebserver_enabled

jcwebserver_enabled=false のようにデーモン設定ファイルに指定すると、jcwebserverのデーモンが起動しなくなります。

デーモン設定ファイルに jcwebserver_enabled=true を設定、またはjcwebserver_enabledパラメータを未設定の場合には、jcwebserverのデーモンは起動する設定となります。

但し、本パラメータはLinuxとWindows環境でのみ有効なパラメータとなります。

■jcwebserverのデーモンを起動しない場合、JobCenter　MG/SVのWebAPIは使用できなくなります。

■HP-UXおよびAIX環境では jcwebserver のデーモンは存在しないため本パラメータは無効となります。

19. JNWENGINE_PKTTIMEOUT

フロー内に数百単位で部品が配置されている場合や多数のサブJNWが配置されている等で、JNW定義が非常に肥大化している場合、ジョブ投入時の待ち時間が60秒(タイムアウト時間のデフォルト値)を超過してしまい、ジョブ投入に失敗する事があります。

このような場合に、本パラメータでタイムアウト時間の調整が可能です。

※部品数等は環境(マシンスペックやCPUの負荷等)に依存するので、具体的な数値はありません。

デーモン設定ファイルおよび環境変数にJNWENGINE_PKTTIMEOUT=<秒>のように指定すると、各モジュールが処理要求パケットを送信してから処理結果パケットが返って来るまでのタイムアウト時間を設定できます。

デーモンプロセスに設定する場合は、デーモン設定ファイルに設定してJobCenterを再起動してください。

コマンドを実行する場合はデーモン設定ファイルを参照しないため、環境変数の設定が必要となります。設定方法はデーモン設定ファイルと同様にJNWENGINE_PKTTIMEOUT=<秒>となります。

Windowsの場合、システム環境変数を変更した場合はJobCenterの再起動が必要です。OSによってはOSの再起動が必要になる場合もあります。

■<秒>は30～180の範囲で指定します。

■範囲外の値の場合には無効な設定となり、デフォルト動作と同じように60秒のタイムアウト時間が設定されます。

各モジュールのタイムアウト時間の設定方法については、以下の表を参照してください。

モジュール名	実行方式	タイムアウト値の設定方法
comagent	デーモンプロセス	daemon.conf
jnwengine	デーモンプロセス	daemon.conf
sclaunchd	デーモンプロセス	daemon.conf
jnwevtcmd	コマンド	環境変数
jnwevtmgr	コマンド	環境変数
jnwop	コマンド	環境変数
jnwsubmitcmd	コマンド	環境変数

5.2.2. Windowsの場合

1. NQSDAEMON_OPT

UNIXと共通のパラメータ以外のものについては、以下のオプションが指定可能です。

オプション	概　要
b バックログ値	nqsdaemon(ポート番号607)のバックログ値 ※指定しなかった場合のデフォルト値300
c	ジョブ実行毎にユーザトークン情報を取得する動作を行います。

	Windows版JobCenterは、初回ジョブ実行時にユーザトークンをキャッシュし、以後はキャッシュしておいたユーザトークンを再利用してジョブを実行します。そのため、初回ジョブ実行後に、ユーザの設定変更(所属グループの変更等)を行ってもJobCenterを再起動しない限り反映されません。 本オプションを指定することでジョブ実行毎にユーザトークン情報を取得しますので、ユーザ設定が即座に反映されます。ただし、その分JobCenterの実行性能は劣化しますので注意してください。

-xオプションのサブオプションは以下の通りです。

```
NQSDAEMON_OPT=-x tcp_keepalivetime=900,tcp_keepaliveinterval=10
```

サブオプション	概　要
tcp_keepalivetime=アイドル時間(秒)	MGのNQSのポートに対して接続されるセッションにおいて、指定されたアイドル時間(tcp_keepalivetime)　SVからの通信がなかった場合に、TCP/IPのKeepAliveパケットをSVへ送信します。本パラメータは5～1800(秒)の間で指定します。デフォルト値は900(秒)です。
tcp_keepaliveinterval=再送間隔(秒)	KeepAliveパケットに対してSVからの応答がない場合、指定された再送間隔(tcp_keepaliveinterval)でKeepAliveパケットを送信します。また、OSで設定されるパケット再送回数分[注]送信を行っても応答が返ってこない場合、そのセッションを切断します。本パラメータは1～30(秒)の間で指定します。デフォルト値は10(秒)です。

[注]WindowsのTCP/IPにおいて、TcpMaxDataRetransmissions として設定される値

2. JNWENGINE_OPT

UNIXと共通のパラメータ以外のものについては、以下のオプションが指定可能です。

オプション	概　要
b バックログ値	jnwengine(ポート番号609)のバックログ値 ※指定しなかった場合のデフォルト値300
B バックログ値	jnwengine(ポート番号10012)のバックログ値　※指定しなかった場合のデフォルト値300

3. COMAGENT_OPT

UNIXと共通のパラメータ以外のものについては、以下のオプションが指定可能です。

オプション	概　要
b バックログ値	comagent(ポート番号611)とcomagent-over-ssl(ポート番号23116)のバックログ値 ※指定しなかった場合のデフォルト値300

5.2.3. UNIXの場合

1. NQSDAEMON_OPT

Windowsと共通のパラメータ以外のものについては、<NQS機能利用の手引き>の「7.3　デーモン起動オプション」を参照してください。

2. NQSDAEMON_PATH_EXPORT

```
NQSDAEMON_PATH_EXPORT=1
```

上記をデーモン設定ファイルに指定することで、ジョブ実行時の環境変数PATHをジョブ投入元からジョブ投入先に引き継ぐ事ができます。

本バージョンでは　NQSDAEMON_PATH_EXPORTに1以外の値が指定されていても本機能が有効となりますが、今後指定方法が変更となる可能性がありますので、必ず1を指定するようにしてください。

なお、本機能を無効にするためには、デーモン設定ファイル(daemon.conf)から「NQSDAEMON_PATH_EXPORT」を削除してください。

環境変数PATHの引き継ぎについては「15.1.2 JobCenter SV側の環境変数」を参照してください。

3. schedule_timezone

```
schedule_timezone=OFF
```

上記をデーモン設定ファイルに指定することで、サーバで設定したOSのタイムゾーン(環境変数:TZの値)を使用してスケジュールの時刻を計算するように変更できます。

> 日本(JST-9)以外のタイムゾーンでJobCenterを使用する場合には、本設定を必ず実施してください。詳細は「16.1 JobCenterセットアップ後に必要な設定」をご覧ください。

4. JNWENGINE_RMTFCHECK_PKTTIMEOUT

JNWENGINE_RMTFCHECK_PKTTIMEOUT=<秒> のようにデーモン設定ファイルに指定すると、ファイル待ち合わせ部品が「他のJobCenterサーバ上のファイルを待つ」を実行する際に、リモートマシンにファイルチェック問い合わせパケットを送信してから処理結果パケットが返って来るまでのタイムアウト時間を変更できます。

■<秒>は1～2147483647の範囲で指定します。

■0や負の値の場合には無効な設定となり、デフォルト動作と同じように900秒でタイムアウト処理を行います。

リモートマシンにファイル待ち合わせ処理パケット送信後、1時間しても結果の応答がない場合にタイムアウトさせる場合は、次のように指定します。

```
JNWENGINE_RMTFCHECK_PKTTIMEOUT=3600
```

5. JCDBS_BOOT_TIMEOUT

JCDBS_BOOT_TIMEOUT=<秒> のようにデーモン設定ファイルに指定すると、JCDBS デーモン起動時のタイムアウト時間を変更できます。

■<秒>は1～2147483647の範囲で指定します。

■0や負の値の場合には無効な設定となり、デフォルト動作と同じように60秒でタイムアウト処理を行います。

6. NQSDAEMON_BOOT_TIMEOUT

NQSDAEMON_BOOT_TIMEOUT=<秒> のようにデーモン設定ファイルに指定すると、NQSDAEMON デーモン起動時のタイムアウト時間を変更できます。

■<秒>は1～2147483647の範囲で指定します。

■0や負の値の場合には無効な設定となり、デフォルト動作と同じように60秒でタイムアウト処理を行います。

7. JNWENGINE_SPAWN_RETRYNUM

JNWENGINE_SPAWN_RETRYNUM=<回数>　　のようにデーモン設定ファイルに指定すると、jnwengine-spawnプロセスの起動が確認できない場合、本パラメータで追加した設定回数まで1秒毎に起動確認のリトライを行います。

■<回数>は1～180の範囲で指定します。

■本パラメータを設定しない場合および、有効範囲外の値が指定された場合は30が設定されます。

5.3. 複数IPアドレス使用時のdaemon.conf設定について

この章では、複数IPアドレス使用時のdaemon.conf設定について説明します。

5.3.1. クラスタ構成／複数NIC使用時のdaemon.conf設定について

クラスタ構成で複数のIPアドレスを使用して運用を行いたい場合、次の設定により複数のIPアドレスで通信を待ち受けできます。

以下は複数のIPアドレスでTCP/IPソケット通信の「待ち受け」を行う設定です。TCP/IPソケット通信で送信する際のIPアドレスについては、後述の wildcard=ON/OFF によるsource IP割り当ての実装についての説明を参照してください。

1. local_daemon (UNIXの場合)

   ローカルサイトとクラスタサイトを同じマシンで共存させる場合、あらかじめローカルサイト側のdaemon.confにサイトモードで起動するようにlocal_daemonパラメータを設定して、ローカルサイトを再起動してください。local_daemonパラメータについては<NQS機能利用の手引き>の「7.3　デーモン起動オプション」を参照してください。

   ```
   local_daemon=SITE
   ```

2. ipaddress (Windows/UNIX共通)

   パラメータipaddressに次の形式(複数のIPアドレスをカンマ（,）区切りで記述)で記述し、JobCenterを再起動してください。詳細は「5.2.1 Windows/UNIX共通」のipaddressを参照してください。

   ```
   ipaddress=IPアドレス1,IPアドレス2,IPアドレス3,IPアドレス4,IPアドレス5
   ```

3. ipcheck (Windows/UNIX共通)

   ジョブ投入元サーバ(MG)のIPアドレスからホスト名の逆引きを行って投入元(MG)のチェックを行いたくない場合には、次の設定をdaemon.confに記述してJobCenterを再起動してください。詳細は「5.2.1 Windows/UNIX共通」のipcheckを参照してください。

   ```
   ipcheck=OFF
   ```

1台のWindows上でクラスタサイトのJobCenter MG/SVとローカルサイトのJobCenter MG/SVが共存している場合は、クラスタサイトだけでなくローカルサイト側のdaemon.confファイルにも上記のipaddress、ipcheckの設定を行ってください。

その際、同一のIPアドレスをローカルサイトとクラスタサイトそれぞれのdaemon.confファイルのipaddressパラメータに重複して指定しないように注意してください。もし重複した場合、通信パケットをローカルサイト側とクラスタサイト側で取り合うことになり、正常に通信できなくなります。

Windowsの場合はOSの名前解決方法の問題により、上記ipaddress、ipcheck設定だけでは正常に通信できない場合があります。その際はresolv.defファイルの設定が追加で必要になります。「2.3 Windows環境における名前解決方法」も参照してください。

なお、複数NIC環境でJobCenterを使用する場合(または、ローカルサイトとクラスタサイトを同一マシン上で同時に使用する場合)、以下の点に注意してください。特にFireWallでTCP/IPのsource　IPによるアクセス制限を行う場合等に注意してください。

wildcardの設定については 「5.2.1 Windows/UNIX共通」 の NQSDAEMON_OPT サブオプション「wildcard」の項を参照してください。

1. JobCenterにおいて、TCP/IP通信のsource IPはwildcard=ON/OFFによって次のように割り当てられるよう実装されています。

JobCenterのプロトコル	wildcard=ON	wildcard=OFF
NQS(607/tcp)	INADDR_ANY	OSによる名前解決で自サイト名から得られたIPアドレス [注]
jccombase(611/tcp)	INADDR_ANY	INADDR_ANY
jcevent(10012/tcp)	INADDR_ANY	INADDR_ANY

[注]UNIX版でlocal_daemon=COMPAT指定時を除く。 source IPの詳細は「5.2.1 Windows/UNIX共通」の NQSDAEMON_OPT wildcard=OFF を参照してください。

INADDR_ANYの場合、次のように動作します。

■TCP/IPのsource IPにdestination IPと通信可能なセグメントに対応したIPアドレスが用いられます。どのIPアドレスが用いられるかの優先順位はOSにより決定されます。(これは複数NIC環境に限りません。同一マシン上で動作するクラスタサイトとローカルサイトの通信で、source IPがローカルサイトのIPアドレスになる場合があることを示します)

■かつ、destination IP側のMG/SVではipcheck=OFFの設定が必要になります。

2. クラスタ構成/複数NIC環境でJobCenterを使用する場合の設定と動作の例を示します。

下記のサーバ(hostA)には次の条件があるものとします。

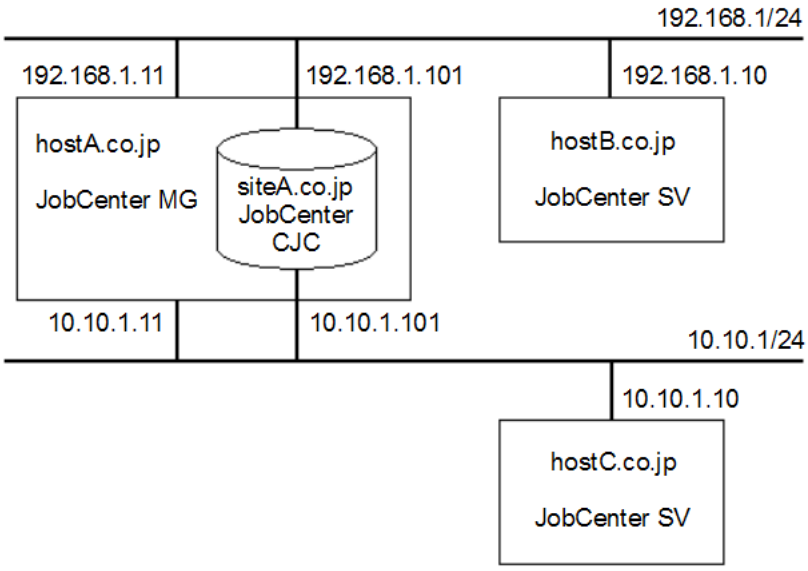

図5.1 クラスタ構成/複数NIC環境構築例

■192.168.1/24へのTCP/IP通信では、source IPとしてサーバの実IPアドレス 192.168.1.11 がOSにより優先される

■10.10.1/24へのTCP/IP通信では、source IPとしてサーバの実IPアドレス 10.10.1.11 がOSにより優先される

■サーバ(hostA)は192.168.1/24と10.10.1/24間のIPフォワーディングを行わない

■SVであるhostB、hostCともに、siteA.co.jpをMGとしてマネージャフレームのマシンアイコンに登録する(連携するJobCenter MG/SV間では、同一サイトは共通する同一マシン名・マシンIDで登録して認識する必要があるため)

ホストマシン	hosts(ネットワーク)設定	daemon.conf設定
hostA (JobCenter MG　をクラスタサイトsiteA.co.jp で運用)	hostA.co.jp　192.168.1.11 hostB.co.jp　192.168.1.10 hostC.co.jp　10.10.1.10 hostA10.co.jp　10.10.1.11 siteA.co.jp　192.168.1.101 * siteA10.co.jp　10.10.1.101 * (* は仮想IPです)	(siteA.co.jp の daemon.conf) ipaddress= 192.168.1.101,10.10.1.101 wildcard=ON
hostsB (JobCenter SV)	hostB.co.jp　192.168.1.10 hostA.co.jp　192.168.1.11 siteA.co.jp　192.168.1.101	ipcheck=OFF
hostsC (JobCenter SV)	hostB.co.jp　10.10.1.10 hostA.co.jp　10.10.1.11 siteA.co.jp　10.10.1.101	ipcheck=OFF

3. クラスタサイトsiteA.co.jpはwildcard=ONに設定しており、かつhostB、hostCではipcheck=OFFに設定していてNQSパケットのIPアドレス逆引きチェックを行いませんので、各SVとも下記の状況で問題なく単位ジョブの投入を受け入れます。

通信方向	操作内容	TCP/IPパケットのIPアドレス	
		source IP	destination IP
siteA → hostB	マシン連携　マシンアイコンの登録 キューの参照 ユーザマッピングの設定 etc.. NQSの単位ジョブリモート投入	192.168.1.11	192.168.1.10
siteA ← hostB	単位ジョブの実行結果を返却	192.168.1.10	192.168.1.101
siteA → hostC	マシン連携　マシンアイコンの登録 キューの参照 ユーザマッピングの設定 etc.. NQSの単位ジョブリモート投入	10.10.1.11	10.10.1.10
siteA ← hostC	単位ジョブの実行結果を返却	10.10.1.10	10.10.1.101

4. もしクラスタサイトsiteA.co.jpのdaemon.confでwildcard=OFFに設定していると、次のようにIPアドレスが割り当てられるため、hostCにはNQSの単位ジョブリモート投入が正常に行えなくなります。

通信方向	操作内容	TCP/IPパケットのIPアドレス	
		source IP	destination IP
siteA → hostC	マシン連携　マシンアイコンの登録 キューの参照 ユーザマッピングの設定 etc.. NQSの単位ジョブリモート投入	192.168.1.101 (hostCのhosts設定の中に無いため、TCP/IPレベル	10.10.1.10

		で通信できません)	

5. 4.に加えて、hostCのhosts設定のうちsiteAのIPアドレスを次のように変更した場合、サーバ(hostA)がIPフォワーディングを行わないため、siteA.co.jp←hostC.co.jpのTCP/IP通信による単位ジョブ実行状況や実行結果の返却ができなくなります。

ホストマシン	hosts(ネットワーク)設定	daemon.conf設定
hostsC (JobCenter SV)	hostC.co.jp 10.10.1.10 hostA.co.jp 10.10.1.11 siteA.co.jp 192.168.1.101 (10.10.1/24側のI/Fからは、 192.168.1.101宛てのパケットがフォワーディングされません)	ipcheck=OFF

6. 4. 5.に加えて、hostCのOSのルーティングテーブルに次のようにエントリを追加すれば、siteA側が複数IPアドレスで待ち受けていますので、通信が可能になります。(以下はWindows OSの場合の設定例です)

```
# route add 192.168.1.101 mask 255.255.255.255 10.10.1.101 ↵
```

ただしhostCのネットワーク設定にこれらの設定変更を行うことで他のPPに影響がないか、良く検討する必要があります。

5.3.2. IPv6アドレスで通信を待ち受ける場合のdaemon.conf設定について

R14.1よりJobCenterではIPv6アドレスを使用して通信を待ち受けできるようになりました。IPv6に対応したOSにつきましては、<リリースメモ>の3章 「動作環境」 を参照してください。

5.3.2.1. JobCenterのIPv6運用環境の前提条件

JobCenterのIPv6運用環境としては、以下の前提条件を全て満たす必要があります。

■使用可能なIPv6アドレスの範囲

アドレスタイプ	ユニキャストアドレス
スコープ	グローバルアドレス

リンクローカルアドレス(例:fe80::20c:29ff:fe8e:d4b)や特殊アドレス(IPv4互換アドレス、IPv4射影アドレス、マルチキャストアドレス、エニーキャストアドレス等)での通信はサポートしていません。

■名前解決

ホスト名からIPv6アドレスの正引き・逆引きが出来る必要があります。

■Windows版の制限

Windows版ではIPv6アドレスのみの環境はサポートしていません。必ずIPv4アドレスが必要となります。

同じホスト名でIPv4アドレス,IPv6アドレスの正引き・逆引きが出来る必要があります。

resolv.defを利用した名前解決を行う場合、resolv.defにIPv4アドレス、IPv6アドレスの両方で同じホスト名から名前解決が出来る様に記述してください。

■マシン間の通信

IPv4からIPv6へ、IPv6からIPv4への通信はサポートしていません。IPv4からIPv4へ、IPv6からIPv6への通信のみサポートしています。

5.3.2.2. IPv6アドレスのみで通信を待ち受ける場合

daemon.confにパラメータのipaddressまたは、bindmodeのどちらかを次のように設定する事により、IPv6アドレスのみで通信を待ち受けることができます。

> Windows版ではIPv6アドレスのみの環境はサポートしていません。必ずIPv4アドレスが必要となります。

1. ipaddress (UNIXのみ)

   複数のIPv6アドレスを使用して通信を待ち受けたい場合、パラメータipaddressに次の形式(複数のIPアドレスをカンマ（,）区切りで記述)で記述し、JobCenterを再起動してください。詳細は「5.2.1 Windows/UNIX共通」のipaddressを参照してください。

```
ipaddress=IPv6アドレス1,IPv6アドレス2,IPv6アドレス3,IPv6アドレス4,IPv6アドレス5
```

2. bindmode (UNIXのみ)

   名前解決で得られたIPv6アドレスを使用して通信を待ち受けたい場合、パラメータbindmodeを次のように記述し、JobCenterを再起動してください。詳細は「5.2.1 Windows/UNIX共通」のbindmodeを参照してください。

```
bindmode=ipv6
```

5.3.2.3. IPv4アドレス・IPv6アドレスの両方で通信を待ち受ける場合

daemon.confにパラメータのipaddressまたは、bindmodeのどちらかを次のように設定する事により、IPv4アドレス・IPv6アドレスの両方で通信を待ち受けることができます。

> Windows版でローカルサイトとクラスタサイトを同じマシンで共存させる場合のローカルサイト側のdaemon.confの設定の場合には、必ずパラメータのipaddressで設定を行ってください。bindmodeで設定した場合には、クラスタサイト側と正しく共存できません。

1. ipaddress (Windows/UNIX共通)

   複数のIPv4アドレス・IPv6アドレスを使用して通信を待ち受けたい場合、パラメータipaddressに次の形式(複数のIPアドレスをカンマ（,）区切りで記述)で記述し、JobCenterを再起動してください。詳細は「5.2.1 Windows/UNIX共通」のipaddressを参照してください。

```
ipaddress=IPv4アドレス1,IPv4アドレス2,IPv4アドレス3,IPv4アドレス4,IPv4アドレス5,IPv6アドレス
1,IPv6アドレス2,IPv6アドレス3,IPv6アドレス4,IPv6アドレス5
```

2. bindmode (Windows/UNIX共通)

   名前解決で得られたIPv4アドレス・IPv6アドレスを使用して通信を待ち受けたい場合、パラメータbindmodeを次のように記述し、JobCenterを再起動してください。詳細は「5.2.1 Windows/UNIX共通」のbindmodeを参照してください。

```
bindmode=all
```

> 同じホスト名でIPv4アドレス,IPv6アドレスの正引き・逆引きが出来る必要があります。

5.4. デーモン設定ファイルのフォーマット

デーモン設定ファイルは、次のフォーマットを参考にして作成してください。

```
#
# Daemon configration file
#
# Copyright 1998 NEC Corporation.
#
#



#
# maintenance
# ------------
#
# Daemon process contorl for SITE environment.
#
# ON    No execute daemon
# OFF   execute daemon (default)
#
#
# Sample:
# maintenance=ON



#
# local_daemon
# -------------
#
# Local daemon mode option.
#
# COMPAT Execute local daemon
#       (compatible mode, exclusive with SITE)
#       (default)
# SITE  Execute local daemon (co-work with SITE)
# OFF   No execute local daemon
#
#
# Sample:
# local_daemon=SITE



#
# NQSDAEMON_OPT
# --------------
#
# nqsdaemon process option.
#
# -x errmail=ON|OFF     Sending mail to job owner at error.
#                       (default ON)
# -x moerr=ON|OFF       Making error status to job when
#                       fails to copy job output files.
#                       (default ON)
# -x reqsig=ON|OFF      Set SIGTERM handler to SIG_DEF.
#                       Specified OFF, set to SIG_IGN.
```

```
#                       (default ON)
# -x ofauth=COMPAT|RESTRICT|LOOSE|JNW
#                       Policy of receiving job output file.
#                       COMPAT          all file receive.
#                       RESTRICT        Receive the file if job
#                                       submittable each other.
#                       LOOSE           Root can not receive the file,
#                                       without standaer JNW output path.
#                       JNW             Everyone can not receive the file,
#                                       without standaer JNW output path.
#                       (default COMPAT)
# -x schevt=ON|OFF      If set "ON" option,  Send request event
#                        to scheduler machine(UMS).
#                       (default OFF)
# -x macgroup_check=<interval>
#                       Set the machine group status check interval.
#                       Valid range is between 300 seconds and 86400 seconds.
#                       Default value is 600 seconds.
# -x shell_uselocallang=ON|OFF
#                       Use local lang when run shell.
#                       (default OFF)
# -x stage_retry_failed_action=copytotempdir|delete
#                       When stage retry failed, whether copy ".o" and ".e" files to
#                       the the home directory of execute user.
#                       (default copytotempdir)
#
#
# Sample:
# NQSDAEMON_OPT=-x errmail=OFF,ofauth=JNW,macgroup_check=600,shell_uselocallang=OFF,
stage_retry_failed_action=copytotempdir

#
# JNWENGINE_OPT
# --------------
#
# jnwengine process option.
#
# -os                   (Compatibility)
#                       Job stop with error and skipped.
#                       The status is error.
# -ot                   (Compatibility)
#                       When a wait parts reashes time-limit,
#                       The status is done.
# -s                    No scan JNW entries.
#
# Sample:
# JNWENGINE_OPT=-os


#
# NOTE:
# =====
#
# These options are over written by each local specification file:
#
#  /usr/spool/nqs/daemon.conf
```

```
#       or
#  /usr/spool/nqs/XXXXXXXX/daemon.conf
#
```

デーモン設定ファイルについては<NQS機能利用の手引き>の「7.3　デーモン起動オプション」にも記載されていますので、参照してください。

5.5. JobCenterの起動時ライセンスチェックについて

OS起動時に、ネットワークI/Fドライバの読み込みタイミングにより、ＩＰ/FへのIPアドレス割当てタイミングよりも、JobCenterサービス起動の方が早くなる場合があります。その際、JobCenterの通信時に使用するIPアドレスが確保できず、JobCenterが自動起動できません。

以下の設定を行うことで、この現象を回避することができます。以下の手順に従って設定を行ってください。

1. 管理者アカウントでWindowsにログオンし、Windowsの［スタート］－［プログラム］－［JobCenter］－［SV］－［サーバの環境設定］を選択してサーバの環境設定を起動します。

2. 左ペインのプロパティを選択し、総合設定をダブルクリックして、総合設定のプロパティを表示します。

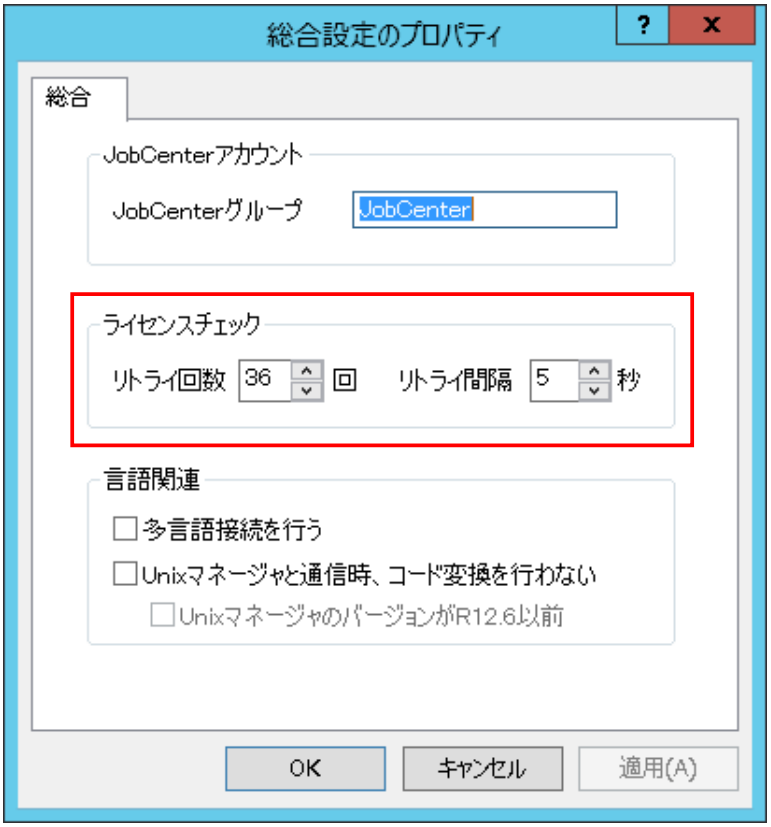

図5.2 ライセンスチェック設定画面

3. ライセンスチェックの中から、以下のパラメータを変更してください。

リトライ回数(回)	0-825 の間で調整
リトライ間隔(秒)	0-90 の間で調整

サイト設定ファイル（site.conf）のBOOT_TIMEOUT設定でサイトの起動制御を行っている場合は、その起動にかける時間制限を越えてライセンスチェックのリトライを行いません。BOOT_TIMEOUT時間の範囲内でライセンスチェックをリトライするよう注意してください。

サイト設定ファイル（site.conf）の詳細については「5.6.3 サイト設定ファイル（site.conf）を作成する」を参照してください。

5.6. Windows版JobCenterの動作設定について

Windows版JobCenter MG/SVでは以下の設定により、動作を変更することが可能です。

5.6.1. ジョブ実行時にユーザプロファイルを読み込む

ジョブの実行設定を変更することで切り替えることができます。詳細は「13.3.3 ジョブの実行設定」を参照してください。

5.6.2. ログオン動作のタイプを変更する

Windows Server 2008以降をプラットフォームとした場合、JobCenterはジョブとして記述されたスクリプトをBATCHログオンタイプで実行していきます。

スクリプトに記述したWindowsコマンドの実行を、対話的(INTERACTIVE)ログオンタイプに切り替える場合には、以下の設定を行ってください。

1. レジストリエディタを開き、以下のキー配下を開く

HKEY_LOCAL_MACHINE\Software\NEC\NetShepherd/SV

2. 新規に以下の値を作成する

名称	LogonUserType
種類	文字列値
値	I ※値はアルファベット「アイ」の大文字です。

3. JobCenterを再起動します。

上記の設定を行った場合には、UAC機能が有効である環境かつJobCenter管理者ユーザがadministratorユーザではない環境という条件の下でJobCenter管理者ユーザでジョブを実行すると、JNW内の実行権限がOSレベルでは一般ユーザになってしまうという制限事項が生じます。

5.6.3. サイト設定ファイル（site.conf）を作成する

次のパスにsite.confというファイルを作成することによってローカルサイト、クラスタサイトの起動・停止処理にかける時間を制御できます。

ローカルサイト	<JobCenterインストールディレクトリ>\etc\site.conf
クラスタサイト	<サイトデータベースへのパス>\etc\site.conf

サイトの起動(再起動)前に実行中のジョブが多数存在した場合や、サイト停止時に実行中のジョブが多数存在するような場合に、サイトの起動・停止に時間がかかる場合があります。

また、サイトの停止コマンドが実行されると、ジョブプロセスの監視を行うプロセス（ローカルサイトの場合はjcservice、クラスタサイトの場合はcjcpw）は、JobCenterの各プロセスの適切な終了処理を行ってから自身を終了するようにします（正常終了）。

なんらかの障害が発生し正常終了ができない場合は、監視を行うプロセスはJobCenterの各プロセスを強制停止させます（強制終了）。

site.confへパラメータの設定を行うことによって、正常終了、強制終了にかける時間を制御できます。

このファイルのフォーマット、および設定可能な項目は以下の通りです。

```
BOOT_TIMEOUT=<値>
STOP_TIMEOUT=<値>
FORCEKILL_TIME=<値>
```

行の先頭や「=」の前後に余分な空白やタブを入れないようにしてください。

■BOOT_TIMEOUT

サイト起動処理のタイムアウト値を秒単位で指定します。この設定値を超えてもサイトを起動できなかった場合は、サイトの起動を中断します。

■STOP_TIMEOUT

サイト停止処理（正常終了にかける時間と強制終了にかける時間の和）のタイムアウト値を秒単位で指定します。この設定値を超えてもサイトを停止できなかった場合は、サイトの停止処理を中断します。

■FORCEKILL_TIME

サイトの強制終了処理にかける時間を秒単位で指定します。

STOP_TIMEOUTの設定値はFORCEKILL_TIMEの設定値より大きくなるようにしてください。

STOP_TIMEOUTの設定値がFORCEKILL_TIMEの設定値より小さい場合、サイト停止コマンドを実行するとJobCenterは各プロセスを強制停止する形で終了処理を行います（停止にかける時間はFORCEKILL_TIMEの値となります）。そのような指定を行うと正常にサイトの停止処理が行われなくなります。

各パラメータのデフォルト値、最大・最小値は以下の通りです（設定値の単位は秒）。site.confを作成しない、またはパラメータ未設定の場合はデフォルト値が設定されます。不正な値(負の数や文字列等)を指定した場合は、最小値に丸められます。

表5.1 site.confのパラメータの設定範囲

設定項目名	デフォルト値	最小値	最大値
BOOT_TIMEOUT	180	30	999999
STOP_TIMEOUT	180	30	999999
FORCEKILL_TIME	10	1	999999

各パラメータの最大値は999999となりますが、あまりに大きな値を指定すると、サイトの起動、停止処理に失敗した場合にJobCenterが長時間動作しなくなることになります。

5.7. jcwebserverの動作設定について

5.7.1. jcwebserver設定ファイル（jcwebserver.conf）の格納場所

jcwebserver設定ファイル（jcwebserver.conf）の格納場所は、環境により異なります。

5.7.1.1. Windowsの場合

jcwebserver設定ファイルを作成する場合は、次の場所に格納します。

JobCenter起動時にローカル環境・クラスタ環境それぞれのサイトごとに設置されたファイルを読み込みます。

ローカルサイト	%InstallDirectory%\etcディレクトリ配下
クラスタサイト	%JobCenterDatabaseDirectory%\etcディレクトリ配下

上記の %InstallDirectory% は JobCenter のインストールディレクトリです。

デフォルトは C:\JobCenter\SV になります。

また %JobCenterDatabaseDirectory% は、cjcmksite でクラスタサイトを構築する際に指定した共有ディスク上のデータベースディレ>クトリです。

5.7.1.2. Linuxの場合

jcwebserver設定ファイルを作成する場合は、次の場所に格納します。

JobCenter起動時に、まずデフォルトで存在する共通ファイル、次に各サイトごとに設置されたファイルがあればそれを読み込みます。後に読み込まれるファイルのパラメータ記述内容が優先されます。

共通	/usr/lib/nqs/rc/jcwebserver.conf (ローカルサイト、クラスタサイトの各jcwebserver設定ファイルよりも先に、デフォルトで存在する本ファイルが読み込まれます)
ローカルサイト	/usr/spool/nqs/jcwebserver.conf
クラスタサイト	<JobCenterDBパス>/nqs/jcwebserver.conf

HP-UX及びAIX環境では、jcwebserver設定ファイル（jcwebserver.conf）はサポートしていません。

5.7.2. jcwebserver設定ファイルの使用可能パラメータ

jcwebserver設定ファイル（jcwebserver.conf）の書式はYAML形式であり、設定できるパラメータを以下の一覧表に示します。

表5.2 jcwebserver設定ファイルのパラメータ一覧

パラメータ	説明
debug	デバッグモードの有効/無効を設定します。 本パラメータは必ず設定する必要があります。 ■true デバッグモードは有効となります。 ■false

パラメータ		説明
		デバッグモードは無効となります。
timeout	apiExecution	JobCenter MG/SVのWebAPIの処理時間のタイムアウト値(秒)を設定します。 本パラメータは必ず設定する必要があります。 タイムアウト値(秒)は 0～3600 の範囲で設定できます。
	readRequest	リクエストの読み込み開始から完了までの処理時間のタイムアウト値(秒)を設定します。 本パラメータは必ず設定する必要があります。 タイムアウト値(秒)は 0～3600 の範囲で設定できます。
	readRequestHeader	リクエストヘッダの読み込み開始から完了までの処理時間のタイムアウト値(秒)を設定します。 本パラメータは必ず設定する必要があります。 タイムアウト値(秒)は 0～3600 の範囲で設定できます。
	writeResponse	リクエストのパラメータの取得開始からレスポンスの書き込み完了までの処理時間のタイムアウト値(秒)を設定します。 本パラメータは必ず設定する必要があります。 タイムアウト値(秒)は 0～3600 の範囲で設定できます。
	keepAlive	リクエストの結果を返した後にkeepAliveに設定された時間(秒)、コネクションを維持して次のリクエストのコネクションとして再利用します。 コネクションを再利用するにはリクエストのHeaderに "Connection: Keep-Alive" の設定が必要です。 keepAliveに設定された時間(秒)以内に新しいリクエストが来なかった場合にはコネクションは終了します。 本パラメータは必ず設定する必要があります。 タイムアウト値(秒)は 0～3600 の範囲で設定できます。
tls		tlsを設定した場合にはHTTPSで通信します。また、tlsを設定した場合、http2,certificate,privateKeyのパラメータは必ず設定する必要があります。 tlsが未設定の場合にはHTTPで通信(HTTP/1.1のプロトコル)します。
	http2	HTTP/2.0の通信プロトコルを利用するかしないかを設定します。 ■true HTTP/2.0を利用します。 ■false HTTP/2.0を利用しません（HTTP/1.1を利用します）。

パラメータ			説明
			tlsを設定した場合、本パラメータも必ず設定する必要があります。
	certificate		用意したサーバ側の　TLS　証明書ファイルのパスを設定します。 tlsを設定した場合、本パラメータも必ず設定する必要があります。
	privateKey		用意したサーバ側の　TLS　秘密鍵ファイルのパスを設定します。 サポートしている秘密鍵のアルゴリズムは以下となります。 ■RSA ■ECDSA ■EDDSA(ED25519) tlsを設定した場合、本パラメータも必ず設定する必要があります。
log	serverLog	maxSize	サーバログファイルの 1 ファイルの最大サイズ(MB)を設定します。 本パラメータは必ず設定する必要があります。 最大サイズ(MB)は 4～100 の範囲で設定できます。
		maxBackups	サーバログファイルのバックアップ数を設定します。 本パラメータは必ず設定する必要があります。 バックアップ数は 1～10 の範囲で設定できます。 設定したバックアップ数よりも古いログファイルは自動的に削除されます。
	accessLog	maxSize	アクセスログファイルの 1 ファイルの最大サイズ(MB)を設定します。 本パラメータは必ず設定する必要があります。 最大サイズ(MB)は 4～100 の範囲で設定できます。
		maxBackups	アクセスログファイルのバックアップ数を設定します。 本パラメータは必ず設定する必要があります。 バックアップ数は 1～10 の範囲で設定できます。 設定したバックアップ数よりも古いログファイルは自動的に削除されます。
	errorLog	maxSize	エラーログファイルの 1 ファイルの最大サイズ(MB)を設定します。 本パラメータは必ず設定する必要があります。 最大サイズ(MB)は 4～100 の範囲で設定できます。
		maxBackups	エラーログファイルのバックアップ数を設定します。

パラメータ			説明
			本パラメータは必ず設定する必要があります。 バックアップ数は 1～10 の範囲で設定できます。 設定したバックアップ数よりも古いログファイルは自動的に削除されます。

パラメータの設定例）

```
debug: false
timeout:
  apiExecution: 300
  readRequest: 20
  readRequestHeader: 10
  writeResponse: 120
  keepAlive: 30
tls:
  http2: true
  certificate: "/usr/share/ssl/ssl_cert"
  privateKey: "/usr/share/ssl/ssl_key"
log:
  serverLog:
    maxSize: 4
    maxBackups: 10
  accessLog:
    maxSize: 4
    maxBackups: 10
  errorLog:
    maxSize: 4
    maxBackups: 10
```

■証明書ファイルと秘密鍵ファイルは、情報採取コマンド(jc_getinfo)にて採取されるディレクトリ以外の場所に配置してください。

証明書ファイルと秘密鍵ファイルを以下のディレクトリに配置した場合には、情報採取コマンド(jc_getinfo)にて採取されます。

- UNIX

  /usr/lib/nqs/rc/

情報採取コマンド(jc_getinfo)の詳細については<コマンドリファレンス>の「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」を参照してください。

6. JobCenter部品パラメータおよび環境の設定変更

ジョブネットワーク、単位ジョブ、ERPジョブ、BIジョブ、PCジョブ、WOBSジョブ、ダイアログ、イベント送信、イベント受信の各JobCenter部品を新規に設定する際の、パラメータのデフォルト値が定義できます。これにより、同じパラメータを設定する部品を複数作成する場合に個々のパラメータの再設定が不要になります。

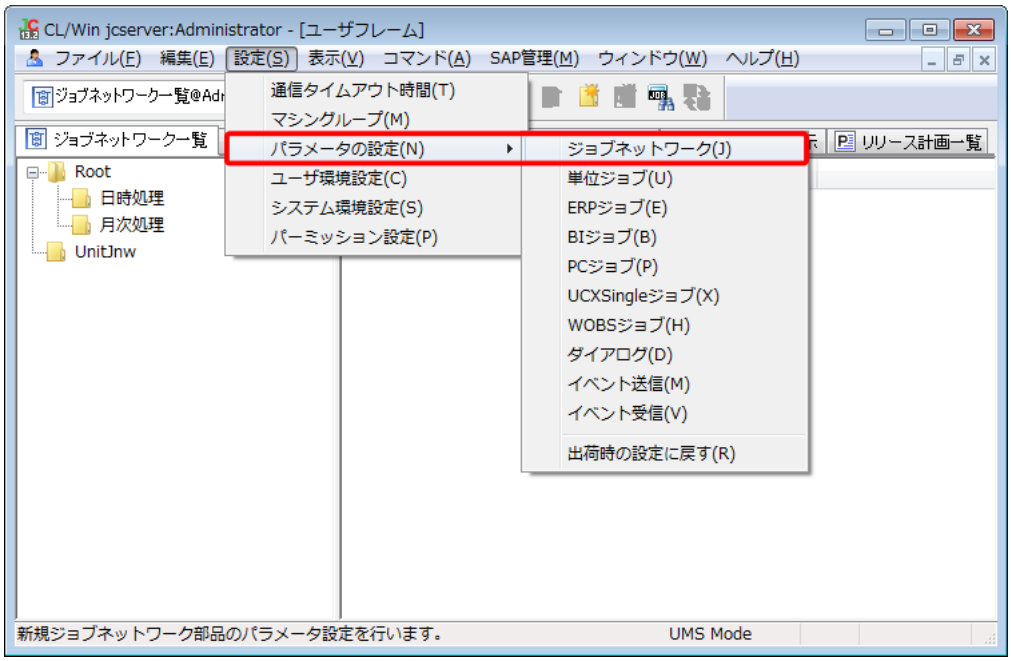

図6.1 ［パラメータの設定］選択画面例

また、現在設定されている実行環境、システム環境および色の設定のパラメータの参照および変更ができます。

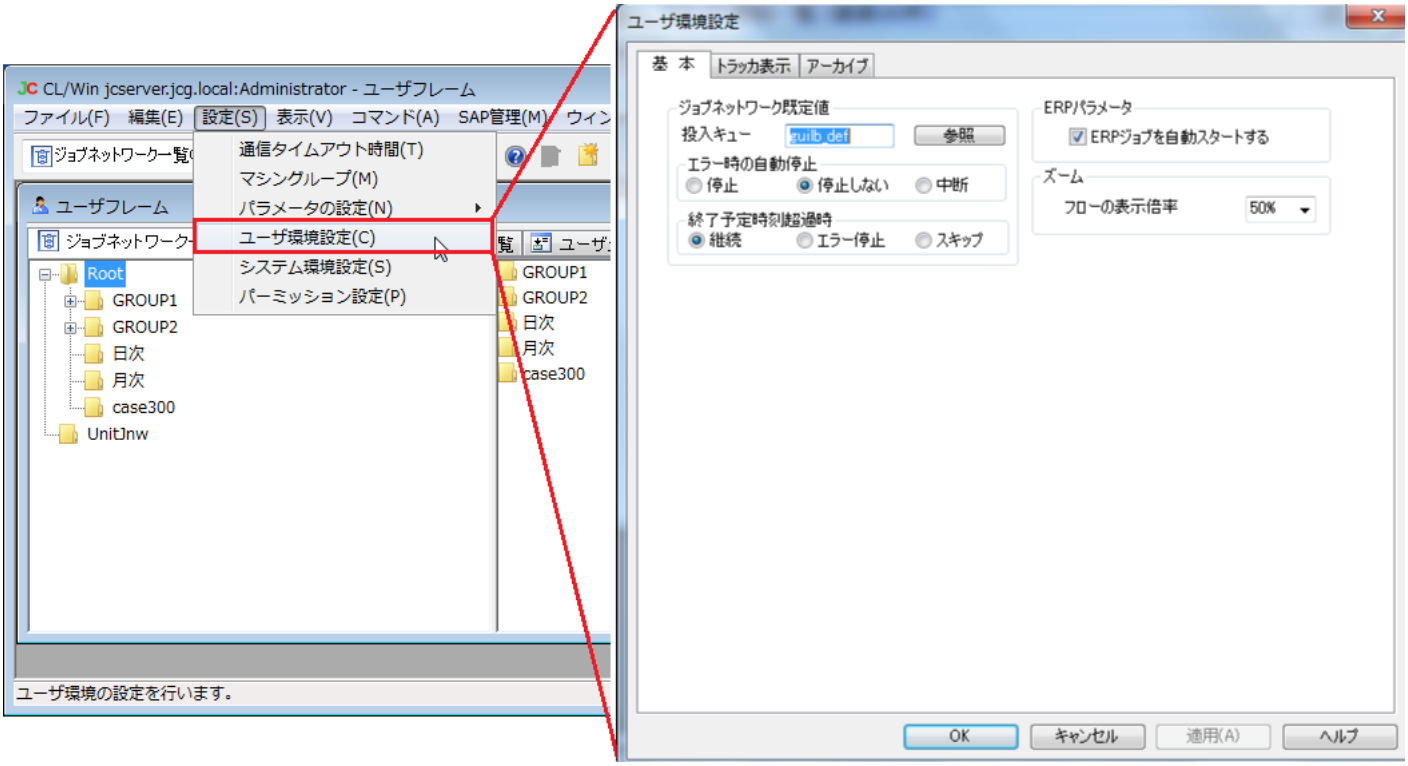

図6.2 ［ユーザ環境設定］選択画面例

6.1. JobCenter部品のデフォルトパラメータの設定を定義する

メニューバーの［設定］－［パラメータの設定］から、ジョブネットワーク、単位ジョブ、ERPジョブ、BIジョブ、PCジョブ、WOBSジョブ、ダイアログ、イベント送信、イベント受信部品を作成するときのパラメータのデフォルト値の定義が行えます。

また、JobCenter CL/Winが持っている出荷時の値（既定値）で各部品のデフォルトパラメータの更新ができます。

■他ユーザのジョブネットワークおよび各部品を作成するときのパラメータのデフォルト値は、ログインユーザではなくそのユーザのパラメータデフォルト値が使用されます。

パラメータのデフォルト値は、ジョブネットワークを所有するユーザ（アクセス先のユーザ）自身でログインしてから設定してください。

■パラメータのデフォルト値の設定は、ジョブネットワークの作成／変更／削除のアクセス権限が必要です。

6.1.1. ジョブネットワーク

［ジョブネットワークのデフォルトパラメータ］ダイアログで、ジョブネットワークのパラメータデフォルト値を設定します。

ジョブネットワークのパラメータ設定については<基本操作ガイド>の「3.3.4　ジョブネットワークのパラメータを設定する」を参照してください。

6.1.2. 単位ジョブ

［単位ジョブのデフォルトパラメータ］ダイアログで、単位ジョブのパラメータデフォルト値を設定します。

単位ジョブのパラメータ設定については<基本操作ガイド>の「4.2.2　単位ジョブの実行条件（パラメータ）を設定する」を参照してください。

6.1.3. ERPジョブ

［ERPジョブのデフォルトパラメータ］ダイアログで、ERPジョブのパラメータデフォルト値を設定します。

ERPジョブのパラメータ設定については <SAP機能利用の手引き>の「1.2.1.4 SAP ERPジョブのパラメータ設定」を参照してください。

6.1.4. BIジョブ

［BIジョブのデフォルトパラメータ］ダイアログで、SAP BIジョブのパラメータデフォルト値を設定します。

SAP BIジョブのパラメータ設定については <SAP機能利用の手引き>の「2.2.1.2 SAP BIジョブのパラメータ設定」を参照してください。

6.1.5. PCジョブ

［PCジョブのデフォルトパラメータ］ダイアログで、SAP PCジョブのパラメータデフォルト値を設定します。

SAP PCジョブのパラメータ設定については <SAP機能利用の手引き>の「2.3.1.2 SAP PCジョブのパラメータ設定」を参照してください。

6.1.6. WOBSジョブ

［WOBSジョブのデフォルトパラメータ］ダイアログで、WOBSジョブのパラメータデフォルト値を設定します。

WOBSジョブのパラメータ設定については <WebOTX Batch Server連携機能利用の手引き>の「3.1 パラメータ一覧」を参照してください。

6.1.7. ダイアログ

［ダイアログのデフォルトパラメータ］ダイアログで、ダイアログのパラメータデフォルト値を設定します。

ダイアログのパラメータ設定については<基本操作ガイド>の「4.4.1　ダイアログの設定をする」を参照してください。

6.1.8. イベント送信

［イベント送信のデフォルトパラメータ］ダイアログで、イベント送信のパラメータデフォルト値を設定します。

イベント送信のパラメータ設定については<基本操作ガイド>の「4.5.1　イベント送信の設定をする」を参照してください。

6.1.9. イベント受信

［イベント受信のデフォルトパラメータ］ダイアログで、イベント受信のパラメータデフォルト値を設定します。

イベント受信のパラメータ設定については<基本操作ガイド>の「4.5.2　イベント受信の設定をする」を参照してください。

6.1.10. 既定値をロード

JobCenter CL/Winが持っている出荷時の値で各デフォルトパラメータを更新します。

6.2. ユーザ環境の設定を変更する

［ユーザ環境設定］ダイアログでは、現在設定されているユーザ実行環境のパラメータの参照および変更ができます。

各設定項目は基本的にサーバに保存され、ログインするユーザ毎の設定となります。

> ユーザ環境の設定はコマンドでも変更可能です。詳細については、<コマンドリファレンス>の「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」のsetcfgサブコマンドの説明を参照してください。

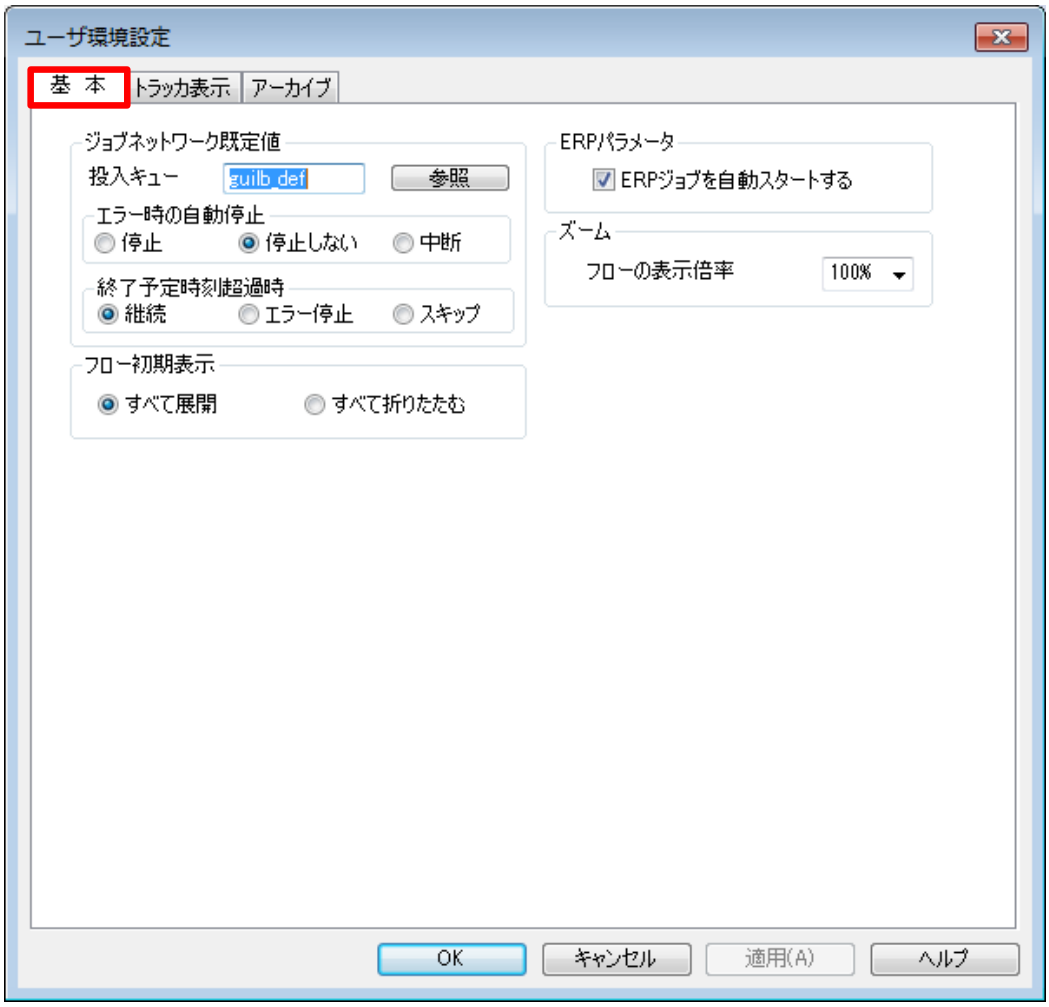

図6.3　［ユーザ環境設定］画面例

パラメータの変更は、修正値の入力後に［適用］または［OK］ボタンをクリックします。修正値を誤って入力した場合は［キャンセル］ボタンをクリックします。

［基本］タブの設定項目は次のとおりです。

［ジョブネットワーク既定値］

実行するジョブネットワークの既定値を指定します。

■投入キュー

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指定します。

ジョブネットワークや単位ジョブのパラメータで投入キューを指定しなかった場合（投入キューの項目が空欄の場合）は、ここで設定したキューを自動的に選択してジョブが投入されます。

［参照］ボタンをクリックして、ポップアップされる［投入先キューの設定］ダイアログからキューを選択します。

なお、JobCenterのセットアップ時にデフォルトで作成されるキューについては、guilb_defのみ選択可能となります。

図6.4 キュー選択画面

■エラー時の自動停止

エラー発生時のジョブネットワークのデフォルトの動作を設定します。

各ジョブネットワークのパラメータで［既定］を選択した場合は、この値が参照されます。

停止	エラー発生時にジョブネットワークを停止します。
停止しない	エラー発生時にジョブネットワークを停止しません。
中断	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。

■終了予定時刻超過時

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します。

ジョブネットワークパラメータの終了予定超過時に［既定］が設定されている場合に、この設定が有効になります。

継続	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。
エラー停止	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。
スキップ	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。

現在の設定は、次の方法で確認します。

1. ユーザフレームの［ジョブネットワーク一覧］で参照したいジョブネットワークを選択します。

2. 右クリックしたときのポップアップメニューから［パラメータ］を選択すると、［パラメータ］の［基本設定］画面が表示されます。

3. ［終了予定時刻超過時］欄を確認します。

■フロー初期表示

ジョブネットワークフロー画面およびトラッカフロー画面を表示した際の、分岐部品の折りたたみの初期状態を設定します。

すべて展開	すべての分岐部品を展開した状態で表示します。
すべて折りたたむ	すべての分岐部品を折りたたんだ状態で表示します。

［ERPパラメータ］

ERPオプションに関して指定します。

■ERPジョブを自動スタートする

ERPジョブは投入されると投入状態になります。

チェックされている場合	JobCenterにより自動的に実行状態になります。
チェックされていない場合	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。

なお、ERPジョブは次の方法で実行（リリース）します。

1. トラッカフロー上のERPジョブ選択時のポップアップメニュー［実行（IMMEDIATELY/ASAP）］を選択します。
2. ［トラッカ操作の確認］画面が表示されますので実行ホスト名（インスタンス名付き）を入力し、［OK］ボタンをクリックします。
3. 操作確認のダイアログが表示されますので［OK］ボタンをクリックします。
4. 対象のERPジョブが実行（RUN）状態になります。

［ズーム］

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。

フローの表示倍率は10～100%で設定することができます。

6.3. トラッカ表示の設定を変更する

［ユーザ環境設定］ダイアログの［トラッカ表示］画面では、現在設定されているユーザのトラッカの表示設定の参照および変更ができます。ユーザフレームやマネージャフレームのトラッカ一覧画面の初期表示は、ここで設定した内容に従い表示されます。

［ユーザ環境設定］ダイアログで変更した内容の反映はCL/Winを再接続するか、次に［ユーザフレーム］ウィンドウまたは［マネージャフレーム］ウィンドウを開いた際に有効になります。

各設定項目は基本的にサーバに保存され、ログインするユーザ毎の設定となります。

ユーザ環境の設定はコマンドでも変更可能です。詳細については、<コマンドリファレンス>の「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」のsetcfgサブコマンドの説明を参照してください。

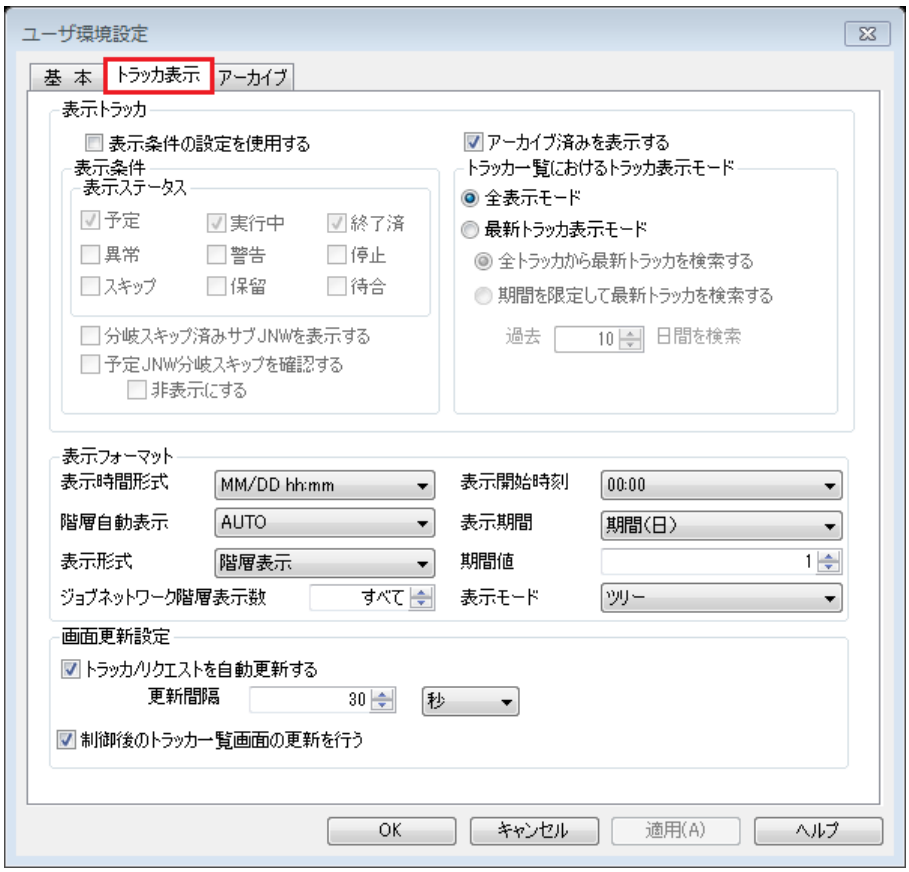

図6.5　［トラッカ表示］画面例

パラメータの変更は、修正値の入力後に［適用］または［OK］ボタンをクリックします。修正値を誤って入力した場合は［キャンセル］ボタンをクリックします。

［トラッカ表示］タブの設定項目は次のとおりです。

［表示トラッカ］

表示するトラッカ一覧に関して指定します。

■表示条件の設定を使用する

［表示条件］の設定を選択するかしないかのいずれかを選択できます。

チェックされている場合	［表示条件］が選択できるようになり、選択した条件に合うトラッカのみが表示されます。
チェックされていない場合	［表示条件］はグレー表示（選択不可）となり、すべての状況のトラッカを表示します。

［表示条件］

表示するトラッカの条件の指定をします。

■表示ステータス

トラッカリストで表示するトラッカのステータスの条件です。次の条件を複数選択できます。

予定	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。
実行中	実行が開始されているトラッカのみを表示します。
終了済	すでに実行が終了しているトラッカのみを表示します。
異常	異常が発生したトラッカのみを表示します。
警告	警告が発生したトラッカのみを表示します。
停止	エラー停止したトラッカのみを表示します。
スキップ	スキップ状態のトラッカのみを表示します。
保留	保留状態のトラッカのみを表示します。
待合	待ち合わせ状態のトラッカのみを表示します。

■分岐スキップ済みサブJNWを表示する

トラッカ一覧において、予定状態以外のトラッカについて実際の処理分岐（カレンダ分岐）を判定のうえ、分岐スキップ済みサブJNWの表示をするかしないかのいずれかを選択できます。

チェックされている場合	実行されないサブJNWを分岐スキップ済みとして表示します。
チェックされていない場合	実行されないサブJNWを表示しません。

この項目の設定値についてはCL/Winが動作するPC側に保存され、接続する端末毎に適用されます。

■予定JNW分岐スキップを確認する

トラッカ一覧において、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選択できます。

判定するとした場合に、実行されないサブJNWについて分岐スキップ済みとして表示するか、もしくは非表示とするかを選択できます。

チェックされている場合	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します。 なお、表示条件でスキップを選択していなければ表示されません。
チェックされていない場合	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。
非表示にする	チェックされている場合には、分岐スキップ済みのサブJNWを非表示とします。

この項目の設定値についてはCL/Winが動作するPC側に保存され、接続する端末毎に適用されます。

■アーカイブ済みを表示する

アーカイブ済みのトラッカを表示するかしないかのいずれかを選択できます。

チェックされている場合	アーカイブ済みのトラッカも表示します。
チェックされていない場合	アーカイブ済みのトラッカは表示しません。

［トラッカ一覧におけるトラッカ表示モード］

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の指定をします。

トラッカ一覧におけるトラッカ表示モードの各設定項目および最新トラッカ表示モードの詳しい説明は、<基本操作ガイド>の「8.3.2 最新トラッカ表示モードを設定する」を参照してください。

［表示フォーマット］

トラッカリストを表示する時の表示形式を指定します。

■表示時間形式

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です。

次のいずれかを選択できます。

hh:mm	時分を表示します。
MM/DD hh:mm	月日時分を表示します。
YYYY/MM/DD hh:mm:ss	年月日時分秒を表示します。

■表示開始時刻

トラッカリストの表示を開始する先頭の時刻をコンボボックスで00:00から1時間単位に23:00まで選択できます。

■階層自動表示

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指定します。次のいずれかを選択できます。

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。

■表示期間

トラッカリストの表示期間のデフォルト値となる期間の単位を設定します。次のいずれかを選択できます。

- 分
- 時間
- 期間（日）

- 期間（週）
- 期間（月）

■表示形式

トラッカの表示形式を選択できます。

階層表示	常にルートジョブネットワークから階層表示します。
絞り込み表示	指定した条件に合うトラッカのみを表示します。

■期間値

トラッカリストの表示期間のデフォルト値を設定します。［表示期間］で選択した単位により、次の値を期間値として選択できます。

表示期間	期間値
分	1～59
時間	1～23
期間（日）	1～7
期間（週）	1～5
期間（月）	1～3

■ジョブネットワーク階層表示数

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指定します。次のいずれかを選択できます。

すべて	すべてのジョブネットワークを表示します。
1~40	1~40階層までのジョブネットワークを表示します。

トラッカの表示形式が「絞り込み表示」の場合、または階層自動表示が「SELECT」あるいは「NON-AUTO」の場合、本設定は無視されます。

■表示モード

トラッカ表示モードのデフォルト値を指定します。次のいずれかを選択できます。

ツリー	トラッカをツリー構造で階層的に表示します。
時間	予定開始時間の昇順に並べ替えて表示します。
ソート	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。

「ソート」モードは、マシンフレームのトラッカ一覧画面および、マネージャフレームのトラッカ一覧@全マシンのみ有効です。他のトラッカ一覧画面では「ツリー」モードで表示します。

［画面更新設定］

実行するトラッカやリクエストに関する指定をします。

■トラッカ/リクエストを自動更新する

トラッカやリクエストに関するウィンドウの更新方法を設定します。

チェックされている場合	一定間隔で表示更新を行います。
チェックされていない場合	自動更新を行いません。

［ユーザ環境設定］ダイアログで変更した内容の反映方法は2通りあります。CL/Winを再接続することで内容が反映されます。もしくは、メニューバーの[ユーザ環境設定]-[画面更新設定]-[トラッカ/リクエストを自動更新する]を一旦チェックを外し、適用ボタンをクリックして一度自動更新を無効化した後に再度チェックをつけることで内容が反映されます。

■更新間隔

トラッカやリクエストに関するウィンドウを定期的に更新する場合の更新間隔を設定します。更新間隔の時間は10秒～3600秒または1分～60分を指定できます。

1. ユーザフレームの［トラッカ一覧］を表示し、更新間隔時間を設定するトラッカを選択します。
2. メニューバーの［表示］－［自動更新］を選択すると、［リアルタイム更新間隔時間の設定］ダイアログが表示されます。10秒～60秒および1分～60分の指定ができ、定期更新されます。

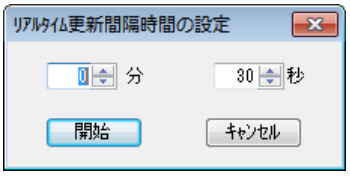

図6.6 リアルタイム更新間隔時間設定画面

■制御後のトラッカ一覧画面の更新を行う

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どうかを指定します。

チェックされている場合	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（既定値）
チェックされていない場合	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。

6.4. アーカイブの設定を変更する

［ユーザ環境設定］ダイアログの［アーカイブ］画面では、現在設定されているユーザのアーカイブの設定の参照および変更ができます。

各設定項目は基本的にサーバに保存され、ログインするユーザ毎の設定となります。

> ユーザ環境の設定はコマンドでも変更可能です。詳細については、<コマンドリファレンス>の「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」のsetcfgサブコマンドの説明を参照してください。

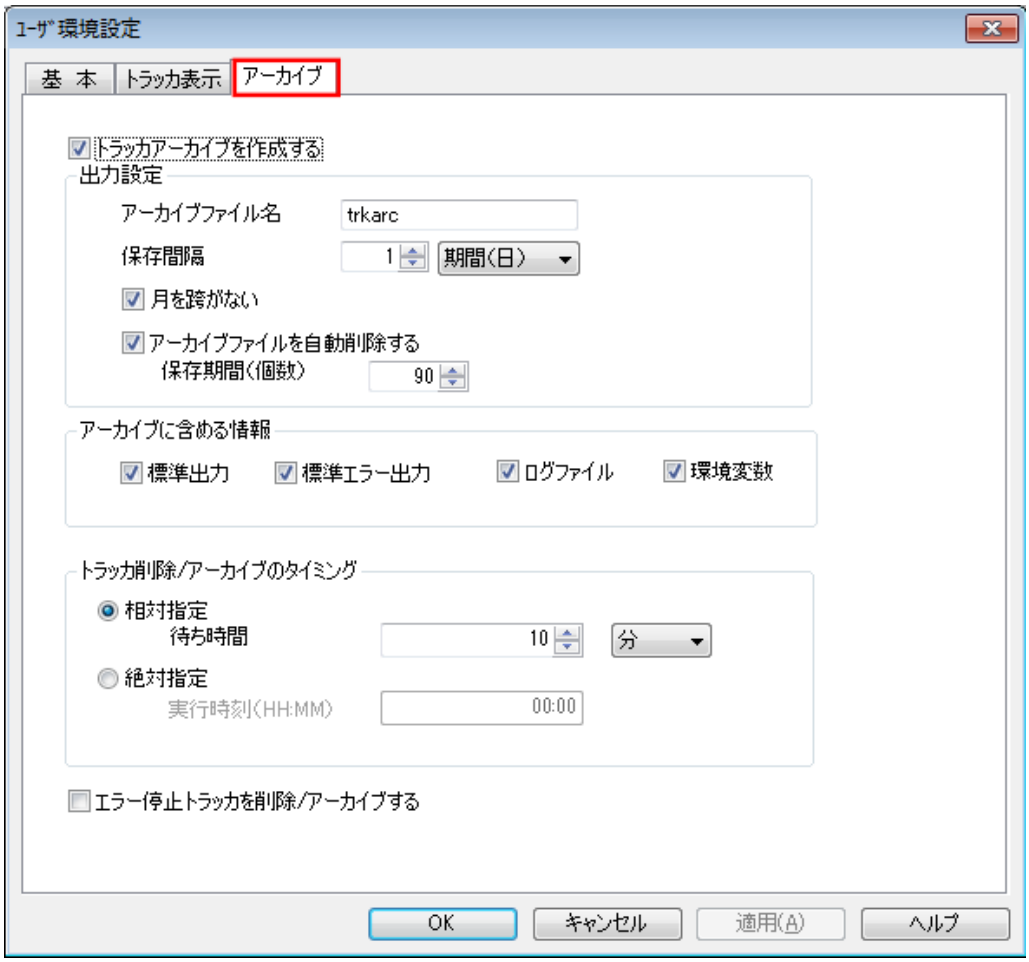

図6.7　［アーカイブ］画面例

パラメータの変更は、修正値の入力後に［適用］または［OK］ボタンをクリックします。修正値を誤って入力した場合は［キャンセル］ボタンをクリックします。

［アーカイブ］タブの設定項目は次のとおりです。

■トラッカアーカイブを作成する

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します。

チェックされている場合	保存します。
チェックされていない場合	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)

[トラッカアーカイブを作成する]のチェックをはずして［適用］すると、その時点で［トラッカ削除/アーカイブのタイミング］に設定されていたアーカイブ待ち時間経過後に、実行終了したトラッカがアーカイブされず直ちに削除されるように動作が変わります。一旦削除されたトラッカを復活することはできませんので注意してください。

［出力設定］

実行されたジョブネットワークのトラッカ情報をアーカイブして保存する条件を指定します。

■アーカイブファイル名

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です。

絶対パスで指定すると、指定したパスにアーカイブファイルを作成します。ファイル名のみを入力した場合には以下の箇所にアーカイブファイルが作成されます。

- Windowsの場合

  %InstallDirectory%\spool\users\<ユーザ名>\archive

- UNIXの場合

  /usr/spool/nqs/users/<ユーザ名>/archive

- %InstallDirectory%はJobCenterのインストールディレクトリです。デフォルトはC:\JobCenter\SVになります。また、クラスタ環境の場合は%InstallDirectory%　　の部分を<JobCenterDBパス>と読み替えてください。
- トラッカアーカイブファイル名には2バイト文字および以下の文字は使用できません。

  「+」「'」「#」「,」「;」「*」「?」「"」「<」「>」「|」

- 絶対パスを指定してアーカイブファイルの保存先を変更すると、それまでのトラッカアーカイブが参照できなくなります。

■保存間隔

アーカイブファイルの保存間隔を設定します。

保存期間、保存間隔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。以下は設定例です。

保存間隔	保存期間（個数）
1（月）	0～12
12（月）	0～1
1（週）	0～53
53（週）	0～1
1（日）	0～365（366）
365（日）	0～1

■月を跨がない

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どうかを設定します。

チェックされている場合	月を跨ぎません。

チェックされていない場合	月を跨ぎます。

（設定例）

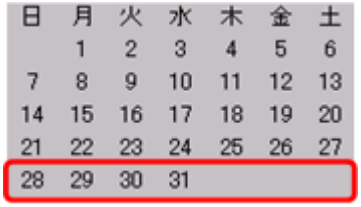

図6.8 カレンダ例

上記のカレンダにおいてチェックされている場合は、28日～31日と、次月の1日～3日までのアーカイブファイルは別のファイルで保存されます。［OFF］の場合は1つのファイルで保存されます。

■アーカイブファイルを自動削除する

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します。

チェックされている場合	自動削除します。
チェックされていない場合	自動削除しません。

■保存期間（個数）

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します。

保存期間（個数）の範囲は0～365です。

■過去一年以上のデータについては手動で保存してください。

■保存期間の個数を増やす、保存間隔を長くするなどにより、アーカイブファイルが大きくなりすぎないように注意してください。

アーカイブファイルが1ファイルにつき2GBを超えると、JobCenterは正常に動作しなくなります。

アーカイブファイルのサイズを計算する際は「23.3.2 トラッカアーカイブのDISK使用容量概算算出方法」(UNIX版)または「23.6.2 トラッカアーカイブのDISK使用容量概算算出方法」(Windows版)を参照してください。

■ただし次の4つの条件を満たしていれば、アーカイブファイルが1ファイルで2GBを超える場合でも動作可能です。

- 1つのトラッカデータのサイズが2GBを超えていない。

  トラッカデータとはログ、出力結果/エラー出力結果などの情報が含まれたデータです。

- OS別のJobCenterのバージョンが以下のいずれか

Windows	R12.5.4以降
HP-UX(IPF)	R12.5.4以降
Linux	R15.1以降

- アーカイブファイルの存在するボリュームがラージファイルをサポートするファイルシステムである
- OSのファイルシステム関連パラメータをラージファイル対応に設定している

ただしユーザやプロセスのファイルサイズの上限が設定されている場合は、その制限によっても最大サイズが規制されます。

■ファイルの最大サイズ超過やディスクの空き容量不足によりトラッカデータをアーカイブファイルに追加する処理が失敗した場合、トラッカデータはアーカイブされません。

アーカイブされなかったトラッカは、アーカイブされた場合と同様に再実行などが行えません。

また、トラッカ一覧の表示もアーカイブ済みと表示されます（これらのトラッカは、JobCenterを再起動することにより、アーカイブ処理が再試行されます）。

ただしJobCenter　R12.5では、アーカイブに失敗したトラッカは失われてアーカイブファイルも不正な状態になる場合があります。

ディスクの空き容量が不足した場合は、アーカイブファイルを削除するか、アーカイブファイルの保存先を変更してください。保存先の変更については「6.4　アーカイブの設定を変更する」のアーカイブファイル名を参照してください。

■保存間隔ごとにアーカイブファイルが切り替わりますが、切り替えはGMTベースで管理しますので、経過日数とアーカイブファイルの個数が一致しない場合があります。

保存期間、保存間隔の考え方

アーカイブファイルの保存期間をアーカイブファイルの保存間隔と個数を指定することにより設定します。

デフォルト値は保存期間（個数）が90、保存間隔が1、保存間隔の単位が期間（日）の推奨値になっています。この場合1日ごとにアーカイブファイルが90個保存されますので、保存期間の合計は90日となります。

（保存期間のイメージ）

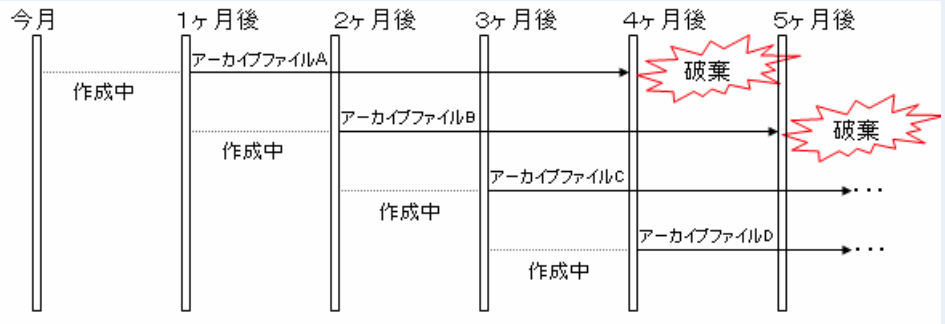

図6.9 保存期間設定画面

上図は、保存期間（個数）を3、保存間隔を1、そして保存間隔の単位を期間（月）として保存期間を3ヶ月にした場合のイメージです。

なお保存間隔の単位を大きくすると、1ファイルあたりのアーカイブファイルサイズが増大しますので、注意してください。

［アーカイブに含める情報］

アーカイブファイルに含めるトラッカの情報を指定します。

■標準出力

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します。

チェックされている場合	保存します。
チェックされていない場合	保存しません。

■標準エラー出力

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します。

チェックされている場合	保存します。
チェックされていない場合	保存しません。

■ログファイル

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します。

チェックされている場合	保存します。
チェックされていない場合	保存しません。

■環境変数

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します。

チェックされている場合	保存します。
チェックされていない場合	保存しません。

［トラッカ削除/アーカイブのタイミング］

トラッカの削除、または、アーカイブするタイミングを指定します。

■相対指定/絶対指定

トラッカの削除、または、アーカイブするタイミングをトラッカの実行終了時からの経過時間か、指定した時刻にするかを設定します。

相対指定	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。
絶対指定	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。

■待ち時間

相対指定の時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します。

ジョブネットワークの実行が完了してから、アーカイブされるまでの間はジョブネットワークの再実行が可能です。デフォルトは10分です。

- アーカイブ待ちの間、未アーカイブ状態のトラッカ情報は常駐プロセスのメモリにも保持されます。この設定を長くした場合、メモリ領域をその間確保し続けるためマシンの性能等に影響を及ぼす場合があります。
- トラッカはユーザごとに1ジョブネットワークあたり1ディレクトリで管理されます(サブジョブネットワークは親ジョブネットワークとは別で個別のディクレトリで管理されるので、ジョブネットワーク内にサブジョブネットワークがあれば、サブジョブネットワーク分さらにディレクトリは増えます)。

従ってアーカイブ待ち時間を長くすると、未アーカイブ状態のトラッカのディレクトリがそれだけ多く残ることになります。1ディレクトリ内に格納できるディレクトリ数がファイルシステムの制限に達しないよう、アーカイブ待ち時間を調整するようにしてください。

- マシンを再起動した場合、待ち時間はリセットされ、カウントし直されます。（例）待ち時間を1時間に設定し、ジョブネットワークの実行完了後30分でマシンを再起動した場合には、再起動後1時間が経過するまでアーカイブされません。

■実行時刻

絶対指定の時のトラッカを削除、または、アーカイブする時刻を設定します。

ジョブネットワークの実行が完了してから、アーカイブされるまでの間はジョブネットワークの再実行が可能です。デフォルトは00:00です。

実行時刻は、00:00から23:59までの時刻をHH:MM形式で設定します。

- マシンを再起動した場合、実行時刻はリセットされ、カウントし直されます。（例）実行時刻を00:00に設定し、ジョブネットワークが23:00に終了。その後、23:30にマシンを停止し00:10にマシンを再起動した場合には、次の00:00になるまでアーカイブされません。

■エラー停止トラッカを削除/アーカイブする

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します。

チェックされている場合	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。
チェックされていない場合	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。

6.5. システム環境の設定を変更する

［システム環境設定］ダイアログの［時刻設定］画面で、現在設定されているシステム環境のパラメータの参照および変更ができます。

［時刻設定］を変更できるのは、管理者アカウントでログインしてUMSモードで接続した場合のみです。

［システム環境設定］ダイアログの操作・実行ログのタブはJobCenter　R12.5.4以降で表示されます。機能の詳細については、 <操作・実行ログ機能 利用の手引き>を参照してください。

色の設定以外のシステム環境の設定はコマンドでも変更可能です。詳細については、<コマンドリファレンス>の「3.34　jc_syscfg　システム環境設定の参照および変更」のsetcfgサブコマンドの説明を参照してください。

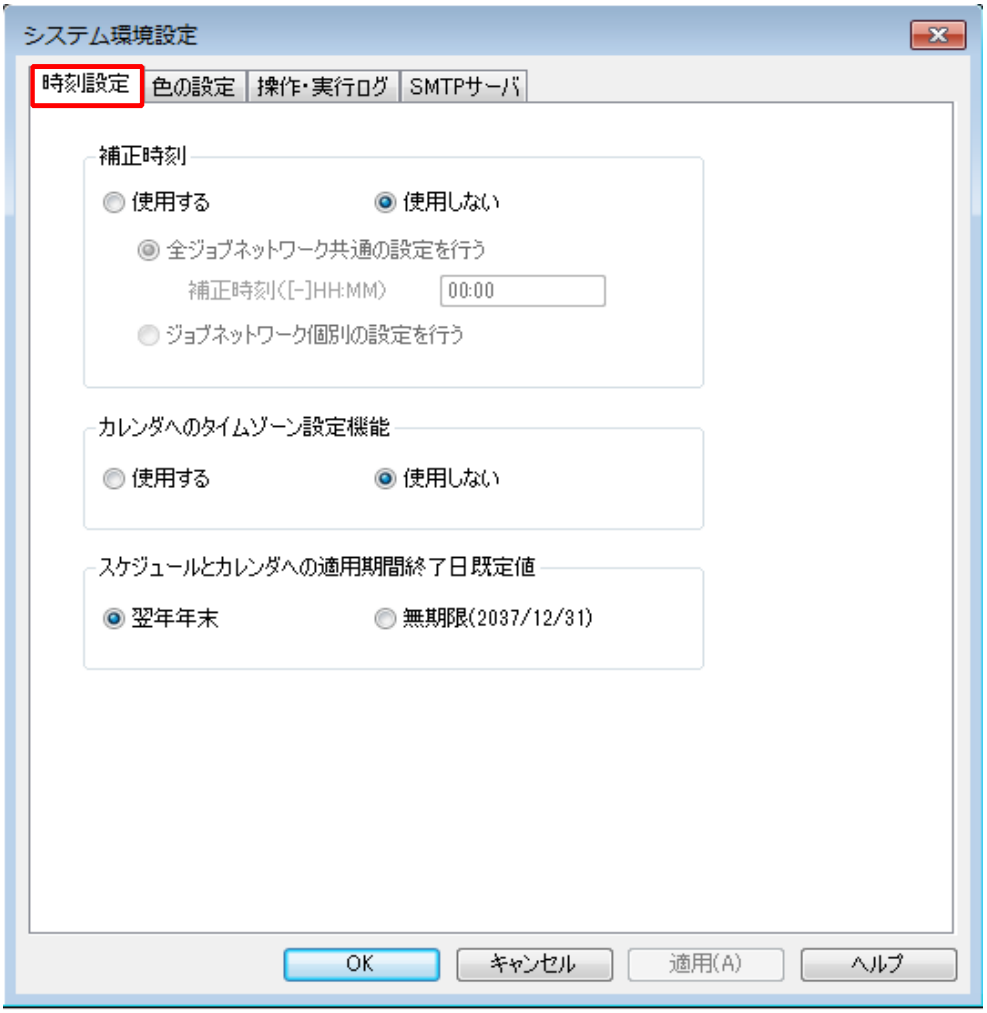

図6.10　［システム環境設定］画面例

パラメータの変更は、修正値の入力後に［適用］ボタンをクリックします。修正値を誤って入力した場合は［キャンセル］ボタンをクリックします。

［時刻設定］タブの設定項目は次のとおりです。

■補正時刻

補正時刻を使用するかしないかを選択します。

使用する	補正時刻設定機能を利用します。 システム全体へ同じ補正時刻を設定するか、ジョブネットワークに個別に設定するかを選択できます。	
	全ジョブネットワーク共通の設定を行う	サイト全体に設定された補正時刻が適用されます。
	ジョブネットワーク個別の設定を行う	ジョブネットワーク毎に補正時刻が設定できます。
使用しない	補正時刻を利用しません。	

JobCenterバージョンによっては、システム(サイト)全体で補正時刻を利用しないようにするために補正時刻の［使用しない］を設定する際に、［ジョブネットワーク個別の設定を行う］を選択したまま全体の［使用しない］を選択して［OK］または［適用］すると、ジョブネットワーク個別の補正時刻設定が有効のまま動作する場合があります。

補正時刻をシステム(サイト)全体で［使用する］の状態から［使用しない］に確実に変更するためには、次の手順で設定するようにしてください。

1. ［システム環境設定］ダイアログの［時刻設定］画面を表示する。
2. (［補正時刻］が［使用しない］の場合はいったん［使用する］を選択した上で)［全ジョブネットワーク共通の設定を行う］を選択する。
3. ［補正時刻］の［使用しない］を選択して［OK］または［適用］をクリックして適用する。

■時間

日付の切り替わる時刻を設定します。

補正時刻の設定を［全ジョブネットワーク共通の設定を行う］にした場合は、［システム環境設定］画面で、［ジョブネットワーク個別の設定を行う］にした場合は、ジョブネットワークのパラメータ設定画面で、補正時刻の設定を行います。

補正時刻は、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。システム全体に設定を行う場合でも、ジョブネットワーク個別に設定を行う場合でも設定範囲は変わりません。

補正時刻を［使用する］に設定すると、ジョブネットワークフローの部品は次のように動作します。

（例）補正時刻を［使用する］［13:00］とし、ジョブネットワークを14:00に起動した場合

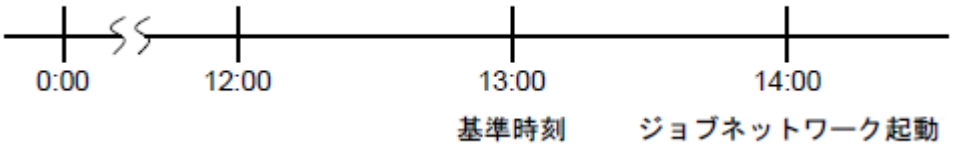

図6.11 時刻補正例

- 待ち合わせ部品の設定時刻が12:30の場合、翌日の12:30まで待ち合わせます。
- 待ち合わせ部品の設定時刻が13:30の場合、ジョブネットワーク起動時には待ち合わせ時間超過として動作します。

補正時刻を［使用する］に設定すると、各オブジェクトは次のような動作をします。

- カレンダ分岐のとき

  補正された基準時刻を日付の切り替わり時刻として、分岐の条件判断に使用します。

- 時刻待ち合わせのとき

  補正された基準時刻を待ち合わせの条件に使用します。

  指定した時刻がすでに過ぎている場合、待ち合わせは行わず、ジョブネットワークの処理を継続します。ただし、基準時刻を設定していない場合は、従来どおり、次の日の該当する時刻まで待ち合わせます。

- ファイル待ち合わせのとき

  補正された基準時刻を待ち合わせの条件に使用します（更新基準時刻、タイムアウト両方の条件で使用します）。

  指定した時刻がすでに過ぎている場合、待ち合わせは行わず、ジョブネットワークの処理を継続します。ただし、基準時刻を設定していない場合は、従来どおり、次の日の該当する時刻まで待ち合わせます。

- クリティカルポイント警告のとき

  補正された基準時刻を警告の条件に使用します。指定した時刻がすでに過ぎている場合、警告および警告時の動作をします。

■カレンダへのタイムゾーン設定機能

カレンダにタイムゾーンの設定を行うことにより、関連付けを行ったスケジュールを設定したタイムゾーンに応じた時刻で起動させたり、設定を行ったタイムゾーンに応じた時刻で待ち合わせや警告動作を行わせたりできる機能です。

詳細は16章 「日本以外のタイムゾーンで利用する」 を参照してください。

■スケジュールとカレンダへの適用期間終了日既定値

スケジュールとカレンダの「ルール設定」画面の適用期間の終了日の初期値を選択します。

翌年年末	適用期間の終了日の初期値として、翌年年末を設定する。
無期限(2037/12/31)	適用期間の終了日の初期値として、2037/12/31を設定する。

6.6. 色の設定を変更する

［システム環境設定］ダイアログの［色の設定］画面では、現在設定されている色の設定のパラメータの参照および変更ができます。

［色の設定］を変更できるのは、管理者アカウントでログインしてUMSモードで接続した場合のみです。

ただし、設定項目中の［EUIモードでの色の変更］を設定すると一般アカウントでの変更が可能になります。

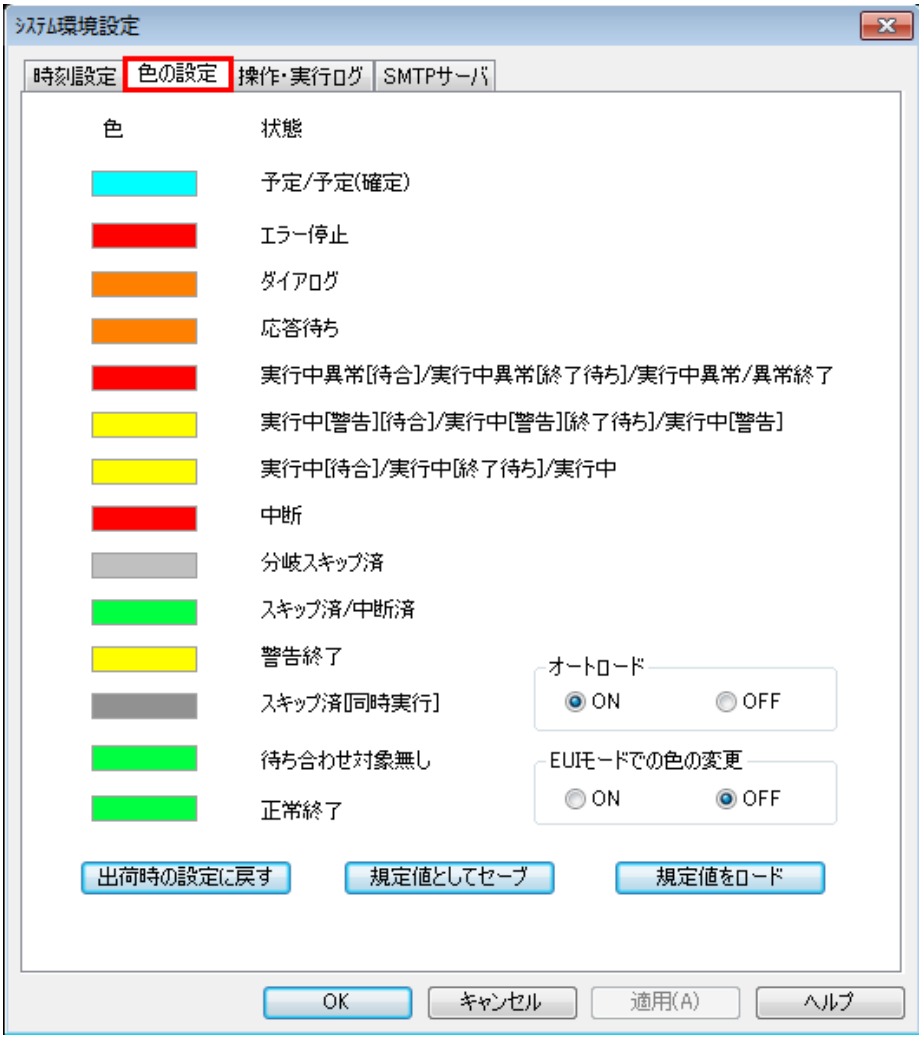

図6.12 ［色の設定］画面例

パラメータの変更は、修正値の入力後に［適用］ボタンをクリックします。修正値を誤って入力した場合は［キャンセル］ボタンをクリックしてください。

［色の設定］タブの設定項目は次のとおりです。

■色／状態

トラッカー覧での状態の表示色を変更できます。

［色のボックス］をクリックすると［色の設定］ダイアログが表示されます。

現在設定されている色を変更できます。

図6.13 色の設定画面

■オートロード

ON	JobCenter CL/Winから再接続したときに、変更した内容が自動的に有効になります。
OFF	変更した内容を自動的にロードしません。

■EUIモードでの色の変更

ON	EUIモードで色の変更が可能になります。
OFF	EUIモードで色の変更はできません。

EUIモードについては、<基本操作ガイド>の「2.3.3 ログインユーザ別のGUI画面モード、サーバ接続時の登録モード」を参照してください。

■出荷時の設定に戻す

出荷時設定の色データに戻します。

■既定値としてセーブ

現在表示されている色データを既定値としてセーブします。

■既定値をロード

既定値としてセーブした色データをロードします。

［EUI Mode］でも有効です。

6.7. 操作・実行ログ

［システム環境設定］ダイアログの［操作・実行ログ］画面では、JobCenter　CL/Winから実行された操作ログ、およびジョブネットワークの実行ログに関する設定のパラメータの参照および変更ができます。

［操作・実行ログ］の設定を変更できるのは、管理者アカウントでログインしてUMSモードで接続した場合のみです。

出力される操作ログ、実行ログのフォーマット、出力場所などの詳細については、 <操作・実行ログ機能 利用の手引き>を参照してください。

> 操作・実行ログの設定はコマンドでも変更可能です。詳細については、<コマンドリファレンス>の「3.34 jc_syscfg システム環境設定の参照および変更」のsetcfgサブコマンドの説明を参照してください。

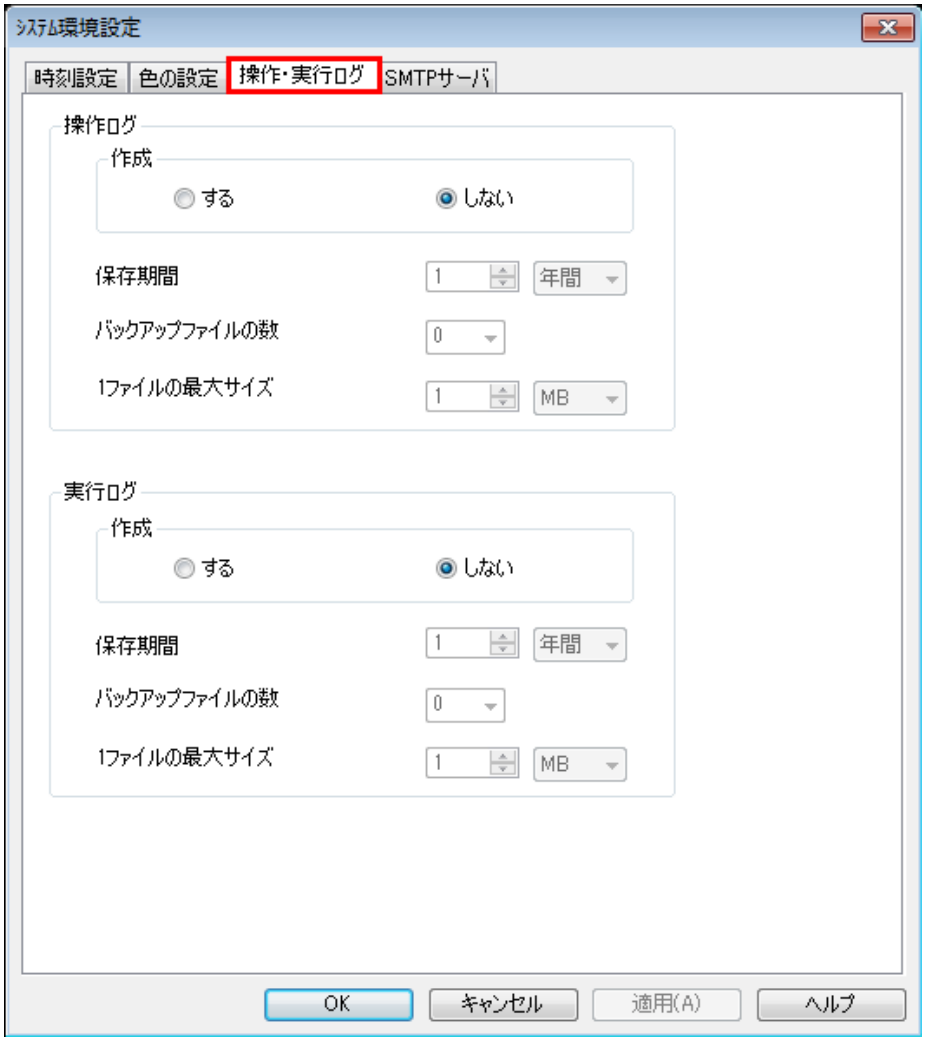

図6.14 ［操作・実行ログ］画面例

パラメータの変更は、修正値の入力後に［適用］ボタンをクリックします。修正値を誤って入力した場合は［キャンセル］ボタンをクリックしてください。

［操作・実行ログ］タブの設定項目は次のとおりです。

■作成

操作ログ、実行ログの出力の有無を設定します。

する	操作ログ、実行ログを出力します。
しない	操作ログ、実行ログを出力しません。

■保存期間

操作ログ、実行ログは、1日単位で出力されます。

ここでは、何日分のログを保存するかを設定します。

■バックアップファイルの数

最大サイズを超えた操作ログ、実行ログは、バックアップファイルとして別名で保存されます。ここでは、1日ごとに最大いくつのバックアップファイルを作成するかを設定します。

■1ファイルの最大サイズ

1つの操作ログ、実行ログの最大サイズを指定します。

6.8. SMTPサーバ

エラー時のメール送信機能で使用するSMTPサーバを設定します。詳しくは<基本操作ガイド>の「15.1 使用するSMTPサーバの設定」を参照してください。

7. 稼働日カレンダの設定

稼働日カレンダは、サイトごとにジョブの運用を行う日を定義したカレンダです。

稼働日カレンダによりジョブの運用を行うように定義された日（「稼働日」と呼びます）、ジョブの運用を行わないように定義された日（「休止日」と呼びます）を設定します。

稼働日カレンダはJobCenter管理者権限のあるユーザのみが作成できます。

各ユーザは稼働日カレンダを任意のスケジュールに適用することにより、作成しているジョブの自動実行スケジュールに、稼働日相対や休日シフトを組み合わせて休止日を考慮したスケジュール設定ができるようになります。

稼働日カレンダは「毎週xx曜日」のようなルールを組み合わせて作成し、サイトごとに200個まで作成可能です。

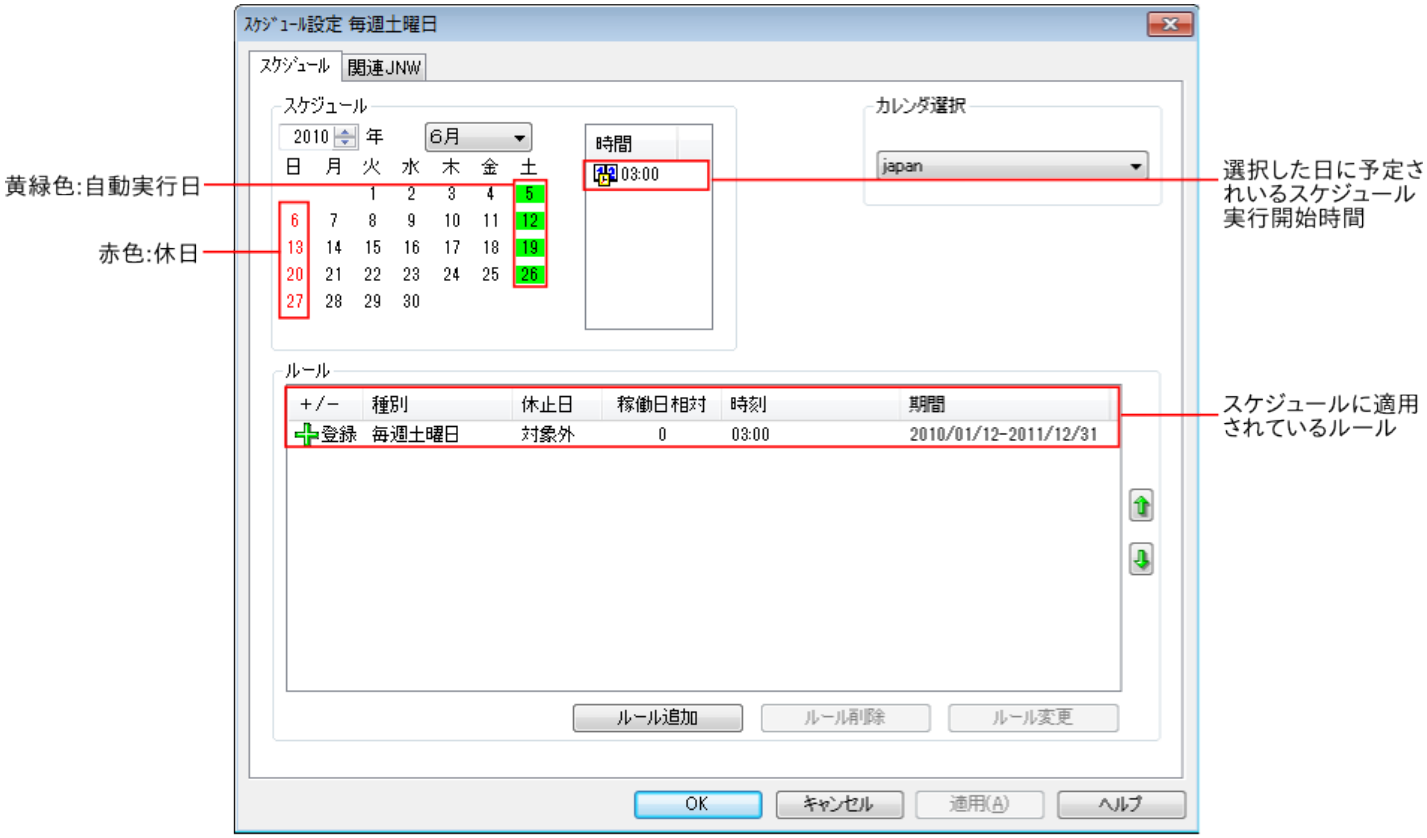

図7.1 ［スケジュール設定］画面例

7.1. 稼働日カレンダを設定する

稼働日カレンダの設定ができるのは、管理者アカウントでログインしたユーザのみです。稼働日カレンダを管理者以外のユーザが参照する場合には、稼働日カレンダのカレンダ部のみの表示となります。

新規に稼働日カレンダを作成する方法は次のとおりです。

1. ［マネージャフレーム］で［カレンダ一覧］を表示します。

2. ［カレンダ一覧］の画面上で右クリックしたときのポップアップメニューから［新規作成］を選択、またはメニューバーの［ファイル］－［新規作成］を選択します。

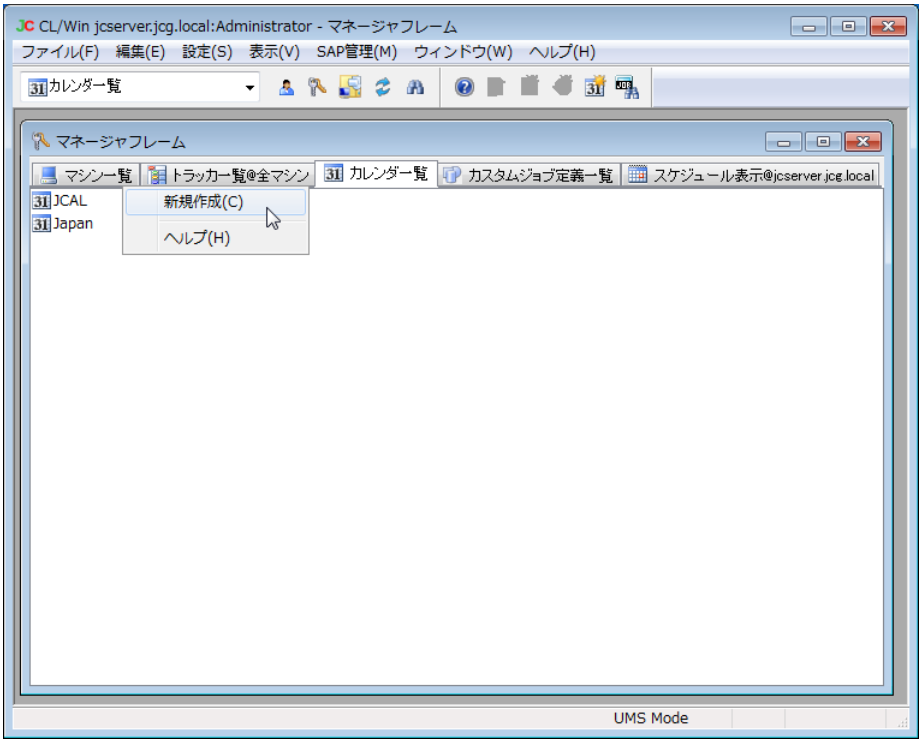

図7.2 稼働日カレンダの［新規作成］メニュー選択画面例

3. 稼働日カレンダの［名前の設定］ダイアログが表示されますので、新規に作成する稼働日カレンダの名前を入力します。

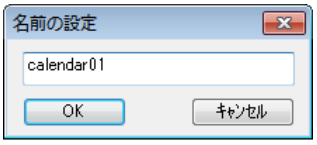

図7.3 稼働日カレンダの［名前の設定］画面例

4. 設定後、［OK］をクリックします。［カレンダ一覧］に戻ります。

5. ［カレンダ一覧］で、設定する稼働日カレンダ名を選択し、ダブルクリックまたは右クリックしたときのポップアップメニューから［開く］を選択します。

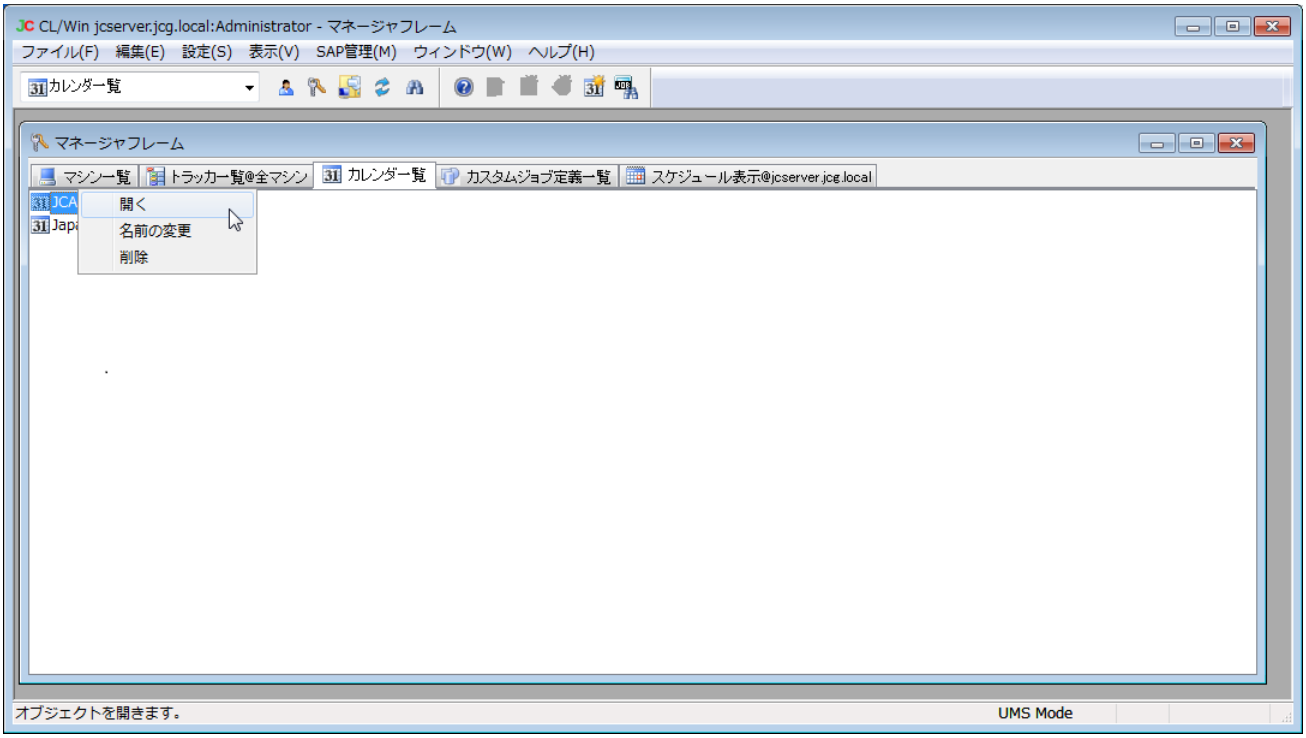

図7.4 稼働日カレンダを［開く］メニュー選択画面例

6. ［カレンダ設定］ダイアログが開きますので、［追加］または［変更］ボタンをクリックします。

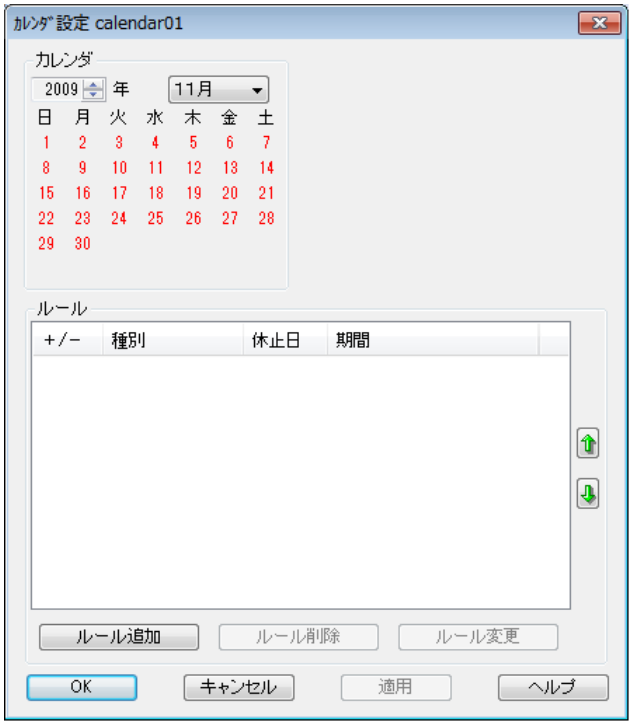

図7.5 ［カレンダ設定］ダイアログ画面例

7. ［ルール設定］ダイアログが表示されますので、ルールを設定します。

図7.6 稼働日カレンダの［ルール設定］ダイアログ画面例

たとえば、［ルール設定］ダイアログを利用して、次のようなルール設定ができます。

■月初週末ルール

カレンダ設定ルールの1つとして「月初週末」が使用できます。

これは、毎月の最初の週末にジョブを実行させるルールです。ほぼ「毎月第1土曜日＋前シフト」というルールと同じですが、前シフトで前月になる場合には翌週末に延期されます。

設定には、カレンダルールとして「特殊」を選択し、さらに「月初週末」を選択します。

■毎年(曜日)ルール

カレンダ設定ルールの1つとして「毎年(曜日)」が使用できます。

これは、「MM月」「第n（一、二、三、四、五、最後）週」の「何曜日」にジョブを実行させるルールです。

ルールの設定方法および設定項目については、<基本操作ガイド>の「6.1.3　スケジュールルールを作成する」を参照してください。

稼働日カレンダ変更に伴う制限事項について

稼働日カレンダに関連付いたスケジュール（及びジョブネットワーク）がある状態で、その稼働日カレンダのルールや適用期間を変更する場合、スケジュール日の再計算によっては意図しない動作となる場合があります。具体的には以下のような事象となりますので注意してください。(以下、3/17に稼働日カレンダの修正を行ったとして例を挙げてあります)

事象：	実行済みのスケジュールが休止日でシフトされ、予定として再スケジュールされる
発生条件：	1. 稼働日後シフトが指定され、かつ、過去にスケジュール実行された稼働日を非稼働日に変更し、かつ、非稼働日となった翌日から変更当日(*1)まで連続して非稼働日の場合。 例)　稼働日カレンダ　3/13　14　15　16　17　18 ×　×　○　×　×　○ スケジュール　予定日を3/13に設定 → 3/15に後シフト (3/15にすでに実行済み) 上記の状況で、3/17に稼働日カレンダを変更して3/15を×にすると、シフトが再計算されて3/13の予定日分が3/18に再スケジュールされます。

2. 稼働日後シフトが指定され、かつ、変更当日に実行済みのスケジュール時刻後に、その当日を非稼働日に変更した場合。

例)　稼働日カレンダ　3/15　16　17　18
×　×　○　○

スケジュール　予定日を3/15に設定 → 3/17に後シフト
(3/17当日にすでに実行済み)

上記の状況で、3/17に稼働日カレンダを変更して3/17を×にすると、シフトが再計算されて3/15の予定分が3/18に再スケジュールされます。

3. 稼働日前シフトが指定され、かつ、前シフトにより過去に実行済みの稼働日の翌日から元のスケジュール日まで連続して非稼働日のときに、変更当日(*2)以降(の一部)を稼働日に変更した場合。

例1)　稼働日カレンダ　3/15　16　17　18　19　20
○　×　×　×　×　×

スケジュール　予定日を3/15、20に設定 → 3/20分が3/15に前シフト (3/15にすでに実行済み)

上記の状況で、3/17に稼働日カレンダを変更して3/18を○にすると、シフトが再計算されて3/20予定分が3/18に再スケジュールされます。

例2)　稼働日カレンダ　3/15　16　17　18
○　×　×　×

スケジュール　予定日を3/18に設定 → 3/15に前シフト
(3/15にすでに実行済み)

上記の状況で、3/17に稼働日カレンダを変更して3/17を○にすると、シフトが再計算されて3/18予定分が3/17に再スケジュールされ、予定時刻によっては時刻超過とみなされて即時実行される場合があります。

事象：	未来日に予定されていたスケジュールが、予定が削除され実行されない
発生条件：	1. 稼働日後シフトが指定され、かつ、スケジュール日から変更当日(*1)まで連続した過去の非稼働日(の一部)を稼働日に変更した場合。 例)　稼働日カレンダ　3/13　14　15　16　17　18 ×　×　×　×　×　○ スケジュール　予定日3/14 → 3/18に後シフトで実行予定 上記の状況で、3/17に稼働日カレンダを変更して3/15を○にすると、シフトが再計算されて3/14予定分が有効かつ3/18後シフトの実行予定分は解除になります。ただし3/14予定分は予定日超過としてJobCenterが認識した時点で即時実行開始されますので注意してください。 2. 稼働日後シフトが指定され、かつ、スケジュール日から変更当日まで連続した非稼働日のときに、スケジュール時刻後に、変更当日を稼働日に変更した場合。 例)　稼働日カレンダ　3/15　16　17　18 ×　×　×　○ スケジュール　予定日3/15 → 3/18に後シフトで実行予定 上記の状況で、3/17に稼働日カレンダを元の3/15のスケジュール時刻より後の時刻以降に変更して3/17を○にすると、シフトが再計算されて3/18後シフトの実行予定分が解除され、当該スケジュールは実行されません。(3/17に3/15分を実行することはありません)

	3. 稼働日前シフトが指定され、かつ、変更当日(*2)からスケジュール日まで連続した稼働日の全てを非稼働日に変更した場合。 例)　稼働日カレンダ　3/16　17　18 ○　○　○ スケジュール　予定日3/18で前シフト設定あり 上記の状況で、3/17に稼働日カレンダを変更して3/17と3/18を×にすると、3/18予定分は前シフトされず、休止日として解除になります。 例)　稼働日カレンダ　3/16　17　18 ○　×　○ スケジュール　予定日3/19で前シフト設定あり 上記の状況で、3/17に稼働日カレンダを変更して3/19を×にすると、3/19予定分は前シフトされず、休止日として解除になります。 (*1) スケジュール時刻がカレンダの変更時刻後の場合は、変更の前日。 (*2) スケジュール時刻がカレンダの変更時刻前の場合は、変更の翌日。

■回避策について

過去日の稼働日・非稼働日が変更されてしまうことによる制限事項については、過去日に影響を与えないように稼働日カレンダを変更することで回避可能です。

例えば適用期間を変更すると適用開始日が変更日以降として扱われるため、変更日以前の日は全て適用期間外（=休止日）として扱われてしまいます。このような場合は既存ルールの適用期間を変更するのではなく、新しいルールで適用期間が異なるものを追加することで対処してください。(ルールは一覧の中で下にあるものほど優先して適用されます)

なお、既存のルールは適用期間が終わった後で、休止日シフトの影響がないぐらいに十分に時間をおいてから削除してください。

当日以降の未来日を変更する場合には、稼働日カレンダの変更後に個別にトラッカを削除、スキップ、または即時投入して対処する必要があります。

7.2. 稼働日カレンダを配布する

マシングループを構成している場合、更新した稼働日カレンダは、マシングループ内の各マシンへ自動配布されます。ただし、停止しているメンバマシンには自動配布されませんので、メンバマシン起動後に手動で配布する必要があります。

また、標準リモートマシン構成の場合も、手動で配布する必要があります。(マシングループ、標準リモートマシン構成については、3章 「JobCenterネットワーク環境構築」 参照)

手動での配布手順は次のとおりです。

1. マネージャフレームの［マシン一覧］画面を表示します。
2. 稼働日カレンダを配布したいマシンを選択し、右クリックしたときのポップアップメニューから「稼働日カレンダの更新」を選択します。
3. 稼働日カレンダが指定したマシンに転送されます。

8. 異なる言語間における接続設定

Windows版JobCenter MG/SVは、原則として同じ言語環境のCL/WinやMG/SVのみ接続できます。例えば次のような場合は接続したり連携動作できません。

■日本語版の CL/Win で、英語でセットアップした Windows 版 MG/SV に接続

■中国語版の CL/Win で、日本語でセットアップした Windows 版 MG/SV に接続

■日本語でセットアップした MG に、英語でセットアップした Windows 版 MG/SV のアイコンを追加して標準リモートマシン構成を設定

■英語でセットアップした MG でマシングループを組み、日本語版の Windows 版 SV をマシングループに追加、etc..

上記のような組み合わせの場合は、接続される側のWindowsマシン、あるいはMGで管理される側(つまりSV側)のWindowsマシンにおいて、サーバの環境設定で「多言語接続」の設定が必要になります。

1. 管理者アカウントでWindowsにログオンし、Windowsの［スタート］－［プログラム］－［JobCenter］－［SV］－［サーバの環境設定］を選択して、サーバの環境設定を起動します。

2. 左ペインのプロパティをクリックし、右ペインの総合設定をダブルクリックして、総合設定のプロパティダイアログを表示します。

3. 「多言語接続を行う」にチェックボックスにチェックを入れて適用します。

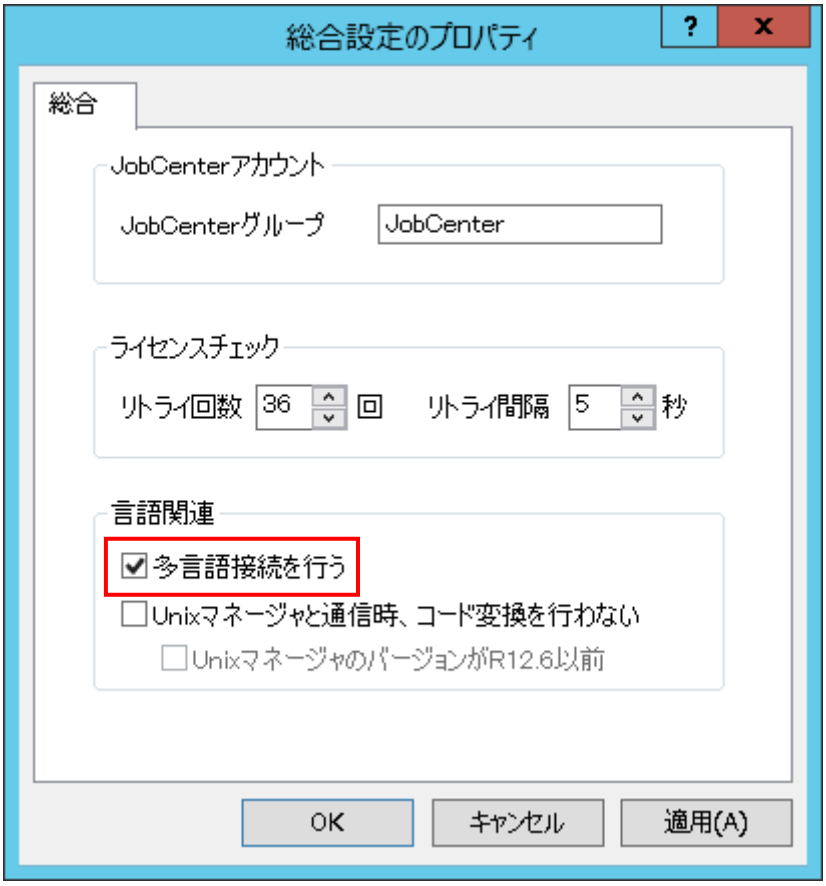

図8.1 総合設定のプロパティ画面例

「多言語接続を行う」チェックボックスのチェックの有無により、異なる言語間でのJobCenter CL/Winによるログインおよび JobCenter MGのマシングループへの追加の可否は次のようになります。

表8.1 多言語チェックボックスの有無と他言語間での接続の関係

多言語接続設定	マシン種類	他言語のJobCenter MGのマシングループに追加したり、 MGからSVのキューを参照	他言語のJobCenter CL/Win からのログイン
オン	JobCenter MG	ー注1	○
	JobCenter SV	○	○
オフ	JobCenter MG	ー注1	×
	JobCenter SV	×	×

注1　JobCenter MGは、言語環境に関わらず他JobCenter MGのマシングループには追加できません。

■他言語のJobCenter CL/WinからJobCenter MGまたはJobCenter SVに接続している間に、接続先マシンの「多言語接続」設定がオフになっても、その設定変更は反映されません。

CL/Winで再度接続する時点で有効になります。

9. 日本語環境での文字コード変換

日本語環境において、UNIX版 JobCenter(SJIS)とWindows版 JobCenterや、UNIX版 JobCenter(SJIS)とUNIX版 JobCenter(EUC)などの異なる文字コードで JobCenterネットワーク構成を構築する場合、文字コード変換の設定が必要になります。

必要な文字コード変換が行われないと、標準出力や標準エラー出力の文字化けが生じたり、変数継承が正しく行われない場合がありますので注意してください。

なお、英語環境、中国語環境においては、文字コード変換を設定する必要はありません。

9.1. 文字コード変換の設定の必要性を判断する

JobCenterネットワーク構成において、UNICODE環境とEUC、またはSJIS環境を混在しない場合は、「9.1.1 文字コード変換の設定の必要性を判断する(UNICODE環境混在なし)」を参照してください。 JobCenterネットワーク構成において、UNICODE環境とEUC、またはSJIS環境を混在する場合は「9.1.2 文字コード変換の設定の必要性を判断する(UNICODE環境混在あり)」を参照してください。

9.1.1. 文字コード変換の設定の必要性を判断する(UNICODE環境混在なし)

以下を元に設定の必要性を確認します。

表9.1 JobCenterネットワーク構成の組み合わせと文字コード変換の設定の必要性

構成の組み合わせ	文字コード変換の設定の必要性
UNIX版(EUC) ←→ UNIX版(EUC)	設定は不要。
UNIX版(SJIS) ←→ UNIX版(SJIS)	設定は不要。
UNIX版(EUC) ←→ UNIX版(SJIS)	SJISでセットアップしたUNIX版JobCenter側にcodecnv.cnfの設定が必要。 設定方法については「9.2.1.1 SJIS側のUNIX版JobCenterの文字コード変換を設定する」を参照してください。
Windows版(非UNICODE) ←→ Windows版(非UNICODE)	設定は不要。
Windows版(非UNICODE) ←→ UNIX版(EUC)	設定は不要。
Windows版(非UNICODE) ←→ UNIX版(SJIS)	Windows版JobCenter側で設定が必要。 設定方法については「9.2.2 Windows版JobCenterの文字コード変換を設定する」を参照してください
Windows版(UNICODE) ←→ UNIX版(UNICODE)	設定は不要。
Windows版(UNICODE) ←→ WINDOWS版(UNICODE)	設定は不要。
UNIX版(UNICODE) ←→ UNIX版(UNICODE)	設定は不要。

■上記の設定要否はMGとSVの「役割」により必要性が決まるものではありません。あくまでも言語環境の異なるJobCenterを混在利用する場合の組み合わせとして判断するようご注意ください。

■英語や中国語、日本語でそれぞれセットアップしたJobCenterを混在使用する場合は、相互に自動的に文字コード変換する設定や機能はありません。その場合は2バイト文字を含まない名前を使って定義する、ジョブスクリプトに2バイト文字を含まないようにする等、英語のみを利用してシステムを構築するようにしてください。

9.1.2. 文字コード変換の設定の必要性を判断する(UNICODE環境混在あり)

以下を元に設定の必要性を確認します。

■Windows同士でJobCenterネットワーク構成を組む場合

設定の必要性はありません。

■UNIX同士でJobCenterネットワーク構成を組む場合

SV側で設定が必要になります。設定方法については、「9.2.1.2 UNIX版同士のUNICODE混在環境JobCenter SVで文字コード変換を設定する」を参照してください。

■WindowsとUNIXでJobCenterネットワーク構成を組む場合

表9.2 JobCenterネットワーク構成の組み合わせと文字コード変換の設定の必要性

構成の組み合わせ	文字コード変換の設定の必要性
Windows版(非UNICODE) ←→ UNIX版(UNICODE)	Windows版JobCenter側で設定が必要。 設定方法については「9.2.2　Windows版JobCenterの文字コード変換を設定する」を参照してください
Windows版(UNICODE) ←→ UNIX版(EUC)	設定は不要。
Windows版(UNICODE) ←→ UNIX版(SJIS)	Windows版JobCenter側で設定が必要。 設定方法については「9.2.2　Windows版JobCenterの文字コード変換を設定する」を参照してください

■上記の設定要否はMGとSVの「役割」により必要性が決まるものではありません。あくまでも言語環境の異なるJobCenterを混在利用する場合の組み合わせとして判断するようご注意ください。

■英語や中国語、日本語でそれぞれセットアップしたJobCenterを混在使用する場合は、相互に自動的に文字コード変換する設定や機能はありません。その場合は2バイト文字を含まない名前を使って定義する、ジョブスクリプトに2バイト文字を含まないようにする等、英語のみを利用してシステムを構築するようにしてください。

■UNICODE環境とEUC、またはSJIS環境を混在して利用する場合、JobCenterを利用する　MG,SV全てを R13.2以降で統一する必要があります。

9.2. 文字コード変換を設定する

UNIX版JobCenterとWindows版JobCenterで設定方法が異なります。

JobCenter CL/Win（GUI画面）からではなく、nmapmgrコマンドからマシン登録を行った場合は、NQS TYPEの設定が正しいかを併せて確認する必要があります。

nmapmgrコマンドによるJobCenterのNQS TYPE確認方法は次のとおりです。

```
# nmapmgr ↵
      NMAPMGR:> show state <ホスト名>
```

詳細については<コマンドリファレンス>の「3.12 nmapmgr ネットワークの構成管理」を参照してください。

9.2.1. UNIX版JobCenterの文字コード変換を設定する

UNIX版JobCenterは連携するJobCenterの組み合わせにより、設定方法がMG側とSV側で分かれます。判別方法は、「9.1 文字コード変換の設定の必要性を判断する」を参照してください。

9.2.1.1. SJIS側のUNIX版JobCenterの文字コード変換を設定する

UNIX版JobCenterの文字コード変換(EUC←→SJIS)を設定するには、nssetupでSJISを指定してセットアップした方のUNIX版JobCenterについて次の設定ファイルを作成し、通信相手のホスト名を記述します。

なお、次にJobCenterを再起動するまで設定内容は反映されませんのでご注意ください。

```
/usr/lib/nqs/codecnv.cnf
```

ホスト名の記述は空白、TABもしくは改行コードで区切って複数設定できます。設定するホスト名は、nmapmgr（マネージャフレームのマシン一覧）に登録されているマシンのプリンシパルな名前を利用してください。

プリンシパルな名前は、nmapmgrサブコマンドのget name <mid>で得られる名前です。詳細については<コマンドリファレンス>の「3.12 nmapmgr ネットワークの構成管理」を参照してください。

例えば、

■MG(HP-UX)で、nssetup時にSJISでセットアップ

■SV(Linux)で、nssetup時にEUCでセットアップ

の状況では、MG側マシンの上記パスにcodecnv.cnfを作成し、その中にSVマシンのプリンシパル名を記述することになります。それによりSV上でEUCコードで出力された単位ジョブ実行結果がMG側に戻される際に、自動的にSJISに変換されてトラッカの出力結果やエラー出力結果に表示されます。

ただしMGからSVに単位ジョブをリモート投入する際の環境変数LANGについては、MG側の値がそのままSVに引き継がれますので、単位ジョブスクリプトの先頭でLANGを設定し直す必要があります。詳細については「15.1.4 MGとSVとのLANGが異なる場合の注意事項」を参照してください。

> ⚠ 変換の必要のないホストに対する記述が設定ファイル内に存在した場合、転送したスクリプトファイル、および戻される結果ファイルの内容が不正になる場合があります。設定後は、正しく日本語情報が標準出力画面に表示されるか確認してください。

9.2.1.2. UNIX版同士のUNICODE混在環境JobCenter SVで文字コード変換を設定する

ジョブを転送する場合、MG側のJobCenterの文字コードが用いられます。そのため、UNICODE混在環境であり、かつMG側とSV側がUNIX版JobCenterの場合、SV側のJobCenterの文字コードに設定する必要があります。設定方法は以下となります。

SV側のデーモン設定ファイル（daemon.conf）にshell_uselocallang=ONを指定する。

```
NQSDAEMON_OPT=-x shell_uselocallang=ON
```

SV側のJobCenterのデーモン設定ファイルを修正した後、設定を反映させるにはJobCenterの再起動が必要です。

本設定を行わない場合は、ジョブの標準エラー出力が文字化けを起こす場合があります。

9.2.2. Windows版JobCenterの文字コード変換を設定する

Windows版JobCenterは連携するJobCenterの組み合わせにより、設定方法がMG側とSV側で分かれません。

Windows版JobCenterの文字コード変換の設定は次のとおりです。

1. 管理者アカウントでWindowsにログオンし、Windowsの［スタート］－［プログラム］－［JobCenter］－［SV］－［サーバの環境設定］を選択して、サーバの環境設定を起動します。
2. 左ペインのプロパティをクリックし、右ペインの総合設定をダブルクリックして、総合設定のプロパティダイアログを表示します。

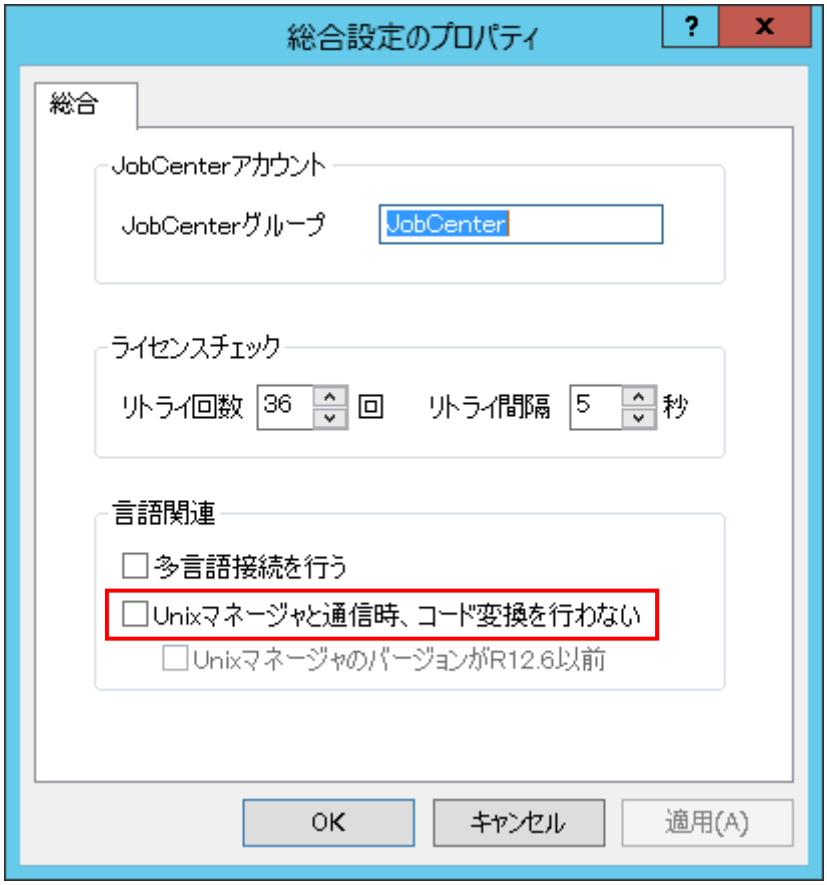

図9.1 総合設定のプロパティダイアログ画面例

3. ［言語関連］部分の「UNIXマネージャと通信時、コード変換を行わない」にチェックを入れます。「OK」または、［適用］ボタンをクリックします

通信先サーバがR12.6.x以前のUNIXの場合、「UNIXマネージャのバージョンがR12.6以前」にも追加でチェックを入れてください。通信先サーバがR12.7以降のUNIXであれば、この追加チェックは不要です。

R12.6以降のWindows版JobCenterは、UNIXサーバへ実行結果を送信する際は上記③のチェックボックスの設定に関わらず改行コードをCR+LFからLFのみに変換して実行結果を送信します。

10. ユーザ権限（パーミッション設定）

JobCenterを使用するユーザ権限の設定は、権限グループを基本として行います。権限グループとは、目的に応じた特定の権限を持つグループです。

JobCenter管理者は、任意に追加したユーザ定義の権限グループ、または、デフォルトで用意された権限グループの、いずれかの権限グループに各ユーザを所属させることにより、各ユーザに適切なアクセス権限を与えます。これにより、次のような運用が可能になります。

1. 目的に応じたユーザ定義の権限グループを追加できます。
2. デフォルトで、一般的な使用形態を考慮した権限グループが用意されています。
3. 他ユーザのジョブネットワークの参照や作成や実行が可能になります。
4. 特定のユーザのみに、他ユーザのジョブネットワークトラッカなどのアクセスを許可できます。
5. ユーザが所属する権限グループごとに、アクセス先のユーザを制限できます。
6. 特定のユーザに対して、ジョブネットワークの実行やトラッカの操作のみを許可し、ジョブネットワークの作成や変更を禁止できます。

権限グループは、R12.4.x以前のバージョンでのユーザレベルA～Dを拡張し、機能強化したものです。

10.1. アクセス権限の種類

権限グループで設定できるアクセス権限の一覧を次に示します。

表10.1 権限グループで設定できるアクセス権限の一覧

アクセス権限	説　明
ユーザ自身のジョブネットワークの一覧と参照	ユーザ自身の以下の定義および一覧タブを参照する権限です。 ■ジョブネットワーク ■スケジュール ■起動トリガ ■リリース計画
ユーザ自身のジョブネットワークの作成、変更、削除	ユーザ自身の以下の定義の作成、変更、削除を行う権限です。 ■ジョブネットワーク ■スケジュール ■起動トリガ ■リリース計画 この権限をONにすると「ユーザ自身のジョブネットワークの一覧と参照」の権限もONになります。
ユーザ自身のジョブネットワークの実行	ユーザ自身のジョブネットワークの実行ができる権限です。 この権限をONにすると「ユーザ自身のトラッカの一覧と参照」と「ユーザ自身のトラッカの操作」の権限もONになります。
ユーザ自身のトラッカの一覧と参照	ユーザ自身のトラッカ一覧やスケジュール表示タブを表示する権限です。ジョブネットワークトラッカを参照できます。
ユーザ自身のトラッカの操作	ユーザ自身のジョブネットワークトラッカの参照および強制起動、スキップ、保留、解除などの操作が可能な権限です。 この権限をONにすると「ユーザ自身のトラッカの一覧と参照」の権限もONになります。
他ユーザのジョブネットワークの一覧と参照	他ユーザの以下の定義および一覧タブを参照する権限です。 ■ジョブネットワーク ■スケジュール ■起動トリガ ■リリース計画
他ユーザのジョブネットワークの作成、変更、削除	他ユーザの以下の定義の作成、変更、削除を行う権限です。 ■ジョブネットワーク ■スケジュール ■起動トリガ ■リリース計画

	この権限をONにすると「他ユーザのジョブネットワークの一覧と参照」の権限もONになります。 また、この権限があれば単位ジョブのユーザ権限切り替えの設定が可能です。詳細については<基本操作ガイド>の「4.2.2　単位ジョブの実行条件（パラメータ）を設定する」を参照してください。
他ユーザのジョブネットワークの実行	他ユーザのジョブネットワークの実行ができる権限です。 この権限をONにすると「他ユーザのトラッカの一覧と参照」と「他ユーザのトラッカの操作」の権限もONになります。 なお、実行時のジョブネットワークトラッカは、そのジョブネットワークの所有者のトラッカに作成されます。
他ユーザのトラッカの一覧と参照	他ユーザのトラッカ一覧やスケジュール表示タブを表示する権限です。他ユーザのジョブネットワークトラッカを参照できます。
他ユーザのトラッカの操作	他ユーザのジョブネットワークトラッカの参照および強制起動、スキップ、保留、解除などの操作が可能な権限です。 この権限をONにすると「他ユーザのトラッカの一覧と参照」の権限もONになります。

他ユーザに関するアクセス先の許可の設定については、<基本操作ガイド>の「14.3　他ユーザへのジョブネットワーク、トラッカのアクセス許可を制限する」を参照してください。

10.2. デフォルトの権限グループ

デフォルトで用意されている権限グループと、それらに与えられているアクセス権限を表10.2「デフォルトで用意されている権限グループとアクセス権限」、デフォルトで用意されている権限グループの用途を表10.3「デフォルトで用意されている権限グループの用途」に示します。

表10.2 デフォルトで用意されている権限グループとアクセス権限

デフォルト権限グループ アクセス権限	JobCenter管理者	ジョブネットワーク開発者	ジョブネットワーク運用者	実行監視者	一般ユーザ
ユーザ自身のジョブネットワークの一覧と参照	○	○	○	×	○
ユーザ自身のジョブネットワークの作成、変更、削除	○	○	×	×	○
ユーザ自身のジョブネットワークの実行	○	○	○	×	○
ユーザ自身のトラッカの一覧と参照	○	○	○	○	○
ユーザ自身のトラッカの操作	○	○	○	×	○
他ユーザのジョブネットワークの一覧と参照	○	○	○	×	×
他ユーザのジョブネットワークの作成、変更、削除	○	○	×	×	×
他ユーザのジョブネットワークの実行	○	○	○	×	×
他ユーザのトラッカの一覧と参照	○	○	○	○	×
他ユーザのトラッカの操作	○	○	○	×	×

■○：許可　×：禁止を示します。

■新しい権限グループを追加定義することにより、任意の権限の設定が可能です。デフォルトで用意されている権限グループの権限の変更はできません。

■デフォルトの権限グループは、権限グループ名の後ろに「*（半角アスタリスク）」が付加されて表示されます。

■アクセス権限の詳細は表10.1「権限グループで設定できるアクセス権限の一覧」を参照してください。

表10.3 デフォルトで用意されている権限グループの用途

権限グループ	説　明
JobCenter管理者	JobCenterシステム構成の管理が可能な権限グループ（特権ユーザ）です。 全ての権限を持ちます。 次のようなJobCenterシステム構成の管理が可能です（ジョブネットワーク開発者には権限がありません）。 　　権限グループの設定と管理

	マシンおよびマシングループの管理とNQS（キュー）の管理や監視 マシングループ内の全トラッカの監視、稼働日カレンダの管理 システムの環境設定、トラッカ色の設定 他の権限グループと異なり、所属するユーザは固定です。 UNIX系： rootとnsumsmgr Windows :サーバの環境設定の管理者アカウントで設定したユーザ
ジョブネットワーク開発者	JobCenterシステムのジョブネットワークの開発やスケジュールの設定を行うユーザを想定した権限グループです。
ジョブネットワーク運用者	JobCenterシステムのジョブネットワークの実行や実行予定または実行中のジョブネットワークのトラッカの操作など、ジョブネットワークの運用のみを行うユーザを想定した権限グループです。
実行監視者	ジョブネットワークの実行の状態（実行予定、実行中、エラー停止など）の監視のみを行うユーザを想定した権限グループです。
一般ユーザ	ユーザ自身の定義の作成やジョブネットワークの実行、トラッカの操作が可能な権限グループです。 なお、この権限グループには、ユーザ定義の権限グループを含む他の権限グループに所属していないユーザが全て含まれます。明示的にユーザを所属することはできません。

10.3. 権限グループを設定する

JobCenter管理者ユーザ（管理者アカウント）でJobCenterにログインします。

メニューバーの［設定］－［パーミッション設定］を選択します。

［パーミッション］ダイアログが表示され、現在マシンに設定されているユーザ権限が表示されます。

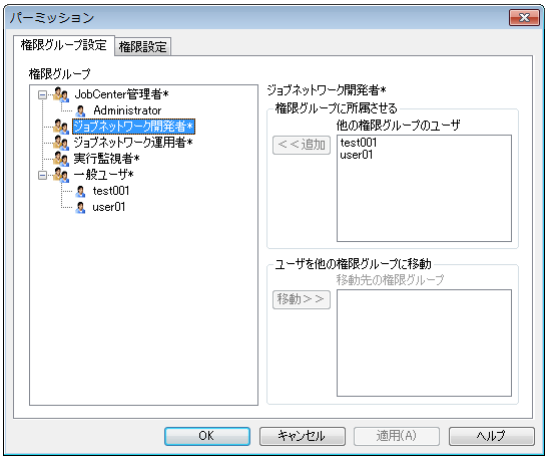

図10.1 ［パーミッション］ダイアログ画面例

［パーミッション］ダイアログから以下の設定を行えます。また、コマンドでも同様に設定可能です。［パーミッション］ダイアログからの設定方法については <基本操作ガイド>、コマンドでの設定方法については <コマンドリファレンス>の各項目を参照してください。

設定内容	基本操作ガイドの章番号	コマンドリファレンスの章番号
権限グループを設定する	「14.1 権限グループを設定する」	「3.35 jc_permcfg パーミッション設定の参照、および変更」
権限グループに所属させる	「14.1.1 権限グループに所属させる」	「3.35 jc_permcfg パーミッション設定の参照、および変更」のassignuserサブコマンド
ユーザを他の権限グループに移動する	「14.1.2 ユーザを他の権限グループに移動する」	「3.35 jc_permcfg パーミッション設定の参照、および変更」のassignuserサブコマンド
権限グループを作成する	「14.2 権限グループを作成する」	「3.35 jc_permcfg パーミッション設定の参照、および変更」のcategrp、copygrp、delgrpサブコマンド
［権限グループ名］を新規追加する	「14.2.1 権限グループ名を新規追加する」	「3.35 jc_permcfg パーミッション設定の参照、および変更」のcategrpサブコマンド
追加した権限グループの権限を設定する	「14.2.2 追加した権限グループの権限を設定する」	「3.35 jc_permcfg パーミッション設定の参照、および変更」のsetpermサブコマンド
他ユーザへのジョブネットワーク、トラッカのアクセス許可を制限する	「14.3 他ユーザへのジョブネットワーク、トラッカのアクセス許可を制限する」	「3.35 jc_permcfg パーミッション設定の参照、および変更」のaddaccuser、delaccuserサブコマンド
変更した権限設定を反映する	「14.4 変更した権限設定を反映する」	－

10.4. 権限グループの設定例

権限グループおよびアクセス許可の設定例を、実際に想定されるジョブネットワーク構築ポリシーに合わせて紹介します。

10.4.1. シナリオ1

1. ポリシー

■複数の開発者で、一連のジョブネットワークの開発を行う。

■運用を行うユーザには、ジョブネットワークの作成や編集を禁止する。（ユーザ制限の例は、シナリオ2に示します）

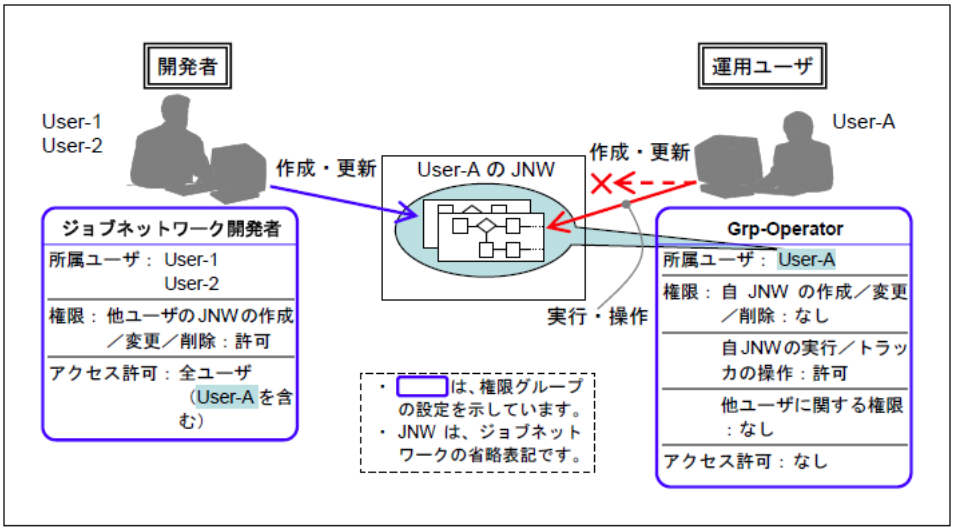

図10.2 シナリオ1のイメージ

2. ユーザの設定

a. ジョブネットワークを所有させるユーザ（以降、User-A）を作成します。

b. ジョブネットワークを開発するユーザを作成します。

ここでは、User-1、User-2の2つを作成した場合の例を説明します。

同一のユーザアカウントで、複数の人が同時にJobCenter　CL/Winにログインして編集できるので、ユーザのアカウントは1つでも可能です。開発者ごとの環境やセキュリティが必要であればユーザアカウントを分けてください。

ユーザの作成とは、ユーザアカウントを作成し、そのユーザアカウントで、一度、JobCenter CL/Winからログインおよびログアウトする操作を示します。

3. 権限グループの設定

a. User-1とUser-2を「ジョブネットワーク開発者」に所属させます。

b. 「一般ユーザ」を複製して独自の権限グループ（ここでは、Grp-Operator）を作成し、「ユーザ自身のジョブネットワークの作成／変更／削除」の権限を削除します。

c. User-Aを図10.2「シナリオ1のイメージ」で作成した権限グループGrp-Operatorに所属させます。

上記の操作については<基本操作ガイド>の「14.1　権限グループを設定する」　<基本操作ガイド>の「14.2 権限グループを作成する」をそれぞれ参照してください。

4. 開発時

■開発者は、User-1またはUser-2のユーザ名でJobCenter CL/Winからログインします。

■他ユーザはUser-Aのユーザフレームを開き、ジョブネットワークを開発します。

5. 運用時

■運用者はUser-Aのユーザ名でJobCenter　CL/Winからログインして、自分のユーザフレームからジョブの投入や実行監視を行います。

ジョブネットワークを開発するマシンと運用するマシンが別であればUser-Aのみ使用し、開発マシンでは「ジョブネットワーク開発者」に所属させ、運用マシンでは、前述の独自の権限グループに所属させることでも可能です。

または、開発時には「ジョブネットワーク開発者」を複製した独自の権限グループに所属させ、運用時にはこの権限グループの権限を縮小することでも可能です。

10.4.2. シナリオ2

1. ポリシー

■複数の開発者でジョブネットワークを利用した大規模なアプリケーション群（以下、AP群）の開発を行う。

■各開発者には、特定のAP群（以下、AP群1, AP群）のジョブネットワークのみアクセスさせる。

■運用を行うユーザは、全てのAP群のジョブネットワークが実行可能とする。

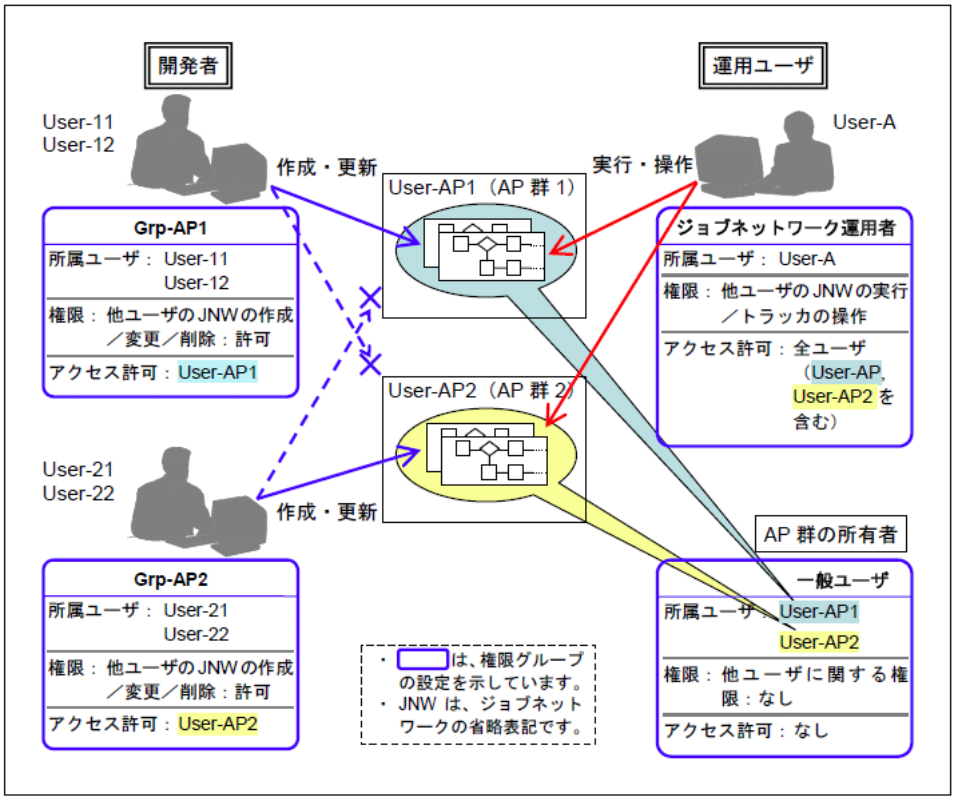

図10.3 シナリオ2のイメージ

2. ユーザの設定

a. 各AP群に対応するユーザ（以下 User-AP1、User-AP2）を作成します。

b. ジョブネットワークを開発するユーザ（以下 User-11、User-12、User-21、User-22）を作成します。

c. 運用を行うユーザ（以下 User-A）を作成します。

> ユーザの作成とは、ユーザアカウントを作成し、そのユーザアカウントで、一度、JobCenter CL/Winからログインおよびログアウトする操作を示します。

3. 権限グループの設定

a. 各AP群に対応した権限グループ（以下　Grp-AP1、Grp-AP2）を、「ジョブネットワーク開発者」を複製して作成します。1つは「ジョブネットワーク開発者」をそのまま使用しても構いません。

b. 権限グループGrp-AP1のアクセスを許可するユーザにUser-AP1を、Grp-AP2のアクセスを許可するユーザにUser-AP2をそれぞれ設定します。

c. 権限グループGrp-AP1にUser-11とUser-12、Grp-AP2にUser-21とUser-22をそれぞれ所属させます。

d. User-Aを「ジョブネットワーク運用者」に所属させます。

e. この状態でUser-AP1とUser-AP2は「一般ユーザ」に所属しています。User-AP1とUser-AP2からログインしたときに修正などを禁止するには、権限を縮小した独自の権限グループを作成し、そこに所属させます。

上記の操作については<基本操作ガイド>の「14.2　権限グループを作成する」　<基本操作ガイド>の「14.3　他ユーザへのジョブネットワーク、トラッカのアクセス許可を制限する」をそれぞれ参照してください。

4. 開発時

■User-11、User-12のユーザ名でCL/Winからログインすると、User-AP1に対応するAP群のみ開発が可能です（他ユーザであるUser-AP1のフレームで開発します）。

■User-21、User-22のユーザ名でCL/Winからログインすると、User-AP2に対応するAP群のみ開発が可能です（他ユーザであるUser-AP2のフレームで開発します）。

5. 運用時

■運用者はUser-Aのユーザ名でJobCenter　　CL/Winからログインして、User-AP1、User-AP2それぞれのユーザフレームからジョブの投入や実行監視を行います。

「シナリオ2」の例は単純化するためAP群の数は2つで、開発の分割単位とAP群の単位が1対1に対応しています。

実際は各AP群はアクセス制限を設けたい単位に応じて分割し、それに対応するユーザをジョブネットワークの所有者として作成します。

また、アクセス先の各AP群の種類により複数の権限グループを作成し、そこに1つまたは複数の各AP群に対応するユーザへのアクセス許可を設定します。

各開発者用には権限グループの単位で1つまたは個人ごとのユーザを作成し、アクセス先の各AP群の種類に応じた権限グループに所属させます。

10.5. 権限設定の注意事項

10.5.1. 権限グループに所属させるユーザやアクセス先のユーザ

権限グループに所属させるユーザや、アクセス先として許可するユーザとして、設定が可能なユーザは、JobCenter CL/Winからログインしたことのあるユーザが対象となります。

事前に、JobCenterを使用する各ユーザでログイン（およびログアウト）操作を行ってから、設定してください。

10.5.2. スケジュールによる投入やコマンドによる操作

アクセス権限の設定は、JobCenter CL/Winから操作した場合に有効です。

スケジュールによるジョブネットワークの投入、およびjnwsubmitcmdやjnwopコマンドによるジョブネットワークの投入やトラッカの操作は、アクセス権限の設定によらず実行可能です。

10.5.3. ログインユーザで行う必要がある操作

1. ジョブネットワークのパラメータのデフォルト値

   ジョブネットワークおよび各部品のパラメータのデフォルト値は、そのジョブネットワークを所有するユーザのパラメータデフォルト値が使用されます。

   パラメータデフォルト値による設定を使用する場合は、あらかじめ、ジョブネットワークを所有するユーザでログインしてパラメータのデフォルト値を設定してください（<基本操作ガイド>の「3.3.4 ジョブネットワークのパラメータを設定する」参照）。

2. ジョブネットワークの実行時の環境変数の設定

   他ユーザのジョブネットワークの実行時に起動されるジョブの環境変数は、ジョブネットワーク所有者に設定された環境が使用されます。

   他ユーザのジョブネットワークを実行する場合や他ユーザのスケジュールを設定する場合は、実行前にそのジョブネットワークを所有するユーザで最低一度はログインしてください。

   またUNIX版の場合、ジョブネットワークの実行時の環境変数は、ログインユーザ自身で自分のスケジュールの変更操作を行った場合のみ更新されます。

   ジョブネットワーク実行時の環境変数を設定する場合は、ジョブネットワークを所有するユーザでログインして「15.1.3 環境変数の設定方法」の注意事項に記述されているスケジュール変更操作により設定してください。

11. ディレクトリサービス連携によるユーザー管理

JobCenter MG/SVの一部プラットフォームでは、ディレクトリサービスによる一般アカウントのユーザー管理に対応しています。 プラットフォームによって、対応する機能の範囲に以下の違いがあります。

■Windows

- ドメイン環境においてJobCenter管理者としてドメインユーザを利用することで、ドメインユーザをJobCenter利用者ユーザとして利用する事ができます。ただし、JobCenterでジョブ実行を行う場合は、ドメインユーザのパスワードをJobCenterで別途管理する必要があります。

- JobCenterのユーザ権限を管理するための権限グループと、ディレクトリサービスのグループを対応付けることで、ディレクトリサービスを用いてJobCenterのユーザ権限を管理することができます。

  連携方法の詳細は、「11.1 LDAPサーバ連携(Windows)」 で説明します。

■Linux

Name Service Switch(NSS) および Pluggable Authentication Modules (PAM) がディレクトリサービスを利用するように設定することで、 ディレクトリサービスによって管理されているユーザを JobCenter利用者ユーザとして利用する事ができます。

管理者ユーザ(nsumsmgrユーザ)は、ディレクトリサービスによる管理に対応していません。ローカルサーバ上にnsumsmgrユーザを作成する必要があります。

連携方法の詳細は、「11.2 LDAPサーバ連携 (Linux)」 で説明します。

■その他のプラットフォーム

- Windows、 Linux 以外のプラットフォームではディレクトリサービス連携によるユーザ管理をサポートしていません。

11.1. LDAPサーバ連携(Windows)

本章は、JobCenterのユーザ権限を一元管理する方法について説明します。

JobCenterのユーザ数が増加していくと、ユーザ権限の管理が煩雑になります。本連携機能を利用することで、ユーザ数が増減してもユーザ権限管理が容易になり、高い拡張性や運用性を実現することができます。

本連携機能では、以下のようなことが実現できます。

■LDAPサーバ側の権限グループの追加・削除

JobCenter側のパーミッション設定の権限グループを追加・削除した場合、LDAPサーバ側の権限グループの追加・削除ができます。また、LDAPサーバ側の権限グループにドメインユーザを追加することでユーザ権限を付与することができます。

■ドメインユーザのユーザ権限の変更

ドメインユーザのユーザ権限を変更することができます。ユーザ権限の変更は、JobCenter側のパーミッション設定、およびLDAPサーバの両方から変更することができます。

連携可能なLDAPサーバは、Active Directoryのみ利用できます。

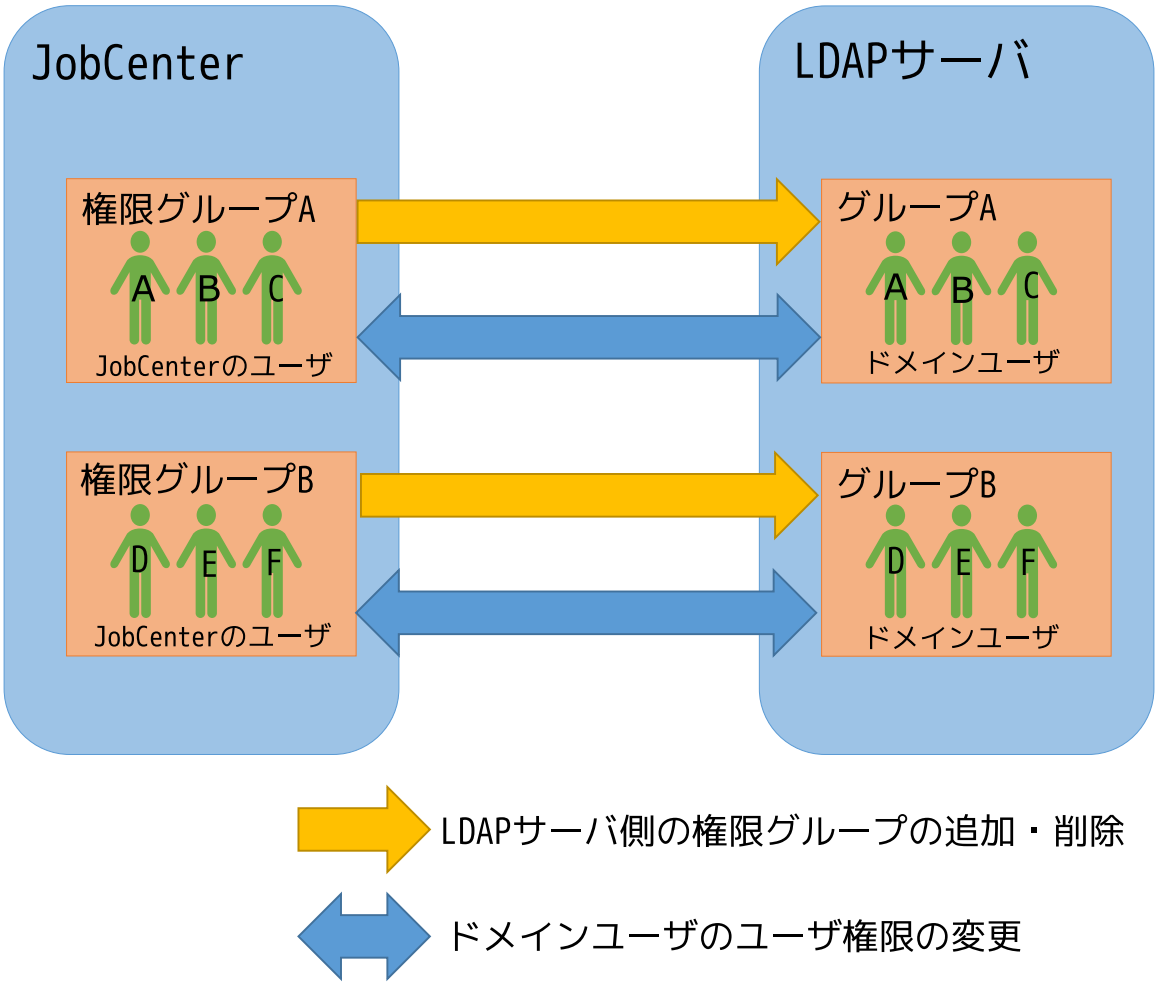

図11.1 LDAPサーバ連携機能の概要

11.1.1. LDAPサーバ連携の設定

LDAPサーバ連携機能を利用する場合、ドメイン環境、かつJobCenter管理者がドメインユーザであることを確認した後に設定を行ってください。

JobCenter管理者は、JobCenterをインストールする時に指定します。詳細は、以下を参照してください。

■<インストールガイド>の「2.4.4 Windows版 (通常インストール)」

■<インストールガイド>の「2.4.5 Windows版 (サイレントインストール)」

LDAPサーバと連携する手順は、以下の通りです。

1. JobCenter側にLDAPサーバ設定を行う
2. LDAPサーバ側の権限グループにドメインユーザを追加する

11.1.1.1. JobCenter側のLDAPサーバ設定

LDAPサーバ設定を行うサイトが「起動」状態の時に設定してください。

サーバの環境設定から［サイト］－［<サイト名>］－［プロパティ］を選択し、「LDAPサーバ設定」タブに連携するLDAPサーバの情報を設定します。

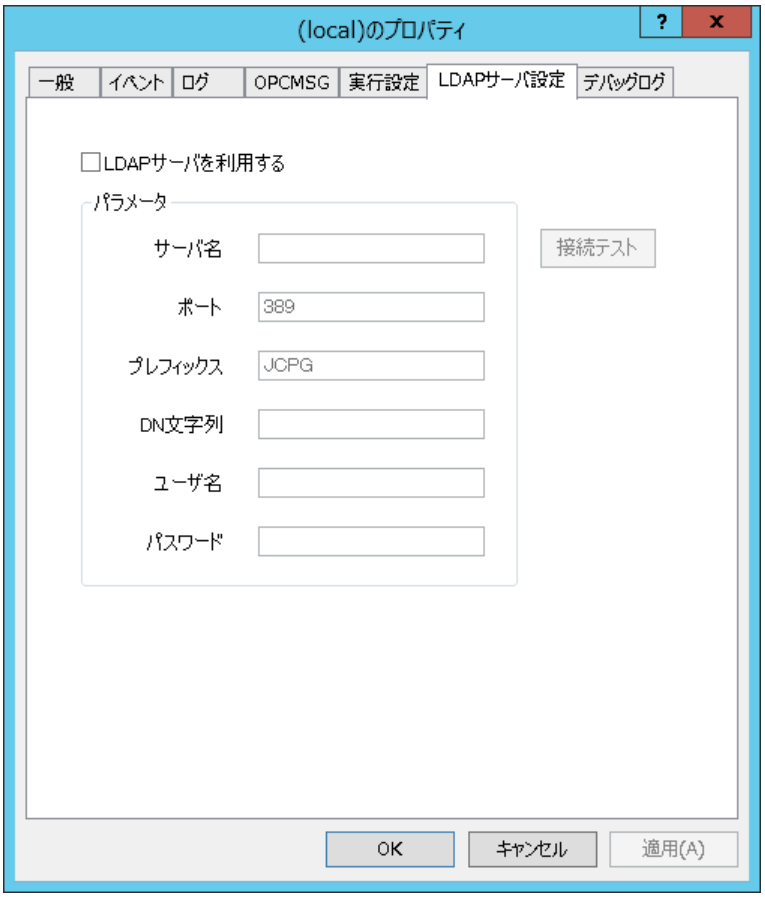

図11.2 LDAPサーバ設定画面

LDAPサーバ設定の各項目は、以下で説明します。

表11.1 LDAPサーバ設定項目

設定項目	設定値
LDAPサーバを利用する	利用する場合は、チェックボックスにチェックを入れる。 チェックを入れると、下記の設定項目が入力できるようになります。
サーバ名	連携先のLDAPサーバを指定します。 指定方法は、以下から選択する事ができます。 ■ドメイン名 ■ホスト名 ■IPアドレス 連携先は、JobCenterをインストールしたマシンが所属するドメインと同じ連携先を設定してください。 連携するLDAPサーバは、全てのサイトで同じ連携先を設定してください。
ポート	LDAPサーバに接続するポート番号を設定します。 デフォルト値は、389
プレフィックス	LDAPサーバ側の権限グループ名に使用するプレフィックスを設定します。 英数字、アンダーラインのみ使用でき、最大５文字まで設定できます。デフォルト値は、JCPG
DN文字列	以下のユーザが所属するオブジェクトの識別名を設定します。 ■LDAPサーバ設定の設定項目「ユーザ名」で指定したユーザ ■JobCenter側でユーザ権限を付与したいドメインユーザ 指定したオブジェクトにLDAPサーバ側の権限グループが作成されます。
ユーザ名	LDAPサーバの管理者権限を付与したLDAPサーバのドメインユーザを設定します。 DN文字列で設定したオブジェクトに所属するユーザを設定してください。
パスワード	ユーザ名で指定したユーザのパスワードを設定します。 ディレクトリサービスでパスワードを変更した場合、本項目も変更してください。

JobCenter側のLDAPサーバ設定の完了後、LDAPサーバ側の権限グループが自動的に作成されます。対応関係は、以下になります。

表11.2 JobCenter側の権限グループとLDAPサーバ側の権限グループの対応表

JobCenter側の権限グループ名	LDAPサーバ側の権限グループ名
ジョブネットワーク開発者	<GRP_PREFIX>_DEVELOPER
ジョブネットワーク運用者	<GRP_PREFIX>_OPERATOR
実行監視者	<GRP_PREFIX>_SURVEILLANT
一般ユーザ	<GRP_PREFIX>_COMMONUSER

JobCenter側の権限グループ名	LDAPサーバ側の権限グループ名
<ユーザが定義した権限グループ名>	<GRP_PREFIX>_<ユーザが定義した権限グループ名>

JobCenter側のLDAPサーバ設定の完了後に権限グループの追加・削除した場合、LDAPサーバ側の権限グループを自動的に追加・削除します。

11.1.1.2. LDAPサーバの設定

LDAPサーバで以下を実施します。

1. LDAPサーバにJobCenterにログインするドメインユーザを追加する。
2. LDAPサーバ側の権限グループに作成したドメインユーザを追加する。

11.1.2. LDAPサーバ連携の設定例

Active Directoryとの連携を例として、LDAPサーバ連携の方法を説明します。

11.1.2.1. JobCenterとActive Directoryとの連携の構築

JobCenterとActive Directoryとの連携を行うための構築手順について説明します。

1. LDAPサーバ連携の設定

   LDAPサーバ連携の設定を行うサイトが「起動」状態であることを確認する。

   JobCenter側のサーバの環境設定から［サイト］-［<サイト名>］-［プロパティ］を選択し、「<サイト名>のプロパティ」画面のLDAPサーバ設定タブから以下のように設定します。

表11.3 LDAPサーバ連携の設定例

設定項目	設定値
LDAPサーバを利用する	チェックボックスにチェックを入れる
ホスト名	ldap.example.co.jp
ポート	389
プレフィックス	JCPG
DN文字列	CN=Users,DC=example,DC=co,DC=jp
ユーザ名	Administrator
パスワード	<Administratorのパスワード>

DN文字列は、連携するActive Directoryで以下のコマンドを実行し、ユーザ名(例：CN=<ユーザ名>)を除いた文字列を指定します。

```
dsquery user -name <ユーザ名>
```

LDAPサーバと接続できる事を確認するため、「接続テスト」ボタンを押し、「LDAPサーバへの接続成功」と表示されることを確認してください。

LDAPサーバと接続できることを確認後、「OK」ボタン、または「適用」ボタンを押して設定を保存してください。

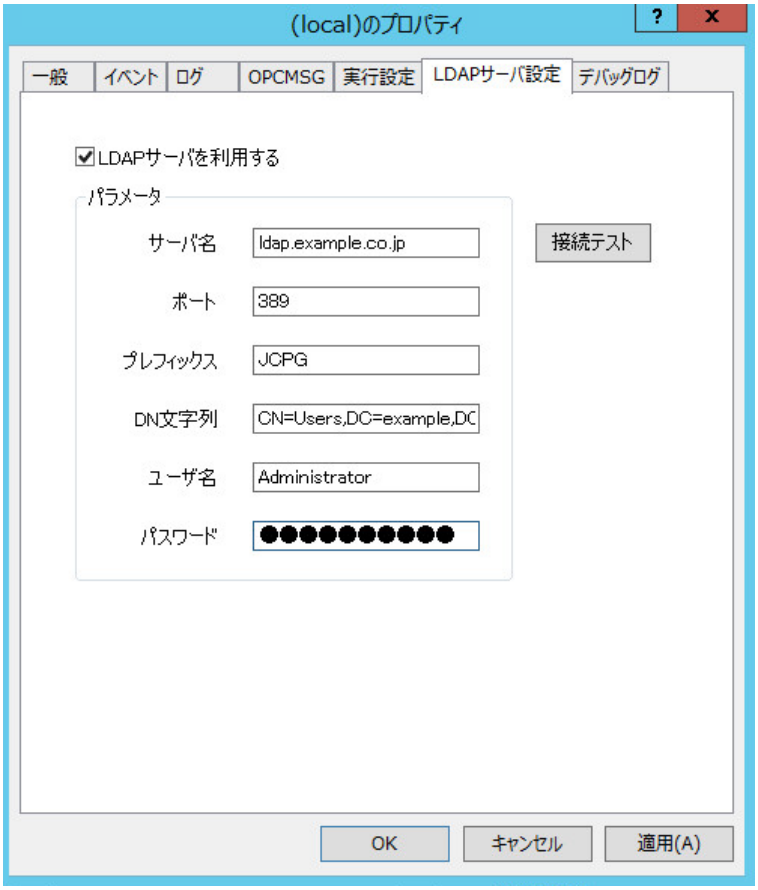

図11.3 JobCenter側のLDAPサーバ設定例

2. Active Directoryの設定

Active Directoryで設定する方法を説明します。

本手順で扱うドメインユーザ、およびドメインユーザが所属する権限グループは、以下の通りです。

表11.4 ユーザの権限グループ

ドメインユーザ名	権限グループ名	LDAPサーバ側の権限グループ名
dev_user_1	ジョブネットワーク開発者	JCPG_DEVELOPER
op_user_1	ジョブネットワーク運用者	JCPG_OPERATOR

a. ドメインユーザの作成

ジョブネットワーク開発者、およびジョブネットワーク運用者で利用するドメインユーザ(dev_user_1、op_user_1)を作成するため、以下の方法でドメインユーザを作成します。

■GUIの場合、「Active Directory ユーザーとコンピューター」から作成します。

■CUIの場合、dsadd userコマンドで作成します。

dev_user_1をUsersコンテナに作成する場合、連携するActive Directoryで以下のコマンドを実行してください。

CNにユーザの情報、DCにドメインの情報を指定します。

```
dsadd user CN=dev_user_1,CN=Users,DC=example,DC=co,DC=jp
```

b. LDAPサーバ側の権限グループにドメインユーザの追加

作成したドメインユーザにJobCenter側のユーザ権限を与えるため、以下の方法でジョブネットワーク開発者グループ(JCPG_DEVELOPER)、およびジョブネットワーク運用者グループ(JCPG_OPERATOR)にドメインユーザを追加します。

■GUIの場合、「Active Directory ユーザーとコンピューター」からグループに追加します。

■CUIの場合、dsmod groupコマンドで追加します。

ジョブネットワーク開発者グループ(JCPG_DEVELOPER)にdev_user_1を追加する場合、連携するActive Directoryで以下のコマンドを実行してください。

ジョブネットワーク開発者グループ(JCPG_DEVELOPER)を指定し、-addmbrに追加するドメインユーザを指定します。

```
dsmod group CN=JCPG_DEVELOPER,CN=Users,DC=example,DC=co,DC=jp -addmbr
 CN=dev_user_1,CN=Users,DC=example,DC=co,DC=jp
```

11.1.2.2. ドメインユーザのユーザ権限の変更

LDAPサーバ連携の設定後、ドメインユーザのユーザ権限の変更は、以下の２つの方法で変更することができます。

■LDAPサーバ側でユーザ権限を変更する

1. 変更元のLDAPサーバ側の権限グループからユーザ権限を変更したいドメインユーザを削除する。
2. ユーザ権限を変更したいドメインユーザを変更先のLDAPサーバ側の権限グループに追加する。

■JobCenter側でユーザ権限を変更する

1. JobCenterにCL/WinでJobCenter管理者としてログインする。
2. メニューバーの［設定］－［パーミッション設定］を選択します。
3. ユーザ権限を変更したいドメインユーザを変更先の権限グループに移動する。

11.1.2.3. JobCenter側のLDAPサーバ設定の変更

LDAPサーバ連携の設定後、JobCenter側のLDAPサーバ設定を変更する場合、以下のような手順で変更します。

■LDAPサーバ連携を解除する

LDAPサーバ連携を解除する場合、LDAPサーバ設定の各項目を変更せず、「LDAPサーバを利用する」のチェックのみ外し、「適用」ボタンを押す。

■LDAPサーバ設定を変更する

連携先のLDAPサーバを変更する場合、以下の手順で設定を変更してください。

1. LDAPサーバ設定の各項目を変更せず、「LDAPサーバを利用する」のチェックのみ外し、「適用」ボタンを押す。

2.「LDAPサーバを利用する」にチェックを入れ、設定項目を変更し、「適用」ボタンを押す。

以下のLDAPサーバ設定を変更した場合、変更前のLDAPサーバ側の権限グループは残存します。

■サーバ名

■プレフィックス

11.1.2.4. LDAPサーバとの連携エラーの回避

複数のドメインコントローラが連携している場合、ドメインコントローラ間の同期に時間がかかる環境に対してLDAPサーバ連携を行うと、連携エラーが発生することがあります。この連携エラーを回避したい場合、以下のファイルにリトライ期間を設定する事で回避する事ができます。

■設定ファイル

<サイトDBパス>\spool\conf\check_member_retry.conf

本ファイルは、新規作成する必要があります。

■設定項目

設定ファイルに以下のパラメータを記述する事で設定が有効になります。

表11.5 LDAPサーバ連携のリトライ設定

パラメータ名	説明	デフォルト値	最小設定値	最大設定値	単位
RetryTimes	リトライ回数	5	5	100	回
RetryInterval	リトライ間隔	1000	100	60000	ミリ秒

未設定、設定値の範囲外、設定値が不正な場合、デフォルト値が設定されます。

■設定反映条件

設定ファイルの保存後、次回LDAPサーバ設定の保存時から反映されます。

■設定例

```
RetryTimes 10
RetryInterval 1000
```

11.1.2.5. ドメインユーザのユーザ権限の剥奪

特定のドメインユーザのユーザ権限を剥奪したい場合、以下の方法で実現することができます。

■LDAPサーバでドメインユーザを削除する

■LDAPサーバでドメインユーザを無効化する

■JobCenterで何も権限のない権限グループを作成し、そのグループにドメインユーザを追加する。

ドメインユーザは、JobCenterにログイン以外の操作ができない状態となります。

11.1.3. LDAPサーバ連携の注意事項

LDAPサーバと連携する場合の注意事項を以下に説明します。

11.1.3.1. ドメインユーザに関する注意事項

■Active Directoryのユーザログオン名

Active Directoryにユーザを追加する際は、フル ネームとユーザログオン名（Windows 2000 より前）を一致させる必要があります。

一致してない場合、Active　Directoryを参照できないため、正しいユーザ権限が付与されません。　この場合、ドメインユーザのユーザ権限は、一般ユーザが付与されます。

■入れ子構造のグループに所属するドメインユーザのユーザ権限

LDAPサーバの入れ子構造のグループに所属するドメインユーザは、ユーザ自身が直接所属するグループのユーザ権限のみを持ちます。

また、ドメインユーザが直接所属するグループ名が<GRP_PREFIX>_で始まらない場合、ユーザ権限は一般ユーザが付与されます

■２つ以上のグループに所属するドメインユーザのユーザ権限

LDAPサーバの２つ以上の権限グループに所属するドメインユーザは、ユーザ権限が不定となります。

１つのドメインユーザに同一サイトの複数のユーザ権限を付与しないようにしてください。

■DN文字列で指定したオブジェクト以外に所属するドメインユーザのユーザ権限

DN文字列で指定したオブジェクト以外に所属するドメインユーザは、LDAPサーバ側の権限グループに所属しても正しいユーザ権限が付与されません。

この場合、ドメインユーザのユーザ権限は、一般ユーザが付与されます。

■ドメインユーザを大量に追加する場合

JobCenter側の権限グループに大量のドメインユーザを追加した場合、追加後にJobCenter側を起動すると、起動に時間がかかる場合があります。

11.1.3.2. 権限グループに関する注意事項

■権限グループ名に利用できない文字

Active　Directoryでは「+」をグループ名に利用できないため、「^」に変換して、グループ名を作成します。

■LDAPサーバ側の権限グループの操作

LDAPサーバ側の権限グループの追加・削除をLDAPサーバ上で行わないでください。　LDAPサーバ側の権限グループの追加・削除は、CL/Winのパーミッション設定から行ってください。

11.1.3.3. 複数サイトのLDAPサーバ連携に関する注意事項

■1つのLDAPサーバで複数のサイトを管理する場合、サイト毎にプレフィックスを分ける必要があります。

■各サイトのLDAPサーバ連携は独立しているため、サイト毎に設定を行ってください。

11.1.4. LDAPサーバ連携の制限事項

LDAPサーバと連携する場合の制限事項を以下に説明します。

11.1.4.1. 連携先ドメインに関する制限事項

■LDAPサーバ設定の[サーバ名]で設定する連携先は、JobCenterをインストールしたマシンが所属するドメインと同じ連携先を設定してください。異なるドメインの指定はサポートしていません。

■LDAPサーバ連携機能は、信頼関係のあるドメインはサポートしていません。

11.2. LDAPサーバ連携 (Linux)

Linux版 JobCenter(MG/SV)は、OSの Name Service Switch(NSS) および Pluggable Authentication Modules (PAM) でディレクトリサービスを利用するように設定することで、 ディレクトリサービスによって管理されているユーザを JobCenter 一般アカウントユーザとして利用する事ができます。 本章では Linux版 JobCenter(MG/SV)がディレクトリサービスと連携するために必要な設定の方法について説明します。

11.2.1. ディレクトリサービス連携の動作概要

Name Service Switch(NSS) の設定により、passwd, group および shadow データベースがディレクトリサービスを利用することで、 ディレクトリサービスによって管理されているユーザを利用して単位ジョブの実行を行なうことができます。

Pluggable Authentication Modules (PAM) の設定により、 ディレクトリサービスによって管理されているユーザを利用してCL/Win, CL/Webからのログインを行なうことができます。

11.2.2. ディレクトリサービス連携の設定

この節では、ディレクトリサービスによって管理されているユーザをJobCenterで利用するための設定方法を説明します。 各設定例は、JobCenterからユーザ情報を利用するために最小限必要な設定を記載しています。 ディレクトリサービスによって管理されているユーザをOSやJobCenter以外のアプリケーションからも利用する場合は、それぞれの連携に必要な設定を追加または変更してください。

この節で例示する設定ファイルおよびコマンドでは、連携するディレクトリサービスの各種パラメータが以下のような場合を例として説明します。 各パラメータを使用している設定箇所を、お客様の環境に合わせて読み替えてください。

ディレクトリサービスホスト名	ldap.example.com
OU (Organizational Unit)	dc=example,dc=com

11.2.2.1. SSSD によるディレクトリサービス連携の設定例

System Security Services Daemon (SSSD) は、認証プロバイダーへのアクセスを提供します。 本節では、認証プロバイダーとして ldap を指定した SSSD の設定例を説明します。

1. SSSD および依存パッケージのインストール

   SSSDの設定のため以下のパッケージおよび、それぞれが依存するパッケージをインストールしてください。

   ■sssd

   ■sssd-client

   ■sssd-client.i686

   ■sssd-ldap

   ■openldap-clients

   > Linux版のJobCenter(MG/SV)は64bitモジュールと32bitモジュールの両方が存在するため、64bit環境でも32bit用のsssd-client.i686が必要です。

2. LDAP サーバとの連携設定

   LDAP サーバとの連携を、 authconfig コマンドを用いて設定するか、 /etc/sssd/sssd.conf を直接編集することで適切に設定してください。

   authconfig による設定例

```
# authconfig --enablesssd --enablesssdauth --enablelocauthorize --disableldap --disableldapauth
 --disableldaptls --update
```

/etc/sssd/sssd.conf の設定例

```
[sssd]
debug_level         = 0
config_file_version = 2
services            = nss, pam, ssh, sudo
domains             = default

[domain/default]
enumerate = true

id_provider     = ldap
auth_provider   = ldap
chpass_provider = ldap
sudo_provider   = ldap

ldap_uri              = ldap://ldap.example.com/
ldap_search_base      = dc=example,dc=com
ldap_id_use_start_tls = False

ldap_search_timeout              = 3
ldap_network_timeout             = 3
ldap_opt_timeout                 = 3
ldap_enumeration_search_timeout  = 60
ldap_enumeration_refresh_timeout = 300
ldap_connection_expire_timeout   = 600

ldap_sudo_smart_refresh_interval = 600
ldap_sudo_full_refresh_interval  = 10800

entry_cache_timeout = 1200
cache_credentials   = True
ldap_tls_reqcert = never

[nss]

homedir_substring = /home
entry_negative_timeout        = 20
entry_cache_nowait_percentage = 50

[pam]

[sudo]

[autofs]

[ssh]

[pac]

[ifp]
```

> /etc/sssd/sssd.confの設定は環境、用途によって異なります。ldap_uriおよびldap_search_baseに、連携するLDAPサーバ及びOUを設定してください。

設定反映のため sssd を再起動します。

RHEL 7.x 系以降の場合

```
# systemctl restart sssd
```

RHEL 6.x 系以前の場合

```
# service sssd restart
```

11.2.2.2. nslcd によるディレクトリサービス連携の設定例

local LDAP name service daemon (nslcd) は、NSS または PAM を通して ディレクトリサービスを参照したいローカルプロセスに代わって LDAP アクセスを提供するデーモンです。 本節では、nslcd を利用して LDAP 連携を行なう場合の設定例を説明します。

1. nslcd および依存パッケージのインストール

   nslcd の設定のため以下のパッケージおよび、それぞれが依存するパッケージをインストールしてください。

   ■nscd

   ■nss-pam-ldapd

   ■nss-pam-ldapd.i686

   ■openldap-clients

   > Linux版のJobCenter(MG/SV)は64bitモジュールと32bitモジュールの両方が存在するため、64bit環境でも32bit用のnss-pam-ldapd.i686が必要です。

2. LDAP サーバとの連携設定

   LDAP サーバとの連携を、 authconfig コマンドを用いて設定するか、 /etc/openldap/ldap.conf および /etc/nslcd.conf を直接編集することで適切に設定してください。

   authconfig による設定例

```
# authconfig --enableldap --enableldapauth --ldapserver=ldap.example.com --
ldapbasedn="dc=example,dc=com" --update
```

   /etc/openldap/ldap.conf の設定例

```
URI ldap://ldap.example.com
BASE dc=example,dc=com
```

   /etc/nslcd.conf の設定例

```
uri ldap://ldap.example.com
base dc=example,dc=com
```

   設定反映のため nslcd を再起動します。

   RHEL 7.x 系以降の場合

```
# systemctl restart nslcd
```

RHEL 6.x 系以前の場合

```
# service nslcd restart
```

11.2.3. JobCenter ログインユーザとしてのLDAP連携設定

LDAPユーザでCL/Win, CL/Webログインを行なうためには、/etc/pam.d/jccombaseのPAM構成ファイルを作成し、認証モジュールとしてLDAPが利用されるように設定する必要があります。 また、daemon.conf ファイルにPAM認証を利用するための設定を追加する必要があります。

1. /etc/pam.d/jccombase の設定例

```
#%PAM-1.0
auth        required      pam_env.so
auth        sufficient    pam_unix.so try_first_pass nullok
auth        sufficient    pam_ldap.so use_first_pass
auth        required      pam_deny.so
```

> /etc/pam.d/jccombase は auth モジュールインタフェースのみを利用します。 account, passwd, session モジュールインタフェースの設定は不要です。

daemon.conf の設定

```
COMAGENT_AUTH_PAM=1
```

> daemon.conf 編集後、設定を反映するためにJobCenterの再起動が必要です。

11.2.4. JobCenter ジョブ実行ユーザとしてのLDAP連携設定

1. NSS データベースの LDAPディレクトリサービス連携設定

   NSS データベース passwd, group および shadow が LDAP と連携するように /etc/nsswitch.conf を編集してください。

   /etc/nsswitch.conf の設定例

```
passwd:     files ldap
shadow:     files ldap
group:      files ldap
hosts:      files dns
bootparams: nisplus [NOTFOUND=return] files
ethers:     files
netmasks:   files
networks:   files
protocols:  files
rpc:        files
services:   files
netgroup:   files
publickey:  nisplus
automount:  files
aliases:    files nisplus
```

> /etc/nsswitch.conf の設定は環境、用途によって異なります。 passwd, group, shadow の 検索対象として ldap が含まれるように設定してください。

2. NSS データベースの LDAPディレクトリサービス連携確認

   getent コマンドで LDAP との連携ができている事を確認します。

```
# getent passwd jcuser

jcuser:x:1777:1777:jcuser:/home/jcuser:/bin/bash
```

11.2.5. 注意事項、制限事項

■LDAPサーバ上のユーザ関連エントリに関する制限事項

- LDAPデータベース上のユーザエントリは少なくとも top, posixAccount, shadowAccount の objectclass に属している必要があります。
- LDAPデータベース上のグループエントリは少なくとも top, posixGroup の objectclass に属している必要があります。

■LDAP連携ユーザをJobCenter(MG/SV)で利用する際の注意事項

- ジョブ実行ユーザとしてLDAP連携するユーザは、ジョブ実行時にそのユーザのホームディレクトリが存在し、そのユーザの権限でホームディレクトリに移動できる必要があります。 事前にホームディレクトリを作成しておくか、automount 等を利用して自動的に作成される様に設定する必要があります。

> authconfig --enablemkhomedir --update や、 pam_mkhomedir.so の設定によってユーザログイン時のホームディレクトリ作成処理を行っても、 JobCenter(MG/SV)へのログインおよびジョブ実行の処理時に、ホームディレクトリは自動作成されません。

12. イベント連携

本章ではイベント連携機能の利用方法について説明します。

12.1. イベント連携の概要

統合監視サーバにてJobCenterのジョブの実行状況を監視したい場合、本機能を利用することで実現することができます。

イベント連携機能は、JobCenterで発生するイベント(例えば、単位ジョブの開始・終了)を　統合監視サーバ側で監視できるように監視対象へのイベント出力、または統合監視サーバへのイベント通知を行う機能です。

イベント連携の概要図は、図12.1「UNIX版のイベント連携のイメージ」、　および図12.2「Windows版のイベント連携のイメージ」を参照してください。

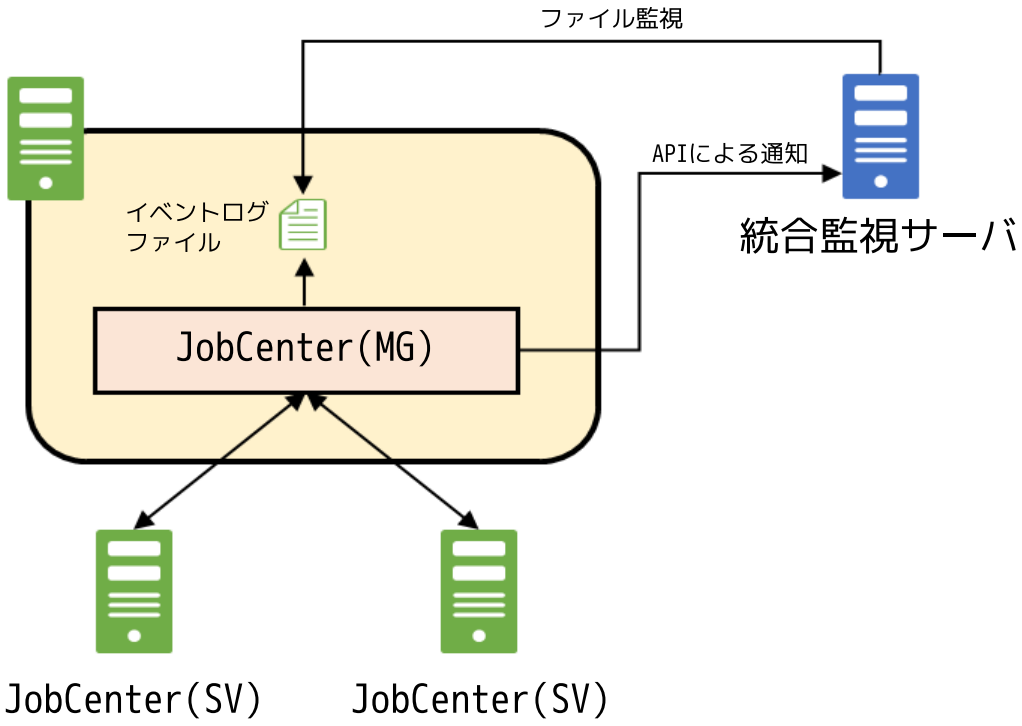

図12.1 UNIX版のイベント連携のイメージ

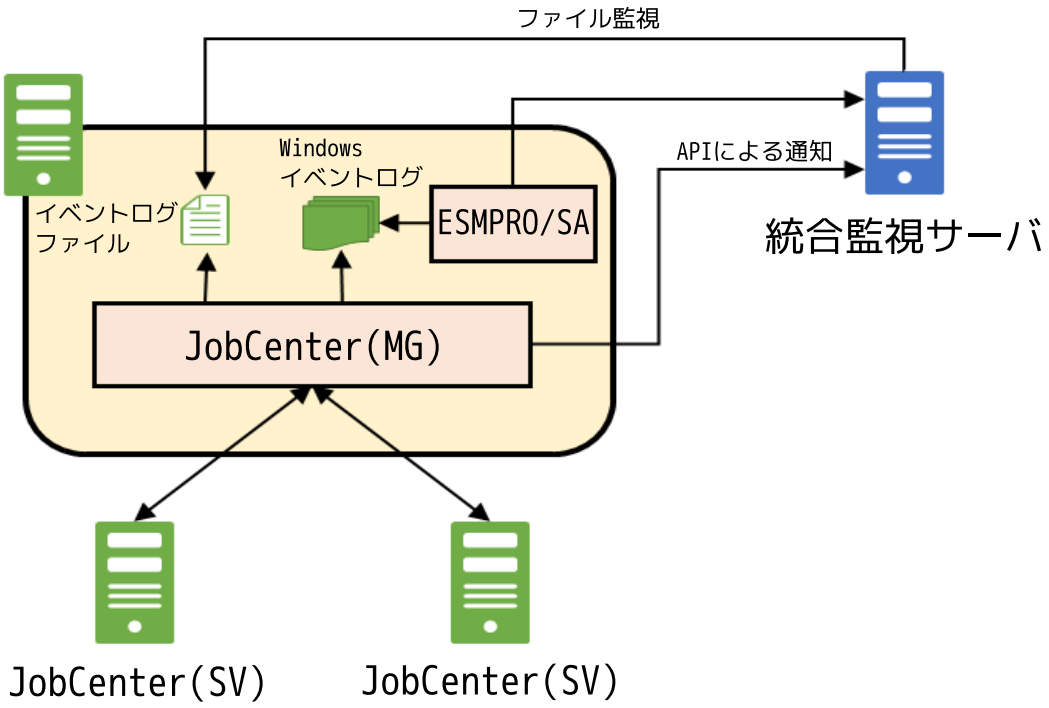

図12.2 Windows版のイベント連携のイメージ

イベントログファイルによるイベント連携時の統合監視ソフトの例としては、MCOperations、SystemManagerがあります。 統合監視ソフトのファイル監視の設定方法などは、統合監視ソフトのマニュアルを参照してください。

■UNIX版JobCenterとWindows版JobCenter間を接続する場合は、「2.2 ネットワーク接続を構築する際の注意事項」を参照してください。

■JobCenterのイベントは、該当するジョブネットワーク定義が存在するマシン(JobCenter (MG))にて出力します。従って、ジョブを転送して実行する場合、転送先のマシン(JobCenter (SV))ではイベントの出力はできません。

イベント連携方法の種類としては、以下の３つがあります。

1. JobCenterのイベントを指定のイベントログファイルに出力する
2. JobCenterのイベントをWindowsイベントログに出力する
3. JobCenterのイベントを統合監視ソフトのAPIを利用して通知する

なお、UNIX版、Windows版でイベント連携方法のサポート範囲が異なります。

表12.1 イベント連携機能がサポートするイベント連携種別

JobCenterのイベントの出力、通知方法	UNIX版	Windows版
指定のイベントログファイルへ出力	○	○
Windowsイベントログへ出力	×	○
統合監視ソフトのAPIによる通知 注1	○	○

注1 対応している統合監視ソフトは、Micro Focus Operations Manager softwareのみ

12.2. UNIX版のイベント連携

UNIX版では、次のイベント連携ができます。

■JobCenterのイベントを指定のイベントログファイルに出力する場合は、「12.2.1 イベントログファイルへの出力方法」を参照してください。

■APIを利用してMicro Focus Operations Manager softwareのメッセージビューアにイベント送信する場合は、以下を参照してください。

「12.2.2 イベント設定ファイル」

「12.2.3 イベント定義ファイル」

12.2.1. イベントログファイルへの出力方法

JobCenterのイベントをイベントログファイルに出力するための設定方法について説明します。

1. イベント設定ファイルにて、イベントログファイルの出力設定を有効にします。イベント設定ファイルの詳細は、「12.2.2 イベント設定ファイル」を参照してください。

   1. LOGFILEにJobCenterのイベントの出力先ファイル名を指定します。

   2. LOGSIZEにイベントログファイルの最大サイズを指定します。

   3. 出力させたいJobCenterのイベント別にログ出力設定を有効にします。

      ログ出力の有効化は、送信イベント設定(EVENT)のLOGの値をONに変更する事で有効にできます。

      出力可能なJobCenterのイベント一覧は、「12.2.4 イベント一覧」を参照してください。

2. イベント設定ファイルを編集したJobCenter MGを再起動します。

12.2.2. イベント設定ファイル

■パス

ローカルサイト	/usr/spool/nqs/gui/jnwcaster.conf
クラスタサイト	共有ディスク上の <JobCenterDBパス>/nqs/gui/jnwcaster.conf

■説明

イベント出力デーモン（jnwcaster）の動作条件を本ファイルに記述します。デーモン起動時に本ファイルは読み込まれます。

■フォーマット

- ログファイル

```
LOGFILE <logfilename>
```

ログファイル名を記述します。ログファイルを指定すると、EVENTタグでLOGがONに指定されているイベントが記録されます。

ログファイル名は絶対パスで記述してください。相対パスで記述した場合は/usr/spool/nqs/gui（クラスタ環境の場合は共有ディスク上の<JobCenterDBパス>/nqs/gui）からの相対になります。ただし誤作動の原因になるため、/usr/spool/nqs（クラスタ環境の場合は共有ディスク上の<JobCenterDBパス>/nqs）配下に上記ログファイルを含め、ユーザのファイルを置かないようにしてください。

- JobCenter起動時に<logfilename>がアクセス不能な状態になっているとエラーとなり、起動に失敗します。十分注意してください。
- ログ出力される文字コードはJobCenterセットアップ時の文字コード設定に依存します。

■ ログファイルサイズ

```
LOGSIZE <size>
```

ログファイルのサイズを指定します。指定するサイズの単位はKbyteです。サイズの制限はlong型の最大値－1と、OSのファイルサイズ制限とを比較して、小さい方となります。0を指定した場合、ログは出力されません。

指定サイズを超えた場合、2世代分（<logfilename>, <logfilename>.bak）保存されます。バックアップ世代数は変更できません。さらに指定サイズを超えた場合、古い方のログファイルが上書きされます。

■ 送信インタフェース

```
EVENTIF {OPCMSG}
```

イベント送信のインタフェースを指定します。EVENTIFを指定する場合は、OPCMSGを必ず設定してください。

EVENTIFタグ	動　作
OPCMSG	Micro Focus Operations Manager softwareのOPCメッセージを利用したイベント送信を行います。

JobCenterを起動する前に連携モジュール(jnwopcr)をMicro Focus Operations Manager software(旧OVO)のバージョンに適したものに置き換えておく必要があります。 連携モジュールは保守契約先のNECサポートポータル、またはNECカスタマーサポートセンターより入手してください。(HP-UX版のみの提供です)

■ NQS送信設定

```
NQSEVENT {ON|OFF}
```

ジョブネットワークのイベントをNQS Daemonに送信するかどうかを設定します。JobCenter R5.1以上では常にOFFにしてください。

■ 送信イベント設定

```
EVENT <Event-Name> MESSAGE={ON|OFF} LOG={ON|OFF}
```

Event-Name	イベント名を指定します。
MESSAGE	該当するイベントを外部に送信するかどうか指定します。通知を行う場合はONにしてください。
LOG	イベントを上記LOGFILEで指定したLOGファイルに出力するかどうか指定します。出力する場合はONにしてください。

■ エラー停止時イベント設定

```
SEND_MSG_ESTOP {ON|OFF}
```

ジョブネットワークトラッカがエラー停止のイベントを出力した場合に、続けて時間超過警告のイベントを出力するかどうかを設定します。

通常、上記EVENT設定のうちエラー停止(jnwsv.xxx.estop)イベントにMESSAGE=ON を設定して、ジョブネットワークトラッカについてエラー停止が発生した場合、 JNW_ERROR_STOP イベントが出力されます。そのフロー中に未到達警報を設定した時刻待ち合わせ部品や、超過警告またはクリティカルポイント警告の設定を行っている部品があり、かつ時間超過警告（jnw.xxx.ovt）のイベントに MESSAGE=ON を設定している場合、ジョブネットワークトラッカがエラー停止状態になると結果的に時間超過が発生して、エラー停止と時間超過の二つのエラーイベントが出力されることになります。

関連する時間超過設定は次のとおりです。

- ジョブネットワークの終了時間超過
- ジョブネットワークのクリティカルポイント
- 単位ジョブの終了時間超過
- 単位ジョブのクリティカルポイント
- 時刻待ち合わせ部品の未開始警告

本パラメータ設定時の動作は次のとおりです。

SEND_MSG_ESTOP	ジョブネットワークパラメータの［エラー時の自動停止］設定	動　作
ON	停止する/停止しない	出力するように設定されている全てのイベントを出力する（従来どおり）。
OFF	停止する	ジョブネットワークがエラー停止した場合、時間超過警告イベントの出力を行わない（多重にエラーイベント出力を行わない）。
	停止しない	時間超過警告でONに設定されている全てのイベントを出力する。

12.2.3. イベント定義ファイル

■パス

ローカルサイト	/usr/spool/nqs/gui/jobmsg.conf
クラスタサイト	共有ディスク上の <JobCenterDBパス>/nqs/gui/jobmsg.conf

■説明

本ファイルは、イベントログファイルに出力するイベント、およびAPIで通知するイベントのフォーマットを指定します。

関連するタグは次のとおりです。

- イベントログファイルに出力するイベント

```
EVENT <Event-Name> <MsgID> <Message-Body>
```

イベントログファイルに出力するイベントのフォーマットを指定します。

Event-Name	イベントの種類を表すイベントタイプ名の文字列です。
MsgID	イベントIDです。

	各イベントの詳細は「12.2.4 イベント一覧」を参照してください。
Message-Body	出力するメッセージパターンを指定します。

- Micro Focus Operations Manager softwareのAPIで通知するイベント

```
OPCMSG <eventname> [APL=application] [OBJ=object] [SEV=severity] [GRP=msggrp] [NODE=node]
```

Micro Focus Operations Manager softwareインタフェースではメッセージパラメータをイベントごとに指定します。

eventnameには上記「イベントログファイルに出力するイベント」で定義したEvent-Nameで指定した名称を使用します。 ただしopcmsg.defaultは、各パラメータを省略したときに用いる値を指定するために使用します。指定は1行で行わなければなりません。

各パラメータKEYの意味は次のとおりです。

APL	アプリケーション名を指定します。
OBJ	オブジェクト名を指定します。
SEV	メッセージのレベルを表します。次のいずれかを指定します。 NORMAL WARNING MINOR MAJOR CRITICAL
GRP	メッセージグループを指定します。
NODE	イベントのノードを指定します。

KEY=value形式の各パラメータ指定では、valueを" "または' 'でくくることができます。

- 共通

［TIMEFMT］

時刻表示の形式に関する補足的なパラメータです。次の形式があります。

```
TIMEFMT starttime1 <time-format>
  TIMEFMT starttime2 <time-format>
  TIMEFMT endtime1   <time-format>
  TIMEFMT endtime2   <time-format>
```

上記はそれぞれのメッセージマクロでの時刻の出力フォーマットを指定します。

フォーマット<time-format>はライブラリ関数strftime(3)で用いるものと同一です。

jobmsg.confではあらかじめ次の通りに設定してあります。

```
TIMEFMT starttime1 %c
  TIMEFMT starttime2 %R
  TIMEFMT endtime1   %T
  TIMEFMT endtime2   %R
```

startime1およびstarttime2、endtime1およびendtime2とそれぞれ二種類あるため、メッセージの種類により時刻出力フォーマットを変更したい場合に使い分けます。

［メッセージ出力パターン］

一般に、EVENT、OPCMSGタグのメッセージ出力パターンに指定した文字列はそのまま出力されます。ただし以下の一覧表のように%でくくられた文字列はマクロ文字として解釈されます。

マクロ文字は、それぞれメッセージの内容に従って決まったパターンに変換されます。

1つのイベント中で使用可能なマクロの数は10個です。これを超えてマクロを使用するとマクロの部分は無視されます。

マクロ	変換対象	指定可能なイベント名
%occur%	イベントが発生したマシン名に変換されます。ホスト名は省略されません。	jnwsv.*
%occurS%	イベントが発生したマシン名に変換されます。ドメイン部および64文字以上は省略されます。	jnwsv.*
%occurD%	イベントが発生したマシン名に変換されます。ドメイン部および61文字以上は省略され"..."と表示されます。	jnwsv.*
%jnwname%	ジョブネットワーク名に変換されます。	jnwsv.*
%jnwnameB%	ジョブネットワーク名のうち、親ジョブネットワーク名を省いたものに変換されます。	jnwsv.*
%starttime1%	ジョブネットワークまたは単位ジョブの投入時刻にTIMEFMT starttime1で設定されたフォーマットで変換されます。	jnwsv.*
%starttime2%	ジョブネットワークまたは単位ジョブの投入時刻にTIMEFMT starttime2で設定されたフォーマットで変換されます。	jnwsv.*
%endtime1%	ジョブネットワークまたは単位ジョブの終了時刻にTIMEFMT endtime1で設定されたフォーマットで変換されます。	jnwsv.*.end jnwsv.uj.err jnwsv.uj.estop
%endtime2%	ジョブネットワークまたは単位ジョブの終了時刻にTIMEFMT endtime2で設定されたフォーマットで変換されます。	jnwsv.*.end jnwsv.uj.err jnwsv.uj.estop
%ujname%	メッセージに対する単位ジョブ名に変換されます。	jnwsv.dialog.* jnwsv.uj.* jnwsv.holdevent.*
%ujexecmac%	メッセージに対応する単位ジョブの実行マシン名に変換されます。	jnwsv.uj.end jnwsv.holdevent.*
%ujerrmsg%	エラーメッセージに変換されます。	jnwsv.uj.err jnwsv.uj.estop
%dialog-msg%	ダイアログメッセージに変換されます。	jnwsv.*
%jnwuser%	ジョブネットワークの投入者に変換されます。	jnwsv.jnw.* jnwsv.uj.*

マクロ	変換対象	指定可能なイベント名
%jnwstat%	ジョブネットワークの状態（NORMAL、ERROR）に変換されます。	jnwsv.jnw.*.end
%trkid%	トラッカIDに変換されます。	jnwsv.holdevent.*
%evtid%	イベントIDに変換されます。	jnwsv.holdevent.*
%sitename%	jnwcasterが起動されているホスト名、またはクラスタサイト名に変換されます。	notice.daemon.* (OPCMSGタグのみ)
%comment%	ジョブネットワークや部品に設定されているコメント文字列に変換されます。	jnwsv.jnw.* jnwsv.eventsend.* jnwsv.eventrecv.* jnwsv.uj.* jnwsv.waitime.notstart
%trkpath%	イベントが発生したトラッカをCL/Webで参照するためのURLのパス部分に変換されます。	jnwsv.*
%partspath%	イベントが発生したトラッカをCL/Webで参照するためのURLのパス部分に変換されます。	jnwsv.uj.* jnwsv.sapj.* jnwsv.sbwj.* jnwsv.spcj.* jnwsv.ucxsj.* jnwsv.wobsj.*

%comment%マクロはR15.5から利用できます。R15.5より前のバージョンからR15.5以降にバージョンアップした場合、イベント定義ファイルがバージョンアップによって引き継がれるため、%comment%マクロがイベント定義ファイルには追加されていない状態になります。%comment%マクロを利用したい場合は、手動で追加してください。

12.2.4. イベント一覧

イベントの一覧を表12.2「イベント一覧」に記載します。

なお「イベント名」はトラッカのログやjnwcaster.confにLOGFILE / LOGSIZEで設定したログに出力される名称で、「イベントサマリ」はイベント連携ソフトのビューアで識別される名称となります。ただし、以下のイベントについては、「イベント名」「イベントサマリ」どちらも同じメッセージが出力されます。

■jnwsv.que.ustart

■jnwsv.que.ustop

■jnwsv.que.sstop

■jnwsv.release.start

■jnwsv.release.end

■jnwsv.release.err

表12.2 イベント一覧

イベントID	イベントタイプ名	イベントの意味	イベント名(イベントログ出力) イベントサマリ
0x2001	jnwsv.jnw.root.start	親ジョブネットワークの実行が開始しました。	JNW_START JOBS_JNW_START
0x2002	jnwsv.jnw.root.end	親ジョブネットワークの実行が終了しました。	JNW_END JOBS_JNW_END
0x2001	jnwsv.jnw.sub.start	サブジョブネットワークの実行が開始しました。	JNW_START JOBS_JNW_START
0x2002	jnwsv.jnw.sub.end	サブジョブネットワークの実行が終了しました。	JNW_END JOBS_JNW_END
0x2004	jnwsv.jnw.ovt	ジョブネットワークの実行が予定時間を超過しました。	JNW_TOTAL_TIME_EXCEEDED JOBS_JNW_OVERTIME
0x2009	jnwsv.jnw.cri	ジョブネットワークで設定されているクリティカルポイント警告時間を超過しました。	JNW_CRITICAL_POINT_CHECK JOBS_JNW_CRITICALPOINT_ALERT
0x2010	jnwsv.dialog.start	ダイアログ待ちになりました。	JNW_DIALOG_START JOBS_DIALOG_START
0x2011	jnwsv.dialog.end	ダイアログ待ち状態から再開しました。	JNW_DIALOG_END JOBS_DIALOG_END
0x2020	jnwsv.eventsend.start	イベント送信が開始しました。	JNW_EVENTSEND_START JOBS_JNW_EVENTSEND_START
0x2021	jnwsv.eventsend.end	イベント送信が終了しました。	JNW_EVENTSEND_END JOBS_JNW_EVENTSEND_END
0x2022	jnwsv.eventsend.timeout	イベント送信が予定時間を超過しました。	JNW_EVENTSEND_TIMEOUT JOBS_JNW_EVENTSEND_TIMEOUT
0x2024	jnwsv.eventrecv.start	イベント受信が開始しました。	JNW_EVENTRECEIVE_START JOBS_JNW_EVENTRECEIVE_START
0x2025	jnwsv.eventrecv.end	イベント受信が終了しました。	JNW_EVENTRECEIVE_END JOBS_JNW_EVENTRECEIVE_END
0x2026	jnwsv.eventrecv.timeout	イベント受信が予定時間を超過しました。	JNW_EVENTRECEIVE_TIMEOUT JOBS_JNW_EVENTRECEIVE_TIMEOUT
0x2030	jnwsv.holdevent.overwrite	保持しているイベントが上書きされました。	JNW_HOLDEVENT_OVERWRITE JOBS_JNW_HOLDEVENT_OVERWRITE

イベントID	イベントタイプ名	イベントの意味	イベント名(イベントログ出力) イベントサマリ
0x2031	jnwsv.holdevent.max	保持しているイベント数が保留可能イベント数を超えました。	JNW_HOLDEVENT_MAX JOBS_JNW_HOLDEVENT_MAX
0x2032	jnwsv.holdevent.timeout	保持しているイベントが有効期間を超えて消失しました。	JNW_HOLDEVENT_TIMEOUT JOBS_JNW_HOLDEVENT_TIMEOUT
0x2006	jnwsv.uj.start	単位ジョブが開始しました。	JNW_UNITJOB_START JOBS_UJOB_START
0x2007	jnwsv.uj.end	単位ジョブが終了しました。	JNW_UNITJOB_END JOBS_UJOB_END
0x2003	jnwsv.uj.err	単位ジョブがエラー終了しました。	JNW_UNITJOB_ERROR JOBS_UJOB_ERROR
0x2008	jnwsv.uj.estop	単位ジョブの実行がエラー停止しました。	JNW_ERROR_STOP JOBS_UJOB_ESTOP
0x2005	jnwsv.uj.ovt	単位ジョブの実行が予定時間を超過しました。	JNW_UNITJOB_TIME_EXCEEDED JOBS_UJOB_OVERTIME
0x200a	jnwsv.uj.cri	単位ジョブで設定されているクリティカルポイント警告時間を超過しました。	JNW_UNITJOB_CRITICAL_POINT_CHECK JOBS_UJOB_CRITICALPOINT_ALERT
0x200b	jnwsv.uj.warn	単位ジョブの実行が警告終了しました。	JNW_UNITJOB_WARNING JOBS_UJOB_WARNING
0x2006	jnwsv.sapj.start	ERPジョブが開始しました。	JNW_ERPJOB_START JOBS_UJOB_START
0x2007	jnwsv.sapj.end	ERPジョブが終了しました。	JNW_ERPJOB_END JOBS_UJOB_END
0x2003	jnwsv.sapj.err	ERPジョブがエラー終了しました。	JNW_ERPJOB_ERROR JOBS_UJOB_ERROR
0x2008	jnwsv.sapj.estop	ERPジョブの実行がエラー停止しました。	JNW_ERPJOB_ERROR_STOP JOBS_UJOB_ESTOP
0x2005	jnwsv.sapj.ovt	ERPジョブの実行が予定時間を超過しました。	JNW_ERPJOB_TIME_EXCEEDED JOBS_UJOB_OVERTIME
0x2006	jnwsv.sbwj.start	BIジョブが開始しました。	JNW_BIJOB_START JOBS_UJOB_START
0x2007	jnwsv.sbwj.end	BIジョブが終了しました。	JNW_BIJOB_END JOBS_UJOB_END

イベントID	イベントタイプ名	イベントの意味	イベント名(イベントログ出力) イベントサマリ
0x2003	jnwsv.sbwj.err	BIジョブがエラー終了しました。	JNW_BIJOB_ERROR JOBS_UJOB_ERROR
0x2008	jnwsv.sbwj.estop	BIジョブの実行がエラー停止しました。	JNW_BIJOB_ERROR_STOP JOBS_UJOB_ESTOP
0x2005	jnwsv.sbwj.ovt	BIジョブの実行が予定時間を超過しました。	JNW_BIJOB_TIME_EXCEEDED JOBS_UJOB_OVERTIME
0x2006	jnwsv.spcj.start	PCジョブが開始しました。	JNW_PCJOB_START JOBS_UJOB_START
0x2007	jnwsv.spcj.end	PCジョブが終了しました。	JNW_PCJOB_END JOBS_UJOB_END
0x2003	jnwsv.spcj.err	PCジョブがエラー終了しました。	JNW_PCJOB_ERROR JOBS_UJOB_ERROR
0x2008	jnwsv.spcj.estop	PCジョブの実行がエラー停止しました。	JNW_PCJOB_ERROR_STOP JOBS_UJOB_ESTOP
0x2005	jnwsv.spcj.ovt	PCジョブの実行が予定時間を超過しました。	JNW_PCJOB_TIME_EXCEEDED JOBS_UJOB_OVERTIME
0x2006	jnwsv.ucxsj.start	UCXSingleジョブが開始しました。	JNW_UCXSJOB_START JOBS_UJOB_START
0x2007	jnwsv.ucxsj.end	UCXSingleジョブが終了しました。	JNW_UCXSJOB_END JOBS_UJOB_END
0x2003	jnwsv.ucxsj.err	UCXSingleジョブがエラー終了しました。	JNW_UCXSJOB_ERROR JOBS_UJOB_ERROR
0x2008	jnwsv.ucxsj.estop	UCXSingleジョブの実行がエラー停止しました。	JNW_UCXSJOB_ERROR_STOP JOBS_UJOB_ESTOP
0x2005	jnwsv.ucxsj.ovt	UCXSingleジョブの実行が予定時間を超過しました。	JNW_UCXSJOB_TIME_EXCEEDED JOBS_UJOB_OVERTIME
0x200a	jnwsv.ucxsj.cri	UCXSingleジョブで設定されているクリティカルポイント警告時間を超過しました。	JNW_UCXSJOB_CRITICAL_POINT_CHECK JOBS_UJOB_CRITICALPOINT_ALERT
0x2006	jnwsv.wobsj.start	WOBSジョブが開始しました。	JNW_WOBSJOB_START JOBS_UJOB_START
0x2007	jnwsv.wobsj.end	WOBSジョブが終了しました。	JNW_WOBSJOB_END JOBS_UJOB_END

イベントID	イベントタイプ名	イベントの意味	イベント名(イベントログ出力) イベントサマリ
0x2003	jnwsv.wobsj.err	WOBSジョブがエラー終了しました。	JNW_WOBSJOB_ERROR JOBS_UJOB_ERROR
0x2008	jnwsv.wobsj.estop	WOBSジョブの実行がエラー停止しました。	JNW_WOBSJOB_ERROR_STOP JOBS_UJOB_ESTOP
0x2005	jnwsv.wobsj.ovt	WOBSジョブの実行が予定時間を超過しました。	JNW_WOBSJOB_TIME_EXCEEDED JOBS_UJOB_OVERTIME
0x200a	jnwsv.wobsj.cri	WOBSジョブで設定されているクリティカルポイント警告時間を超過しました。	JNW_WOBSJOB_CRITICAL_POINT_CHECK JOBS_UJOB_CRITICALPOINT_ALERT
0x200b	jnwsv.wobsj.warn	WOBSジョブの実行が警告終了しました。	JNW_WOBSJOB_WARNING JOBS_UJOB_WARNING
0x2012	jnwsv.waittime.notstart	時刻待ち合わせ部品で指定されている時刻に開始状態になりませんでした。	JNW_WAITTIME_NOT_START JOBS_WAITTIME_NOSTART
0x0151	notice.daemon.startup	JobCenterが稼動しました。	NODE_DAEMON_START JOBS_NODE_START
0x0152	notice.daemon.shutdown	JobCenterが終了しました。	NODE_DAEMON_STOP JOBS_NODE_END
0x2033	jnwsv.que.ustart	ユーザ操作によって、キューが開始しました。	User start the queue(<キュー名>).
0x2034	jnwsv.que.ustop	ユーザ操作によって、キューが停止しました。	User stop the queue(<キュー名>).
0x2035	jnwsv.que.sstop	JobCenterが、キューを停止しました。	Since abnormalities occurred, the queue (<キュー名>:status=<ステータス>) was stopped(<詳細>). Please check the state of queue, job, system, etc. And start queue after resolving the problem.
0x2036	jnwsv.release.start	計画リリースが開始されました。	RELEASE_START (HostName=<ホスト名>, User=<ユーザ名>, EstimateTime=<リリース予定時刻>).
0x2037	jnwsv.release.end	計画リリースが正常終了しました。	RELEASE_END (HostName=<ホスト名>, User=<ユーザ名>, EstimateTime=<リリース予定時刻>).
0x2038	jnwsv.release.err	計画リリースが異常終了しました。	RELEASE_ERROR (HostName=<ホスト名>, User=<ユーザ名>, EstimateTime=<リリース予定時刻>).

■メッセージ中に%（マクロ定義を除く）は使用できません。

■マクロ（%～%）を連続して記述できません。

■notice.*のイベントではマクロを使用できません(OPCMSGタグのパラメータを除く)。

■送信先で障害が発生している場合、未送信のイベントは削除されることがあります。

■設定ファイルに上記エントリがない場合は追加する必要があります。

■イベント出力の設定はイベント保持を行っている送信先マシンで行う必要があります。

■jnwsv.uj.start～jnwsv.uj.criのイベント、jnwsv.sapj.start～jnwsv.sapj.ovtのイベント、jnwsv.sbwj.start～jnwsv.sbwj.ovtのイベント、jnwsv.ucxsj.start～jnwsv.ucxsj.criのイベントについてはイベントサマリが共通のため、イベントメッセージの内容でお互いを区別する必要があります。

■カスタムジョブは実行上単位ジョブと同様に扱い、イベント出力も単位ジョブと同様になります。 カスタムジョブをイベント連携に使用する場合は、jnwsv.uj.* のイベントタイプ名を指定してください。

■UCXSingleジョブはR14.1で廃止されました。

12.3. Windows版のイベント連携

Windows版では、次のイベント連携ができます。

■JobCenterのイベントを指定のイベントログファイルに出力する場合は、「12.3.1 イベントログファイルへの出力方法」を参照してください。

■JobCenterのイベントをWindowsイベントログに出力し、そのイベントをESMPRO/SA経由でESMPRO/統合ビューア（AlertManager）に通知する場合は、「12.3.2 ESMPRO/統合ビューア」を参照してください。

■APIを利用してMicro Focus Operations Manager softwareのメッセージビューアに通知する場合は、「12.3.3 Micro Focus Operations Manager software」を参照してください。

12.3.1. イベントログファイルへの出力方法

JobCenterのイベントをイベントログファイルに出力するための設定方法について説明します。

1. イベントログファイルの出力設定を有効にします。
   1. 管理者アカウントでWindowsにログオンし、サーバの環境設定を起動します。
   2. 左ペインの[サイト]を選択します。
   3. 右ペインに表示されたサイトの中で、イベント設定を行いたいサイトをダブルクリックします。
   4. [ログ]タブを選択し、[ログファイルを出力する]のチェックボックスをチェックします。
   5. ログファイル関連の設定を指定します。

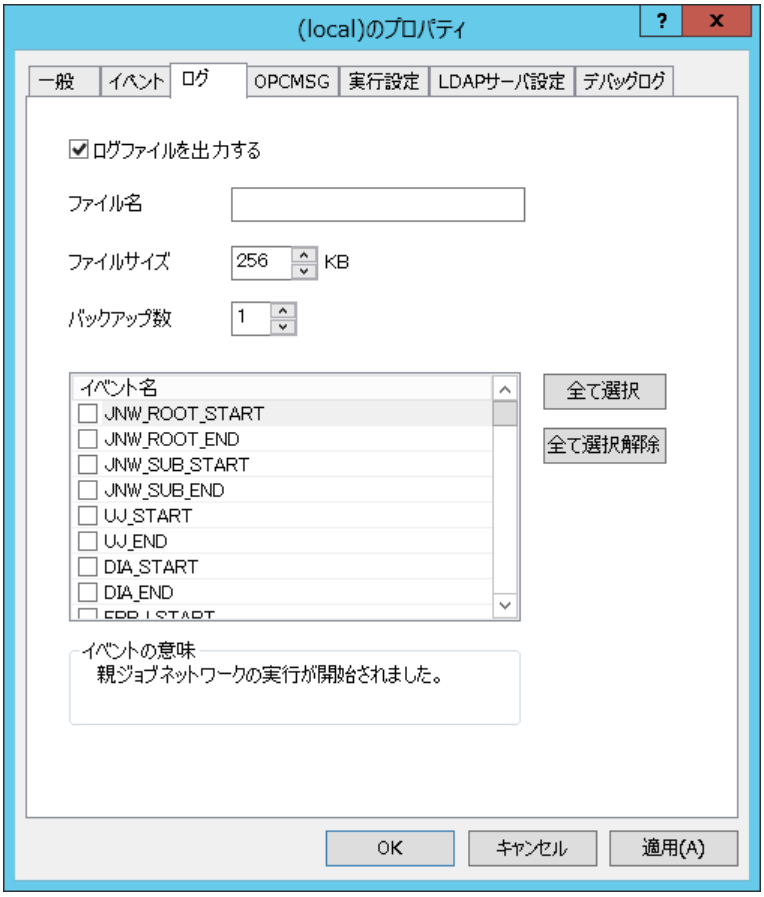

図12.3 ログタブ画面例

■ファイル名

ログファイルのパスは、絶対パスまたは相対パスでも指定できます。パスの指定例は次のとおりです。

絶対パス	d:\mywork\me\jnwenginevent.log
相対パス	\aaa\bbb\event.log

ローカルサイトまたはクラスタ環境で相対パスを指定するときの起点フォルダは、次のとおりです。

ローカルサイト	%InstallDirectory%\spool
クラスタサイト	<JobCenterDBパス>\spool

%InstallDirectory%はJobCenterのインストールディレクトリです。デフォルトはC:\JobCenter\SVになりS。

<JobCenterDBパス>は、cjcmksite時に指定した共有ディスク上のデータベースディレクトリです。

■ファイルサイズ

ファイルサイズの有効範囲は、64KB～4096KBになります。

■バックアップ数

バックアップ数の有効範囲は、1～10になります。

6. 出力させたいJobCenterのイベント名のチェックボックスをチェックします。

イベントの詳細は、「12.3.5 イベントの種類」を参照してください。

■［全て選択］ボタン

全てのイベント名にチェックを付けます。

■［全て選択解除］ボタン

全てのイベント名のチェックを外します。

2. [OK]ボタン、または[適用]ボタンを押し、設定を有効にします。

次のダイアログが表示されるので、サイトを再起動すると本設定が有効になります。

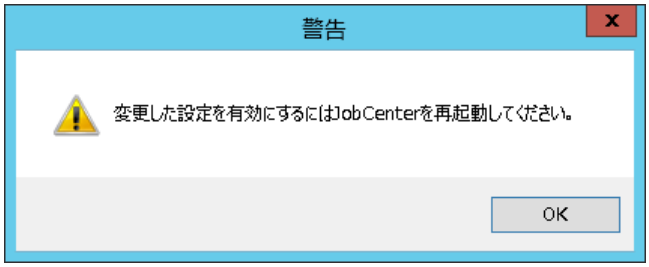

図12.4 再起動画面例

ログは文字コードの設定によらずSJISで出力されます。

12.3.2. ESMPRO/統合ビューア

運用管理者へのジョブの実行状況通知を行えます。

NEC Express5800シリーズ（ここでは、Windowsサーバのことを指します）では、NECが独自に開発した『ESMPRO』と総称されるソフトウェアにより、運用管理の基盤となる「ESMPRO/ServerAgent・ServerManager」をサーバに標準添付しています。

Expressサーバの運用管理は、まず、この ESMPRO/ServerAgent・ServerManager により、Expressサーバの各ハードウェア・コンポーネントで発生する障害状況を監視・管理することが基本となります。

このExpressサーバにJobCenter MG/SVを構築すると、1台の管理用PCに構築したESMPRO/統合ビューア(ESMPRO/ServerManager) にジョブの実行状況の通知を行えますので、運用管理者はアラートビューア／データビューアから通知された実行状況を監視できます。

以降の説明ではESMPRO/ServerAgent,ServerManager Ver.6 の画面を利用して説明します。

12.3.2.1. インストール

次に示す順にインストール、設定を行います。

1. ESMPRO/ServerAgentのインストールおよび設定

   ExpressサーバにESMPRO/ServerAgentをインストールします。

   ESMPRO/ServerAgentのインストールおよび設定の詳細については、 EXPRESSBUILDER CD-ROM に同梱されている「ESMPRO/ServerAgentインストレーションガイド」を参照してください。

2. ESMPRO/ServerManagerのインストールおよび設定

   1台の管理用PCに ESMPRO/ServerManager をインストールします。ESMPRO/ServerManager を Expressサーバまたはワークステーション本体にインストールしても構いません。

   ESMPRO/ServerManagerのインストールおよび設定の詳細については、 EXPRESSBUILDER CD-ROM に同梱されている「ESMPRO/ServerManager インストレーションガイド」を参照してください。

3. ESMPRO/ServerAgent・ServerManager間の通報テスト

   JobCenter SVをインストールする前に、1でインストールした ESMPRO/ServerAgent と ServerManager 間の通報テストを行ってください。

   ■エージェント側の設定

   a. Windowsの［スタート］－［設定］－［コントロールパネル］－［ESMPRO ServerAgent］を選択します。

   b. ［ESMPRO ServerAgent Vx.xxのプロパティ］ダイアログ －［全般］が表示されますので、［通報設定］ボタンをクリックします。

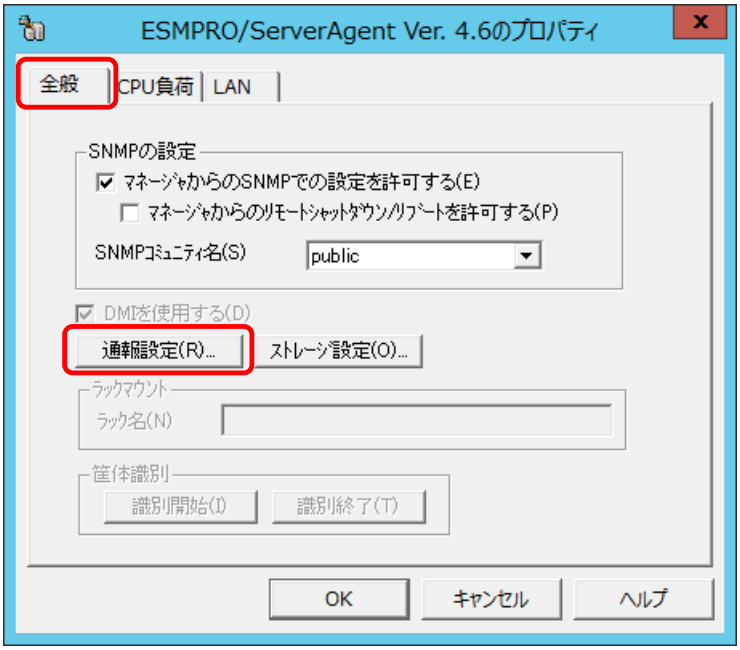

図12.5 ［ESMPRO ServerAgent Vx.xxのプロパティ］ダイアログ画面例

c. ［アラートマネージャ］ウィンドウが表示されます。メニューバーの［設定］－［通報先リストの設定］を選択します。

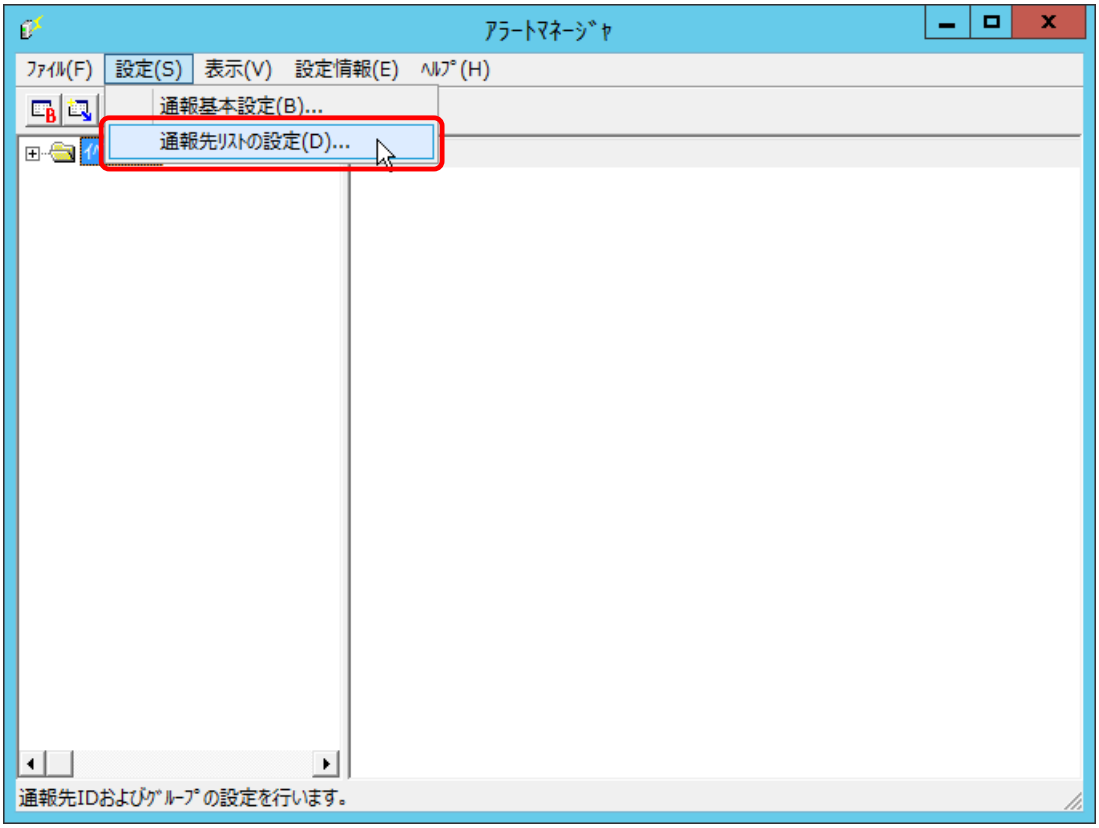

図12.6 ［アラートマネージャ］ウィンドウ画面例

d. ここでは、通報手段に［マネージャ通報(TCP/IP In-Band)］を使用したモデルを使用します。［通信先リストの設定］ダイアログのIDで、［通報手段］：［マネージャ通報(TCP/IP In-Band)］にカーソルをあわせ、［修正］ボタンをクリックします。

図12.7 ［通報先リストの設定］ダイアログ画面例

e. ［ID設定］ダイアログが表示されますので、［宛先設定］ボタンをクリックします。

図12.8 ［ID設定］ダイアログ画面例

f. ［マネージャ通報(TCP/IP In-Band)の設定］ダイアログが表示されますので、［通報テスト］ボタンをクリックします。ESMPRO/統合ビューア［ESMPRO/ServerManager］にアラートが送付されます。

図12.9 ［マネージャ通報(TCP/IP In-Band)の設定］ダイアログ

■マネージャ側の設定

a. ESMPRO ServerManagerのログイン画面から［アラートビューア］を選択します。

図12.10 ログイン画面例

b. ［アラートビューア］が表示されますので、アラート一覧から［TEST_TRAP_NAME］を選択し、ダブルクリックします。

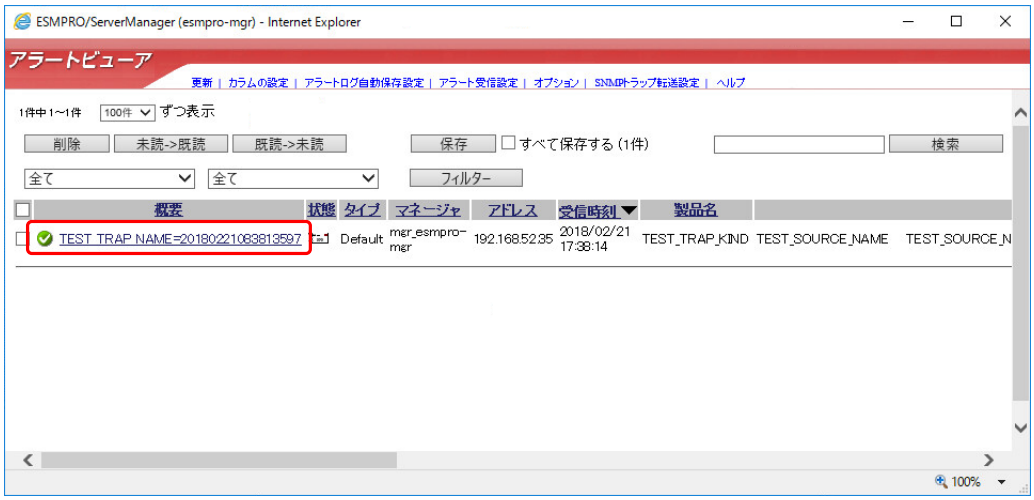

図12.11 ［アラートビューア］画面例

c. ［アラート詳細］が表示されますので、通報テストが正しくできたかどうかを確認します。

図12.12 ［アラート詳細］画面例

4. JobCenter SVのインストールおよび設定

ExpressサーバにJobCenter SVをインストールします。 ESMPRO/ServerAgent と ServerManager は、JobCenter SV および CL/Win よりも先にインストールするようにしてください。

JobCenter SV のインストールおよび設定の詳細については<インストールガイド>を参照してください。

> Windows版JobCenterは、MG、SVともに同一のインストールパッケージです。

5. ESMPRO/ServerAgent・ServerManager間におけるJobCenter監視イベントの通報テスト

JobCenter SVをインストールした後に、ESMPRO/ServerAgent と ServerManager 間で以下のようにJobCenter の監視イベントの通報テストを行ってください。

■エージェント側の設定

a. Windowsの［スタート］－［設定］－［コントロールパネル］－［ESMPRO ServerAgent］を選択します。

b. ［ESMPRO ServerAgent］を選択し、ダブルクリックします。

c. ［ESMPRO ServerAgent Vx.xxのプロパティ］ダイアログ－［全般］が表示されますので、［通報設定］ボタンをクリックします。

d. ［アラートマネージャ］ウィンドウが表示されます。メニューバーの［設定］－［通報先リストの設定］を選択します。ここでは、通報手段に［マネージャ通報(TCP/IP In-Band)］を使用したモデルを使用します。

e. ［イベントログ］ツリーの階層を開いて、［イベントログ種別］アプリケーションの下に［イベントソース名］JNWEXEがあることを確認します。

f. ［イベントソース名］JNWEXEの［イベントID］を選択し、右クリックしたときのポップアップメニューから［通報テスト］を選択します。

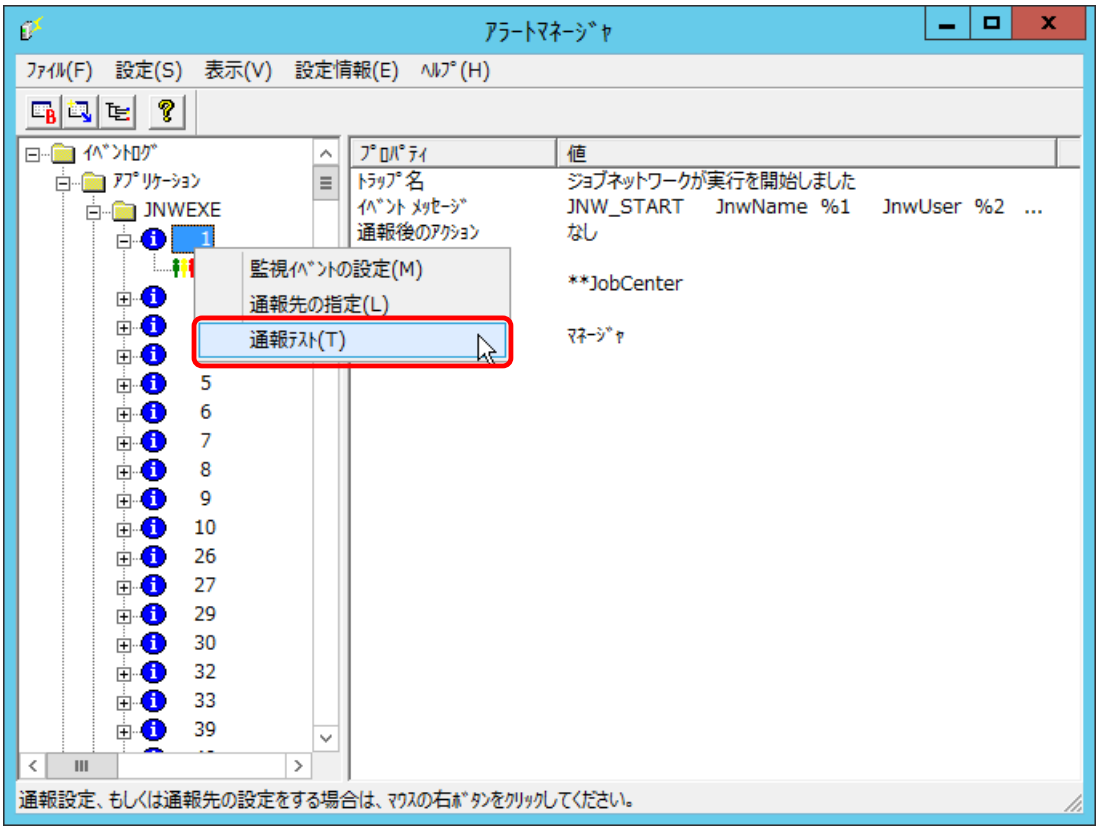

図12.13 ［アラートマネージャ］ウィンドウ画面例

g. ESMPRO/統合ビューア［ESMPRO/ServerManager］にアラートが送付されます。

■マネージャ側の設定

a. ESMPRO ServerManagerのログイン画面から［アラートビューア］を選択します。

b. ［アラートビューア］が表示されますので、アラート一覧から該当するアラートを選択し、ダブルクリックします。

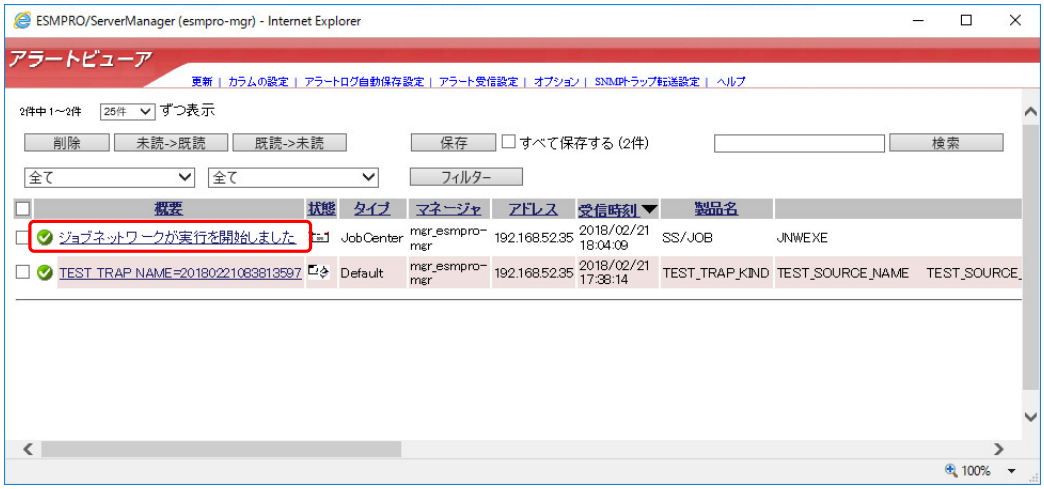

図12.14 ［アラートビューア］画面例

c. ［アラート詳細］が表示されますので、通報テストが正しくできたかどうかを確認します。

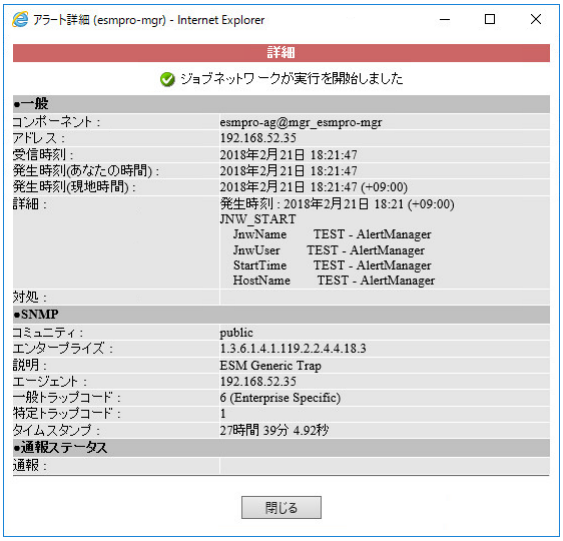

図12.15 ［アラート詳細］画面例

12.3.2.2. 通知するイベントの選択

サーバの環境設定プログラムで、通知するイベントを設定する事ができます。

1. 管理者アカウントでWindowsにログオンし、Windowsの［スタート］－［プログラム］－［JobCenter］－［SV］－［サーバの環境設定］を選択して、サーバの環境設定を起動します。

2. 左ペインの「サイト」を選択します。

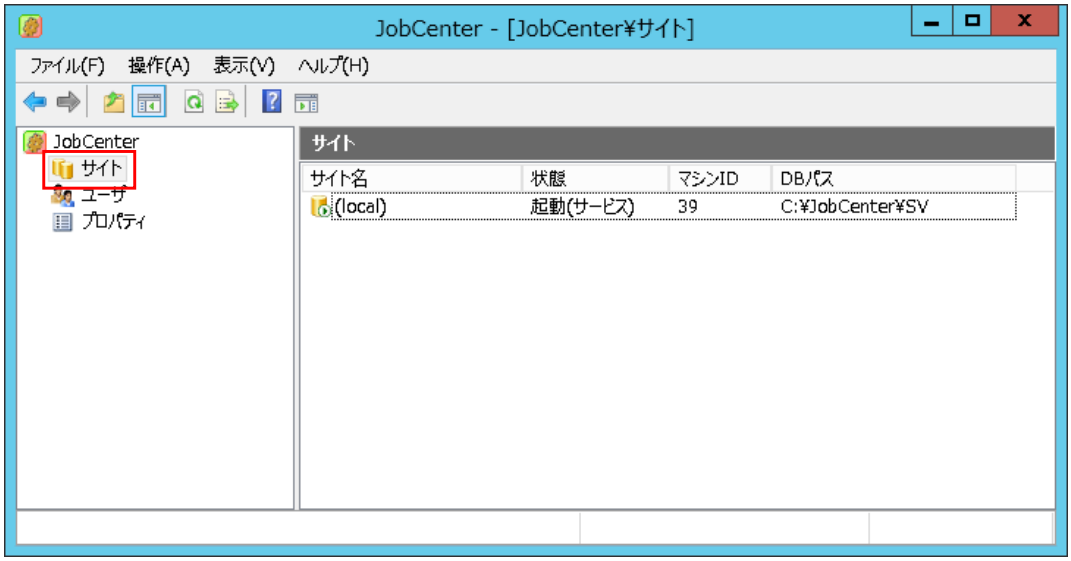

図12.16 ［サーバの環境設定］のサイト画面例

3. イベント設定を行いたいサイトをダブルクリックして、サイトのプロパティを表示し、「イベント」タブを選択します。

4. 通知するイベントについて設定します。該当するイベントを送信するかどうか選択します。通知を行う場合はイベント名の先頭のチェックボックスをチェックしてください。

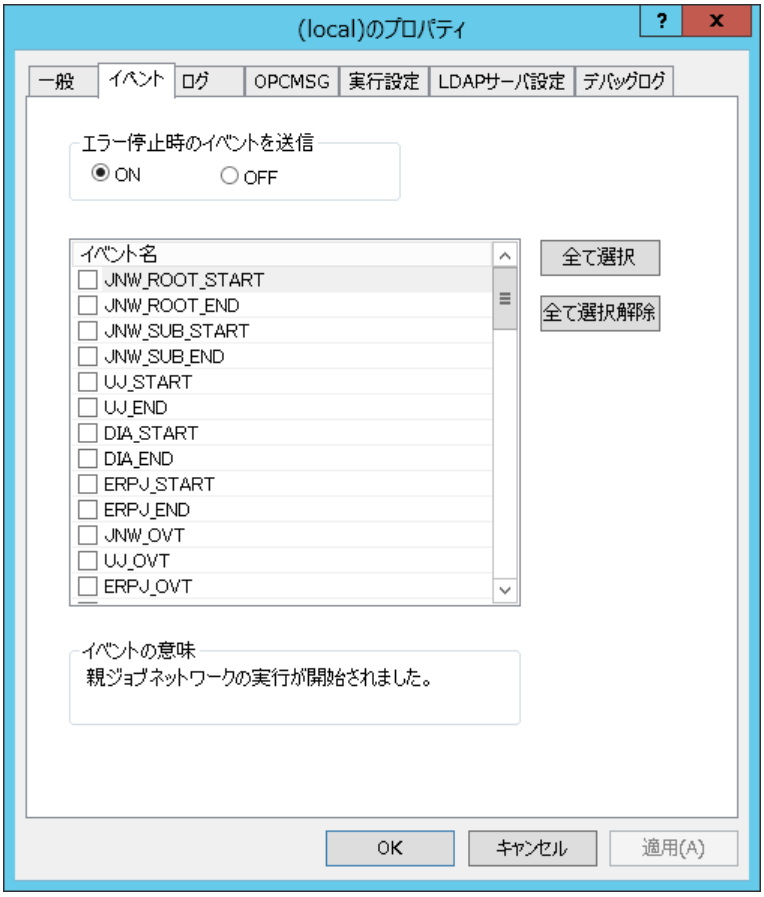

図12.17 イベントタブの画面例

［イベント］タブの設定項目は次のとおりです。

■エラー停止時のイベントを送信

ジョブネットワークトラッカがエラー停止のイベントを出力した場合に、続けて時間超過警告のイベントを出力するかどうかを設定します。

通常、上記 EVENT 設定のうちエラー停止系イベント（XXX_ESTOP）を選択して、ジョブネットワークトラッカについてエラー停止が発生した場合、 JNW_ERROR_STOP イベントが出力されます。

そのフロー中に未到達警報を設定した時刻待ち合わせ部品や、超過警告またはクリティカルポイント警告の設定を行っている部品があり、かつ時間超過警告系イベント（XXX_OVT）を選択している場合、ジョブネットワークトラッカがエラー停止状態になると結果的に時間超過が発生して、エラー停止と時間超過の二つのエラーイベントが出力されることになります。

関連する時間超過設定は次のとおりです。

- ジョブネットワークの終了時間超過
- ジョブネットワークのクリティカルポイント
- 単位ジョブの終了時間超過
- 単位ジョブのクリティカルポイント
- 時刻待ち合わせ部品の未到達警告

表12.3 ［エラー停止時のイベントを送信］の設定と動作

エラー停止時のイベントを送信	ジョブネットワークパラメータの［エラー時の自動停止］設定	動　作
ON	停止する/停止しない	出力するように設定されている全てのイベントを出力する(従来どおり)。
OFF	停止する	ジョブネットワークがエラー停止の場合、時間超過警告イベントの出力を行わない(多重にエラーイベント出力を行わない)。
	停止しない	時間超過警告でONに設定されている全てのイベントを出力する。

■イベントメッセージの選択

各イベントについて、イベントメッセージを採取する場合はイベント名先頭のチェックボックスをチェックしてください。

イベントの詳細は、「12.3.5 イベントの種類」を参照してください。

■［全て選択］ボタン

全てのイベント名にチェックを付けます。

■［全て選択解除］ボタン

全てのイベント名のチェックを外します。

■［キャンセル］ボタン

変更をキャンセルします。

■［OK］または［適用］ボタン

設定を適用します。次のダイアログが表示されるので、サイトを再起動すると本設定が有効になります。

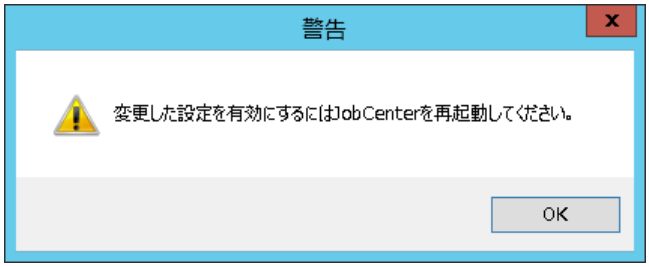

図12.18 再起動ダイアログ

12.3.2.3. Windowsイベントビューア アプリケーションログ

サーバの環境設定の「イベントの選択」で対応するイベントを設定することで、［イベントビューア］の［アプリケーションログ］に次の情報を通知できます。

■「jnwexe」というソース名にてジョブの実行や状態

■「NetShepherd」というソース名にてキューの状態

出力フォーマットは、ServerManagerのアラートビューアなどとは異なるため、以下の表を参照してください。なお、表のメッセージフォーマット中にある%nは改行、%sはスペース、%tはタブを表しています。

表12.4 イベント一覧(ソース名「jnwexe」)

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
1	情報	JNW_ROOT_START	JNW_START%n %s%tJnwName%t%t%1%n %s%tJnwUser%t%t%2%n %s%tStartTime%t%t%3%n %s%tHostName%t%t%4%n %s%tComment%t%t%5	%1:ジョブネットワーク名 %2:ジョブネットワークの所有者名 %3:実績開始時間 %4:実行ホスト名 %5:ジョブネットワークのコメント	RootJNW(子JNWではない)が開始した
2	情報	JNW_ROOT_END	JNW_END%n %s%tJnwName%t%t%1%n %s%tJnwUser%t%t%2%n %s%tEndTime%t%t%3%n %s%tJnwStat%t%t%4%n %s%tHostName%t%t%5%n %s%tComment%t%t%6	%1:ジョブネットワーク名 %2:ジョブネットワークの所有者名 %3:実績終了時間 %4:ジョブネットワーク終了状態(Normal, Error) %5:実行ホスト名 %6:ジョブネットワークのコメント	RootJNW(子JNWではない)が終了した
3	情報	JNW_SUB_START	SUBJNW_START%n %s%tJnwName:SubJnwName%t%1%n	%1:ジョブネットワーク名:サブジョブネットワーク名	SubJNWが開始した

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
			%s%tJnwUser%t%t%2%n %s%tStartTime%t%t%3%n %s%tHostName%t%t%4%n %s%tComment%t%t%5	%2:ジョブネットワークの所有者名 %3:実績開始時間 %4:実行ホスト名 %5:ジョブネットワークのコメント	
4	情報	JNW_SUB_END	SUBJNW_END%n %s%tJnwName:SubJnwName%t%1%n %s%tJnwUser%t%t%2%n %s%tEndTime%t%t%3%n %s%tJnwStat%t%t%4%n %s%tHostName%t%t%5%n %s%tComment%t%t%6	%1:ジョブネットワーク名:サブジョブネットワーク名 %2:ジョブネットワークの所有者名 %3:サブジョブネットワークの実績終了時間 %4:サブジョブネットワーク終了状態(Normal, Error) %5:実行ホスト名 %6:ジョブネットワークのコメント	SubJNWが終了した
5	情報	UJ_START	JNW_UNITJOB_START%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tHostName%t%t%4%n %s%tComment%t%t%5	%1:ジョブネットワーク名 %2:単位ジョブ名 %3:ジョブネットワークの所有者名 %4:実行ホスト名 %5:単位ジョブのコメント	単位ジョブが開始した
6	情報	UJ_END	JNW_UNITJOB_END%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime-EndTime%t%4 - %5%n %s%tHostName%t%t%6%n %s%tComment%t%t%7	%1:ジョブネットワーク名 %2:単位ジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実績終了時間 %6:実行ホスト名 %7:単位ジョブのコメント	単位ジョブが終了した
7	情報	DIA_START	JNW_DIALOG_START%n %s%tJnwName%t%t%1%n %s%tJnwUser%t%t%2%n %s%tDIALOG:UjName%t%3%n	%1:ジョブネットワーク名 %2:ジョブネットワークの所有者名	ダイアログジョブが開始した

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
			%s%tDIALOG:Dialog-Msg%t%4%n %s%tHostName%t%t%5	%3:ダイアログジョブ名 %4:「ダイアログのメッセージ設定」ダイアログでメッセージの内容 %5:実行ホスト名	
8	情報	DIA_END	JNW_DIALOG_END%n %s%tJnwName%t%t%1%n %s%tJnwUser%t%t%2%n %s%tDIALOG:UjName%t%3%n %s%tDIALOG:Dialog-Msg%t%4%n %s%tHostName%t%t%5	%1:ジョブネットワーク名 %2:ジョブネットワークの所有者名 %3:ダイアログジョブ名 %4:「ダイアログのメッセージ設定」ダイアログでメッセージの内容 %5:実行ホスト名	ダイアログジョブが終了した
9	情報	ERPJ_START	JNW_ERPJOB_START%n %s%tJnwName:ERPjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tHostName%t%t%4	%1:ジョブネットワーク名 %2:ERPジョブ名 %3:ジョブネットワークの所有者名 %4:実行ホスト名	ERPジョブが開始した
10	情報	ERPJ_END	JNW_ERPJOB_END%n %s%tJnwName:ERPjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime-EndTime%t%4 - %5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:ERPジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実績終了時間 %6:実行ホスト名	ERPジョブが終了した
11	警告	JNW_OVT	JNW_TOTAL_TIME_EXCEEDED%n %s%tJnwName%t%t%1%n %s%tJnwUser%t%t%2%n %s%tStartTime%t%t%3%n %s%tWarning Time%t%4%n %s%tAction%t%t%5%n %s%tHostName%t%t%6%n %s%tComment%t%t%7	%1:ジョブネットワーク名 %2:ジョブネットワークの所有者名 %3:実績開始時間 %4:期待の実行終了時刻 %5:動作(forcestop, off, skip) %6:実行ホスト名	JNWの実行時間が設置した期待の実行時間より長かった

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
				%7:ジョブネットワークのコメント	
12	警告	UJ_OVT	JNW_UNITJOB_TIME_EXCEEDED %n %s%tJnwName:UjName%t%1: %2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tWarning Time%t%5%n %s%tAction%t%t%6%n %s%tHostName%t%t%7%n %s%tComment%t%t%8	%1:ジョブネットワーク名 %2:単位ジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:期待の実行終了時刻 %6:動作(forcestop, off, skip) %7:実行ホスト名 %8:単位ジョブのコメント	単位ジョブの実行時間が設置した期待の実行時間より長かった
13	警告	ERPJ_OVT	JNW_ERPJOB_TIME_EXCEEDED %n %s%tJnwName:ERPjName%t%1: %2%n %s%tJnwUser%t%t%3%n %s%tStartTime%t%t%4%n %s%tHostName%t%t%5	%1:ジョブネットワーク名 %2:ERPジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名	ERPジョブの実行時間が設置した期待の実行時間より長かった
14	警告	JNW_CRIS	JNW_CRITICAL_POINT START%n %s%tJnwName%t%t%1%n %s%tJnwUser%t%t%2%n %s%tWarning Time%t%3%n %s%tAction%t%t%4%n %s%tHostName%t%t%5%n %s%tComment%t%t%6	%1:ジョブネットワーク名 %2:ジョブネットワークの所有者名 %3:期待の実行開始時刻 %4:動作(off, skip, hold) %5:実行ホスト名 %6:ジョブネットワークのコメント	期待の実行開始時刻になっても、JNWがまだ実行されなかった
15	警告	JNW_CRIE	JNW_CRITICAL_POINT END%n %s%tJnwName%t%t%1%n %s%tJnwUser%t%t%2%n %s%tWarning Time%t%3%n %s%tAction%t%t%4%n %s%tHostName%t%t%5%n %s%tComment%t%t%6	%1:ジョブネットワーク名 %2:ジョブネットワークの所有者名 %3:期待の実行終了時刻 %4:動作(off, skip, hold, forcestop) %5:実行ホスト名	期待の実行終了時刻になっても、JNWの実行が終了していなかった

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
				%6:ジョブネットワークのコメント	
16	警告	UJ_CRIS	UNITJOB_CRITICAL_POINT START%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tAction%t%t%5%n %s%tHostName%t%t%6%n %s%tComment%t%t%7	%1:ジョブネットワーク名 %2:単位ジョブ名 %3:ジョブネットワークの所有者名 %4:期待の実行開始時刻 %5:動作(off, skip, hold) %6:実行ホスト名 %7:単位ジョブのコメント	期待の実行開始時刻になっても、単位ジョブがまだ実行されなかった
17	警告	UJ_CRIE	UNITJOB_CRITICAL_POINT END %n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tAction%t%t%5%n %s%tHostName%t%t%6%n %s%tComment%t%t%7	%1:ジョブネットワーク名 %2:単位ジョブ名 %3:ジョブネットワークの所有者名 %4:期待の実行終了時刻 %5:動作(off, skip, hold, forcestop) %6:実行ホスト名 %7:単位ジョブのコメント	期待の実行終了時刻になっても、単位ジョブが終了していなかった
18	警告	ERPJ_CRIS	ERPJOB_CRITICAL_POINT START %n %s%tJnwName:ERPjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tAction%t%t%5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:ERPジョブ名 %3:ジョブネットワークの所有者名 %4:期待の実行開始時刻 %5:動作(off,s kip, hold) %6:実行ホスト名	期待の実行開始時刻になっても、ERPジョブがまだ実行されなかった
19	警告	ERPJ_CRIE	ERPJOB_CRITICAL_POINT END %n %s%tJnwName:ERPjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tAction%t%t%5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:ERPジョブ名 %3:ジョブネットワークの所有者名 %4:期待の実行終了時刻	期待の実行終了時刻になっても、ERPジョブの実行が終了していなかった

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
				%5:動作(off, skip, hold, forcestop) %6:実行ホスト名	
20	警告	UJ_WARN	JNW_UNITJOB_WARNING%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sWarningMessage: %6%n %s%tComment%t%t%7	%1:ジョブネットワーク名 %2:単位ジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果 %7:単位ジョブのコメント	単位ジョブのパラメータで警告終了コードを設定して、単位ジョブの終了コードは設定した範囲になった
21	警告	WT_NSTA	JNW_WAITTIME_NOT_START%n %s%tJnwName%t%t%1%n %s%tJnwUser%t%t%2%n %s%tWait Time%t%t%3%n %s%tHostName%t%t%4%n %s%tComment%t%t%5	%1:ジョブネットワーク名 %2:ジョブネットワークの所有者名 %3:待ち合わせ時間 %4:実行ホスト名 %5:時刻待ち合わせジョブのコメント	定義した期待時間が来ても、時刻待ち合わせジョブが実行されなかった
22	エラー	UJ_ERR	JNW_UNITJOB_ERROR%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6%n %s%tComment%t%t%7	%1:ジョブネットワーク名 %2:単位ジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果 %7:単位ジョブのコメント	単位ジョブがエラーになった
23	エラー	UJ_ESTOP	JNW_ERROR_STOP%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6%n %s%tComment%t%t%7	%1:ジョブネットワーク名 %2:単位ジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	JNWパラメータで「エラー時の自動停止」を「停止する」に設置して、ERP、BIではないジョブがエラーになった

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
				%7:単位ジョブのコメント	
24	エラー	ERPJ_ERR	JNW_ERPJOB_ERROR%n %s%tJnwName:ERPjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:ERPジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	ERPジョブがエラーになった
25	エラー	ERPJ_ESTOP	JNW_ERPJOB_ERROR_STOP%n %s%tJnwName:ERPjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:ERPジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	JNWパラメータで「エラー時の自動停止」を「停止する」に設置して、ERPジョブがエラーになった
26	情報	EVTSND_START	JNW_EVENTSEND_START%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tHostName%t%t%4%n %s%tComment%t%t%5	%1:ジョブネットワーク名 %2:イベント送信ジョブ名 %3:ジョブネットワークの所有者名 %4:実行ホスト名 %5:イベント送信ジョブのコメント	イベント送信ジョブが開始した
27	情報	EVTSND_END	JNW_EVENTSEND_END%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime-EndTime%t%4 - %5%n %s%tHostName%t%t%6%n %s%tComment%t%t%7	%1:ジョブネットワーク名 %2:イベント送信ジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実績終了時間 %6:実行ホスト名 %7:イベント送信ジョブのコメント	イベント送信ジョブが終了した
28	警告	EVTSND_OVT	JNW_EVENTSEND_TIMEOUT%n %s%tJnwName:UjName%t%1:%2%n	%1:ジョブネットワーク名	イベント送信ジョブの実行時間が設置した期

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
			%s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tHostName%t%t%5%n %s%tComment%t%t%6	%2:イベント送信ジョブ名 %3:ジョブネットワークの所有者名 %4:タイムアウト時間 %5:実行ホスト名 %6:イベント送信ジョブのコメント	待の実行時間より長かった
29	情報	EVTRCV_START	JNW_EVENTRECV_START%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tHostName%t%t%4%n %s%tComment%t%t%5	%1:ジョブネットワーク名 %2:イベント受信ジョブ名 %3:ジョブネットワークの所有者名 %4:実行ホスト名 %5:イベント受信ジョブのコメント	イベント受信ジョブが開始した
30	情報	EVTRCV_END	JNW_EVENTRECV_END%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime-EndTime%t%4 - %5%n %s%tHostName%t%t%6%n %s%tComment%t%t%7	%1:ジョブネットワーク名 %2:イベント受信ジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実績終了時間 %6:実行ホスト名 %7:イベント受信ジョブのコメント	イベント受信ジョブが終了した
31	警告	EVTRCV_OVT	JNW_EVENTRECV_TIMEOUT%n %s%tJnwName:UjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tHostName%t%t%5%n %s%tComment%t%t%6	%1:ジョブネットワーク名 %2:単位ジョブ名 %3:ジョブネットワークの所有者名 %4:タイムアウト時間 %5:実行ホスト名 %6:イベント受信ジョブのコメント	イベント受信ジョブの実行時間が設置した期待の実行時間より長かった
32	情報	BIJ_START	JNW_BIJOB_START%n %s%tJnwName:BIjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tHostName%t%t%4	%1:ジョブネットワーク名 %2:BIジョブ名 %3:ジョブネットワークの所有者名	BIジョブが開始した

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
				%4:実行ホスト名	
33	情報	BIJ_END	JNW_BIJOB_END%n %s%tJnwName:BIjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime-EndTime%t%4 - %5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:BIジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実績終了時間 %6:実行ホスト名	BIジョブが終了した
34	エラー	BIJ_ERR	JNW_BIJOB_ERROR%n %s%tJnwName:BIjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:BIジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	BIジョブがエラーになった
35	エラー	BIJ_ESTOP	JNW_BIJOB_ERROR_STOP%n %s%tJnwName:BIjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:BIジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	BIジョブパラメータで「エラー時の自動停止」を「停止する」に設置して、BIジョブがエラーになった
36	警告	HLDEVT_OVW	JNW_HOLDEVENT_OVERWRITE%n %s%tEvtHost:ObjName%t%1:%2.%3/%4%n %s%tEVENTID%t%5%n %s%tHostName%t%t%6	%1:ホスト名 %2:ジョブネットワーク名 %3:トラッカID %4:イベント送信ジョブ名 %5:イベントID %6:実行ホスト名	新しい送信ジョブのイベントIDはHoldList中のイベント送信ジョブのと同じ、古いイベント情報が上書された
37	警告	HLDEVT_MAX	JNW_HOLDEVENT_MAX%n %s%tEvtHost:ObjName%t%1:%2.%3/%4%n %s%tEVENTID%t%5%n %s%tHostName%t%t%6	%1:ホスト名 %2:ジョブネットワーク名 %3:トラッカID %4:イベント送信ジョブ名 %5:イベントID	送信ジョブの最大HoldList数が到着して、Topイベントが削除された

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
				%6:実行ホスト名	
38	警告	HLDEVT_OVT	JNW_HOLDEVENT_TIMEOUT%n %s%tEvtHost:ObjName%t%1:%2.%3/%4%n %s%tEVENTID%t%5%n %s%tHostName%t%t%6	%1:ホスト名 %2:ジョブネットワーク名 %3:トラッカID %4:イベント送信ジョブ名 %5:イベントID %6:実行ホスト名	送信ジョブのイベント保持期間がタイムアウトしてイベントが消失した。
39	情報	UCXSJ_START	JNW_UCXSJOB_START%n %s%tJnwName:UCXSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tHostName%t%t%4	%1:ジョブネットワーク名 %2:UCXSingleジョブ名 %3:ジョブネットワークの所有者名 %4:実行ホスト名	UCXSingleジョブが開始した
40	情報	UCXSJ_END	JNW_UCXSJOB_END%n %s%tJnwName:UCXSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime-EndTime%t%4 - %5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:UCXSingleジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実績終了時間 %6:実行ホスト名	UCXSingleジョブが終了した
41	エラー	UCXSJ_ERR	JNW_UCXSJOB_ERROR%n %s%tJnwName:UCXSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:UCXSingleジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	UCXSingleジョブがエラーになった
42	エラー	UCXSJ_ESTOP	JNW_UCXSJOB_ERROR_STOP%n %s%tJnwName:UCXSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:UCXSingleジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名	JNWパラメータで「エラー時の自動停止」を「停止する」に設置して、UCXSingleジョブがエラーになった

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
				%6:エラー出力結果	
43	警告	UCXSJ_OVT	JNW_UCXSJOB_TIME_EXCEEDED %n %s%tJnwName:UCXSjName%t %1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime%t%t%4%n %s%tHostName%t%t%5	%1:ジョブネットワーク名 %2:UCXSingleジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名	UCXSingleジョブの実行時間が設置した期待の実行時間より長かった
44	警告	UCXSJ_CRIS	UCXSJOB_CRITICAL_POINT START%n %s%tJnwName:UCXSjName%t %1:%2%n %s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tAction%t%t%5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:UCXSingleジョブ名 %3:ジョブネットワークの所有者名 %4:期待の実行開始時刻 %5:動作(off, skip, hold) %6:実行ホスト名	期待の実行開始時刻になっても、UCXSingleジョブがまだ実行されなかった
45	警告	UCXSJ_CRIE	UCXSJOB_CRITICAL_POINT END %n %s%tJnwName:UCXSjName%t %1:%2%n %s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tAction%t%t%5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:UCXSingleジョブ名 %3:ジョブネットワークの所有者名 %4:期待の実行終了時刻 %5:動作(off, skip, hold, forcestop) %6:実行ホスト名	期待の実行終了時刻になっても、UCXSingleジョブが終了していなかった
46	情報	PCJ_START	JNW_PCJOB_START%n %s%tJnwName:PCjName%t%1: %2%n %s%tJnwUser%t%t%3%n %s%tHostName%t%t%4	%1:ジョブネットワーク名 %2:PCジョブ名 %3:ジョブネットワークの所有者名 %4:実行ホスト名	PCジョブが開始した
47	情報	PCJ_END	JNW_PCJOB_END%n% tJnwName:PCjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime-EndTime%t%4 - %5%n	%1:ジョブネットワーク名 %2:PCジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間	PCジョブが終了した

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
			%s%tHostName%t%t%6	%5:実績終了時間 %6:実行ホスト名	
48	エラー	PCJ_ERR	JNW_PCJOB_ERROR%n %s%tJnwName:PCjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:PCジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	PCジョブがエラーになった
49	エラー	PCJ_ESTOP	JNW_PCJOB_ERROR_STOP%n %s%tJnwName:PCjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:PCジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	PCジョブパラメータで「エラー時の自動停止」を「停止する」に設置して、PCジョブがエラーになった
50	情報	WOBSJ_START	JNW_WOBSJOB_START%n %s%tJnwName:WOBSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tHostName%t%t%4	%1:ジョブネットワーク名 %2:WOBSジョブ名 %3:ジョブネットワークの所有者名 %4:実行ホスト名	WOBSジョブが開始した
51	情報	WOBSJ_END	JNW_WOBSJOB_END%n %s%tJnwName:WOBSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime-EndTime%t%4 - %5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:WOBSジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実績終了時間 %6:実行ホスト名	WOBSジョブが終了した
52	エラー	WOBSJ_ERR	JNW_WOBSJOB_ERROR%n %s%tJnwName:WOBSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:WOBSジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名	WOBSジョブがエラーになった

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
				%6:エラー出力結果	
53	エラー	WOBSJ_ESTOP	JNW_WOBSJOB_ERROR_STOP%n %s%tJnwName:WOBSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n %s%tHostName%t%t%5%n %sErrorMessage: %6	%1:ジョブネットワーク名 %2:WOBSジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	JNWパラメータで「エラー時の自動停止」を「停止する」に設置して、WOBSジョブがエラーになった
54	警告	WOBSJ_OVT	JNW_WOBSJOB_TIME_EXCEEDED%n %s%tJnwName:WOBSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tStartTime%t%t%4%n %s%tHostName%t%t%5	%1:ジョブネットワーク名 %2:WOBSジョブ名 %3:ジョブネットワークの所有者名 %4:実績開始時間 %5:実行ホスト名	WOBSジョブの実行時間が設置した期待の実行時間より長かった
55	警告	WOBSJ_CRIS	WOBSJOB_CRITICAL_POINT START%n %s%tJnwName:WOBSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tAction%t%t%5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:WOBSジョブ名 %3:ジョブネットワークの所有者名 %4:期待の実行開始時刻 %5:動作(off, skip, hold) %6:実行ホスト名	期待の実行開始時刻になっても、WOBSジョブがまだ実行されなかった
56	警告	WOBSJ_CRIE	WOBSJOB_CRITICAL_POINT END%n %s%tJnwName:WOBSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tWarning Time%t%4%n %s%tAction%t%t%5%n %s%tHostName%t%t%6	%1:ジョブネットワーク名 %2:WOBSジョブ名 %3:ジョブネットワークの所有者名 %4:期待の実行終了時刻 %5:動作(off, skip, hold, forcestop) %6:実行ホスト名	期待の実行終了時刻になっても、WOBSジョブが終了していなかった
57	警告	WOBSJ_WARN	WOBSJOB_WARNING%n %s%tJnwName:WOBSjName%t%1:%2%n %s%tJnwUser%t%t%3%n %s%tSTART%t%t%4%n	%1:ジョブネットワーク名 %2:WOBSジョブ名 %3:ジョブネットワークの所有者名	WOBSジョブのパラメータで警告終了コードを設定して、WOBSジョブの終了

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
			%s%tHostName%t%t%5%n %sWarningMessage: %6	%4:実績開始時間 %5:実行ホスト名 %6:エラー出力結果	コードは設定した範囲になった

■カスタムジョブは実行上単位ジョブと同様に扱い、イベント出力も単位ジョブと同様になります。カスタムジョブをイベント連携に使用する場合は、JNW_UNITJOB_*** および UNITJOB_*** メッセージを参照してください。

■UCX SingleジョブはR14.1で廃止されました。

表12.5 イベント一覧(ソース名「NetShepherd」)

ID	種類	選択イベント名	説明のメッセージフォーマット	メッセージフォーマット変数の補足	出力の要因
29	警告	QUE_SSTOP	異常が発生したためキュー(%1：状態=%2)を停止しました(022)。 キュー、ジョブ、システム等の状態を確認し、原因を取り除いた後にキューを開始してください。	%1:キュー名 %2:キューの状態を表す数値	左記説明のとおり
4103	情報	QUE_USTOP	ユーザがキュー(%1)を停止しました。	%1:キュー名	左記説明のとおり
4104	情報	QUE_USTART	ユーザがキュー(%1)を開始しました。	%1:キュー名	左記説明のとおり

12.3.3. Micro Focus Operations Manager software

Windows版のJobCenterはMicro Focus Operations Manager software(旧OVO)と連携して、ジョブの実行や状態などをイベントとして通知できます。

1. 管理者アカウントで Windows にログオンし、Windows の[スタート]-[プログラム]-[JobCenter]-[SV]-[サーバの環境設定]を選択して、サーバの環境設定を起動します。

2. 左ペインの「サイト」を選択します。

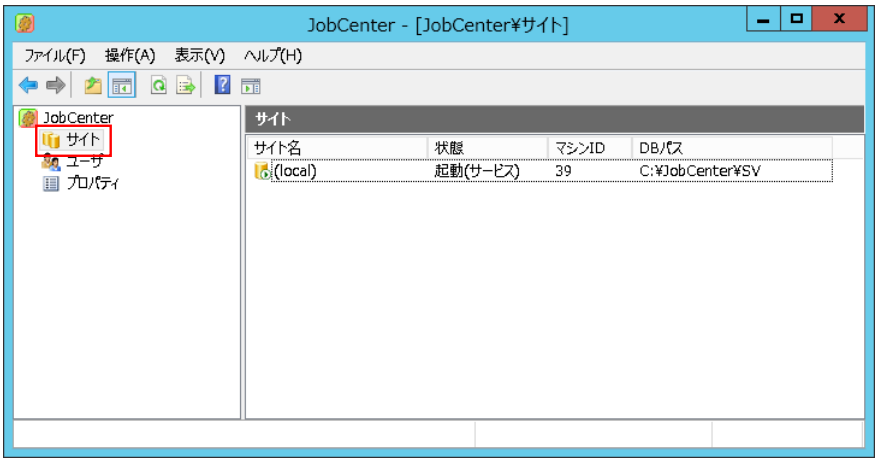

図12.19 ［サーバの環境設定］のサイト画面例

3. イベント連携(OPCMSG　　連携)を行いたいサイトをダブルクリックして、サイトのプロパティを表示し、「OPCMSG」タブを開きます。

図12.20 ［詳細設定］ダイアログ－［OPCMSG］画面例

［イベントタブ］画面において、イベント名にチェックを付けたイベントのみがイベントとして通知されますので、OPCMSG連携を行う場合は［イベント設定］タブで通知したいイベントを採取するように設定してください。

OPCMSGタブの設定項目は次のとおりです。

■OPCメッセージを送信する

OPCMSG連携機能を使用するかしないかを選択します。

ON	OPCMSG連携機能を使用します。
OFF	OPCMSG連携機能を使用しません。

■パラメータ

Micro Focus Operations Manager softwareへ通知するイベントのパラメータを設定します。

■APL

アプリケーション名を指定します。

■OBJ

オブジェクト名を指定します。

■GRP

メッセージグループを指定します。

■イベント名

Micro Focus Operations Manager softwareに通知可能なイベント一覧です。

実際にMicro Focus Operations Manager softwareに通知するには、上記の注意事項に記載したとおり ［イベント設定］ タブで通知したいイベントを採取するように設定してください。

イベントの詳細は、「12.3.5 イベントの種類」を参照してください。

■セベリティ

Micro Focus Operations Manager softwareに通知するイベントは

- NORMAL
- WARNING
- MINOR
- MAJOR
- CRITICAL

のいずれかのセベリティを持ち、変更できます(デフォルトはNORMAL)。

セベリティを変更するには変更したいイベントをダブルクリックするか、イベントを選択して「修正」ボタンを押してください。

以下のダイアログが表示され、ドロップダウンリストから変更するセベリティを選択します。

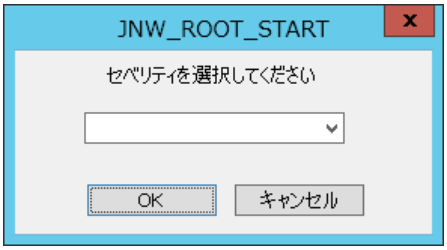

図12.21 セベリティ選択画面

■ ［キャンセル］ ボタン

設定した変更をキャンセルします。

■ ［OK］ または ［適用］ ボタン

設定を有効にします。次のダイアログが表示されるので、JobCenterを再起動すると本設定が有効になります。

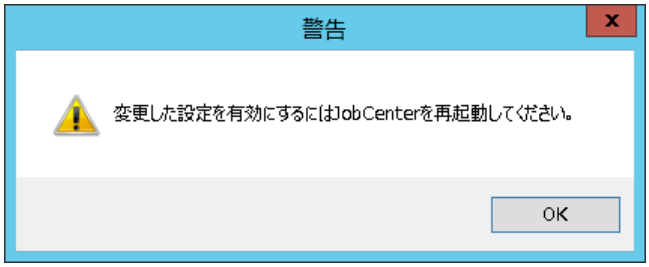

図12.22 再起動画面

12.3.4. イベント定義ファイル

■パス

ローカルサイト	%InstallDirectory%\spool\conf\JNWEXEjobmsg.conf
クラスタサイト	共有ディスク上の <JobCenterDBパス>\spool\conf\JNWEXEjobmsg.conf

■説明

本ファイルは、イベント送信先に出力するフォーマットを指定します。ただしイベントログ出力のフォーマットは「12.3.2.3 Windowsイベントビューア アプリケーションログ」のフォーマット固定であり、本ファイルの指定は反映されません。

フォーマットは次のとおりです。

```
EVENT <Event-Name> <Message-Body>
```

Event-Name	イベントの種類を表すイベントタイプ名の文字列です。
Message-Body	出力するメッセージパターンを指定します。

［メッセージ出力パターン］

一般に、EVENTタグのMessage-Body指定した文字列はそのまま出力されます。ただし以下の一覧表のように%でくくられた文字列はマクロ文字として解釈されます。

マクロ文字は、それぞれメッセージの内容に従って決まったパターンに変換されます。

1つのイベント中で使用可能なマクロの数は10個です。これを超えてマクロを使用するとマクロの部分は無視されます。

マクロ	変換対象	指定可能なイベント名
%jnwname%	ジョブネットワーク名に変換されます。	jnwsv.*
%starttime%	ジョブネットワークまたは単位ジョブの投入時刻に"%H:%M:%S" のフォーマットで変換されます。 ただし、jnwsv.*.cris の場合は"%Y/%m/%d %H:%M:%S" のフォーマットで変換されます。	jnwsv.*
%endtime%	ジョブネットワークまたは単位ジョブの終了時刻に"%H:%M:%S" のフォーマットで変換されます。 ただし、jnwsv.*.crie の場合は"%Y/%m/%d %H:%M:%S" のフォーマットで変換されます。	jnwsv.*.end jnwsv.*.crie jnwsv.uj.err jnwsv.uj.estop
%ujname%	メッセージに対する単位ジョブ名に変換されます。	jnwsv.dialog.* jnwsv.uj.*

マクロ	変換対象	指定可能なイベント名
		jnwsv.holdevent.*
%ujexecmac%	メッセージに対応する単位ジョブの実行マシン名に変換されます。	jnwsv.uj.end jnwsv.holdevent.*
%ujerrmsg%	エラーメッセージに変換されます。	jnwsv.uj.err jnwsv.uj.estop
%dialog-msg%	ダイアログメッセージに変換されます。	jnwsv.*
%jnwuser%	ジョブネットワークの投入者に変換されます。	jnwsv.jnw.* jnwsv.uj.*
%jnwstat%	ジョブネットワークの状態（NORMAL、ERROR）に変換されます。	jnwsv.jnw.*.end
%trkid%	トラッカIDに変換されます。	jnwsv.holdevent.*
%evtid%	イベントIDに変換されます。	jnwsv.holdevent.*
%sitename%	jnwcasterが起動されているホスト名、またはクラスタサイト名に変換されます。	notice.daemon.* (OPCMSGタグのみ)
%comment%	ジョブネットワークや部品に設定されているコメント文字列に変換されます。	jnwsv.jnw.* jnwsv.eventsend.* jnwsv.eventrecv.* jnwsv.uj.* jnwsv.waitime.notstart
%trkpath%	イベントが発生したトラッカをCL/Webで参照するためのURLのパス部分に変換されます。	jnwsv.*
%partspath%	イベントが発生したトラッカをCL/Webで参照するためのURLのパス部分に変換されます。	jnwsv.uj.* jnwsv.sapj.* jnwsv.sbwj.* jnwsv.spcj.* jnwsv.ucxsj.* jnwsv.wobsj.*

%comment%マクロはR15.5から利用できます。R15.5より前のバージョンからR15.5以降にバージョンアップした場合、イベント定義ファイルがバージョンアップによって引き継がれるため、%comment%マクロがイベント定義ファイルには追加されていない状態になります。%comment%マクロを利用したい場合は、手動で追加してください。

12.3.5. イベントの種類

Windowsイベントログへの出力、イベントログファイルへの出力、APIによるイベント通知時に選択する　イベントの名称と意味については、以下の表を参照してください。

表12.6 イベント名とイベントの意味一覧

選択イベント名	出力イベントメッセージ	イベントの意味
BIJ_END	JNW_BIJOB_END	BIジョブが終了しました。
BIJ_ERR	JNW_BIJOB_ERROR	BIジョブがエラー終了しました。

選択イベント名	出力イベントメッセージ	イベントの意味
BIJ_ESTOP	JNW_BIJOB_ERROR_STOP	BIジョブの実行がエラー停止しました。
BIJ_START	JNW_BIJOB_START	BIジョブが開始しました。
DIA_END	JNW_DIALOG_END	ダイアログ待ち状態から再開しました。
DIA_START	JNW_DIALOG_START	ダイアログ待ちになりました。
EVTRCV_END	JNW_EVENTRECEIVE_END	イベント受信が終了しました。
EVTRCV_OVT	JNW_EVENTRECEIVE_TIMEOUT	イベント受信がタイムアウトしました。
EVTRCV_START	JNW_EVENTRECEIVE_START	イベント受信が開始されました。
EVTSND_END	JNW_EVENTSEND_END	イベント送信が終了しました。
EVTSND_OVT	JNW_EVENTSEND_TIMEOUT	イベント送信がタイムアウトしました。
EVTSND_START	JNW_EVENTSEND_START	イベント送信が開始されました。
HLDEVT_MAX	JNW_HOLDEVENT_MAX	保留可能なイベント数を超過しました。
HLDEVT_OVT	JNW_HOLDEVENT_TIMEOUT	イベントが有効期限切れで消失しました。
HLDEVT_OVW	JNW_HOLDEVENT_OVERWRITE	イベントが上書きされました。
JNW_CRIE	JNW_CRITICAL_POINT_CHECK_END	ジョブネットワークで設定されているクリティカルポイント警告時刻(終了時)を超過しました。
JNW_CRIS	JNW_CRITICAL_POINT_CHECK_START	ジョブネットワークで設定されているクリティカルポイント警告時刻(開始時)を超過しました。
JNW_OVT	JNW_TOTAL_TIME_EXCEEDED	ジョブネットワークの実行が予定時間を超過しました。
JNW_ROOT_END	JNW_END	親ジョブネットワークの実行が終了しました。
JNW_ROOT_START	JNW_START	親ジョブネットワークの実行が開始されました。
JNW_SUB_END	JNW_END	サブジョブネットワークの実行が終了しました。
JNW_SUB_START	JNW_START	サブジョブネットワークの実行が開始されました。
ERPJ_CRIE	JNW_ERPJOB_CRITICAL_POINT	ERPジョブで設定されているクリティカルポイント警告時刻(終了時)を超過しました。
ERPJ_CRIS	JNW_ERPJOB_CRITICAL_POINT	ERPジョブで設定されているクリティカルポイント警告時刻(開始時)を超過しました。
ERPJ_END	JNW_ERPJOB_END	ERPジョブが終了しました。

選択イベント名	出力イベントメッセージ	イベントの意味
ERPJ_ERR	JNW_ERPJOB_ERROR	ERPジョブがエラー終了しました。
ERPJ_ESTOP	JNW_ERPJOB_ERROR_STOP	ERPジョブの実行がエラー停止しました。
ERPJ_OVT	JNW_ERPJOB_TIME_EXCEEDED	ERPジョブの実行が予定時間を超過しました。
ERPJ_START	JNW_ERPJOB_START	ERPジョブが開始しました。
UJ_CRIE	JNW_UNITJOB_CRITICAL_POINT_CHECK_END	単位ジョブで設定されているクリティカルポイント警告時刻(終了時)を超過しました。
UJ_CRIS	JNW_UNITJOB_CRITICAL_POINT_CHECK_START	単位ジョブで設定されているクリティカルポイント警告時刻(開始時)を超過しました。
UJ_END	JNW_UNITJOB_END	単位ジョブが終了しました。
UJ_ERR	JNW_UNITJOB_ERROR	単位ジョブがエラー終了しました。
UJ_ESTOP	JNW_ERROR_STOP	単位ジョブの実行がエラー停止しました。
UJ_OVT	JNW_UNITJOB_TIME_EXCEEDED	単位ジョブの実行が予定時間を超過しました。
UJ_START	JNW_UNITJOB_START	単位ジョブが開始しました。
UJ_WARN	JNW_UNITJOB_WARNING	単位ジョブの実行が警告終了しました。
UCXSJ_CRIE	JNW_UCXSJOB_CRITICAL_POINT_CHECK_END	UCXSingleジョブで設定されているクリティカルポイント警告時刻(終了時)を超過しました。
UCXSJ_CRIS	JNW_UCXSJOB_CRITICAL_POINT_CHECK_START	UCXSingleジョブで設定されているクリティカルポイント警告時刻(開始時)を超過しました。
UCXSJ_END	JNW_UCXSJOB_END	UCXSingleジョブが終了しました。
UCXSJ_ERR	JNW_UCXSJOB_ERROR	UCXSingleジョブがエラー終了しました。
UCXSJ_ESTOP	JNW_UCXSJOB_ERROR_STOP	UCXSingleジョブの実行がエラー停止しました。
UCXSJ_OVT	JNW_UCXSJOB_TIME_EXCEEDED	UCXSingleジョブの実行が予定時間を超過しました。
UCXSJ_START	JNW_UCXSJOB_START	UCXSingleジョブが開始しました。
PCJ_END	JNW_PCJOB_END	PCジョブが終了しました。
PCJ_ERR	JNW_PCJOB_ERROR	PCジョブがエラー終了しました。
PCJ_ESTOP	JNW_PCJOB_ERROR_STOP	PCジョブの実行がエラー停止しました。
PCJ_START	JNW_PCJOB_START	PCジョブが開始しました。

選択イベント名	出力イベントメッセージ	イベントの意味
WOBSJ_CRIE	JNW_WOBSJOB_CRITICAL_POINT_CHECK_END	WOBSジョブで設定されているクリティカルポイント警告時刻(終了時)を超過しました。
WOBSJ_CRIS	JNW_WOBSJOB_CRITICAL_POINT_CHECK_START	WOBSジョブで設定されているクリティカルポイント警告時刻(開始時)を超過しました。
WOBSJ_END	JNW_WOBSJOB_END	WOBSジョブが終了しました。
WOBSJ_ERR	JNW_WOBSJOB_ERROR	WOBSジョブがエラー終了しました。
WOBSJ_ESTOP	JNW_WOBSJOB_ERROR_STOP	WOBSジョブの実行がエラー停止しました。
WOBSJ_OVT	JNW_WOBSJOB_TIME_EXCEEDED	WOBSジョブの実行が予定時間を超過しました。
WOBSJ_START	JNW_WOBSJOB_START	WOBSジョブが開始しました。
WOBSJ_WARN	JNW_WOBSJOB_WARNING	WOBSジョブの実行が警告終了しました。
WT_NSTA	JNW_WAITTIME_NOT_START	時刻待ち合わせ部品で指定されている時刻に開始状態になりませんでした。
QUE_USTART	-	ユーザ操作によって、キューが開始しました。
QUE_USTOP	-	ユーザ操作によって、キューが停止しました。
QUE_SSTOP	-	JobCenterが、キューを停止しました。
RELEASE_START	RELEASE_START (HostName=<ホスト名>, User=<ユーザ名>, EstimateTime=<リリース予定時刻>).	計画リリースが開始されました。
RELEASE_END	RELEASE_END (HostName=<ホスト名>, User=<ユーザ名>, EstimateTime=<リリース予定時刻>).	計画リリースが正常終了しました。
RELEASE_ERROR	RELEASE_ERROR (HostName=<ホスト名>, User=<ユーザ名>, EstimateTime=<リリース予定時刻>).	計画リリースが異常終了しました。

■カスタムジョブは実行上単位ジョブと同様に扱い、イベント出力も単位ジョブと同様になります。カスタムジョブをイベント連携に使用する場合は、UJ_*** のイベント名を選択してください。

■UCX SingleジョブはR14.1で廃止されました。

13. サーバの環境設定(Windows版)

「サーバの環境設定」は、Windows版JobCenter(MG/SV)の環境設定を行うGUIプログラムです。

サーバの環境設定では、JobCenterの起動状態、JobCenterのユーザ管理、ログの設定等、Windows版JobCenter(MG/SV)の運用に関する設定を行うことができます。

13.1. サーバの環境設定の起動

1. Windowsの［スタート］から、［すべてのプログラム］－［JobCenter］－［SV］－［サーバの環境設定］を選択します。

> サーバの環境設定の起動や操作は、JobCenter管理者ユーザで行ってください。

2. サーバの環境設定の起動すると、以下のGUI画面が表示されます。

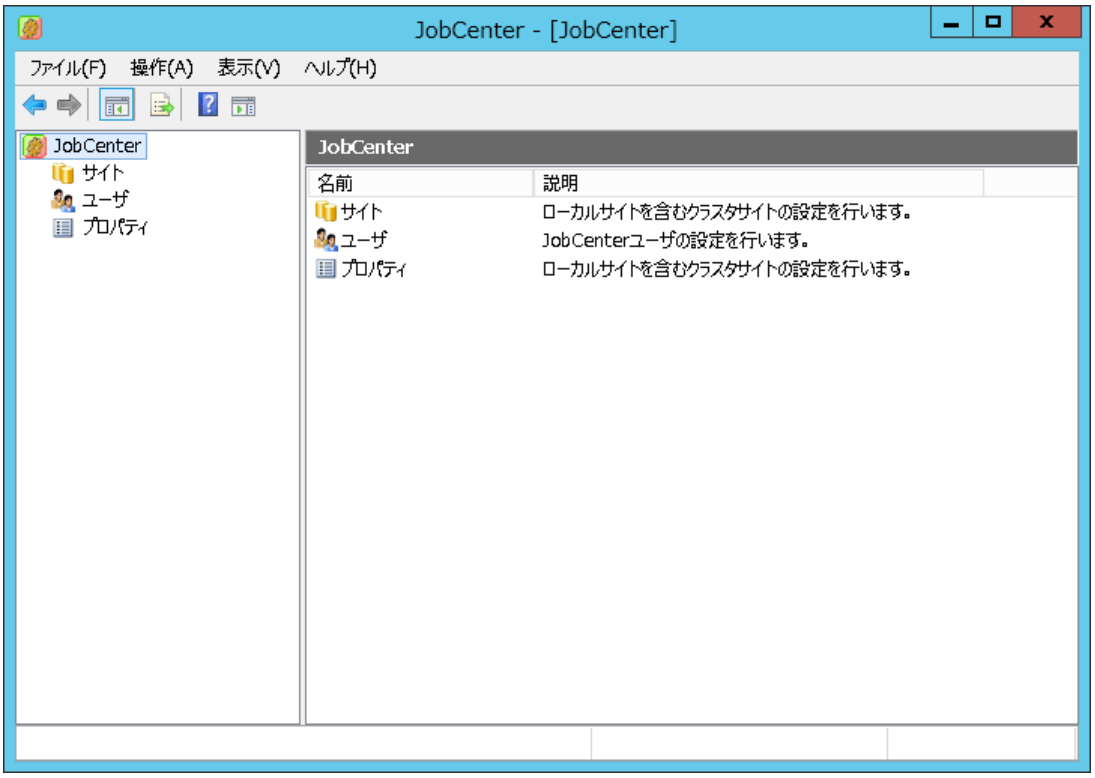

図13.1 サーバの環境設定の起動画面例

13.2. バージョン情報の確認

メニューバーの［ヘルプ］－［JobCenterの環境設定］を選択すると以下のダイアログが表示されます。

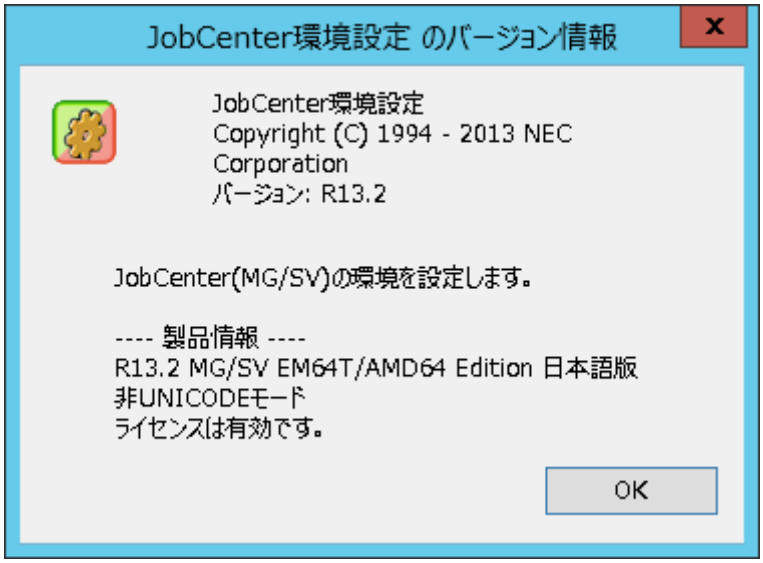

図13.2 バージョン情報画面例

バージョン情報ダイアログの製品情報から以下のことが確認できます。

■インストールタイプ

JobCenter のバージョン、アーキテクチャ、言語タイプが確認できます。

(例) R13.2 MG/SV EM64T/AMD64 Edition 日本語版

■UNICODEモード

インストール時に、UNICODE モードでインストールしたか、非 UNIOCODE モードでインストールしたかを確認できます

■ライセンス状態

JobCenter のライセンスが有効か無効かを確認できます。

■ライセンスが無効になっている場合は、 <インストールガイド>を参照してコードワードが正しく登録されているか確認してください。

■ライセンス状態は、お試し期間中(インストール後60日)はコードワードの登録の有無に関わらず有効になります。

13.3. サイトの設定

左ペインのツリーから「サイト」を選択します。選択後の右ペインには、現在登録されているサイトが表示されます。

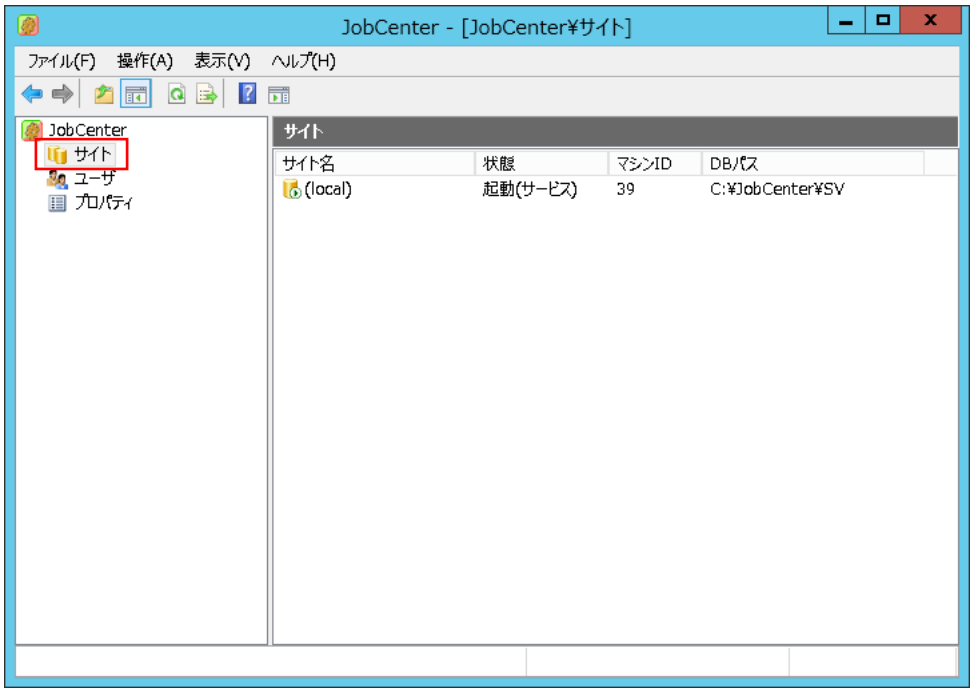

図13.3 サイト画面例

サイトとはJobCenterの実行環境のことです。インストール後には必ずローカルサイト(local)が存在します。

クラスタ環境を構築したい場合は、別のサイト(クラスタサイト)を作成して追加登録する必要があります。クラスタ環境の構築については <クラスタ機能利用の手引き>を参照してください。

図13.3「サイト画面例」の右ペインに表示されるサイト情報の意味は次のとおりです。

表13.1 サイトの列項目

項　目	説　明	例
サイト名	サイトを識別する名前。ローカルサイトの名前は「(local)」固定。	(local)
状態	サイトの状態。状態の詳細は表13.2「サイトの状態」を参照。	起動(サービス)
マシンID	サイトのマシンID。 マシンIDはサイトを一意に識別するための1～2147483647の数値です。ローカルサイトの場合はインストール時に指定したIDが表示されます。	1000
DBパス	サイトDBのパス。(ローカルサイトの場合はインストールしたパス)	C:\JobCenter\SV

表13.2 サイトの状態

状　態	説　明
起動(サービス)	サイトがWindowsのサービスとして起動しています。 インストール直後はローカルサイトはサービスとして起動します。

起動(cjcpw)	サイトがcjcpwで起動しています。 cjcpwについては<クラスタ機能利用の手引き>を参照してください。
起動(非監視)	サイトが非監視モードで起動しています。 非監視モードについては<クラスタ機能利用の手引き>を参照してください。
停止	サイトが停止しています。
非接続	サイトDBにアクセスできません。 クラスタ環境の場合、共有ディスクが切り離されている等の理由でサイトDBにアクセスできない場合に「非接続」の状態になります。
不正	サイトのプロセスが不正な状態です。 サイトに必要なプロセスが一部起動していない状態です。この状態はサイト起動時や停止時に一時的になる可能性があります。その場合は、F5キーやメニューバーの「操作」－「最新の情報に更新」を選択して、状態を最新にして再確認してください。

13.3.1. サイトのプロパティ

右ペインでプロパティを表示したいサイトをダブルクリックするか、右クリックで「プロパティ」を選択すると、プロパティダイアログが表示されます。

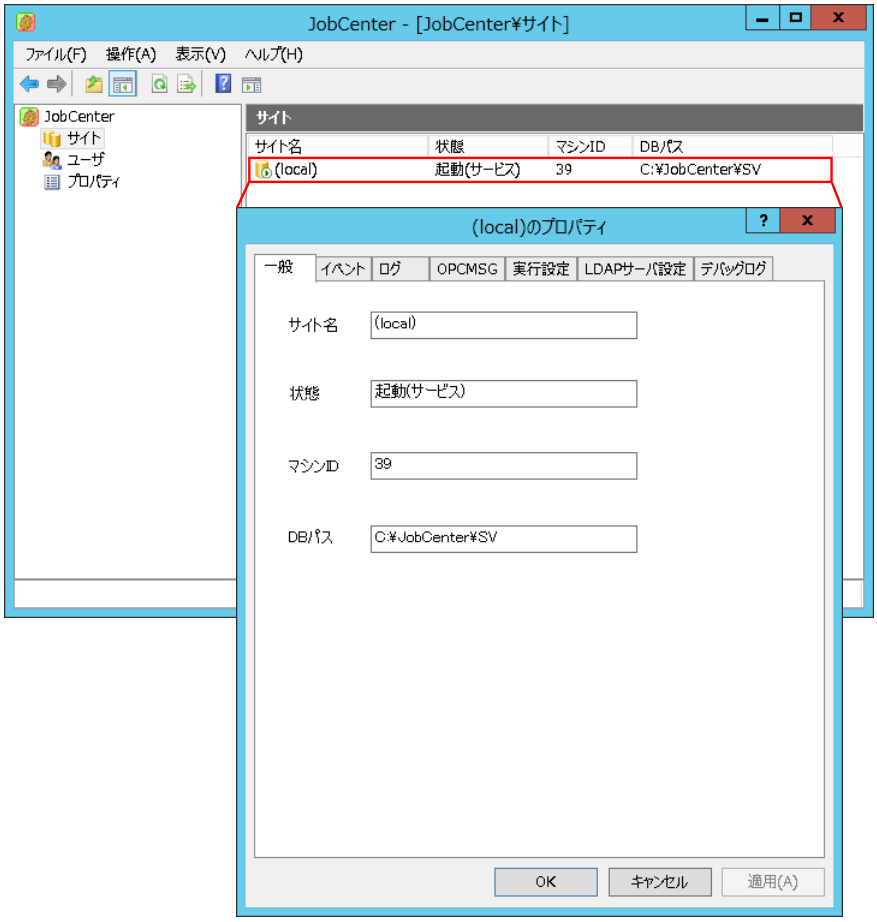

図13.4 サイトのプロパティ画面例

■一般タブ

一般タブでは、サイトの右ペインに表示されていた項目と同じ内容が確認できます。

図13.5 一般タブの画面例

■イベントタブ

イベントタブでは、イベント連携のための設定を行います。詳細は「12.3.2.2 通知するイベントの選択」を確認してください。

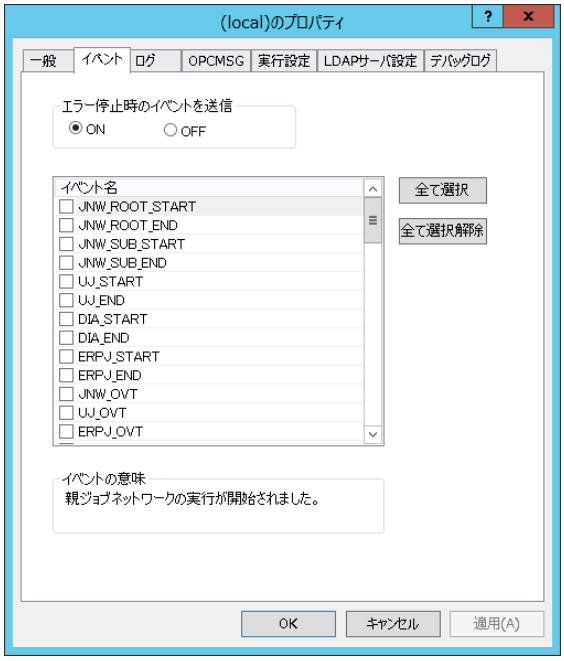

図13.6 イベントタブの画面例

■ログタブ

ログタブでは、JobCenterのイベントをログファイルに出力するための設定を行います。詳細は「12.3.2.2 通知するイベントの選択」を確認してください。

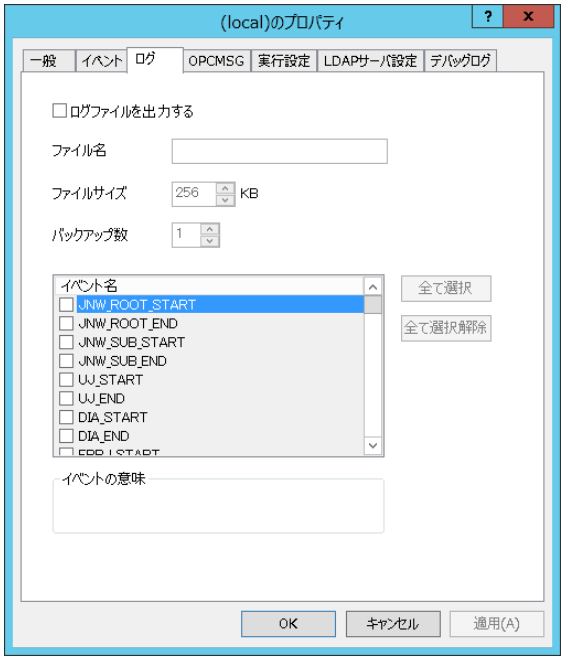

図13.7 ログタブの画面例

■OPCMSGタブ

OPCMSGタブでは、OPCMSGを用いてイベント連携を行うための設定を行います。詳細は「12.3.3 Micro Focus Operations Manager software」を確認してください。

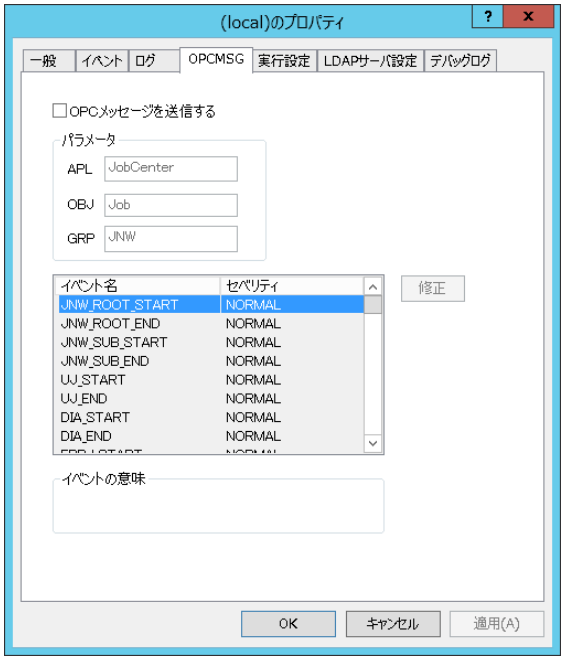

図13.8 OPCMSGタブの画面例

■実行設定タブ

実行設定タブでは、単位ジョブの実行設定を行います。 詳細は「13.3.3 ジョブの実行設定」を確認してください。

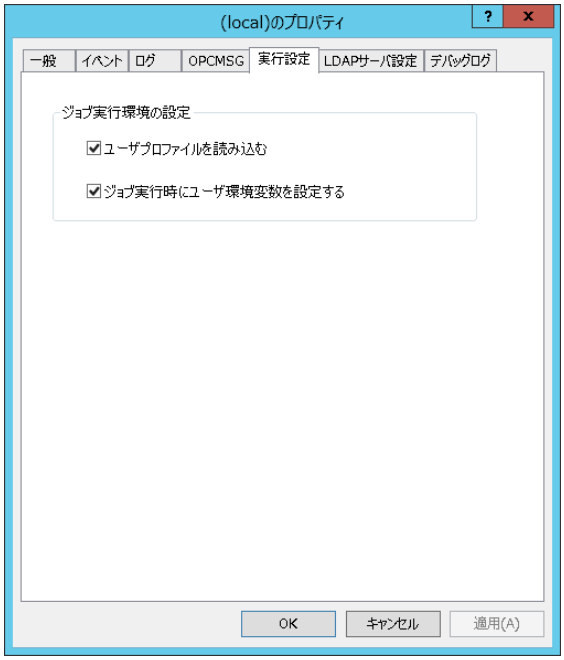

図13.9 実行設定タブの画面例

■LDAPサーバ設定タブ

LDAPサーバ設定タブでは、LDAPを用いてJobCenterのユーザ権限を管理するための設定を行います。詳細は「11.1 LDAPサーバ連携(Windows)」を確認してください。

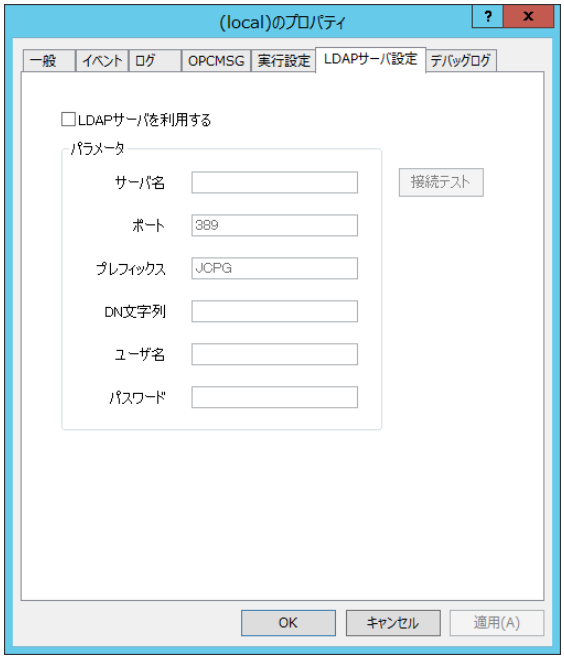

図13.10 LDAPサーバ設定タブの画面例

■デバッグログタブ

デバッグログタブでは、JobCenterの常駐プロセス及び内部コマンドが出力するエラーログと詳細ログ（デバッグログ）の設定を行います。詳細は「25.2　エラーログファイルの設定を変更する」を確認してください。

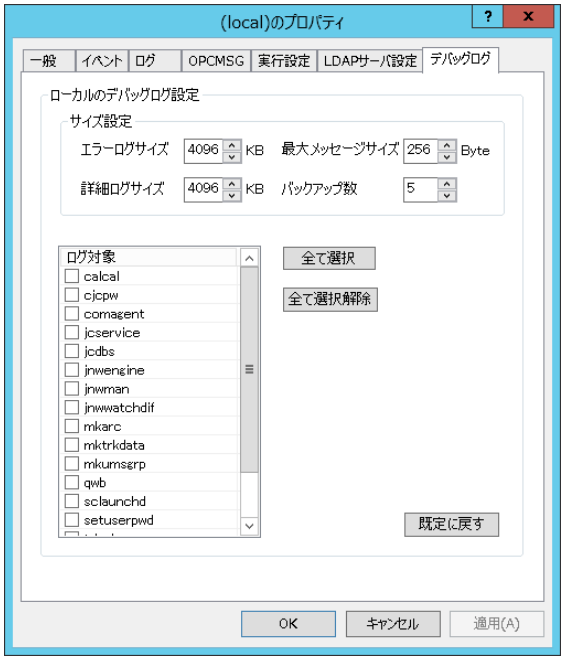

図13.11 デバッグログタブの画面例

13.3.2. サイトの起動と停止

サイトを起動する場合は、サイトの状態が「停止」になっていることを確認してください。次に右クリックで「起動(サービス)」を選択してサイトを起動します。

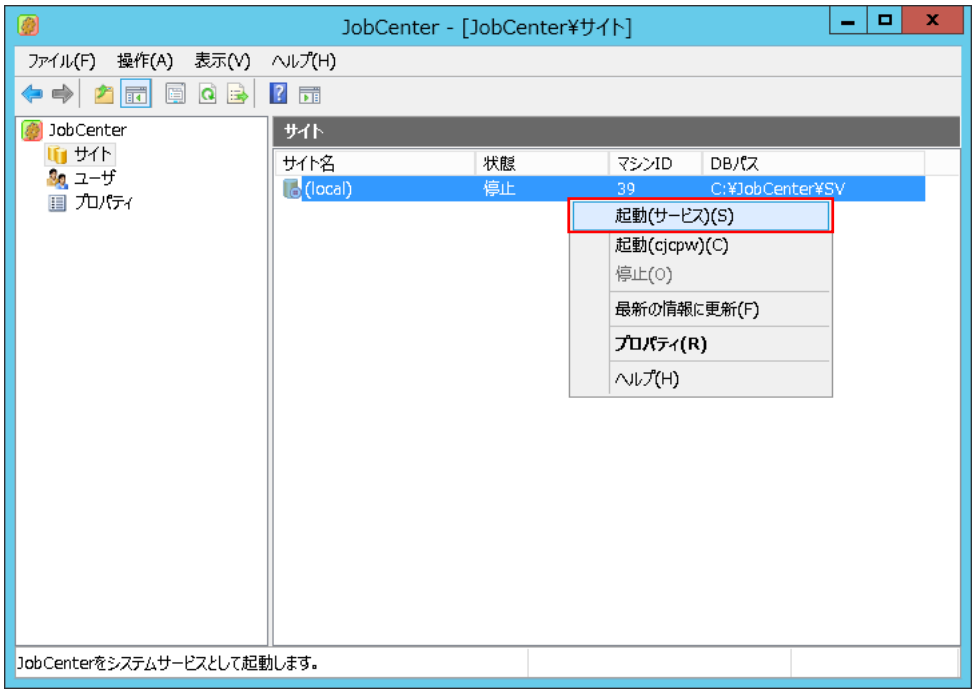

図13.12 サイトの起動画面例

サイトを停止する場合は、サイトの状態が「起動(サービス)」か「起動(cjcpw)」になっていることを確認してください。

次に右クリックで「停止」を選択してサイトを停止します。

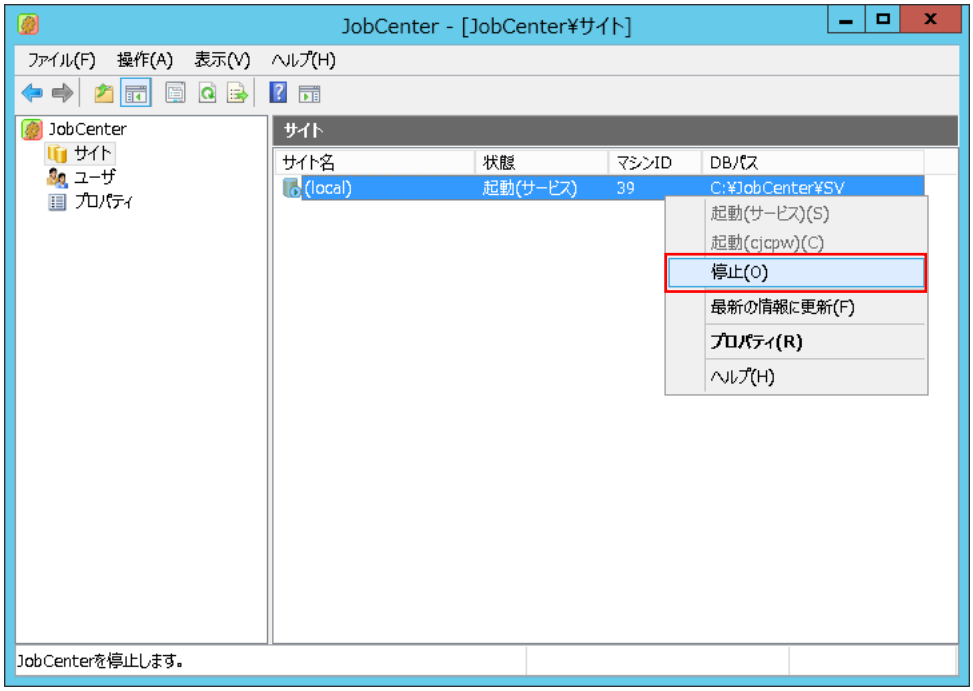

図13.13 サイトの停止画面例

サイトのプロパティを変更すると、サイトの再起動が要求される場合があります。その場合は、上記手順でサイトを停止した後に再起動してください。

13.3.2.1. 起動(サービス)と起動(cjcpw)

サイトの起動方法には、起動(サービス)と起動(cjcpw)の2通りがあります。各起動方法の違いは以下の通りです。

■起動(サービス)

- JobCenterをWindowsのサービスとして起動します。起動方法は、JobCenter環境設定画面の「起動(サービス)」メニュー、cjcpwコマンドの-serviceオプション指定および、Windowsのnetコマンドやscコマンドなどのサービス操作を利用することができます。
- ローカルサイトはインストール時点でサービスとして登録されます。クラスタサイトは登録されていませんが、サービスとして登録・起動することも可能です。なお、クラスタサイトを初めて「起動(サービス)」で起動させようとした場合には、Windowsのサービスへの登録確認画面が表示されます。その画面で「いいえ」を選択した場合には、クラスタサイトの起動自体を中止します。
- JobCenterのトッププロセスはjcservice.exeになり、このプロセスがJobCenterの常駐プロセスの確認を行います。JobCenterのプロセス監視を行う場合は、このjcservice.exeを対象にしてください。
- LSAキューを利用する場合、サイトがサービス起動されている必要があります。LSAキューの詳細については<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。

■起動(cjcpw)

- JobCenterをコマンド(非サービス)として起動します。起動方法は、JobCenter環境設定画面の「起動(cjcpw)」メニュー、cjcpwコマンドを利用することができます。cjcpwについては　<クラスタ機能利用の手引き>、 <コマンドリファレンス>を参照してください。
- cjcpwはコマンドであるため、このサーバの環境設定の画面から起動した場合、サーバの環境設定を起動したデスクトップからユーザがログオフしてしまうと、cjcpwおよびそこから起動されたJobCenterの各プ

ロセスは全て強制的に停止されてしまいます(これはWindowsの仕様です)。したがって、このサーバ環境設定の画面のcjcpwによるサイト起動はテスト等の目的に限定として、通常運用の場合にはクラスタ管理ソフトやプロセス管理ソフトなどの、サービスとして常駐するプログラムからcjcpwコマンドにより起動するようにしてください。

- JobCenterのトッププロセスはcjcpw.exeになり、このプロセスがJobCenterの常駐プロセスの確認を行います。JobCenterのプロセス監視を行う場合は、このcjcpw.exeを対象にしてください。

- 起動(cjcpw)ではLSAキューを使用することはできません。起動(cjcpw)した場合、LSAキューは[DISABLED:STOPPED]状態に変更されます。[DISABLED:STOPPED]に変更されたキューの状態は、自動的に有効状態に変更されないため、手動でキューの状態を設定する必要があります。

13.3.2.2. OSのシャットダウンに関する注意事項

Windowsをシャットダウンする際、JobCenterの終了処理が完了する前に、Windowsから強制的にJobCenterの各プロセスが終了されてしまうことがあります。その結果、JobCenterの正常動作を保証できなくなる可能性があります。

（例）マシン連携している場合、連携先のマシンにTCPコネクションが残ってしまう。

これを回避するには、あらかじめJobCenterサービスを停止してからWindowsをシャットダウンする必要があります。簡単で確実な方法は、WindowsのシャットダウンスクリプトでJobCenterの停止処理を行うことです。グループポリシーエディタ(gpedit.msc)を起動し、以下の操作でJobCenterの停止を行うスクリプトを追加してください。

コンピュータの構成→Windowsの設定→スクリプト

JobCenterの停止スクリプトは、通常のバッチファイルを作成して下記のコマンドを記述してください。

```
net stop JobCenter
```

13.3.3. ジョブの実行設定

「実行設定」タブでは、JobCenterで実行する単位ジョブの実行動作設定を行うことができます。

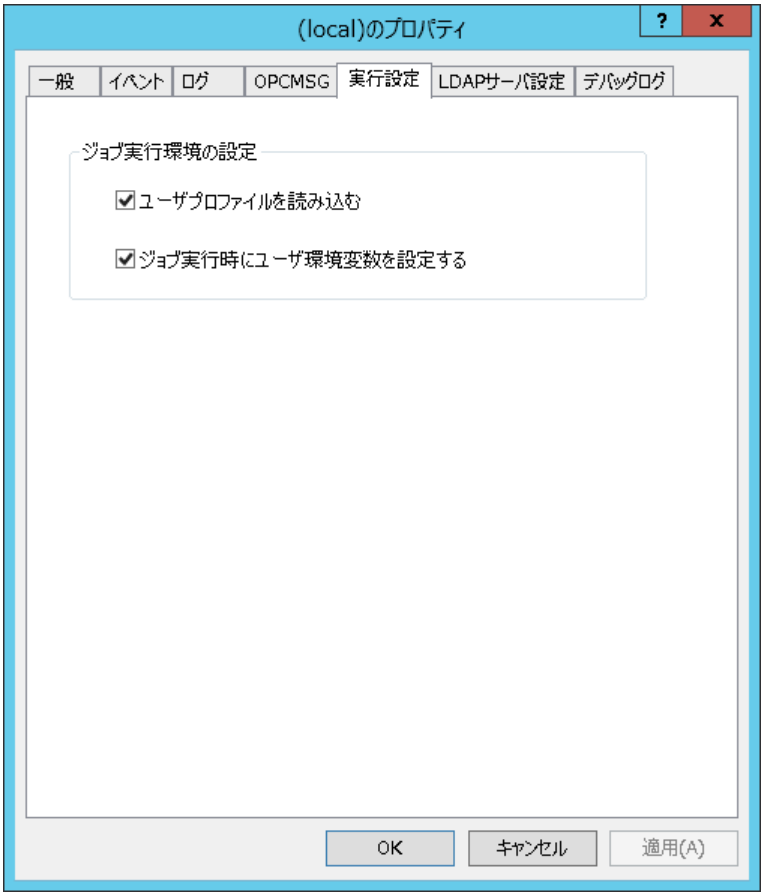

図13.14 実行設定タブの画面例

■ユーザプロファイルを読み込む

単位ジョブ実行時に、ユーザプロファイルを読み込む動作の有無を設定できます。単位ジョブの実行に際して、ユーザプロファイルの読み込みが必要となるコマンドを正常に実行させるためには、本チェックを有効としてください。

ユーザプロファイルの読み込みを行うと、下記のような影響が考えられます。システム環境に合わせて適切に設定するようにしてください。

- 処理性能について

  ジョブ実行毎にユーザプロファイルを読み込む処理を行うため、プロファイルのデータ容量が大きい場合には、処理性能が低下する可能性があります。

- アクセス競合時にジョブ実行に失敗する可能性がある

  ユーザプロファイル読み込み処理がJobCenterプロセスと他プロセスで競合すると、NQSキューが停止する可能性があります。特にウィルススキャンソフトがプロファイルデータをスキャンしていると現象が発生する可能性が高いため、スキャン対象外にする等の設定をシステム側で行う必要があります。

  詳細については、<インストールガイド>の「2.1　インストールの準備をする」を参照してください。

■ジョブ実行時にユーザ環境変数を設定する

単位ジョブを実行した際に、コマンドプロンプトから実行した時と同等の環境変数設定を保持したプロセスでジョブを実行する場合に、本チェックを有効にしてください。

JobCenterのジョブ実行時の環境変数については、「15.2 Windows版JobCenterの環境変数」を参照ください。

「ユーザプロファイルを読み込む」のチェックを外している場合、当該ユーザの「ユーザ環境変数」を一部取得することができません。

「実行設定」はJobCenterサイト単位で設定し、JobCenterグループに所属する全ユーザのジョブ実行時に反映されます。

個別ユーザ単位でジョブの実行設定を行うためには、下記設定ファイルを作成・編集します。

ローカルサイト	%InstallDirectory%\spool\users\%ユーザ名%\jobexe.conf
クラスタサイト	共有ディスク上の<JobCenterDBパス>\spool\users\%ユーザ名%\jobexe.conf

jobexe.conf実行設定ファイルに設定できるパラメータは以下の通りです。

user_profile,1	ユーザプロファイルを読み込む
user_profile,0	ユーザプロファイルを読み込まない
user_env,1	ジョブ実行時にユーザ環境変数を設定する
user_env,0	ジョブ実行時にユーザ環境変数を設定しない

ユーザ個別のjobexe.conf(実行設定ファイル)を作成した場合、「実行設定」タブから設定したJobCenterサイト単位の設定より、優先的に適用されます。

13.4. ユーザの設定

左ペインのツリーから「ユーザ」を選択すると、右ペインに現在JobCenterに登録されているユーザが表示されます。

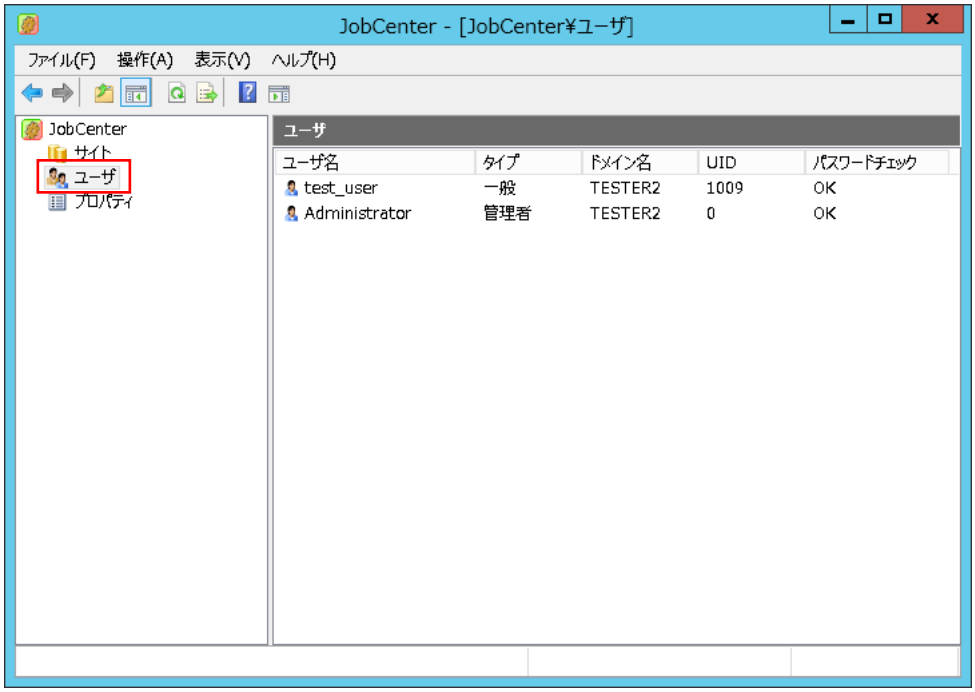

図13.15 ユーザ画面例

Windows版JobCenter(MG/SV)では、ジョブネットワークの作成や実行をするためには事前にユーザ登録が必要です。

図13.15「ユーザ画面例」の右ペインの列の意味は次のとおりです。

表13.3 ユーザの列項目

項　目	意　味
ユーザ名	ユーザ名を表示します。
タイプ	JobCenterのユーザタイプを表示します。 タイプには「管理者」と「一般」があります。「管理者」はインストールされているシステムに1人だけ存在し、システム全般に関する重要な設定は「管理者」ユーザでないと行えないものがあります。 CL/Winでログインする場合、管理者はUMSモードでログインし、一般ユーザはEUIモードでログインします。UMSモードとEUIモードの詳細は <基本操作ガイド>を参照してください。
ドメイン名	ユーザの所属するWindowsドメインを表示します。ローカルユーザの場合はコンピュータ名が表示されます。
UID	JobCenterがユーザを一意に識別するための数値です。 UIDは0～2147483647の数値となります。ただしWindows版の管理者ユーザのUIDはユーザ名に関わらず0固定で扱われますので、0を一般ユーザに対して割り当てることはできません。
パスワードチェック	JobCenterに登録されているユーザのパスワードの状態を表示します。

パスワードチェックの意味は次の表のとおりです。NGまたはNOTSETの場合は、次節で説明するユーザのプロパティで正しいパスワードを設定してください。

表13.4 パスワードチェックの説明

項　目	説　明
OK	正しいパスワードが設定されています。
NG	OSに登録されているパスワードとJobCenterが認識しているパスワードが異なっています。
NOTSET	パスワードが設定されていません。

13.4.1. ユーザのプロパティ

右ペインでプロパティを表示したいユーザをダブルクリックするか、右クリックで「プロパティ」を選択すると、プロパティダイアログが表示されます。下記画面では、「test_user」のプロパティを開いています。

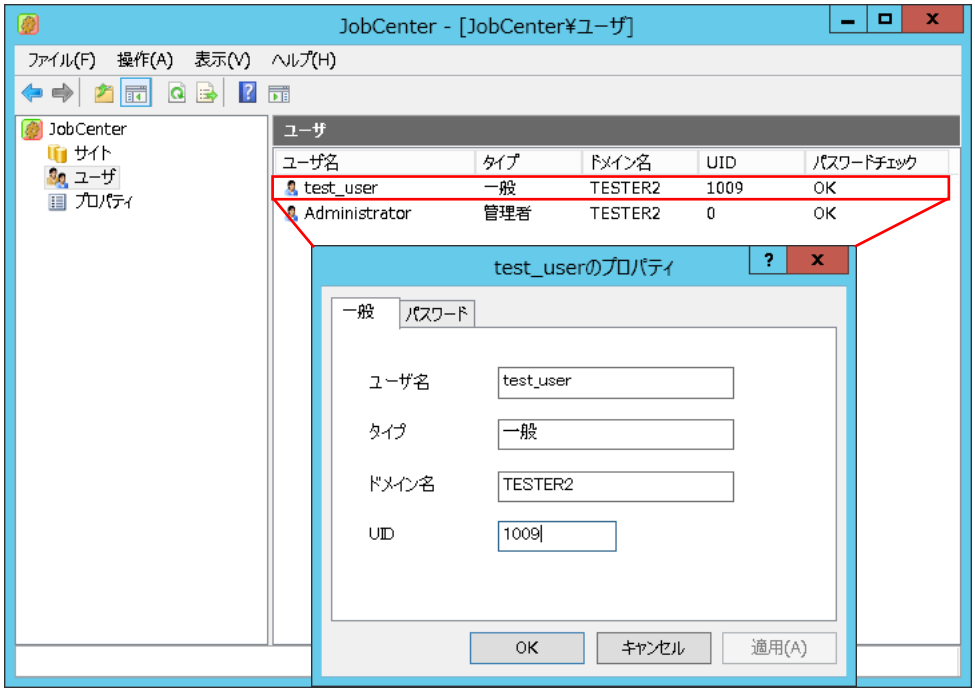

図13.16 ユーザのプロパティ画面例

■一般タブ

一般タブでは、ユーザの右ペインの表示項目と同じ内容が確認できます。

このタブではUID(ユーザID)を変更できます。UIDはJobCenter内部でユーザを一意に認識するための0～2147483647の数値です。JobCenter内部で独自に管理している値であり、変更してもOSやドメインコントローラには影響ありません。

クラスタ構成では、フェイルオーバ後もユーザマッピング設定を一意に有効とするために、ノードマシン間でUIDを統一するよう変更が必要になる場合があります。詳細は<クラスタ機能利用の手引き>の「2.2.1 ユーザIDの統一」を参照してください。

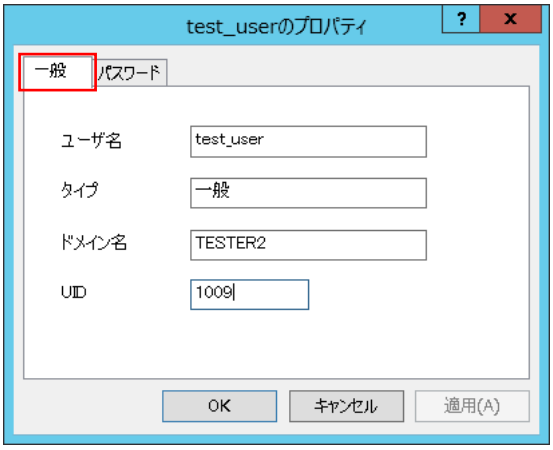

図13.17 一般タブの画面例

UIDの変更を行う場合は、「14.4　ユーザのUIDを変更する」の「14.4.1　事前確認作業」および「14.4.2 ユーザのUID変更作業」に沿って実施してください。

■パスワードタブ

パスワードタブではユーザのパスワードを設定します。

ここで設定するパスワードは、指定されたジョブ実行ユーザの権限でジョブを実行するために、バックグラウンドでOSにログオン/ログオフする際に使用されます。

パスワードチェックに「NG」または「NOTSET」が表示されている場合は、OSに設定してある正しいパスワードを入力して「OK」または「適用」ボタンにより適用してください。

パスワード変更の際にサイトを停止する必要はありませんが、次の場合には変更後に再起動が必要です。

- JobCenter管理者のパスワードを変更した場合。（ファイル待ち合わせ部品でリモートマシンの待ち合わせ対象をUNC表記で指定していない場合は、再起動はでなくqmgrコマンドの「reset　usertoken」サブコマンドの実行でも対応可能です。）
- パスワードを変更したユーザで実行するジョブ(スクリプト)の中に、リモートマシンへアクセスする処理(リモートマシンへのファイルのコピー等)がある場合。

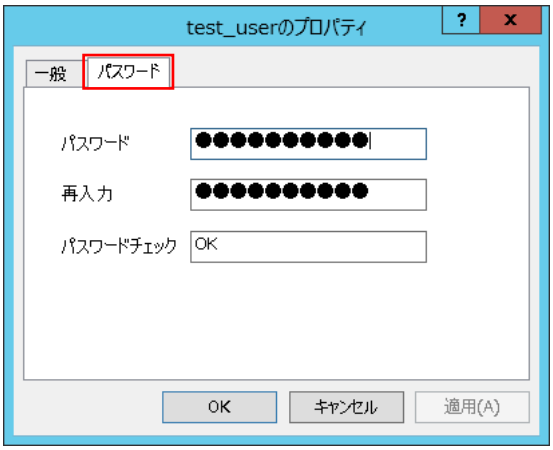

図13.18 パスワードタブの画面例

- 正しいパスワードを設定していないと、ジョブ投入時にこのユーザのジョブ実行が失敗し、キューが停止します。その場合は、上記のプロパティで正しいパスワードを設定して[OK]になったことを確認した後に、CL/WinからJobCenter管理者でMG/SVに接続して、マネージャフレームから対象マシンのマシンフレームを表示し、停止したキューを「開始」にしてください。
- 正しいパスワードを設定しても、JobCenterを再起動しただけでは、キューは自動的に［開始］にはなりませんので注意してください。
- JobCenterに登録されているユーザについて、OSやドメインコントローラ側でそのユーザのパスワードを直接変更した場合は、その直後に必ず上記のパスワードタブでJobCenter側にもパスワードの再設定を行ってください。

13.4.2. ユーザの追加

新しくユーザを追加するには、左ペインのユーザを右クリックし、「ユーザの追加」をクリックします。クリック後、ユーザの追加ダイアログが表示されます。

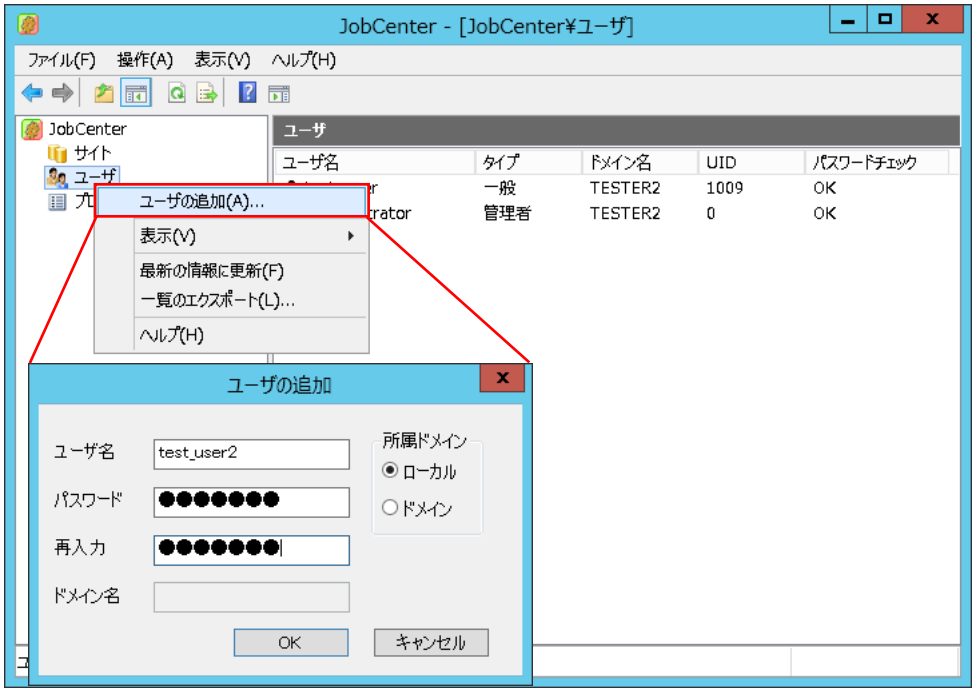

図13.19 ユーザの追加画面例

ユーザの追加ダイアログでは「ユーザ名」「パスワード」「再入力」「所属ドメイン」「ドメイン名」の欄に入力します。ドメインを選択した場合、信頼関係を結んでいる別ドメインのユーザを指定することも可能です。インストールされているマシンがWindowsドメインに所属していない場合は、図13.19「ユーザの追加画面例」のように所属ドメインはローカル固定になり、所属ドメインおよびドメイン名はグレーアウトします。

ローカルまたはドメインに新規にユーザを追加するには、サーバの環境設定を起動したユーザについて、新規ユーザ追加を行うためのOSの権限が必要です。なお、ドメインユーザではなくローカルアカウントをJobCenter管理者としてセットアップしている場合は、［ユーザの追加］でドメインユーザを追加しても正常動作は保証しておりませんので注意してください。

追加しようとしたユーザがすでにOS上に存在していて、入力したパスワードが正しければ、JobCenterにユーザが登録されます。

追加するOS上に存在しているユーザに必要な権限についてはリリースメモの<リリースメモ>の「3.3.5 必要な権限」を参照してください。

OS上に存在しないユーザの場合、以下の警告ダイアログが表示されます。OKを選択するとOSに新規にユーザが作成されたのちにJobCenterに登録されます。キャンセルを選択した場合はユーザの追加は行われません。（なお、[所属ドメイン]で「ドメイン」を選択した場合はドメインコントローラにユーザが追加されます。）

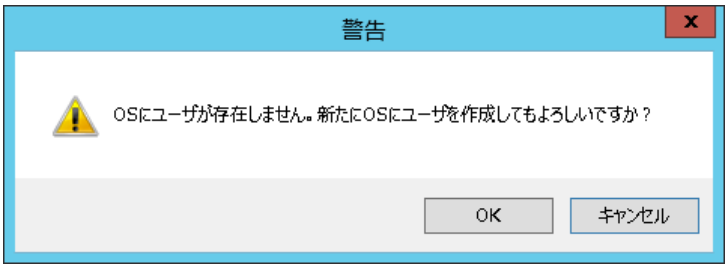

図13.20 ユーザ作成ダイアログ

OS上に存在するユーザの追加はコマンドでも可能です。詳細については、<コマンドリファレンス>の「3.32 jc_usermgr JobCenterのユーザの管理」のaddサブコマンドの説明を参照してください。

13.4.3. ユーザの削除

■本操作は取り消しが効きませんので、十分注意して行ってください。

■事前にサイトを停止しておく必要があります。停止方法は「13.3.2 サイトの起動と停止」を参照してください。

右クリックでポップアップメニューの「削除」を選択して、ユーザの削除ダイアログを表示します。下記画面では「test_user」を削除しようとしています。

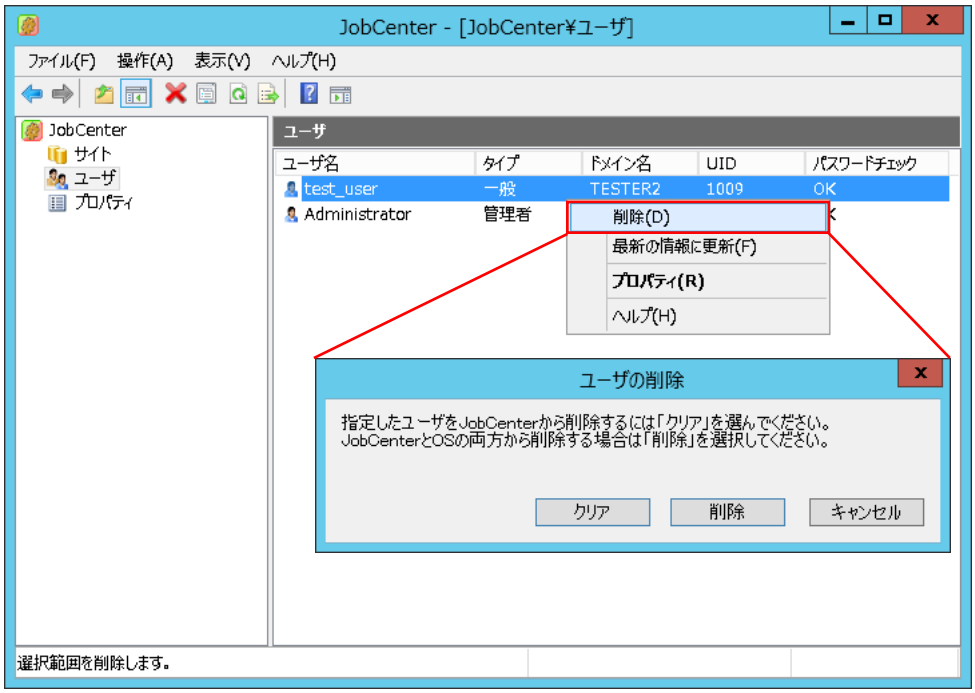

図13.21 ユーザの削除画面例

ダイアログで「クリア」を選択した場合は、JobCenterのユーザ登録からは削除されますが、OSのユーザとしては何も影響はありません。

ダイアログで「削除」を選択した場合は、JobCenterのユーザ登録が削除されるだけでなく、OS上(またはドメインコントローラ)からもユーザが削除されますので注意してください。

13.5. プロパティの設定

左ペインのツリーから「プロパティ」を選択します。右ペインには「総合設定」があります。

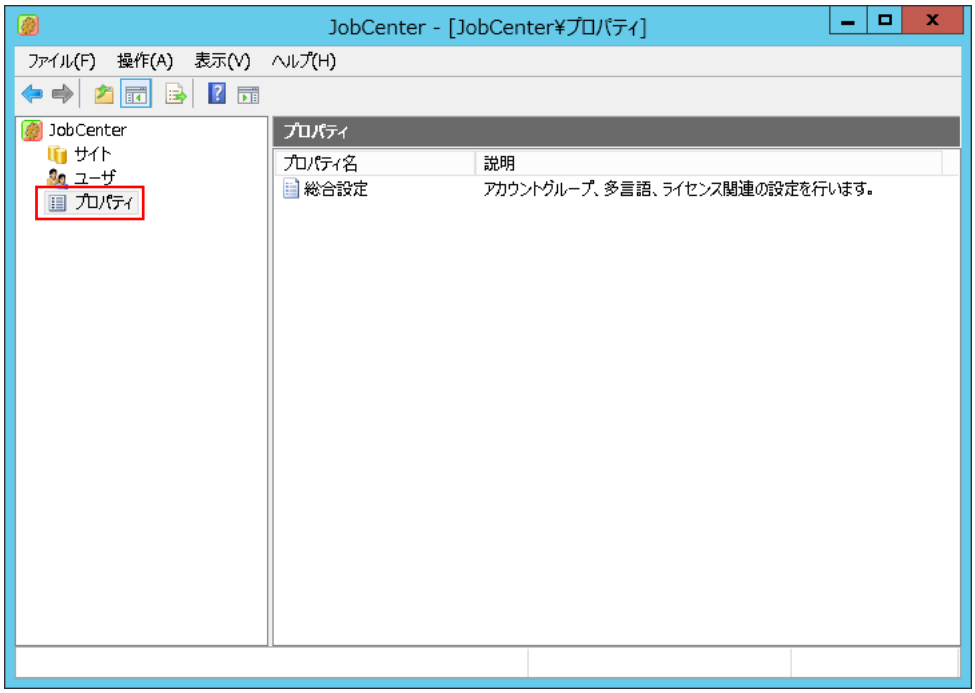

図13.22 総合設定のプロパティ画面例

13.5.1. 総合設定のプロパティ

右ペインで「総合設定」をダブルクリックするか、右クリックで「プロパティ」を選択すると、総合設定のプロパティダイアログが表示されます。

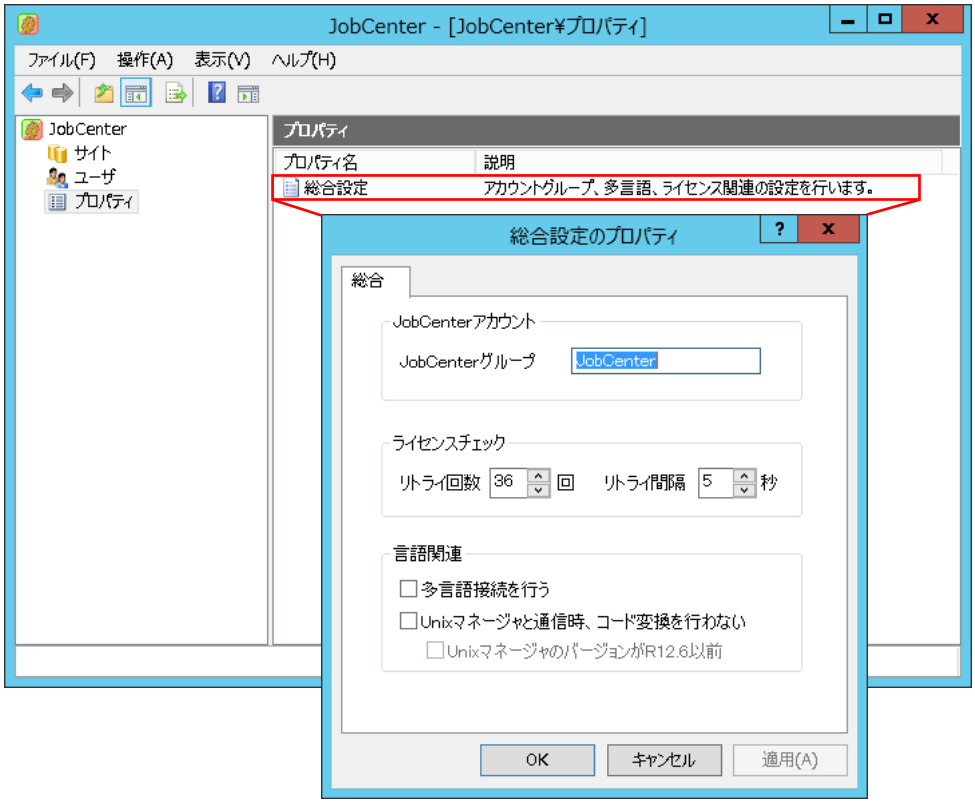

図13.23 総合設定のプロパティ画面例

■「JobCenterグループ」は、インストール時に設定したJobCenter登録ユーザを所属させるグループを変更したい場合に設定します。JobCenterグループを変更する場合は、JobCenterの全てのサイトを停止してから行ってください。

■ライセンスチェックの設定に関しては、「5.5 JobCenterの起動時ライセンスチェックについて」を参照してください。

■言語関連の「多言語接続を行う」の設定に関しては8章 「異なる言語間における接続設定」 を参照してください。

■言語関連の「UNIXマネージャと通信時、コード変換を行わない」の設定に関しては、9章 「日本語環境での文字コード変換」 を参照してください。

14. 環境移行

本章では環境移行の方法と注意点について説明します。

14.1. 環境移行の種類

本章で説明する環境移行の種類としては、以下の3パターンがあります。

1. 異なるマシンへユーザ定義データを移行する

   ジョブネットワーク定義やスケジュール定義、起動トリガ定義、監視対象テキストログ 、稼働日カレンダ定義、カスタムジョブ定義を異なるマシンへ移行する場合です。

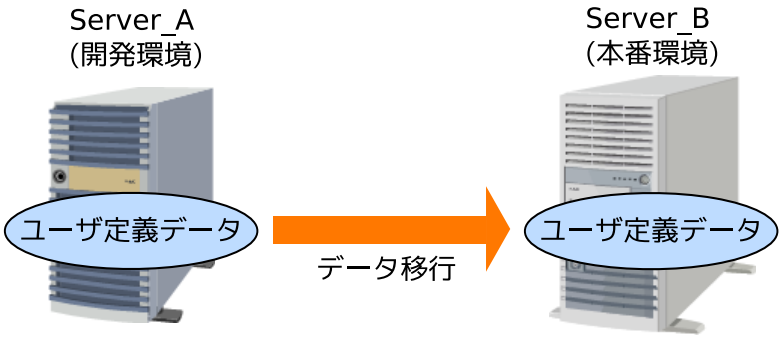

図14.1 異なるマシンへの移行

例えば、開発環境で作成・検証を行ったJNWやスケジュールなどの定義データを、本番環境など異なるマシンへ移行したい場合を想定します。この場合、移行先のホスト名やユーザ名は異なっても構いませんが、ホスト名が異なる場合には定義データ内のホスト名の変更作業が必要になります。

> ⚠ 本セクションではアップロード・ダウンロード機能を利用して環境移行することを説明しています。アップロード・ダウンロード機能が利用できないバージョン(R12.7.x以前)からの移行を行うに場合は、「14.2.4　補足:アップロード・ダウンロード機能がないバージョンからのデータ移行について」を参照してください。

2. ホスト名を変更する

   物理サーバのホスト名が変更になる場合や、クラスタの場合の仮想ホスト名(クラスタサイト名)が変更になる場合です。

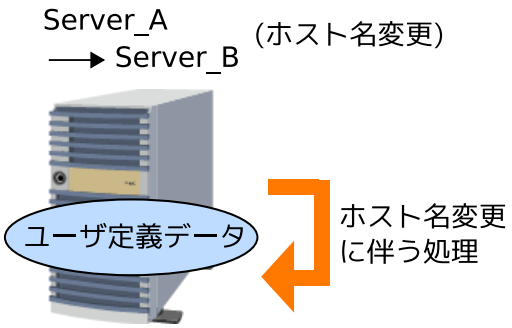

図14.2 ホスト名の変更

例えば、サーバの移設等によりホスト名を変更せざるを得ない場合などを想定します。この場合、直接的なユーザ定義データの移行は発生しませんが、マシン名や場合によってはマシンIDの変更、また、ホスト名変更に伴うユーザ定義データの変更作業が必要になります。

JobCenterではFQDNを意識して動作するため、ネットワークの変更等でドメインのサフィックスのみが変更になる場合でも、本章で説明する作業が必要になります。

3. ユーザのUIDを変更する

物理サーバでJobCenterで使用しているユーザのUIDが変更になる場合や、クラスタの場合のノードマシン間でユーザのUIDを統一するために変更になる場合です。

例えば、クラスタ構成では、フェイルオーバ後もユーザマッピング設定を一意に有効とするために、ノードマシン間でUIDを統一するよう変更が必要になる場合があります。

14.2. 異なるマシンへユーザ定義データを移行する

開発環境で作成・検証を行ったJNWやスケジュールなどの定義データを、本番環境など異なるマシンへ移行したい場合を想定しています。

移行できるユーザ定義データは以下の6種類です。

■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

■稼働日カレンダ定義

■カスタムジョブ定義

■NQS関連の設定やユーザ環境設定、システム環境設定、パーミッション設定などの構成情報については移行できませんので、移行先のマシンにおいて別途設定する必要があります。

■トラッカやトラッカアーカイブデータについては移行できません。

■本章の内容は、移行元のJobCenterがHelper機能をサポートしているバージョンを対象に記載しています。　移行元のJobCenterがHelper機能をサポートしていない場合、インポート・エクスポート機能を使用して定義の移行を行ってください。

- インポート・エクスポート機能を使用する場合、「14.2.1　定義データのダウンロード」、「14.2.3 定義データのアップロード」中の、”ダウンロード”、”アップロード”をそれぞれ”エクスポート”、”インポート”に読み替えてください。
- インポート・エクスポート機能で作成するエクスポートファイルの編集はサポートしておりませんので、「14.2.2.3 ホスト名の手動変更」に相当する操作を行うことはできません。
- ジョブネットワーク定義のパラメータの環境変数の環境変数名、環境変数値、説明に「,」を使用している場合、インポート・エクスポート機能は利用できません。

移行する際、移行先のホスト名やユーザ名は異なっても構いませんが、ホスト名が異なる場合には定義データ内のホスト名の変更作業が必要になります。

以下の図は、異なるマシンへ定義データを移行する場合の作業の流れについて表したものです。

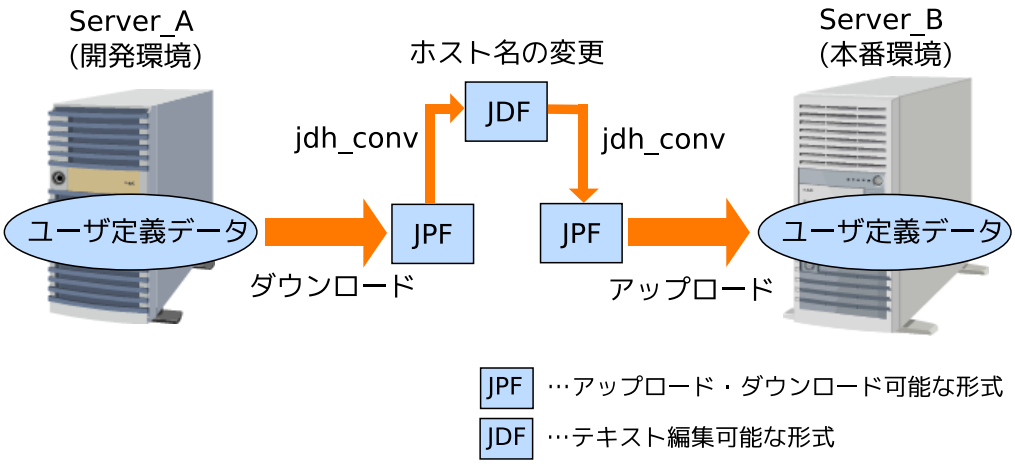

図14.3 異なるマシンへの移行

以降で、実際の移行手順について説明します。

14.2.1. 定義データのダウンロード

移行元のマシンから定義データをダウンロードします。ダウンロードしたデータはJPFファイルという形式で保存されます。

■ダウンロードは定義データを持つユーザ毎に行ってください。管理者ユーザで操作する場合に限り、一括で全ユーザの定義をダウンロードする事が可能です。

■稼働日カレンダ定義とカスタムジョブ定義についてはシステム共通の定義データであり、特にユーザの区別はありませんが、移行先のマシンにアップロードできるのはJobCenter管理者のみとなるため、これら2つの定義データについてはJobCenter管理者ユーザで作業を行うようにしてください。

定義データのダウンロード方法については<基本操作ガイド>の「12.1　定義情報/構成情報/トラッカアーカイブ情報(JPFファイル)ダウンロード」を参照してください。

14.2.2. 定義データ内のホスト名変更（移行先ホスト名が異なる場合）

移行先のホスト名が異なる場合、ダウンロードしたデータの更新が必要になる場合があります。移行先のホスト名が一致している場合には必要ありませんので、本手順はスキップし「14.2.3　定義データのアップロード」へ進んでください。

14.2.2.1. ホスト名の自動変換が可能なパラメータ

以下で説明するパラメータは、JPFファイルのアップロード時にホスト名の変換ルールを指定する事で、自動的にホスト名変換を行うことが可能です。以下のパラメータに使用されているホスト名のみを変更する場合、本手順はスキップし「14.2.3　定義データのアップロード」へ進んでください。その場合、移行先のサーバにアップロードを行う際に、ホスト名の変換ルールを設定して変換を行ってください。

■ジョブネットワークパラメータ - デフォルト投入キュー

■単位ジョブパラメータ - 投入キュー

■カスタムジョブ部品パラメータ - 投入キュー

■WOBSジョブパラメータ - 投入キュー

■イベント送信パラメータ - 送信先ホスト名

■イベント受信パラメータ - 送信元ホスト名

■カスタムジョブ定義パラメータ - 投入キュー

14.2.2.2. ホスト名の手動変換が必要なパラメータ

以下のパラメータは、JPFファイルのアップロード時にホスト名変換が行われないパラメータです。これらのパラメータを設定してる場合には、定義データ内のホスト名の手動更新が必要になる場合があります。各パラメータの説明を確認し、更新を行う場合には「14.2.2.3 ホスト名の手動変更」の手順を実施してください。

■WOBSジョブパラメータ - バッチドメインエージェント:ホスト名

WOBSジョブの接続先バッチドメインエージェイントのホスト名(またはIPアドレス)です。自ホスト名を指定している場合には、移行先の環境で正しく動作するように変更が必要になります。

リモートホストを指定している場合には、必要に応じて変更してください。

■ERPジョブパラメータ - 実行ホスト

実行ホスト名は「<ホスト名>_<SAPシステム名>_<SAPシステム番号>」の形式で指定します。自ホスト名を指定している場合には、移行先の環境で正しく動作するように変更が必要になります。

リモートホストを指定している場合には、必要に応じて変更してください。

> 上記以外にも、ジョブネットワークやスケジュールのコメント、単位ジョブのスクリプトなど、任意の文字列を入力可能なものについてはホスト名が入る可能性がありますが、それらについては記載していません。
>
> ただし、ジョブのスクリプトなど、ホスト名の変更が行われていないと正しく動作しないことがわかっている場合には、以降で説明するデータ更新作業で変更することが可能ですので、必要に応じて変更を行ってください。

JPFはサーバとのやりとりするための形式です。JPFを専用コマンド(jdh_conv)で変換すると、テキスト形式のフォーマット(JDF)が定義データごとに生成されます。それらJDFの中身を更新することで必要部分のホスト名の変更を行います。

> テキスト定義機能の詳細については<テキスト定義機能の利用手引き>にまとめていますので、そちらを参照してください。

14.2.2.3. ホスト名の手動変更

ダウンロードした定義データ内のホスト名を、手動で変更する手順について説明します。

1. JPF形式をJDF形式に変換する

   JPFからJDFへの変換には以下のコマンドを実行します。

   ■UNIX版

   ```
   /usr/lib/nqs/gui/bin/jdh_conv -x [-o $output] [$JPFFile]
   ```

   ■Windows版

   ```
   %InstallDirectory%\bin\jdh_conv -x [-o $output] [$JPFFile]
   ```

   以下にUNIX環境での実行例を記載します。

```
/usr/lib/nqs/gui/bin/jdh_conv -x def_jobmanager01_20170512180000.jpf ↵
Output JDFs to "jpf_20170512183000/jpf".
```

コマンドが成功するとカレントディレクトリにディレクトリが作成され、JDF形式のファイルが展開されます。

■展開先のディレクトリを指定することも可能です。コマンドの詳細および展開したJPFファイル内のディレクトリ構成については<コマンドリファレンス>の「3.21 jdh_conv JDFファイルとJPFファイルの相互変換」を参照してください。

2. ホスト名の変更

1で展開したJDF形式のファイル(xxxx.jdf)はテキスト形式のファイルとなっています。これらについて、任意のテキストエディタやツール等を用いて必要なホスト名の変更作業を行ってください。

3. JDF形式からJPF形式への変換

更新したJDF形式のファイルをJPF形式に圧縮して、サーバとやりとりできる形式に戻します。

JDFからJPFへの変換には以下のコマンドを実行します。

■UNIX版

```
/usr/lib/nqs/gui/bin/jdh_conv [-o $output] [$JDFFile ...]
```

■Windows版

```
%InstallDirectory%\bin\jdh_conv [-o $output] [$JDFFile ...]
```

JDFファイルの更新を行ったかどうかに関わらず、必ず全てのJDFファイルを指定してjdh_convコマンドを実行してください。指定しない場合、その種類のデータは移行先に持っていくJPF内に含まれないことになりますので注意してください。

以下にUNIX環境での実行例を記載します。

```
/usr/lib/nqs/gui/bin/jdh_conv jnw.jdf sched.jdf trg.jdf wkcal.jdf customjob_template.jdf
 operation.jdf ↵
JPF file "jc_def_20170512190000.jpf" conversion successfully.
```

コマンドが成功するとカレントディレクトリにJPFファイルが作成されます。

出力先のファイル名を指定することも可能です。詳細については<コマンドリファレンス>の「3.21 jdh_conv JDFファイルとJPFファイルの相互変換」を参照してください。

オプションのJDFファイル指定にoperation.jdfは必要です。

14.2.3. 定義データのアップロード

移行先のマシンへ定義データ(JPFファイル)をアップロードします。

■アップロードは、移行先マシンで定義データを持たせたいユーザ毎に行ってください。

■最初に必ず、稼働日カレンダ定義、カスタムジョブ定義を含むJPFを、JobCenter管理者ユーザでアップロードしてください。一般ユーザではこれら2つの定義データは更新されません(JPFに含ん

でいても、移行先のサーバにはアップロードはされません)。そのため、これら2つの定義と依存性のあるジョブネットワーク定義やスケジュール定義を一般ユーザで先にアップロードしようとすると、移行先サーバにそのデータが無い状態ではエラーとなります。

なお、稼働日カレンダやカスタムジョブを利用していない場合にはこの限りではありません。

■「14.2.2.1 ホスト名の自動変換が可能なパラメータ」 に一覧されているパラメータのホスト名変換を行う場合、ホスト名の変換ルールを指定してください。

定義データのアップロード方法については<基本操作ガイド>の「12.1　定義情報/構成情報/トラッカアーカイブ情報(JPFファイル)ダウンロード」を参照してください。

14.2.4. 補足:アップロード・ダウンロード機能がないバージョンからのデータ移行について

アップロード・ダウンロード機能が存在しないR12.7.x以前のバージョンから環境移行を行うような場合、本セクションで説明した手順では移行できません。このような場合には、インポート・エクスポート機能を利用してください。R13.1以降でもコマンドラインに限りインポート・エクスポート機能は利用可能です。

ただし、移行前と移行先でプラットフォームが異なったり、JobCenterの文字コード設定が異なったりする場合には、インポート・エクスポート機能では移行できません。例えば、移行前の環境がWindowsで、移行先の環境がLinuxとなるような場合です。このような場合には、移行前の環境からエクスポートしたデータを、アップロード・ダウンロードで扱えるJPF形式に変換します。アップロード・ダウンロード機能の場合、プラットフォームや文字コードの差異を吸収して移行することが可能です。

この変換コマンドの詳細については<コマンドリファレンス>の「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」を参照してください。

変換後は、「14.2.2 定義データ内のホスト名変更（移行先ホスト名が異なる場合）」と「14.2.3 定義データのアップロード」を実施してください。

14.3. ホスト名を変更する

ここではホスト名を変更する際に必要となる作業について説明します。またホスト名の変更だけでなく、それに伴い合わせて発生する可能性のある作業についても説明します。

■以下の手順ではOSのhostnameを「ホスト名」、JobCenterに登録されているローカルサイトやクラスタサイトのサイト名を「マシン名」として区別しています。

■他のマシンのマシン一覧に登録されているマシンや、マシングループに参加しているマシンのホスト名やクラスタサイト名の変更を行う場合は、事前にスケジューラマシンのマシングループとマシン一覧から一旦削除してから変更してください。

■ホスト名を変更する

ホスト名を変更する場合、自マシンのマシンリストの登録内容を変更するだけでなく、連携先の他のMG/SVのマシンリストに登録されている対象マシンを再登録する必要があります。

またユーザ定義データの変換作業が必要になります。

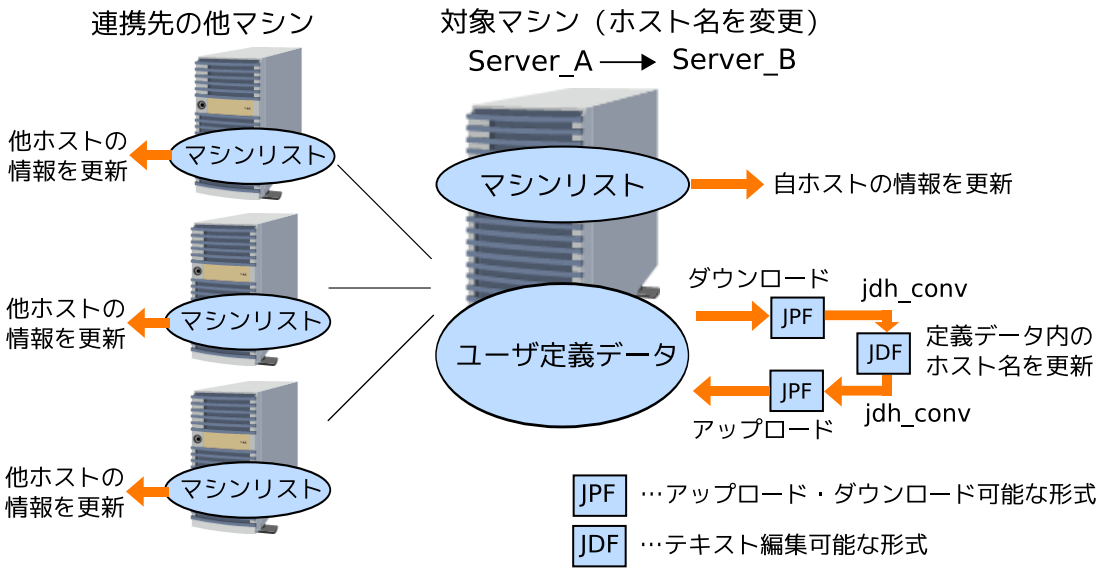

図14.4 ホスト名の変更

■マシンIDを変更する

マシンIDを変更する場合、自マシンのマシンIDを変更するだけでなく、連携先の他のMG/SVのマシンリストに登録されている対象マシンを再登録する必要があります。

また、変更したマシン上のキューを転送先として指定している全てのパイプキューについて、その転送先を再設定する必要があります。

なお、マシンIDのみを変更してホスト名を変更しない場合には、ユーザ定義データの変換は不要です。

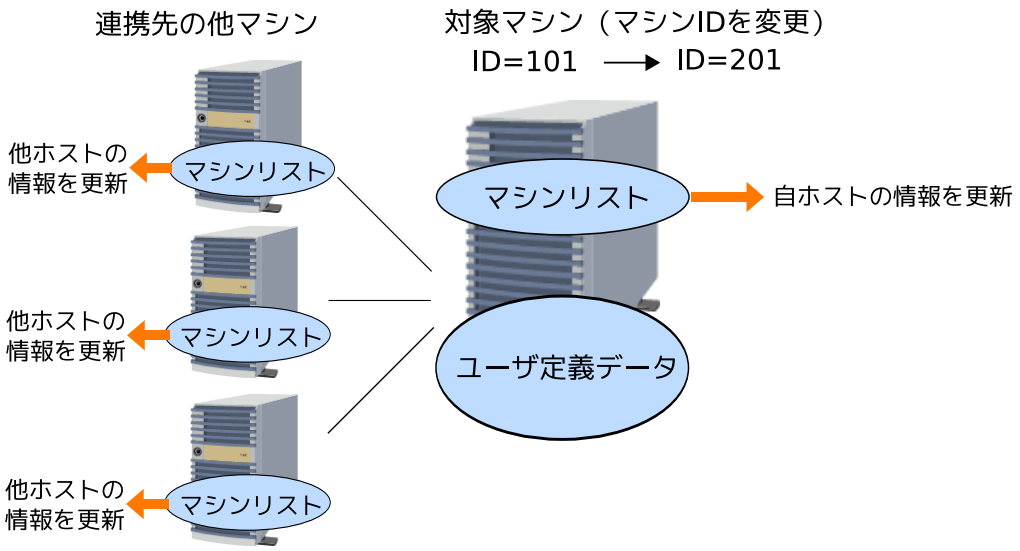

図14.5 マシンIDの変更

■IPアドレスを変更する

IPアドレスのみ変更するときは、JobCenter側で設定を変更する必要はありません。ただし、必ずOSのネットワーク設定で名前解決の正引き、および逆引きができるようにしておいてください。

以下の場合はそれぞれの設定ファイル内のIPアドレス設定を変更内容に合わせて設定しなおしてください。

- daemon.confに利用IPアドレスを明示的に指定(ipaddress=xx.xx.xx.xx)している場合
- JobCenterで名前解決の調整のためresolv.defファイルを設定している場合

以下では、ホスト名を変更する場合とマシンIDを変更する場合のそれぞれについて詳細を説明します。

14.3.1. ホスト名の変更

テスト環境から本番環境へのマシン移設などでホスト名に変更がある場合、以下のようにJobCenterのマシン名設定変更やデータ変換作業を行います。変更作業の前にjc_checkコマンドでシステム情報を採取して、現状の設定状況を必ず保存しておくようにします。

■変更したいマシン名がマシングループに所属している場合は、事前にそのマシングループから削除してから変更作業を行い、変更終了後にマシングループに再登録してください。

■変更するマシンがマシングループのマネージャ(スケジューラ)の場合は、マシングループからメンバマシンを一旦全て削除してから作業を行い、変更終了後にマシングループを再設定してください。

なお、nmapmgrの詳細については<NQS機能利用の手引き>の「6.5.2　リモートマシン定義」<コマンドリファレンス>の「3.12 nmapmgr ネットワークの構成管理」を参照してください。

14.3.1.1. 他ホスト名の登録内容を変更

マシンアイコンに登録されている他ホストのマシン名を自マシン上で変更する場合、以下のように変更を行います。

ホスト名を変更する場合、変更後のホスト名はシステム内で関連する全てのJobCenterインストールマシンで正しく名前解決できている必要があります。

（例）UNIXにrootユーザでログインしたターミナルで、ローカルサイトにマシンID=111で登録されているhost1のマシン名をhost2に変更する

```
# nmapmgr ↵
NMAPMGR>: change name 111 host2 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: exit ↵
#
```

（例）WindowsにJobCenter管理者でログオンして、コマンドプロンプトから、クラスタサイトcluseter2にマシンID=1002で登録されているhost1のマシン名をhost2に変更する

```
> set NQS_SITE=cluster2 ↵
> nmapmgr ↵
NMAPMGR>: change name 1002 host2 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: exit ↵
>
```

クラスタサイトの場合は、必ずnmapmgr起動前に環境変数NQS_SITEにクラスタサイト名を設定します。ローカルサイトの設定変更の場合はNQS_SITEは設定しないでください。

14.3.1.2. 自ホスト名の登録内容を変更

自マシン名を変更する場合、以下のように変更を行います。

ホスト名を変更する場合、変更後のホスト名はシステム内で関連する全てのJobCenterインストールマシンで正しく名前解決できている必要があります。

■自マシン名を変更する場合には、JobCenterを停止した状態で行ってください。

■他ホスト名の変更のみで自ホスト名を変更しない場合は、この作業を行う必要はありません。

■以下の作業は途中で中止することはできません。もし途中で作業を中止した場合、JobCenterが正常に起動できなくなりますのでご注意ください。

■Windowsドメイン参加済みでホスト名を変更する場合や、ホスト名変更とWindowsドメイン参加を併せて行う場合は、以下の手順で作業してください。

1. ホスト名を変更しWindowsドメインから離脱（Windowsドメイン参加済みの場合）／ホスト名を変更（Windowsドメイン参加前の場合）
2. 本手順に沿って自ホスト名の登録内容を変更
3. Windowsドメインに参加

JobCenterでは、localinfoファイルに自マシンのNQSのマシンIDとマシン名を保存しています。

自ホスト名を変更する場合、JobCenterが自マシンを認識するlocalinfoファイルにも変更を反映する必要があります。

localinfoは次のパスにあります。ただしテキストエディタ等で直接編集すると、正常に動作しなくなりますのでご注意ください。

UNIX	/usr/spool/nqs/nmap/localinfo

Windows	%InstallDirectory%\spool\nmap\localinfo

クラスタ環境の場合は /usr/spool や %InstallDirectory% の部分を、cjcmksiteでサイト構築時に指定した <JobCenterDBパス>に読み替えてください。

なお%InstallDirectory%はWindows版JobCenterのインストールディレクトリで、デフォルトはC:\JobCenter\SVになりMます。

自ホスト名を変更し、localinfoファイルに変更を反映する例は以下のとおりです。

（例）UNIXにrootユーザでログインしたターミナルで、マシンID=100で登録されているローカルサイトhost1の自マシン名をhost2に変更する

```
# nmapmgr ↵
NMAPMGR>: change name 100 host2 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: set local name host2 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: exit ↵
#
```

（例）WindowsにJobCenter管理者でログオンして、コマンドプロンプトから、マシンID=1001で登録されているクラスタサイトcluster1.co.jpの自マシン名をcluster2.co.jpに変更する

```
> set NQS_SITE=cluster1.co.jp ↵
> nmapmgr ↵
NMAPMGR>: change name 1001 cluster2.co.jp ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: set local name cluster2.co.jp ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: exit ↵
>
```

クラスタサイトの場合は、必ずnmapmgr起動前に環境変数NQS_SITEにクラスタサイト名を設定します。ローカルサイトの設定変更の場合はNQS_SITEは設定しないでください。

自ホスト名を変更した後はJobCenterを再起動して、CL/Winで変更後のマシン名で接続できることを確認してください。

14.3.1.3. ホスト名変更後の作業

ホスト名を変更してJobCenterのマシン名登録内容を変更した場合、それに関わる他の設定内容も変更する必要があります。次のそれぞれの状況により適切な変更作業を行ってください。

■マシン名を変更したマシンがマシングループに所属していない場合

ホスト名を変更したマシンとジョブリモート投入等で連携する全ての他マシン上のnmapmgrの設定について、上記「14.3.1.1 他ホスト名の登録内容を変更」の作業を行う必要があります。

マシン名を変更したマシン上のキューに対して直接ジョブを投入するよう設定したジョブネットワークが存在する場合、同時にジョブネットワークの各投入先についても、全て再設定する必要があります。

ユーザマッピングも再設定が必要になる場合があります。

■マシン名を変更したマシンがマネージャ管理下のマシン(メンバマシン)である場合

変更対象のマシン名を事前にマシングループから削除してからホスト名変更作業を行うため、上記「ホスト名を変更したマシンがマシングループに所属していない場合」に加えて、マシングループへの新マシン名の再登録が必要になります。

■ホスト名を変更したマシンがマシングループのマネージャ(スケジューラ)の場合

マネージャで管理している全てのメンバマシン上のnsumsmgrユーザのホームディレクトリに.rhostsファイルが設定してある場合は、上記「ホスト名を変更したマシンがマシングループに所属していない場合」「ホスト名を変更したマシンがマネージャ管理下のマシン(メンバマシン)である場合」に加えて、.rhostsに記述されているマシン名を変更後の新マシン名に変更します。

自ホスト名を変更した場合は、次の「14.3.1.4 定義データの更新」の作業も行う必要があります。

14.3.1.4. 定義データの更新

ホスト名を変更した場合、定義データ内に含まれるホスト名の変換が必要になります。これは実際にパラメータとして含まれるホスト名だけではなく、JobCenter内部のDBに格納されているデータも含むため、ジョブネットワークのパラメータなどに明示的にホスト名を含む値(デフォルト投入キューなど)を設定していなくても必要になります(明示的に指定していない場合は、以下の手順1と3のみが必要で、手順2は不要です)。

定義データ内のホスト名の変更方法については、以下を参照して行ってください。なお、参照先では異なるマシンへの移行方法として記載しますが、移行元/移行先のサーバが物理的には同一であり、ホスト名変更している、として読み替えてください。

1. 定義データのダウンロード

   サーバから全定義データをJPF形式でダウンロードします。定義データをもつ全ユーザ分行ってください。管理者ユーザで操作する場合に限り、一括で全ユーザの定義をダウンロードする事が可能です。

2. 定義データ内のホスト名変更

   必要に応じて、定義データ内のホスト名の変更を行います。パラメータ内にホスト名を含む値を設定していなければ変更は不要です。どのようなパラメータが該当するかについては上記リンク先を参照してください。

3. 定義データのアップロード

   JPFをアップロードしてサーバ上の定義データを更新します。全ユーザ分行ってください。変更後のホスト名に対応した形で上書き更新されます。

14.3.2. マシンIDの変更作業

マシンIDを誤って登録してしまった場合や、他のマシンとIDが重複してしまい、マシンIDの割り当てを変更したい場合、以下の手順で変更作業を行います。

マシンIDを変更した場合、IDを変更したマシン上のキューを転送先として設定してある全てのパイプキューについて、CL/Winまたはqmgrのset destinationサブコマンドによる転送先の再設定が必要になります。

マシンIDのみ変更してホスト名の変更は行わない場合は、定義データの更新作業は不要です。

■マシンIDを変更するマシンがマシングループに所属している場合は、事前にそのマシングループから削除しておいてから作業を行い、変更終了後にマシングループに再登録してください。

■マシンIDを変更するマシンがマシングループのマネージャ(スケジューラ)の場合は、マシングループからメンバマシンを一旦全て削除してから作業を行い、変更終了後にマシングループを再設定してください。

なおnmapmgr、qmgrの詳細については<NQS機能利用の手引き>の「6.5.2 リモートマシン定義」や<コマンドリファレンス>の「3.12 nmapmgr ネットワークの構成管理」参照してください。

14.3.2.1. 他マシンIDの登録内容を変更

マシンアイコンに登録されている他ホストのマシンIDを自マシン上で変更する場合、以下のように変更を行います。

（例）UNIXにrootユーザでログインしたターミナルで、ローカルサイトにマシンID=110で登録されている他のマシン名host1のマシンIDを200に変更する

```
# nmapmgr ↵
NMAPMGR>: delete mid 110 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: add mid 200 host1 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: exit ↵
#
```

（例）WindowsにJobCenter管理者でログオンして、コマンドプロンプトから、クラスタサイトcluseter1にマシンID=110で登録されているhost1のマシンIDを200に変更する

```
> set NQS_SITE=cluster1 ↵
> nmapmgr ↵
NMAPMGR>: delete mid 110 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: add mid 200 host1 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: exit ↵
>
```

クラスタサイトの場合は、必ずnmapmgr起動前に環境変数NQS_SITEにクラスタサイト名を設定します。ローカルサイトの設定変更の場合はNQS_SITEは設定しないでください。

14.3.2.2. 自マシンIDの登録内容を変更

マシンアイコンに登録されている自ホストのマシンIDを自マシン上で変更する場合、JobCenterを事前に停止してから以下のように変更を行います。

この場合、localinfoファイルへの変更反映とマシングループの初期化を同時に行う必要があります。また、作業を途中で中止することはできません。もし途中で作業を中止した場合、JobCenterが正常に起動できなくなりますのでご注意ください。

（例）UNIXにrootユーザでログインしたターミナルで、マシンID=101で登録されているローカルサイトhost2.co.jpの自マシンIDを201に変更する

```
# nqsstop ↵
# nmapmgr ↵
NMAPMGR>: delete mid 101 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: add mid 201 host2.co.jp ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: set local name host2.co.jp ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: exit ↵
# nqsstart ↵
```

```
# qmgr ↵
Mgr: set machine_group=(host2.co.jp) ↵
NQS manager[TCML_COMPLETE  ]: Transaction complete at local host.
Mgr: set manager nsumsmgr:m ↵
NQS manager[TCML_COMPLETE  ]: Transaction complete at local host.
Mgr: exit ↵
#
```

（例）WindowsにJobCenter管理者でログオンして、コマンドプロンプトから、マシンID=1001で登録されているクラスタサイトcluster2.co.jpの自マシンIDを2001に変更する

```
> cjcpw -stop cluster2.co.jp ↵
> set NQS_SITE=cluster2.co.jp ↵
> nmapmgr ↵
NMAPMGR>: delete mid 1001 ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: add mid 2001 cluster2.co.jp ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: set local name cluster2.co.jp ↵
NMAP_SUCCESS: Successful completion.
NMAPMGR>: exit ↵
> cjcpw cluster2.co.jp d:\jcdb ↵
> qmgr ↵
Mgr: set machine_group=(cluster2.co.jp) ↵
NQS manager[TCML_COMPLETE  ]: Transaction complete at local host.
Mgr: exit ↵
>
```

クラスタサイトの場合は、必ずnmapmgr起動前に環境変数NQS_SITEにクラスタサイト名を設定します。ローカルサイトの設定変更の場合はNQS_SITEは設定しないでください。

なおqmgrでマシングループを初期化した後、NQSの管理者ユーザを設定するサブコマンド「set manager nsumsmgr:m」はUNIXの場合のみ実行してください。Windowsでは設定不要です。

自ホスト名を変更した後はJobCenterを再起動して、CL/Winで変更後のマシン名で接続できることを確認してください。

14.3.2.3. マシンID変更後の作業

マシンIDを変更した場合、IDを変更したマシン上のキューを転送先として設定してある全てのパイプキューについて、CL/WinまたはqmgrのSet destinationサブコマンドによる転送先の再設定が必要になります。パイプキューの転送先を再設定しないと、正常にジョブリクエストが転送できなくなりますので必ずチェックして再設定を行ってください。

なお、マシンIDの設定状況はnmapmgrのshow stateサブコマンドで参照可能です。

（例）UNIXのローカルサイトhost2のリモート転送用パイプキューremoteqの転送先が、リモートマシンhost1のマシンIDを110から200に変更した結果[旧ID]表示になり「転送先マシン名不明」の状態になったため、再設定する

```
# qmgr ↵
Mgr: show long queue remoteq ↵
remoteq@host2.co.jp;  type=PIPE;  [ENABLED, INACTIVE];  pri=10
  0 depart;  0 route;  0 queued;  0 wait;  0 hold;  0 arrive;
  Run_limit = 1;
  User run_limit : Unlimited
  Group run_limit : Unlimited
```

```
  Cumulative system space time = 0.00 seconds
  Cumulative user space time = 0.00 seconds
  Unrestricted access
  Queue server: C:\JobCenter\SV\bin\NSpipecl.exe
  Destset = {batch1@[110]};
  Reboot mode=RESTART
Mgr: set destination=batch1@host1 remoteq ↵
NQS manager[TCML_COMPLETE  ]: Transaction complete at local host.
Mgr: exit ↵
#
```

（例）Windowsのクラスタサイトcluster1.co.jpのデフォルトキューguilb_defの転送先が、自マシンIDを1001から2001に変更した結果guitp_1@[旧ID]表示になり「転送先マシン名不明」の状態になったため、再設定する

```
> qmgr ↵
Mgr: show long queue guilb_def ↵
guilb_def@cluster1.co.jp; type=PIPE; [ENABLED, INACTIVE]; pri=10
  0 depart;  0 route;  0 queued;  0 wait;  0 hold;  0 arrive;
  Run_limit = 6;
  User run_limit : Unlimited
  Group run_limit : Unlimited
  Reserved_run_limit = 5;
  Destination_retry_wait= 3600;
  Cumulative system space time = 0.00 seconds
  Cumulative user space time = 0.00 seconds
  Unrestricted access
  Load_balance
  Queue server: C:\JobCenter\SV\bin\NSpipecl.exe
  Destset = {guitp_1@[1001]};
  Reboot mode=RESTART
Mgr: set destination=guitp_1@cluster1.co.jp guilb_def ↵
NQS manager[TCML_COMPLETE  ]: Transaction complete at local host.
Mgr: exit ↵
>
```

クラスタサイトの場合は、必ずnmapmgr起動前に環境変数NQS_SITEにクラスタサイト名を設定します。ローカルサイトの設定変更の場合はNQS_SITEは設定しないでください。

また上記の再設定は、CL/WinでJobCenter管理者でログインして、マネージャフレームの各マシンアイコンごとに、キュー一覧上でGUI操作により行うことも可能です。

14.4. ユーザのUIDを変更する

ここでは、ユーザのUIDを変更する際に必要となる作業について説明します。

14.4.1. 事前確認作業

ユーザのUIDの変更を行う場合には、変更を行うマシン上で動作する全てのサイトを停止する必要があります。

UID変更に関する事前確認と合わせ、下記の手順で全てのサイトを停止してください。

■JobCenter SVの場合にはJobCenter MG側のスケジュールの全停止(無効化)

■スケジュールの全停止(無効化)

■「14.4.1.1 ユーザマッピングの確認」

■「14.4.1.2 NQSの管理者リストの確認」

■「14.4.1.3 NQSのメールアカウントの確認」

■「14.4.1.4 キューのキューユーザの確認」

■「14.4.1.5 実行中のリクエストの確認」

■サイトの停止

■スケジュールの無効化の詳細については<コマンドリファレンス>の「3.9 schctrl スケジュールの有効化/無効化」を参照してください。

■クラスタサイトの場合は、必ずschctrlコマンドの実行前に環境変数NQS_SITEにクラスタサイト名(Windows環境の場合にはNQS_SITEDBも)を設定します。ローカルサイトの設定変更の場合はNQS_SITE(Windows環境の場合にはNQS_SITEDBも)は設定しないでください。

■クラスタサイトの場合は、必ずqmgrコマンド及びnmapmgrコマンドの起動前に環境変数NQS_SITEにクラスタサイト名を設定します。ローカルサイトの設定変更の場合はNQS_SITEは設定しないでください。

14.4.1.1. ユーザマッピングの確認

JobCenterのユーザマッピングの設定は、ユーザのUIDを変更する場合には再設定が必要です。

ユーザマッピングの設定の確認を行い、ユーザのUIDを変更するユーザのユーザマッピングの設定のメモをおこなってください。

ユーザマッピングの設定の確認は以下でおこなうことができます。

■CL/Winのマネージャフレームの［マシン一覧］でマシン選択後のマウスの右クリックのポップアップメニューの［ユーザIDのマッピング］

詳細については「3.2.1 CL/Win（ビューア）からユーザマッピングを行う」を参照してください。

■nmapmgrの「SHow State」コマンド

詳細については<コマンドリファレンス>の「3.12　nmapmgr　ネットワークの構成管理」を参照してください。

ユーザマッピングの設定の確認は、転送先のマシン(JobCenter (SV))でも必要です。

14.4.1.2. NQSの管理者リストの確認

NQSの管理者リストの設定は、ユーザのUIDを変更する場合には再設定が必要です。

NQSの管理者リストの設定の確認を行い、NQSの管理者リストの設定のメモをおこなってください。

NQSの管理者リストの設定の確認は、qmgrの「SHOw MANagers」コマンドでおこなえます。

詳細については<コマンドリファレンス>の「3.13 qmgr 構成管理および運用管理」を参照してください。

14.4.1.3. NQSのメールアカウントの確認

NQSのメールアカウントの設定は、ユーザのUIDを変更する場合には再設定が必要です。

NQSのメールアカウントの設定の確認を行い、NQSのメールアカウント設定のメモをおこなってください。

NQSのメールアカウントの設定の確認は、qmgrの「SHOw Parameters」コマンドでおこなえます。

メールアカウントの設定のパラメータは「Mail account」の部分です。

詳細については<コマンドリファレンス>の「3.13 qmgr 構成管理および運用管理」を参照してください。

14.4.1.4. キューのキューユーザの確認

キューのキューユーザの設定は、ユーザのUIDを変更する場合には再設定が必要です。

各キューのキューユーザの設定の確認を行い、ユーザのUIDを変更するユーザが設定されているキューのキューユーザの設定のメモをおこなってください。

各キューのキューユーザの設定の確認は、qmgrの「SHOw LOng Queue」コマンドでおこなえます。

各キューのキューユーザの設定のパラメータは「Users with access」の部分です。

詳細については<コマンドリファレンス>の「3.13 qmgr 構成管理および運用管理」を参照してください。

14.4.1.5. 実行中のリクエストの確認

ユーザのUIDを変更するマシンのサイトの実行中のリクエストが存在しないことを確認してください。

実行中のリクエストの確認は、qmgrの「SHOw Queue」または「SHOw LOng Queue」コマンドでおこなえます。

詳細については<コマンドリファレンス>の「3.13 qmgr 構成管理および運用管理」を参照してください。

また、ユーザのUIDを変更するマシンがJobCenter MGの場合には、転送先のマシン(JobCenter (SV))で、ユーザのUIDを変更するマシンのサイトのリクエストが実行中でないことの確認も必要です。

14.4.2. ユーザのUID変更作業

各サイトの停止後に下記の手順で、ユーザのUIDの変更および全サイトのJobCenterのユーザのUIDが関わる各設定の変更をおこなってください。

■「14.4.2.1 ユーザのUIDの変更」

■各サイトでJobCenterのユーザのUIDが関わる各設定の変更

- 「14.4.2.2 ユーザマッピングの変更」
- サイトの起動
- 「14.4.2.3 NQSの設定変更」

■全ユーザのスケジュールの停止の解除(有効化)

■JobCenter SVの場合にはJobCenter MG側の全ユーザのスケジュールの停止の解除(有効化)

■スケジュールの有効化の詳細については<コマンドリファレンス>の「3.9 schctrl スケジュールの有効化/無効化」を参照してください。

■クラスタサイトの場合は、必ずschctrlコマンドの実行前に環境変数NQS_SITEにクラスタサイト名(Windows環境の場合にはNQS_SITEDBも)を設定します。ローカルサイトの設定変更の場合はNQS_SITE(Windows環境の場合にはNQS_SITEDBも)は設定しないでください。

■クラスタサイトの場合は、必ずqmgrコマンド及びnmapmgrコマンドの起動前に環境変数NQS_SITEにクラスタサイト名を設定します。ローカルサイトの設定変更の場合はNQS_SITEは設定しないでください。

14.4.2.1. ユーザのUIDの変更

ユーザのUIDの変更方法は次のとおりです。

■UNIX環境でのユーザのUIDの変更方法

OSのコマンドで変更可能です。詳細はmanページやOSのマニュアルを参照してください。

■Windows環境でのユーザのUIDの変更方法

JobCenterのサーバ環境設定でユーザのUIDの変更が可能です。詳細は「13.4　ユーザの設定」を参照してください。なお、ユーザのUIDの変更はJobCenterが管理するユーザ情報のみ変更しますので、OSのユーザ情報そのものには何ら変更を加えるものではありません。

14.4.2.2. ユーザマッピングの変更

JobCenterのユーザマッピングの設定を変更後のユーザのUIDに併せてnmapmgrコマンドで変更をおこなってください。

nmapmgrコマンドの詳細については<コマンドリファレンス>の「3.12 nmapmgr ネットワークの構成管理」を参照してください。

■ユーザマッピングの設定の変更は、転送先のマシン(JobCenter (SV))でも必要です。

■ユーザのUIDを変更したマシン側ではJobCenterを停止した状態で実施してください。

14.4.2.3. NQSの設定変更

qmgrコマンドにて以下のNQSの設定でUIDを変更したユーザが設定されていたものについて変更前のUIDの削除、変更後のUIDの追加をおこなってください。

■キューのキューユーザ

- 削除:qmgrサブコマンド DElete Users = $user $queue
- 追加:qmgrサブコマンド ADd Users = $user $queue

■NQSのメールアカウント

- 更新:qmgrサブコマンド SEt MAIL $user

■NQSの管理者リスト

- 削除:qmgrサブコマンド DElete Managers $user:{m,o}
- 追加:qmgrサブコマンド ADd Managers $user:{m,o}

■JobCenter使用不可ユーザ

- 使用不可解除:qmgrサブコマンド ENable User = $user
- 使用不可:qmgrサブコマンド DIsable User = $user

qmgrコマンドの詳細については<コマンドリファレンス>の「3.13 qmgr 構成管理および運用管理」を参照してください。

■JobCenterの起動後に変更をおこなう必要があります。

■NQSの設定の変更が完了するまでUIDを変更したユーザのリクエストが実行されないようにしてください。変更が完了する前に該当のユーザのリクエストが実行された場合、キューのキューユーザまたはJobCenter使用不可ユーザの設定により、リクエストの実行に失敗する場合があります。

14.5. 本番環境での確認作業

実際にジョブネットワークの即時実行やスケジュール投入を行い、問題なく動作することを確認してください。

ホスト名が不正な状態になってしまっている場合、正常に実行されなかったり、スケジュール投入されなかったりする場合があります。設定内容を採取するjc_checkコマンドを利用して、設定内容や整合性に問題がないか確認してください。

jc_checkコマンドの詳細については<コマンドリファレンス>の「6.1 jc_check JobCenterの設定環境を確認」を参照してください。

15. ジョブ実行時の環境変数の取り扱い

JobCenterからジョブを投入および実行した場合、基本的にJobCenterは、投入元のユーザ環境の環境変数をジョブ実行時に引き継ぎません。

これは、ジョブの投入元の環境と実行先の環境でプラットフォームなどの環境が異なる場合、投入元の環境変数をそのまま実行先の環境に適用すると不正な状態に陥ることがあるためです。

また、ジョブの投入先の実行サーバ上のユーザ環境の環境変数については、UNIX版とWindows版で違いがあります。

UNIX版では、基本的に有効となりません。これは、JobCenterがジョブ実行時に所定のシェルを起動する際、シェルを「ログインシェル」として起動しないためです。ただし、実行サーバ上の環境変数のうち、必要最小限のものについては、JobCenterが/etc/passwdなどのシステム情報を元に、JobCenter内部で設定しなおしたうえで再設定しているため、参照可能です。

Windows版では、R12.9以前のバージョンでは有効となりませんが、R12.10以降のバージョンではデフォルトで有効となります。設定を変えることで、UNIX版と同じく無効にできます。

上記のような理由のため、JobCenterからジョブの実行を行う場合、ジョブ実行時に必要な環境変数は、次に記載する方法で別途明示的に設定する必要があります。

15.1. UNIX版JobCenterの環境変数

15.1.1. JobCenter MG側の環境変数

UNIX版のJobCenterからジョブを投入した場合、JobCenter MGは、基本的にジョブの転送元のJobCenter MGの環境変数を転送先のJobCenter SVに引き継ぎません。ただし、次のファイルで設定された環境変数は転送先に引き継がれます。

1. /etc/profile

2. ~/.nsifrc (ジョブネットワーク投入ユーザのhomeディレクトリに設置)

設定例)

```
VAR1=VAL1
export VAR1
```

/etc/profileと.nsifrcに同じ環境変数名で異なる値を設定した場合、~/.nsifrc内の設定が優先されます。

また、JobCenter MG側の基本的な環境変数は、「15.3 ジョブ投入時に独自に設定される環境変数」 のJobCenter固有の環境変数に保存され、ジョブ実行時にexportされます。この環境変数を参照することにより、値を取り出すことが可能です。

例外として、MG側に/etc/profileまたは~/.nsifrcを設置してその中にLANGを設定しても、その値はMGをセットアップした際に指定した言語設定の値で上書きされた上でSV側に転送されますので、注意してください。

MG側でnsumsmgr（JobCenter管理者）により「トラッカ一覧@全マシン」画面等で他のユーザ(root等)の予定トラッカを保留/スキップ等の操作を行うと、予定(確定)トラッカに変更されます。

このときの操作対象ユーザ(root等)が、OSログイン時にLANG環境変数が存在しない環境の場合、LANG環境変数がトラッカ実行時でも認識されず、SVでのジョブ実行時にLANG環境変数が不正になり正常に動作しない場合があります。

それを回避するには操作対象ユーザについて~/.nsifrcを設置して、その中で export LANG=xxxx を設定してください。

JobCenterはバッチキューのジョブ実行時に内部でユーザ実行権限を切り替えます。そのためOSの仕様によっては.nsifrcに環境変数を設定してもSV側に引き継がれず、NULLになる場合があります。(例として、AIXでLIBPATHが引き継がれない等) .nsifrcに環境変数を設定しても引き継がれない場合は、次のいずれかの方法で対処してください。

■環境変数を/etc/profileや.nsifrcに定義せずに、単位ジョブスクリプトの先頭で明示的に定義する。

■いったん目的の環境変数以外の任意の環境変数に必要な値を設定し、単位ジョブスクリプトまたはそこから実行されるユーザコマンド・シェル内で実行時にその値を目的の環境変数に設定し直す。

15.1.2. JobCenter SV側の環境変数

UNIX版のJobCenter SVへジョブを投入した場合、JobCenter SVはジョブの実行時、実行先環境でジョブ実行のためのシェルを起動します。しかし、そのシェルは「ログインシェル」として起動されないため、".login",".profile"などの設定ファイルを読み込みません。従って、それらの設定ファイルに記述された環境変数はJobCenterで実行されたジョブから参照できません。

ジョブ実行時に必要な環境変数は、以降の説明に従って、別途設定しなおす必要があります。

ただしcsh系の場合には".cshrc"の内容が有効になります。これはcshが起動時に".cshrc"を読み込むというcshの仕様のためです。

15.1.2.1. ジョブ実行時に必要な環境変数

ジョブの転送先のJobCenter SV側の環境変数のうち、次に挙げるものについては、ジョブの実行時にJobCenter SV側のシステム環境に合わせた値に設定しなおしたうえでexportします。

1. HOME
2. LOGNAME
3. MAIL
4. SHELL
5. TZ
6. USER
7. NQS_SITE
8. QUEUENAME

15.1.2.2. 環境変数「ENVIRONMENT」の設定値

ジョブ実行時に、環境変数「ENVIRONMENT」の値を固定値に設定した上でexportします。

一般ユーザのとき	BATCH
ルートユーザのとき	BATCH

15.1.2.3. 環境変数「PATH」の設定値

ジョブ実行時、環境変数「PATH」の値をそれぞれ次の固定値に設定した上でexportします。

■HP-UX

一般ユーザのとき	/bin:/usr/bin
ルートユーザのとき	/bin:/etc:/usr/bin

■Linux

一般ユーザのとき	/bin:/usr/bin
ルートユーザのとき	/bin:/etc:/usr/bin

■AIX

一般ユーザのとき	/bin:/usr/bin
ルートユーザのとき	/bin:/etc:/usr/bin

15.1.3. 環境変数の設定方法

JobCenterから起動されるジョブに環境変数を設定する場合、以下の「15.1.3.1 単位ジョブスクリプト内での対処」から「15.1.3.6 JobCenter SV側で設定する場合の対処（環境変数「PATH」のみ）」のいずれかの方法で設定してください。

なお、各設定方法の優先順位は次の通りです。(ジョブネットワークパラメータの「環境変数」設定を含む)

設定方法	優先順位
「15.1.3.1 単位ジョブスクリプト内での対処」	高
「15.1.3.2 環境変数継承用の単位ジョブでの対処」	↑
「15.1.3.3 ジョブネットワークの即時投入時の対処」	|
「15.1.3.4 ジョブネットワークパラメータの「環境変数」タブでの対処」	|
「15.1.3.5 JobCenter MG側で設定する場合の対処」	↓
「15.1.3.6 JobCenter SV側で設定する場合の対処（環境変数「PATH」のみ）」	低

環境変数の設定方法に関する注意事項

■UNIX系とWindows系のSVが混在している場合は、「15.2　Windows版JobCenterの環境変数」も併せて参照してください。

■環境変数「NQS_PATH_UNIX」「NQS_PATH_WIN」に関する機能は、ジョブ転送先のSVがR12.5.5以降の場合のみ有効となります。

■「NQSDAEMON_EXPORT」に関する機能は、SVがR12.6以降の場合のみ有効となります。

■環境変数に設定する値として、複数行となる値はサポート対象外となります。

■以下の注意事項はバージョンによって異なります。

R12.8.2まで

- 1リクエストあたり使用可能な環境変数の最大数は400個です。制限数を超えて投入を行った場合、リクエストはエラーとなります。
- Windows版SVに投入されるジョブリクエストについては、投入側であらかじめ1環境変数あたり環境変数名(=も含む)、値あわせて8189byte以内にしておく必要があります。

R12.8.3以降

- ジョブ起動時に渡すことのできる環境変数の合計サイズはOS,PF毎のプロセス生成時に指定可能な環境変数サイズの制限(execveシステムコール実行時の環境変数(envp)と引数リスト(argv)の合計サイズでの制限)によります。
- 各OS毎の環境変数サイズ目安

HP-UX	2MByte
Linux	128KByte
AIX	24KByte(デフォルト)

※上記値は各OSのバージョン毎に異なります。利用されるシステムの詳細な環境変数サイズについては、OSの仕様を確認してください。

- HP-UX版でかつjnwengineのオプション:-A を使用した場合のみ、以下の様なJobCenter独自の環境変数の合計サイズ、個数の制限が発生します。

環境変数の最大個数	2400
環境変数の最大合計サイズ	120KByte

15.1.3.1. 単位ジョブスクリプト内での対処

■単位ジョブのスクリプトで、ジョブの実行に必要な環境変数を直接設定してexportします。

■単位ジョブのスクリプトの先頭に#!/bin/shを記述して、ジョブの実行に必要な環境変数を設定するためのスクリプトをSV側に別途用意します。単位ジョブスクリプトで用意したスクリプトを.(dot)コマンドでインクルードします。

■単位ジョブのスクリプトの先頭に#!/bin/cshを記述して、ジョブの実行に必要な環境変数をSV側の.cshrcに記述します。

なお設定した環境変数の有効範囲は、設定した単位ジョブスクリプト内のみとなります。

15.1.3.2. 環境変数継承用の単位ジョブでの対処

■単位ジョブのEXPORTVARの機能を使用して、任意の環境変数を設定できます。この機能を使用するには、作成した単位ジョブパラメータ設定画面の結果タブにある「変数継承:STDOUT」の設定が必要となります。

詳細な設定方法や注意事項・制限については<基本操作ガイド>の「4.2.3　単位ジョブ間で情報を引き継ぐ」を参照してください。

■Windows系MGの場合やUNIX系SVとWindows系のSVが混在した環境の場合等、環境変数[PATH]がそのまま継承できないときは、環境変数「NQS_PATH_UNIX」にPATHの値を設定することで、ジョブ実行時にSVの環境変数「PATH」へ継承できます。

単位ジョブのスクリプトに環境変数「NQS_PATH_UNIX」を定義する例

```
echo EXPORTVAR
echo NQS_PATH_UNIX=/usr/local/bin:/bin:/usr/bin:/home/jobcenter/bin
echo EXPORTVAR
```

作成した単位ジョブには変数継承以外の記述をしないでください。

変数継承の設定と業務処理を単位ジョブ内に混在すると、業務処理でエラーが発生した場合に後続の単位ジョブに環境変数が継承されなくなります。

15.1.3.3. ジョブネットワークの即時投入時の対処

ジョブネットワークを即時投入する際、「複数の環境変数を設定する」を選択することで、投入するジョブネットワークの環境変数を設定できます。

詳細な設定方法や注意事項・制限については<基本操作ガイド>の「3.2.1　即時投入する」を参照してください。

15.1.3.4. ジョブネットワークパラメータの「環境変数」タブでの対処

ジョブネットワークパラメータの「環境変数」タブで、単位ジョブ実行時に付与される環境変数を設定できます。

詳細な設定方法や注意事項・制限については<基本操作ガイド>の「3.3.4.4 環境変数」を参照してください。

またサブジョブネットワークについては、サブジョブネットワークパラメータの「環境変数」タブで、単位ジョブ実行時に付与される環境変数を設定できます。

詳細な設定方法や注意事項・制限については<基本操作ガイド>の「4.2.10.1 [環境変数]タブ」を参照してください。

15.1.3.5. JobCenter MG側で設定する場合の対処

■UNIX系MG － UNIX系SVの場合

JobCenter　MG側がUNIX版である場合には、次のいずれかのシェルスクリプト内で任意の環境変数を設定し、exportすることで、サーバ単位、またはユーザ単位でジョブ実行時の環境変数を設定できます。ただし、環境変数「PATH」については、環境変数「NQS_PATH_UNIX」の設定が必要となります。

1. /etc/profile

2. ~/.nsifrc (ジョブネットワーク投入ユーザのhomeディレクトリに設置)

設定例)

```
VAR1=VAL1
export VAR1
```

/etc/profileと.nsifrcに同じ環境変数名で異なる値を設定した場合、~/.nsifrc内の設定が優先されます。

~/.nsifrcを設定した場合、スケジュール投入による運用を行っているユーザで必ずスケジュール定義の変更操作を行い、設定した環境変数をJobCenterに反映させる必要があります。

スケジュール変更の必要がないときは、一時的に仮のスケジュールの作成、保存、削除の操作を行ってください。

なお、他ユーザのジョブネットワークの実行を行う場合や、他ユーザのスケジュールを設定する場合は、そのジョブネットワーク所有者に設定された環境変数が使用されます。

/etc/profileや~/.nsifrcを設定したとき、ジョブネットワークの実行時の環境変数はログインユーザ自身のスケジュールの変更操作においてのみ設定されますので、必ずジョブネットワークを所有するユーザでログインし直して、前述のスケジュール変更操作を行ってください。

> すでに予定［確定］状態で表示されているトラッカ上については、上記の操作を行っても~/.nsifrcの設定内容が反映されませんので、［確定］状態のトラッカを一旦削除する必要があります。

■Windows系MG － UNIX系SVの場合

MG側がWindows系の場合には、MG側のシステム環境変数に値を定義をしておくことで、ジョブ実行時の環境変数へ設定できます。

ただし環境変数「PATH」についてはそのまま継承できませんので、環境変数「NQS_PATH_UNIX」への設定が別途必要となります。

> システム環境変数を変更した場合はJobCenterの再起動が必要です。OSによってはOSの再起動が必要になる場合もあります。

15.1.3.6. JobCenter SV側で設定する場合の対処（環境変数「PATH」のみ）

■JobCenter MG側・SV側共にUNIX版である場合には、デーモン設定ファイル（daemon.conf）にNQSDAEMON_PATH_EXPORT=1を指定することにより、ジョブ実行時のJobCenter MG側の環境変数「PATH」を引き継げます。

■指定方法については「5.2.3 UNIXの場合」を参照してください。

15.1.4. MGとSVとのLANGが異なる場合の注意事項

MGからSVにジョブを転送する場合に環境変数LANGはMG側からSV側にそのままの値が 転送されます。そのため、MGとSVとの言語環境が違う場合か、同一の言語でもLANGの設定値(設定文字列)が異なる場合には設定が必要です。設定方法は「9.2.1.2 UNIX版同士のUNICODE混在環境JobCenter SVで文字コード変換を設定する」を参照してください。

また、ジョブ個別に設定したい場合は次の設定が必要です。

転送するジョブのスクリプトの最初に、SV側で動作させたいLANGの設定を明示的に記述してください。

スクリプト記述例(SV側のLANGが"ja_JP.SJIS"の場合)

```
LANG=ja_JP.SJIS
 export LANG
 #
 command1
 command2
 …
```

上記の設定を行っても、単位ジョブを転送実行すると標準エラー出力結果に以下のようなワーニングが記録される場合があります。

(例) MGがAIX SJIS (Ja_JP)、SVがHP-UX SJIS (ja_JP.SJIS) の場合

```
Warning! One or more of your selected locales are not available.
Please invoke the commands "locale" and "locale -a" to verify your
selections and the available locales.

Continuing processing using the "C" locale.
```

MG側に/etc/profileまたは~/.nsifrcを設置してその中にLANGを設定しても、その値はMGをセットアップした際に指定した言語設定の値で上書きされた上でSV側に転送されますので、注意してください。

15.1.5. 環境変数TZに関する注意事項(Linux、AIX版)

■Linux版JobCenterでの注意事項

環境変数「TZ」が標準で設定されないLinuxマシンでJobCenterを利用する場合、単位ジョブの環境変数「TZ」には「JST-9JDT(またはJST-9JST等、OS側の環境に依存)」が設定されます。これによって、単位ジョブのタイムゾーンが通常「JST-9」と異なるものとなり、dateコマンド実行時などに表示される時刻が1時間ずれる場合があります。この場合は、環境変数「TZ」を設定した環境でJobCenterを使用してください。

- 環境変数「TZ」を設定してJobCenterを再起動する場合の対処方法

1. JobCenterを停止する。

```
# /usr/lib/nqs/nqsstop ↵                          (ローカルサイトの例)
# /usr/lib/nqs/cluster/cjcpw -stop cluster.co.jp ↵  (クラスタサイトの例)
```

2. nqs起動ファイル /usr/lib/nqs/rc/nqs.sh を編集する。

「/usr/lib/nqs/nqsdaemon $NQSDAEMON_OPT >/dev/null 2>&1」の直前に、TZを設定する2行を追記する。

```
case $1 in
'start')
     if [ -x /usr/lib/nqs/nqsdaemon ]
     then
          TZ=JST-9
          export TZ
          /usr/lib/nqs/nqsdaemon $NQSDAEMON_OPT >/dev/null 2>&1
          status=$?
```

3. jnwengine起動ファイル /usr/lib/nqs/rc/jnwengine.sh を編集する。

   「/usr/lib/nqs/gui/bin/jnwengine」の直前に、次の2行を記述する。

```
case $1 in
'start')

    if [ -x /usr/lib/nqs/gui/bin/jnwengine ]
    then
        TZ=JST-9
        export TZ
        /usr/lib/nqs/gui/bin/jnwengine $JNWENGINE_OPT >/dev/null
        status=$?
```

4. JobCenterを起動する。

```
# /usr/lib/nqs/nqsstart                              (ローカルサイトの例)
# /usr/lib/nqs/cluster/cjcpw cluster.co.jp /jcdb    (クラスタサイトの例)
```

5. /etc/profile、または~/.nsifrcに上記 2. 3. と同様に環境変数TZの設定を行う。

■Linux、AIX版での注意事項

Linux、AIX版JobCenterでは、ユーザフレームの「スケジュール表示」タブを参照する際に、カレンダ分岐部品によって日付の変わり目がGMTで判断されて本来分岐しないはずのフローに分岐しているように見えてしまい、正常に表示できない場合があります。(ただし実際のカレンダ分岐部品の日付判定動作には問題ありません)

正しく表示するためには、/etc/profile　または　~/.nsifrcファイルに上記「■Linux版JobCenterでの注意事項」のTZ環境変数設定を追加してください。(JobCenterの再起動は不要です) ~/.nsifrcファイルについては「15.1.3 環境変数の設定方法」を参照してください。

15.2. Windows版JobCenterの環境変数

15.2.1. JobCenter MG側の環境変数

JobCenter MGからジョブを投入した場合、投入元JobCenter MGの環境変数のうち、次に挙げるものは実行先でのジョブ実行時に引き継がれます。

■「システム環境変数」に設定された環境変数

逆に、次に挙げる環境変数は引き継がれないため、参照できません。

■「ユーザ環境変数」に設定された環境変数

■JobCenter MGのenvvarsファイルに設定された環境変数

ただし、JobCenter MG側の環境変数のうち、ジョブ実行時にJobCenter SV側で再設定される環境変数（「15.2.2 JobCenter SV側の環境変数」参照）については、JobCenter MG側での設定値をジョブ実行時に参照できません。なお、MG側の基本的な環境変数の値は、UNIX版同様JobCenter固有の環境変数により参照可能です。「15.3 ジョブ投入時に独自に設定される環境変数」を参照してください。

> システム環境変数を変更した場合はJobCenterの再起動が必要です。OSによってはOSの再起動が必要になる場合もあります。

> Windows版では環境変数の引き継ぎに関して制限があります。また、UNIX版ではバージョンにより引き継ぎ個数に制限があります。詳細については「15.2.3 環境変数の設定方法」の注意事項を参照してください。

15.2.2. JobCenter SV側の環境変数

R12.9以前のバージョンではジョブ実行時にジョブ実行ユーザが保持する環境変数を用いませんでしたが、R12.10以降のバージョンではデフォルトでジョブ実行ユーザが保持する環境変数を用いてジョブ実行する動作に変更しています。

> 「サーバの環境設定」から「実行設定」タブの「ジョブ実行時にユーザ環境変数を設定する」のチェックを外すことで、R12.9以前と同様の動作にすることができます。 「13.3.3 ジョブの実行設定」 を参照ください。

それぞれの場合において、環境変数を扱う上での注意点を次項以降で説明します。

15.2.2.1. ジョブ実行時に必要な環境変数

ジョブの転送先のJobCenter SV側の環境変数のうち、次に挙げるものについては、下表の注意が必要です。

1. ComSpec
2. NetShepherdUserName
3. NQS_SITE
4. Os2LibPath
5. STANDARD_ERROR
6. STANDARD_OUTPUT
7. SystemRoot
8. SystemDrive

9. USERNAME
10. USERPROFILE
11. windir

対象バージョンと条件	注意点
R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを入れた場合(デフォルト)	ジョブ実行ユーザが保持する環境変数でcmd.exeを起動します。従ってジョブ実行ユーザの当該環境変数によって、これらの環境変数を不用意に上書きしないようにする必要があります。
R12.9以前のバージョン R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを外した場合	JobCenterの設定や、SV側のシステム設定に応じた値に再設定された環境変数でcmd.exeを起動します。 従ってジョブ実行時にコマンド固有の値が要求される環境変数については「15.2.3 環境変数の設定方法」により別途設定し直す必要があります。

15.2.2.2. 環境変数「ENVIRONMENT」の設定値

JobCenterは、ジョブ実行時、環境変数「ENVIRONMENT」に次の固定値を設定します。

```
BATCH
```

したがって、下表の注意点があります。

対象バージョンと条件	注意点
R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを入れた場合(デフォルト)	ジョブ実行ユーザの環境変数「ENVIRONMENT」によって、上書きしないようにする必要があります。
R12.9以前のバージョン R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを外した場合	強制的に上記の値が設定されるため、任意の値を設定することはできません。

15.2.2.3. 環境変数「PATH」の設定値

JobCenterは、ジョブ実行時、環境変数「PATH」に次の値を設定します。

対象バージョンと条件	設定値
R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを入れた場合(デフォルト)	ジョブ実行ユーザの環境変数「PATH」の値
R12.9以前のバージョン R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを外した場合	Windowsシステムディレクトリ;Windowsディレクトリ

15.2.2.4. 環境変数「TMP」、「TEMP」の設定値

JobCenterは、ジョブ実行時、環境変数「TMP」に次の値を設定します。

対象バージョンと条件	設定値
R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを入れた場合(デフォルト)	ジョブ実行ユーザの環境変数「TMP」の値
R12.9以前のバージョン	レジストリキー"HKEY_USERS\.DEFAULT\Environment"の、値の名前"TMP"に設定されている値

対象バージョンと条件	設定値
R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを外した場合	

JobCenterは、ジョブ実行時、環境変数「TEMP」に次の値を設定します。

対象バージョンと条件	設定値
R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを入れた場合(デフォルト)	ジョブ実行ユーザの環境変数「TEMP」の値
R12.9以前のバージョン R12.10以降のバージョンで「ジョブ実行時にユーザ環境変数を設定する」のチェックを外した場合	レジストリキー"HKEY_USERS\.DEFAULT\Environment"の、値の名前"TEMP"に設定されている値

上記の各レジストリはOSの種類により異なります。

実際にどのような固定値に展開されるかについては、ジョブスクリプトの先頭に set コマンドを記述して、投入実行して確認してください。

15.2.3. 環境変数の設定方法

JobCenterから起動されるジョブに環境変数を設定する場合、以下の「15.2.3.1 単位ジョブスクリプト内での対処」～「15.2.3.8 JobCenter MG側で設定する場合の対処（環境変数「PATH」以外）」のいずれかの方法で設定してください。

なお、各設定方法の優先順位は次の通りです。(ジョブネットワークパラメータの「環境変数」設定を含む)

設定方法	優先順位
「15.2.3.1 単位ジョブスクリプト内での対処」	高
「13.3.3 ジョブの実行設定」	↑
「15.2.3.2 JobCenter SV側で設定する場合の対処（envvarsファイル）」	|
「15.2.3.3 環境変数継承用の単位ジョブでの対処」	|
「15.2.3.4 ジョブネットワークの即時投入時の対処」	|
「15.2.3.5 ジョブネットワークパラメータの「環境変数」タブでの対処」	|
「15.2.3.6 JobCenter MG側で設定する場合の対処（環境変数「PATH」のみ）」	|
「15.2.3.7 JobCenter SV側で設定する場合の対処（環境変数「PATH」のみ）」	↓
「15.2.3.8 JobCenter MG側で設定する場合の対処（環境変数「PATH」以外）」	低

■UNIX系とWindows系のSVが混在している場合は、「15.1 UNIX版JobCenterの環境変数」も併せて参照してください。

■システム環境変数を変更した場合は、JobCenterの再起動が必要です。OSによっては、OSの再起動が必要になる場合もあります。

■環境変数「NQS_PATH_UNIX」,「NQS_PATH_WIN」に関する機能は、ジョブ転送先のSVがR12.5.5以降の場合のみ有効となります。

■「NQSDAEMON_EXPORT」に関する機能は、SVがR12.6以降の場合のみ有効となります。

■R12.8.2まで、およびR12.9の場合、1リクエストあたり使用可能な環境変数の最大数は400個です。制限数を超えて投入を行った場合、リクエストはエラーとなります。R12.8.3以降、およびR12.9.1以降の場合は個数の制限はありません。

■Windows版SVに投入されるジョブリクエストについては、次の制限があります。

- 投入側であらかじめ1環境変数あたりのバイト数を環境変数名(=も含む)と値あわせて8186byte以内にしておく必要があります。ただし後述のenvvarsファイルをWindows版SV側に設定する場合、envvarsに記述できるのは1環境変数あたり環境変数名(=も含む)と値あわせて1023byte以内となります。

- 以下の注意事項はバージョンによって異なります。

  R12.8.2まで、およびR12.9

  - ジョブ起動時に渡すことのできる環境変数の合計サイズの上限値はデフォルトで最大10KB(ただしR12.5.4より以前は8KB)です。システム全体の環境変数（システム環境変数とJobCenterが使用している環境変数および単位ジョブ間で引き継ぐ環境変数）の合計サイズがこの上限値を超えた場合、後続のジョブはエラーとなります。

  - ジョブ起動時に渡すことのできる環境変数の合計サイズの上限値はR12.5.4以降では次のように拡張できます。(ただし次項の通り、拡張した結果OSシステム全体の環境変数の合計サイズを超えないよう注意する必要があります)

    例：上限値を15KBに拡張する場合

    1. 変数名：NQS_ENVIRONMENT_BUFFER、変数値：15000を、環境変数が引き継がれる(ジョブリクエストを実行する)SVのOSシステム環境変数に設定する

    2. SVのJobCenterを再起動する

  - OSシステム全体の環境変数の合計サイズが29KBを超えた場合、JobCenterは正常に動作できません。そのためJobCenterの単位ジョブ間で引継ぐ環境変数をNQS_ENVIRONMENT_BUFFER設定で拡張する場合、実質的な上限サイズは16KB以内を目安として運用してください。

15.2.3.1. 単位ジョブスクリプト内での対処

ジョブの実行に必要な環境変数を、単位ジョブのスクリプトでsetコマンドで設定します。この場合、有効範囲は設定した単位ジョブスクリプト内のみとなります。

15.2.3.2. JobCenter SV側で設定する場合の対処（envvarsファイル）

%InstallDirectory%\spool\private\root 配下に envvars というテキストファイルを作成し、必要な環境変数を記述することにより、環境変数を一括して定義できます。

なお %InstallDirectory% はJobCenter MG/SVのインストールディレクトリです。デフォルトは C:\JobCenter\SV となります。

envvars ファイルの記述フォーマット

```
環境変数名1=値
環境変数名2=値
....
```

■envvarsファイル内では、次のように設定しても%PATH%部分はマクロ置換されませんのでご注意ください。

```
BATCHPATH=c:\jobcenter\SV\lib;%PATH%
```

■envvarsファイルに設定された環境変数は、全ユーザに適用されます。

■envvarsに記述できるのは、環境変数名=値 の記述1行につき1023byte以内となります。

15.2.3.3. 環境変数継承用の単位ジョブでの対処

■単位ジョブのEXPORTVARの機能を使用して、任意の環境変数を設定できます。この機能を使用するには、作成した単位ジョブパラメータ設定画面の結果タブにある「変数継承:STDOUT」の設定が必要となります。

詳細な設定方法や注意事項・制限については<基本操作ガイド>の「4.2.3　単位ジョブ間で情報を引き継ぐ」を参照してください。

■UNIX系MGの場合やUNIX系SVとWindows系のSVが混在した環境の場合等、環境変数「PATH」がそのまま継承できない場合には、環境変数「NQS_PATH_WIN」を設定することで、ジョブ実行時にSVの環境変数「PATH」に値を継承できます。

設定例) 単位ジョブのスクリプトに環境変数「NQS_PATH_WIN」を定義する

```
echo off
echo EXPORTVAR
echo NQS_PATH_WIN=C:\Windows;C:\Windows\System32;
echo EXPORTVAR
```

作成した単位ジョブには変数継承以外の記述をしないでください。

変数継承の設定と業務処理を単位ジョブ内に混在すると、業務処理でエラーが発生した場合に後続の単位ジョブに環境変数が継承されなくなります。

15.2.3.4. ジョブネットワークの即時投入時の対処

ジョブネットワークを即時投入する際、「複数の環境変数を設定する」を選択することで、投入するジョブネットワークの環境変数を設定できます。

詳細な設定方法や注意事項・制限については<基本操作ガイド>の「3.2.1　　即時投入する」を参照してください。

15.2.3.5. ジョブネットワークパラメータの「環境変数」タブでの対処

ジョブネットワークパラメータの「環境変数」タブで、単位ジョブ実行時に付与される環境変数を設定できます。

詳細な設定方法や注意事項・制限については<基本操作ガイド>の「3.3.4.4 環境変数」を参照してください。

またサブジョブネットワークについては、サブジョブネットワークパラメータの「環境変数」タブで、単位ジョブ実行時に付与される環境変数を設定できます。

詳細な設定方法や注意事項・制限については<基本操作ガイド>の「4.2.10.1 [環境変数]タブ」を参照してください。

15.2.3.6. JobCenter MG側で設定する場合の対処（環境変数「PATH」のみ）

■UNIX系MG － Windows系SVの場合

MG側がUNIX版である場合には、次のいずれかのシェルスクリプト内で環境変数「NQS_PATH_WIN」を設定しexportすることで、サーバ単位、またはユーザ単位でジョブ実行時にSVの環境変数「PATH」へ値を設定できます。

1. /etc/profile

2. ~/.nsifrc (ジョブネットワーク投入ユーザのhomeディレクトリに設置)

設定例)

```
VAR1=VAL1
export VAR1
```

/etc/profileと.nsifrcに同じ環境変数名で異なる値を設定した場合、~/.nsifrc内の設定が優先されます。

~/.nsifrcを設定した場合、スケジュール投入による運用を行っているユーザで必ずスケジュール定義の変更操作を行い、設定した環境変数をJobCenterに反映させる必要があります。

スケジュール変更の必要がないときは、一時的に仮のスケジュールの作成、保存、削除の操作を行ってください。

なお、他ユーザのジョブネットワークの実行を行う場合や、他ユーザのスケジュールを設定する場合は、そのジョブネットワーク所有者に設定された環境変数が使用されます。

/etc/profileや~/.nsifrcを設定したとき、ジョブネットワークの実行時の環境変数はログインユーザ自身のスケジュールの変更操作においてのみ設定されますので、必ずジョブネットワークを所有するユーザでログインし直して、前述のスケジュール変更操作を行ってください。

> すでに予定［確定］状態で表示されているトラッカ上については、上記の操作を行っても~/.nsifrcの設定内容が反映されませんので、［確定］状態のトラッカを一旦削除する必要があります。

■Windows系MG － Windows系SVの場合

MG側がWindows系の場合には、MG側のシステム環境変数に「NQS_PATH_WIN」の定義をしておくことで、ジョブ実行時にSVの環境変数「PATH」に値を設定できます。

15.2.3.7. JobCenter SV側で設定する場合の対処（環境変数「PATH」のみ）

■MG側・SV側共にWindows版である場合には、SV側のシステム環境変数にNQSDAEMON_PATH_EXPORT=1を指定することにより、ジョブ実行時にMG側の環境変数PATHを引き継げます。

変数名	NQSDAEMON_PATH_EXPORT
変数値	1

> ■本バージョンでは、NQSDAEMON_PATH_EXPORTに1以外の値が指定されていても、本機能が有効となりますが、今後指定方法が変更となる可能性がありますので、1を指定するようにしてください。
>
> ■本機能を無効にするためには、システム環境変数から「NQSDAEMON_PATH_EXPORT」を削除してください。

15.2.3.8. JobCenter MG側で設定する場合の対処（環境変数「PATH」以外）

■UNIX系MGの場合

MG側がUNIX版である場合は、次のシェルスクリプト内で任意の環境変数を設定しexportすることで、サーバ単位、またはユーザ単位でジョブ実行時にSVの環境変数を設定できます。

1. /etc/profile

2. ~/.nsifrc (ジョブネットワーク投入ユーザのhomeディレクトリに設置)

設定例)

```
VAR1=VAL1
export VAR1
```

/etc/profileと.nsifrcに同じ環境変数名で異なる値を設定した場合、~/.nsifrc内の設定が優先されます。

~/.nsifrcを設定した場合、スケジュール投入による運用を行っているユーザで必ずスケジュール定義の変更操作を行い、設定した環境変数をJobCenterに反映させる必要があります。

スケジュール変更の必要がないときは、一時的に仮のスケジュールの作成、保存、削除の操作を行ってください。

なお、他ユーザのジョブネットワークの実行を行う場合や、他ユーザのスケジュールを設定する場合は、そのジョブネットワーク所有者に設定された環境変数が使用されます。

/etc/profileや~/.nsifrcを設定したとき、ジョブネットワークの実行時の環境変数はログインユーザ自身のスケジュールの変更操作においてのみ設定されますので、必ずジョブネットワークを所有するユーザでログインし直して、前述のスケジュール変更操作を行ってください。

すでに予定［確定］状態で表示されているトラッカ上については、上記の操作を行っても~/.nsifrcの設定内容が反映されませんので、［確定］状態のトラッカを一旦削除する必要があります。

■Windows系MGの場合

MG側がWindows系の場合は、MG側のシステム環境変数に任意の環境変数の定義をしておくことで、ジョブ実行時にSVの環境変数へ値を設定できます。

システム環境変数を変更した場合は、JobCenterの再起動が必要です。OSによっては、OSの再起動が必要になる場合もあります。

15.3. ジョブ投入時に独自に設定される環境変数

JobCenterはジョブの投入時に独自に次の環境変数を設定しています。これらはジョブの実行時に参照可能です。JobCenterは投入元の環境変数を転送先にexportしませんが、投入元の基本的な環境変数の値は、これらの値を取り出すことで参照可能です。

15.3.1. UNIX版

UNIX版でキューにジョブリクエストが投入される時点で独自に設定される環境変数は次のとおりです。

表15.1 UNIX版の環境変数一覧

環境変数	内　容
QSUB_HOME	ジョブリクエスト投入時の環境変数HOMEの値
QSUB_HOST	ジョブリクエストの投入元マシン(サイト)の名前
QSUB_LOGNAME	ジョブリクエスト投入時の環境変数LOGNAMEの値（一部のシステム）
QSUB_MAIL	ジョブリクエスト投入時の環境変数MAILの値
QSUB_PATH	ジョブリクエスト投入時の環境変数PATHの値
QSUB_REQID	ジョブリクエストのリクエストID
QSUB_REQNAME	ジョブネットワーク内で付与されている単位ジョブ名（ジョブネットワークから投入の場合） ジョブリクエスト投入時のシェルスクリプト名（qsubで直接投入した場合）
QSUB_SHELL	ジョブリクエスト投入時の環境変数SHELLの値
QSUB_TZ	ジョブリクエスト投入時の環境変数TZの値
QSUB_USER	ジョブリクエストの投入ユーザ
QSUB_WORKDIR	ジョブリクエストが属するトラッカに割り当てられたディレクトリパス（ジョブネットワークから投入の場合）　ジョブリクエスト実行時のカレントディレクトリ（qsubで直接投入した場合）

また、CL/Winの即時実行、スケジュール実行、起動トリガのテキストログ監視実行、またはコマンドでジョブネットワークを投入した時点で、次の環境変数も設定されます。

表15.2 UNIX版の環境変数一覧（ジョブネットワークを投入した場合）

環境変数	内　容
NSJNW_BASEDATE	ジョブネットワークの予定開始時刻（ローカルタイム）（Windows版では投入運用日）
NSJNW_BASETIME	ジョブネットワークの予定開始日（ローカルタイム）（基準時刻を設定している場合は補正後の日付）
NSJNW_DIR	トラッカに割り当てられたディレクトリパス
NSJNW_HOST	ジョブネットワークの投入マシン(サイト)名
NSJNW_JNWNAME	ジョブネットワーク名
NSJNW_JNWTIME	トラッカを識別する文字列（開始(予定)時刻のGMT）
NSJNW_PARAM	ジョブネットワーク投入時に指定されたパラメータ
NSJNW_TRG_LOG_MATCH_LINE[注1]	起動トリガのテキストログ監視で、監視キーワードに一致したキーワードを含む行
NSJNW_TRG_LOG_MATCH_0[注1]	起動トリガのテキストログ監視で、監視キーワードの正規表現に一致した文字列

環境変数	内　容
NSJNW_TRG_LOG_MATCH_1～9[注1]	起動トリガのテキストログ監視で、監視キーワードの正規表現の小括弧「(」「）」に囲まれた部分 詳細は<基本操作ガイド>の「7.2.1.3　監視するテキストログファイルの情報を定義する」を参照してください。
NSJNW_TRG_NAME[注1]	ジョブネットワークを投入した起動トリガ定義の名前
NSJNW_UJNAME	ジョブネットワーク内で付与されている単位ジョブ名
NSJNW_UJERREC	異常終了とみなすエラーコードの最小値
NSJNW_USER	ジョブネットワークの投入ユーザ名

[注1]起動トリガのテキストログ監視実行で投入されたジョブネットワークの時に設定される環境変数です。

15.3.2. Windows版

Windows版でキューにジョブリクエストが投入される時点で独自に設定される環境変数は次のとおりです。

ただし、R12.10以降のバージョンのデフォルトの設定では、ジョブ実行ユーザが同じ変数名の環境変数を保持している場合、その値に上書きされます。

表15.3 Windows版の環境変数一覧

環境変数	内　容
QSUB_ComSpec	ジョブリクエスト投入元マシン(サイト)のcmd.exeのパス
QSUB_HOMEDRIVE	ジョブリクエスト投入時の環境変数HOMEDRIVEの値
QSUB_HOMEPATH	ジョブリクエスト投入時の環境変数HOMEPATHの値
QSUB_HOST	ジョブリクエスト投入元マシン(サイト)の名前
QSUB_Os2LibPath	ジョブリクエスト投入時の環境変数Os2LibPathの値
QSUB_Path	ジョブリクエスト投入時の環境変数PATHの値
QSUB_REQID	ジョブリクエストのリクエストID
QSUB_REQNAME	ジョブリクエストに対応する単位ジョブ
QSUB_SystemDrive	ジョブリクエスト投入時の環境変数SystemDriveの値
QSUB_SystemRoot	ジョブリクエスト投入時の環境変数SystemRootの値
QSUB_temp	ジョブリクエスト投入時の環境変数TEMPの値
QSUB_tmp	ジョブリクエスト投入時の環境変数TMPの値
QSUB_USERNAME	ジョブリクエスト投入時の環境変数USERNAMEの値
QSUB_windir	ジョブリクエスト投入時の環境変数windirの値
QSUB_WORKDIR	ジョブリクエストが属するトラッカに割り当てられたディレクトリパス

また、CL/Winの即時実行、スケジュール実行、起動トリガのテキストログ監視実行、またはコマンドでジョブネットワークを投入した時点で、次の環境変数も設定されます。

これらも、R12.10以降のバージョンのデフォルトの設定では、ジョブ実行ユーザが同じ変数名の環境変数を保持している場合、その値に上書きされます。

表15.4 Windows版の環境変数一覧（ジョブネットワークを投入した場合）

環境変数	内　容
NSJNW_BASEDATE	ジョブネットワークの投入時刻（Windows版では投入日付）
NSJNW_DIR	トラッカに割り当てられたディレクトリパス

環境変数	内　容
NSJNW_HOST	ジョブネットワークの投入マシン(サイト)名
NSJNW_JNWNAME	ジョブネットワーク名
NSJNW_JNWTIME	トラッカを識別する文字列（開始(予定)時刻のGMT）
NSJNW_PARAM	ジョブネットワーク投入時に指定されたパラメータ
NSJNW_TRG_LOG_MATCH_LINE[注1]	起動トリガのテキストログ監視で、監視キーワードに一致したキーワードを含む行
NSJNW_TRG_LOG_MATCH_0[注1]	起動トリガのテキストログ監視で、監視キーワードの正規表現に一致した文字列
NSJNW_TRG_LOG_MATCH_1～9[注1]	起動トリガのテキストログ監視で、監視キーワードの正規表現の小括弧「(」「）」に囲まれた部分 詳細は<基本操作ガイド>の「7.2.1.3　監視するテキストログファイルの情報を定義する」を参照してください。
NSJNW_TRG_NAME[注1]	ジョブネットワークを投入した起動トリガ定義の名前
NSJNW_UJNAME	ジョブネットワーク内で付与されている単位ジョブ名
NSJNW_UJERREC	異常終了とみなすエラーコードの最小値
NSJNW_USER	ジョブネットワークの投入ユーザ名

[注1]起動トリガのテキストログ監視実行で投入されたジョブネットワークの時に設定される環境変数です。

15.4. 環境変数「NQS_SITE」、「NQS_SITEDB」の設定が必要なモジュール

環境変数「NQS_SITE」、「NQS_SITEDB」の設定が必要なモジュールに関しては<コマンドリファレンス>の表1.1「コマンド一覧」を参照してください。

16. 日本以外のタイムゾーンで利用する

この章では、日本以外のタイムゾーンでJobCenterを利用したり、タイムゾーンが異なる環境にジョブを転送したりする場合に設定が必要な項目や、そのような環境で利用するにあたって有用な機能の説明を行います。

16.1. JobCenterセットアップ後に必要な設定

日本以外のタイムゾーンで利用する場合、JobCenterのインストールを行った後で、ご利用のプラットフォームに応じて本節の設定を行うようにしてください。

■UNIX版JobCenterで必要な設定

日本以外のタイムゾーンでJobCenterを利用する場合、daemon.confに「schedule_timezone」パラメータの設定が必要になります。

daemon.confファイルに以下の行をエディタで追加して、JobCenterを再起動してください。

```
schedule_timezone=OFF
```

daemon.confの詳細や、daemon.confに設定可能なその他のパラメータについては、5章　「JobCenter起動時の設定を変更する」 を参照してください。

■環境変数「TZ」が設定されない環境で必要な設定

「15.1.5　環境変数TZに関する注意事項(Linux、AIX版)」を参照して環境変数「TZ」を適切な値に設定してください。(例ではJST-9に設定していますが、利用するタイムゾーンに応じた値に設定してください)

■Linux、AIX版JobCenterで必要な設定

Linux、AIX版JobCenterでは、ユーザフレームの「スケジュール表示」タブを参照する際に、カレンダ分岐部品によっては本来分岐しないはずのフローに分岐しているように見える場合があります。

これは日付の変わり目がGMTで判断されているためで、表示上の問題であり実際のカレンダ分岐部品の日付判定動作には影響ありません。

カレンダ分岐部品のフロー表示についても正しく表示したい場合は、上述の「環境変数「TZ」が設定されない環境で必要な設定」の5を参照して、/etc/profile または ~/.nsifrc ファイルにTZ環境変数の設定を追加してください。（JobCenterの再起動は不要です）

~/.nsifrc ファイルについては「15.1.3.4 ジョブネットワークパラメータの「環境変数」タブでの対処」を参照してください。

16.2. カレンダへのタイムゾーン設定機能

通常、スケジュールの実行時刻はマネージャのタイムゾーンを基準とした時刻になりますが、日本とアメリカなど、異なるタイムゾーンに存在するサーバにジョブを転送して、JobCenterのスケジュール実行をマネージャとは異なるタイムゾーンを基準とした時刻で行いたい場合があります。

本機能を利用することにより、たとえば日本のタイムゾーンを基準としたスケジュール実行、アメリカのタイムゾーンを基準としたスケジュール実行が可能になり、システム内でタイムゾーンが異なるサーバが混在している場合でも、それぞれのタイムゾーンに応じた運用を行えます。

本機能を利用するには、マネージャ(ここではカレンダやスケジュール定義を持つマシンの意味)のOS自体のタイムゾーンが、夏時間を採用していないタイムゾーンである必要があります。JobCenterのスケジュール実行はスケジュール定義を持つサーバ側で行われますが、そのサーバのOSが夏時間を採用しているタイムゾーンで動作する環境では、夏時間と通常時間の切り替わりのタイミングでOSの時刻調整が行われるため、本機能が正しく動作しません。

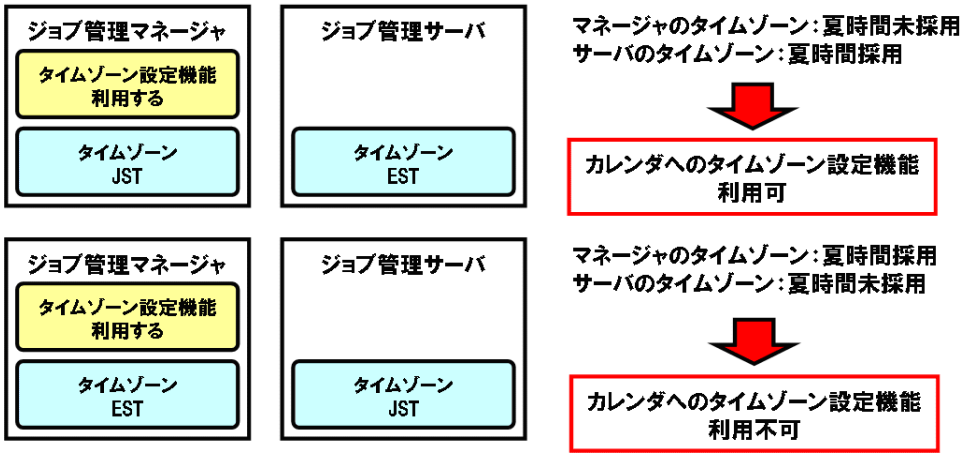

図16.1 カレンダへのタイムゾーン設定機能の利用条件例

16.2.1. 動作イメージ

カレンダへのタイムゾーン設定機能の動作イメージは以下の図のようになります。

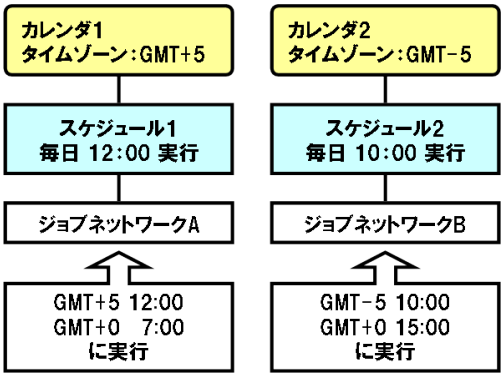

図16.2 カレンダへのタイムゾーン設定機能の動作イメージ

カレンダにタイムゾーン設定を行うと、そのカレンダを参照するスケジュールは設定されたタイムゾーンに応じて動作します。そのスケジュールと目的のジョブネットワークを関連付けると、そのジョブネットワークはマネージャマシンとは異なるタイムゾーンで動作するようになります。

上記の動作イメージでは、

1. スケジュール1はカレンダ1を稼働日の基準として参照しているため、スケジュール1と関連付けされたジョブネットワークAはGMT+5を基準とした時刻で起動されます。
2. また、スケジュール2はカレンダ2を稼働日の基準として参照しているため、スケジュール2と関連付けされたジョブネットワークBはGMT-5を基準とした時刻で起動されます。
3. もしスケジュール1についてカレンダ2を稼働日の基準とするように変更すると、スケジュール1と関連付けされたジョブネットワークAはGMT-5を基準として実行されるようになります。

タイムゾーンの設定を行わないカレンダを参照するスケジュールや、稼働日カレンダを参照しないスケジュールは、従来通りJobCenterがインストールされているOSのタイムゾーンを基準とした時刻で動作します。

16.2.2. カレンダへのタイムゾーン設定機能を有効にする

インストール直後では本機能は無効になっていて、カレンダへのタイムゾーンの設定はできません。本機能を利用するためには、まずカレンダへのタイムゾーン設定機能を有効にしてください。

カレンダへのタイムゾーン設定機能の有効・無効化は、[システム環境設定]ダイアログの[時刻設定]タブで行います。JobCenter管理者権限が必要です。

> カレンダへのタイムゾーン設定機能の有効・無効化はコマンドでも変更可能です。詳細については、<コマンドリファレンス>の「3.34 jc_syscfg システム環境設定の参照および変更」のsetcfgサブコマンドの説明を参照してください。

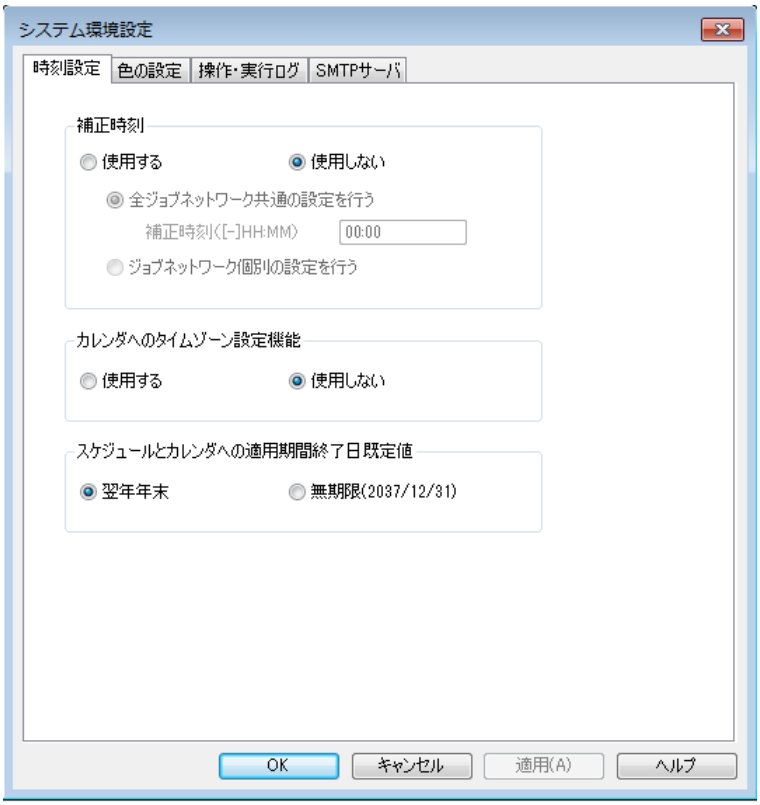

図16.3 システム環境設定

■カレンダへのタイムゾーン設定機能

項　目	意　味
使用する	カレンダへのタイムゾーン設定機能が有効になります。
使用しない	カレンダへのタイムゾーン設定機能が無効になります。

16.2.3. カレンダにタイムゾーンの設定を行う

[カレンダへのタイムゾーン設定機能]を有効にしてからカレンダ定義を右クリックすると、メニューに「タイムゾーン設定」項目が追加されます。「タイムゾーン設定」を選択すると、以下のタイムゾーン設定ダイアログが表示されます。

図16.4 カレンダへのタイムゾーンの設定ダイアログ

■カレンダへのタイムゾーン設定

選択したカレンダにタイムゾーンの設定を行うかどうかを選択します。

項　目	意　味
タイムゾーン設定を行わない	このカレンダを稼働日の基準としたスケジュールは、JobCenterインストールマシンのOSのタイムゾーンを基準として動作します。
タイムゾーン設定を行う	カレンダがどのタイムゾーンを基準時刻とするかが設定できるようになります。

［タイムゾーン設定］

■GMTからのオフセット

カレンダに設定するタイムゾーンを GMT+0 からのオフセット値として設定します。-12:59～12:59の範囲で設定します。

■夏時間の設定を行う

カレンダに夏時間の設定を行うかどうかを選択します。

「夏時間の設定を行う」を選択すると、以下の夏時間開始、終了時刻と夏時間オフセットの設定が可能になります。

■夏時間開始時刻、夏時間終了時刻

夏時間の開始時刻、終了時刻を設定します。設定可能な範囲は以下の通りです。

項　目	設定範囲
月	1月 ～ 12月
週	第一 ～ 第四、最終
曜日	日曜日 ～ 土曜日
時刻	0:00 ～ 23:59

■夏時間オフセット

夏時間期間中のオフセット値を設定します。-12:59 ～ 12:59の範囲で設定します。

夏時間期間中は、「夏時間オフセット」で指定した時間分、時刻の調整が行われます。

16.2.4. スケジュールにタイムゾーンの設定を行ったカレンダを指定する

タイムゾーンの設定を行ったカレンダをスケジュールの稼働日の基準として指定します。

設定方法は<基本操作ガイド>の「6.1.3.3　独立した稼働日カレンダを元にスケジュールを作成する」と同様です。

カレンダへのタイムゾーン設定機能を「使用する」に設定している場合は、スケジュール設定ダイアログのカレンダ選択フィールドが以下のように変化します。

図16.5 カレンダ選択フィールド

項　目	意　味
カレンダ名	稼働日の基準として指定するカレンダ名が表示されます。
タイムゾーン設定	カレンダに設定されたタイムゾーンが表示されます。 タイムゾーンを設定していない場合は「未設定」と表示されます。
夏時間オフセット	夏時間期間中のオフセット値が表示されます。 夏時間の設定を行っていない場合は、「夏時間未使用」と表示されます。

これらの設定を行った上でスケジュールの[関連JNW]にジョブネットワークを関連付けることで、カレンダに設定されたタイムゾーンを基準とした時刻で、ジョブネットワークのスケジュール実行を行います。

16.2.5. ジョブネットワークや部品オブジェクトへの影響

カレンダへのタイムゾーン設定機能を利用してスケジュール実行を行った場合、ジョブネットワークや部品オブジェクトのパラメータには以下のような影響があります。

> ⚠ スケジュール実行だけでなく、スケジュールに関連付けられたジョブネットワークを即時投入した場合でも同様の影響がありますので注意してください。

■基準時刻がカレンダに設定されたタイムゾーンになるパラメータ

以下のパラメータを絶対時刻で指定した場合、指定した時刻はカレンダに設定されたタイムゾーンを基準として動作します。

- ジョブネットワークパラメータのクリティカルポイント警告の実行開始点、実行終了点
- 単位ジョブ、ERPジョブ、BIジョブ、PCジョブ、WOBSジョブのパラメータのクリティカルポイント警告の実行開始点、実行終了
- 外部ファイル待ち合わせの更新基準時刻、タイムアウト
- 時刻待ちの待ち合わせ時刻
- イベント送信の受信確認、タイムアウト

- イベント受信のタイムアウト

たとえば、GMT+3の設定を行ったカレンダを稼働日の基準として指定したスケジュールと、クリティカルポイント警告の実行終了点に10:00の設定がされたジョブネットワークを関連付けた場合、ジョブネットワークに設定した実行終了点の10:00はGMT+3の10:00で警告動作が行われます。

■カレンダ分岐

カレンダ分岐で選択されたスケジュールの実行日の判断基準が、スケジュールが稼働日の基準として指定しているカレンダのタイムゾーンに依存します。

16.2.6. 夏時間の設定を行う場合の動作

カレンダへのタイムゾーン設定機能では、スケジュールのルールに設定された時刻をカレンダに設定されたタイムゾーン相当の時刻に変換して予定トラッカとして投入します。

カレンダに夏時間を考慮したタイムゾーンの設定を行った場合の動作例は、以下のとおりです。

■夏時間を設定した場合のスケジュール実行例

東部夏時間(GMT-5 3月第二日曜日 2:00 ～ 11月第一日曜日 2:00 夏時間オフセット +1:00)の条件でカレンダに東部夏時間のタイムゾーン設定を行うと、次のように動作します。

1. スケジュールの実行時刻が標準時間帯(夏時間期間外)の場合は、GMT-5相当の時刻に変換してスケジュール登録を行います。

2. 実行時刻が夏時間帯の場合は、GMT-4相当の時刻に変換してスケジュール登録を行います。

例えば、毎日12:00に実行するスケジュールと、東部夏時間のタイムゾーン設定がされたカレンダを稼働日の基準として設定した場合の動作は次のとおりです。

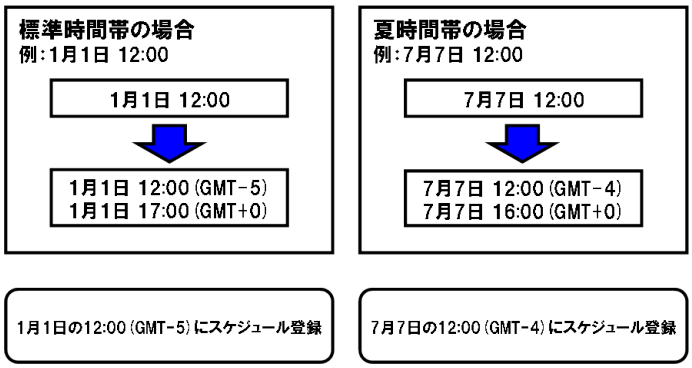

図16.6 夏時間動作例

■夏時間開始時刻周辺でのスケジュール実行時刻

東部夏時間の場合は2:00が夏時間開始時刻になるので、2:00より前(1:59まで)のスケジュールは標準時間帯のスケジュールとして、2:00以降は夏時間帯のスケジュールとして登録されます。

ただし、たとえば1:30、2:30、3:30に実行といったスケジュールを作成している場合は、1:30と2:30のスケジュールがそれぞれ1:30(GMT-5)、2:30(GMT-4)で計算されるので同じ時刻にスケジュールされることになります。

このようにスケジュールの実行時刻の重複が発生した場合は、標準時間帯のスケジュールとして実行されます。なおジョブネットワーク内の部品のクリティカルポイント警告等の動作も、標準時間帯のタイムゾーンを基準とした動作になります。

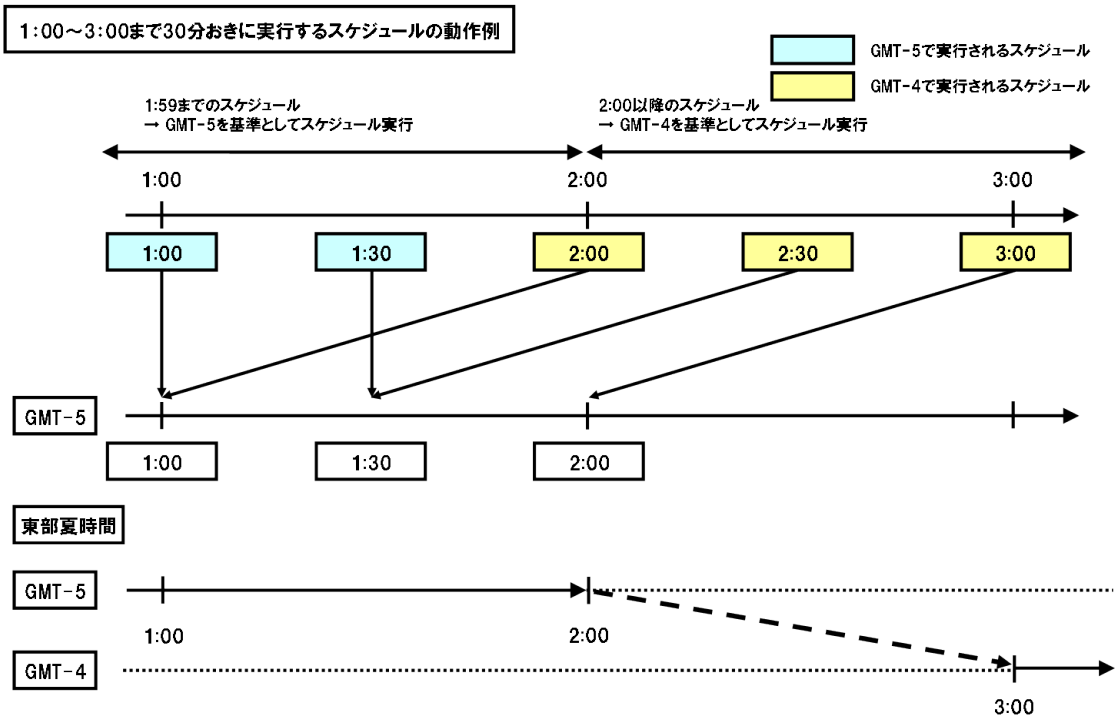

図16.7 夏時間開始時刻周辺での動作例

■夏時間終了時刻周辺でのスケジュール実行時刻

東部夏時間の場合、2:00が夏時間終了時刻になりますが、このとき、1:00 ～ 1:59までの時間帯が夏時間帯と標準時間帯で重複した時間となります。

本機能では、夏時間終了日の1:00 ～ 1:59までのスケジュールを標準時間帯のタイムゾーンのスケジュールとして時刻の計算を行います。夏時間帯の1:00 ～ 1:59でのスケジュール実行は行われません。

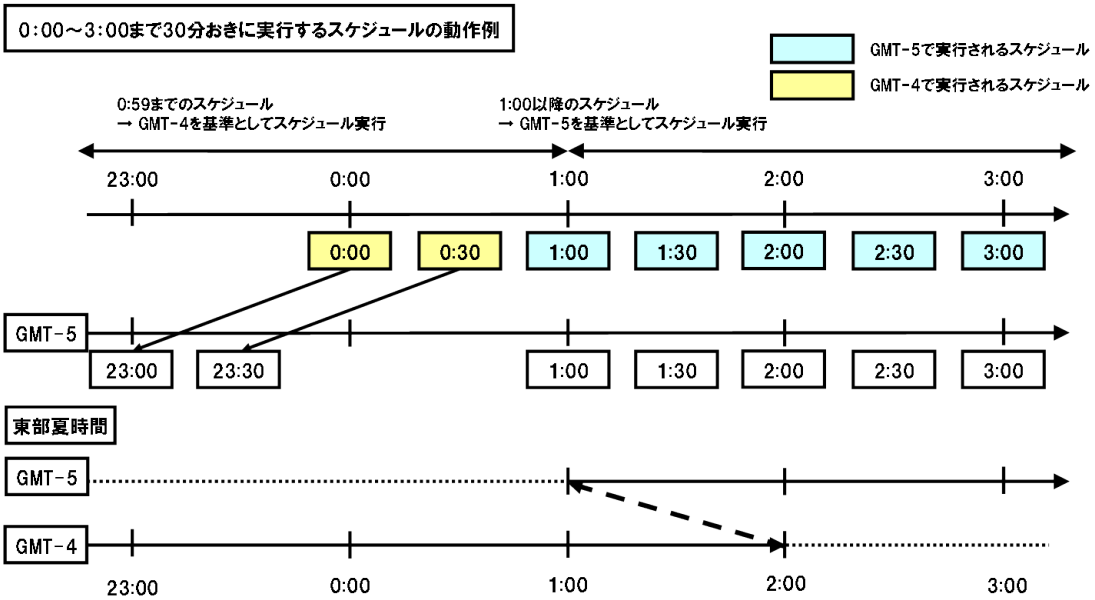

図16.8 夏時間終了時刻での動作例

16.2.7. タイムゾーンに応じた監視を行う

CL/Winは、デフォルトではインストールされているOSのタイムゾーンでトラッカの予想開始、終了時間などの表示を行います。

OSが海外のタイムゾーンで動作するマシンにJobCenterをインストールして監視したり、カレンダへのタイムゾーン設定機能を用いて海外のタイムゾーンを基準時刻としてスケジュール実行するために、CL/Winの「CL/Winの表示タイムゾーン変更機能」により、時刻表示の基準タイムゾーンを変更できます。

たとえば、GMT+1のタイムゾーンで動作するJobCenter上のトラッカを日本(GMT+9)から監視する場合、本機能を利用して、表示タイムゾーンを「GMT+1」に変更すると、GMT+1上で動作するトラッカを適切なタイムゾーンで監視できます。

また、夏時間を採用しているタイムゾーンの場合でも、夏時間帯と標準時間帯でCL/Winの表示タイムゾーンを切り替えることで監視を行うこともできます。

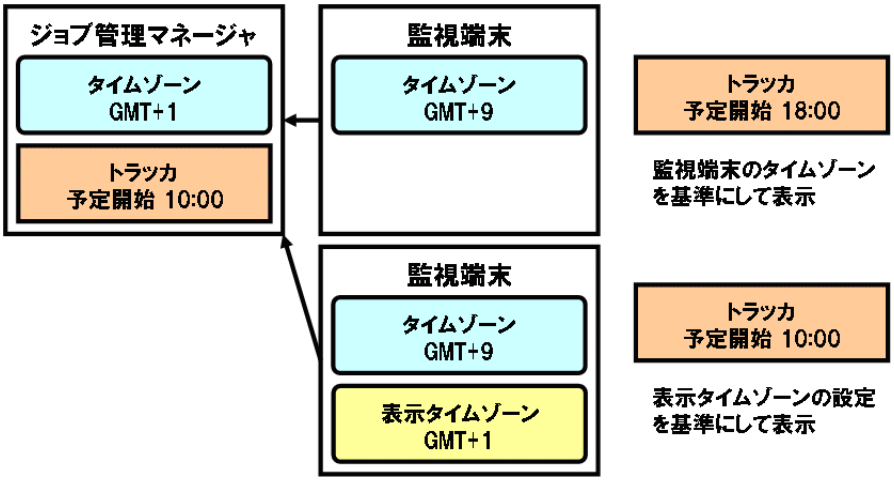

図16.9 表示タイムゾーン変更機能の利用例

CL/Winの表示タイムゾーンを変更する手順は次のとおりです。

1. メインメニューの「表示」 - 「表示タイムゾーンの設定」を選択します。
2. 「表示タイムゾーンの設定」ダイアログで「表示タイムゾーンを変更」を選択して、表示したいタイムゾーンを指定します。-12:59 ～ 12:59までの範囲で設定します。

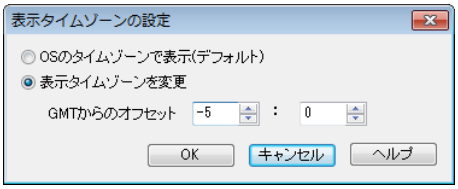

図16.10 表示タイムゾーン設定ダイアログ

CL/Winの表示タイムゾーン変更機能を利用すると、以下の各画面の項目の時刻について、上記で設定したタイムゾーンを基準としたものに変更されます。

画面名	タイムゾーン変更が反映される設定(表示)項目
トラッカ一覧(テキストモード)	予定開始時間、予定終了時間、実績開始時間、実績終了時間
トラッカ一覧(ガントモード)	ガントチャート期間
トラッカ印刷ダイアログ	印刷期間

トラッカフロー画面(部品一覧表示時)	予定開始時間、予定終了時間、実績開始時間、実績終了時間
トラッカ詳細情報画面	予定開始時間、予定終了時間、実績開始時間、実績終了時間
部品検索設定ダイアログ	検索期間設定の検索開始日
部品検索フレーム	予定開始時間、予定終了時間、実績開始時間、実績終了時間
スケジュール設定画面	スケジュールの実行時間リスト
カレンダ設定画面	時間、適用期間
スケジュール表示	予定開始時間、予定終了時間、実績終了時間
スケジュール印刷設定ダイアログ	出力範囲
ジョブネットワークの即時投入ダイアログ	投入時間
ERPフレーム	計画開始日時
ERPジョブ検索ダイアログ	開始条件日時
トラッカフロー画面	時刻待ち合わせ部品の実際待ち時刻に表示される時間
カレンダ分岐部品設定ダイアログ	スケジュール実行時間
イベント一覧	タイムアウト時刻

特に、ジョブネットワークの即時投入ダイアログの投入時間の時刻が設定されたタイムゾーンを基準とした時刻になるので、任意のタイムゾーンに応じた予定(確定)トラッカを作成することができます。

16.2.8. 注意事項

■環境変数「TZ」への影響

本機能は、カレンダに設定されたタイムゾーンに応じて、スケジュールの実行時刻を調整する機能です。カレンダに設定されたタイムゾーンとジョブ実行時の環境変数「TZ」とは関係しません。

UNIX版の場合、マネージャマシンに設定されている環境変数「TZ」が実行されるジョブにも設定されるため、ジョブを別のOSタイムゾーンで動作するマシンにリモート転送して実行する環境で、スクリプト内のコマンドがTZの影響を受ける場合は、スクリプト内でTZを再設定する等の注意が必要です。

■カレンダへのタイムゾーン設定機能の影響を受けないもの

カレンダへのタイムゾーン設定機能により、スケジュールの実行時刻やジョブネットワーク内の部品の時刻待ち等の動作は影響を受けますが、以下のものは本機能の影響を受けず、JobCenterインストールマシンのOSのタイムゾーンを基準とした時刻で表示、記録されます。

- ジョブネットワークトラッカのトラッカログ(トラッカフロー画面の「ログ」タブの内容)
- 操作･実行ログ、イベントログ
- CUIで提供されるJobCenterの各種コマンド(jnwsummaryやjnwschprt等)

■運用中の設定変更について

システムの運用が始まった後でカレンダへのタイムゾーン設定機能の有効化・無効化、カレンダのタイムゾーンの設定値を変更する場合は、スケジュールのルールを無効化する、またはジョブネットワークとスケジュールの関連付けを解除してから行うようにしてください。

これらの設定を変更すると、変更した時点でスケジュール実行時刻の計算が行われるので、予期しないスケジュール実行が開始してしまう可能性があります。

そのため、設定を変更した後は必ずトラッカ一覧で予定(確定)トラッカを確認してください。

17. JobCenter環境のバックアップ・復元

この章では、JobCenter 環境のバックアップおよび復元手順について説明します。 「17.2 JobCenterの構成情報をバックアップ・復元する」ではJobCenter の機能によるアプリケーションレベルでのバックアップ・復元方法を、 「17.3 イメージバックアップを利用したJobCenter環境のバックアップ・復元」では仮想化環境やファイルシステム、専用アプリケーションの機能によるイメージバックアップを用いたバックアップ・復元手順について説明します。

どちらの手順で復旧を行なう場合も、「17.1 バックアップ・復元に関する諸注意」を確認の上実施してください。

17.1. バックアップ・復元に関する諸注意

17.1.1. 復元時の影響

連携するJobCenter MG/SV は、お互いに状態を更新して整合性をとりながら動作しています。そのため、連携を行っている一方または一部の JobCenter 環境のみをバックアップから復元すると、他方の環境との状態不整合が発生します。

状態不整合が発生している場合、次の動作影響がありますので、バックアップからの復元を全環境に対して実施するか、整合性を取るための操作を実施する必要があります。

整合性を取るための操作は、バックアップから復元した環境のJobCenterを起動する前に実施する必要があります

■MGからSVへジョブ投入している状態でMGをバックアップから復元する場合

復元後のMGからSVに投入する単位ジョブの実行がエラー終了になる可能性があります。

■MGからSVへジョブ投入している状態でSVをバックアップから復元する場合

SVで実行中だった単位ジョブは「実行中」のままとなり、自動的には終了になりません。

■イベント送信部品、イベント受信部品で連携を行っている状態でバックアップから復元する場合

イベント送信・受信部品による待ち合わせが終了しない、またはタイムアウトになる可能性があります。

動作影響が発生する条件の詳細および、整合性を取るための操作の詳細は「17.4 復元後の整合性確認」を参照してください

17.1.2. 運用上の注意

バックアップ方法によって、オンラインバックアップの可否やバックアップを取ることができる範囲が異なります。定期的なJobCenter環境のバックアップ運用を検討する場合は、運用方法に合わせてバックアップの方法を決定してください。

■JobCenter のバックアップ・復元機能

- JobCenter を起動した状態でバックアップ情報を取得することが可能です。
- JobCenter定義のバックアップは<基本操作ガイド>の12章 「アップロード・ダウンロード機能」 の機能と組み合わせる必要があります。

■イメージバックアップを利用するバックアップ・復元

- 必ず JobCenter を停止した状態でイメージバックアップ、スナップショットを取得してください。
- JobCenterサイトが利用するディスク領域をイメージバックアップすることで、JobCenter定義も同時にバックアップすることができます。

17.2. JobCenterの構成情報をバックアップ・復元する

この章では、JobCenterの設定ファイルやキュー情報、ユーザマッピング等のJobCenter構成情報のバックアップと復元方法について説明します。

17.2.1. 構成情報のバックアップ・復元機能

JobCenterの動作に必要な情報には、以下の２種類があります。

構成情報	JobCenter　MG/SVのキュー、ユーザマッピング、マシン一覧、マシングループなどの各種情報。　詳細は「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。
定義情報	JobCenter　MGのジョブネットワーク、スケジュール、カレンダに関する定義データやパラメータデータ。

この章では、構成情報のバックアップ・復元機能について説明します。定義情報のバックアップ・復元機能については、 <Helper機能利用の手引き>を参照してください。

本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現在サポートしていません。

17.2.1.1. 動作イメージ

構成情報のバックアップ・復元機能の動作イメージは以下の図のようになります。

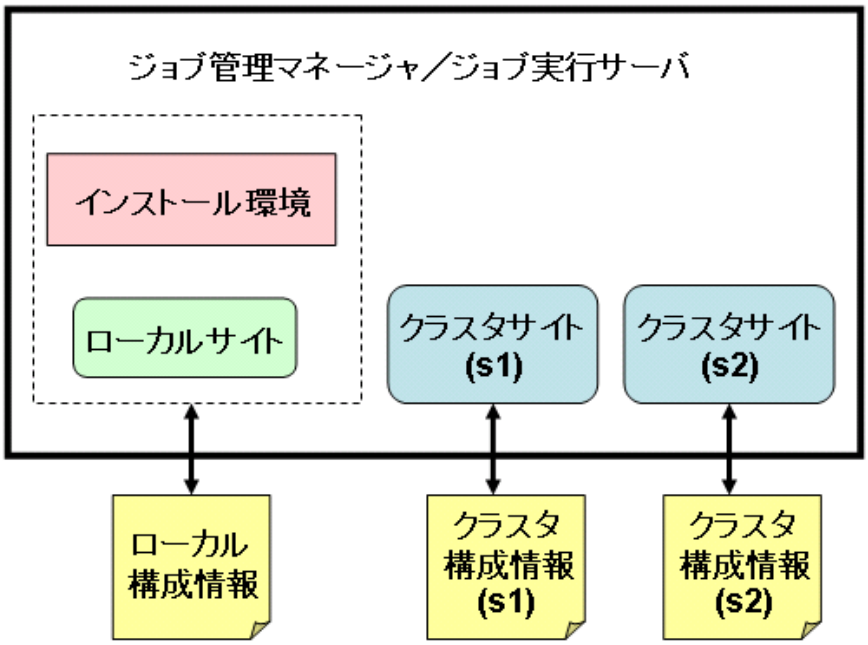

図17.1 構成情報のバックアップ・復元機能の動作イメージ

JobCenterの構成情報は、「インストール環境」、「ローカルサイト」、「クラスタサイト」に存在します。「インストール環境」は、JobCenterがインストールされているマシンの環境のことです。「ローカルサイト」は、JobCenterインストール時にデフォルトで作成されるジョブネットワーク実行環境のことです。「クラスタサイト」は、JobCenterクラスタ機能を利用して作成されるサイト環境です。サイト環境については　<クラスタ機能利用の手引き>を参照してください。

本機能では、jc_backupコマンドを用いて、「インストール環境」、「ローカルサイト」、「クラスタサイト」から構成情報を取得し、ファイルに保存することができます。また、jc_restoreコマンドを用いてファイルから構成情報を復元することができます。

構成情報ファイルには「ローカル構成情報」と「クラスタ構成情報」の2種類あります。「ローカル構成情報」は、インストール環境とローカルサイトの構成情報を保存しています。「クラスタ構成情報」にはクラスタサイトの構成情報を保存しています。「ローカル構成情報」のバックアップ・復元はJobCenterがインストールされているマシン単位で行います。「クラスタ構成情報」のバックアップ・復元は、クラスタサイト単位で行う必要があります。また、構成情報のバックアップ・復元は、ジョブ管理マネージャだけでなく、ジョブ実行サーバでも同様に行う必要があります。

構成情報のバックアップ・復元の手順を図17.1「構成情報のバックアップ・復元機能の動作イメージ」を例に簡単に説明すると、初期環境として、JobCenterインストール後に、クラスタサイト(s1)とクラスタサイト(s2)が作成されているとします。この場合、全ての環境の構成情報をバックアップするには、jc_backupコマンドを使って、ローカル構成情報とクラスタ構成情報(s1)、クラスタ構成情報(s2)の3つをバックアップします。復元はjc_restoreコマンドを使って、3つの構成情報ファイルから行います。バックアップおよび復元する順番は任意です。また、バックアップした構成情報ファイルのうち、いくつか選択して復元することも可能です。

構成情報のバックアップ時には、JobCenterは起動していても、停止していても問題ありませんが、復元時にはJobCenterは停止している必要があります。具体的には、ローカル構成情報の復元時にはローカルのJobCenterが停止している必要があり、クラスタ構成情報の復元時には、対応するクラスタサイトが停止している必要があります。

17.2.1.2. バックアップ・復元手順について

■ローカル構成情報のバックアップ手順

1. 事前確認

   - JobCenterの動作状態

     起動していても停止していても問題ありません。

2. ローカル構成情報のバックアップ

   以下のコマンドを実行して構成情報のバックアップを行います。このコマンドは、UNIXの場合はrootユーザ、Windowsの場合はJobCenter管理者で実行する必要があります。-oオプションには、出力ファイル名を絶対パスまたは相対パスで指定します。省略した場合は、コマンドを実行したディレクトリ直下にjc_conf_local_YYYYMMDDhhmmss.jpfというファイル名で作成されます。

   UNIXの場合

   ```
   /usr/lib/nqs/gui/bin/jc_backup conf [-o <output>]
   ```

   Windowsの場合

   ```
   <インストールディレクトリ>\bin\jc_backup conf [-o <output>]
   ```

3. バックアップファイルの確認

   「Backup　successfully.」というメッセージが出力されたらバックアップは成功です。ローカル構成情報ファイルが作成されていることを確認してください。-oオプションを指定した場合は、指定したファイル名で作成されます。指定しなかった場合は、コマンドを実行したディレクトリ直下にデフォルトのファイル名で作成されます。

■ローカル構成情報の復元手順

1. 事前確認

- JobCenterの動作状態

  停止している必要があります。

- OSのユーザアカウント

  バックアップ時にJobCenterで使用していたユーザが、復元環境にも存在する必要があります。バックアップ時に存在したユーザの情報は、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もバックアップ時と一致させる必要があります。UNIXの場合は、OSのコマンドで追加/変更できます。Windowsの場合は、「13.4　ユーザの設定」を参照してください。

- ユーザ情報作成

  CL/Winから、全JobCenterユーザで漏れなくログインした実績が必要です。ログインすることで、ユーザごとの設定ファイルが作られますので、必ず全ユーザで実行してください。

2. ローカル構成情報の復元

以下のコマンドを実行して構成情報の復元を行います。このコマンドは、UNIXの場合はrootユーザ、Windowsの場合はJobCenter管理者で実行する必要があります。復元にはjc_backupコマンドで作成したローカル構成情報ファイルが必要です。ローカル構成情報ファイルの指定は、絶対パスおよび相対パスのどちらでも可能です。

・UNIXの場合

```
/usr/lib/nqs/gui/bin/jc_restore conf <ローカル構成情報ファイル>
```

・Windowsの場合

```
<インストールディレクトリ>\bin\jc_restore conf <ローカル構成情報ファイル>
```

3. 復元の確認

最後に「Restore configuration successfully.」というメッセージが出力されたら復元操作は成功です。復元内容は、「17.2.2　バックアップ・復元対象の構成情報」を参照して、確認を行うようにしてください。

> 構成情報の中でも、ユーザマッピングやマシングループ、マシン一覧等の設定は連携先のマシンの設定と整合的でないといけないので、なるべく連携先のマシンとセットでバックアップ・復元を行うようにしてください。

■クラスタ構成情報のバックアップ手順

1. 事前確認

- JobCenterの動作状態

  対象のクラスタサイトは、起動していても停止していても問題ありません。

- クラスタ関連のリソースの状態

  JobCenter用のリロケータブルIPアドレスが有効かつJobCenterのクラスタDBへのアクセスが可能な状態で作業を行う必要があります。

2. クラスタ構成情報のバックアップ

以下のコマンドを実行して構成情報のバックアップを行います。このコマンドは、UNIXの場合はrootユーザ、Windowsの場合はJobCenter管理者で実行する必要があります。-cオプションには、クラスタDBパスを絶対パスまたは相対パスで指定します。-oオプションには、出力ファイル名を絶対パスまたは相対パスで指定します。省略した場合は、コマンドを実行したディレクトリ直下にjc_conf_cluster_YYYYMMDDhhmmss.jpfというファイル名で作成されます。

UNIXの場合

```
/usr/lib/nqs/gui/bin/jc_backup conf -c <クラスタDBパス> [-o <output>]
```

Windowsの場合

```
<インストールディレクトリ>\bin\jc_backup conf -c <クラスタDBパス> [-o <output>]
```

3. バックアップファイルの確認

「Backup　successfully.」というメッセージが出力されたらバックアップは成功です。クラスタ構成情報ファイルが作成されていることを確認してください。-oオプションを指定した場合は、指定したファイル名で作成されます。指定しなかった場合は、コマンドを実行したディレクトリ直下にデフォルトのファイル名で作成されます。

■クラスタ構成情報の復元手順

1. 事前確認

- JobCenterの動作状態

  対象のクラスタサイトは、停止している必要があります。ただし、最低一度は起動した実績が必要です。クラスタサイトの起動/停止方法については、下記のマニュアルを参照してください。

  - UNIX版

    <クラスタ機能利用の手引き>の「2.3.7 サイトの起動確認（運用系・待機系）」

  - Windows版

    <クラスタ機能利用の手引き>の「2.4.7 サイトの起動確認」

- クラスタ関連のリソースの状態

  JobCenter用のリロケータブルIPアドレスが有効かつJobCenterのクラスタDBへのアクセスが可能な状態で作業を行う必要があります。

- OSのユーザアカウント

  バックアップ時にJobCenterで使用していたユーザが、復元環境にも存在する必要があります。バックアップ時に存在したユーザの情報は、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もバックアップ時と一致させる必要があります。UNIXの場合は、OSのコマンドで追加/変更できます。Windowsの場合は、「13.4　ユーザの設定」を参照してください。また、クラスタ環境でのUIDについては、<クラスタ機能利用の手引き>の「2.2.1 ユーザIDの統一」も参照してください。

- ユーザ情報作成

  CL/Winから、全JobCenterユーザで漏れなくログインした実績が必要です。ログインすることで、ユーザごとの設定ファイルが作られますので、必ず全ユーザで実行してください。

2. クラスタ構成情報の復元

以下のコマンドを実行して構成情報の復元を行います。このコマンドは、UNIXの場合はrootユーザ、Windowsの場合はJobCenter管理者で実行する必要があります。復元にはjc_backupコマンドで作成したクラスタ構成情報ファイルが必要です。クラスタ構成情報ファイルの指定は、絶対パスおよび相対パスのどちらでも可能です。-cオプションには、クラスタDBパスを絶対パスまたは相対パスで指定します。

・UNIXの場合

```
/usr/lib/nqs/gui/bin/jc_restore conf -c <クラスタDBパス> <クラスタ構成情報ファイル>
```

・Windowsの場合

```
<インストールディレクトリ>\bin\jc_restore conf -c <クラスタDBパス> <クラスタ構成情報ファイル>
```

3. 復元の確認

最後に「Restore configuration successfully.」というメッセージが出力されたら復元操作は成功です。復元内容は、「17.2.2　バックアップ・復元対象の構成情報」を参照して、確認を行うようにしてください。

> ⚠ 構成情報の中でも、ユーザマッピングやマシングループ、マシン一覧等の設定は連携先のマシンの設定と整合的でないといけないので、なるべく連携先のマシンとセットでバックアップ・復元を行うようにしてください。

17.2.2. バックアップ・復元対象の構成情報

構成情報のバックアップ・復元機能では以下の情報をバックアップ・復元します。

表17.1 バックアップ対象の構成情報

存在場所	構成情報	説明
ローカルサイト/クラスタサイト[注1]	キュー	作成されたキューの情報や設定および、NQSの重要な設定情報がバックアップ・復元されます。キューについては、4章 「キューの作成」 や<基本操作ガイド>の10章 「ネットワークキューイングシステム（NQS）の利用方法」 を参照してください。
	ユーザマッピング	ユーザマッピングの情報がバックアップ・復元されます。ユーザマッピングについては「3.2　ユーザの関連付けを行う（ユーザマッピング）」を参照してください。
	マシン一覧	マシン一覧画面の登録内容がバックアップ・復元されます。マシン一覧画面については、4章 「キューの作成」 や<基本操作ガイド>の「10.1.1　マシン一覧へマシンを追加する」等を参照してください。
	マシングループ	マシングループの情報がバックアップ・復元されます。「3.1.3 マシングループ構成（JobCenter　CL/Winからの登録）」等を参照してください。
	パーミッション	パーミッション設定の情報がバックアップ・復元されます。パーミッションについては、10章　「ユーザ権限（パーミッション設定）」 等を参照してください。
	イベント連携設定	イベント連携設定の情報がバックアップ・復元されます。イベント連携設定の詳細は、12章 「イベント連携」 等を参照してください。
	システム環境設定	システム環境設定の情報がバックアップ・復元されます。システム環境設定の詳細は、「6.5　システム環境の設定を変更する」等を参照してください。

	ユーザ環境設定	ユーザ環境設定の情報がバックアップ・復元されます。ユーザ環境設定の詳細は、<基本操作ガイド>の「8.9 トラッカー覧の表示開始時間を指定して表示する」等を参照してください。
	デフォルトパラメータ	各ユーザのデフォルトパラメータ設定の情報がバックアップ・復元されます。デフォルトパラメータの詳細は、「6.1 JobCenter部品のデフォルトパラメータの設定を定義する」等を参照してください。
	その他	キュー～ユーザ環境設定以外の設定ファイルがバックアップ・復元されます。具体的な内容は、以下のとおりです。[注2]
		○UNIX版 ■デーモン設定ファイル(5章 「JobCenter起動時の設定を変更する」 参照) ■jcwebserver設定ファイル(「5.7 jcwebserverの動作設定について」参照) ■jnwschprtコマンドのコンフィグレーションファイル(<コマンドリファレンス>の「3.2.2.1 コンフィグレーションファイル」参照) ■SAP RFC用システム設定ファイル(<SAP機能利用の手引き>の「1.1.1 接続パラメータファイルを設定する」参照) ■WebOTX Batch Server連携設定ファイル(<WebOTX Batch Server連携機能利用の手引き>の「2.1.2 設定ファイルを作成する」参照) ■ログ設定ファイル(「25.2 エラーログファイルの設定を変更する」参照) ■jcres設定ファイル(<コマンドリファレンス>の「3.30 jcres JobCenter MG/SV専用のHTTPデーモン」参照)
		○Windows版 ■デーモン設定ファイル(5章 「JobCenter起動時の設定を変更する」 参照) ■jcwebserver設定ファイル(「5.7 jcwebserverの動作設定について」参照) ■jnwschprtコマンドのコンフィグレーションファイル(<コマンドリファレンス>の「3.2.2.1 コンフィグレーションファイル」参照) ■ジョブ実行環境設定ファイル(「13.3.3 ジョブの実行設定」参照) ■SAP RFC用システム設定ファイル(<SAP機能利用の手引き>の「1.1.1 接続パラメータファイルを設定する」参照) ■WebOTX Batch Server連携設定ファイル(<WebOTX Batch Server連携機能利用の手引き>の「2.1.2 設定ファイルを作成する」参照) ■サイト設定ファイル(「5.6.3 サイト設定ファイル（site.conf）を作成する」参照) ■サーバ環境のマッピング情報(HOSTS.NQS) ■ログ設定情報(「25.2 エラーログファイルの設定を変更する」参照) ■envvarsファイル(「15.2.3.2 JobCenter SV側で設定する場合の対処（envvarsファイル）」参照)[注3]

		■jcres設定ファイル(<コマンドリファレンス>の「3.30　jcres JobCenter MG/SV専用のHTTPデーモン」参照) ■LDAPサーバ設定(「11.1 LDAPサーバ連携(Windows)」参照)
インストール環境[注1]		キュー～ユーザ環境設定以外の設定ファイルがバックアップ・復元されます。具体的な内容は、以下のとおりです。[注2]
		○UNIX版 ■デーモン設定ファイル(5章 「JobCenter起動時の設定を変更する」 参照) ■jcwebserver設定ファイル(「5.7　jcwebserverの動作設定について」参照) ■名前解決設定ファイル(「2.4　UNIX環境における名前解決方法」参照) ■SAP 接続先設定ファイル(<SAP機能利用の手引き>の「1.1.1 接続パラメータファイルを設定する」参照) ■文字コード変換設定ファイル(「9.2.1 UNIX版JobCenterの文字コード変換を設定する」参照) ■言語環境ファイル(「25.1 トラブルシューティングQ&A」参照) ■起動スクリプト(comagent.sh, jnwcaster.sh, jnwengine.sh, jcdbs.sh, nqs.sh) ■デーモンパラメータファイル(nqsstart.param) ■jcdbs設定ファイル(jcdbs.conf) ■~/.nsifrcファイル(「15.1.3.5 JobCenter MG側で設定する場合の対処」参照)[注3]
		○Windows版 ■デーモン設定ファイル(5章 「JobCenter起動時の設定を変更する」 参照) ■jcwebserver設定ファイル(「5.7　jcwebserverの動作設定について」参照) ■名前解決設定ファイル(「2.3　Windows環境における名前解決方法」参照) ■SAP 接続先設定ファイル(<SAP機能利用の手引き>の「1.1.1 接続パラメータファイルを設定する」参照) ■WebOTX Batch Server連携設定ファイル(<WebOTX Batch Server連携機能利用の手引き>の「2.1.2　設定ファイルを作成する」参照) ■サイト設定ファイル(「5.6.3 サイト設定ファイル（site.conf）を作成する」参照) ■サーバ環境のマッピング情報(HOSTS.NQS) ■jcdbs設定ファイル(jcdbs.conf)
■JobCenter MG/SVのレジストリ情報[注4]		

注1　図17.1「構成情報のバックアップ・復元機能の動作イメージ」参照

注2　お客様ご自身で設定/変更するファイルを記載しています。JobCenterが自動的に設定/変更する内部ファイルは記載していません。バックアップされた全設定ファイルを知りたい場合

はjpf_statコマンド(<コマンドリファレンス>の「3.17　jpf_stat　JPFファイルの情報確認」)をご利用ください。

注3　R14.1.1以上のバージョンでバックアップ・復元できます。このファイルを利用している場合はR14.1.1以上の累積パッチを適用してください。

注4　レジストリ情報の内容は以下のとおりです。

情報	バックアップ対象	復元対象
JobCenter MG/SV のインストール情報	○	×
サーバの環境設定の総合設定のプロパティの設定値	○	○
JobCenter MG/SV に登録されているユーザの情報	○	×

「JobCenter　MG/SV　のインストール情報」は、インストールディレクトリのパスやバージョン情報など、インストール時に設定される情報であるため復元されません。

「サーバの環境設定の総合設定のプロパティ」について、詳細は「13.5.1 総合設定のプロパティ」を参照してください。

「JobCenter　MG/SV　に登録されているユーザの情報」は、復元後にサーバの環境設定でユーザの設定を行ってください。設定方法は「13.4 ユーザの設定」を参照してください。

17.2.3. 注意事項

■バックアップ・復元機能は、JobCenter環境が正常なときにおける構成情報のバックアップ・復元を目的としています。原因不明の障害時にはJobCenterのバックアップ・復元機能を使用しないでください。

■異なるプラットフォーム(OS)間のバックアップ・復元はできません。

■バックアップ・復元機能は、JobCenterのマシンIDも復元します。MG、SV間で連携している場合は、自分や相手先のマシン一覧や、マシングループの設定を確認して連携に問題がないか確認してください。

■バックアップ・復元機能は、Windowsのユーザプロファイルのバックアップ・復元はできません。ユーザプロファイルの設定内容がバックアップ時と復元時で異なると、JobCenterの動作に影響が出る可能性がありますので、設定内容を合わせるようにしてください。

■バックアップ・復元機能は、暗号化パラメータのバックアップ・復元はできません。暗号化パラメータを復元する場合は、<コマンドリファレンス>の「3.29　paramctrl　暗号化DBへのパラメータ登録」を参照の上、同じ値を再設定してください。

■JobCenter　MG/SV　のセットアップ言語（Windows版の場合はUNICODEモード/非UNICODEモードの設定値も含みます）について、復元する際は必ずバックアップした時の設定値とインストール時の設定値を同じ値にしてください。

■ジョブリクエストが特定のキュー上に残っている状態で復元を行わないでください。

17.3. イメージバックアップを利用したJobCenter環境のバックアップ・復元

この章では仮想化環境やファイルシステム、専用アプリケーションの機能によって、JobCenter の環境をディスクイメージとしてバックアップ・復元する場合の手順について説明します。

「ディスクイメージ」はスナップショットやチェックポイント等の機能によって作成されたデータや、それらの機能を介して作成されたイメージファイルを代表とした、静止点が確保されたデータを指します。静止点を設けず、ファイルコピーによってバックアップしたデータは本手順を適用することはできません。

17.3.1. イメージバックアップの作成

バックアップ・復元に利用する製品または機能のマニュアル等に従って、イメージバックアップを作成してください。この際、以下の条件を全て満たす状態で実施してください。

■バックアップするJobCenterサイトに実行中の処理がない状態にしてください

- 実行中のトラッカが存在しないこと

  CL/Win の [マネージャフレーム] - [マシン一覧] - [バックアップ対象のマシン(またはサイト)] - [トラッカ一覧] を確認し、実行中のトラッカが存在しないことを確認してください。

- 実行中、および実行待機中のジョブリクエストが存在しないこと

  CL/Win の [マネージャフレーム] - [マシン一覧] - [バックアップ対象のマシン(またはサイト)] - [リクエスト一覧] を確認し、実行中のリクエストが存在しないことを確認してください。

  また、CL/Win の [マネージャフレーム] - [マシン一覧] - [バックアップ対象のマシン(またはサイト)] - [キュー一覧] を確認し、全てのキューの Queued, Running, Waiting, Holding の値が0になっていることを確認してください。

- リモートマシン(他サイト)に対してイベント送信(受信確認ON)を実行していないこと

  CL/Win の [マネージャフレーム] - [マシン一覧] - [バックアップ対象のマシン(またはサイト)] - [イベント一覧] を確認し、表示されるイベントが存在しないことを確認してください。

■バックアップを作成する時は、JobCenterのプロセスは全て停止させてください

- クラスタサイトの領域をバックアップする場合は、対象のクラスタサイトが停止していることを確認してください。

- ローカルサイトの領域をバックアップする場合は、ローカルサイトおよび全てのクラスタサイトが停止していることを確認してください。

17.3.2. イメージバックアップからの復元

バックアップに利用した製品または機能のマニュアル等に従って、イメージバックアップからJobCenter環境を復元してください。

イメージバックアップからの復元によって、JobCenter環境はバックアップ取得時から停止され続けていたことと同じ状態になります。そのためバックアップ作成時点よりも後に予定されていたスケジュール定義や、起動トリガ定義の設定に従って、復元後の初回起動時にトラッカーが一斉に起動します。トラッカの大量起動を抑制するため、以下の設定を確認し、必要に応じて設定を行ってください。定義ファイルを持っていない環境(JobCenter SV)ではこの確認および設定は不要です。

■スケジュールによる過去時刻分のトラッカー投入抑制

- 復元後の初回起動時、JobCenter環境を停止していた期間にスケジュールされていたトラッカが一斉に投入されます。これを抑止したい場合、デーモン設定ファイル（daemon.conf）で設定変更が可能です。

たとえば、再起動時に1日経過したスケジュールを起動しないようにするには、次のように指定します。

```
SCLAUNCHD_FORGET=86400
```

詳細については「5.2.1 Windows/UNIX共通」のSCLAUNCHD_FORGETの項目を参照してください

- 先述の方法で復元後に対処を行なう代わりに、全てのスケジュールを無効化してバックアップを作成し、復元後にスケジュールを有効化することでもトラッカの一斉投入を抑止することができます。

  スケジュールの有効化・無効化については<基本操作ガイド>の「6.2.4 スケジュールの有効/無効を設定する」を参照してください。

■テキストログ監視によるトラッカー投入抑制

- テキストログ監視で指定しているファイルが、バックアップ作成時刻以降に更新されている場合、復元後の初回起動後に監視条件に従ってトラッカが一斉に投入されます。これを抑止したい場合、（daemon.conf）で設定変更が可能です。

  再起動時にテキスト監視機能によるトラッカ起動しないようにするには、次のように指定します。

```
JL_LOGMONITOR_FORGET=ON
```

  詳細については「5.2.1 Windows/UNIX共通」のJL_LOGMONITOR_FORGETの項目を参照してください

- 先述の方法で復元後に対処を行なう代わりに、全てのテキストログ監視を無効化してバックアップを作成し、復元後にテキストログ監視を有効化することでもトラッカの一斉投入を抑止することができます。

  テキストログ監視の有効化・無効化については<基本操作ガイド>の「7.1.2.4 起動トリガ定義の有効/無効を設定する」を参照してください。

17.4. 復元後の整合性確認

連携するJobCenter MG/SV は、お互いに状態を更新して整合性をとりながら動作しています。そのため、連携を行っている一方または一部の JobCenter 環境のみをバックアップから復元すると、他方の環境との状態不整合が発生します。

状態不整合が発生している場合、下記の動作影響がありますので、バックアップからの復元を全環境に対して実施するか、整合性を取るための操作を実施する必要があります。

17.4.1. MG-SV構成のMG(リクエスト投入元サーバー)を復元する場合

本節では、ジョブネットワークやスケジュール等の定義を管理し、リモートマシン(SV)または自サイト自身に対して単位ジョブのリクエストの投入を行っている環境を復元する場合に、連携している環境間の整合性を取るための手順を説明します。

実行中のトラッカーが存在する場合、単位ジョブの実行の際に実行先のサーバー、またはサイトに対してジョブリクエスト投入が行われます。このリクエストは実行先サーバーでの実行完了し、投入元に実行結果が返却されるまで存在します。この実行中リクエストが存在する状況で、投入元サーバー(MG)をバックアップから復元すると、リクエストに与えられる識別子(ID)が重複生成される可能性が有ります。重複したIDのリクエストが投入された場合、単位ジョブはエラー終了となります。

このエラーを回避するために、次の手順で投入済みリクエストの削除を行ってください。

> 本手順は、復元する環境(リクエスト投入元サーバー)がジョブリクエストを投入している全てのSV(リクエスト投入先サーバー)が確認、対処実施の対象になります。

1. MGから投入されているリクエストを確認します。

   コマンドライン、またはCL/Winを利用して確認することができます。

   ■コマンドラインによるリクエストIDの確認

   a. SV(リクエスト投入先サーバー)で、UNIX(Linux)版ではrootユーザ、Windows版ではJobCenter管理者アカウントで qmgrコマンド [注1] を起動してください。

   b. qmgr のプロンプトで、以下のコマンドを入力してください。

   ```
   show long queue
   ```

   c. qmgrを実行した環境に投入されているリクエストが、次のようなフォーマットで出力されます。出力例1行目の Id=12345.example.com がリクエストIDです。リクエストIDのうち、バックアップ対象のマシン(またはサイト)名を含むIDが削除対象になります。該当するリクエストIDが存在しない場合は、リクエスト削除の操作は不要です。

   ```
   Request    1:  Name=JOB1  Id=12345.example.com
   Owner=user1  Priority=31  RUNNING  Pgrp=766
   Created at Mon Mar 29 15:28:09 JST 1993
   Mail = [NONE]
   Per-proc. core file size limit = UNLIMITED <DEFAULT>
   Per-proc. data size limit = UNLIMITED <DEFAULT>
   Per-proc. permanent file size limit = UNLIMITED <DEFAULT>
   Per-proc. memory size limit = UNLIMITED <DEFAULT>
   Per-proc. stack size limit = UNLIMITED <DEFAULT>
   Per-proc. CPU time limit = UNLIMITED <DEFAULT>
   Per-req.  CPU time limit= UNLIMITED <DEFAULT>
   ```

[注1] qmgr コマンドの詳細は <コマンドリファレンス>の「3.13 qmgr 構成管理および運用管理」 を参照してください。

```
Per-req.  temporary file space limit= UNLIMITED <DEFAULT>
Per-req.  process number limit = UNLIMITED <DEFAULT>
Per-req.  physical memory limit = UNLIMITED <DEFAULT>
Execution nice priority = 0 <DEFAULT>
Standard-error access mode = SPOOL
Standard-error name = netware:/home/user1/STDIN.e183
Standard-output access mode = SPOOL
Standard-output name = netware:/home/user1/STDIN.o183
Shell = /bin/sh
Umask =  22
```

■CL/Winを利用するリクエストIDの確認

a. SV(リクエスト投入先サーバー)にCL/Win でログインし、[マネージャフレーム] - [マシン一覧] - [バックアップ対象のマシン(またはサイト)] - [リクエスト一覧] を表示してください。

b. [リクエストID]列に表示されるIDのうち、バックアップ対象のマシン(またはサイト)名を含むIDが削除対象です。該当するリクエストIDが存在しない場合は、リクエスト削除の操作は不要です。

2. MGから投入されているリクエストを削除します。

a. SV(リクエスト投入先サーバー)で、UNIX(Linux)版ではrootユーザ、Windows版ではJobCenter管理者アカウントで qmgr コマンドを起動してください。

b. qmgr のプロンプトで、以下のコマンドを入力します。コマンド例中の $request_id は、事前に確認したリクエストIDに置き換えて入力してください。

```
delete request $request_id
```

3. リクエストの削除が成功している事を確認します。

手順(1)と同じ手順で、リクエストIDの存在を確認し、手順(2)で指定したリクエストが削除されていることを確認します。

リクエストを処理しているタイミングによっては、手順(2)の操作でリクエスト削除を受け付けることができない場合があります。リクエストが削除されていない場合は次の手順(4)を実施してください。

4. リクエストを強制的に削除します。

qmgr　コマンドによるリクエスト削除が処理されない場合、リクエストの強制的な削除を行なう必要があります。

> ⚠ 本操作は特定のジョブリクエストを削除するのではなく、特定のキューまたは全てのキュー上のリクエストを全て削除する操作です。復元する環境以外から投入されているジョブリクエストが存在する場合、そのリクエストも削除されます。
>
> 復元する環境からのジョブリクエストのみが残っている状況であることを確認の上で実施してください。

a. SV(リクエスト投入先サーバー)で、UNIX(Linux)版ではrootユーザ、Windows版ではJobCenter管理者アカウントで qmgr コマンドを起動してください。

b. qmgr のプロンプトで、以下のコマンドを入力します。

```
set queue reboot_mode = purge
```

復元する環境からのジョブリクエストが特定のキュー上のみに残っている場合、キューを指定して設定することもできます。

```
set queue reboot_mode = purge $queue_name
```

c. JobCenter環境を再起動してください。

d. JobCenter再起動後、手順(a)および(b)と同じ手順でキューにジョブリクエストが残っていないことを確認してください。

17.4.2. MG-SV構成のSV(リクエスト投入先サーバー)を復元する場合

本節では、単位ジョブのリクエストが投入される環境を復元する場合に、連携している環境間の整合性を取るための手順を説明します。

MG(リクエスト投入元サーバー)ではジョブリクエストをSV(リクエスト投入先サーバー)に転送した後、実行結果の返却を待ち合わせています。ジョブリクエストを実行している状態のSVを復元すると、実行結果の返却が行われないため、MGのトラッカー上では単位ジョブが[実行中]から進行しない状態になります。

トラッカーの状態を進行させるために、次の手順で喪失したジョブリクエストの特定と対処を行ってください。

> 本手順は、復元する環境(リクエスト投入先サーバー)にジョブリクエストを投入している全てのMG(リクエスト投入元サーバー)が確認、対処実施の対象になります。

実行中の単位ジョブ部品の一覧を確認します。

1. MG(リクエスト投入元サーバー)で、jnwsummary コマンド [注] を次のコマンドラインオプションを指定し実行してください。

```
jnwsummary -j run -p format=csv -p style=1
```

2. 次の形式でトラッカーの一覧が出力されます。

```
#TYPE=SUMMARY
jnwname,jnwdate,user,machine,starttime,endtime,amounttime,(中略),otherdoneskip,othererror
NewJnw,20180330014150,nsumsmgr,example.com,2018/03/30 10:41:50,,0,2018/03/30 10:41:50,
2018/03/30 10:42:50,run,2,1,0,0,1,0,0,0,1,1,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

#TYPE=JNW
type,parentjnw,jnwdate,name,starttime,endtime,amounttime,(中略),otherdoneskip,othererror
JNW,"",20180330014150,NewJnw,2018/03/30 10:41:50,,0,2018/03/30 10:41:50,2018/03/30 10:42:50,
run,1,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

#TYPE=UNITJOB
type,parentjnw,jnwdate,name,starttime,endtime,amounttime,expstarttime,expendtime,status,
exitcode,errcode,requestid,machine
UNITJOB,NewJnw,20180330014150,JOB1,,,0,2018/03/30 10:41:50,2018/03/30 10:42:50,wait,0,0,-,-
```

3. 先頭が "UNITJOB" で開始する行の、第10列目の要素が "run" または "submit" になっているものが実行中の単位ジョブです。

   実行中の単位ジョブのトラッカID(第2列目および第3列目)と部品名(第4列目) からトラッカー部品の設定を確認し、復元する環境に投入している場合は、この部品をMG(リクエスト投入元サーバー)に接続したCL/Winのトラッカ画面よりコントロール解除してください。

[注] jnwsummary コマンドの詳細は <コマンドリファレンス>の「3.1 jnwsummary ジョブネットワーク実行実績を出力」 を参照してください。

その後、MG(リクエスト投入元サーバー)で、trfdelete コマンド [注] を実行し、トラッキングファイルを削除してください。

17.4.3. イベント送信部品、イベント受信部品による連携を行っている環境を復元する場合

イベントによる連携を行っている環境を復元すると、連携先のJobCenter環境ではイベントの待ち合わせ状態が不整合となります。次の手順で、イベント待ち合わせの状況を確認し、対処を行ってください。

> ⚠ 本手順は、復元する環境とイベント連携している全てのJobCenter環境が確認、対処実施の対象になります。

復元する環境に向けて送信したイベント、および復元する環境から送信されているイベントの一覧を確認します。

1. SV(リクエスト投入先サーバー)で、jnwevtmgr -l コマンド [注] を起動してください。次の形式でイベントの一覧が出力されます。

```
*** EventWaitList ***
****      TRACKER-ID.yyyymmddhhmmss OBJECTNAME   HOSTNAME    EVENTID   TIMEOUT
1)          Event.20180330000000  EVENTRCV1          -     -               -
*** EventSendList ***
****      TRACKER-ID.yyyymmddhhmmss OBJECTNAME   HOSTNAME    EVENTID   TIMEOUT
1)          Event.20180330000000  EVENTSND1    example.com EventTest       -
*** EventHoldList ***
****      TRACKER-ID.yyyymmddhhmmss OBJECTNAME   HOSTNAME    EVENTID   TIMEOUT
3)          Event.20180330000000  EVENTSND2    example.com EventTest       -
```

   このイベント一覧で、[HOSTNAME]列　が復元する環境のサイト名になっているイベントが確認対象です。該当するイベントが存在しない場合は、以降の確認および対処は不要です。

2. EventWaitList (受信待ちをしている受信部品一覧)について確認、対処します。

   jnwevtmgr コマンドを実行した環境で実行中の、イベント受信部品が一覧されています。これらのイベント受信部品宛のイベントは、復元するJobCenter環境からイベント送信が行われるまで待ち合わせを行い、イベントが送信されない場合はタイムアウトになります。トラッカIDを確認し、当該のイベント受信部品についてスキップや強制停止行なうか、または復元する環境からイベントが送信されるまで待つなどの判断を行ってください。

3. EventSendList (受信確認で待ち合わせている送信部品一覧)について確認、対処します。

   jnwevtmgr コマンドを実行した環境で実行中の、イベント送信部品が一覧されています。これらのイベント送信部品には受信確認が設定されており、復元する環境のイベント受信部品によって受信されるまで待ち合わせを行いますが、当該のイベントは環境の復元によって失われているため、このイベント送信部品に受信確認が通知されることはありません。トラッカIDを確認し、当該のイベント送信部品についてスキップや強制停止を行なうなどの判断を行ってください。

4. EventHoldList (保持しているイベント一覧) について確認、対処します。

   jnwevtmgr コマンドを実行した環境に対して送信されたイベントで、イベント受信部品が受信していないイベントが一覧されています。復旧後に改めて同じイベントを送信する可能性がある場合(例:ジョブネットワークを改めて起動する場合)など、イベントによる待ち合わせ動作に影響がないか確認し、不必要なイベントの場合は jnwevtmgr -d コマンドにより削除を行ってください。

[注] trfdelete コマンドの詳細は <コマンドリファレンス>の「3.26 trfdelete トラッキングファイルの削除」 を参照してください。
[注] jnwevtmgr コマンドの詳細は <コマンドリファレンス>の「3.6 jnwevtmgr イベントの確認と削除」 を参照してください。

18. jcresの使用方法

この章では、JobCenter MG/SV専用のHTTPデーモンjcresの使用方法について説明します。

jcresコマンドの詳細は、<コマンドリファレンス>の「3.30 jcres JobCenter MG/SV専用のHTTPデーモン」を参照してください。

18.1. 起動方法

jcresを起動するには、以下のコマンドを実行してください。

■UNIX版

```
/usr/lib/nqs/gui/tools/jcres
```

■Windows版

```
%InstallDirectory%\tools\jcres.bat
```

バックグラウンドで起動したい場合は、以下のように実行してください。

■UNIX版

```
/usr/lib/nqs/gui/tools/jcres &
```

■Windows版

jcresを起動するvbsスクリプトを作成します。

JobCenterのインストールディレクトリを「C:\JobCenter\SV」とした場合のvbsスクリプトのサンプルは以下のとおりです。

インストールディレクトリが異なる場合は、「C:\JobCenter\SV」の部分を適切なパスに変更してください。

```
CreateObject("WScript.Shell").Run "C:\JobCenter\SV\tools\jcres.bat",0
```

vbsスクリプトのファイル名は任意ですが、拡張子を.vbsで保存してください。(例:jcres.vbs)

作成したvbsスクリプトを任意のフォルダーに保存して実行してください。

UNIX版ではroot、Windows版ではJobCenter管理者ユーザで実行してください。

18.2. 停止方法

jcresを停止するには、以下のコマンドを実行してください。

■UNIX版

```
/usr/lib/nqs/gui/tools/jcres -e
```

■Windows版

```
%InstallDirectory%\tools\jcres.bat -e
```

UNIX版ではroot、Windows版ではJobCenter管理者ユーザで実行してください。

18.3. 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。

設定ファイルの詳細は<コマンドリファレンス>の「3.30.3 設定ファイル」を参照してください。

18.4. ブラウザからMG-SV間の疎通確認を行う

18.4.1. 使用方法

ブラウザからMG-SV間の疎通確認を行う方法について説明します。

疎通確認はjcresのAPIであるjc_pingを使用して行います。APIの詳細は<コマンドリファレンス>の「3.30.6.2 jc_ping」を参照してください。

以下の図はMGがクラスタサイトの場合の疎通確認のイメージ図です。

クライアントマシンでMGからSVに対する疎通確認を行うHTTPリクエストを運用系マシンのjcresに発行すると、jcresを介して運用系マシンからjc_pingコマンドが実行されます。jc_pingコマンドの実行に成功すると、MGからSVに対する疎通確認の結果がクライアントマシンに返却されます。同様の手順で、SVからMGに対する疎通確認を行うHTTPリクエストをSVマシンのjcresに発行し、双方向の疎通確認を行います。

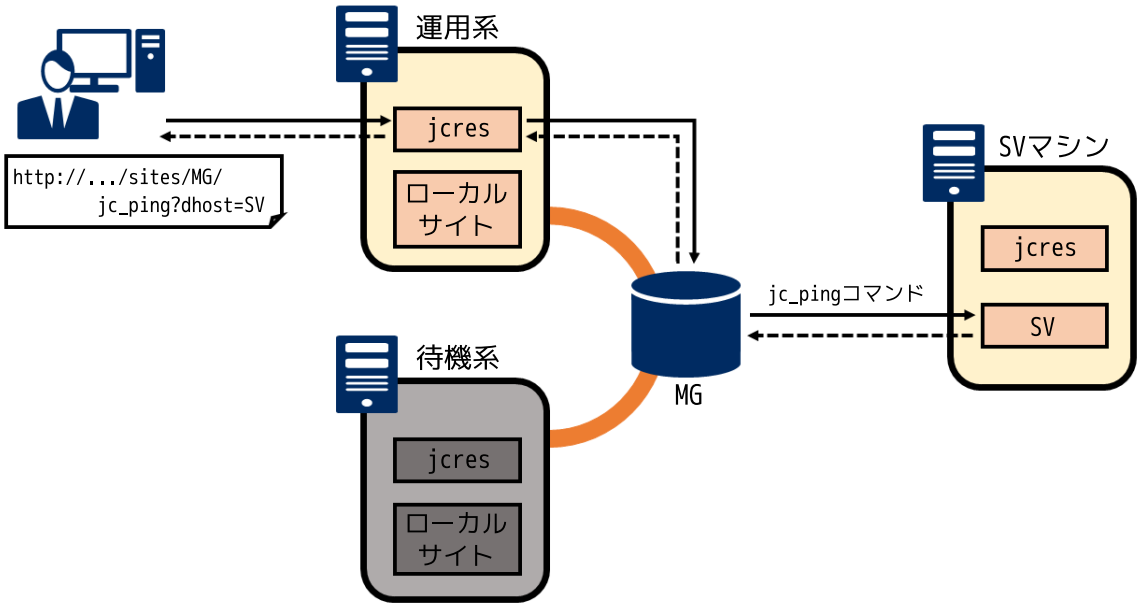

図18.1 MGからSVに対する疎通確認の構成図

説明内のサンプルで使用している値は、以下の環境を想定しています。上図と合わせて確認してください。

名称	ホスト名 (IPアドレス)
運用系のjcres	mgjcres
SVマシンのjcres	svjcres
MG	mgsite (192.168.1.1)
SV	svsite (192.168.1.2)

1. jcresの起動

   疎通確認を行う各マシンでjcresを起動してください。jcresの起動方法は「18.1 起動方法」を参照してください。

2. MGからSVへの疎通確認

   まず、MGからSVに対する疎通確認を行います。ブラウザを起動して以下のURLを入力してください。

■MGがローカルサイトの場合

```
http://<MGマシンのjcresのホスト名またはIPアドレス>:<MGマシンのjcresのポート番号>/jc_ping?
dhost=<SVのサイト名>&af=<アドレスファミリ>&ctw=<タイムアウト時間>
```

■MGがクラスタサイトの場合

```
http://<MGマシンのjcresのホスト名またはIPアドレス>:<MGマシンのjcresのポート番号>/sites/<MGのサイト名>/jc_ping?dhost=<SVのサイト名>&af=<アドレスファミリ>&ctw=<タイムアウト時間>
```

■jcresのIPアドレス及びポート番号は設定ファイルで変更することができます。初期値はIPアドレスが0.0.0.0、ポート番号が50080で起動します。

■アドレスファミリ、タイムアウト時間は省略可能です。省略した場合、アドレスファミリはIPv4アドレス、タイムアウト時間は30秒となります。

図18.1「MGからSVに対する疎通確認の構成図」の環境を例にすると、MGはクラスタサイトになりますので、URLのサンプルは以下のようになります。

```
http://mgjcres:50080/sites/mgsite/jc_ping?dhost=svsite
```

APIの実行に成功すると、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<コマンドリファレンス>の「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。返却された結果に「<SVのホスト名> is alive」と表示されればMGからSVに対する疎通確認は成功です。

疎通確認の結果のサンプルは以下のとおりです。

```
results :
 destination_host = svsite
 destination_ip   = 192.168.1.2
 source_host      = mgsite
 source_ip        = 192.168.1.1

connect :
 nqs       : [OK]
 jccombase : [OK]
 jcevent   : [OK]

svsite is alive.
```

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. SVからMGへの疎通確認

次に、SVからMGに対する疎通確認を行います。「MGからSVへの疎通確認」と同様の手順で実施します。

■SVがローカルサイトの場合

```
http://<SVマシンのjcresのホスト名またはIPアドレス>:<SVマシンのjcresのポート番号>/jc_ping?
dhost=<MGのサイト名>&af=<アドレスファミリ>&ctw=<タイムアウト時間>
```

■SVがクラスタサイトの場合

```
http://<SVマシンのjcresのホスト名またはIPアドレス>:<SVマシンのjcresのポート番号>/sites/<SVのサイト名>/jc_ping?dhost=<MGのサイト名>&af=<アドレスファミリ>&ctw=<タイムアウト時間>
```

図18.1「MGからSVに対する疎通確認の構成図」の環境を例にすると、SVはローカルサイトになりますので、URLのサンプルは以下のようになります。

```
http://svjcres:50080/jc_ping?dhost=mgsite
```

返却された結果に「<MGのホスト名> is alive」と表示されればSVからMGに対する疎通確認は成功です。

4. jcresの停止

jcresが不要な場合はjcresを停止します。jcresを起動した各マシンでjcresを停止してください。jcresの停止方法は「18.2 停止方法」を参照してください。

18.4.2. トラブルシュート

MG-SV間の疎通確認に失敗した場合のトラブルシュートを、以下の図の環境を例にして説明します。

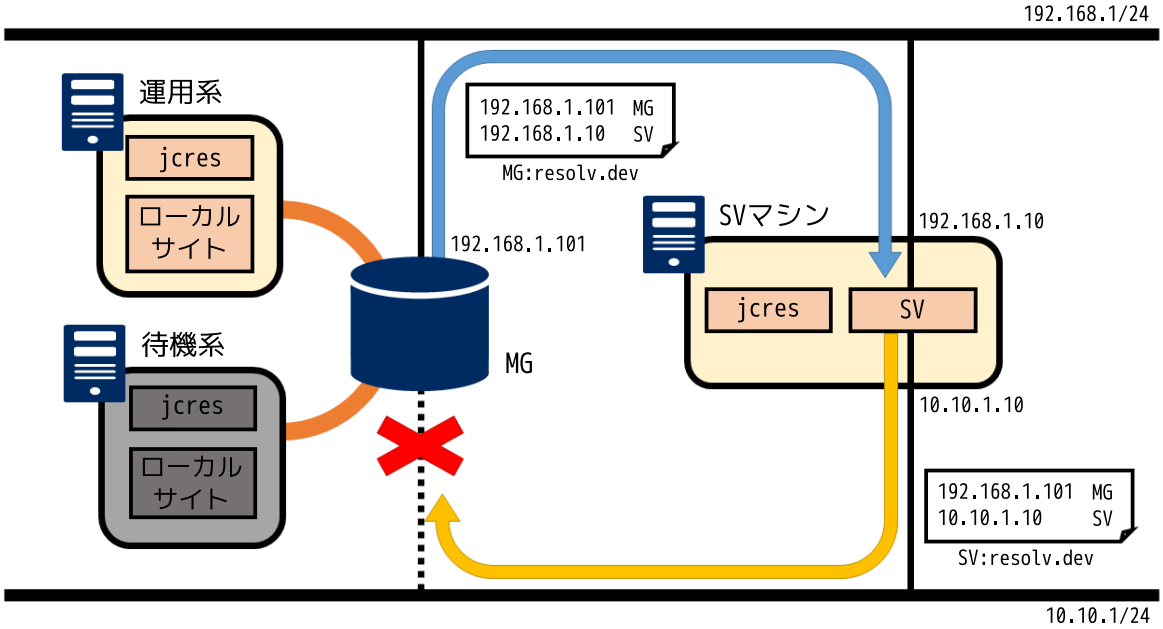

図18.2 MG-SV間の疎通確認失敗の構成図

前提条件は以下のとおりです。

■MGはWindows版JobCenterのクラスタサイト、SVはWindows版JobCenterのローカルサイトとする。

■MG、SVともに各サイトの名前解決はresolv.defで行う。

■MGは192.168.1/24と10.10.1/24のネットワークに所属しているが、192.168.1/24のネットワークのみで待ち合わせている。

■SVは192.168.1/24と10.10.1/24のネットワークに所属し、192.168.1/24と10.10.1/24の両方のネットワークで待ち合わせている。

■MG、SVともに192.168.1/24と10.10.1/24間のIPフォワーディングは行わない。

説明内のサンプルで使用している値は、以下の環境を想定しています。上図と合わせて確認してください。

表18.1 MGの環境

サイト名	mgsite
待ち受けているIPアドレス	192.168.1.101

resolv.defの設定	192.168.1.101 mgsite 192.168.1.10 svsite

表18.2 SVの環境

サイト名	svsite
待ち受けているIPアドレス	10.10.1.10, 192.168.1.10
resolv.defの設定	192.168.1.101 mgsite 10.10.1.10 svsite

1. MGからSVへの疎通確認

   まず、MGからSVに対する疎通確認を行います。確認手順は「18.4.1 使用方法」を参照してください。

   今回の例では、MGからSVに対するネットワーク構成に問題はありませんので、疎通確認の結果は以下のように表示されます。

```
results :
 destination_host = svsite
 destination_ip   = 192.168.1.10
 source_host      = mgsite
 source_ip        = 192.168.1.101

connect :
 nqs       : [OK]
 jccombase : [OK]
 jcevent   : [OK]

svsite is alive.
```

2. SVからMGへの疎通確認

   次に、SVからMGに対する疎通確認を行います。確認手順は「18.4.1 使用方法」を参照してください。

   疎通確認の結果は失敗となり、以下のように表示されます。

```
results :
 destination_host = mgsite
 destination_ip   = 192.168.1.101
 source_host      = svsite
 source_ip        = 10.10.1.10

connect :
 nqs       : [NG]
   Unable to make connection with NQS daemon at transaction peer. Retry later. (
E$nqs_ping(): failed to connect.(errno=10051, r-adr[192.168.1.101] r-port[607] l-adr[10.10.1.10]
 l-port[996]))
 jccombase : [OK]
 jcevent   : [OK]

Communication Failure.
```

お客様の環境で上記のようなresultsやconnectの結果が表示されずにエラーメッセージのみ表示された場合、疎通確認より前の段階で失敗しています。例えば、URLのパラメータに誤りがあったり、MGまたはSVの名前解決に失敗した場合が考えられます。

<コマンドリファレンス>の「6.3.6　主要メッセージ」を参考に問題を解決した後、再度SVからMGへの疎通確認を実施してください。

3. 問題の調査

疎通確認に失敗した場合、まずconnectの結果を確認します。NQSの疎通確認に失敗しているため、NQSのポート番号(デフォルト607)をファイアウォール等で遮断していないか確認します。今回はファイアウォールの設定に問題はなかったと仮定します。

次にresultsの結果を確認します。source_ipが10.10.1.10、destination_ipが192.168.1.101となっているため、SVのIPアドレス10.10.1.10からMGのIPアドレス192.168.1.101に対して疎通確認を実施したことが分かります。これが疎通確認に失敗した原因です。

JobCenterのプロトコルについて、jccombase及びjceventはINADDR_ANYで通信を行いますが、Windows版JobCenterのNQSはデフォルトの設定の場合、名前解決で自サイト名から得られたIPアドレスを用いて通信を行う仕様となっています。詳細は「5.3　複数IPアドレス使用時のdaemon.conf設定について」を参照してください。

今回の環境で正しく通信できるようにするには、SVの自サイトの名前解決を192.168.1.10となるように設定するか、daemon.confに「NQSDAEMON_OPT=-x wildcard=ON」の設定をする必要があります。

4. 問題の解決

a. SVの自サイトの名前解決を192.168.1.10となるように設定する場合

SVの自サイトの名前解決が192.168.1.10となるように設定します。今回の例では、resolv.devで名前解決を行っているので、SVのresolv.defを以下のように修正します。

resolv.defの設定	192.168.1.101 mgsite 192.168.1.10 svsite

b. daemon.confに「NQSDAEMON_OPT=-x wildcard=ON」の設定をする場合

NQSをINADDR_ANYで通信するように設定します。SVのdaemon.confに以下の設定を追加します。

```
NQSDAEMON_OPT=-x wildcard=ON
```

また、SVの192.168.1/24のNICに複数のIPアドレスを設定している場合は、MGのdaemon.confに「ipcheck=OFF」の設定が必要です。

```
ipcheck=OFF
```

「NQSDAEMON_OPT=-x wildcard=ON」の詳細は「5.3 複数IPアドレス使用時のdaemon.conf設定について」を参照してください。

daemon.confの設定を反映するには、JobCenterを再起動する必要があります。

5. 修正確認

再度SVからMGに対する疎通確認を行い、問題が解決しているか確認します。解決方法によって、疎通確認の結果は以下のようになります。

a. SVの自サイトの名前解決を192.168.1.10となるように設定した場合

```
results :
 destination_host = mgsite
```

```
 destination_ip   = 192.168.1.101
 source_host      = svsite
 source_ip        = 192.168.1.10

connect :
 nqs       : [OK]
 jccombase : [OK]
 jcevent   : [OK]

mgsite is alive.
```

connectのNQSの結果が「OK」になっていることが確認できます。

また、resultsのsource_ipの値から、SVの名前解決で自サイト名から得られたIPアドレスが192.168.1.10に修正されたことが確認できます。

b. daemon.confに「NQSDAEMON_OPT=-x wildcard=ON」の設定をした場合

```
results :
 destination_host = mgsite
 destination_ip   = 192.168.1.101
 source_host      = svsite
 source_ip        = 10.10.1.10

connect :
 nqs       : [OK]
   wildcard=ON
 jccombase : [OK]
 jcevent   : [OK]

mgsite is alive.
```

connectのNQSの結果に「wildcard=ON」のメッセージが表示され、結果が「OK」になっていることが確認できます。

18.5. 自動起動・停止の設定方法

OSの起動・停止でjcresを自動で起動・停止する設定方法の例を紹介します。

18.5.1. Red Hat Enterprise Linux 6

Linuxのinitプロセスを利用して自動起動・停止を行う設定例を説明します。

各作業はrootユーザで実施してください。

18.5.1.1. 設定方法

1. rcスクリプトの作成

   jcresのrcスクリプトを作成します。

   rcスクリプトのサンプルは以下のとおりです。

```
#!/bin/sh
#
# JobCenter jcres statr-up
#
# For Linux chkconfig
# chkconfig: 2345 99 00
# description: JobCenter jcres statr-up

case $1 in
start)
    if [ ! -x /usr/lib/nqs/gui/tools/jcres ]
    then
        exit 0
    fi
    touch /var/lock/subsys/jcres 1>&- 2>&-
    /usr/lib/nqs/gui/tools/jcres &
    exit $?
    ;;

stop)
    if [ ! -x /usr/lib/nqs/gui/tools/jcres ]
    then
        exit 0
    fi
    rm -f /var/lock/subsys/jcres 1>&- 2>&-

    ps -ef|grep -v grep|grep jcres > /dev/null
    if [ $? -eq 0 ] ; then
        /usr/lib/nqs/gui/tools/jcres -e
        exit $?
    fi
    exit 0
    ;;

*)
    exit 1
esac
```

rcスクリプトのファイル名は任意です。

rcスクリプトの編集が完了したら、以下のコマンドでファイルの権限を設定します。

```
/bin/chmod 755 <rcスクリプトのパス>
```

最後に、作成したrcスクリプトを/etc/init.d/配下に保存してください。

2. サービスの登録

以下のコマンドでjcresのサービスを登録します。

```
/sbin/chkconfig --add <rcスクリプト名>
```

以下のコマンドでjcresのサービスが登録されていることを確認してください。

```
/sbin/chkconfig --list <rcスクリプト名>
```

以下のように表示されれば、正常にサービスが登録されています。

```
<rcスクリプト名>          0:off    1:off    2:on     3:on     4:on     5:on     6:off
```

18.5.1.2. 設定削除方法

1. サービスの削除

以下のコマンドでjcresのサービスを削除します。

```
/sbin/chkconfig --del <rcスクリプト名>
```

2. rcスクリプトの削除

「18.5.1.1 設定方法」で作成した/etc/init.d/配下のrcスクリプトを削除します。

18.5.2. Red Hat Enterprise Linux 7

Linuxのsystemdを利用して自動起動・停止を行う設定例を説明します。

各作業はrootユーザで実施してください。

18.5.2.1. 設定方法

1. rcスクリプトの作成

jcresのrcスクリプトを作成します。

rcスクリプトのサンプルは以下のとおりです。

```
#!/bin/sh
#
# JobCenter jcres statr-up
#
# For Linux chkconfig
# chkconfig: 2345 99 00
# description: JobCenter jcres statr-up
```

```
case $1 in
start)
    if [ ! -x /usr/lib/nqs/gui/tools/jcres ]
    then
        exit 0
    fi
    touch /var/lock/subsys/jcres 1>&- 2>&-
    /usr/lib/nqs/gui/tools/jcres &
    exit $?
    ;;

stop)
    if [ ! -x /usr/lib/nqs/gui/tools/jcres ]
    then
        exit 0
    fi
    rm -f /var/lock/subsys/jcres 1>&- 2>&-

    ps -ef|grep -v grep|grep jcres > /dev/null
    if [ $? -eq 0 ] ; then
        /usr/lib/nqs/gui/tools/jcres -e
        exit $?
    fi
    exit 0
    ;;

*)
    exit 1
esac
```

rcスクリプトのファイル名は任意です。

rcスクリプトの編集が完了したら、以下のコマンドでファイルの権限を設定します。

```
/bin/chmod 755 <rcスクリプトのパス>
```

最後に、作成したrcスクリプトを/etc/init.d/配下に保存してください。

2. serviceファイルの作成

jcresのserviceファイルを作成します。

rcスクリプト名をjcresdとした場合のserviceファイルのサンプルは以下のとおりです。rcスクリプト名が異なる場合は、ExecStartとExecStopの「/etc/init.d/jcresd」の部分を適切なパスに変更してください。

```
[Unit]
Description=JobCenter jcres
Requires=network-online.target
After=network-online.target syslog.target

[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/etc/init.d/jcresd start
ExecStop=/etc/init.d/jcresd stop
```

```
[Install]
WantedBy=multi-user.target
```

serviceファイルのファイル名は任意ですが、拡張子を.serviceで保存してください。
(例:jcres.service)

上記の例では、OSのネットワークを管理するサービスが有効である必要があります。例えばNetworkManagerを利用している環境の場合、以下のサービスを有効にしてください。

■NetworkManager-wait-online.service

■NetworkManager.service

serviceファイルの編集が完了したら、以下のコマンドでファイルの権限を設定します。

```
/bin/chmod 644 <serviceファイルのパス>
```

最後に、作成したserviceファイルを/usr/lib/systemd/system配下に保存してください。

3. 自動起動・停止の有効化

以下のコマンドでjcresの自動起動・停止を有効化します。

```
/bin/systemctl enable <serviceファイル名>
```

4. サービスの起動

以下のコマンドでjcresのサービスを起動します。

```
/bin/systemctl start <serviceファイル名>
```

以下のコマンドでjcresのサービスが起動(active)していることを確認してください。

```
/bin/systemctl status <serviceファイル名>
```

rcスクリプト名がjcresd、serviceファイル名がjcres.serviceとした場合の出力結果のサンプルは以下のとおりです。

```
jcres.service - JobCenter jcres
   Loaded: loaded (/usr/lib/systemd/system/jcres.service; enabled)
   Active: active (exited) since 水 2015-12-02 16:11:30 JST; 1min 7s ago
  Process: 5033 ExecStart=/etc/init.d/jcresd start (code=exited, status=0/SUCCESS)
 Main PID: 5033 (code=exited, status=0/SUCCESS)

12月 02 16:11:30 hostname systemd[1]: Started JobCenter jcres.
```

18.5.2.2. 設定削除方法

1. 自動起動・停止の無効化

以下のコマンドでjcresの自動起動・停止を無効化します。

```
/bin/systemctl disable <serviceファイル名>
```

2. rcスクリプト及びserviceファイルの削除

「18.5.2.1　設定方法」で作成した/etc/init.d/配下のrcスクリプト及び/usr/lib/systemd/system配下のserviceファイルを削除します。

3. systemdのリロード

   以下のコマンドでsystemdのserviceファイルを再読み込みします。

```
/bin/systemctl daemon-reload
```

> systemdのリロードを行わなかった場合、systemctlコマンドでjcresのサービスのステータスを確認すると「Warning: Unit file changed on disk, 'systemctl daemon-reload' recommended.」という警告が表示されます。

18.5.3. Windows Server 2012 R2

Windowsのタスクスケジューラを利用して自動起動・停止を行う設定例を説明します。

18.5.3.1. 設定方法

1. vbsスクリプトの作成

   jcresを起動するvbsスクリプトを作成します。

   JobCenterのインストールディレクトリを「C:\JobCenter\SV」とした場合のvbsスクリプトのサンプルは以下のとおりです。インストールディレクトリが異なる場合は、「C:\JobCenter\SV」の部分を適切なパスに変更してください。

```
CreateObject("WScript.Shell").Run "C:\JobCenter\SV\tools\jcres.bat",0
```

> vbsスクリプトのファイル名は任意ですが、拡張子を.vbsで保存してください。(例:jcres.vbs)

   最後に、作成したvbsスクリプトを任意のフォルダーに保存してください。本説明の例では、「C:\JobCenter\SV\tools\jcres.vbs」として保存しています。

2. タスクスケジューラの起動

   コントロールパネルからタスクスケジューラを起動します。

   コントロールパネルをカテゴリ表示に設定している場合、[コントロールパネル] - [システムとセキュリティ] - [管理ツール] - [タスクスケジューラ]から起動できます。

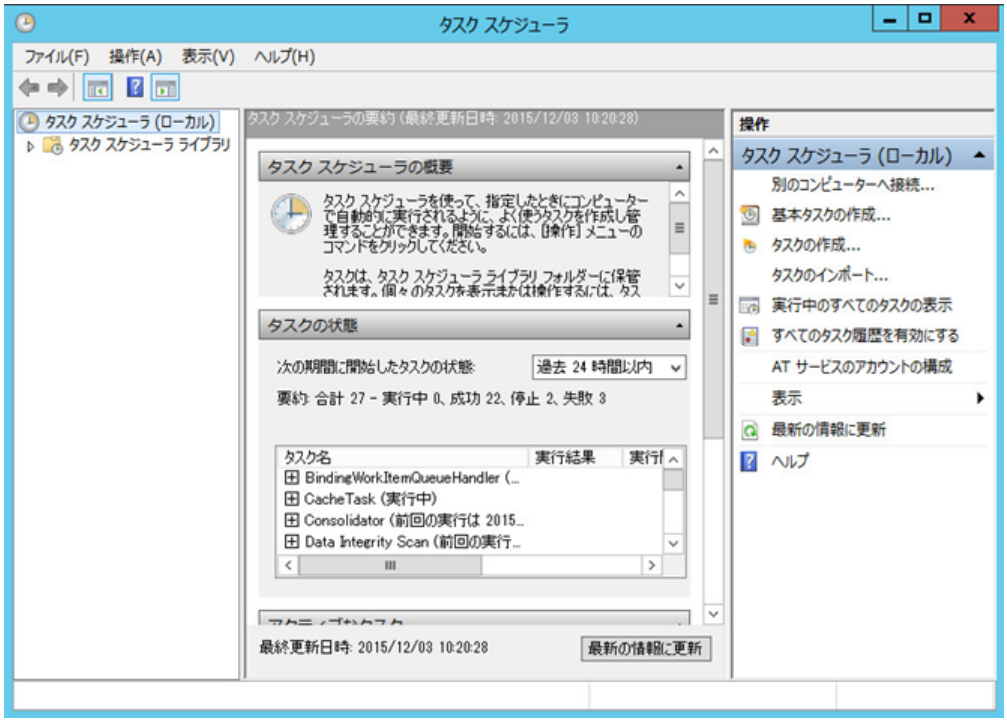

図18.3 タスクスケジューラ

3. タスクスケジューラの設定

a. 左ペインの[タスクスケジューラライブラリ]を右クリックして[新しいフォルダー]をクリックします。

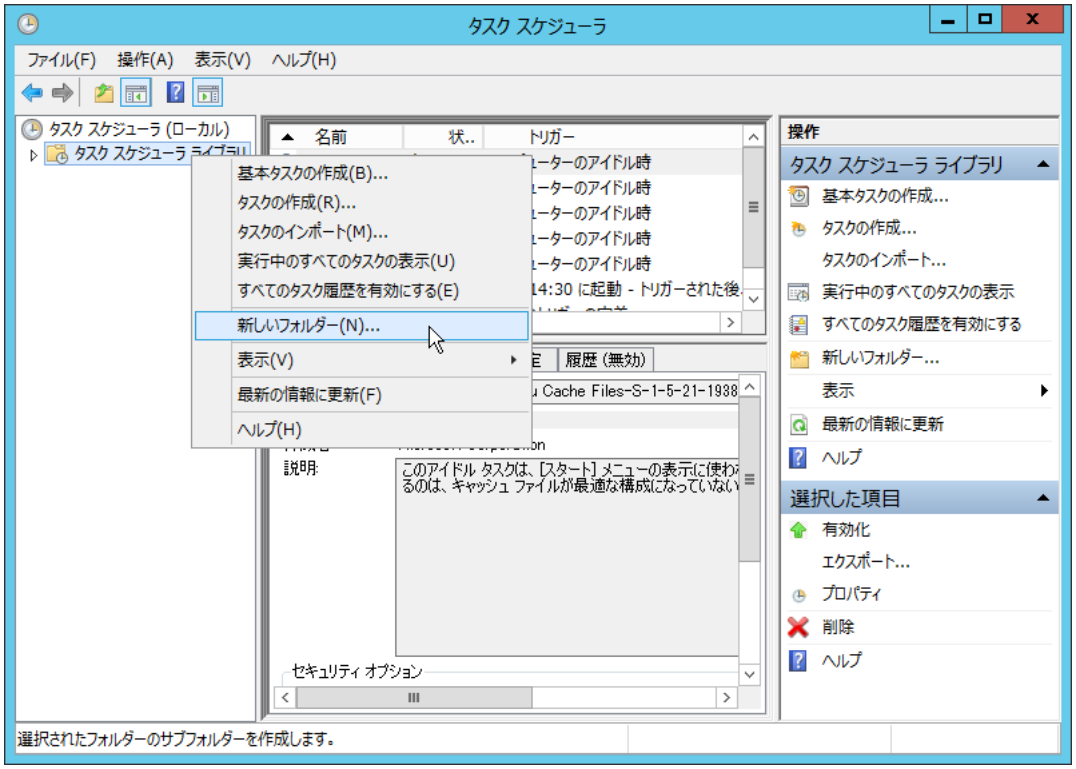

図18.4 タスクスケジューラの[新しいフォルダー]

[新しいフォルダー名の入力]ダイアログが表示されるので、任意のフォルダー名を指定してください。本説明の例では、jcresという名前を指定しています。

b. 左ペインの作成したフォルダーを右クリックして[タスクの作成]をクリックします。

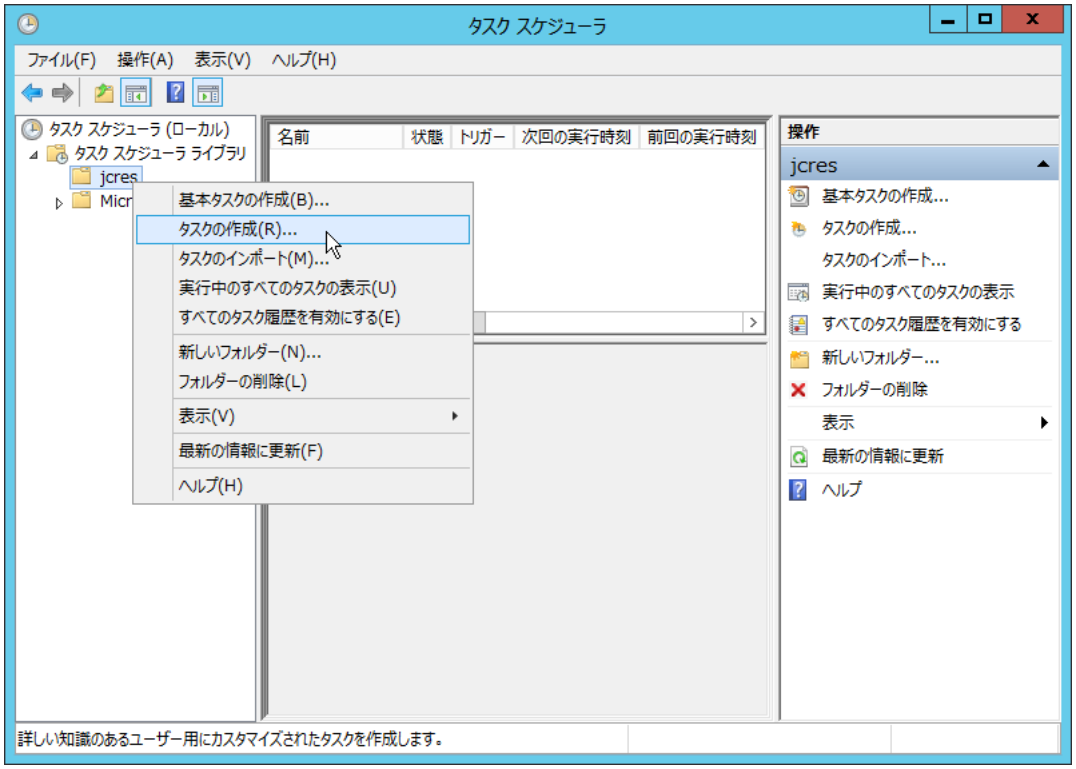

図18.5 タスクスケジューラの[タスクの作成]

c. [全般]タブの設定をします。

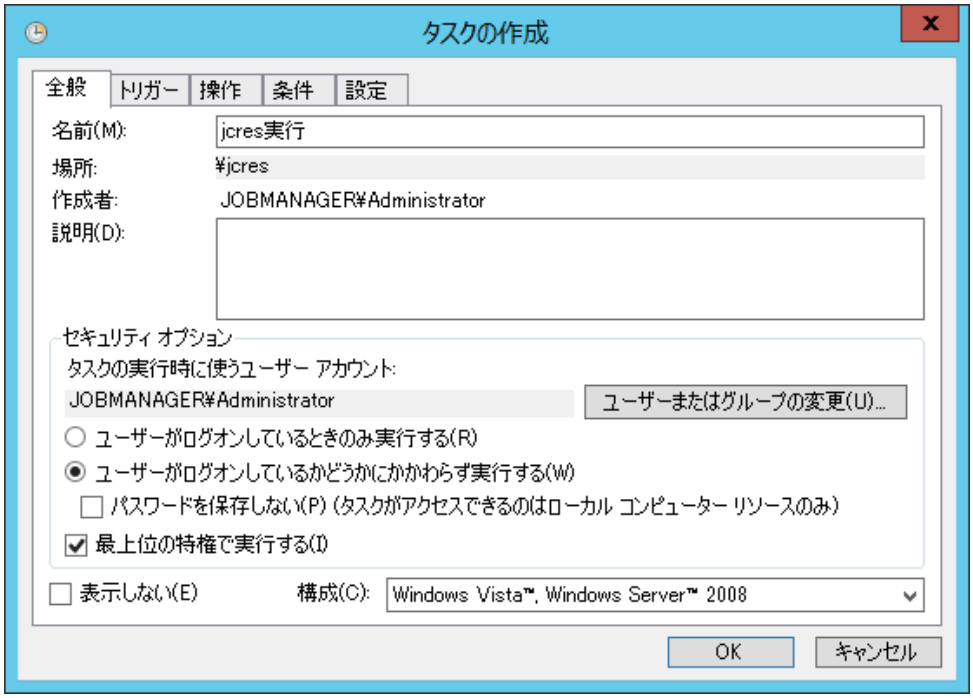

図18.6 タスクスケジューラの[全般]タブ

■[名前]に任意の名前を指定します。本説明の例では「jcres実行」を指定しています。

■[タスクの実行時に使うユーザアカウント]にJobCenter管理者を指定します。

■[ユーザがログオンしているかどうかにかかわらず実行する]を選択します。

■[最上位の特権で実行する]にチェックします。

d. [トリガー]タブの設定をします。

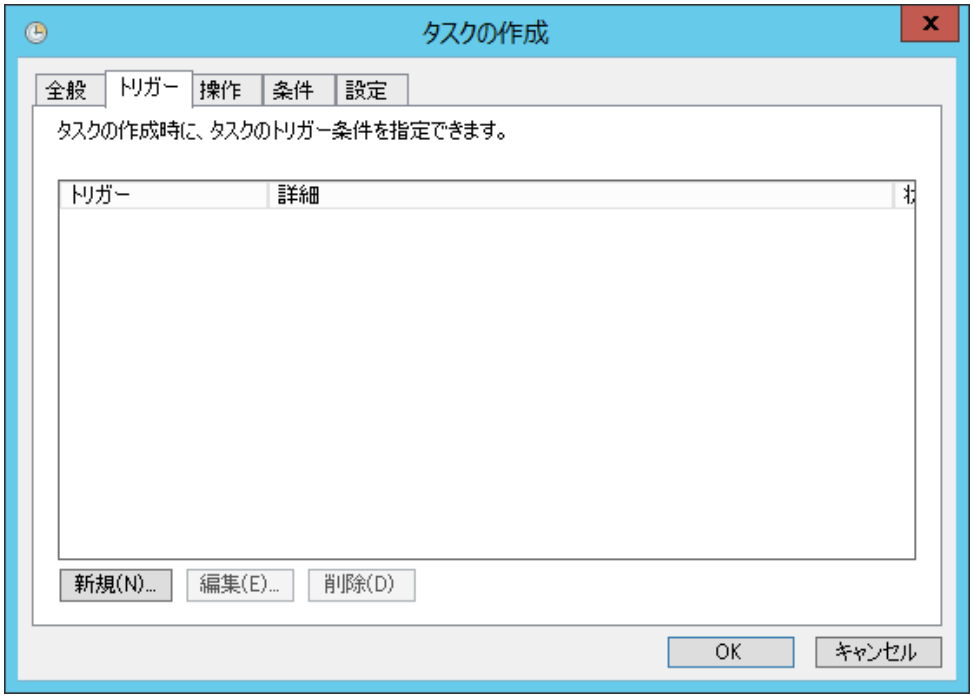

図18.7 タスクスケジューラの[トリガー]タブ

[トリガー]タブの[新規]ボタンをクリックして[新しいトリガー]ダイアログを表示します。

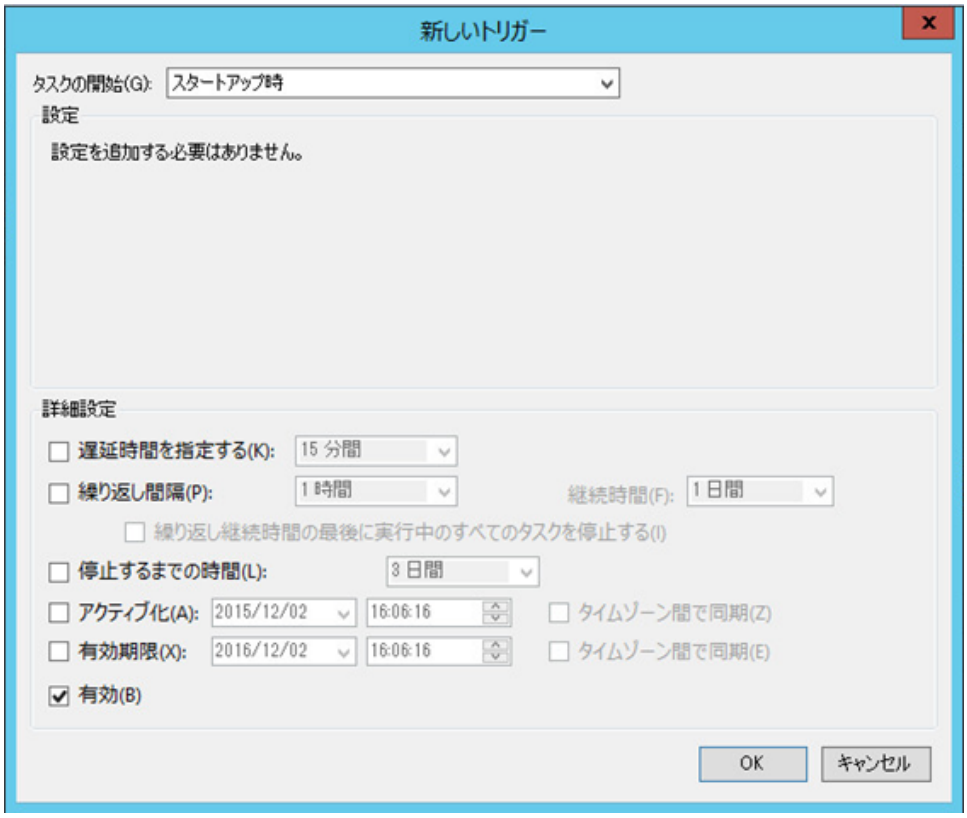

図18.8 タスクスケジューラの[新しいトリガー]ダイアログ

■[タスクの開始]を[スタートアップ時]に変更します。

設定が完了したら[OK]ボタンをクリックしてください。一覧にスタートアップ時のトリガーが登録されます。

e. [操作]タブの設定をします。

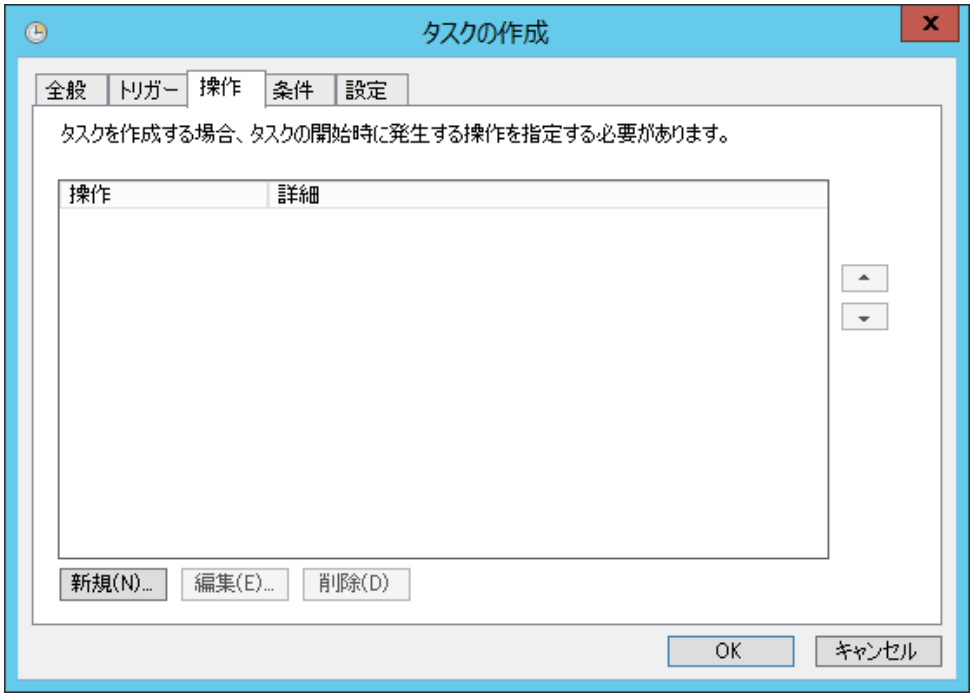

図18.9 タスクスケジューラの[操作]タブ

[操作]タブの[新規]ボタンをクリックして[新しい操作]ダイアログを表示します。

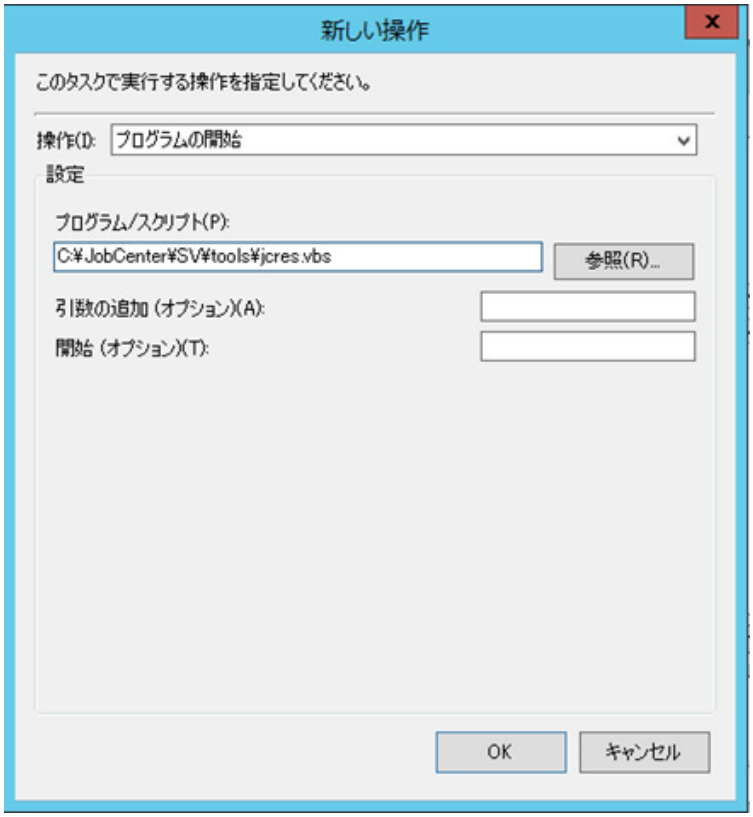

図18.10 タスクスケジューラの[新しい操作]ダイアログ

■[プログラム/スクリプト]の[参照]ボタンをクリックして作成したvbsスクリプトを選択します。本説明の例では、「C:\JobCenter\SV\tools\jcres.vbs」を指定しています。

設定が完了したら[OK]ボタンをクリックしてください。一覧にvbsスクリプトの操作が登録されます。

f. [設定]タブの設定をします。

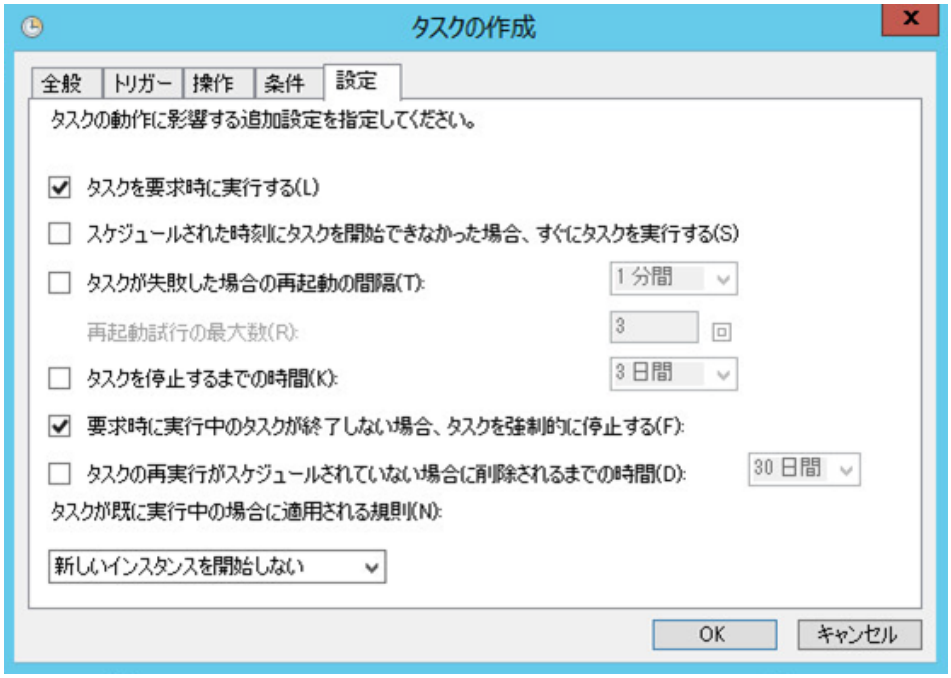

図18.11 タスクスケジューラの[設定]タブ

■[タスクを停止するまでの時間]のチェックを外します。

g. 全ての設定が完了したら[OK]ボタンをクリックしてください。

タスクを実行するユーザのパスワードを要求されますので、JobCenter管理者のパスワードを入力してください。

> 環境によって、OSが起動してからjcresが起動するまで時間がかかる場合があります。

18.5.3.2. 設定削除方法

1. タスクスケジューラの起動

コントロールパネルからタスクスケジューラを起動します。

起動方法は「18.5.3.1 設定方法」の「2. タスクスケジューラの起動」を参照してください。

2. タスクの削除

左ペインのjcresのフォルダーを選択します。中央のタスク一覧からjcresのタスクを右クリックして[削除]をクリックします。

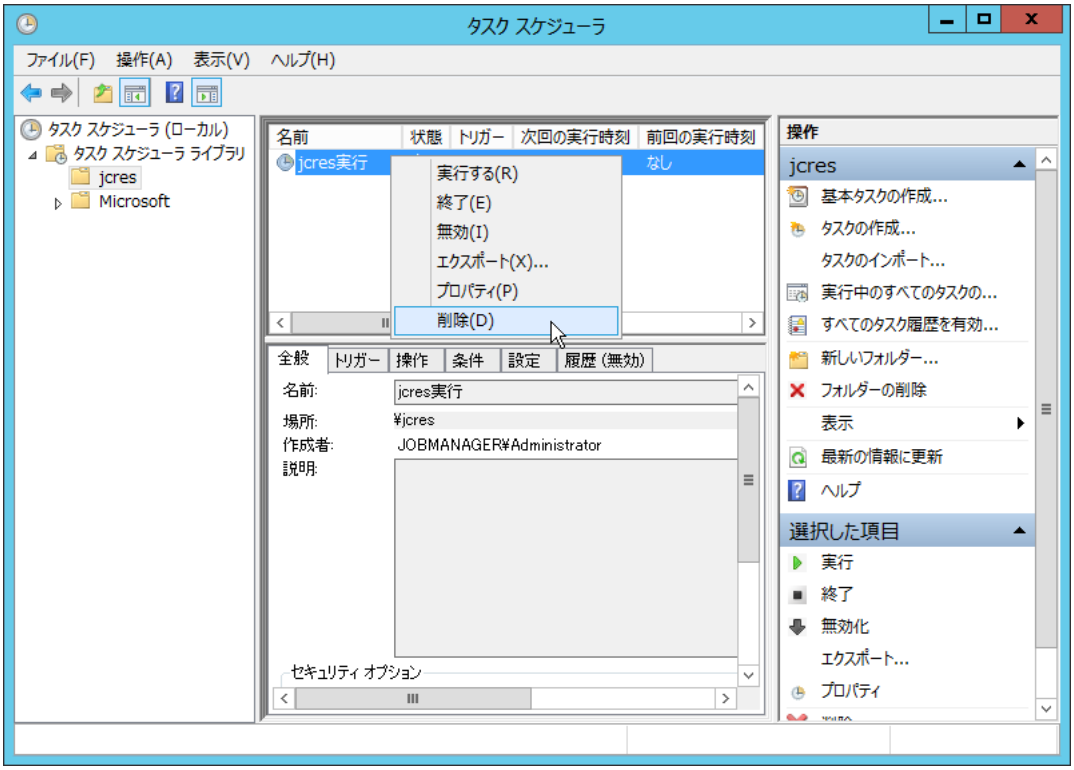

図18.12 タスクスケジューラのタスクの削除

3. フォルダーの削除

左ペインのjcresのフォルダーを右クリックして[フォルダーの削除]をクリックします。

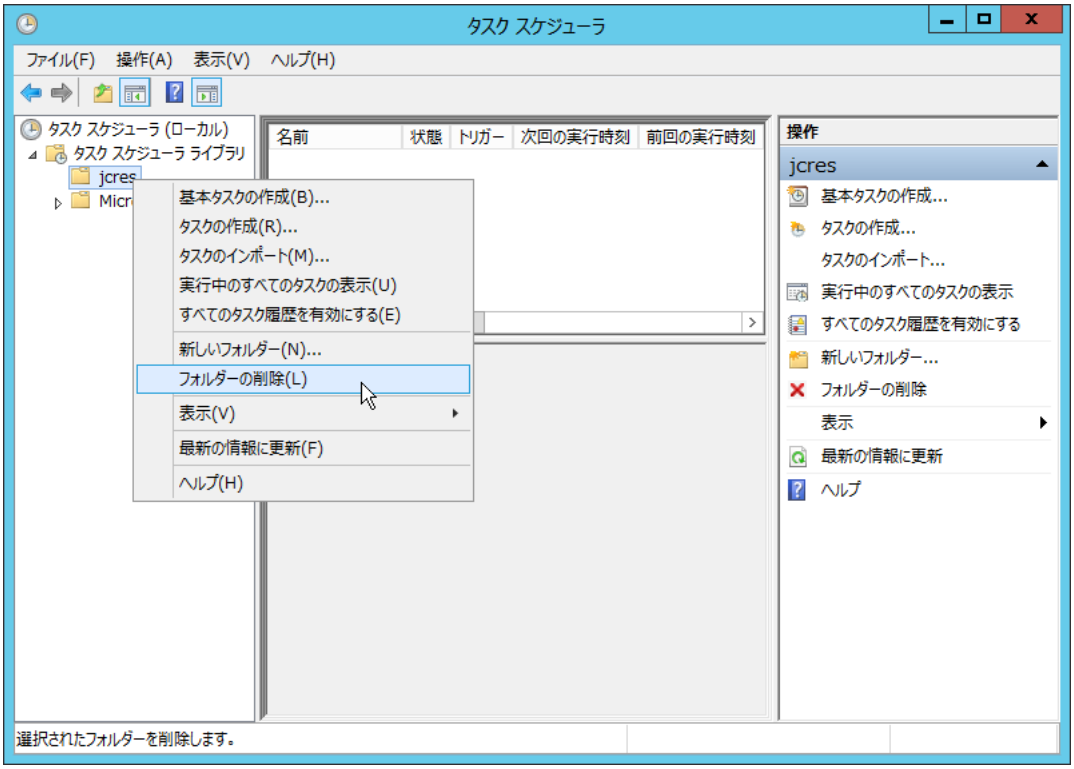

図18.13 タスクスケジューラのフォルダーの削除

4. vbsスクリプトの削除

   「18.5.3.1 設定方法」で作成したvbsスクリプトを削除します。

19. WebAPI機能

JobCenter で提供しているHTTP/HTTPSプロトコルを利用したWebベースのAPI機能について説明します。

WebAPIは、JobCenterが管理するジョブネットワークやトラッカ等の情報にHTTP/HTTPSを使用してアクセスするためのインタフェースです。

本機能はJobCenter　　のサイトのjcwebserverポートに対し、特定のフォーマットでHTTPリクエストを送ると、「19.6 WebAPI version 2」で記載した情報の取得や制御を行うことができます。

本機能が利用できるJobCenter のサイトは、以下の条件を満たしている必要があります。

■R16.1以降のWindows版またはLinux版のJobCenter であること

■JobCenter MGのライセンスおよびJobCenter CL/Webのライセンスを適用していること

■該当のサイトでJobCenter のjcwebserverデーモンが起動していること

■jcwebserverプロセスの設定の詳細については、「5.2 デーモン設定ファイルの使用可能パラメータ」および「5.2 デーモン設定ファイルの使用可能パラメータ」のjcwebserver_enabledパラメータを参照してください。

■jcwebserverポートの詳細については、「2.1　JobCenterで使用するTCPポート」を参照してください。

■WebAPIを使用するJobCenter　のサイトの環境が複数NICまたはクラスタ環境の場合、HTTPリクエストに指定するホスト名またはIPアドレスは対象のサイトのものを指定する必要があります。複数NICまたはクラスタ環境の場合にサイトに設定されるIPアドレスについては「5.3.1　クラスタ構成／複数NIC使用時のdaemon.conf設定について」で設定した内容を基に確認をしてください。

19.1. ライセンスについて

WebAPI機能を利用したい場合は、JobCenter　MGとJobCenter　CL/Webのライセンスの購入が必要となります。

JobCenter CL/WebのライセンスはJobCenterのオプションとして別途有償にて提供されているものです。

19.2. WebAPI導入のイメージ

WebAPIを導入することでJobCenterをインストールすることなくジョブネットワークの参照や即時投入、トラッカの状況参照等のアプリケーションを独自に構築することができます。

API機能は該当のサイトのJobCenterが起動していなければ利用することが出来ません。API機能の動作に不整合があった場合には該当のサイトの起動状況をご確認ください。

■WebAPI利用の効果

従来ではCL/WinでJobCenterにログインを行い、CL/Winの画面仕様にそって操作を行う必要がありました。

WebAPIを導入するとジョブネットワークの参照や即時投入、トラッカの状況参照等の独自のアプリケーションをクライアントに提供することができます。

また、R16.1より前ではCL/Webサーバ経由でしかWebAPIは利用できませんでしたが、R16.1からは直接、JobCenterのサーバに対してWebAPIを利用できる様になりました。

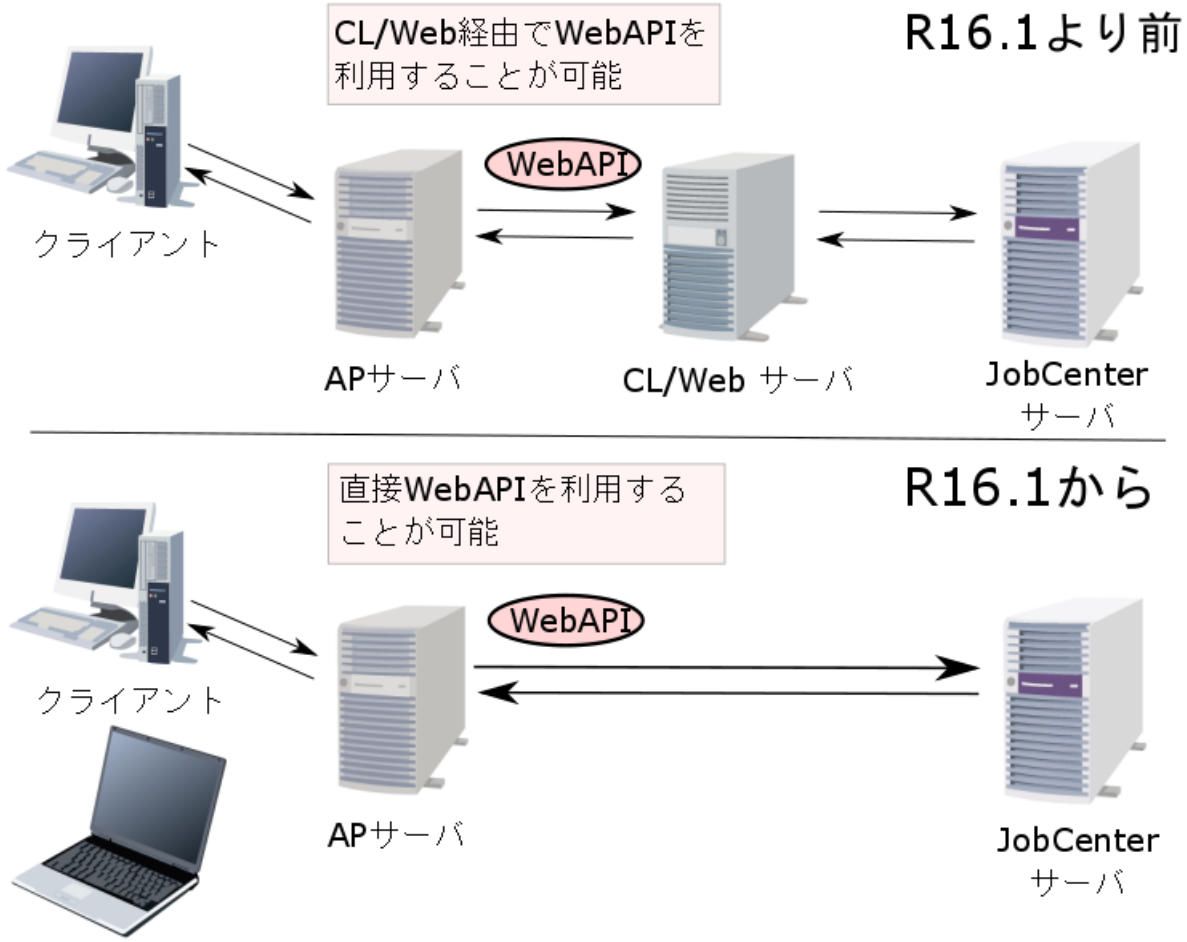

図19.1 導入の効果

一般ユーザはクライアントからAPサーバへとアクセスし、そのアクセスに対応したアプリケーションを開発者は作成します。そのアプリケーションには必要に応じてWebAPIを利用し、JobCenterの操作を組み込むことができます。

19.3. WebAPIのリクエスト内容

リクエストに指定する内容について説明します。

■WebAPIの入出力パラメータの文字コードはUTF-8です。

■クエリパラメータは、URLの末尾に「?」を付与し、続けて以下のように指定します。

```
パラメータ名=値
```

■クエリパラメータが複数ある場合は、「&」で区切って指定します。

```
パラメータ１=値&パラメータ２=値
```

■クエリパラメータにURLで指定できない文字を指定する場合は、その文字をURLエンコードする必要があります。

■WebAPIでは、URLに使用できる最大文字数は10,240文字です。 environment_variableをクエリパラメータに指定する場合は、最大文字数を超えないよう注意してください。

なお、ブラウザ仕様により、最大文字数まで使用できない場合があります。

Internet ExplorerではURLに最大2,083文字が使用可能です。詳細は、次のMicrosoft公開情報を参照してください。

https://support.microsoft.com/ja-jp/help/208427/

19.4. WebAPI一覧

JobCenter MG/SVで提供しているWebAPIのバージョンは version 2 のみとなります。

インタフェース詳細は「19.6 WebAPI version 2」を参照してください。

19.4.1. WebAPI version 2 機能一覧

JobCenter MG/SVのWebAPI version2が提供する機能一覧は次のとおりです。

表19.1 WebAPI version 2 機能一覧

リソース	機 能	参照先
APIアクセスキー	APIアクセスキーを作成する	「19.6.2.1 APIアクセスキーの作成」
	APIアクセスキーを削除する	「19.6.2.2 APIアクセスキーの削除」
ジョブネットワーク	ジョブネットワーク一覧を取得する	「19.6.3.1 ジョブネットワーク一覧の取得」
	ジョブネットワークを操作する	「19.6.3.2 ジョブネットワークの操作」
トラッカ	トラッカ一覧を取得する	「19.6.4.1 トラッカ一覧の取得」
	トラッカの詳細情報を取得する	「19.6.4.2 トラッカ詳細情報の取得」
	トラッカ部品一覧を取得する	「19.6.4.3 トラッカ部品一覧の取得」
	トラッカ部品の詳細情報を取得する	「19.6.4.4 トラッカ部品の詳細情報の取得」
	トラッカ部品の予想実行時間を変更する	「19.6.4.5 トラッカ部品の予想実行時間の変更」
	トラッカを操作する	「19.6.4.6 トラッカの操作」
	トラッカ部品を操作する	「19.6.4.7 トラッカ部品の操作」

19.5. WebAPIの認証

JobCenter MG/SVのWebAPIはAPIアクセスキー認証の認証方式のみをサポートしています。

認証は、「APIアクセスキー認証」で行うようにしてください。

19.5.1. APIアクセスキー認証

APIアクセスキーは、JobCenter MG/SV内で保持されるユーザ固有の情報（40バイトの文字列）です。

APIアクセスキーは個人情報（ユーザ名、パスワード等）から生成されたものではないので、APIアクセスキーからユーザが特定されることはありません。

APIアクセスキーを利用してユーザ認証する場合、HTTPリクエスト時に以下を指定します。

■クエリパラメータ：APIアクセスキー

パラメータ指定方法は、APIの各バージョンにおける説明を参照してください。

19.5.1.1. APIアクセスキーの取得方法

APIアクセスキーの取得方法は、「19.6.2.1 APIアクセスキーの作成」にて取得してください。

19.5.1.2. APIアクセスキーの削除方法

APIアクセスキーの削除方法は、「19.6.2.2 APIアクセスキーの削除」にて削除してください。

19.6. WebAPI version 2

JobCenter MG/SVのWebAPI version 2（以下、WebAPI(V2)）について説明します。

19.6.1. 利用方法

WebAPI(V2)の基本的な利用方法について説明します。

19.6.1.1. 制限事項

WebAPI(V2)を利用するにあたり、以下の制限事項があります。

■対象の JobCenter MG/SV のサイトに以下のユーザで事前にCL/Winでログインしている必要があります。

- Linux
  - root
  - WebAPI(V2)を利用するユーザ
- Windows
  - JobCenter管理者
  - WebAPI(V2)を利用するユーザ

> CL/Winによるログインでなく、jc_usercfgコマンドによるユーザ環境の作成でも可能です。
>
> jc_usercfgコマンドの詳細については<コマンドリファレンス>の「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。

19.6.1.2. API認証方法

WebAPI(V2)における認証方法について説明します。

■APIアクセスキー認証

URL中のクエリパラメータのapi_keyにAPIアクセスキーを設定します。APIアクセスキーの取得方法については「19.6.2.1 APIアクセスキーの作成」を参照してください。

```
api_key=<APIアクセスキー>
```

19.6.1.3. パラメータフォーマット

WebAPI(V2)で使用するパラメータの型について説明します。

表19.2 パラメータフォーマット一覧

型	値の例
String	"文字列"
Integer	0
Boolean	true
Hash	{"key":value}
Array	["run"]
Null	null

型	値の例
PartsName	"JOB1"
Jobnetwork	"JNW1"
TrackerId	"JNW1:JNW1-1.20180402000000"
ISO8601	"2018-04-02T00:00:00+09:00"

■String

文字列が入るパラメータです。

■Integer

整数値が入るパラメータです。

■Boolean

真理値が入るパラメータです。

trueもしくは1のときに真、falseもしくは0のときに偽を意味します。

■Hash

ハッシュが入るパラメータです。

"key":valueの形式のオブジェクトで表します。カンマ（,）で区切ることにより複数のハッシュを表します。

例）{"key1":value1,"key2":value2}

■Array

配列が入るパラメータです。

Array[Hash]のように要素型を明示しているものは、その型の要素が入ります。

要素が存在しない配列の場合、空配列（[]）となります。

■Null

nullが入るパラメータです。データが存在しないことを表します。

■Jobnetwork

String の特殊型です。ジョブネットワークの名前を表現し、次の規則があります。

- 40バイト以内であること。
- 先頭の文字がアルファベット、アンダーライン、数字、2バイト文字であること。
- スペース、タブ、改行、制御文字、半角カタカナおよび次の記号が利用されないこと。
  - + ! " # $ % & ' ( ) * , . / : ; < = > ? @ [\ ] ^ ` { | } ~

バイト数はリクエスト先のJobCenter MG/SV セットアップ言語によって計算方法が異なります。
日本語環境の場合Shift-JISでエンコードした文字列のバイト数を計算します。

■PartsName

String の特殊型です。ジョブネットワークフローに配置する部品の名前を表現し、次の規則があります。

- 40バイト以内であること。
- 先頭の文字がアルファベット、アンダーライン、数字、2バイト文字であること。
- スペース、タブ、改行、制御文字および次の記号が利用されないこと。
  - ! " # $ % & ' ( ) * , . / : ; < = > ? @ [\ ] ^ ` { | } ~

> バイト数はリクエスト先のJobCenter MG/SV セットアップ言語によって計算方法が異なります。日本語環境の場合Shift-JISでエンコードした文字列のバイト数を計算します。

■TrackerId

String の特殊型です。トラッカIDを表現し、次の規則があります。

<ジョブネットワーク名>.<投入時刻>

- ジョブネットワーク名

  ジョブネットワーク名を表します。ジョブネットワーク名については、Jobnetwork型の制限に従います。

  サブジョブネットワークの場合、<最上位のジョブネットワーク名>:・・・・:<サブジョブネットワーク名>のフォーマットで表します。

  例）JNW1:JNW1-1:JNW1-1-1

  ジョブネットワークの階層数は最大10階層までです。

- 投入時刻

  ジョブネットワークを投入した時刻（GMTに準ずる）をYYYYMMDDhhmmssの形式で表します。

  例）20180402000000

  値の制限は次のとおりです。

  - 下限:19970101000000
  - 上限:20371231235959

■ISO8601

String の特殊型です。時刻を ISO8601 の拡張形式のフォーマットである <日付>T<時刻><タイムゾーン>の形式で表現し、次の規則があります。

- 日付

  年月日をYYYY-MM-DDの形式で表します。

  例）2018-04-02

- 時刻

  時分秒をhh:mm:ssの形式で表します。

  例）18:30:59

- タイムゾーン

タイムゾーンを表します。UTC時刻表記（Z）または+(-)hh:ssの形式で表します。

例）+09:00

レスポンスデータにおいては、タイムゾーンはUTC時刻表記で表示します。

値の制限は次のとおりです。

- 下限:1997-01-01T00:00:00+00:00
- 上限:2037-12-31T23:59:59+00:00

19.6.1.4. リクエスト形式

WebAPI(V2)ではJobCenter　MG/SVのサイトのjcwebserverポートに対して次の形式でリクエストを送信することでAPIが実行されます。

```
<メソッド> /api/<バージョン>/<リソース>?<クエリパラメータ>
```

また、リクエストのURLのホスト名（IPアドレス）の部分は、対象となるサイトのホスト名またはIPアドレスとjcwebserverポート番号を以下の形式で指定します。

```
<ホスト名（IPアドレス）>:<jcwebserverポート番号>
```

URL例

```
https://example.com:23180/api/v2/jobnetworks?api_key=dsfvoeishfhseiofhoseihfiosheiofhsioehfi
```

リクエストのURLにIPv6アドレスを指定する場合には、IPv6アドレスを[]括弧で囲ってください。

```
https://[2001:db8:0:0:12:0:0:1]:23180/api/v2/jobnetworks?
api_key=dsfvoeishfhseiofhoseihfiosheiofhsioehfi
```

リクエスト形式の構成内容について説明します。

表19.3 リクエスト形式の構成内容

項目	説明	例
メソッド	HTTPリクエストのメソッドを指定します。指定できるメソッドはGET・POST・PATCH・DELETEとなります。	GET
バージョン	APIのバージョンを指定します。指定できるバージョンは「v2」です。	v2
リソース	アクセスするリソースを指定します。詳細については各APIのリクエストURLの説明を参照してください。	jobnetworks
クエリパラメータ	クエリパラメータを指定します。詳細については「19.6.1.4.2 クエリパラメータ」を参照してください。	api_key=<APIアクセスキー>

接続しているMG/SVの環境に関わらず、各APIにおける入力パラメータにおいて大文字小文字は区別します。

19.6.1.4.1. リクエストヘッダ

JobCenter　MG/SVのサイトのjcwebserverポートに対して、HTTPリクエストを行う際に必要となるリクエストヘッダについて説明します。リクエストヘッダに必要なフィールドは次のとおりになります。

表19.4 リクエストヘッダ

フィールド名	説明	例
Host	リクエストを行うJobCenter　MG/SVのサイトのホスト情報を指定します。ホスト情報は「<ホスト名>:<ポート番号>」の形式で指定します。	example.com:23180
Accept	レスポンスデータに指定したいメディアタイプを指定します。固定で「application/json」を指定してください。省略した場合は全てのデータ形式を許容する「*/*」が指定されたと見なします。	application/json
Content-Type	リクエストボディがある場合に、リクエストボディのメディアタイプを指定します。固定で「application/json」を指定してください。	application/json
Content-Length	リクエストボディがある場合に、リクエストボディの長さ(バイト)を指定します。	1234

メディアタイプは「トップレベルタイプ名/サブタイプ名」の形式で指定します。トップレベルタイプ名に*を指定した場合、サブタイプ名の値に関わらず「*/*」と指定したものと見なします。

Content-Lengthを指定していない場合や、指定した値がbodyのサイズよりも大きい場合等、HTTPリクエストとして不正なエラーが起きた場合、レスポンスはHTML形式で返却します。またこのときのリクエストは証跡ログには記録されません。

19.6.1.4.2. クエリパラメータ

WebAPI(V2)ではAPIアクセスキー認証をおこないます。よって、APIアクセスキーの作成・削除以外のAPIでは次のクエリパラメータを指定する必要があります。

表19.5 クエリパラメータ

パラメータ	説明	例
api_key	APIアクセスキー認証でHTTPリクエストを行う場合に、APIアクセスキー（40バイトの文字列）を指定します。	dsfvo...（40バイト）

また、上記以外に、各API毎に用意されているクエリパラメータを指定します。

POSTメソッドを使用するAPIの場合、クエリパラメータに指定するパラメータをリクエストボディで指定することも可能です。クエリパラメータとリクエストボディの両方で指定したパラメータはクエリパラメータの値で動作します。

クエリ内で同一のパラメータを複数回指定した場合、後から指定されているパラメータの値で動作します。

19.6.1.4.3. リクエストボディ

HTTPリクエストを行う際のリクエストボディについて説明します。

リクエストボディはJSON形式で指定する必要があります。リクエストボディで指定できるパラメータは各APIのリクエストボディの説明を参照してください。

リクエストボディに指定するパラメータはクエリパラメータで指定することも可能です。クエリパラメータとリクエストボディの両方で指定したパラメータはクエリパラメータの値で動作します。

リクエストボディ内で同一のパラメータを複数回指定した場合、後から指定されているパラメータの値で動作します。

19.6.1.5. レスポンス形式

JobCenter MG/SVが返すレスポンスの形式の基本構成について説明します。

レスポンスボディはJSON形式です。

19.6.1.5.1. レスポンスヘッダ

レスポンス返却時のHTTPヘッダは次のとおりです。

表19.6 レスポンスヘッダ

フィールド名	説明	例
Content-Type	レスポンスボディがある場合に、レスポンスボディのメディアタイプを示します。	application/json
Content-Length	レスポンスボディがある場合に、レスポンスボディの長さ(バイト)を示します。	1234
Date	レスポンスの日時を示します。	Wed, 07 Feb 2018 01:35:13 GMT

19.6.1.5.2. API成功時のレスポンスボディ

APIが成功した時にJobCenter MG/SVが返すレスポンスボディについて説明します。

19.6.1.5.2.1. 単一のレスポンスデータ

レスポンスデータが単一の場合のレスポンスボディは次のとおりです。

```
{
  "data":{
    <各APIの結果>
  }
}
```

表19.7 レスポンスボディのパラメータ（単一）

パラメータ	型	説明	例
data	Hash	各APIの結果をHashで返却します。詳細については各APIのレスポンスボディの説明を参照してください。	{"tracker_id":"jnw1.20171016052922"}

19.6.1.5.2.2. 複数のレスポンスデータ

レスポンスデータが複数の場合のレスポンスボディは次のとおりです。

```
{
  "count":<データの数>,
  "data":[
    {<各APIの結果>},
    {<各APIの結果>}
  ]
}
```

表19.8 レスポンスボディのパラメータ（複数）

パラメータ	型	説明	例
count	Integer	dataパラメータの要素数を返却します。	1

パラメータ	型	説明	例
data	Array[Hash]	各APIの結果をHashの配列にして返却します。詳細については各APIのレスポンスボディの説明を参照してください。	{"jobnetwork_name":"jnw1", "comment":"コメント", "group_path":"UnitJnw"}

19.6.1.5.3. API失敗時のレスポンスボディ

APIが失敗した時にJobCenter MG/SVが返すレスポンスボディのフォーマットは次のとおりです。

```
{
  "error_code":<エラーコード>,
  "errors":[
    {
      "message":<エラーメッセージ>
    }
  ]
}
```

表19.9 レスポンスボディのパラメータ(失敗時)

パラメータ	型	説明	例
error_code	String	エラーの分類を表すエラーコードです。返却されるエラーコードについては「19.6.5 WebAPIのエラー一覧」を参照してください。	"NOT_FOUND"
errors	Array[Hash]	エラーメッセージを格納するHashの配列です。クエリまたはボディで指定したパラメータに複数のエラーが存在する場合は、複数のエラーメッセージを格納します。	[{"message":"The resource is not found."}]
message	String	エラーの詳細を表すエラーメッセージです。エラーメッセージの詳細については「19.6.5 WebAPIのエラー一覧」を参照してください。	"The resource is not found."

19.6.2. APIアクセスキーリソース

APIアクセスキーリソースに対して実行するAPIについて説明します。

19.6.2.1. APIアクセスキーの作成

19.6.2.1.1. 機能概要

指定したユーザのAPIアクセスキーを作成して、そのAPIアクセスキーの情報を返信します。既にAPIアクセスキーを作成していた場合には、再作成せずに既存のAPIアクセスキーの情報を返信します。

本APIの実行には「19.6.1.1 制限事項」にくわえ以下の制限事項があります。

■対象の JobCenter MG/SV のマシン上でのみ実行可能

■リクエストURLのIPアドレス部分には以下のみ指定可能

IPアドレスではなくホスト名を指定する場合、名前解決後のIPアドレスが以下であること

- ローカルサイト

  ローカルIP または ローカルループバックIP

- クラスタサイト

  ローカルIP

ローカルサイトでローカルループバックIPが指定できるのは、daemon.confに「local_daemon=SITE」および「ipaddress=IPアドレス」が設定されていない環境の場合のみとなります。

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 デーモン起動オプション」を、ipaddressパラメータの詳細については「5.2.1 Windows/UNIX共通」のipaddressパラメータを参照してください。

19.6.2.1.2. HTTPリクエスト仕様

■メソッド

POST

■リクエストURL

/api/v2/apikey

■リクエストヘッダ

リクエストヘッダについては「19.6.1.4.1 リクエストヘッダ」を参照してください。

■パラメータ

WebAPI(V2)共通で使用するクエリパラメータについては「19.6.1.4.2 クエリパラメータ」を参照してください。

API固有のパラメータについては次のとおりです。

表19.10 リクエストパラメーター一覧

パラメータ	指定箇所	型	必須	説明	バージョン制約
user	リクエストボディ	String	○	APIアクセスキーを作成するユーザ名を指定します。	-

パラメータ	指定箇所	型	必須	説明	バージョン制約
password	リクエストボディ	String	○	APIアクセスキーを作成するユーザのパスワードを指定します。	-

■リクエスト例

```
POST
https://example.com:23180/api/v2/apikey
{
  "user": "Administrator",
  "password": "password"
}
```

19.6.2.1.3. HTTPレスポンス仕様

■API成功時

- HTTPステータスコード

  201

- レスポンスボディ

  API成功時のレスポンスボディのフォーマットについては「19.6.1.5.2 API成功時のレスポンスボディ」を参照してください。

  以下ではdataパラメータに格納するAPIの実行結果のフォーマットについて説明します。

表19.11 レスポンスボディのパラメーター一覧

パラメータ	型	説明	例
api_key	String	指定したユーザのAPIアクセスキーを返却します。	-

レスポンスボディ例

```
{
  "data":{
    "api_key": "dsfvoeishfhseiofhoseihfiosheiofhsioehfi"
  }
}
```

■API失敗時

「19.6.1.5.3 API失敗時のレスポンスボディ」を参照してください。

19.6.2.1.4. 注意事項

■リクエストのURLにホスト名を指定した際の名前解決でIPv6のリンクローカルアドレスが優先で返ってくる環境の場合、ホスト名でなくIPアドレスを直接、URLに指定してください。

19.6.2.2. APIアクセスキーの削除

19.6.2.2.1. 機能概要

指定したユーザのAPIアクセスキーを削除します。指定したユーザのAPIアクセスキーが作成されていない場合でもリクエストは成功を返します。

本APIの実行には「19.6.1.1 制限事項」にくわえ以下の制限事項があります。

■対象の JobCenter MG/SV のマシン上でのみ実行可能

■リクエストURLのIPアドレス部分には以下のみ指定可能

IPアドレスではなくホスト名を指定する場合、名前解決後のIPアドレスが以下であること

- ローカルサイト

  ローカルIP または ローカルループバックIP

- クラスタサイト

  ローカルIP

ローカルサイトでローカルループバックIPが指定できるのは、daemon.confに「local_daemon=SITE」および「ipaddress=IPアドレス」が設定されていない環境の場合のみとなります。

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 デーモン起動オプション」を、ipaddressパラメータの詳細については「5.2.1 Windows/UNIX共通」のipaddressパラメータを参照してください。

19.6.2.2.2. HTTPリクエスト仕様

■メソッド

DELETE

■リクエストURL

/api/v2/apikey

■リクエストヘッダ

リクエストヘッダについては「19.6.1.4.1 リクエストヘッダ」を参照してください。

■パラメータ

WebAPI(V2)共通で使用するクエリパラメータについては「19.6.1.4.2 クエリパラメータ」を参照してください。

API固有のパラメータについては次のとおりです。

表19.12 リクエストパラメーター一覧

パラメータ	指定箇所	型	必須	説明	バージョン制約
user	リクエストボディ	String	○	APIアクセスキーを削除するユーザ名を指定します。	-

パラメータ	指定箇所	型	必須	説明	バージョン制約
password	リクエストボディ	String	○	APIアクセスキーを削除するユーザのパスワードを指定します。	-

■リクエスト例

```
DELETE
https://example.com:23180/api/v2/apikey
{
  "user": "Administrator",
  "password": "password"
}
```

19.6.2.2.3. HTTPレスポンス仕様

■API成功時

- HTTPステータスコード

  204

- レスポンスボディ

  なし

■API失敗時

「19.6.1.5.3 API失敗時のレスポンスボディ」を参照してください。

19.6.2.2.4. 注意事項

■リクエストのURLにホスト名を指定した際の名前解決でIPv6のリンクローカルアドレスが優先で返ってくる環境の場合、ホスト名でなくIPアドレスを直接、URLに指定してください。

19.6.3. ジョブネットワークリソース

ジョブネットワークリソースに対して実行するAPIについて説明します。

19.6.3.1. ジョブネットワーク一覧の取得

本APIの機能は CL/Webの WebAPI version 2 の「ジョブネットワーク一覧の取得」と同じため、機能説明・HTTPリクエスト仕様・HTTPレスポンス仕様・注意事項等の詳細については<Web機能利用の手引き>の「6.5.2.1 ジョブネットワーク一覧の取得」を参照してください。

但しHTTPリクエスト仕様のリクエスト例の「example.com」の部分は「example.com:23180」に、「CL/Webサーバ」の部分は「JobCenter MG/SV」に読み替えてください。

19.6.3.2. ジョブネットワークの操作

本APIの機能は CL/Webの WebAPI version 2 の「ジョブネットワークの操作」と同じため、機能説明・HTTPリクエスト仕様・HTTPレスポンス仕様・注意事項等の詳細については<Web機能利用の手引き>の「6.5.2.2 ジョブネットワークの操作」を参照してください。

但しHTTPリクエスト仕様のリクエスト例の「example.com」の部分は「example.com:23180」に、「CL/Webサーバ」の部分は「JobCenter MG/SV」に読み替えてください。

19.6.4. トラッカリソース

トラッカリソースに対して実行するAPIについて説明します。

19.6.4.1. トラッカー覧の取得

本APIの機能は CL/Webの WebAPI version 2 の「トラッカー覧の取得」と同じため、機能説明・HTTPリクエスト仕様・HTTPレスポンス仕様・注意事項等の詳細については<Web機能利用の手引き>の「6.5.3.1 トラッカー覧の取得」を参照してください。

但し、HTTPリクエスト仕様のリクエスト例の「example.com」の部分は「example.com:23180」に、「CL/Webサーバ」の部分は「JobCenter MG/SV」に読み替えてください。

19.6.4.2. トラッカ詳細情報の取得

本APIの機能は CL/Webの WebAPI version 2 の「トラッカ詳細情報の取得」と同じため、機能説明・HTTPリクエスト仕様・HTTPレスポンス仕様・注意事項等の詳細については<Web機能利用の手引き>の「6.5.3.2 トラッカ詳細情報の取得」を参照してください。

但し、HTTPリクエスト仕様のリクエスト例の「example.com」の部分は「example.com:23180」に、「CL/Webサーバ」の部分は「JobCenter MG/SV」に読み替えてください。

19.6.4.3. トラッカ部品一覧の取得

本APIの機能は CL/Webの WebAPI version 2 の「トラッカ部品一覧の取得」と同じため、機能説明・HTTPリクエスト仕様・HTTPレスポンス仕様等の詳細については<Web機能利用の手引き>の「6.5.3.3 トラッカ部品一覧の取得」を参照してください。

但し、HTTPリクエスト仕様のリクエスト例の「example.com」の部分は「example.com:23180」に、「CL/Webサーバ」の部分は「JobCenter MG/SV」に読み替えてください。

19.6.4.4. トラッカ部品の詳細情報の取得

本APIの機能は CL/Webの WebAPI version 2 の「トラッカ部品の詳細情報の取得」と同じため、機能説明・HTTPリクエスト仕様・HTTPレスポンス仕様・注意事項等の詳細については<Web機能利用の手引き>の「6.5.3.4 トラッカ部品の詳細情報の取得」を参照してください。

但し、HTTPリクエスト仕様のリクエスト例の「example.com」の部分は「example.com:23180」に、「CL/Webサーバ」の部分は「JobCenter MG/SV」に読み替えてください。

19.6.4.5. トラッカ部品の予想実行時間の変更

19.6.4.5.1. 機能概要

トラッカおよび部品を指定し、対象となる部品の予想実行時間を変更します。予想実行時間を変更できる部品種別は次のとおりです。

■単位ジョブ部品

■カスタムジョブ部品（拡張カスタムジョブ部品も含みます）

■WOBSジョブ部品

■ダイアログ部品

19.6.4.5.2. HTTPリクエスト仕様

■メソッド

PATCH

■リクエストURL

/api/v2/trackers/<トラッカID>/parts/<部品識別子>/<部品名>

■リクエストヘッダ

リクエストヘッダについては「19.6.1.4.1 リクエストヘッダ」を参照してください。

■パラメータ

WebAPI(V2)共通で使用するクエリパラメータについては「19.6.1.4.2　クエリパラメータ」を参照してください。

API固有のパラメータについては次のとおりです。

表19.13 リクエストパラメーター一覧

パラメータ	指定箇所	型	必須	説明	バージョン制約
<トラッカID>	URL	TrackerId	○	対象の部品が存在するトラッカのIDを指定します。	-
<部品識別子>	URL	String	○	対象の部品の部品識別子を指定します。	-
<部品名>	URL	PartsName	○	対象の部品名を指定します。	-
exptime	リクエストボディ	Integer	○	対象の部品の予想実行時間を指定します。	-

各パラメータの詳細は次のとおりです。

<部品識別子>

部品の種別を示す識別子です。部品種別と識別子と対応表は次のとおりです。

部品種別	部品識別子
単位ジョブ部品	unit_job
カスタムジョブ部品、拡張カスタムジョブ部品	custom_job

部品種別	部品識別子
WOBSジョブ部品	wobs_job
ダイアログ部品	dialog

exptime
指定した部品の変更する予想実行時間の値を指定します。値の単位は秒です。 指定できる値の範囲は 0 ～ 31536000 です。

■リクエスト例

```
PATCH
https://example.com:23180/api/v2/trackers/NewJnw.20220210012345/parts/unit_job/JOB1?
api_key=dsfvoeishfhseiofhoseihfiosheiofhsioehfi
{
  "exptime": 5
}
```

19.6.4.5.3. HTTPレスポンス仕様

■API成功時

- HTTPステータスコード

  204

- レスポンスボディ

  なし

■API失敗時

「19.6.1.5.3 API失敗時のレスポンスボディ」を参照してください。

19.6.4.5.4. 注意事項

■予想実行時間を変更できるのは、ステータスがWAIT状態の部品のみとなります。WAIT状態以外の部品に対して予想実行時間の変更を実行した場合、APIの実行はHTTPステータスコード:403の以下のエラーになります。

```
{
  "error_code": "CANNOT_UPDATE",
  "errors": [
    { "message": "The parameter can not be updated.(exptime)" }
  ]
}
```

19.6.4.6. トラッカの操作

19.6.4.6.1. 機能概要

指定したトラッカに対して「action」パラメータで指定した動作に対応した「value」パラメータの命令を実行します。

19.6.4.6.2. HTTPリクエスト仕様

■メソッド

POST

■リクエストURL

/api/v2/trackers/<トラッカID>

■リクエストヘッダ

リクエストヘッダについては「19.6.1.4.1 リクエストヘッダ」を参照してください。

■パラメータ

WebAPI(V2)共通で使用するクエリパラメータについては「19.6.1.4.2　クエリパラメータ」を参照してください。

API固有のパラメータについては次のとおりです。

表19.14 リクエストパラメーター一覧

パラメータ	指定箇所	型	必須	説明	バージョン制約
<トラッカID>	URL	TrackerId	○	操作するトラッカのIDを指定します。	-
action	リクエストボディ	String	○	トラッカに対する動作を指定します。	-
value	リクエストボディ	String	○	指定したトラッカの動作に対する命令を指定します。	-

各パラメータの詳細は次のとおりです。

action

トラッカに対する動作を指定します。

指定できる値は次のとおりです。

- control

  トラッカを操作します。

value

指定したトラッカの動作に対する命令を指定します。

指定できる値は次のとおりです。

actionの値	valueの値	操作の内容
control	skip	指定したトラッカをスキップします。
	release_skip	指定したトラッカをスキップ解除します。

actionの値	valueの値	操作の内容
	hold	指定したトラッカを保留します。
	release_hold	指定したトラッカを保留解除します。
	force_stop	指定したトラッカを強制停止します。

■リクエスト例

```
POST
https://example.com:23180/api/v2/trackers/NewJnw.20220210012345?
api_key=dsfvoeishfhseiofhoseihfiosheiofhsioehfi
{
  "action": "control",
   "value": "skip"
}
```

19.6.4.6.3. HTTPレスポンス仕様

■API成功時

- HTTPステータスコード

  204

- レスポンスボディ

  なし

■API失敗時

「19.6.1.5.3 API失敗時のレスポンスボディ」を参照してください。

19.6.4.7. トラッカ部品の操作

19.6.4.7.1. 機能概要

トラッカおよび部品を指定し、対象となる部品に対して「action」パラメータで指定した動作に対応した「value」パラメータの命令を実行します。本APIで指定できる部品の種別は次のとおりです。

■サブジョブネットワーク部品

■単位ジョブ部品

■カスタムジョブ部品（拡張カスタムジョブ部品も含みます）

■ERPジョブ部品

■BIジョブ部品

■PCジョブ部品

■WOBSジョブ部品

■ダイアログ部品

■イベント送信部品

■イベント受信部品

19.6.4.7.2. HTTPリクエスト仕様

■メソッド

POST

■リクエストURL

/api/v2/trackers/<トラッカID>/parts/<部品識別子>/<部品名>

■リクエストヘッダ

リクエストヘッダについては「19.6.1.4.1 リクエストヘッダ」を参照してください。

■パラメータ

WebAPI(V2)共通で使用するクエリパラメータについては「19.6.1.4.2　クエリパラメータ」を参照してください。

API固有のパラメータについては次のとおりです。

表19.15 リクエストパラメータ一覧

パラメータ	指定箇所	型	必須	説明	バージョン制約
<トラッカID>	URL	TrackerId	○	対象の部品が存在するトラッカのIDを指定します。	-
<部品識別子>	URL	String	○	対象の部品の部品識別子を指定します。	-
<部品名>	URL	String	○	対象の部品名を指定します。	-
action	リクエストボディ	String	○	対象の部品に対する動作を指定します。	-
value	リクエストボディ	String	○	対象の部品の動作に対する命令を指定します。	-

各パラメータの詳細は次のとおりです。

<部品識別子>

部品の種別を示す識別子です。部品種別と識別子の対応表は次のとおりです。

部品種別	部品識別子
サブジョブネットワーク部品	sub_jobnetwork
単位ジョブ部品	unit_job
カスタムジョブ部品、拡張カスタムジョブ部品	custom_job
ERPジョブ部品	erp_job
BIジョブ部品	bi_job
PCジョブ部品	pc_job
WOBSジョブ部品	wobs_job
ダイアログ部品	dialog
イベント送信部品	event_transmission
イベント受信部品	event_reception

<部品名>

対象の部品名を指定します。

部品種別により指定できる文字及びバイト数は異なります。

- 部品種別がジョブネットワーク部品以外の場合

  「19.6.1.3 パラメータフォーマット」のPartsNameパラメータの規則に従います。

- 部品種別がジョブネットワーク部品の場合

  「19.6.1.3 パラメータフォーマット」のJobnetworkパラメータの規則に従います。

  但し、エイリアス名が設定されているサブジョブネットワーク部品を指定する場合、サブジョブネットワーク部品名は <エイリアス名(サブジョブネットワーク名)> の形式で指定し、個々の「エイリアス名」「サブジョブネットワーク名」の部分はJobnetworkパラメータの規則に従います。

action

対象の部品に対する動作を指定します。

指定できる値は次のとおりです。

- control

  トラッカの部品を操作します。

value

対象の部品の動作に対する命令を指定します。

指定できる値は次のとおりです。

actionの値	valueの値	操作の内容
control	skip	指定した部品をスキップします。

actionの値	valueの値	操作の内容
	release_skip	指定した部品をスキップ解除します。
	hold	指定した部品を保留します。
	release_hold	指定した部品を保留解除します。
	force_stop	指定した部品を強制停止します。
	force_start	指定した部品を強制起動します。
	rerun	指定した部品を再実行します。
	suspend	指定した部品をサスペンドにします。
	resume	指定した部品をサスペンド解除します。
	start_immediately	指定したERPジョブ部品のスタート操作（IMMEDIATELY）をします。
	start_asap	指定したERPジョブ部品のスタート操作（ASAP）をします。
	reply_ok	指定したダイアログ部品をOKにします。
	reply_error	指定したダイアログ部品をErrorにします。
	restart	指定した部品をリスタートします。
	release_control	指定した部品をコントロール解除します。
	set_normal	指定した部品を正常扱いに変更します。

■リクエスト例

```
POST
https://example.com:23180/api/v2/trackers/NewJnw.20220210012345/parts/unit_job/JOB1?
api_key=dsfvoeishfhseiofhoseihfiosheiofhsioehfi
{
  "action": "control",
   "value": "skip"
}
```

19.6.4.7.3. HTTPレスポンス仕様

■API成功時

- HTTPステータスコード

  204

- レスポンスボディ

  なし

■API失敗時

「19.6.1.5.3 API失敗時のレスポンスボディ」を参照してください。

19.6.5. WebAPIのエラー一覧

JobCenter MG/SVのWebAPI(V2)で返却するエラーは次のとおりです。

表19.16 エラー一覧

ステータスコード	エラーコード	エラーメッセージ	エラー原因
400	INVALID_PARAM	The required key is not specified.(<パラメータ名>)	必須パラメータが指定されていない場合に返却します。
		The value format is invalid.(<パラメータ名>)	パラメータの値の形式が不正な場合に返却します。
		The value range is invalid.(<パラメータ名>)	パラメータの値の範囲が不正な場合に返却します。
		The value is invalid.(<パラメータ名>)	パラメータの値が不正な場合に返却します。
		The key format is invalid.(<パラメータ名>)	各APIで定義されていないパラメータが指定された場合や、hash型のパラメータのkeyが不正な場合に返却します。
		The key includes one or more invalid characters.(<パラメータ名>)	hash型のパラメータのkeyに使用禁止文字が含まれている場合に返却します。
		The value includes one or more invalid characters.(<パラメータ名>)	hash型のパラメータのvalueに使用禁止文字が含まれているに返却します。
		The key is too long.(<パラメータ名>)	hash型のパラメータのkey名が最大文字数を超えている場合に返却します。
		The value is too long.(<パラメータ名>)	パラメータの値が最大文字数を超えている場合に返却します。
		The value type is invalid.(<パラメータ名>)	パラメータの値の型が不正な場合に返却します。
		Empty string can not be specified to the key.(<パラメータ名>)	hash型のパラメータのkey名が空文字の場合に返却します。
		Empty string can not be specified to the value.(<パラメータ名>)	パラメータの値が空文字の場合に返却します。
		The request body format is invalid.	リクエストボディの形式がJSON以外の場合に返却します。
		The key with dependency is not specified.(<パラメータ名>)	依存関係がある他のパラメータが指定されていない場合に返却します。
		The specified period is invalid.(<パラメータ名>)	指定した期間が不正な場合に返却します。
		The key is mutually exclusive.(<パラメータ名>)	同時に指定できないパラメータが指定された場合に返却します。
401	AUTHENTICATION_FAILED	The api_key is invalid.	APIアクセスキーが不正な場合に返却します。

ステータスコード	エラーコード	エラーメッセージ	エラー原因
		The user name or password is invalid.(user=<ユーザ名>)	認証時のユーザ名、またはパスワードが不正な場合に返却します。
		User does not exist.(user=<ユーザ名>)	認証時にユーザ名が存在しない場合に返却します。
		The client ip is not allowed to create or delete api key.(<クライアントIP>)	クライアントIPからAPIアクセスキーの作成または削除する権限がない場合に返却します。
403	PERMISSION_DENIED	You do not have permission.	以下の場合に返却します。 ■対象のジョブネットワークの参照権限がない ■対象のジョブネットワークの投入権限がない ■対象ユーザのトラッカの参照権限がない ■対象ユーザのトラッカの変更・削除権限がない ■対象ユーザのトラッカの操作権限がない
	PARTS_NOT_SUPPORTED	The part type is not support.(<部品種別>)	対象のAPIでサポートしていない部品がURLで指定された場合に返却します。
	CANNOT_UPDATE	The parameter can not be updated.(<パラメータ名>)	対象のユーザのトラッカ部品はWAIT状態でないため予想実行時間の更新ができない場合に返却します。
404	NOT_FOUND	The resource is not found.	存在しないURLが指定された場合に返却します。
405	METHOD_NOT_ALLOWED	The resource do not support http method.(<HTTPメソッド>)	指定したURLに対してHTTPメソッドがサポートされていない場合に返却します。
406	NOT_ACCEPTABLE	The request is not acceptable.	リクエストヘッダで指定されたAcceptがサポートされていない場合に返却します。
415	MEDIATYPE_NOT_SUPPORTED	The media type is not support.	リクエストヘッダで指定されたContent-Typeがサポートされていない場合に返却します。
500	INTERNAL_SERVER_ERROR	The server was unable to complete your request.	HTTPD内部の予期しないエラー、およびMG/SVの想定外のエラーが起きた場合に返却します。

20. JobCenterのプロセスを監視する

この章では、JobCenterのプロセスを監視ツールで監視する方法について説明します。

20.1. JobCenterプロセスの監視方法（Windowsの場合）

Windows版JobCenterのプロセスを監視する場合は以下のプロセスを監視してください。

起動方法	監視するプロセス
サービスから起動	%InstallDirectory%\bin\jcservice.exe
cjcpwコマンドから起動	%InstallDirectory%\bin\cluster\cjcpw.exe

jcserviceプロセスとcjcpwプロセスはJobCenterの常駐プロセスを監視します。JobCenterに異常が発生し、いずれかの常駐プロセスが停止した場合、常駐プロセスに対し停止命令を送信し、JobCenterの終了処理を行います。また、その際自身のプロセスも終了する動作となります。そのため、本プロセスを監視することでJobCenterのプロセス監視を行うことができます。

cjcpwコマンドの詳細については<コマンドリファレンス>の「4.2 cjcpw デーモンプロセスの起動と監視、停止」を参照してください。

Windows版JobCenterの常駐プロセスは以下の通りです。

表20.1 JobCenter常駐プロセス一覧（Windows）

プロセス	説明
%InstallDirectory%\bin\jcservice.exe	サービスから起動した場合に実行されるプロセスで、JobCenterの常駐プロセスの起動、停止、監視を行います。
%InstallDirectory%\bin\nqsdaemon.exe	NQS関連の統括的な役割を果たし、ジョブの実行を担うプロセスです。
%InstallDirectory%\bin\netdaemon.exe	リモートサーバからのNQS関連の通信を待ち受けするプロセスです。
%InstallDirectory%\bin\logdaemon.exe	NQS関連プロセスが出力するログ情報を書き込むプロセスです。
%InstallDirectory%\bin\jnwengine.exe	ジョブネットワークの実行制御を行うプロセスです。
%InstallDirectory%\bin\qwb.exe	NQSが実行したジョブのステータスをnqsdaemonからjnwengineに通知するプロセスです。
%InstallDirectory%\bin\trkrdr.exe	jnwengineからの要求に従ってデータベースから対象ジョブのステータスを通知するプロセスです。
%InstallDirectory%\bin\sclaunchd.exe	ジョブネットワークのスケジュール起動を行うプロセスです。
%InstallDirectory%\bin\comagent.exe	CL/Winからの接続や操作に応じた通信を受け付け、サーバ上でその操作に応じたコマンドを実行するプロセスです。
%InstallDirectory%\bin\jcdbs.exe	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プロセスです。
%InstallDirectory%\bin\jnwlauncher.exe	起動トリガ関連のプロセスを起動、停止、監視するプロセスです。
%InstallDirectory%\bin\jl_modules\jl_logmonitor.exe	起動トリガで設定したテキストログを監視するプロセスです。
%InstallDirectory%\bin\jl_modules\jl_submit.exe	起動トリガ関連のプロセスからの要求に従って、ジョブネットワークの投入を行うプロセスです。
%InstallDirectory%\bin\jl_modules\jl_notifier.exe	SystemManager GなどのJobCenter外部からの通知を拡張カスタムジョブの待ち合わせ部品に送信するプロセスです。
%InstallDirectory%\bin\jcwebserver.exe	JobCenter MG/SVのWebAPIの通信を受け付け、サーバ上でその操作に応じたコマンドを実行するプロセスです。

■daemon.confのjcwebserver_enabledパラメータの設定がfalseのサイトの場合、 jcwebserver はプロセスとして存在しません。

詳細については「5.2 デーモン設定ファイルの使用可能パラメータ」を参照してください。

20.2. JobCenterプロセスの監視方法（UNIXの場合）

UNIX版JobCenterのプロセスを監視する場合は、JobCenterをcjcpwコマンドを用いて起動します。

表20.2 cjcpwコマンド

ローカルサイトの場合	/usr/lib/nqs/cluster/cjcpw -local
クラスタサイトの場合	/usr/lib/nqs/cluster/cjcpw <サイト名> <DBディレクトリ>

cjcpwコマンドの詳細については<コマンドリファレンス>の「4.2 cjcpw デーモンプロセスの起動と監視、停止」を参照してください。

cjcpwコマンドを用いて起動すると、cjcpwプロセスがJobCenterの常駐プロセスを監視します。JobCenterに異常が発生し、いずれかの常駐プロセスが停止した場合、常駐プロセスに対し停止命令を送信し、JobCenterの終了処理を行います。また、その際自身のプロセスも終了する動作となります。

そのため、表20.2「cjcpwコマンド」で記載したコマンドで実行されたcjcpwプロセスの死活を監視することで、JobCenterのプロセス監視を行うことができます。

> nqsstart, cjcpw -c オプションでサイトを起動する場合は、表20.3「JobCenter常駐プロセス一覧（UNIX）」を監視してください。

UNIX版JobCenterの常駐プロセスは以下の通りです。

表20.3 JobCenter常駐プロセス一覧（UNIX）

プロセスイメージ名	説明
/usr/lib/nqs/nqsdaemon	NQS関連の統括的な役割を果たし、ジョブの実行を担うプロセスです。
/usr/lib/nqs/netdaemon	リモートサーバからのNQS関連の通信を待ち受けするプロセスです。
/usr/lib/nqs/logdaemon	NQS関連プロセスが出力するログ情報を書き込むプロセスです。
/usr/lib/nqs/gui/bin/jnwengine	ジョブネットワークの実行制御を行うプロセスです。
/usr/lib/nqs/gui/bin/qwb	NQSが実行したジョブのステータスをnqsdaemonからjnwengineに通知するプロセスです。
/usr/lib/nqs/gui/bin/trkrdr	jnwengineからの要求に従って、NQSのデータベースから対象ジョブのステータスを通知するプロセスです。
/usr/lib/nqs/gui/bin/jnwspawn	jnwengineから要求されたジョブを投入するプロセスです。HP-UX版、Linux版JobCenterにのみ存在します。
/usr/lib/nqs/gui/bin/sclaunchd	ジョブネットワークのスケジュール起動を行うプロセスです。
/usr/lib/nqs/jnwcaster	CL/Winや他の監視系製品へイベントの送信を行うプロセスです。
/usr/lib/nqs/combase/comagent	CL/Winからの接続や操作に応じた通信を受け付け、サーバ上でその操作に応じたコマンドを実行するプロセスです。
/usr/lib/nqs/gui/bin/jcdbs	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プロセスです。
/usr/lib/nqs/gui/bin/jnwlauncher	起動トリガ関連のプロセスを起動、停止、監視するプロセスです。
/usr/lib/nqs/gui/bin/jl_modules/jl_logmonitor	起動トリガで設定したテキストログを監視するプロセスです。
/usr/lib/nqs/gui/bin/jl_modules/jl_submit	起動トリガ関連のプロセスからの要求に従って、ジョブネットワークの投入を行うプロセスです。

プロセスイメージ名	説明
/usr/lib/nqs/gui/bin/jl_modules/jl_notifier	SystemManager GなどのJobCenter外部からの通知を拡張カスタムジョブの待ち合わせ部品に送信するプロセスです。
/usr/lib/nqs/gui/bin/jcwebserver	JobCenter MG/SVのWebAPIの通信を受け付け、サーバ上でその操作に応じたコマンドを実行するプロセスです。

■HP-UXおよびAIX環境では jcwebserver のプロセスは存在しません。

■daemon.confのjcwebserver_enabledパラメータの設定がfalseのサイトの場合、 jcwebserver はプロセスとして存在しません。

詳細については「5.2 デーモン設定ファイルの使用可能パラメータ」を参照してください。

20.2.1. CJCPW起動のOSサービス化

UNIX版におけるOS起動時のJobCenterサービス起動(ローカルサイトの自動起動)では、プロセスの監視を行いません。JobCenterプロセス監視のために、ローカルサイトの自動起動をcjcpwコマンドで行いたい場合、サービス制御の設定を変更する必要があります。

設定を変更する際、OSが提供するサービス制御がフォアグラウンドのプロセス起動に対応するか否かでcjcpwコマンドの起動方法が異なります。

■サービス制御がフォアグラウンド 起動によるサービス化をサポートする場合(systemd等)

cjcpw -local によってcjcpwコマンドを起動する

■サービス制御がフォアグラウンド 起動によるサービス化をサポートしない場合(SysVinit等)

cjcpw -local &によってcjcpw コマンドをバックグラウンド起動する

以下の章では、サービス制御の仕組みに合わせた設定例を記載します。

20.2.1.1. CJCPWによるローカルサイトの自動起動設定(systemd)

本章では、RHEL7 以降におけるサービス制御の仕組みであるsystemdを例に、cjcpwのフォアグラウンド起動設定例を記載します。本設定は、管理者(root)ユーザの権限で操作を行う必要があります。

他のサービス制御を利用する場合は、設定の変更例を適宜読み替えてください。

■本設定で編集するファイルは、サービス起動に関わるため事前にバックアップを作成することを推奨します。

■本設定は、ローカルサイトが起動していない状態で実施してください。

■本設定で編集するファイルは、JobCenter累積パッチ適用時に、インストールパッケージ同梱のものに置換する場合があります。累積パッチの適用後は本設定が有効になっていることを確認し、無効になっていた場合は再度本設定を実施する必要があります。

1. serviceファイルの編集

   サービス制御ファイルを編集します。編集時は、ファイルへの書き込み権を付与する必要があります。

   対象ファイル

```
/usr/lib/systemd/system/nqs.service
```

変更前

```
[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/etc/init.d/nqs start
ExecStop=/etc/init.d/nqs stop
```

変更後

```
[Service]
Type=simple
RemainAfterExit=no
ExecStart=/usr/lib/nqs/cluster/cjcpw -local
ExecStop=/usr/lib/nqs/cluster/cjcpw -local -stop
```

2. systemdのリロード

以下のコマンドにより、systemdのserviceファイルを再読み込みします。

```
# systemctl daemon-reload
```

以上で設定の変更は完了です。

20.2.1.2. CJCPWによるローカルサイトの自動起動設定(SysVinit互換)

本章では、RHEL6 以前および、HP-UX, AIXにおけるサービス制御の仕組みであるSysVinit(互換機能を含む)を例に、cjcpwによる起動設定例を記載します。本設定は、管理者(root)ユーザの権限で操作を行う必要があります。

他のサービス制御を利用する場合は、設定の変更例を適宜読み替えてください。

■本設定で編集するファイルは、サービス起動に関わるため事前にバックアップを作成することを推奨します。

■本設定は、ローカルサイトが起動していない状態で実施してください。

■本設定で編集するファイルは、JobCenter累積パッチ適用時に、インストールパッケージ同梱のものに置換する場合があります。累積パッチの適用後は本設定が有効になっていることを確認し、無効になっていた場合は再度本設定を実施する必要があります。

1. rcスクリプト編集

サービス制御ファイルを編集します。編集時は、ファイルへの書き込み権を付与する必要があります。

対象ファイル

■Linuxの場合

```
/etc/init.d/nqs
```

■HP-UXの場合

```
/sbin/init.d/nqs
```

■AIXの場合

```
/usr/lib/nqs/util/rc.nqs
```

サービス起動時

変更前

```
/usr/lib/nqs/nqsstart -f
```

変更後

```
cd "任意のワーキングディレクトリ"
/usr/bin/nohup /usr/lib/nqs/cluster/cjcpw -local &
```

■nohup　コマンドの出力ファイル(nohup.out)出力先を制御するため、ワーキングディレクトリの指定が必要です。

■バックグラウンド起動(&)の指定が必須です。本指定を行わないと、init によるサービス開始処理が終了しなくなります。

サービス停止時

変更前

```
/usr/lib/nqs/nqsstop
```

変更後

```
/usr/lib/nqs/cluster/cjcpw -stop -local
```

以上で設定の変更は完了です。

21. SystemManager GでJobCenterの障害情報を監視する

NECのWebSAM SystemManager G(以下、SystemManagerGと記します)のアプリケーションログ監視機能を用いて、JobCenterの障害情報を監視する方法について説明します。

21.1. 機能概要

JobCenterで障害や警告が発生した場合、予定時刻にジョブネットワークが実行されないなど運用上致命的な問題が発生する可能性があります。

障害の中にはお客様ご自身で問題解決できるものやその後の対策が可能なものがあります。本機能では、そのような障害を対象にSystemManagerGのアプリケーションログ監視機能を用いて監視します。

SystemManagerGのアプリケーションログ監視機能は、JobCenterのエラーログファイルを監視し、指定したメッセージが記録されるとSystemManagerGのメッセージとして通知します。

エラーログファイルとは、JobCenterの常駐プロセスや内部コマンドが障害情報を記録するためのファイルです。エラーログファイルの詳細および設定方法については「25.2 エラーログファイルの設定を変更する」を参照してください。

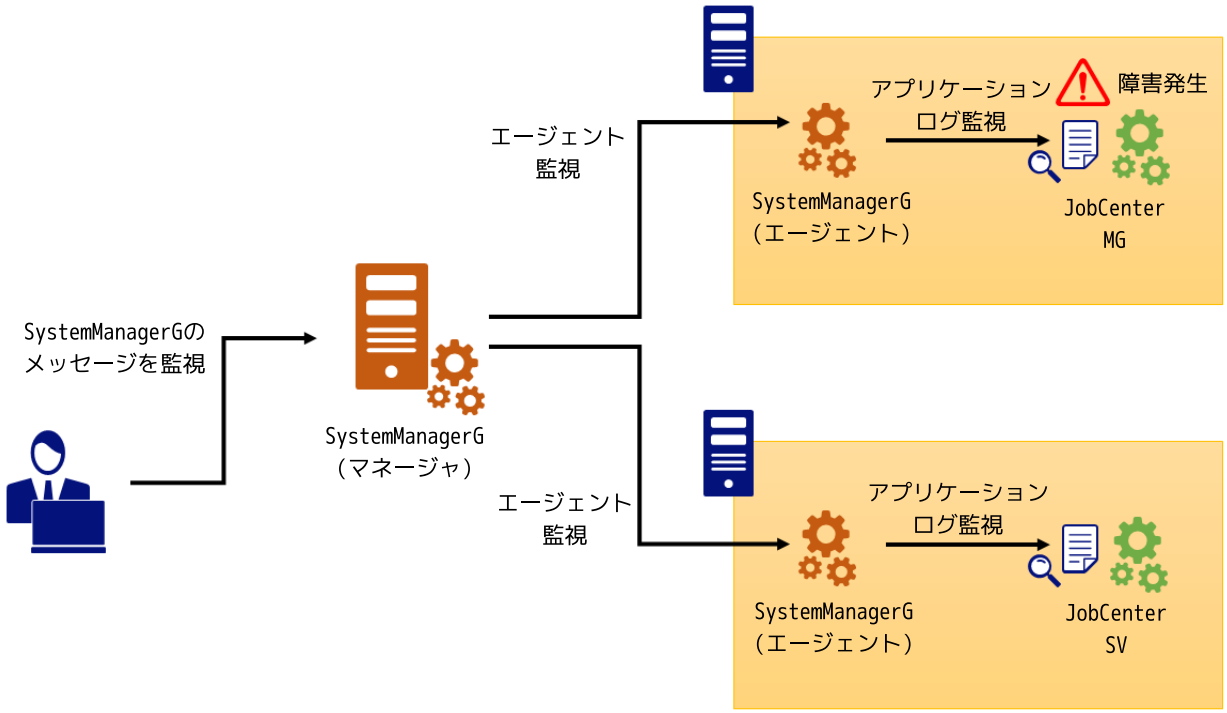

図21.1 JobCenterの障害情報監視イメージ

■エラーログファイルのフォーマットや詳細情報について、「21.3 障害情報一覧」で説明しているメッセージ以外の内容は公開しておりませんのでご了承ください。

21.2. 構築手順

SystemManagerGのアプリケーションログ監視機能を用いて、JobCenterの障害情報を監視するための環境構築手順について説明します。

本手順を実施する前に以下の製品をインストールしてください。

■SystemManagerG(マネージャ)

■SystemManagerG(エージェント)

■JobCenter

SystemManagerG(エージェント)とJobCenterは同一マシンにインストールしてください。JobCenterのインストール方法については<インストールガイド>を参照してください。SystemManagerGのインストール方法については、SystemManagerGのインストールガイドを参照してください。

■監視対象のJobCenterごとにSystemManagerG(エージェント)が必要です。例えばJobCenter MGとJobCenter SVの両方とも監視したい場合は、それぞれにSystemManagerG(エージェント)をインストールする必要があります。監視対象のログは「21.3 障害情報一覧」の各障害情報の章を参照してください。

■R16.1以降のJobCenter MGおよびSVのエラーログファイルを監視可能です。

■アプリケーションログ監視機能の設定については、本マニュアルで記載していないパラメータについても任意に設定可能です。お客様の環境に合わせて設定してください。パラメータの詳細はSystemManagerGの設定ガイドを参照してください。

1. ブラウザでSystemManagerG(マネージャ)のWeb GUIに接続し、画面上部の[監視]→[設定]→[エージェント監視]をクリックします。

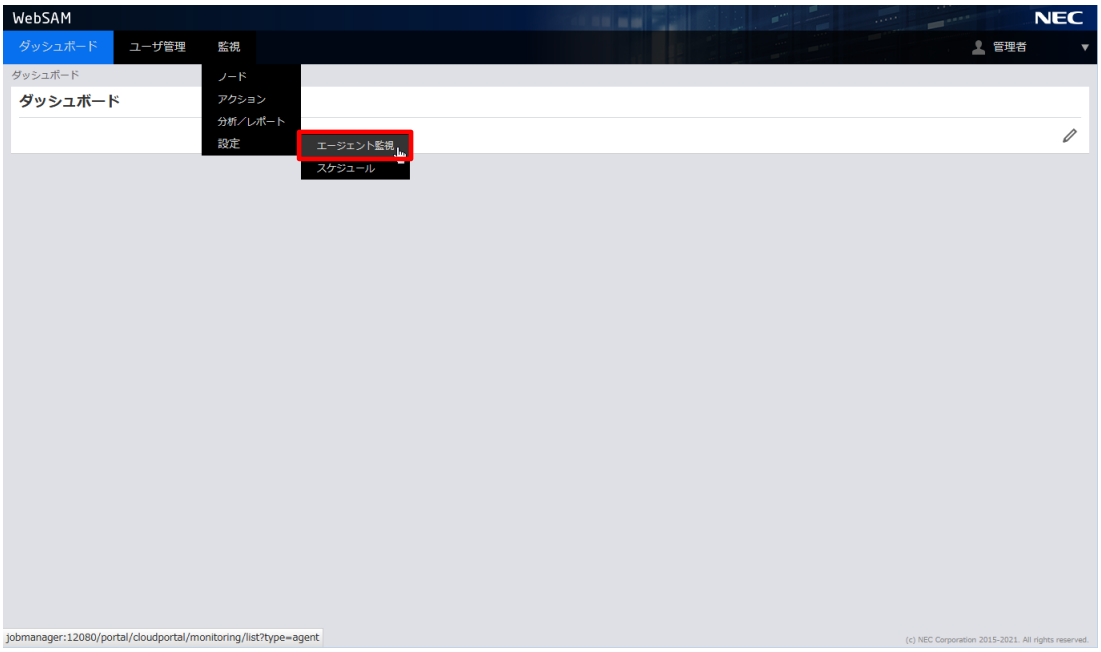

図21.2 エージェント監視の設定画面の表示方法

2. アプリケーションログ監視を設定するノード(エージェント)を選択し、[アプリケーションログ監視]をクリックします。

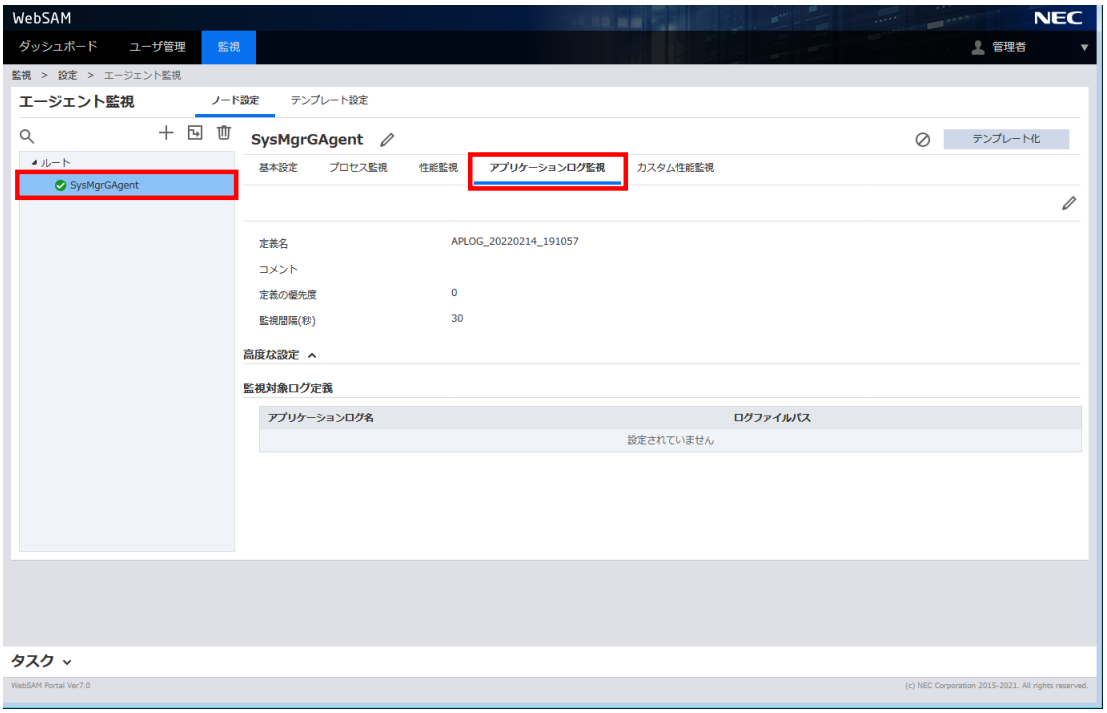

図21.3 アプリケーションログ監視の設定画面

3. 画面右側にある鉛筆アイコンをクリックし、アプリケーションログ監視の編集画面を表示します。

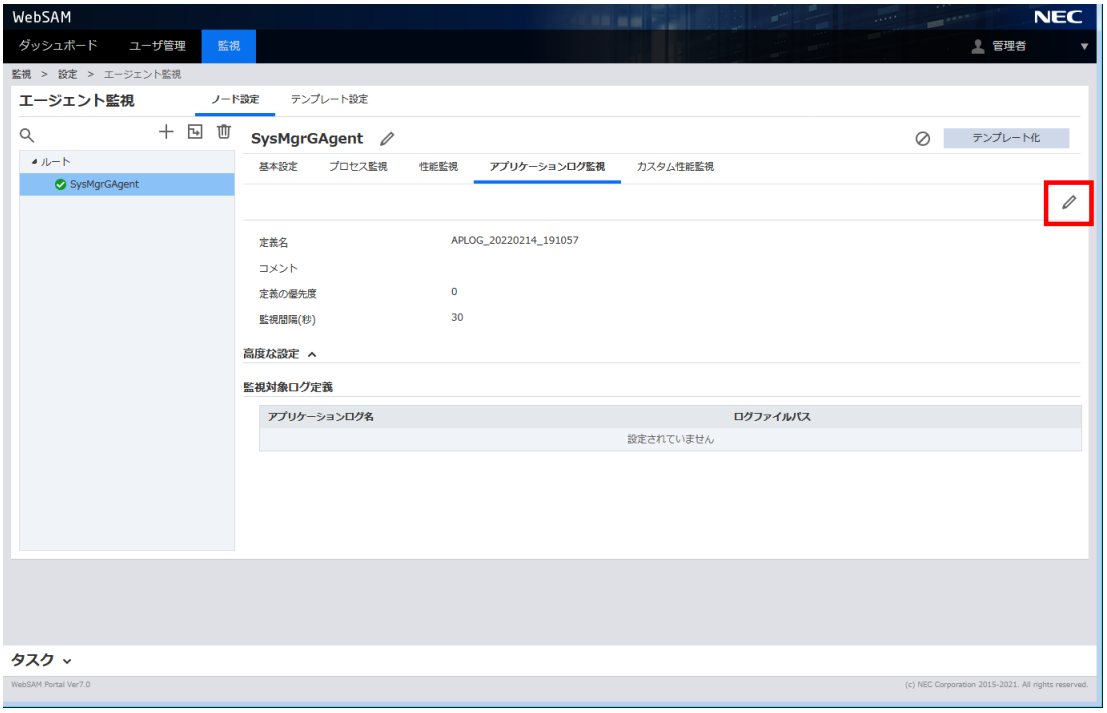

図21.4 アプリケーションログ監視の編集画面の表示方法

4. アプリケーションログ監視の各パラメータを設定します。

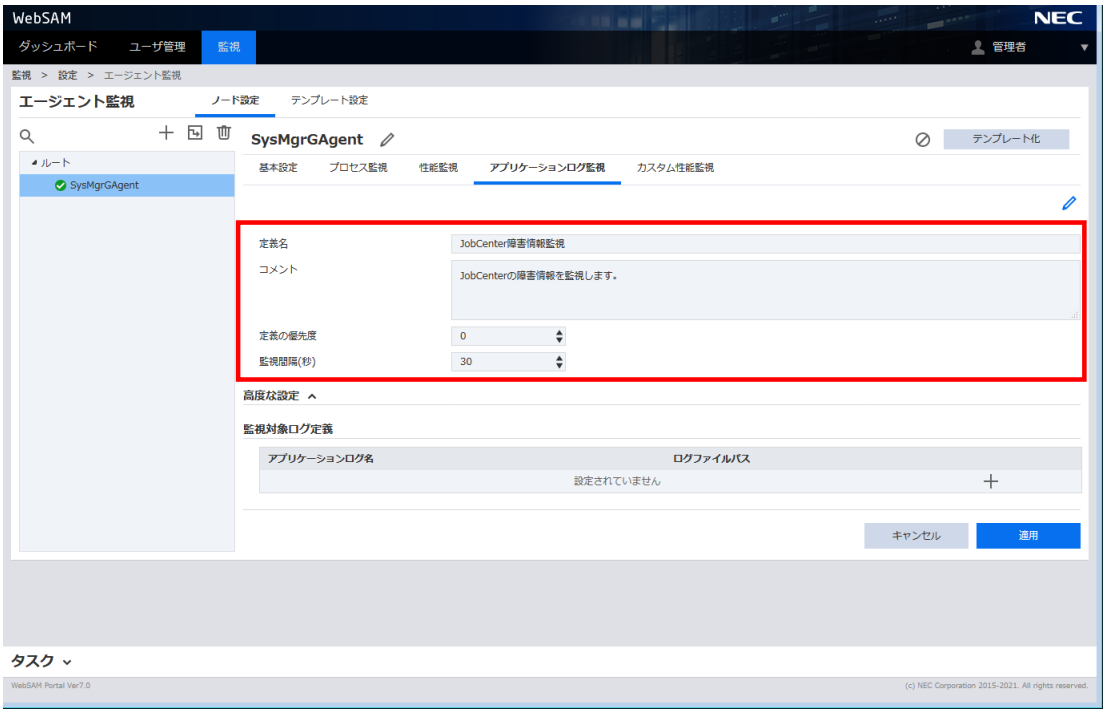

図21.5 アプリケーションログ監視の編集画面

表21.1 アプリケーションログ監視のパラメータ

項目名	説明	設定例
定義名	アプリケーションログ監視の定義名を入力します。	JobCenter障害情報監視

項目名	説明	設定例
	設定値は任意です。	
コメント	コメントを入力します。 設定値は任意です。	JobCenterの障害情報を監視します。
定義の優先度	定義の優先度を指定します。 既定値の0で問題ありません。	0
監視間隔(秒)	監視の時間間隔を秒単位で指定します。 既定値の30で問題ありません。	30

5. 監視対象ログ定義の右側にある＋アイコンをクリックし、監視対象ログ定義の編集画面を表示します。

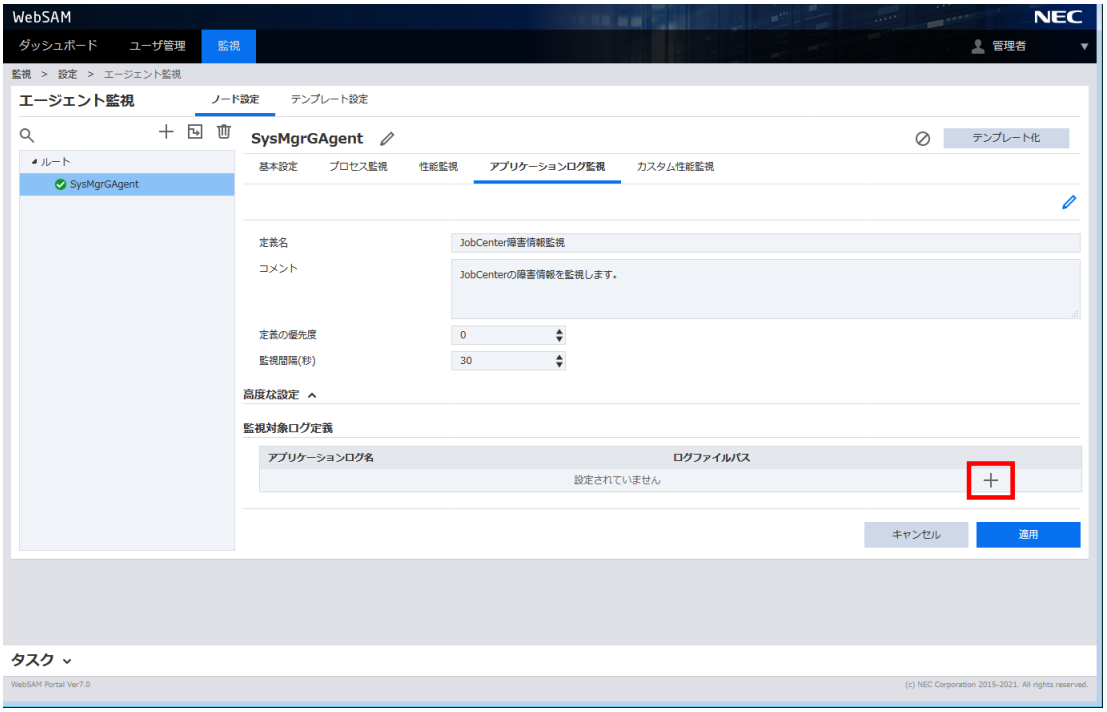

図21.6 監視対象ログ監視の編集画面の表示方法

6. 監視対象ログ定義の各パラメータを設定します。

> 監視対象ログ定義は、監視するログファイル単位で作成します。
>
> 監視する障害情報を追加する際に、既に同一のログファイルの監視対象ログ定義を作成済みの場合は、次のフィルタ定義の設定手順へ進んでください。

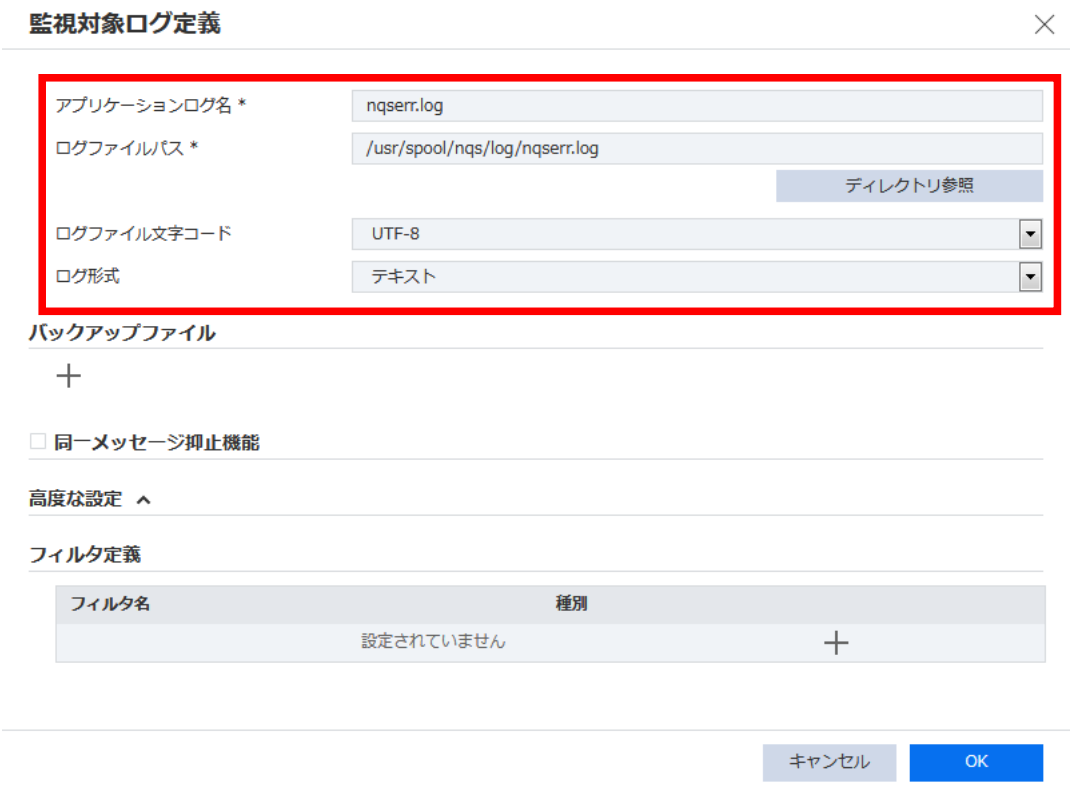

図21.7 監視対象ログ定義の編集画面

表21.2 監視対象ログ定義のパラメータ

項目名	説明	設定例
アプリケーションログ名	アプリケーションログ名の表示名を指定します。 設定値は任意です。	nqserr.log
ログファイルパス	監視対象のログファイルパスの絶対パスを指定します。 監視するJobCenterの障害情報のログファイルを指定します。ログファイルについては「21.3 障害情報一覧」の各障害情報の章を参照してください。	/usr/spool/nqs/log/nqserr.log
ログファイル文字コード	ログファイルの文字コードを指定します。 JobCenterのセットアップ言語に合わせて設定してください。 ■Windows版JobCenter JobCenterのセットアップ言語 / ログファイル文字コード 日本語 / SHIFT-JIS 英語 / ASCII ■UNIX版JobCenter	UTF-8

<table>
<tr><th>項目名</th><th>説明</th><th>設定例</th></tr>
<tr><td></td><td><table><tr><th>JobCenterのセットアップ言語</th><th>ログファイル文字コード</th></tr><tr><td>English</td><td>ASCII</td></tr><tr><td>EUC</td><td>EUC-JP</td></tr><tr><td>Shift-JIS</td><td>SHIFT-JIS</td></tr><tr><td>JP.UTF-8</td><td>UTF-8</td></tr></table></td><td></td></tr>
<tr><td>ログ形式</td><td>ログの形式を指定します。
「テキスト」を指定してください。</td><td>テキスト</td></tr>
</table>

7. フィルタ定義の右側にある＋アイコンをクリックし、フィルタ定義の編集画面を表示します。

図21.8 フィルタ定義の編集画面の表示方法

8. フィルタ定義の各パラメータを設定します。

図21.9 フィルタ定義の編集画面

表21.3 フィルタ定義のパラメータ

項目名	説明	設定例
フィルタ名	フィルタ名を指定します。 設定値は任意です。	キューがSTOP状態
種別	フィルタの種別を指定します。 「格納」を指定してください。	格納
メッセージテキスト	通知対象とするログを正規表現で指定します。 監視するJobCenterの障害情報の監視文字列を指定します。監視文字列については「21.3 障害情報一覧」の各障害情報の章を参照してください。	Since abnormalities occurred, the queue.*was stopped

9. 続けて「発行メッセージのカスタマイズ」にチェックを入れて、「重要度」に「異常」を指定します。

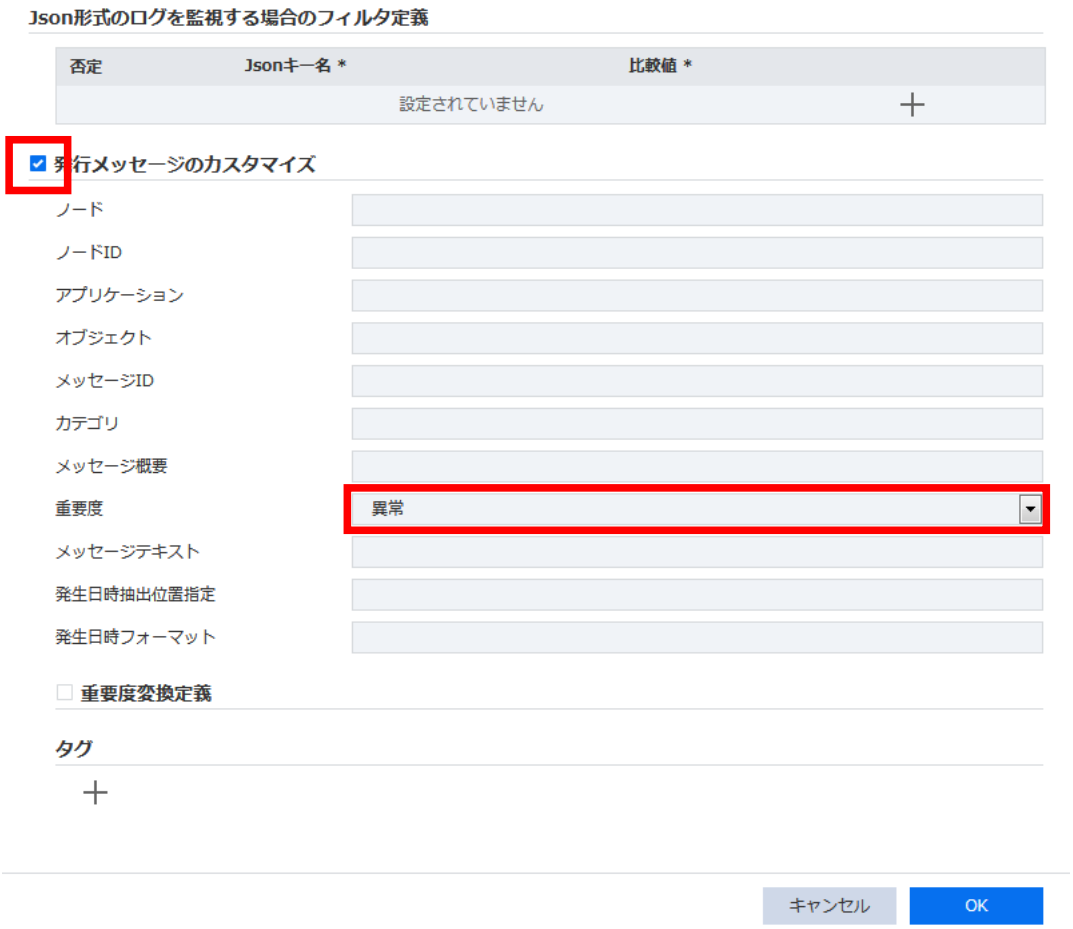

図21.10 発行メッセージの重要度の設定

10. 各設定画面で「OK」ボタンをクリックし、設定を保存してください。

    設定した障害がJobCenterで発生すると、アプリケーションログ監視を設定したノード(エージェント)からメッセージが通知されます。

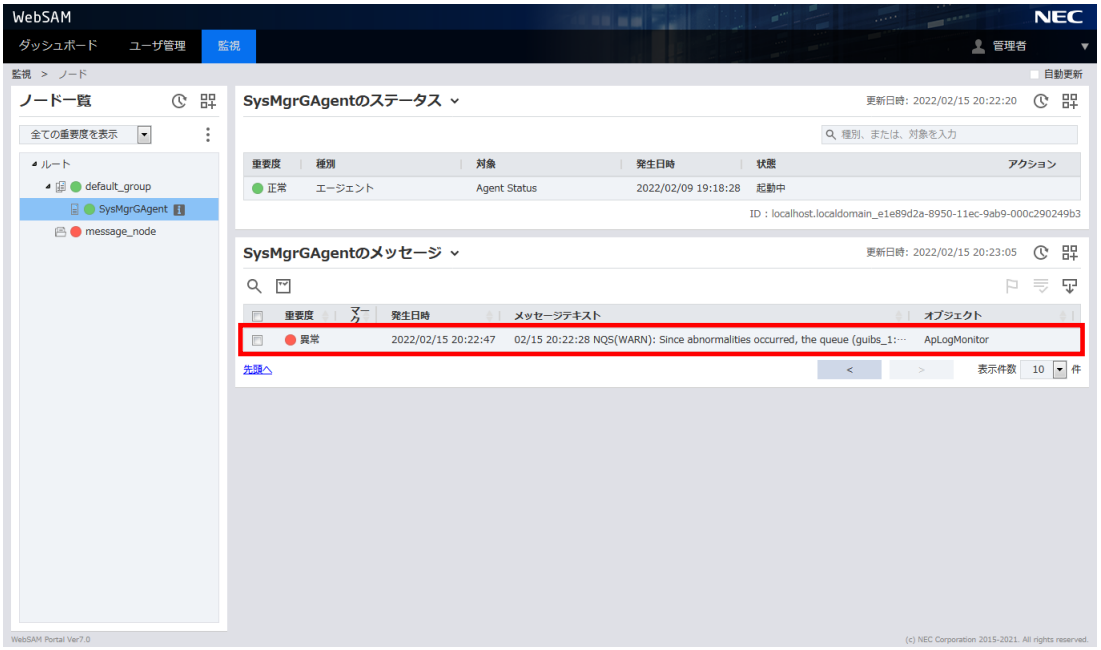

図21.11 SystemManagerGのメッセージ監視

アプリケーションログ監視で通知されるメッセージはカスタマイズ可能です。

手順9のフィルタ定義の設定画面にある「発行メッセージのカスタマイズ」で設定します。詳細はSystemManagerGの設定ガイドを参照してください。

■JobCenterの障害情報「21.3.2 キューがSTOP状態」のカスタマイズ例

「発行メッセージのカスタマイズ」の「メッセージテキスト」に以下の値を設定します。

```
異常が発生したためキューを停止しました（$SPLIT("()",4)$）
```

「$SPLIT("A",B)$」はログ本文から特定の文字列を抽出する置換文字列です。Aはセパレータの文字、Bは抽出位置を意味します。詳細はSystemManagerGの設定ガイドを参照してください。

「メッセージテキスト」に設定したフォーマットに従って、メッセージが通知されます。

```
異常が発生したためキューを停止しました（<キュー名>:status=<キューのステータス>）
```

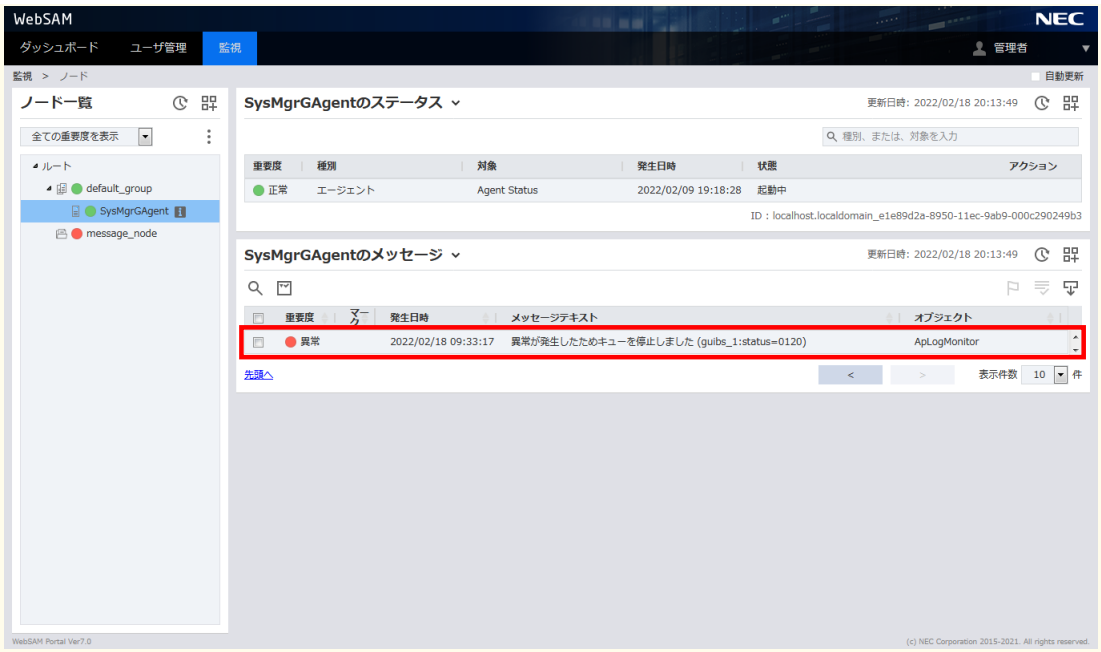

図21.12 メッセージテキストのカスタマイズ例

21.3. 障害情報一覧

SystemManagerGのアプリケーションログ監視機能を用いて監視可能なJobCenterの障害情報は以下のとおりです。

監視したい障害情報の章にある「監視するメッセージ」の表を参考にアプロケーションログ監視を設定してください。「監視するメッセージ」の表の見方は「21.3.1 監視するメッセージの表の見方」に記載していますので最初にご確認ください。

表21.4 監視可能な障害情報一覧

障害内容	原因
単位ジョブが実行中のままになる	「21.3.2 キューがSTOP状態」
	「21.3.3 MG→SVのTCP通信が失敗する」
	「21.3.4 SVのOSまたはJobCenterが停止している」
	「21.3.5 SV→MGのTCP通信が失敗する」
	「21.3.6 NQSの発信ポートが不足している」
単位ジョブの投入に失敗して異常終了する	「21.3.7 環境変数が大きすぎる」
	「21.3.8 ユーザマッピングができていない」
	「21.3.9 ユーザマッピング先のユーザが存在しない」
	「21.3.10 MGにてSVの名前解決が出来ない」
	「21.3.11 SVにてMGの名前解決が出来ない」
	「21.3.12 MG→SV転送時にリクエストIDの重複が発生した」
トラッカが予定確定のままになる	「21.3.13 同時起動可能数を超えた」
JNWの投入に失敗する	「21.3.14 ジョブ実行ユーザに指定したユーザが存在しない」
トラッカを参照できない	「21.3.15 2GBを超えるトラッカのアーカイブに失敗」

21.3.1. 監視するメッセージの表の見方

各障害情報の「監視するメッセージ」の章に記載している表の見方について説明します。

21.3.1.1. 監視するメッセージ

監視するメッセージの1行分の文字列です。

メッセージテキストをカスタマイズしていない場合は、通知されるメッセージの「メッセージテキスト」に表示される文字列となります。

以下のログファイルはメッセージの先頭に日付情報があります。実際に障害が発生した日付に読み替えてください。

■UNIX版JobCenterのnqserr.log

- 日付情報の書式

```
MM/DD hh:mm:ss
```

MM/DDは月日(0埋めあり)、hh:mm:ssは時間分(0埋めあり)を表します。

- 例：1月1日1時30分0秒の場合

```
01/01 01:30:00
```

■Windows版JobCenterのnqserr.log

- 日付情報の書式

```
MM/DD/YY hh:mm:ss
```

MM/DDは月日年(月日は0埋めなし、年は下2桁かつ0埋めあり)、hh:mm:ssは時間分(0埋めあり)を表します。

- 例：2001年1月1日1時30分0秒の場合

```
 1/ 1/01 01:30:00
```

21.3.1.2. フィルタ定義のメッセージテキスト

SystemManagerGのフィルタ定義の「メッセージテキスト」に指定する文字列です。

設定方法は「21.2 構築手順」の手順8を参照してください。

21.3.1.3. コンポーネント

監視するメッセージが記録されるJobCenterです。

21.3.1.4. ログファイル

SystemManagerGの監視対象ログ定義の「ログファイルパス」に指定するログファイル名です。

「ログファイルパス」は絶対パスで指定する必要がありますのでご注意ください。

設定方法は「21.2 構築手順」の手順6を参照してください。

ログファイルの格納先は以下のとおりです。

■UNIX版JobCenter

ローカルサイト	/usr/spool/nqs/log
クラスタサイト	<JobCenterDBパス>/nqs/log

■Windows版JobCenter

ローカルサイト	%InstallDirectory%\spool\log\error
クラスタサイト	<JobCenterDBパス>\spool\log\error

%InstallDirectory%はJobCenterのインストールディレクトリです。初期値でインストールした場合はC:\JobCenter\SVとなります。

21.3.2. キューがSTOP状態

21.3.2.1. 障害内容

単位ジョブの投入キューが停止状態の場合、単位ジョブのステータスはSUBMITの状態で待機します。キューが開始状態になるまで単位ジョブは開始しないため、予定したジョブが実行されない問題が発生します。

キューが停止状態となる原因は以下があります。本障害情報はJobCenterの障害が原因で発生する3の場合を対象に監視します。

■1. ユーザ操作によってキューを停止した

例えば、JobCenter管理者ユーザでCL/Winで接続し、キュー一覧画面から対象のキューを停止した場合です。

■2. daemon.confの「NQSDAEMON_OPT=-s」パラメータを設定した状態でJobCenterを起動した

daemon.confの「NQSDAEMON_OPT=-s」パラメータを設定した状態でJobCenterを起動すると、全てのキューが停止状態となります。

「NQSDAEMON_OPT=-s」パラメータの詳細は「5.2　デーモン設定ファイルの使用可能パラメータ」を参照してください。

■3. JobCenterによってキューが停止された

ジョブネットワーク投入時に予期せぬエラーが発生した際に、JobCenterがキューを停止する場合があります。

JobCenterをインストールしたマシンのリソース不足などが原因で発生する場合があります。

> 1と3についてはイベント連携機能でも監視することが可能です。UNIX版JobCenterの場合はjnwsv.que.ustopとjnwsv.que.sstop、Windows版JobCenterの場合はQUE_USTOPとQUE_SSTOPが対応するイベントとなります。イベント連携の詳細は12章　「イベント連携」　を参照してください。

21.3.2.2. 監視するメッセージ

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Since abnormalities occurred, the queue (<キュー名>:status=<キューのステータス>) was stopped(<終了コード>). Please check the state of queue, job, system, etc. And start queue after resolving the problem.	
フィルタ定義のメッセージテキスト	
Since abnormalities occurred, the queue.*was stopped	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ
MM/DD/YY hh:mm:ss NQS(WARN): Since abnormalities occurred, the queue (<キュー名>:status=<キューのステータス>) was stopped(<終了コード>). Please check the state of queue, job, system, etc. And start queue after resolving the problem.
フィルタ定義のメッセージテキスト

Since abnormalities occurred, the queue.*was stopped	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	nqserr.log

ジョブネットワーク投入時に予期せぬエラーが発生し、JobCenterがキューを停止した場合に記録されます。

JobCenterをインストールしたマシンのリソース不足が原因で発生する場合があります。

■「監視するメッセージ」の「NQS(WARN):」の右側にある空白について、UNIX版JobCenterは半角スペース2つです。Windows版JobCenterは半角スペース1つです。

■<キュー名>は停止したキュー名です。

■<キューのステータス>はキューのステータスを表す値です。

サポート窓口にて障害の原因究明を行う際に利用します。ステータスの詳細情報は非公開ですのでご了承ください。

■<終了コード>はJobCenterの内部処理の終了コードです。

サポート窓口にて障害の原因究明を行う際に利用します。終了コードの詳細情報は非公開ですのでご了承ください。

21.3.2.3. 解決方法

対象のキューを開始状態にしてください。開始状態にする方法は以下のとおりです。

■CL/Winを使用する場合

JobCenter管理者ユーザでCL/Winで接続し、キュー一覧画面から対象のキューを右クリック→[開始]を選択します。

■qmgrコマンドを使用する場合

qmgrコマンドのSTArt Queueサブコマンドで対象のキューを開始状態にします。qmgrコマンドの詳細は<コマンドリファレンス>の「3.13 qmgr 構成管理および運用管理」を参照してください。

本障害が頻発する場合は、サポート窓口へお問い合わせください。

UNIX版JobCenterの場合でDefaultNetQueキューが停止状態になった場合、キュー一覧画面に表示されないため、CL/Winからキューを開始状態にできません。

DefaultNetQueキューを開始状態にする場合は、qmgrコマンドを使用してください。

21.3.3. MG→SVのTCP通信が失敗する

21.3.3.1. 障害内容

JobCenter MGとJobCenter SV間のネットワークに障害が発生した場合、単位ジョブのステータスがSUBMITのまま実行されない状態になります。

本障害内容は、JobCenter MGからJobCenter SVへの通信時を対象にしています。JobCenter SVからJobCenter MGへの通信時については、「21.3.5 SV→MGのTCP通信が失敗する」を参照してください。

21.3.3.2. 監視するメッセージ

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01456 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01456 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01044 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01044 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01406 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01406 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01433 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01433 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01506 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]

フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01506 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01406 by COMMIT,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01406 by COMMIT	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01433 by COMMIT,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01433 by COMMIT	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01506 by COMMIT,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01506 by COMMIT	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01406 by DELIVER,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01406 by DELIVER	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01433 by DELIVER,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01433 by DELIVER	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01506 by DELIVER,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01506 by DELIVER	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01456 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01456 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01044 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01044 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01406 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01406 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01433 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01433 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01506 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]

フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01506 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01406 by COMMIT,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01406 by COMMIT	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01433 by COMMIT,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01433 by COMMIT	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01506 by COMMIT,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01506 by COMMIT	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01406 by DELIVER,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01406 by DELIVER	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01433 by DELIVER,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01433 by DELIVER	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01506 by DELIVER, (<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01506 by DELIVER	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

リモートジョブ投入時に、JobCenter　MGからJobCenter　SVへの通信時に問題が発生した場合に記録されます。

ネットワークに障害が発生したタイミングによってメッセージが異なりますので、上記のメッセージ全てを監視するように設定してください。

■「監視するメッセージ」の「NQS(WARN):」の右側にある空白について、UNIX版JobCenterは半角スペース2つです。Windows版JobCenterは半角スペース1つです。

■<リクエストID(seqno)>は、リクエストごとに採番されたIDです。

■<MGのマシンID>は、JobCenter MGのマシンIDです。

■<SV側のキュー名>は、投入しようとしたJobCenter SV側のキューの名前です。

■<SVのマシンID>は、投入しようとしたJobCenter SVのマシンIDです。

21.3.3.3. 解決方法

JobCenter MGとJobCenter SV間のネットワークに障害が発生していないか確認してください。

21.3.4. SVのOSまたはJobCenterが停止している

21.3.4.1. 障害内容

JobCenter　SVがインストールされたマシンまたはJobCenter　SVが停止している場合、単位ジョブのステータスがSUBMITのまま実行されない状態になります。

21.3.4.2. 監視するメッセージ

21.3.4.2.1. SVがインストールされたマシンが停止している場合

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN):　Pipeclient: Receive unfortune TCM=01033 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01033 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01033 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01033 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

リモートジョブ投入時に、JobCenter SVがインストールされたマシンが停止している場合に記録されます。

■「監視するメッセージ」の「NQS(WARN):」の右側にある空白について、UNIX版JobCenterは半角スペース2つです。Windows版JobCenterは半角スペース1つです。

■<リクエストID(seqno)>は、リクエストごとに採番されたIDです。

■<MGのマシンID>は、JobCenter MGのマシンIDです。

■<SV側のキュー名>は、投入しようとしたJobCenter SV側のキューの名前です。

■<SVのマシンID>は、投入しようとしたJobCenter SVのマシンIDです。

21.3.4.2.2. SVが停止している場合

■UNIX版JobCenter

監視するメッセージ
MM/DD hh:mm:ss NQS(WARN):　Pipeclient: Receive unfortune TCM=01461 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]
フィルタ定義のメッセージテキスト
Receive unfortune TCM=01461 by QUEREQ

コンポーネント	JobCenter MG
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01461 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01461 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

リモートジョブ投入時に、JobCenter SVが停止している場合に記録されます。

■「監視するメッセージ」の「NQS(WARN):」の右側にある空白について、UNIX版JobCenterは半角スペース2つです。Windows版JobCenterは半角スペース1つです。

■<リクエストID(seqno)>は、リクエストごとに採番されたIDです。

■<MGのマシンID>は、JobCenter MGのマシンIDです。

■<SV側のキュー名>は、投入しようとしたJobCenter SV側のキューの名前です。

■<SVのマシンID>は、投入しようとしたJobCenter SVのマシンIDです。

21.3.4.3. 解決方法

JobCenter SVがインストールされたマシンまたはJobCenter SVが停止していないか確認してください。

21.3.5. SV→MGのTCP通信が失敗する

21.3.5.1. 障害内容

JobCenter MGとJobCenter SV間のネットワークに障害が発生した場合、単位ジョブのステータスがSUBMITのまま実行されない状態になります。

本障害内容は、JobCenter　SVからJobCenter　MGへの通信時を対象にしています。JobCenter　MGからJobCenter SVへの通信時については、「21.3.3 MG→SVのTCP通信が失敗する」を参照してください。

21.3.5.2. 監視するメッセージ

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): Netclient:do_stdout:trythere fail	
フィルタ定義のメッセージテキスト	
Netclient:do_stdout:trythere fail	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): Netclient:do_stderr:trythere fail	
フィルタ定義のメッセージテキスト	
Netclient:do_stderr:trythere fail	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): idc_connect: error at establishasync	
フィルタ定義のメッセージテキスト	
idc_connect: error at establishasync	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): Netclient:do_stdout:trythere fail	
フィルタ定義のメッセージテキスト	
Netclient:do_stdout:trythere fail	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

監視するメッセージ
MM/DD/YY hh:mm:ss NQS(ERROR): Netclient:do_stderr:trythere fail
フィルタ定義のメッセージテキスト
Netclient:do_stderr:trythere fail

コンポーネント	JobCenter SV
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): idc_connect: error at establishasync	
フィルタ定義のメッセージテキスト	
idc_connect: error at establishasync	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

リモートジョブ投入時に、JobCenter　SVからJobCenter　MGへの通信時に問題が発生した場合に記録されます。

ネットワークに障害が発生したタイミングによってメッセージが異なりますので、上記のメッセージ全てを監視するように設定してください。

21.3.5.3. 解決方法

JobCenter MGとJobCenter SV間のネットワークに障害が発生していないか確認してください。

21.3.6. NQSの発信ポートが不足している

21.3.6.1. 障害内容

リモートジョブ投入を実行する際、JobCenter　MGとJobCenterSV間でジョブの転送や実行結果の受け渡しを行うために、ジョブ1つに対して1つのポートを使用します。

JobCenterで実行中のジョブが大量に存在する場合、使用するポートを全て使い果たしてしまい、リモートジョブ投入が失敗します。

また、JobCenter MGとJobCenterSV間のネットワーク上にルータやFireWallが存在すると、無通信状態を検出したルータが片側のみセッション切断を行い、ハーフオープンセッション状態が発生してTCP/IP通信上の問題が発生する場合があります。その場合、実行中のジョブが少ない場合でも、本障害内容が発生する場合があります。

NQSの発信ポートの詳細および使用するポートについては、「2.1　JobCenterで使用するTCPポート」の「2.1.1 NQS」を参照してください。

21.3.6.2. 監視するメッセージ

21.3.6.2.1. MGからSVへのジョブ転送時

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): pipeclient(): error at bind.(no port) (errno=<エラー番号>,tryport=<ポート番号>,desthostname=<接続先のサイト名>)	
フィルタ定義のメッセージテキスト	
pipeclient\(\): error at bind\.\(no port\)	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): Nqslib:estblshreq(): error at NMS_CheckPort.(no port) (errno=<エラー番号>,tryport=<ポート番号>,desthostname=<接続先のサイト名>)	
フィルタ定義のメッセージテキスト	
estblshreq\(\): error at NMS_CheckPort\.\(no port\)	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

NQSの発信ポートを全て使い果たしてしまい、リモートジョブ投入が失敗した場合に記録されます。

本メッセージは、JobCeter MGからJobCenter SVへジョブを転送するタイミングで発生した場合に記録されます。JobCenter MGのログファイルを監視してください。

■<エラー番号>は、JobCenter内部で実行しているAPIのerrnoです。

サポート窓口にて障害の原因究明を行う際に利用します。

■<ポート番号>は、使用しようとしたポート番号です。

■<接続先のサイト名>は、接続先のJobCenterのサイト名です。

本メッセージの場合、JobCenter SVのサイト名となります。

21.3.6.2.2. SVからMGへのジョブ実行結果の転送時

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): estblshout(): error at bind.(no port) (errno=<エラー番号>,tryport=<ポート番号>,desthostname=<接続先のサイト名>)	
フィルタ定義のメッセージテキスト	
estblshout\(\): error at bind\.\(no port\)	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): Nqslib:estblshout(): error at NMS_CheckPort.(no port) (errno=<エラー番号>,tryport=<ポート番号>,desthostname=<接続先のサイト名>)	
フィルタ定義のメッセージテキスト	
Too many environment variables or environment variable is too large to run batch request	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): Nqslib:estblshout(): error at bind.(no port) (errno=<エラー番号>,tryport=<ポート番号>,desthostname=<接続先のサイト名>)	
フィルタ定義のメッセージテキスト	
estblshout\(\): error at bind\.\(no port\)	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

NQSの発信ポートを全て使い果たしてしまい、リモートジョブ投入が失敗した場合に記録されます。

本メッセージは、JobCeter SVからJobCenter MGへジョブ実行結果を転送するタイミングで発生した場合に記録されます。JobCenter SVのログファイルを監視してください。

■<エラー番号>は、JobCenter内部で実行しているAPIのerrnoです。

サポート窓口にて障害の原因究明を行う際に利用します。

■<ポート番号>は、使用しようとしたポート番号です。

■<接続先のサイト名>は、接続先のJobCenterのサイト名です。

本メッセージの場合、JobCenter MGのサイト名となります。

21.3.6.2.3. MGとMG間の共有セッション接続時

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): establishasync_sub: error at bind.(no port) (errno=<エラー番号>,tryport=<ポート番号>,desthostname=<接続先のサイト名>)	
フィルタ定義のメッセージテキスト	
establishasync_sub: error at bind\.\(no port\)	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): establishasync_sub: error at connect.(no port) (errno=<エラー番号>,tryport=<ポート番号>,desthostname=<接続先のサイト名>)	
フィルタ定義のメッセージテキスト	
establishasync_sub: error at connect\.\(no port\)	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): Nqsdaemon:establishasync_sub: error at bind.(no port) (errno=<エラー番号>,tryport=<ポート番号>,desthostname=<接続先のサイト名>)	
フィルタ定義のメッセージテキスト	
establishasync_sub: error at bind\.\(no port\)	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): Nqsdaemon:establishasync_sub: error at connect.(no port) (errno=<エラー番号>,tryport=<ポート番号>,desthostname=<接続先のサイト名>)	
フィルタ定義のメッセージテキスト	
establishasync_sub: error at connect\.\(no port\)	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	nqserr.log

NQSの発信ポートを全て使い果たしてしまい、リモートジョブ投入が失敗した場合に記録されます。

本メッセージは、JobCeter SVとJobCenter MG間の共有セッションに接続するタイミングで発生した場合に記録されます。JobCenter MGとJobCenter SV両方のログファイルを監視してください。

■<エラー番号>は、JobCenter内部で実行しているAPIのerrnoです。

サポート窓口にて障害の原因究明を行う際に利用します。

■<ポート番号>は、使用しようとしたポート番号です。

■<接続先のサイト名>は、接続先のJobCenterのサイト名です。

21.3.6.3. 解決方法

リモートジョブ投入で実行するジョブの量が集中しないように、スケジュールを見直してください。

実行中のジョブが少ない状態で本障害が発生した場合、ハーフオープンセッションとなっているポートがないか確認してください。ハーフオープンセッションの回避策については、「2.1　JobCenterで使用するTCPポート」の「2.1.1 NQS」を参照してください。

21.3.7. 環境変数が大きすぎる

21.3.7.1. 障害内容

ジョブを実行時する際の環境変数のサイズは上限があります。設定可能なサイズの上限を超えている場合、ジョブの実行に失敗します。

環境変数のサイズの上限は、JobCenter MGおよびJobCenter SVのコンポーネントに依存します。JobCenterにおける環境変数の取り扱いおよびサイズの上限については、15章 「ジョブ実行時の環境変数の取り扱い」を参照してください。

21.3.7.2. 監視するメッセージ

21.3.7.2.1. MGからSVへのジョブ転送時

■UNIX版JobCenter

監視するメッセージ	
svunit_RunUjSubmit(<行数>):qsub exec error:<部品名> of uid=<ユーザID>:<トラッカID>: Too many environment variables to run batch request.	
フィルタ定義のメッセージテキスト	
qsub exec error:.*: Too many environment variables to run batch request	
コンポーネント	JobCenter MG
ログファイル	jeerr.log

監視するメッセージ	
svunit_RunUjSubmitEnd(<行数>):submit command error:<部品名> of uid=<ユーザID>:<トラッカID>:spawn: Too many environment variables to run batch request.	
フィルタ定義のメッセージテキスト	
submit command error:.*:spawn: Too many environment variables to run batch request	
コンポーネント	JobCenter MG
ログファイル	jeerr.log

■Windows版JobCenter

監視するメッセージはありません。

JobCenter MGがUNIX版JobCenterの場合、MGからSVへのジョブ転送時に環境変数の合計サイズに上限があります。

JobCenterにおける環境変数の取り扱いおよびサイズの上限については、15章 「ジョブ実行時の環境変数の取り扱い」 を参照してください。

JobCenter MGがWindows版JobCenterの場合、監視するメッセージはありません。「21.3.7.2.2 SVでのジョブ実行時」のメッセージのみ監視してください。

■svunit_RunUjSubmit(<行数>)およびsvunit_RunUjSubmitEnd(<行数>)はJobCenterの内部処理の発生箇所を表す情報です。

詳細情報は非公開ですのでご了承ください。

■<部品名>は環境変数のサイズの上限を超えた部品名です。

■<ユーザID>は対象トラッカのジョブ実行ユーザのUIDです。

■<トラッカID>は対象トラッカのトラッカIDです。

「<ジョブネットワーク名>.YYYYMMDDhhmmss」の形式で表示されます。

21.3.7.2.2. SVでのジョブ実行時

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): shepherd: Too many environment variables to run batch request. (mid=<マシンID>,seqno=<リクエストID>)	
フィルタ定義のメッセージテキスト	
shepherd: Too many environment variables to run batch request	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): Shepherd:shepherdprocess: Too many environment variables or environment variable is too large to run batch request. (mid=<マシンID>,seqno=<リクエストID>)	
フィルタ定義のメッセージテキスト	
estblshout\(\): error at NMS_CheckPort\.\(no port\)	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

環境変数のサイズの上限を超えていたため、ジョブの実行に失敗した場合に記録されます。

設定可能なサイズの上限は、ジョブ実行マシンのコンポーネントに依存します。JobCenterにおける環境変数の取り扱いおよびサイズの上限については、15章 「ジョブ実行時の環境変数の取り扱い」 を参照してください。

■<マシンID>は、JobCenter MGのマシンIDです。

■<リクエストID>は、リクエストごとに採番されたIDです。

21.3.7.3. 解決方法

実行に失敗した単位ジョブの環境変数の設定を確認してください。

JobCenterにおける環境変数の取り扱いおよびサイズの上限については、15章 「ジョブ実行時の環境変数の取り扱い」 を参照してください。

21.3.8. ユーザマッピングができていない

21.3.8.1. 障害内容

単位ジョブをSVのキューに投入する際、ユーザマッピングができていない場合、単位ジョブが異常終了します。

21.3.8.2. 監視するメッセージ

21.3.8.2.1. ユーザマッピングの設定がない場合

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01454 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01454 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Netserver: Failed to get user mapping from nmap. (ruser=<MG側のユーザ名>, ruid=<MG側のUID>, rhost=<MGのサイト名>, rmid=<MGのマシンID>, mapmode=<ユーザマッピングのモード>)	
フィルタ定義のメッセージテキスト	
Netserver: Failed to get user mapping from nmap	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01454 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01454 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Netserver(PID=<netdaemonのPID>): Failed to get user mapping from nmap. (ruser=<MG側のユーザ名>, ruid=<MG側のUID>, rhost=<MGのサイト名>, rmid=<MGのマシンID>, mapmode=<ユーザマッピングのモード>)	
フィルタ定義のメッセージテキスト	
Failed to get user mapping from nmap	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

JobCenter SVとのユーザマッピングができていない状態で、ジョブをSVに投入した際、記録されます。

■UNIX版JobCenter SVでは、「21.3.9 ユーザマッピング先のユーザが存在しない」と同一のメッセージが出力されます。UNIX版JobCenter SVのメッセージが通知された場合、「21.3.9 ユーザマッピング先のユーザが存在しない」も併せて確認してください。

■「監視するメッセージ」の「NQS(WARN):」の右側にある空白について、UNIX版JobCenterは半角スペース2つです。Windows版JobCenterは半角スペース1つです。

■<リクエストID(seqno)>は、リクエストごとに採番されたIDです。

■<MGのマシンID>は、JobCenter MGのマシンIDです。

■<SV側のキュー名>は、投入しようとしたJobCenter SV側のキューの名前です。

■<SVのマシンID>は、投入しようとしたJobCenter SVのマシンIDです。

■<MG側のユーザ名>は、JobCenter MG側のユーザの名前です。

■<MG側のUID>は、JobCenter MG側のユーザのユーザIDです。

■<MGのサイト名>は、JobCenter MGのサイト名です。

■<ユーザマッピングのモード>は、ユーザマッピングの種別です。詳細は<NQS機能利用の手引き>の「6.5.1.2 ユーザに関するネットワーク環境」を参照してください。

ユーザマッピングのモードと表示される値の対応は以下のとおりです。

モード	表示される値
TYPE1	0
TYPE2	1
TYPE3	2

■<netdaemonのPID>は、netdaemonのプロセスIDです。

21.3.8.3. 解決方法

ユーザマッピングを実施してください。

ユーザマッピングの詳細については、「3.2 ユーザの関連付けを行う（ユーザマッピング）」を参照してください。

21.3.9. ユーザマッピング先のユーザが存在しない

21.3.9.1. 障害内容

単位ジョブをSVのキューに投入する際、ユーザマッピング先のユーザが存在しない場合、単位ジョブが異常終了します。

21.3.9.2. 監視するメッセージ

21.3.9.2.1. ユーザマッピング先のユーザが存在しない場合

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Netserver: Failed to get user mapping from nmap. (ruser=<MG側のユーザ名>, ruid=<MG側のUID>, rhost=<MGのサイト名>, rmid=<MGのマシンID>, mapmode=<ユーザマッピングのモード>)	
フィルタ定義のメッセージテキスト	
Netserver: Failed to get user mapping from nmap	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Nqslib:NSGenerateSecurity:LookupAccountName() error; objname=<ユーザ名> (ec=1332)	
フィルタ定義のメッセージテキスト	
NQS\(WARN\): Nqslib:NSGenerateSecurity:LookupAccountName\(\) error	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

ユーザマッピング先のユーザが存在しない状態で、ジョブをSVに投入した際、記録されます。

■UNIX版JobCenter SVでは、「21.3.8 ユーザマッピングができていない」と同一のメッセージが出力されます。UNIX版JobCenter SVのメッセージが通知された場合、「21.3.8 ユーザマッピングができていない」も併せて確認してください。

■「監視するメッセージ」の「NQS(WARN):」の右側にある空白について、UNIX版JobCenterは半角スペース2つです。Windows版JobCenterは半角スペース1つです。

■<MG側のユーザ名>は、JobCenter MG側のユーザの名前です。

■<MG側のUID>は、JobCenter MG側のユーザのユーザIDです。

■<MGのサイト名>は、JobCenter MGのサイト名です。

■<MGのマシンID>は、JobCenter MGのマシンIDです。

■<ユーザマッピングのモード>は、ユーザマッピングの種別です。詳細は<NQS機能利用の手引き>の「6.5.1.2 ユーザに関するネットワーク環境」を参照してください。

ユーザマッピングのモードと表示される値の対応は以下のとおりです。

モード	表示される値
TYPE1	0
TYPE2	1
TYPE3	2

■<ユーザ名>は、JobCenter SV側で見つからなかったユーザの名前です。

21.3.9.3. 解決方法

存在しないユーザとのユーザマッピングを削除し、存在するユーザでユーザマッピングをやり直してください。

ユーザマッピングの詳細については、「3.2　ユーザの関連付けを行う（ユーザマッピング）」を参照してください。

21.3.10. MGにてSVの名前解決が出来ない

21.3.10.1. 障害内容

単位ジョブをSVのキューに投入する際、SVのサイト名をMG上で名前解決できない場合、単位ジョブが異常終了します。

21.3.10.2. 監視するメッセージ

21.3.10.2.1. SVのキューに直接投入した場合

■UNIX版JobCenter

監視するメッセージ	
svunit_RunUjSubmitEnd(<行数>):submit command error:<部品名> of uid=<ユーザID>:<トラッカID>:No such destination machine name in NMAP.:arg=<キュー名>@<サイト名>	
フィルタ定義のメッセージテキスト	
No such destination machine name in NMAP	
コンポーネント	JobCenter MG
ログファイル	jeerr.log

■Windows版JobCenter

監視するメッセージ	
ERROR:submit command error:<部品名> of uid=<ユーザID>:<トラッカID>:No such destination machine name in NMAP.:arg=<キュー名>@<サイト名>	
フィルタ定義のメッセージテキスト	
No such destination machine name in NMAP	
コンポーネント	JobCenter MG
ログファイル	jnwengineErr.log

ジョブの投入キューに、JobCenter SVのキューを直接指定した場合のメッセージです。

ジョブネットワーク投入時、SVの名前解決ができず、通信に失敗した際に記録されます。

■svunit_RunUjSubmitEnd(<行数>)はJobCenterの内部処理の発生箇所を表す情報です。

詳細情報は非公開ですのでご了承ください。

■<部品名>は、エラーが発生したジョブの名前です。

■<ユーザID>は、ジョブネットワークを投入したユーザのIDです。

■<トラッカID>は、エラーが発生したトラッカのIDです。

■<キュー名>は、投入しようとしたJobCenter SVのキューの名前です。

■<サイト名>は、投入しようとしたJobCenter SVのサイト名です。

21.3.10.2.2. MGのパイプキュー経由でSVのキューに投入した場合

■UNIX版JobCenter

監視するメッセージ

MM/DD hh:mm:ss NQS(ERROR): estblshreq(): failed to getaddrinfo: node=<SVのサイト名> service=<ポート> (eai_errno=<エラーコード>).	
フィルタ定義のメッセージテキスト	
estblshreq\(\): failed to getaddrinfo	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): Nqslib:estblshreq: failed to jc_getaddrinfo: node=<SVのサイト名> service=<ポート> ai_family=<アドレスファミリー> (ec=<エラーコード>).	
フィルタ定義のメッセージテキスト	
Nqslib:estblshreq: failed to jc_getaddrinfo	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

ジョブの投入キューにJobCenter　MGのパイプキューを指定し、パイプキューの転送先がJobCenter　SVのキューである場合のメッセージです。

ジョブネットワーク投入時、SVの名前解決ができず、通信に失敗した際に記録されます。

■<SVのサイト名>は、投入しようとしたJobCenter SVのサイト名です。

■<ポート>は、通信ポート番号、または、サービス名です。

■<アドレスファミリー>は、通信のアドレスの種類を表す値です。

サポート窓口にて障害の原因究明を行う際に利用します。

■<エラーコード>は、エラーの内容を示す値です。

サポート窓口にて障害の原因究明を行う際に利用します。エラーコードの詳細情報は非公開ですのでご了承ください。

21.3.10.3. 解決方法

MG/SVの間で、相互に正引き・逆引きの名前解決できる状態にしてください。

名前解決の設定については、「2.3 Windows環境における名前解決方法」、または、「2.4 UNIX環境における名前解決方法」を参照してください。

21.3.11. SVにてMGの名前解決が出来ない

21.3.11.1. 障害内容

単位ジョブをSVのキューに投入する際、MGのサイト名をSV上で名前解決できない場合、単位ジョブが異常終了します。

21.3.11.2. 監視するメッセージ

21.3.11.2.1. SVでMGの名前解決ができない場合

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Pipeclient: Receive unfortune TCM=01451 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01451 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(WARN): Netserver: client internet addr(<送信元IPアドレス>) unknown to local host.(<詳細エラーメッセージ>)	
フィルタ定義のメッセージテキスト	
client internet addr.* unknown to local host	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Pipeclient:PipeCL: Receive unfortune TCM=01451 by QUEREQ,(<リクエストID(seqno)>.<MGのマシンID>),dest=<SV側のキュー名>@[<SVのマシンID>]	
フィルタ定義のメッセージテキスト	
Receive unfortune TCM=01451 by QUEREQ	
コンポーネント	JobCenter MG
ログファイル	nqserr.log

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(WARN): Netserver: client internet addr(<送信元IPアドレス>) unknown to local host.(<詳細エラーメッセージ>)	
フィルタ定義のメッセージテキスト	
client internet addr.* unknown to local host	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

ジョブネットワーク投入時、MGの名前解決ができず、通信に失敗した際に記録されます。

■「監視するメッセージ」の「NQS(WARN):」の右側にある空白について、UNIX版JobCenterは半角スペース2つです。Windows版JobCenterは半角スペース1つです。

■<リクエストID(seqno)>は、リクエストごとに採番されたIDです。

■<MGのマシンID>は、JobCenter MGのマシンIDです。

■<SV側のキュー名>は、投入しようとしたJobCenter SV側のキューの名前です。

■<SVのマシンID>は、投入しようとしたJobCenter SVのマシンIDです。

■<送信元IPアドレス>は、JobCenter SV側のIDアドレスです。

■<詳細エラーメッセージ>は、エラーの詳細内容です。

サポート窓口にて障害の原因究明を行う際に利用します。エラーの詳細内容についての情報は非公開ですのでご了承ください。

21.3.11.3. 解決方法

MG/SVの間で、相互に正引き・逆引きの名前解決できる状態にしてください。

名前解決の設定については、「2.3 Windows環境における名前解決方法」、または、「2.4 UNIX環境における名前解決方法」を参照してください。

21.3.12. MG→SV転送時にリクエストIDの重複が発生した

21.3.12.1. 障害内容

ジョブネットワークを実行する際、JobCenter内部では単位ジョブ1つにつきリクエストという単位で管理されます。リクエストIDとは、リクエストを識別する番号のことです。

リクエストIDは、ジョブ実行時にJobCenter MGで1から100,000の範囲の値を連番で採番し、100,000を超えると未使用の番号を再利用します。JobCenter SVでもリクエスト情報は保持しており、リクエストIDで管理されます。

リクエストIDは重複しないようにJobCenterで管理されているため、リクエストIDが重複することはありません。しかし、停電などによりJobCenter SVが正常に停止されなかった場合、ごく稀にJobCenter SVにリクエスト情報が残存する場合があります。JobCenter SVに残存したリクエスト情報のリクエストIDと、JobCenter MGで新しく採番したリクエストIDが重複した場合、ジョブの実行に失敗します。

21.3.12.2. 監視するメッセージ

■UNIX版JobCenter

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): tracking_newreq: request-id overlapped(<リクエストID(seqno)>.<MGのマシンID>)	
フィルタ定義のメッセージテキスト	
tracking_newreq: request-id overlapped	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

監視するメッセージ	
MM/DD hh:mm:ss NQS(ERROR): Netserver: QUEREQ: previous req=<リクエストID(seqno)>.<MGのマシンID> is running	
フィルタ定義のメッセージテキスト	
Netserver: QUEREQ: previous req=.* is running	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

■Windows版JobCenter

監視するメッセージ	
MM/DD/YY hh:mm:ss NQS(ERROR): tracking_newreq: request-id overlapped(<リクエストID(seqno)>.<MGのマシンID>)	
フィルタ定義のメッセージテキスト	
tracking_newreq: request-id overlapped	
コンポーネント	JobCenter SV
ログファイル	nqserr.log

監視するメッセージ
MM/DD/YY hh:mm:ss NQS(ERROR): Netserver: QUEREQ: previous req=<リクエストID(seqno)>.<MGのマシンID> is running
フィルタ定義のメッセージテキスト
Netserver: QUEREQ: previous req=.* is running

コンポーネント	JobCenter SV
ログファイル	nqserr.log

ジョブ実行時に、JobCenter SVに残存したリクエスト情報のリクエストIDと、JobCenter MGで新しく採番したリクエストIDが重複した場合に記録されます。

■<リクエストID(seqno)>は、リクエストごとに採番されたIDです。

■<MGのマシンID>は、JobCenter MGのマシンIDです。

21.3.12.3. 解決方法

本障害が発生した場合は、サポート窓口へお問い合わせを行い、JobCenter SVに残存したリクエストIDを削除して復旧してください。

21.3.13. 同時起動可能数を超えた

21.3.13.1. 障害内容

同時実行数制御を設定しているジョブネットワークは、先行しているジョブネットワークが未完了の場合、JobCenterが自動的に予定状態に変更したりスキップの処理を行います。

同時実行数制御については<基本操作ガイド>の「3.3.4.2 同時実行状態」を参照してください。

同時実行数制御を予定状態に変更する設定を行っている場合、先行しているジョブネットワークの終了が想定よりも遅延していると後続のジョブネットワークの実行開始に影響します。本障害内容を監視することで、同時実行数制御が実行されたタイミングを検知できます。

21.3.13.2. 監視するメッセージ

21.3.13.2.1. JNW単独の排他の場合

■UNIX版JobCenter

監視するメッセージ	
svqueue_Resched(<行数>):> The single exclusive tracker is running and waits for completion. (<トラッカID>)	
フィルタ定義のメッセージテキスト	
The single exclusive tracker is running and waits for completion	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	jeerr.log

■Windows版JobCenter

監視するメッセージ	
INFO:The single exclusive tracker is running and waits for completion. (<トラッカID>)	
フィルタ定義のメッセージテキスト	
The single exclusive tracker is running and waits for completion	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	jnwengineErr.log

ジョブネットワークの同時実行状態に「JNW単独の排他」および「同時起動可能数」を設定した場合のメッセージです。

「同時起動可能数」の設定値を超えて同一のジョブネットワークを実行した際に記録されます。

■svqueue_Resched(<行数>)はJobCenterの内部処理の発生箇所を表す情報です。

詳細情報は非公開ですのでご了承ください。

■<トラッカID>は、同時実行数制御により予定状態またはスキップとなったトラッカIDです。

21.3.13.2.2. JNW同士の排他の場合

■UNIX版JobCenter

監視するメッセージ
svqueue_Resched(<行数>):> The exclusive tracker (<実行中トラッカID>) is running and waits for completion. (<排他待ちトラッカID>)

フィルタ定義のメッセージテキスト	
The exclusive tracker .* is running and waits for completion	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	jeerr.log

■Windows版JobCenter

監視するメッセージ	
INFO:The exclusive tracker (<実行中トラッカID>) is running and waits for completion. (<排他待ちトラッカID>)	
フィルタ定義のメッセージテキスト	
The exclusive tracker .* is running and waits for completion	
コンポーネント	JobCenter MG、JobCenter SV
ログファイル	jnwengineErr.log

ジョブネットワークの同時実行状態に「JNW同士の排他」を設定した場合のメッセージです。

「排他JNW」の設定先ジョブネットワークが実行中に、設定元ジョブネットワークを実行した際に記録されます。

■svqueue_Resched(<行数>)はJobCenterの内部処理の発生箇所を表す情報です。

詳細情報は非公開ですのでご了承ください。

■<実行中トラッカID>は、「排他JNW」の設定先ジョブネットワークのトラッカのIDです。

■<排他待ちトラッカID>は、排他待ちが発生したトラッカのIDです。

21.3.13.3. 解決方法

監視するメッセージから待ち合わせているトラッカを特定し、対象のジョブネットワークの同時実行数制御の設定が正しいか確認してください。

同時実行数制御の設定が正しい場合、先行しているジョブネットワークの完了が想定より遅延している可能性があります。ジョブネットワークが遅延する原因は、JobCenterの障害やジョブ実行サーバの負荷など様々な原因があります。その他の障害情報や22章　「SystemManager　GでJobCenterのメトリクスを収集・分析する」 のメトリクスを合わせて監視することで、原因調査や改善方法の検討に利用できます。

21.3.14. ジョブ実行ユーザに指定したユーザが存在しない

21.3.14.1. 障害内容

単位ジョブのパラメータ「ジョブ実行ユーザ」に指定したユーザが、JobCenter MGをインストールしたマシン上に存在しない場合、ジョブネットワークの投入に失敗します。

21.3.14.2. 監視するメッセージ

■UNIX版JobCenter

監視するメッセージ	
svunit_Init(<行数>):> exec_username not found:<ジョブ実行ユーザ名> of uid=<ユーザID>:<トラッカID>/<単位ジョブ名>	
フィルタ定義のメッセージテキスト	
exec_username not found:.* of uid=	
コンポーネント	JobCenter MG
ログファイル	jeerr.log

■Windows版JobCenter

監視するメッセージ	
ERROR:pwdcache_name_to_uid:unknown user(name=<ジョブ実行ユーザ名>), jc_GetPWbyname_internal error, name=<ジョブ実行ユーザ名>	
フィルタ定義のメッセージテキスト	
pwdcache_name_to_uid:unknown user\(name=.*\)	
コンポーネント	JobCenter MG
ログファイル	jnwengineErr.log

単位ジョブのパラメータ「ジョブ実行ユーザ」に指定したユーザが、JobCenter MGをインストールしたマシン上に存在しない状態でジョブネットワークを実行した場合に記録されます。

■svunit_Init(<行数>)はJobCenterの内部処理の発生箇所を表す情報です。

詳細情報は非公開ですのでご了承ください。

■<ジョブ実行ユーザ名>は対象トラッカのジョブ実行ユーザの名前です。

■<ユーザID>は対象トラッカのジョブ実行ユーザのUIDです。

■<トラッカID>は対象トラッカのトラッカIDです。

トラッカIDは「ジョブネットワーク名.YYYYMMDDhhmmss」の形式で記録されます。

■<単位ジョブ名>は「ジョブ実行ユーザ」パラメータが設定された単位ジョブ名です。

21.3.14.3. 解決方法

単位ジョブのパラメータ「ジョブ実行ユーザ」に指定したユーザ名が正しいか確認してください。

21.3.15. 2GBを超えるトラッカのアーカイブに失敗

21.3.15.1. 障害内容

JobCenterはトラッカのデータサイズが2GBを超える場合の動作をサポートしておりません。データサイズが2GBを超えるトラッカが存在する場合、JobCenterが正常に動作しなくなります。

トラッカのデータの主な内容は、単位ジョブの標準出力結果・標準エラー出力結果です。通常の運用時ではデータサイズが2GBを超えることはありませんが、例えば単位ジョブのスクリプト内でデバッグログを出力する設定のまま元に戻すのを失念していた場合に発生するケースがあります。

R16.1以降のJobCenterでは、データサイズが2GBを超えるトラッカをアーカイブする際に、自動的にトラッカデータを退避させ、アーカイブファイルに記録しないようにする機能が実装されました。この機能が実行された時のメッセージを監視することで、2GBを超えるトラッカが作成された場合に検知できます。

トラッカデータは以下の場所に退避されます。

■UNIX版JobCenter

ローカルサイト	/usr/spool/nqs/users/<ユーザ名>/excluded_archive
クラスタサイト	<JobCenterDBパス>/nqs/users/<ユーザ名>/excluded_archive

■Windows版JobCenter

ローカルサイト	%InstallDirectory%\spool\users\<ユーザ名>\excluded_archive
クラスタサイト	<JobCenterDBパス>\spool\users\<ユーザ名>\excluded_archive

■単位ジョブの標準出力結果または標準エラー出力結果の1つが2GBを超える場合は、トラッカデータの退避機能およびメッセージの監視はできません。出力結果1つにつき2GB未満である必要があります。

■退避されたトラッカデータは、サポート窓口にて障害の原因究明を行う際に利用します。トラッカデータのフォーマットは公開しておりませんのでご了承ください。

21.3.15.2. 監視するメッセージ

■UNIX版JobCenter

監視するメッセージ	
svjnwent_ArchiveCmdEnd(<行数>):The threshold for the size of the tracker that can be archived has been exceeded. (<ジョブ実行ユーザ名> of uid=<ユーザID>:<トラッカID>)	
フィルタ定義のメッセージテキスト	
The threshold for the size of the tracker that can be archived has been exceeded	
コンポーネント	JobCenter MG
ログファイル	jeerr.log

■Windows版JobCenter

監視するメッセージ
ERROR:The threshold for the size of the tracker that can be archived has been exceeded. (<ジョブ実行ユーザ名> of uid=<ユーザID>:<トラッカID>)
フィルタ定義のメッセージテキスト
The threshold for the size of the tracker that can be archived has been exceeded

コンポーネント	JobCenter MG
ログファイル	jnwengineErr.log

トラッカサイズが2GBを超えていたため、アーカイブファイルにトラッカ情報を登録できなかった場合に記録されます。

■svjnwent_ArchiveCmdEnd(<行数>)はJobCenterの内部処理の発生箇所を表す情報です。

詳細情報は非公開ですのでご了承ください。

■<ジョブ実行ユーザ名>は対象トラッカのジョブ実行ユーザの名前です。

■<ユーザID>は対象トラッカのジョブ実行ユーザのUIDです。

■<トラッカID>は対象トラッカのトラッカIDです。

トラッカIDは「ジョブネットワーク名.YYYYMMDDhhmmss」の形式で記録されます。

21.3.15.3. 解決方法

対象トラッカのジョブネットワークに配置している単位ジョブ部品のスクリプトを確認し、標準出力結果または標準エラー出力結果に大量のログを出力していないか確認してください。

また、トラッカのデータサイズが2GBを超えていない場合でも、想定外のデータサイズとなる可能性があります。データサイズが巨大なトラッカが複数存在すると、ディスクフルなどの障害が発生する可能性があります。そのため、22章 「SystemManager GでJobCenterのメトリクスを収集・分析する」 の以下のメトリクスもあわせて監視することを推奨します。

■「22.3.6 未アーカイブトラッカ全体サイズ」

■「22.3.7 アーカイブ済みトラッカ全体サイズ」

■「22.3.8 最大トラッカサイズ」

22. SystemManager GでJobCenterのメトリクスを収集・分析する

NECのWebSAM　SystemManager　G(以下、SystemManagerGと記します)のカスタム性能監視機能を用いて、ジョブ実行データの統計や性能値のメトリクスを収集・分析する方法について説明します。

22.1. 機能概要

特定の時間帯にジョブの実行が集中したり、未アーカイブのトラッカデータが多く存在すると、サーバーのリソースが消費され十分な性能が出せません。また、サーバー上で動作している他のサービスにも影響を及ぼします。適切なスケジュール設定、アーカイブ期間、パイプキューを利用した負荷分散などを実施することで改善可能です。

本機能では、SystemManagerGのカスタム性能監視機能を用いて、ジョブ実行データやキューの利用状況のメトリクスを集計・分析できます。ジョブの実行が集中する時間帯や使用頻度の高いキューを把握することで、適切な運用改善ができます。

SystemManagerGのカスタム性能監視機能は、ジョブ実行データやキューの利用状況などのメトリクスを収集するJobCenterのコマンドを定期的に実行し、実行結果をSystemManagerGに蓄積します。蓄積したメトリクスを用いて、グラフの表示や分析ができます。

JobCenterのメトリクスを収集するコマンドについては<コマンドリファレンス>の9章　「メトリクス収集コマンド」 を参照してください。

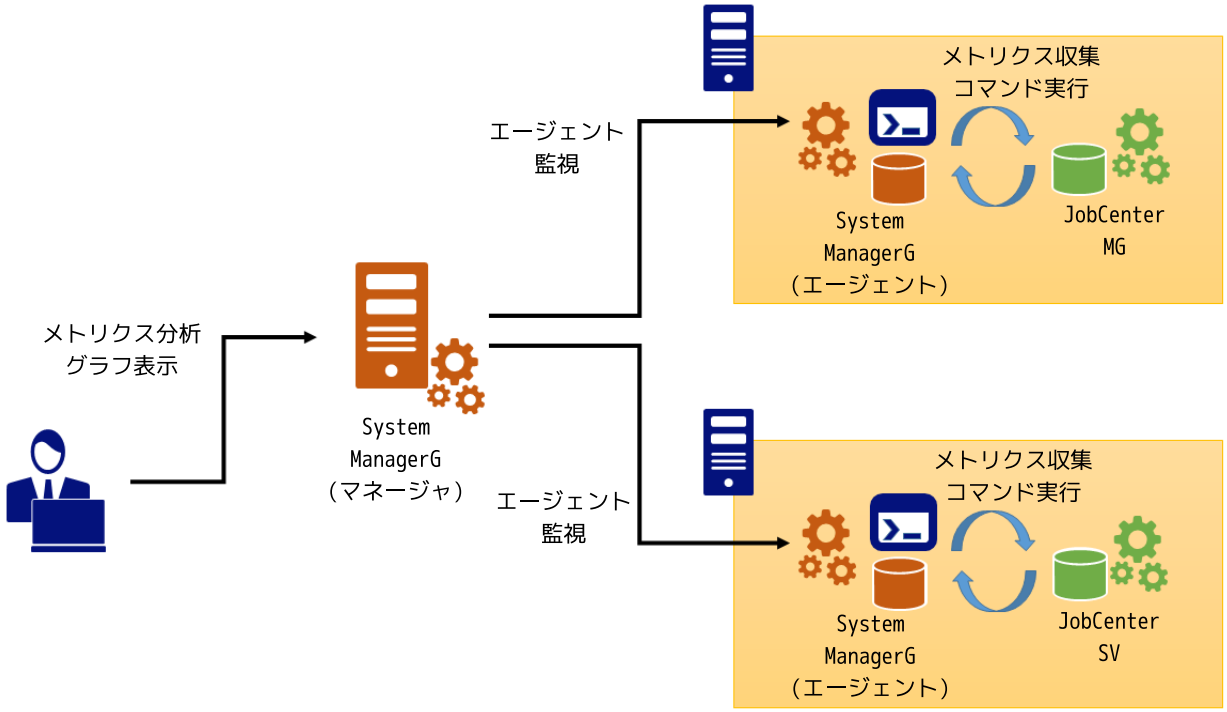

図22.1 JobCenterのメトリクス収集イメージ

22.2. 構築手順

SystemManagerGのカスタム性能監視機能を用いて、ジョブ実行データの統計や性能値のメトリクスを収集・分析するための環境構築手順について説明します。

本手順を実施する前に以下の製品をインストールしてください。

■SystemManagerG(マネージャ)

■SystemManagerG(エージェント)

■JobCenter

SystemManagerG(エージェント)とJobCenterは同一マシンにインストールしてください。JobCenterのインストール方法については<インストールガイド>を参照してください。SystemManagerGのインストール方法については、SystemManagerGのインストールガイドを参照してください。

■メトリクスを収集するJobCenterごとにSystemManagerG(エージェント)が必要です。例えばJobCenter MGとJobCenter SVの両方ともメトリクスを収集したい場合は、それぞれにSystemManagerG(エージェント)をインストールする必要があります。

■R16.1以降のJobCenter MGおよびSVでメトリクスを収集可能です。

■カスタム性能監視機能の設定については、本マニュアルで記載していないパラメータについても任意に設定可能です。お客様の環境に合わせて設定してください。パラメータの詳細はSystemManagerGの設定ガイドを参照してください。

22.2.1. 性能値取得設定ファイル(getcustomperflist.properties)の設定

カスタム性能監視の設定を行う前に、SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)の設定が必要です。

SystemManagerG(エージェント)をインストールしたサーバーで実施してください。

1. 以下のパスにgetcustomperflist.propertiesが存在するか確認します。

■UNIX版SystemManagerG

<agent-install-path>/conf/performancemonitor/custom/getcustomperflist.properties

■Windows版SystemManagerG

<agent-install-path>\conf\performancemonitor\custom\getcustomperflist.properties

上記パスにgetcustomperflist.propertiesが存在する場合は、既存のファイルに設定を追記します。

存在しない場合は、以下にサンプルがありますので、コピーして上記パスに格納してください。

■UNIX版SystemManagerG

<agent-install-path>/conf/sample/performancemonitor/custom/getcustomperflist.properties

■Windows版SystemManagerG

<agent-install-path>\conf\sample\performancemonitor\custom\getcustomperflist.properties

> <agent-install-path>はSystemManagerG(エージェント)のインストールディレクトリです。初期値でインストールした場合は以下となります。
>
> ■UNIX版SystemManagerG
>
> /opt/nec/pf/opm/agent
>
> ■Windows版SystemManagerG
>
> C:\Program Files\NEC\pf\opm\agent

2. JobCenterのメトリクスの設定を記述します。

テキストエディタでgetcustomperflist.propertiesを開き、「22.3 メトリクス一覧」の収集したいメトリクスの章にある「性能値取得設定ファイルのパラメータ」の表を参考に記述してください。

■JobCenterのメトリクス「22.3.2 単位ジョブ実行数」の設定例

```
object.count=1

object1.name=JobCenter Performance monitoring
object1.instance.count=1
object1.instance1.name=executed unit job
object1.instance1.command=/usr/lib/nqs/gui/bin/jc_collect_runjob -r $INTERVAL$
object1.instance1.command_delimiter=","
object1.instance1.counter.count=1
object1.instance1.counter1.name=executed unit job
object1.instance1.counter1.position_row=1
object1.instance1.counter1.position_column=2
object1.instance1.counter1.text=Number of executed unit jobs in the last minute.
```

JobCenterのメトリクスの設定を複数記述する場合は、1つのobject配下にメトリクスの設定項目をinstanceパラメータとして記述する階層構造となります。イメージ図で表すと以下のようになります。

```
object1 (JobCenterのメトリクス全体の設定)
    instance1 (JobCenterのメトリクスAの設定)
        counter1
        counter2
           :
    instance2 (JobCenterのメトリクスBの設定)
        counter1
        counter2
           :
    instance3 (JobCenterのメトリクスCの設定)
       :
```

例えば、「22.3.2 単位ジョブ実行数」と「22.3.3 ジョブネットワーク実行数」の2つのメトリクスを設定する場合は以下のようになります。

■「22.3.2 単位ジョブ実行数」と「22.3.3 ジョブネットワーク実行数」の設定例

```
object.count=1

object1.name=JobCenter Performance monitoring
object1.instance.count=2
object1.instance1.name=executed unit job
object1.instance1.command=/usr/lib/nqs/gui/bin/jc_collect_runjob -r $INTERVAL$
object1.instance1.command_delimiter=","
object1.instance1.counter.count=1
object1.instance1.counter1.name=executed unit job
object1.instance1.counter1.position_row=1
object1.instance1.counter1.position_column=2
object1.instance1.counter1.text=Number of executed unit jobs in the last minute.
object1.instance2.name=running jobnetwork
object1.instance2.command=/usr/lib/nqs/gui/bin/jc_collect_runjnw
object1.instance2.command_delimiter=","
object1.instance2.counter.count=2
object1.instance2.counter1.name=Number of unarchived trackers
object1.instance2.counter1.position_row=1
object1.instance2.counter1.position_column=2
object1.instance2.counter1.text=The time average of the number of unarchived trackers in the
 last minute.
object1.instance2.counter2.name=Number of trackers with a running status
object1.instance2.counter2.position_row=1
object1.instance2.counter2.position_column=3
object1.instance2.counter2.text=The time average of the number of trackers with a running
 status for the last minute.
```

■object1.instance.countパラメータはobject1のinstanceの定義数です。設定するJobCenterのメトリクスの数を指定してください。

上記の例では、2つのメトリクスを設定しているので「2」を指定しています。

■カスタム性能監視の監視項目選択画面で性能値を選択する際に、object単位で表示されます。同一のobjectとして定義することで、JobCenterのメトリクスをグルーピングできます。

既にJobCenterとは別の設定が記載されている場合は、object.countパラメータの値をインクリメントし、object[n]パラメータの添え字(n)を重複しない数字に置き換えてファイルの末尾に追記してください。

■既にJobCenterとは別の設定(Object1)が記載されている場合の設定例

```
object.count=2

object1.name=Object1
object1.instance.count=1
  :
object1.instance1.counter2.text=Description2

object2.name=JobCenter Performance monitoring
object2.instance.count=1
object2.instance1.name=executed unit job
object2.instance1.command=/usr/lib/nqs/gui/bin/jc_collect_runjob -r $INTERVAL$
object2.instance1.command_delimiter=","
object2.instance1.counter.count=1
object2.instance1.counter1.name=executed unit job
object2.instance1.counter1.position_row=1
object2.instance1.counter1.position_column=2
object2.instance1.counter1.text=Number of executed unit jobs in the last minute.
```

object.countパラメータはobjectの定義数です。getcustomperflist.propertiesに記述しているobjectの数を設定してください。

上記の例では、Object1とJobCenterのメトリクスの合計2つのobjectを設定しているので「2」を指定しています。

3. SystemManagerG(エージェント)を再起動します。

■UNIX版SystemManagerG

以下のコマンドでmsc_agentサービスを再起動します。

```
systemctl restart msc_agent
```

■Windows版SystemManagerG

Windowsの[コントロールパネル]→[サービス]を起動し、以下のサービスを再起動します。

SystemManager G Agent Service

22.2.2. カスタム性能監視の設定

「22.2.1　性能値取得設定ファイル(getcustomperflist.properties)の設定」の完了後、カスタム性能監視の設定を行います。

1. ブラウザでSystemManagerG(マネージャ)のWeb GUIに接続し、画面上部の[監視]→[設定]→[エージェント監視]をクリックします。

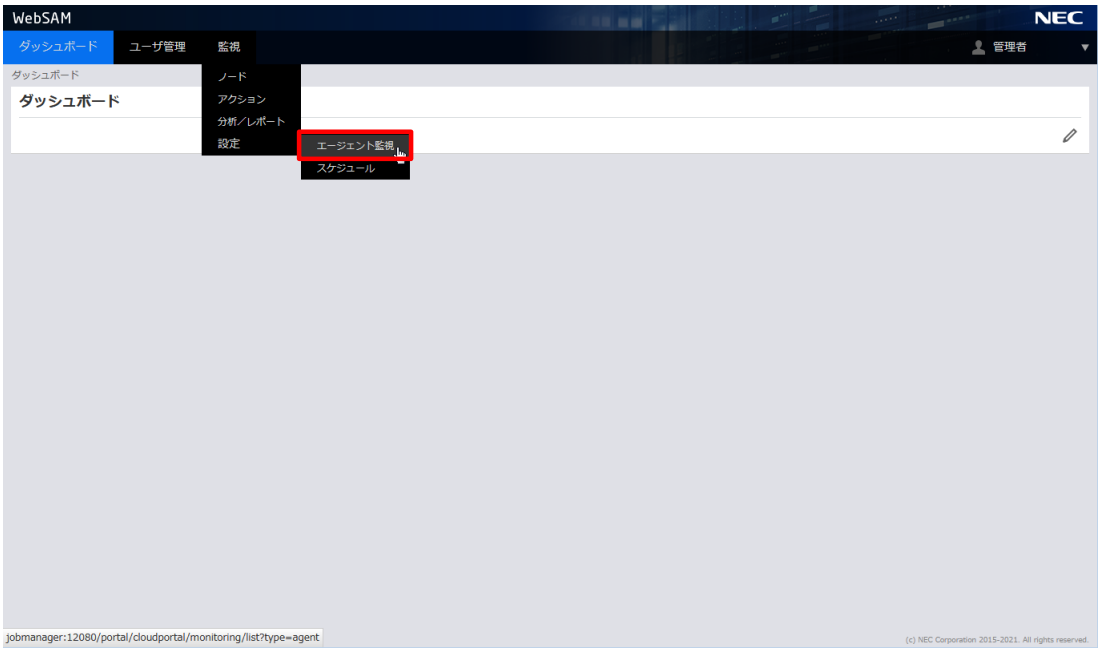

図22.2 エージェント監視の設定画面の表示方法

2. カスタム性能監視を設定するノード(エージェント)を選択し、[カスタム性能監視]をクリックします。

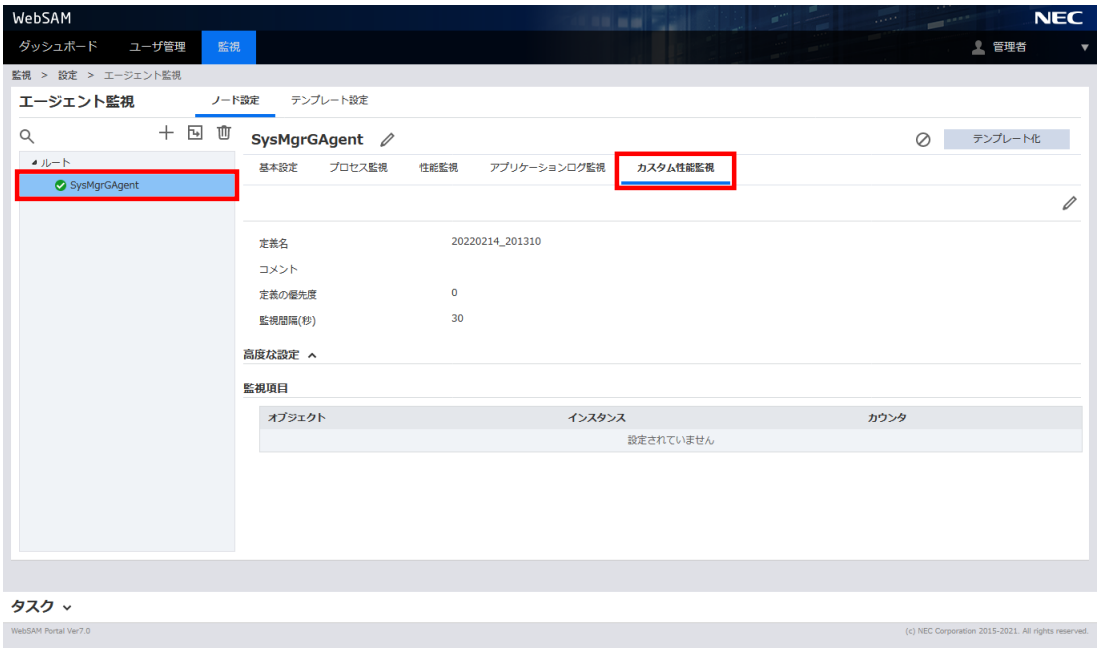

図22.3 カスタム性能監視の設定画面の表示方法

[カスタム性能監視]が表示されない場合は、以下を確認してください。

■SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)の設定内容が正しいこと。

■SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)の設定後、SystemManagerG(エージェント)を再起動していること。

3. 画面右側にある鉛筆アイコンをクリックし、カスタム性能監視の編集画面を表示します。

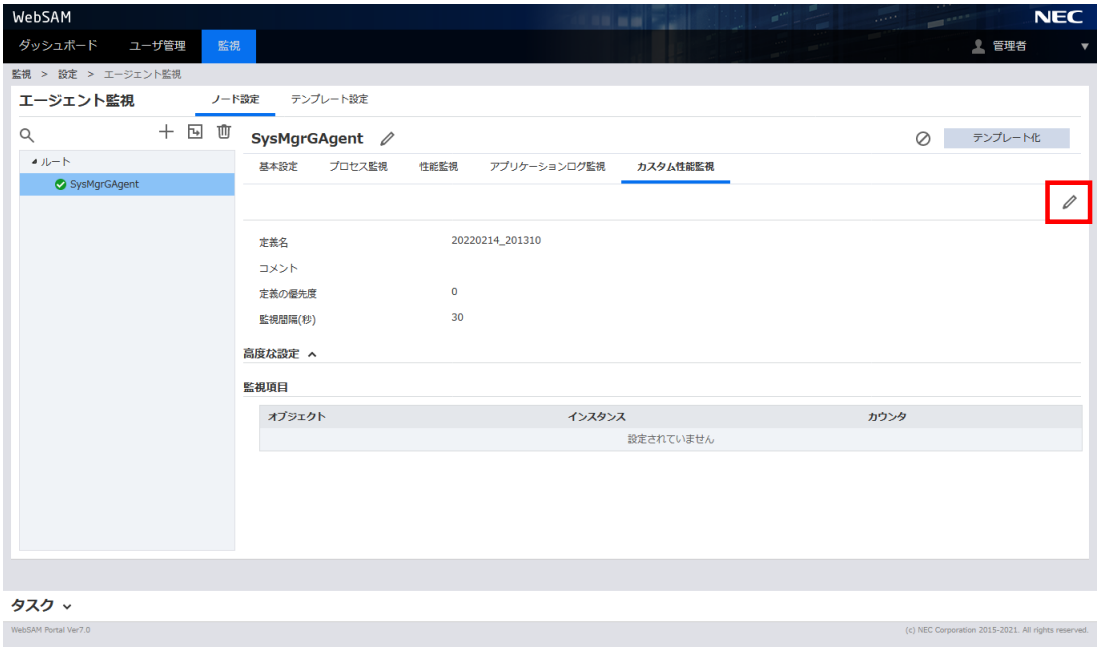

図22.4 カスタム性能監視の編集画面の表示方法

4. カスタム性能監視の各パラメータを設定します。

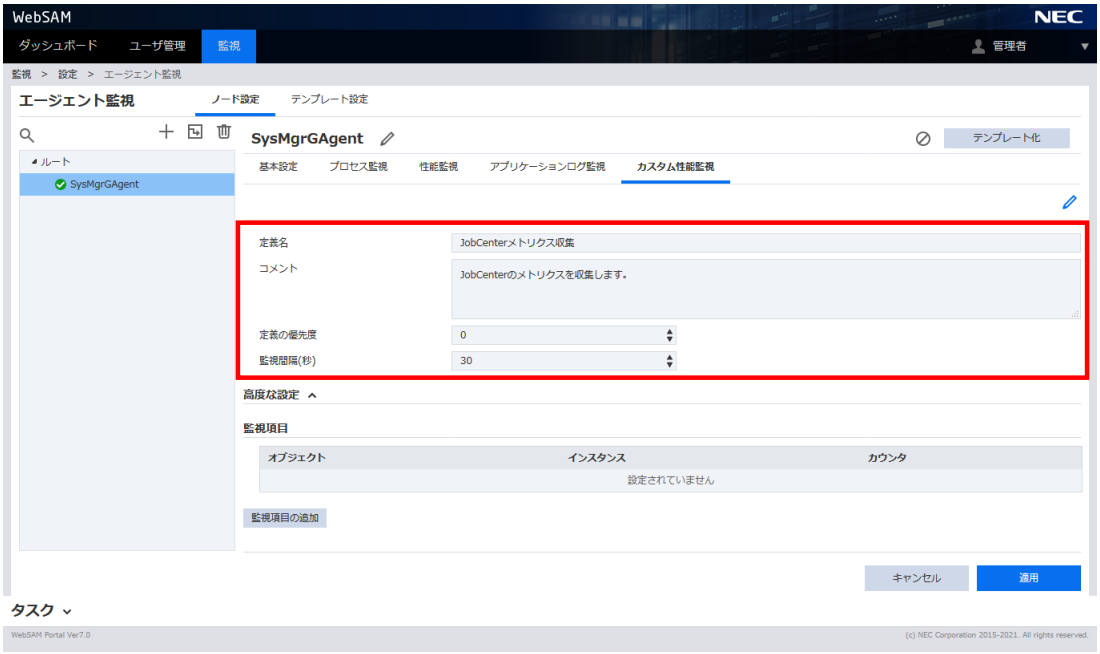

図22.5 カスタム性能監視の編集画面

表22.1 カスタム性能監視のパラメータ

項目名	説明	設定例
定義名	カスタム性能監視の定義名を入力します。 設定値は任意です。	JobCenterメトリクス収集
コメント	コメントを入力します。 設定値は任意です。	JobCenterのメトリクスを収集します。
定義の優先度	定義の優先度を指定します。 既定値の0で問題ありません。	0
監視間隔(秒)	監視の時間間隔を秒単位で指定します。 既定値の30で問題ありません。	30

5. 監視項目の左側にある[監視項目の追加]ボタンをクリックし、監視項目選択画面を表示します。

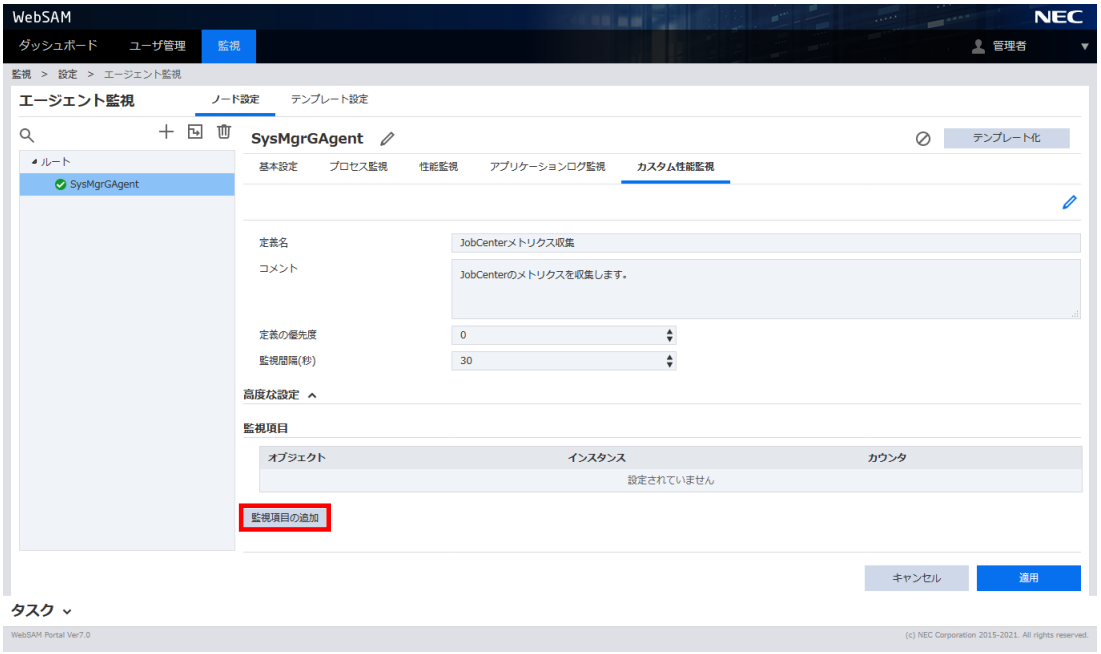

図22.6 監視項目選択画面の表示方法

6. 監視項目を選択します。

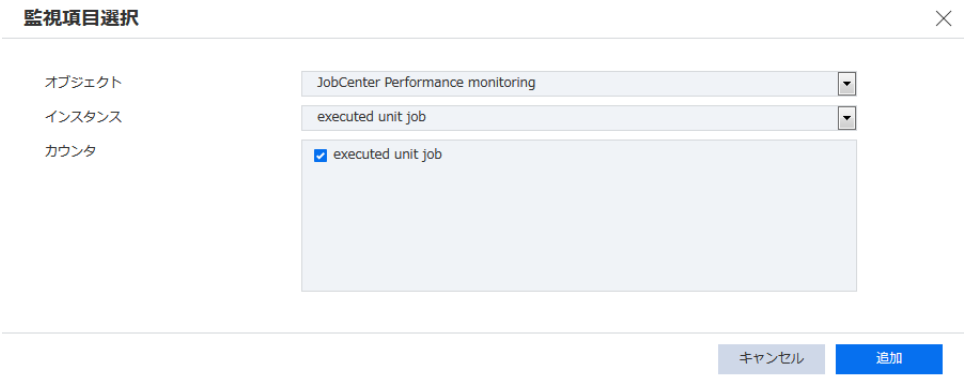

図22.7 監視項目選択画面

表22.2 監視項目選択のパラメータ

項目名	説明	設定例
オブジェクト	監視対象のオブジェクトを指定します。 SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定されたオブジェクト(object[n].name)が一覧で表示されます。「22.2.1 性能値取得設定ファイル(getcustomperflist.properties)の設定」で設定したJobCenterのオブジェクトを選択してください。	JobCenter Performance monitoring
インスタンス	監視対象のインスタンスを指定します。 「オブジェクト」パラメータに設定されたインスタンス(object[n].instance[n].name)が一覧で表示さ	executed unit job

項目名	説明	設定例
	れます。収集したいJobCenterのメトリクスに対応したインスタンスを選択してください。	
カウンタ	監視対象のカウンタを指定します。 「インスタンス」パラメータに設定されたカウンタ(object[n].instance[n].counter[n].name)が一覧で表示されます。収集したいJobCenterのメトリクスに対応したカウンタを選択してください。	executed unit job

7. 監視項目の一覧に手順6で選択した監視項目が追加されます。

   追加した監視項目の右側にある鉛筆アイコンをクリックし、カウンタダイアログを表示します。

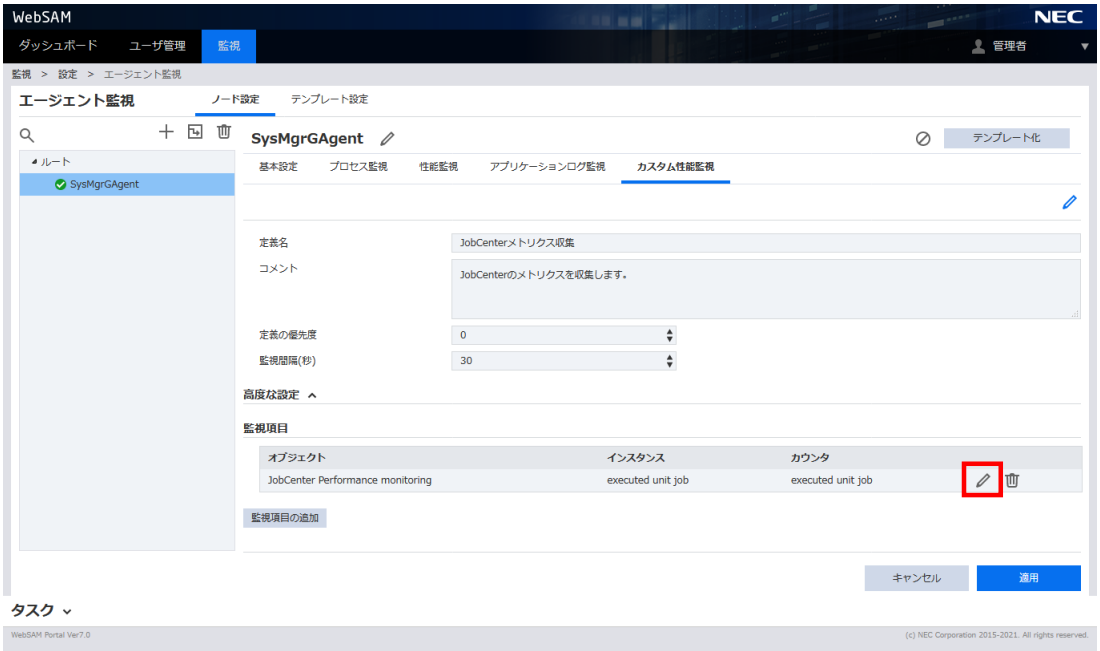

図22.8 カウンタダイアログの表示方法

8. 監視間隔を設定します。

図22.9 カウンタダイアログ

表22.3 カウンタダイアログのパラメータ

項目名	説明	設定例
監視間隔(秒)	カウンタ単位の監視間隔を秒単位で指定します。 JobCenterのメトリクスを収集するコマンドの実行間隔となります。60(1分間隔)を指定することを推奨します。	60

■監視間隔が非常に小さい場合、JobCenterに大きな負荷を与えるため、ジョブ実行などの動作に影響する可能性があります。

■以下のメトリクスは、メトリクス採取時刻の直近1分間のデータを対象とします。「監視間隔(秒)」とメトリクス収集コマンドの取得範囲を一致させるために、60(1分間隔)を指定することを推奨します。

- 「22.3.3 ジョブネットワーク実行数」
- 「22.3.4 投入待ちジョブ部品数」
- 「22.3.9 キュー使用率」
- 「22.3.10 リクエスト増減数」

■以下のメトリクスは、取得範囲の期間を指定可能です。

- 「22.3.2 単位ジョブ実行数」
- 「22.3.5 ジョブネットワーク開始時間差」

取得範囲の期間の指定は、メトリクス収集コマンドのオプション(-r $range)を利用します。コマンドの詳細は<コマンドリファレンス>の9章 「メトリクス収集コマンド」 を参照してください。

以下の点に注意して「監視間隔(秒)」の値を設定してください。

- 取得範囲の期間($range)が非常に大きい場合、取得するデータ量が巨大となるため、メトリクス収集コマンドの実行時間が「監視間隔(秒)」までに完了しない可能性があります。60(1分間隔)を指定することを推奨します。
- 取得範囲の期間($range)と「監視間隔(秒)」は同じ時間となるように設定することを推奨します。同じ時間に設定することで、「監視間隔(秒)」とメトリクス収集コマンドの取得範囲を一致させることができます。

  SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)では、メトリクス収集コマンドの-rオプションに置換文字列「$INTARVAL$」を使用することで、取得範囲の期間($range)と「監視間隔(秒)」を連動させることが可能です。詳細な設定方法は、各メトリクスの章を参照してください。

9. 閾値設定の右側にある＋アイコンをクリックし、閾値設定ダイアログを表示します。

図22.10 閾値設定ダイアログの表示方法

10. 閾値を設定します。

図22.11 閾値設定ダイアログ

表22.4 閾値設定ダイアログのパラメータ

項目名	説明	設定例
閾値判定種別	閾値判定の種類を指定します。 以下の項目を選択可能です。詳細はSystemManagerGの設定ガイドを参照してください。 ■逐次 性能値が設定した閾値を初めて一致もしくは超過したときのみ、状態変化を検出します。 ■連続 性能値が設定した回数連続して閾値を一致もしくは超過したときのみ、状態変化を検出します。 ■平均 性能値の平均値が設定した閾値を一致もしくは超過したときのみ、状態変化を検出します。	逐次
状態名	状態の名前を指定します。 SystemManagerGのノード(エージェント)のステータスの「状態」列に表示されます。	上限警告
監視種別	閾値監視で、上限・下限どちらの監視をするかを指定します。 「上限監視」を指定してください。	上限監視
重要度	重要度を指定します。 SystemManagerGのノード(エージェント)のステータスの「重要度」列に表示されます。	警告
閾値	閾値監視で、通知する際の閾値を指定します。 「22.3 メトリクス一覧」の各メトリクスの章にある「閾値の推奨値」を参考に、お客様の環境に合わせて設定してください。	300

11. 各設定画面で「OK」ボタンをクリックし、設定を保存してください。

    カスタム性能監視を設定したノード(エージェント)のステータスに、JobCenterのメトリクスの状態が表示されます。

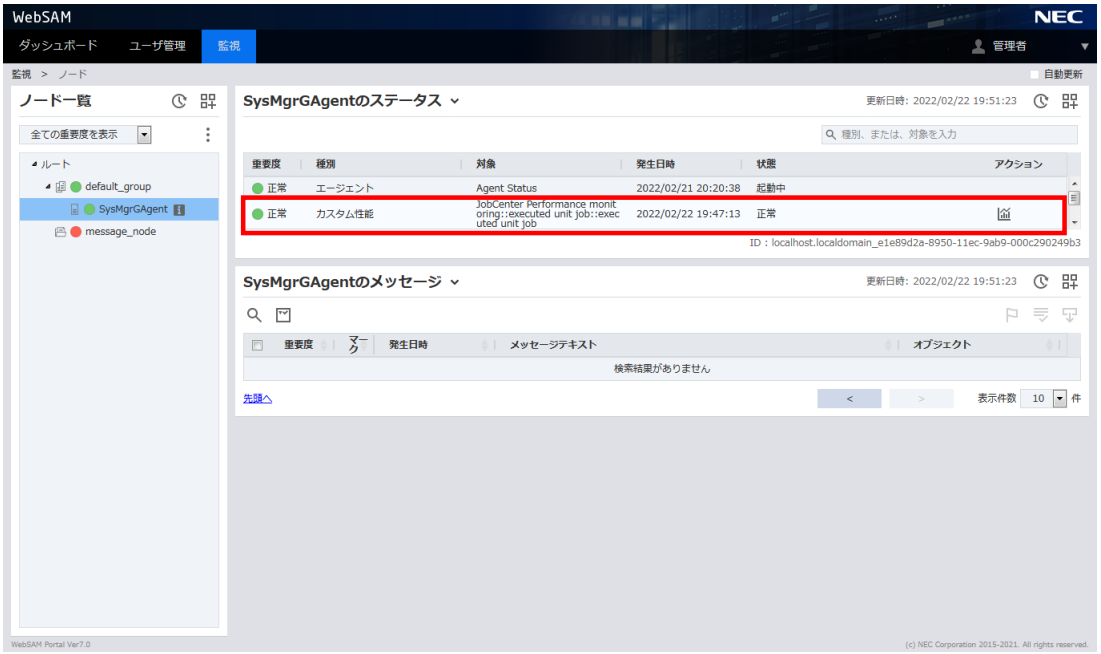

図22.12 SystemManagerGのカスタム性能監視(正常時)

JobCenterのメトリクスの閾値を超過した場合、手順10で設定した閾値設定に従ってステータスの状態が更新されます。また、閾値超過のメッセージも通知されます。

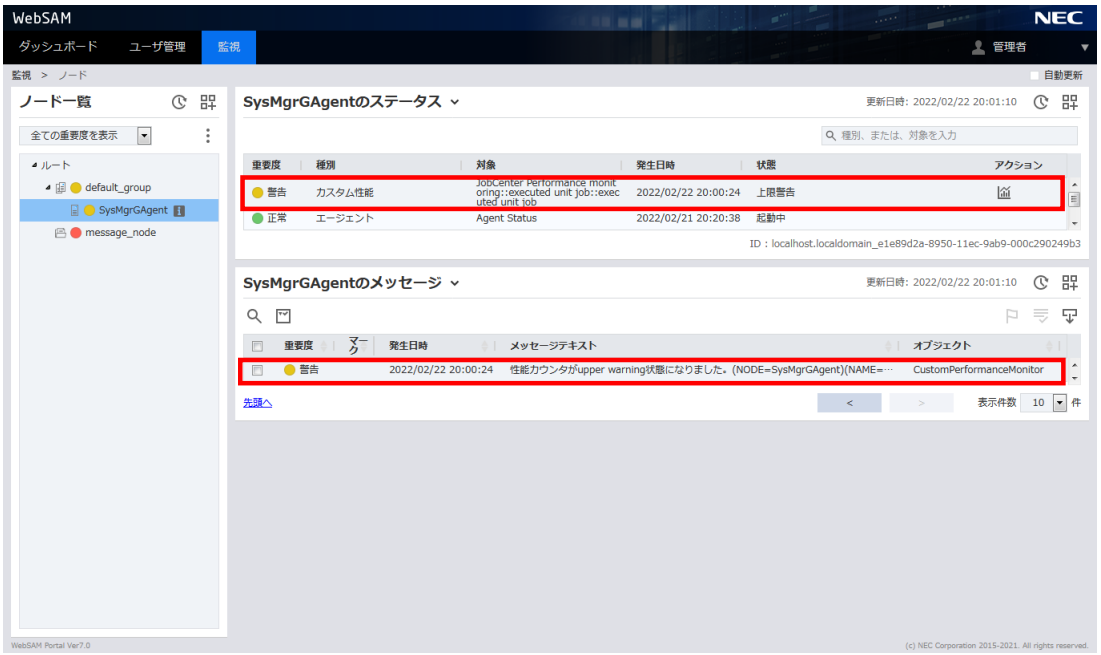

図22.13 SystemManagerGのカスタム性能監視(閾値超過時)

22.2.3. 性能グラフの表示

収集したJobCenterのメトリクスを用いて、性能グラフを表示できます。

性能グラフは、直近のメトリクスを確認したい場合に利用します。性能グラフの詳細はSystemManagerGの操作ガイドを参照してください。

1. カスタム性能監視を設定したノード(エージェント)のステータスの右側にある性能グラフアイコンをクリックし、性能グラフを表示します。

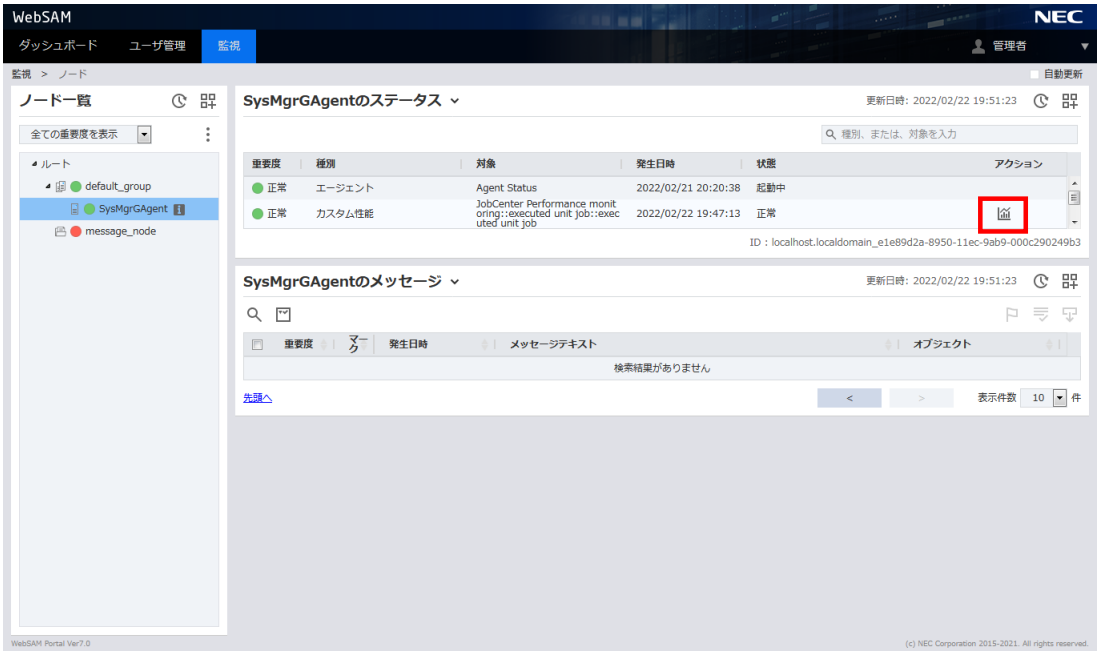

図22.14 性能グラフの表示方法

2. JobCenterのメトリクスの性能グラフが表示されます。

   また、性能グラフの上側にあるボタンで表示期間を変更できます。画像の例では「期間指定」で表示しています。

性能 - グラフ表示
SysMgrGAgentの性能情報
過去１時間
過去１日間
過去１週間
期間指定
期間
2022/02/24
17:45:00
～
2022/02/24
18:45:00
グラフ表示
JobCenter Performance monitoring::executed unit job::executed unit job
400
350
300
250
200
150
100
50
0
2022/02/24 18:00
2022/02/24 18:15
2022/02/24 18:30
400
適用
0
閉じる

図22.15 性能グラフ

22.2.4. 分析／レポート機能

収集したJobCenterのメトリクスを用いて、グラフの作成や分析ができます。

分析／レポート機能の詳細はSystemManagerGの操作ガイドを参照してください。

分析機能を利用するには、最初にグラフの設定が必要です。グラフの設定手順は以下のとおりです。

1. ブラウザでSystemManagerG(マネージャ)のWeb　GUIに接続し、画面上部の[監視]→[分析／レポート]をクリックします。

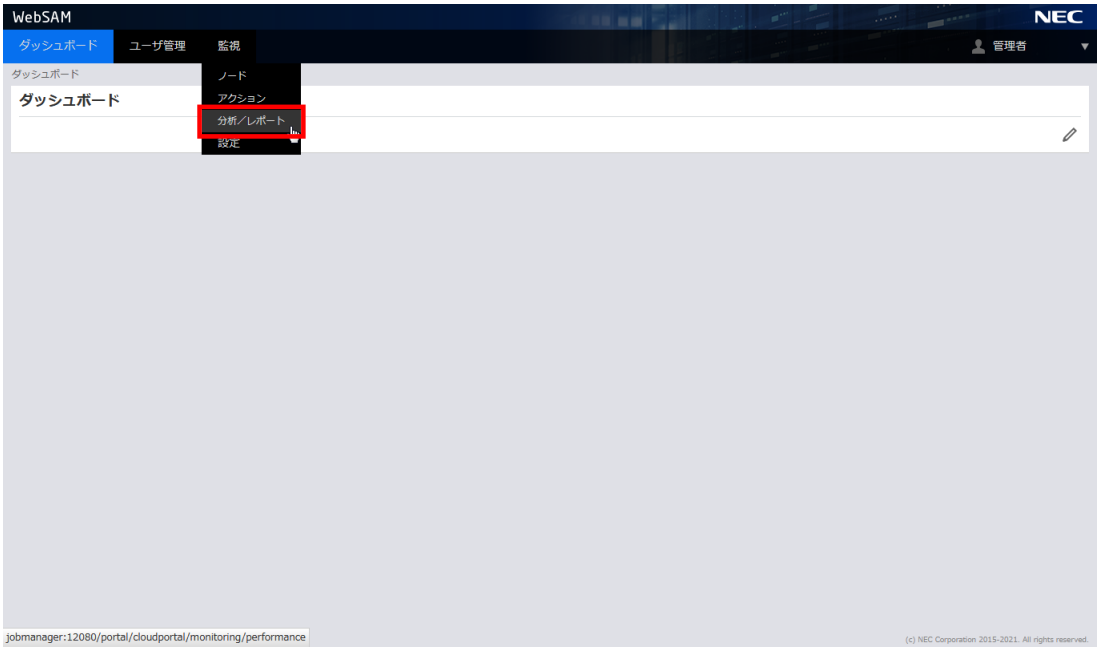

図22.16 分析／レポートの設定画面の表示方法

2. 画面下側にある＋アイコンをクリックし、ビュー追加ダイアログを表示します。

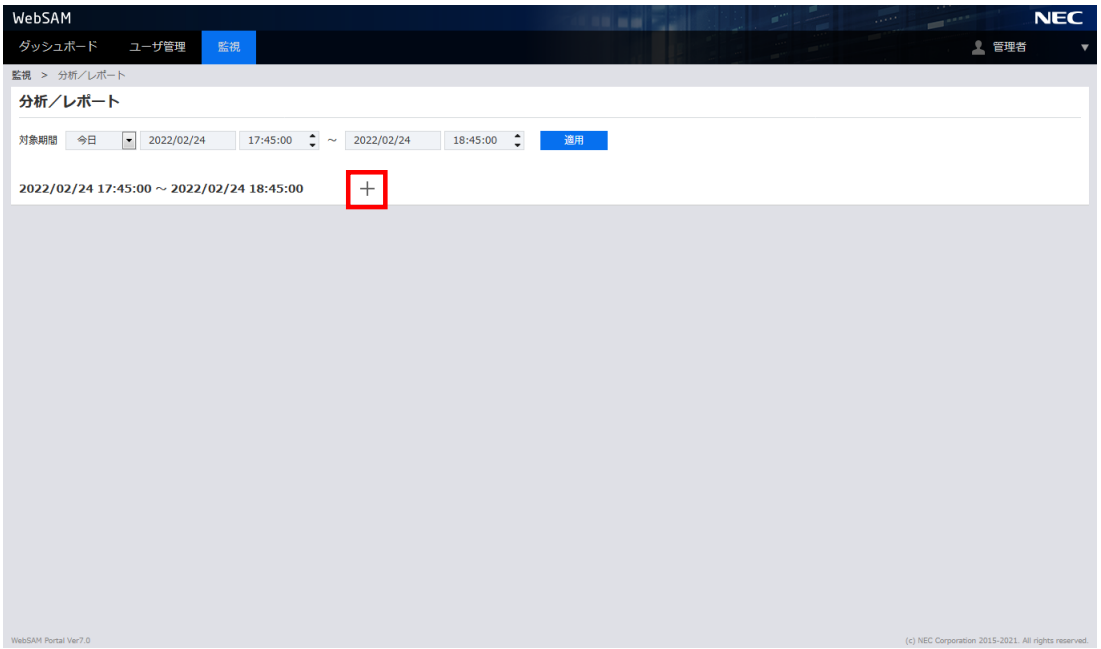

図22.17 ビュー追加ダイアログの表示方法

3. ビュー名を入力します。

   設定値は任意です。

図22.18 ビュー追加ダイアログ

4. 作成したビューを選択し、画面右側にある鉛筆アイコンをクリックしてビュー変更画面を表示します。

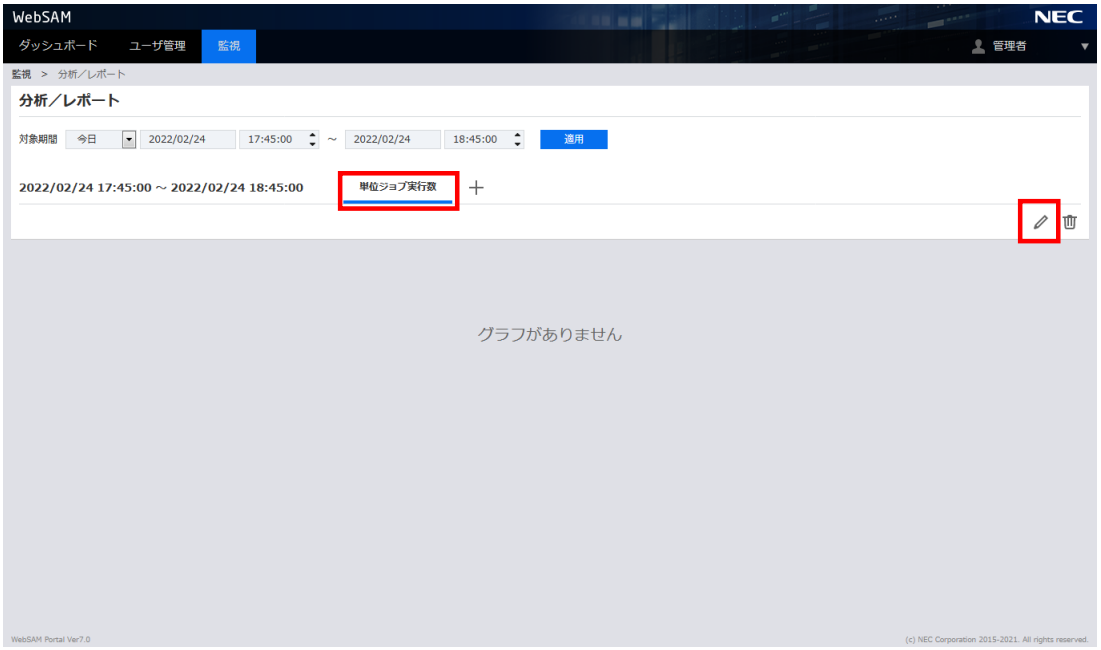

図22.19 ビュー変更画面の表示方法

5. 画面右側にある＋アイコンをクリックし、グラフ設定ダイアログを表示します。

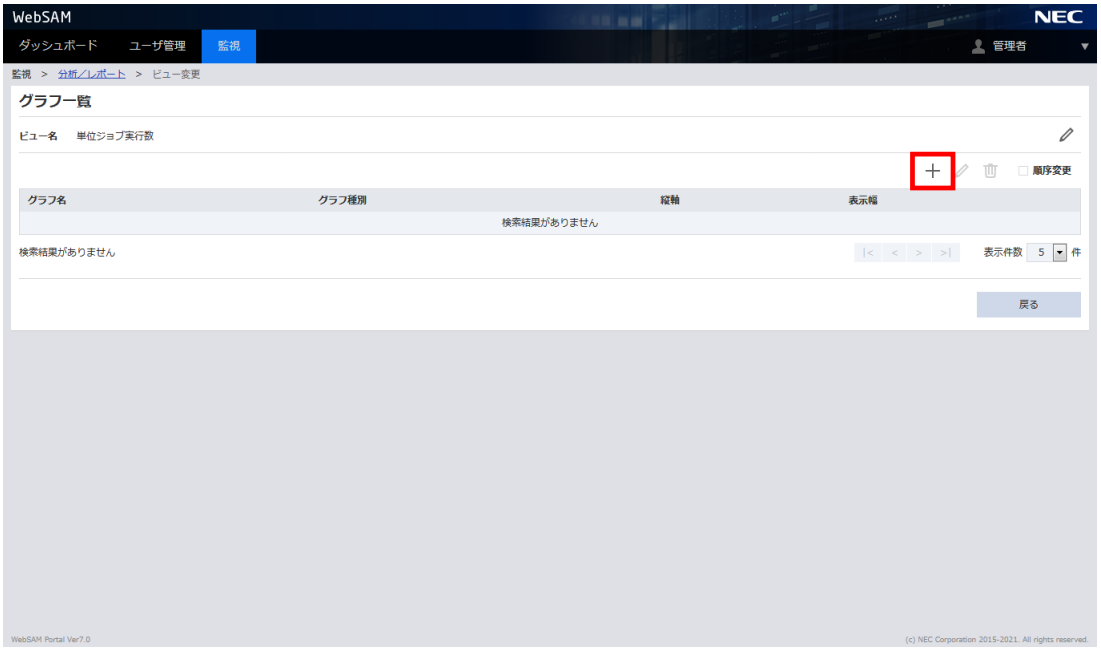

図22.20 グラフ設定ダイアログの表示方法

6. グラフを設定します。

図22.21 グラフ設定ダイアログ

作成可能なグラフの種類は以下のとおりです。

JobCenterのメトリクスに適したグラフは「性能グラフ」または「ランキング」です。本章では「性能グラフ」の設定方法について説明します。他のグラフの設定方法はSystemManagerGの操作ガイドを参照してください。

■性能グラフ

収集したデータを折れ線グラフで表示します。

■ランキング

指定期間内のトップ項目を棒グラフで表示します。

■稼働状況

稼働状況を円グラフで表示します。

表22.5 グラフ設定ダイアログ(性能グラフ)のパラメータ

項目名	説明	設定例
グラフ名	グラフの名前を入力します。 設定値は任意です。	単位ジョブ実行数
グラフ種別	グラフの種類を選択します。 本章では「性能グラフ」を指定した場合を例に説明しています。	性能グラフ
データ種別	データ種別を選択します。 本章では「監視データ」を指定した場合を例に説明しています。 ■監視データ サーバー監視間隔毎に収集した性能データをグラフ化します。障害調査目的など短期間のデータ分析に用います。 ■統計データ 1時間もしくは1日単位で生成される統計データ(平均、最小、最大、最終)をグラフ化します。キャパシティプランニングなど長期間のデータ分析に用います。	監視データ
縦軸	グラフの目盛の描画方法を指定します。 ■自動 プロットされる値の最小値を下辺、最大値を上辺としてグラフ描画します。 ■パーセント 下辺を0、上辺を100%として描画します。	自動
表示幅	グラフの描画サイズを選択します。	100%

7. 作成したグラフを選択し、グラフの詳細の右側にある＋アイコンをクリックしてカウンタ選択ダイアログを表示します。

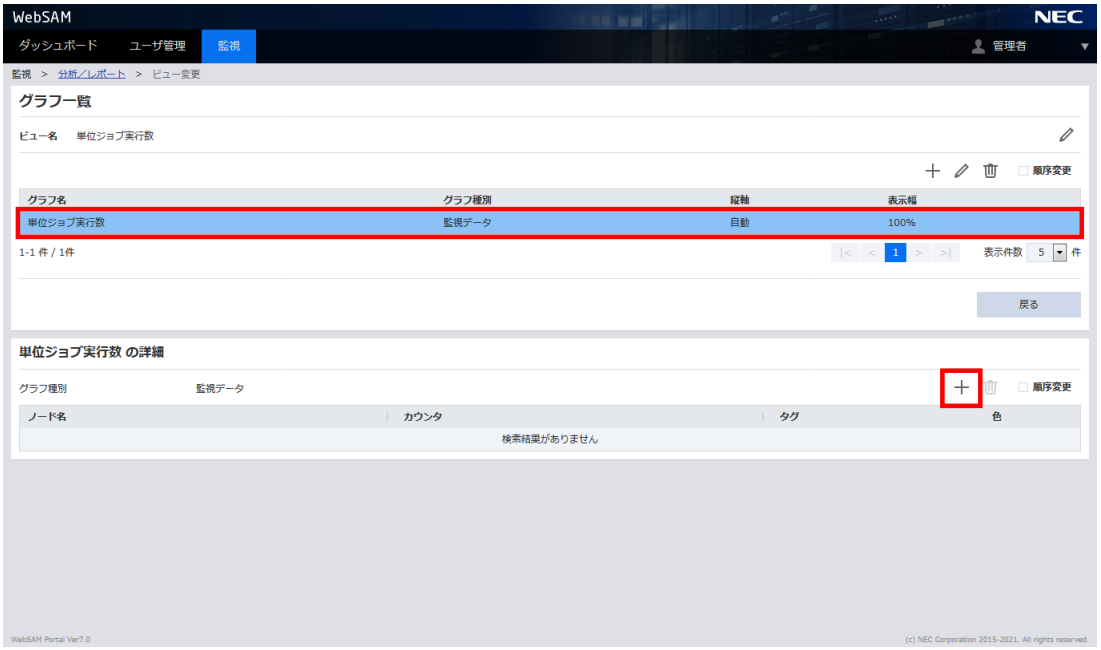

図22.22 カウンタ選択ダイアログの表示方法

8. SystemManagerGで収集した性能データが一覧で表示されます。

   グラフを作成したいJobCenterのメトリクスのチェックボックスにチェックを入れてください。

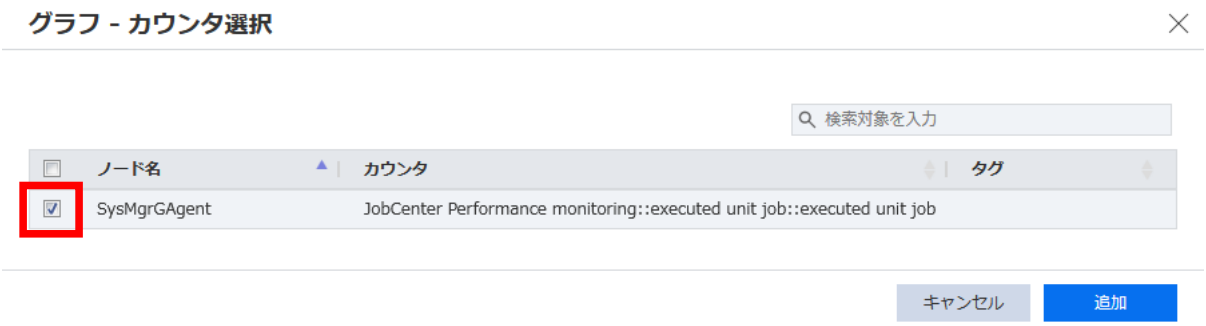

図22.23 カウンタ選択ダイアログ

9. 以上で設定は完了です。

   分析／レポート画面に戻り、作成したビューを選択すると、JobCenterのメトリクスの性能グラフが表示されます。

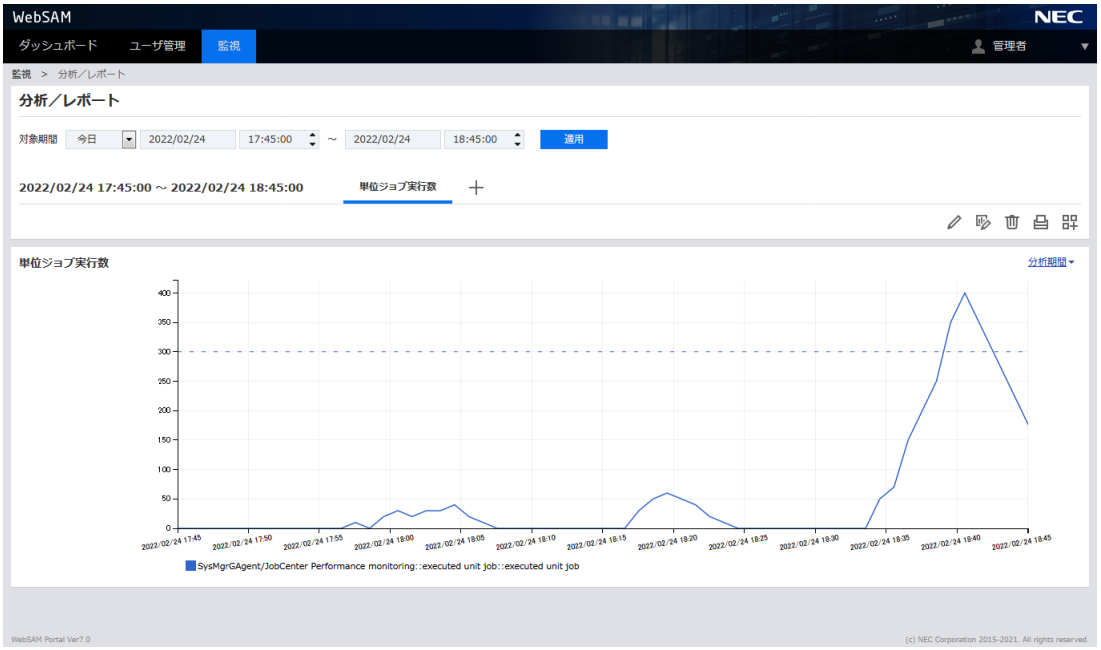

図22.24 分析／レポート画面(性能グラフ)

続いて、分析機能の利用方法について説明します。

作成したグラフを用いて、将来の予測シミュレーションができます。

1. 作成したグラフを選択し、グラフの右側にある[分析期間]をクリックして、表示したい分析期間を選択します。

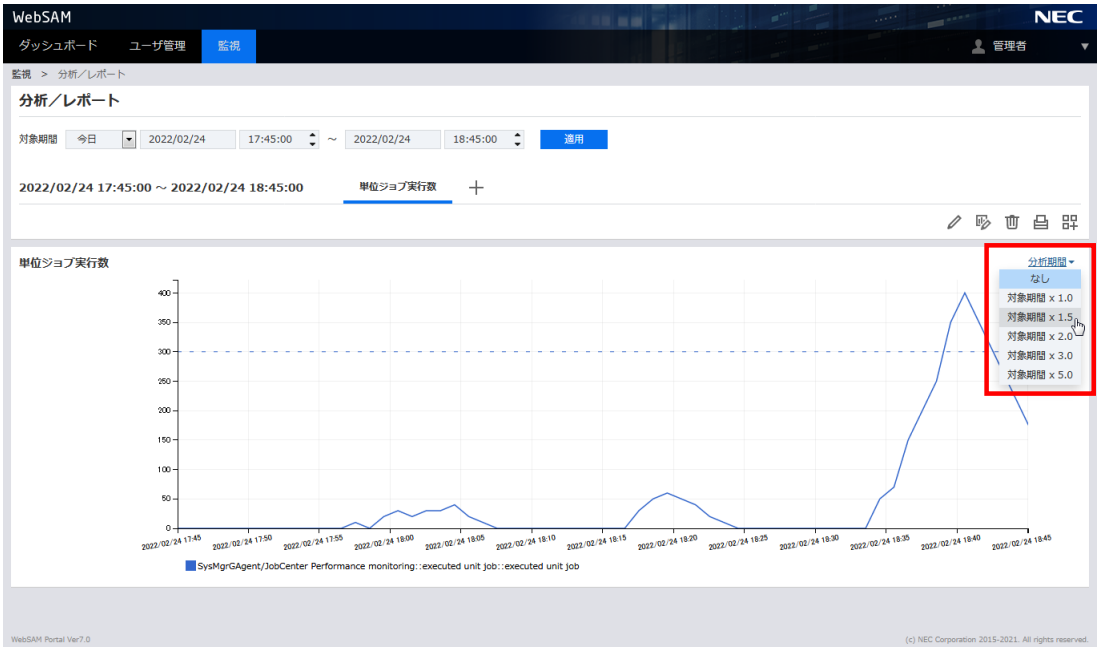

図22.25 分析期間の指定

分析期間は、現在表示している「対象期間」の倍率で指定します。

例えば「対象期間」が18:00～19:00、「分析期間」が「対象期間×2.0」を指定した場合、18:00～20:00が分析する期間となります。

2. 現在表示している「対象期間」のグラフ表示データを基に、近似値直線が表示されます。

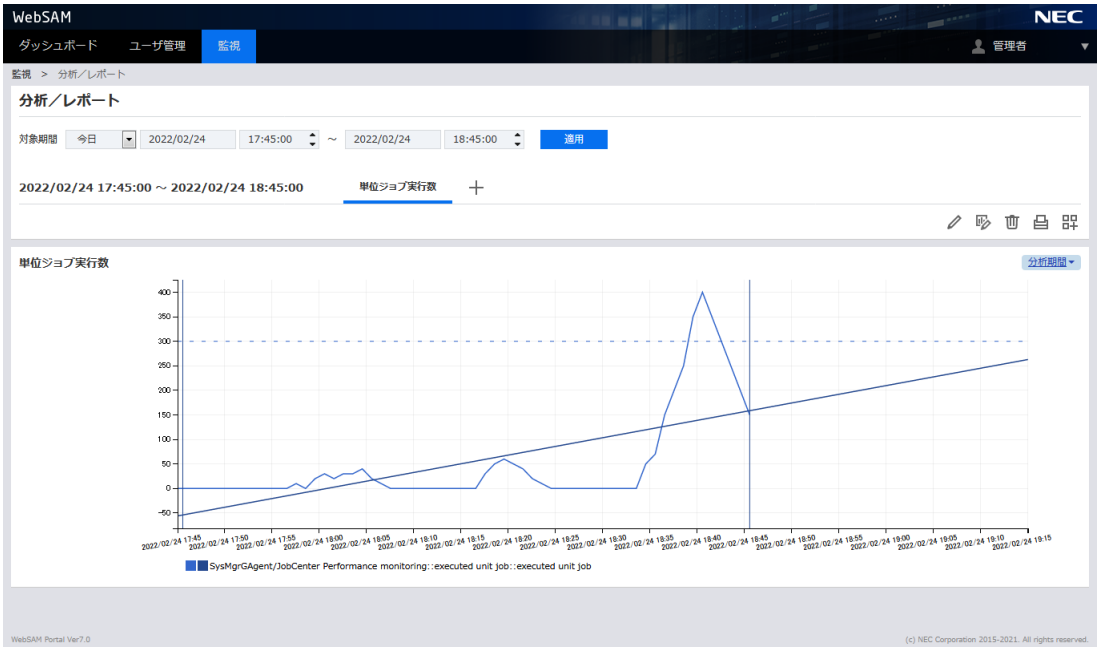

図22.26 分析結果の表示

22.3. メトリクス一覧

SystemManagerGのカスタム性能監視機能を用いて収集・分析可能なJobCenterのメトリクスは以下のとおりです。

収集したいメトリクスの章にある「性能値取得設定ファイルのパラメータ」の表を参考にカスタム性能監視を設定してください。性能値取得設定ファイルのパラメータの説明は「22.3.1　性能値取得設定ファイルのパラメータ」に記載していますので最初にご確認ください。

収集コマンドの詳細は<コマンドリファレンス>の9章 「メトリクス収集コマンド」 を参照してください。

表22.6 収集・分析可能な性能値一覧

分類	メトリクス	収集コマンド
トラッカ	「22.3.2 単位ジョブ実行数」	「9.1 jc_collect_runjob 単位ジョブ実行数のメトリクスを収集」
	「22.3.3 ジョブネットワーク実行数」	「9.2 jc_collect_runjnw ジョブネットワーク実行数のメトリクスを収集」
	「22.3.4 投入待ちジョブ部品数」	「9.3 jc_collect_waitjob 投入待ちジョブ部品数のメトリクスを収集」
	「22.3.5 ジョブネットワーク開始時間差」	「9.4 jc_collect_trk_starttime_delay ジョブネットワーク開始時間差のメトリクスを収集」
ディスク容量	「22.3.6 未アーカイブトラッカ全体サイズ」	「9.5 jc_collect_trk_size トラッカサイズのメトリクスを収集」
	「22.3.7 アーカイブ済みトラッカ全体サイズ」	
	「22.3.8 最大トラッカサイズ」	
キュー	「22.3.9 キュー使用率」	「9.6 jc_collect_queue キュー使用率およびリクエスト増減数のメトリクスを収集」
	「22.3.10 リクエスト増減数」	

22.3.1. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定するパラメータについて説明します。

> パラメータの添え字(n)はgetcustomperflist.propertiesの設定内容によって指定する値が変わります。設定方法は「22.2.1　性能値取得設定ファイル(getcustomperflist.properties)の設定」を参照してください。

22.3.1.1. 共通パラメータ

JobCenterのメトリクスの共通パラメータです。getcustomperflist.propertiesファイル内で1つ設定してください。

各メトリクスの「性能値取得設定ファイルのパラメータ」の章に記載している表では記載を省略していますのでご注意ください。

表22.7 性能値取得設定ファイルの共通パラメータ

キー	説明
object.count	オブジェクトの定義数です。 getcustomperflist.propertiesファイルに記述しているobjectの数を設定してください。
object[n].name	オブジェクト名です。 カスタム性能監視の監視項目選択画面でオブジェクトを選択する際に表示される名称になります。 「22.2.1 性能値取得設定ファイル(getcustomperflist.properties)の設定」で説明している設定内容は、JobCenterのメトリクスを1つのオブジェクトとしてグルーピングしていますので、getcustomperflist.propertiesファイル内で1つ設定してください。
object[n].instance .count	インスタンスの定義数です。 設定するJobCenterのメトリクスの数を指定します。例えば「22.3.2 単位ジョブ実行数」と「22.3.3 ジョブネットワーク実行数」の2つを設定する場合は「2」を指定します。

22.3.1.2. 個別パラメータ

JobCenterのメトリクスの個別のパラメータです。

各メトリクスの「性能値取得設定ファイルのパラメータ」の章に記載している表のとおりに設定してください。

表22.8 性能値取得設定ファイルの個別パラメータ

キー	説明
object[n].instance[n] .name	インスタンス名です。 カスタム性能監視の監視項目選択画面でインスタンスを選択する際に表示される名称になります。
object[n].instance[n] .command	対象のメトリクスを収集するためのコマンドおよび引数を指定します。コマンドは絶対パスで指定してください。 各メトリクスの「性能値取得設定ファイルのパラメータ」の章に記載している表は以下のように記載しています。

キー	説明
	<メトリクス収集コマンド格納ディレクトリ>/<コマンド名> <引数> <メトリクス収集コマンド格納ディレクトリ>はJobCenterのコンポーネントに合わせて読み替えてください。 ■UNIX版JobCenter /usr/lib/nqs/gui/bin ■Windows版JobCenter %InstallDirectory%\bin ■%InstallDirectory%はJobCenterのインストールディレクトリです。初期値でインストールした場合はC:\JobCenter\SVとなります。 ■%InstallDirectory%に半角スペースが含まれる場合、以下のようにコマンドのパス全体をダブルクォーテーションで括る必要があります。 "%InstallDirectory%\bin\<コマンド名>" <引数> ■クラスタサイトのメトリクスを収集する場合、シェルスクリプトやバッチファイル経由でメトリクス収集コマンドを実行する必要があります。詳細は、表外の「クラスタサイトのメトリクスを収集する場合」を参照してください。
object[n].instance[n] .command_delimiter	コマンド実行結果のカラムの区切り文字です。 メトリクス収集コマンドの実行結果はCSV形式で出力されますので、「","」を指定してください。
object[n].instance[n] .counter.count	カウンタ定義数を指定します。 対象のメトリクスで収集する値の数を指定します。例えば「22.3.3 ジョブネットワーク実行数」は「未アーカイブのトラッカ数」と「実行中となっているトラッカ数」の2つの値を収集しますので「2」を指定します。
object[n].instance[n] .counter[n].name	カウンタ名です。 カスタム性能監視の監視項目選択画面でカウンタを選択する際に表示される名称になります。
object[n].instance[n] .counter[n].position_row	コマンド実行結果のデータの位置(行)です。 メトリクス収集コマンドの実行結果は1行で出力されますので、「1」を指定してください。
object[n].instance[n] .counter[n].position_column	コマンド実行結果のデータの位置(列)です。 各メトリクスの「性能値取得設定ファイルのパラメータ」の章に記載している表のとおりに設定してください。
object[n].instance[n] .counter[n].text	カウンタの説明を指定します。 設定値は任意です。また、パラメータの設定自体も省略可能です。

■クラスタサイトのメトリクスを収集する場合

クラスタサイトのメトリクスを収集する場合、環境変数NQS_SITEにクラスタサイト名を設定してメトリクス収集コマンドを実行する必要があります。また、コマンドによっては環境変数NQS_SITEDBにサイトDBのパスを設定する必要があります。コマンドの実行に必要な環境変数については<コマンドリファレンス>の表1.1「コマンド一覧」を参照してください。

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)では、コマンド実行時の環境変数を設定できませんので、シェルスクリプトやバッチファイル経由でメトリクス収集コマンドを実行します。

例えば、メトリクス「22.3.2 単位ジョブ実行数」をクラスタサイトで収集する場合は、以下の手順で設定します。

1. 環境変数NQS_SITEおよびNQS_SITEDBを設定して、メトリクス収集コマンドjc_collect_runjobを実行するスクリプトを作成します。

   - UNIX版JobCenterの場合のシェルスクリプト例

   ```
   #!/bin/sh
   NQS_SITE="cluster"
   export NQS_SITE

   NQS_SITEDB="/cluster_db"
   export NQS_SITEDB

   /usr/lib/nqs/gui/bin/jc_collect_runjob -r $1
   ```

   上記の例では、クラスタサイト名が「cluster」、サイトDBのパスが「/cluster_db」となります。

   - Windows版JobCenterの場合のバッチファイル例

   ```
   @echo off
   set NQS_SITE=cluster
   set NQS_SITEDB=C:\cluster_db

   C:\JobCenter\SV\bin\jc_collect_runjob -r %1
   ```

   上記の例では、クラスタサイト名が「cluster」、サイトDBのパスが「C:\cluster_db」、JobCenterのインストールディレクトリが「C:\JobCenter\SV」となります。

2. 作成したシェルスクリプトまたはバッチファイルを、JobCenterがインストールしているマシンに格納します。

   UNIX版JobCenterの場合は、シェルスクリプトの実行権限も付与してください。

3. SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)のobject[n].instance[n].commandに、作成したシェルスクリプトまたはバッチファイルの絶対パスおよび引数を指定します。

   ```
   <シェルスクリプトまたはバッチファイルの格納パス> $INTERVAL$
   ```

22.3.2. 単位ジョブ実行数

22.3.2.1. 概要

指定期間内に実行された、または実行中のJobCenter全体の単位ジョブ部品の数です。

本メトリクスは、メトリクス収集コマンドjc_collect_runjobで収集します。jc_collect_runjobコマンドの詳細は<コマンドリファレンス>の「9.1 jc_collect_runjob 単位ジョブ実行数のメトリクスを収集」を参照してください。

特定の時間帯にジョブの実行が集中すると、キューにリクエストが滞留したりジョブ実行サーバーの負荷が増大しJobCenterの動作に影響を及ぼします。

本メトリクスと「22.3.9 キュー使用率」や「22.3.10 リクエスト増減数」と合わせて比較・分析することで、ジョブの実行が集中している時間帯とジョブ実行サーバの特定ができます。ジョブ実行の時間帯の見直しやパイプキューを利用した負荷分散を実施することで適切な運用改善ができます。

■本メトリクスの対象は単位ジョブ部品のみです。

■指定期間内に存在するトラッカ内の単位ジョブ部品が、以下の状態の場合に対象となります。

- DONE（「実績開始時刻」が指定期間内）
- RUN
- ERRORSTOP

22.3.2.2. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定する値は以下のとおりです。

表22.9 単位ジョブ実行数の性能値取得設定ファイルの設定値

必須	キー	値
○	object[n].instance[n].name	executed unit job
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_runjob -r $INTERVAL$
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	1
○	object[n].instance[n].counter1.name	executed unit job
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	2
	object[n].instance[n].counter1.text	Number of executed unit jobs in the last minute.

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

22.3.2.3. 閾値の推奨値

単位ジョブ実行数の推奨値はありません。

本メトリクスと「22.3.9 キュー使用率」や「22.3.10 リクエスト増減数」と合わせて、ジョブの実行が集中している時間帯とジョブ実行サーバの特定にご利用ください。

22.3.3. ジョブネットワーク実行数

22.3.3.1. 概要

実行中となっているトラッカ数の時間平均値および未アーカイブのトラッカ数の時間平均値です。

本メトリクスは、メトリクス収集コマンドjc_collect_runjnwで収集します。jc_collect_runjnwコマンドの詳細は<コマンドリファレンス>の「9.2 jc_collect_runjnw ジョブネットワーク実行数のメトリクスを収集」を参照してください。

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。

実行中となっているトラッカ数や未アーカイブのトラッカ数が多いということは、それだけ多くのトラッカをメモリ上にロードしている状態であるため、メモリが不足しJobCenterの動作に影響を及ぼす可能性があります。

実行中となっているトラッカ数が多いタイミングでは処理が集中している可能性があり、他の処理が終了するまで待ち合わせが発生し予定していた時刻に処理を開始できない可能性があります。

適切なスケジュール設定、アーカイブ期間、パイプキューを利用した負荷分散などを実施することで改善可能です。

> ⚠ 本メトリクスを使用する場合、事前にjc_perfコマンドのconfigサブオプションで、rtrkをonにしておく必要があります。
>
> jc_perfコマンドの詳細については<コマンドリファレンス>の「3.25 jc_perf 実行状況の分析」を参照してください。

22.3.3.2. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定する値は以下のとおりです。

表22.10 ジョブネットワーク実行数の性能値取得設定ファイルの設定値

必須	キー	値
○	object[n].instance[n].name	running jobnetwork
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_runjnw
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	2
○	object[n].instance[n].counter1.name	Number of unarchived trackers
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	2
	object[n].instance[n].counter1.text	The time average of the number of unarchived trackers in the last minute.
○	object[n].instance[n].counter2.name	Number of trackers with a running status
◎	object[n].instance[n].counter2.position_row	1
◎	object[n].instance[n].counter2.position_column	3
	object[n].instance[n].counter2.text	The time average of the number of trackers with a running status for the last minute.

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

22.3.3.3. 閾値の推奨値

ジョブネットワーク実行数の推奨値はありません。

「23.2 メモリ使用量概算算出方法（UNIX版）」または「23.5 メモリ使用量概算算出方法（Windows版）」を参考に閾値を決定してください。

22.3.4. 投入待ちジョブ部品数

22.3.4.1. 概要

ジョブネットワークのフロー上で、前段の部品が実行完了し、投入可能状態(ジョブの実行待ち状態)となっているジョブの数を表します。

このジョブの数は、1分間の時間平均値です。

本メトリクスは、メトリクス収集コマンドjc_collect_waitjobで収集します。jc_collect_waitjobコマンドの詳細は<コマンドリファレンス>の「9.3 jc_collect_waitjob 投入待ちジョブ部品数のメトリクスを収集」を参照してください。

並列度の高いジョブネットワークや、大量のジョブネットワークを同時に稼働させるときなど、並列処理能力を超えた量のジョブが投入されると投入待ちジョブ部品数は増加します。

投入待ちジョブ部品数が多い場合、他のジョブが完了するのを待っているジョブが増加している状態です。投入待ちジョブ部品数が多いほど、それらのジョブが開始されるまでの時間が長くなります。

本メトリクスを使用する場合、事前にjc_perfコマンドのconfigサブオプションで、rjobをonにしておく必要があります。

jc_perfコマンドの詳細については<コマンドリファレンス>の「3.25 jc_perf 実行状況の分析」を参照してください。

22.3.4.2. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定する値は以下のとおりです。

表22.11 投入待ちジョブ部品数の性能値取得設定ファイルの設定値

必須	キー	値
○	object[n].instance[n].name	Number of waiting job
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_waitjob
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	1
○	object[n].instance[n].counter1.name	Number of waiting job
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	2
	object[n].instance[n].counter1.text	The time average of the number of waiting job in the last minute.

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

22.3.4.3. 閾値の推奨値

投入待ちジョブ部品数の推奨値は 1 です。

1を超える場合、1分間に平均1つ以上のジョブが投入待ち状態となっていることを示します。

22.3.5. ジョブネットワーク開始時間差

22.3.5.1. 概要

一定期間内に実行を開始したトラッカの開始予定時刻と実績開始時刻の差分の平均値です。

本メトリクスは、メトリクス収集コマンドjc_collect_trk_starttime_delayで収集します。jc_collect_trk_starttime_delayコマンドの詳細は<コマンドリファレンス>の「9.4 jc_collect_trk_starttime_delay ジョブネットワーク開始時間差のメトリクスを収集」を参照してください。

本メトリクスの値が大きくなっている場合、ジョブネットワークの実行開始に遅延が発生しスケジュール通りにジョブが実行されない可能性があります。

運用にて特定の時間にスケジュール起動が集中している場合はスケジュールの見直しを行い起動時間を分散させてください。

また、ジョブネットワークの同時実行数制御の設定を行っている場合、先行ジョブネットワークの完了が遅延している影響で実行待ちとなっている可能性もあります。ジョブネットワークの同時実行数制御は21章「SystemManager GでJobCenterの障害情報を監視する」 の障害情報「21.3.13 同時起動可能数を超えた」で監視可能です。

22.3.5.2. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定する値は以下のとおりです。

表22.12 ジョブネットワーク開始時間差の性能値取得設定ファイルの設定値

必須	キー	値
○	object[n].instance[n].name	jobnetwork start time difference
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_trk_starttime_delay -r $INTERVAL$
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	1
○	object[n].instance[n].counter1.name	jobnetwork start time difference
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	2
	object[n].instance[n].counter1.text	The average time difference between the jobnetwork started in the last minute and the scheduled start time. (seconds)

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

22.3.5.3. 閾値の推奨値

閾値を1秒にすると誤差程度の遅延も検知してしまい、運用上の問題か否かの判断が難しくなります。ジョブネットワーク開始に明確な遅延が発生したことを検知するため、閾値は2秒を推奨します。

22.3.6. 未アーカイブトラッカ全体サイズ

22.3.6.1. 概要

JobCenter全体の未アーカイブトラッカが占めるディスク容量です。

本メトリクスは、メトリクス収集コマンドjc_collect_trk_sizeで収集します。jc_collect_trk_sizeコマンドの詳細は<コマンドリファレンス>の「9.5 jc_collect_trk_size トラッカサイズのメトリクスを収集」を参照してください。

未アーカイブトラッカが占めるディスク容量が増加した場合、ディスクフルによってJobCenterをインストールしているマシン全体に障害が発生する可能性があります。

未アーカイブトラッカのディスク容量増加の原因は以下が考えられます。本メトリクスは、前者の「未アーカイブトラッカが大量に存在している」を監視するのに有効です。後者については「22.3.8　最大トラッカサイズ」のメトリクスをご利用ください。

■未アーカイブトラッカが大量に存在している

- ジョブを実行するマシンの性能を上回るジョブを実行している
- 未完了のトラッカが大量に存在している
- 完了済みの未アーカイブトラッカが大量に存在している

■データサイズが巨大なトラッカが存在している

- 単位ジョブ部品の標準出力結果または標準エラー出力結果に大量のログを出力している

本メトリクスを監視することで、未アーカイブトラッカのディスク容量の推移を監視し、ディスクフルの障害を未然に防ぐことができます。

■トラッカは以下の2種類があります。本メトリクスでは未アーカイブトラッカを対象にします。

- 未アーカイブトラッカ

  実行中またはステータスが[予定(確定)]のトラッカです。[エラー停止]のトラッカも対象となります。

  未アーカイブトラッカのデータは、JobCenterのサイトの各種データを格納しているスプールディレクトリに格納され、アーカイブされるまで保持されます。スプールディレクトリの詳細は<リリースメモ>の「3.2.3.1 スプールディレクトリ」を参照してください。

  また、未アーカイブトラッカの一部のデータはJobCenterのメモリ上にも展開されます。JobCenterのリソースを消費し続けるため、定期的にアーカイブすることを推奨します。

- アーカイブ済みトラッカ

  トラッカの実行完了後、アーカイブファイルにトラッカ情報が記録された状態のトラッカです。トラッカのステータスに[アーカイブ済]と表示されます。

  アーカイブのタイミングやアーカイブファイルの格納先は、ユーザ環境設定の[アーカイブ]タブで設定可能です。設定方法は「6.4 アーカイブの設定を変更する」を参照してください。

■トラッカのディスク容量に関連して、以下のメトリクスも合わせて監視することを推奨します。

- 「22.3.7 アーカイブ済みトラッカ全体サイズ」
- 「22.3.8 最大トラッカサイズ」

22.3.6.2. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定する値は以下のとおりです。

表22.13 未アーカイブトラッカ全体サイズの性能値取得設定ファイルの設定値

必須	キー	値
○	object[n].instance[n].name	Unarchived tracker overall size
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_trk_size tracker -t total
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	1
○	object[n].instance[n].counter1.name	Unarchived tracker overall size
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	2
	object[n].instance[n].counter1.text	Disk capacity occupied by the unarchived trackers for all users. (byte)

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

<コマンドリファレンス>の「9.5 jc_collect_trk_size トラッカサイズのメトリクスを収集」は、-t allオプションを指定することで「22.3.6 未アーカイブトラッカ全体サイズ」と「22.3.8 最大トラッカサイズ」のメトリクスを同時に取得することが可能です。

「22.3.6 未アーカイブトラッカ全体サイズ」と「22.3.8 最大トラッカサイズ」のメトリクスを両方収集する場合は、以下の設定を行うことで同時に取得可能です。

表22.14 未アーカイブトラッカ全体サイズと最大トラッカサイズを同時に取得する場合の設定値

必須	キー	値
○	object[n].instance[n].name	Unarchived tracker size
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_trk_size tracker -t all
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	2
○	object[n].instance[n].counter1.name	Unarchived tracker overall size
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	2
	object[n].instance[n].counter1.text	Disk capacity occupied by the unarchived trackers for all users. (byte)
○	object[n].instance[n].counter2.name	Maximum tracker size
◎	object[n].instance[n].counter2.position_row	1
◎	object[n].instance[n].counter2.position_column	3
	object[n].instance[n].counter2.text	Disk capacity value of the largest tracker among all unarchived trackers. (byte)

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

22.3.6.3. 閾値の推奨値

未アーカイブトラッカのデータは、アーカイブファイルに記録された後に削除されます。そのため、通常は本メトリクスが増加し続けることはありません。

本メトリクスが増加し続ける場合、ステータスが[応答待ち]や[エラー停止]などの未完了のトラッカが増加し続けているなど、運用上の問題が発生している可能性があります。本メトリクスの性能グラフが増加傾向にないかを監視するためにご利用ください。

閾値については、お客様の環境で本メトリクスのピーク時の値を参考に設定してください。

22.3.6.4. 問題解決までの流れ

未アーカイブトラッカが大量に存在している場合の解決方法は以下のとおりです。

1. SystemManagerGのカスタム性能監視にて、本メトリクスを監視します。
2. SystemManagerGのエージェントから、本メトリクスの閾値超過のメッセージを受信します。
3. JobCenter管理者ユーザとしてCL/WinでJobCenterに接続し、[マネージャフレーム]画面の[トラッカ一覧@全マシン]タブを表示します。
4. [コントロールダイアログ]から未完了のトラッカを検索します。[コントロールダイアログ]の使用方法は<基本操作ガイド>の「8.10 トラッカ一覧をフィルタリングして表示する」を参照してください。
5. 検索結果によって、以下の対処を行います。

■実行中のトラッカが大量に存在している

ジョブを実行するマシンの性能を上回るジョブを実行している可能性があります。パイプキューを利用した負荷分散機能の利用や、マシン性能の見直しをご検討ください。

ジョブが問題なく処理できているかは、「22.3.9 キュー使用率」や「22.3.10 リクエスト増減数」のメトリクスで監視・分析可能です。

■未完了のトラッカが大量に存在している

ステータスが[応答待ち]や[エラー停止]のトラッカは自動的にアーカイブされません。該当するトラッカを確認し、完了状態にしてください。

■完了済みの未アーカイブトラッカが大量に存在している

アーカイブのタイミングを見直してください。

アーカイブファイルのタイミングは、ユーザ環境設定の[アーカイブ]タブで設定可能です。設定方法は「6.4 アーカイブの設定を変更する」を参照してください。

■未完了のトラッカも完了済みの未アーカイブトラッカもほとんど存在しない

データサイズが巨大なトラッカが存在している可能性があります。「22.3.7 アーカイブ済みトラッカ全体サイズ」や「22.3.8 最大トラッカサイズ」のメトリクスを監視して、「問題解決までの流れ」の章を参考に原因のトラッカを調査してください。

22.3.6.5. 注意事項

■メトリクス収集コマンドjc_collect_trk_sizeについて、HP-UXおよびAIXの環境はサポートしていませんので、本メトリクスの利用もサポートしておりません。

22.3.7. アーカイブ済みトラッカ全体サイズ

22.3.7.1. 概要

JobCenter全体のアーカイブ済みトラッカが占めるディスク容量です。

本メトリクスは、メトリクス収集コマンドjc_collect_trk_sizeで収集します。jc_collect_trk_sizeコマンドの詳細は<コマンドリファレンス>の「9.5 jc_collect_trk_size トラッカサイズのメトリクスを収集」を参照してください。

アーカイブ済みトラッカが占めるディスク容量が増加した場合、ディスクフルによってJobCenterをインストールしているマシン全体に障害が発生する可能性があります。

■アーカイブ済みトラッカはアーカイブファイルに記録されます。すなわち、本メトリクスはJobCenter全体のアーカイブファイルの合計サイズとなります。

■本メトリクスは、ユーザ環境設定の[アーカイブ]タブの[アーカイブファイル名]に設定している格納先のアーカイブファイルを対象に集計します。現在の設定値ではない格納先のアーカイブファイルは集計対象外となります。

アーカイブ済みトラッカのディスク容量増加の原因は以下が考えられます。

■アーカイブ済みトラッカが大量に存在している

■データサイズが巨大なトラッカが存在している

- 単位ジョブ部品の標準出力結果または標準エラー出力結果に大量のログを出力している

本メトリクスを監視することで、アーカイブ済みトラッカのディスク容量の推移を監視し、ディスクフルの障害を未然に防ぐことができます。

■トラッカは以下の2種類があります。本メトリクスではアーカイブ済みトラッカを対象にします。

- 未アーカイブトラッカ

  実行中またはステータスが[予定(確定)]のトラッカです。[エラー停止]のトラッカも対象となります。

  未アーカイブトラッカのデータは、JobCenterのサイトの各種データを格納しているスプールディレクトリに格納され、アーカイブされるまで保持されます。スプールディレクトリの詳細は<リリースメモ>の「3.2.3.1 スプールディレクトリ」を参照してください。

  また、未アーカイブトラッカの一部のデータはJobCenterのメモリ上にも展開されます。JobCenterのリソースを消費し続けるため、定期的にアーカイブすることを推奨します。

- アーカイブ済みトラッカ

  トラッカの実行完了後、アーカイブファイルにトラッカ情報が記録された状態のトラッカです。トラッカのステータスに[アーカイブ済]と表示されます。

  アーカイブのタイミングやアーカイブファイルの格納先は、ユーザ環境設定の[アーカイブ]タブで設定可能です。設定方法は「6.4 アーカイブの設定を変更する」を参照してください。

■トラッカのディスク容量に関連して、以下のメトリクスも合わせて監視することを推奨します。

- 「22.3.6 未アーカイブトラッカ全体サイズ」

- 「22.3.8 最大トラッカサイズ」

22.3.7.2. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定する値は以下のとおりです。

表22.15 アーカイブ済みトラッカ全体サイズの性能値取得設定ファイルの設定値

必須	キー	値
○	object[n].instance[n].name	Archive tracker overall size
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_trk_size archive
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	1
○	object[n].instance[n].counter1.name	Archive tracker overall size
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	2
	object[n].instance[n].counter1.text	Disk capacity occupied by the archive trackers for all users. (byte)

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

22.3.7.3. 閾値の推奨値

ディスクフルが発生すると、JobCenterをインストールしているマシン全体に障害が発生するため、余裕のある閾値を設定してください。

JobCenterが利用するディスク容量全体の60%を推奨します。

22.3.7.4. 問題解決までの流れ

アーカイブ済みトラッカのディスク容量増加の解決方法は以下のとおりです。

1. SystemManagerGのカスタム性能監視にて、本メトリクスを監視します。
2. SystemManagerGのエージェントから、本メトリクスの閾値超過のメッセージを受信します。
3. 障害解析コマンドjc_analyze_trk_archiveの「アーカイブファイルの一覧表示」機能を利用して、JobCenter全体のアーカイブファイル一覧を確認します。

障害解析コマンドjc_analyze_trk_archiveの詳細は<コマンドリファレンス>の「10.1 jc_analyze_trk_archive アーカイブファイル内のトラッカ情報を解析」を参照してください。

■障害解析コマンドjc_analyze_trk_archiveの「アーカイブファイルの一覧表示」機能の実行例

```
# /usr/lib/nqs/gui/bin/jc_analyze_trk_archive list archive
User            Archive                                                    Size
user3           /usr/spool/nqs/users/user3/archive/trkarc20211209-20211209   30458905
user3           /usr/spool/nqs/users/user3/archive/trkarc20211206-20211206   15259
user1           /usr/spool/nqs/users/user1/archive/trkarc20211207-20211207   11270
user2           /usr/spool/nqs/users/user2/archive/trkarc20211129-20211129   9973
                     :
```

4. 検索結果によって、以下の対処を行います。

■データサイズが巨大なアーカイブファイルが存在している

データサイズが巨大なトラッカが存在している可能性があります。原因のトラッカを特定するため、次の手順へ進んでください。

■データサイズが巨大なアーカイブファイルはない（アーカイブファイルが大量に存在している）

アーカイブファイルの格納先をディスク容量に余裕のあるディレクトリに変更するか、不要なトラッカを削除してください。

アーカイブファイルの格納先は、ユーザ環境設定の[アーカイブ]タブで設定可能です。設定方法は「6.4 アーカイブの設定を変更する」を参照してください。

5. データサイズが巨大なトラッカを特定するため、手順4で確認したアーカイブファイルを検証環境にコピーします。

■アーカイブファイルの解析には、対象トラッカのデータサイズの最大2倍のディスク容量が必要です。本番環境で解析を行うとディスクフルの危険性があるため、検証環境で実施してください。

■アーカイブファイルをコピーする際は、以下のどちらかの状態で実施してください。

- JobCenterが停止している。
- 対象のアーカイブファイルが非アクティブの状態、かつ対象のアーカイブファイルのトラッカを操作していない状態である。

非アクティブの状態とは、未アーカイブトラッカの記録対象から外れた状態のことを指します。例えば、ユーザ環境設定の[アーカイブ]タブで[保存間隔]を1日に設定している場合、1日前のアーカイブファイルは非アクティブの状態となります。

6. 障害解析コマンドjc_analyze_trk_archiveの「アーカイブファイル内のトラッカの一覧表示」機能を利用して、データサイズが巨大なトラッカのトラッカIDを確認します。

■障害解析コマンドjc_analyze_trk_archiveの「アーカイブファイル内のトラッカの一覧表示」機能の実行例

```
# /usr/lib/nqs/gui/bin/jc_analyze_trk_archive list tracker -r /tmp/trkarc20211209-20211209
TrackerID             Size
jnw2.20211209001607   30408905
jnw3.20211209001757   1512
jnw3.20211209001516   1311
jnw1.20211209011607   1297
                      :
```

7. 障害解析コマンドjc_analyze_trk_archiveの「単位ジョブの出力結果をファイル化」機能を利用して、データサイズが巨大なトラッカに含まれる単位ジョブ部品の標準出力結果および標準エラー出力結果をファイル化します。

■障害解析コマンドjc_analyze_trk_archiveの「単位ジョブの出力結果をファイル化」機能の実行例

```
# /usr/lib/nqs/gui/bin/jc_analyze_trk_archive extract -r /tmp/trkarc20211209-20211209 -t
 jnw2.20211209001607 -o /tmp/output
Tracker data was extracted successfully. (/tmp/output)
```

8. 指定したトラッカの単位ジョブの標準出力結果および標準エラー出力結果の内容がテキストファイルとして保存されます。

   OSのエクスプローラ機能やテキストエディタ等でファイルサイズや出力結果の内容を確認し、出力結果が巨大となった原因を調査してください。

22.3.7.5. 注意事項

■メトリクス収集コマンドjc_collect_trk_sizeについて、HP-UXおよびAIXの環境はサポートしていませんので、本メトリクスの利用もサポートしておりません。

22.3.8. 最大トラッカサイズ

22.3.8.1. 概要

JobCenter全体の未アーカイブトラッカのうち、データサイズが最大となるディスク容量です。

本メトリクスは、メトリクス収集コマンドjc_collect_trk_sizeで収集します。jc_collect_trk_sizeコマンドの詳細は<コマンドリファレンス>の「9.5 jc_collect_trk_size トラッカサイズのメトリクスを収集」を参照してください。

未アーカイブトラッカが占めるディスク容量が増加した場合、ディスクフルによってJobCenterをインストールしているマシン全体に障害が発生する可能性があります。

未アーカイブトラッカのディスク容量増加の原因は以下が考えられます。本メトリクスは、後者の「データサイズが巨大なトラッカが存在している」を監視するのに有効です。前者については「22.3.6　未アーカイブトラッカ全体サイズ」のメトリクスをご利用ください。

■未アーカイブトラッカが大量に存在している

- ジョブを実行するマシンの性能を上回るジョブを実行している
- 未完了のトラッカが大量に存在している
- 完了済みの未アーカイブトラッカが大量に存在している

■データサイズが巨大なトラッカが存在している

- 単位ジョブ部品の標準出力結果または標準エラー出力結果に大量のログを出力している

本メトリクスを監視することで、未アーカイブトラッカのディスク容量の推移を監視し、ディスクフルの障害を未然に防ぐことができます。

■トラッカは以下の2種類があります。本メトリクスでは未アーカイブトラッカを対象にします。

- 未アーカイブトラッカ

  実行中またはステータスが[予定(確定)]のトラッカです。[エラー停止]のトラッカも対象となります。

  未アーカイブトラッカのデータは、JobCenterのサイトの各種データを格納しているスプールディレクトリに格納され、アーカイブされるまで保持されます。スプールディレクトリの詳細は<リリースメモ>の「3.2.3.1 スプールディレクトリ」を参照してください。

  また、未アーカイブトラッカの一部のデータはJobCenterのメモリ上にも展開されます。JobCenterのリソースを消費し続けるため、定期的にアーカイブすることを推奨します。

- アーカイブ済みトラッカ

  トラッカの実行完了後、アーカイブファイルにトラッカ情報が記録された状態のトラッカです。トラッカのステータスに[アーカイブ済]と表示されます。

  アーカイブのタイミングやアーカイブファイルの格納先は、ユーザ環境設定の[アーカイブ]タブで設定可能です。設定方法は「6.4 アーカイブの設定を変更する」を参照してください。

■トラッカのディスク容量に関連して、以下のメトリクスも合わせて監視することを推奨します。

- 「22.3.6 未アーカイブトラッカ全体サイズ」
- 「22.3.7 アーカイブ済みトラッカ全体サイズ」

22.3.8.2. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定する値は以下のとおりです。

表22.16 最大トラッカサイズの性能値取得設定ファイルの設定値

必須	キー	値
○	object[n].instance[n].name	Maximum tracker size
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_trk_size tracker -t max
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	1
○	object[n].instance[n].counter1.name	Maximum tracker size
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	2
	object[n].instance[n].counter1.text	Disk capacity value of the largest tracker among all unarchived trackers. (byte)

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

<コマンドリファレンス>の「9.5 jc_collect_trk_size トラッカサイズのメトリクスを収集」は、-t allオプションを指定することで「22.3.6 未アーカイブトラッカ全体サイズ」と「22.3.8 最大トラッカサイズ」のメトリクスを同時に取得することが可能です。

「22.3.6 未アーカイブトラッカ全体サイズ」と「22.3.8 最大トラッカサイズ」のメトリクスを両方収集する場合は、以下の設定を行うことで同時に取得可能です。

表22.17 未アーカイブトラッカ全体サイズと最大トラッカサイズを同時に取得する場合の設定値

必須	キー	値
○	object[n].instance[n].name	Unarchived tracker size
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_trk_size tracker -t all
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	2
○	object[n].instance[n].counter1.name	Unarchived tracker overall size
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	2
	object[n].instance[n].counter1.text	Disk capacity occupied by the unarchived trackers for all users. (byte)
○	object[n].instance[n].counter2.name	Maximum tracker size
◎	object[n].instance[n].counter2.position_row	1
◎	object[n].instance[n].counter2.position_column	3
	object[n].instance[n].counter2.text	Disk capacity value of the largest tracker among all unarchived trackers. (byte)

（凡例）◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

22.3.8.3. 閾値の推奨値

トラッカのデータの主な内容は、単位ジョブの標準出力結果・標準エラー出力結果です。1つのトラッカの標準出力結果・標準エラー出力結果の合計サイズが10MBを超える場合は、異常な状態であると考えられます。

そのため、閾値は10,485,760(10MB)を推奨します。お客様の環境に合わせて見直してください。

22.3.8.4. 問題解決までの流れ

データサイズが巨大なトラッカが存在している場合の解決方法は、「22.3.7 アーカイブ済みトラッカ全体サイズ」と同様です。「22.3.7.4 問題解決までの流れ」の手順を参照してください。

22.3.8.5. 注意事項

■メトリクス収集コマンドjc_collect_trk_sizeについて、HP-UXおよびAIXの環境はサポートしていませんので、本メトリクスの利用もサポートしておりません。

22.3.9. キュー使用率

22.3.9.1. 概要

指定したキューの許容値（同時実行可能数）に対して、どの程度リクエストが送られているかを表した値です。

本メトリクスは、メトリクス収集コマンドjc_collect_queueで収集します。jc_collect_queueコマンドの詳細は<コマンドリファレンス>の「9.6 jc_collect_queue キュー使用率およびリクエスト増減数のメトリクスを収集」を参照してください。

キュー使用率の値については以下の通りです。

表22.18 キュー使用率の値について

値	説明
0%の場合	その時間帯において指定したキューは使用されていない
1～100%の場合	その時間帯において指定したキューの処理能力範囲内のリクエストが送られている
101%～の場合	その時間帯において指定したキューの処理能力を超えたリクエストが送られている

キュー使用率の値が100%を超えている場合、処理待ち状態のリクエストが発生しているため、リクエストの処理開始が遅れることによるジョブの実行開始/終了時間の遅延に繋がります。

本メトリクスと「22.3.2 単位ジョブ実行数」や「22.3.10 リクエスト増減数」と合わせて比較・分析することで、ジョブの実行が集中している時間帯とジョブ実行サーバの特定ができます。ジョブ実行の時間帯の見直しやパイプキューを利用した負荷分散を実施することで適切な運用改善ができます。

本メトリクスを使用する場合、事前にjc_perfコマンドのconfigサブオプションで、qreqをonにしておく必要があります。

jc_perfコマンドの詳細については<コマンドリファレンス>の「3.25 jc_perf 実行状況の分析」を参照してください。

22.3.9.2. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定する値は以下のとおりです。

表22.19 キュー使用率の性能値取得設定ファイルの設定値

必須	キー	値
○	object[n].instance[n].name	Queue use rate
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_queue -q $queue
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	1
○	object[n].instance[n].counter1.name	Queue use rate
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	3
	object[n].instance[n].counter1.text	The ratio of how much is used to the number of concurrent executions of the target queue in the last minute.

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

22.3.9.3. 閾値の推奨値

キュー使用率の値が100%を超えるとキューの同時実行可能数を超えるリクエストが来ていることになり、リクエストの待ちが発生している可能性が高いです。よって、推奨値は 100 となります。

22.3.10. リクエスト増減数

22.3.10.1. 概要

指定したキューにキューイングされたリクエストのうち、直前1分間のリクエストの増減を表した値です。

本メトリクスは、メトリクス収集コマンドjc_collect_queueで収集します。jc_collect_queueコマンドの詳細は<コマンドリファレンス>の「9.6 jc_collect_queue キュー使用率およびリクエスト増減数のメトリクスを収集」を参照してください。

リクエスト増減数の値については以下となります。

表22.20 リクエスト増減数の値について

値	説明
0の場合	その時間帯において投入されたリクエストをすべて処理できている状態
正の場合	その時間帯において処理したリクエスト数以上に、リクエストが投入されている状態（リクエストが滞留）
負の場合	その時間帯において投入されたリクエスト数以上にリクエストの処理がされた状態（滞留していたリクエストを処理）

リクエスト増減数の値が正の場合が続くほど、処理されず滞留しているリクエスト数が増えているため、リクエストの処理開始が遅れることによるジョブの実行開始/終了時間の遅延に繋がります。よって、キューの同時実行可能数の上限を上げるなどの対処が必要です。

リクエスト増減数の取得にはjc_collect_queueコマンドを使用します。コマンドを実行すると指定したキューの直前1分間のキュー使用率とリクエスト増減数の値を出力します。コマンドの詳細については<コマンドリファレンス>の「9.6 jc_collect_queue キュー使用率およびリクエスト増減数のメトリクスを収集」を参照してください。

本メトリクスと「22.3.2　単位ジョブ実行数」や「22.3.9　キュー使用率」と合わせて比較・分析することで、ジョブの実行が集中している時間帯とジョブ実行サーバの特定ができます。ジョブ実行の時間帯の見直しやパイプキューを利用した負荷分散を実施することで適切な運用改善ができます。

本メトリクスを使用する場合、事前にjc_perfコマンドのconfigサブオプションで、qreqをonにしておく必要があります。

jc_perfコマンドの詳細については<コマンドリファレンス>の「3.25　jc_perf　実行状況の分析」を参照してください。

22.3.10.2. 性能値取得設定ファイルのパラメータ

SystemManagerGの性能値取得設定ファイル(getcustomperflist.properties)に設定する値は以下のとおりです。

表22.21 リクエスト増減数の性能値取得設定ファイルの設定値

必須	キー	値
○	object[n].instance[n].name	Request fluctuation
◎	object[n].instance[n].command	<メトリクス収集コマンド格納ディレクトリ>/jc_collect_queue -q $queue
◎	object[n].instance[n].command_delimiter	","
◎	object[n].instance[n].counter.count	1
○	object[n].instance[n].counter1.name	Request fluctuation

必須	キー	値
◎	object[n].instance[n].counter1.position_row	1
◎	object[n].instance[n].counter1.position_column	4
	object[n].instance[n].counter1.text	The increase / decrease in the number of requests queued in the target queue for the last minute.

(凡例) ◎：必須(固定値)、○：必須(任意の値)、(空欄)：省略可(任意の値)

22.3.10.3. 閾値の推奨値

リクエスト増減数の推奨値はありません。

本メトリクスと「22.3.2 単位ジョブ実行数」や「22.3.9 キュー使用率」と合わせて、ジョブの実行が集中している時間帯とジョブ実行サーバの特定にご利用ください。

23. システム利用資源

JobCenterにおける利用資源の考え方を説明します。

JobCenterは処理の内容により、瞬間的に複数のファイルを開いたりサブプロセスを実行したりしながら動作していますので、目安としてお考えください。

JobCenterで大量のジョブリクエストを短時間に生成して実行する場合、OSの様々なカーネルパラメータ、またはシェル制限の上限値に抵触する可能性があります(例えば、Linuxのfs.file-max、kernel.threads-max、nproc等)。

本章のリソース使用量を参照の上、集中的にジョブリクエストを実行する際に消費するリソースについて、カーネルパラメータやシェル制限のチューニングを行ってください。

システム全体のリソース制限はカーネルパラメータで設定してください。ユーザ毎のリソース制限はシェル制限（例えば、Linuxの/etc/security/limits.confファイルやulimitコマンド）で設定してください。カーネルパラメータやシェル制限の内容、および設定方法の詳細については、ご使用のOSのマニュアル等を参照してください。

23.1. UNIX版JobCenterのプロセスが使用する資源

UNIX版JobCenterのプロセスが使用する資源について説明します。

ここでは、Red Hat Enterprise Linux 7 を例として記述しています。カーネルパラメータやシェル制限は、OSやカーネルのバージョン毎に異なるため、適宜読み替えて設定してください。

本文中に出てくるカーネルパラメータやシェル制限の意味は次のとおりです。

表23.1 本文中に出てくるカーネルパラメータやシェル制限の意味

パラメータ名	パラメータの意味	備考
kernel.threads-max	システム全体の最大プロセス数	
fs.file-max	システム全体の最大ファイル数	
nproc	ユーザ毎に指定できる最大プロセス数	シェル制限
nofile	ユーザ毎に指定できる1プロセスあたりのオープン可能な最大ファイル数	シェル制限
data	ユーザ毎に指定できるプロセスを実行するためのデータセグメントの最大サイズ	シェル制限
nflocks	システム全体でロック可能な最大ファイル数	HP-UXの場合に設定可能

UNIX版JobCenterに関するカーネルパラメータやシェル制限の計算式は次のようになります。

この値はJobCenterが単独で動作する場合に最低限必要な値であり、システム全体の設定値としては、ジョブとして実行するアプリケーションが使用する資源についてもあわせて考慮する必要があります。

表23.2 カーネルパラメータやシェル制限と計算式一覧

パラメータ名	計算式 （W：同時実行ジョブリクエスト数 X：同時接続CL/Win数 Y：同時実行WebAPI数）
kernel.threads-max およびnproc	(4 × W) + 2 + 2 + X + 1 + 2 + 1 + (6 + Y)
fs.file-max	(21 × W) + 40 + 37 + (20 × X) + 16 + 48 + 19 + (20 + 10 × Y)
nofile	各プロセスのnofileの計算で最大となるものを基準としてください。
nflocks	6 × W

UNIX版JobCenterのプロセス構成は大きく分けて次の8つがあります。

プロセス名	処理内容
nqsdaemon	リクエストを実行
jnwengine	ジョブネットワーク制御実行
sclaunchd	スケジュール実行
comagent	他のマシンとの通信用
jnwlauncher	起動トリガの各デーモンの管理
jl_submit	起動トリガのジョブネットワーク投入の管理
jl_logmonitor	起動トリガのテキストログ監視実行
jcwebserver	WebAPI通信用モジュール

23.1.1. nqsdaemon（リクエスト実行）

ジョブリクエストは単位ジョブ実行時にキューに投入され、使用されます。例えば単位ジョブ内でsleepをcshで実行した場合、生成されるプロセスは次の4つです。

プロセス名	処理内容
nqsdaemon	リクエスト処理用プロセス
shell	loginシェル
shell	ジョブスクリプト実行用シェル
sleep	ユーザコマンドプロセス

<使用資源>

パラメータ	1リクエスト当たり使用数
nofile	21
kernel.threads-max およびnproc	4
fs.file-max	21

23.1.2. jnwengine（ジョブネットワーク制御実行）

単位ジョブの実行でジョブリクエストをNQSに投入する際にqsubコマンドを発行します。qsubコマンドは逐次実行されるため、使用する資源はジョブ数の影響を受けません。

nofileは資源を多く消費するqsubコマンドが使用するファイル数になります。

kernel.threads-max、nprocはjnwengineがopenしているファイル数とqsubコマンドが使用するファイル数の合計になります。

<使用資源>

パラメータ	1リクエスト当たり使用数	備　考
nofile	24	qsubコマンド使用分
kernel.threads-max およびnproc	2	jnwengine使用分 = 1、qsubコマンド使用分 = 1
fs.file-max	40	jnwengine使用分 = 16、qsubコマンド使用分 = 24

このほか、jnwengineプロセスは未アーカイブ状態のトラッカについてプロセス内のメモリにデータを展開して保持するため、HP-UXでは32bit環境の場合、大量のトラッカを保持する状況でOSのメモリ空間割当て制限によりプロセスメモリの上限値2GBを超えることができず、正常動作できなくなる場合があります。(IPF版等の64bit環境では2GB制限はありませんが、プロセスメモリの利用方法については32bit環境と同様となります)

従ってアーカイブファイルの保存期間(個数)や保存間隔をチューニングして、未アーカイブ状態のトラッカが大量に残らないようにご注意ください。

23.1.3. sclaunchd（スケジュール実行）

スケジュール起動でジョブネットワーク投入時に子プロセスを起動します。ジョブネットワークの投入処理は逐次実行されるため、使用する資源はジョブネットワークの数には影響を受けません。

nofileは資源を多く消費する子プロセスの値になります。

fs.file-maxはsclaunchdがopenしているファイル数と子プロセス(jnwman)が使用するファイル数の合計になります。

<使用資源>

パラメータ	1JNW投入処理当たり使用数	備　考
nofile	28	jnwmanコマンド使用分
kernel.threads-max およびnproc	2	sclaunchd使用分 = 1、jnwmanコマンド使用分 = 1
fs.file-max	37	sclaunchd使用分 = 13、jnwmanコマンド使用分 = 24

23.1.4. comagent（通信用モジュール）

JobCenter CL/Winからジョブの実行状態の収集や各種設定を行うときに使用されます（ジョブの転送、実行には使用されません）。このため、資源はJobCenter CL/Winの同時起動数に依存します。

通信を受信するごとに子プロセス(comagent、他)を起動しますが、処理が終わると終了します。

次の資源は、通信部分のみ（通信を受信し、その先に呼ばれる処理は除く）です。

<使用資源>

パラメータ	1回の通信処理当たり使用数
nofile	20
kernel.threads-max およびnproc	1
fs.file-max	20

23.1.5. jnwlauncher（起動トリガの各デーモンの管理）

起動トリガの各デーモンの起動・監視・停止に使用されます。使用する資源は起動トリガ定義の数には影響を受けません。

<使用資源>

パラメータ	使用数
nofile	16
kernel.threads-max およびnproc	1
fs.file-max	16

23.1.6. jl_submit（起動トリガのジョブネットワーク投入の管理）

起動トリガの各監視によるジョブネットワーク投入時に子プロセスを起動します。ジョブネットワークの投入処理は逐次実行されるため、使用する資源はジョブネットワーク数には影響を受けません。

nofileは資源を多く消費する子プロセスの値になります。

fs.file-maxはjl_submitがopenしているファイル数と子プロセス(jnwman)が使用するファイル数の合計になります。

<使用資源>

パラメータ	1JNW投入処理当たり使用数	備考
nofile	28	jnwmanコマンド使用分
kernel.threads-max およびnproc	2	jl_submit使用分 = 1、jnwmanコマンド使用分 = 1
fs.file-max	48	jl_submit使用分 = 24、jnwmanコマンド使用分 = 24

23.1.7. jl_logmonitor（起動トリガのテキストログ監視実行）

起動トリガのテキストログ監視時に使用されます。テキストログ監視は逐次実行されるため、使用する資源はテキストログ監視の数には影響を受けません。

<使用資源>

パラメータ	使用数
nofile	19
kernel.threads-max およびnproc	1
fs.file-max	19

23.1.8. jcwebserver（WebAPI通信用モジュール）

ブラウザからJobCenter　MG/SVのWebAPIの実行を行うときに使用されます。このため、資源はJobCenter MG/SVのWebAPIの同時実行数に依存します。

JobCenter MG/SVのWebAPIのリクエストを受信するごとにスレッドを起動してサブプロセス(jnwman等)を起動しますが、処理が終わると終了します。

次の資源は、通信部分のみ（リクエストを受信してサブプロセス起動後の処理は除く）です。

<使用資源>

パラメータ	1回の通信処理当たり使用数	
nofile	20 + (10 × 同時実行WebAPI数)	jcwebserverプロセス使用分
kernel.threads-max およびnproc	6 + (1 × 同時実行WebAPI数)	jcwebserverのメイン処理使用分 = 6 jcwebserverの通信処理使用分 = 1
fs.file-max	20 + (10 × 同時実行WebAPI数)	jcwebserverのメイン処理使用分 = 20 jcwebserverの通信処理使用分 = 10

23.1.9. その他

23.1.9.1. リクエスト実行時のファイルロック

リクエスト実行時に一部のファイルをロックします。ジョブの同時実行数に依存します。HP-UXの場合に影響があります。

<使用資源>

パラメータ	1ジョブリクエスト当たり使用数
nflocks	6

23.1.9.2. 64bit対応したプロセス

Linux版JobCenterにおいて以下のプロセスが32bitから64bitに変更されました。

■jnwengine

■jcdbs

■jnwman

23.2. メモリ使用量概算算出方法（UNIX版）

UNIX版JobCenterのメモリの算定方法について記述します。Windowsを除く他のプラットフォーム（Linuxなど）についても同じ値になります。

運用中にメモリが大きく増減するのは以下の2プロセスです。

■jnwengine

■jcdbs

以降ではこれら2プロセスのメモリ使用量の見積り方について説明します。

23.2.1. jnwengineが必要とするメモリ使用量

jnwengineはジョブネットワークフローの実行制御を行っており、実行時には対象ジョブネットワークの情報をメモリ上に展開します。

1つのジョブネットワークが1回動作するために必要とするメモリの概算使用量は、「23.2.1.1 メモリ使用のイメージ」、「23.2.1.2 概算の算出方法」で説明するメモリ使用量の総和です。誤差と安全率を考慮して、概算使用量の2倍のメモリ空間が確保できる状態であれば円滑に動作できます。

不足している場合は、メモリ不足に陥りエラーが発生することがありますので、必要に応じてカーネルパラメータやシェル制限（たとえば、Linuxの場合はdata）の変更もしくはメモリの増設をするようにしてください。

23.2.1.1. メモリ使用のイメージ

ジョブネットワーク（JNW）メモリ使用域	ヘッダ部：約40Kbyte 情報部（JNW名/パラメータ/コメント/投入キュー名/環境変数/ステータス）：約4Kbyte 環境変数=2Kbyteと仮定しています。
サブジョブネットワーク(SubJNW）1 メモリ使用域 SubJNW2 メモリ使用域 ・ ・ ・ SubJNWj メモリ使用域	ヘッダ部：約0.6Kbyte 情報部（SubJNW名）：約0.25Kbyte SubJNW名は ［ROOTJNW:SubJNW1:・・・:JNWNAME］ の形式です。
ジョブ部品1 メモリ使用域 ジョブ部品2 メモリ使用域 ・ ・ ・ ジョブ部品k メモリ使用域	ヘッダ部：約3.5Kbyte 情報部（ジョブ名/パラメータ/スクリプト/ジョブ名@所属JNW名）：約2Kbyte 所属JNW名は ［ROOTJNW:SubJNW1:・・・:JNWNAME］ の形式です。
分岐ノード1 メモリ使用域 分岐ノード2 メモリ使用域 ・ ・ ・ 分岐ノードm メモリ使用域	ヘッダ部：約0.1Kbyte 情報部：約0.125Kbyte 条件分岐/カレンダ分岐/並列処理において分岐したそれぞれの流れを1として加算します。

ジョブネットワーク待ち合わせ or ジョブ待ち合わせ or 時間待ち合わせ1 メモリ使用域 ジョブネットワーク待ち合わせ or ジョブ待ち合わせ or 時間待ち合わせ2 メモリ使用域 ・ ・ ・ ジョブネットワーク待ち合わせ or ジョブ待ち合わせ or 時間待ち合わせo メモリ使用域	ヘッダ部：約0.7Kbyte 情報部：約0.1Kbyte
ファイル待ち合わせ1 メモリ使用域 ファイル待ち合わせ2 メモリ使用域 ・ ・ ・ ファイル待ち合わせp メモリ使用域	ヘッダ部：約1.4Kbyte 情報部（パスの長さ）：約0.25Kbyte パスの長さ=0.125Kbyteとして仮定しています。
イベント送受信1 メモリ使用域 イベント送受信2 メモリ使用域 ・ ・ ・ イベント送受信q メモリ使用域	ヘッダ部：約1.5Kbyte 情報部：約0.5Kbyte

23.2.1.2. 概算の算出方法

「23.2.1.1 メモリ使用のイメージ」から、概算の算出は次の数式の通りになります。なお数式は「23.2.1.1 メモリ使用のイメージ」の条件のまま作成してあります。

■ジョブネットワーク（JNW）ブロックのメモリ使用量

約44Kbyte　･･･①

■サブジョブネットワーク（SubJNW）オブジェクトブロックのメモリ使用量の総和

SubJNWオブジェクト数(j) ×（約0.85Kbyte）　･･･②

■ジョブ部品オブジェクトブロックのメモリ使用量の総和

ジョブ部品オブジェクト(k) ×（約5.5Kbyte）　･･･③

■分岐ノードブロックのメモリ使用量の総和

分岐ノード数(m) ×（約0.225Kbyte）　･･･④

■待ち合わせ部品ブロックのメモリ使用量の総和

- ジョブネットワーク待ち合わせ/ジョブ待ち合わせ/時間待ち合わせ

  待ち合わせオブジェクト数(o) ×（約0.8Kbyte）　･･･⑤

- ファイル待ち合わせ

  待ち合わせオブジェクト数(p) ×（約1.65Kbyte）　･･･⑥

■イベント送受信部品ブロックのメモリ使用量の総和

イベント送信または受信オブジェクト数(q) × (約2Kbyte)　・・・⑦

従って、1つのジョブネットワークが動作するために必要とするメモリ使用量の概算は次の通りです。

1つのジョブネットワークが円滑に動作するメモリ使用量 = 2 × (①+②+③+④+⑤+⑥+⑦)

23.2.2. jcdbsが必要とするメモリ使用量

jcdbsは、JobCenterの様々なデータをメモリ上で管理するデータベースとしての役割を担います。

以下に、メモリ上に格納されるデータとその使用量をまとめます。

データ名	メモリ使用量
システム環境設定	0.75Kbyte
トラッカ表示色	0.75Kbyte
パーミッション設定	0.5Kbyte
カスタムジョブ定義 (1定義あたり)	0.5Kbyte
稼働日カレンダ定義 (1定義あたり)	0.5Kbyte
ユーザ環境設定	1Kbyte
環境変数データ	(対象ユーザがもつ環境変数のサイズ)
デフォルトパラメータ	8KByte
JNWグループ	ヘッダ情報: 1Kbyte 1ノードごとに0.1Kbyte
JNW定義 (1JNWあたり)	JNW情報: 1.4Kbyte 部品情報: 1部品ごとに0.1Kbyte
スケジュールグループ	ヘッダ情報: 1Kbyte 1ノードごとに0.1Kbyte
スケジュール定義 (1スケジュールあたり)	1.2Kbyte
起動トリガ(テキストログ監視) (1定義あたり)	3.1Kbyte
監視対象テキストログ (1定義あたり)	1.9Kbyte
実行中のトラッカデータ (1トラッカあたり)	JNW情報: 3.6Kbyte 部品情報: 1部品ごとに0.4Kbyte

上記データのうち、JNW定義とトラッカデータのメモリ使用量が構築・運用において大きく変わりますので、予め見積りを行ってください。

23.3. DISK使用容量の概算算出方法（UNIX版）

UNIX版JobCenterのDISK容量の算定方法について記述しています。

Windowsを除く他のプラットフォーム（Linuxなど）においても、通常は同じ値になります。(OS表示上の容量はファイルシステムのセクタサイズにより上下する可能性があります)

■ジョブネットワークを自マシンに設定・定義して自マシンに投入して実行する場合は、以下の「23.3.1 ジョブネットワークが必要とするDISK使用容量」～「23.3.3 ジョブリクエストのNQS実行時のDISK使用容量概算算出方法」のDISK使用容量の合計で見積もってください。

■ジョブネットワーク定義が他のマシンにあり、そこから自マシンにジョブリクエストをリモート転送して実行する場合は、以下の「23.3.3 ジョブリクエストのNQS実行時のDISK使用容量概算算出方法」のDISK使用容量で見積もってください。

23.3.1. ジョブネットワークが必要とするDISK使用容量

1つのジョブネットワークがn回動作してn個のトラッカを生成するために必要とするDISK使用容量の概算の算出方法です。

1つのジョブネットワークがn回動作するために必要とするDISK使用容量の概算は「23.3.1.1 DISK使用のイメージ(1トラッカ当たり)」、「23.3.1.2 概算の算出方法」で説明するDISK使用容量の総和です。このDISK空間が確保できる状態であれば動作できます。

DISKの空き容量が不足している場合、エラーが発生することがありますのでDISK領域の増設をするようにしてください。

なお、それぞれの使用容量については、ジョブネットワーク内に作りこむジョブの処理内容の複雑さや、単位ジョブの実行結果として得られるジョブ出力結果・エラー出力結果のデータ量により大幅に増減します。業務に即してジョブ実行内容をよく検討した上で算出してください。

23.3.1.1. DISK使用のイメージ(1トラッカ当たり)

<table>
<tr><td rowspan="4">ジョブネットワーク
（JNW）フロー関連</td><td>単位ジョブ部品定義1</td><td rowspan="4">約3,000Kbyte

1Kbyteのジョブスクリプトを含む単位ジョブが直列で2,000個定義されていると仮定しています。</td></tr>
<tr><td>単位ジョブ部品定義2</td></tr>
<tr><td>・
・</td></tr>
<tr><td>単位ジョブ部品定義m</td></tr>
<tr><td colspan="2">JNWログ JNW START/ENDログ</td><td>約0.4Kbyte</td></tr>
<tr><td colspan="2">JNWログ 単位ジョブ1 START/ENDログ
JNWログ 単位ジョブ2 START/ENDログ
・
・
・
JNWログ 単位ジョブm START/ENDログ</td><td>約0.4Kbyte
約0.4Kbyte
・
・
・
約0.4Kbyte</td></tr>
<tr><td colspan="2">JNW詳細情報</td><td>約0.4Kbyte</td></tr>
<tr><td colspan="2">単位ジョブ詳細情報1
単位ジョブ詳細情報2
・
・
・
単位ジョブ詳細情報m</td><td>約0.4Kbyte
約0.4Kbyte
・
・
・
約0.4Kbyte</td></tr>
</table>

単位ジョブ出力結果1 単位ジョブ出力結果2 ・ ・ ・ 単位ジョブ出力結果m	シェルスクリプトに依存 シェルスクリプトに依存 ・ ・ ・ シェルスクリプトに依存 （スクリプトの標準出力結果のDISK使用容量は業務内容に依存しますので、お客様ご自身で計算してください）
単位ジョブエラー出力結果1 単位ジョブエラー出力結果2 ・ ・ ・ 単位ジョブエラー出力結果m	シェルスクリプトに依存 シェルスクリプトに依存 ・ ・ ・ シェルスクリプトに依存 （スクリプトの標準エラー出力結果のDISK使用容量は業務内容に依存しますので、お客様ご自身で計算してください）

23.3.1.2. 概算の算出方法

「23.3.1.1 DISK使用のイメージ(1トラッカ当たり)」から1トラッカあたりの概算の算出は次の数式の通りになります。なお数式は「23.3.1.1 DISK使用のイメージ(1トラッカ当たり)」の条件のまま作成してあります。(トラッカ生成の元になるジョブネットワークの定義・設定データを格納するための容量も別途必要になりますのでご注意ください)

■ジョブネットワーク（JNW）フロー関連のDISK使用容量(定義・設定データの容量に同じ)

約3,000Kbyte ･･･①

■ジョブネットワークログのDISK使用容量の総和

約0.4Kbyte ＋ 単位ジョブオブジェクト数(m) × (約0.4Kbyte)　･･･②

■ジョブネットワーク詳細情報のDISK使用容量の総和

約0.4Kbyte ＋ 単位ジョブオブジェクト数(m) × (約0.4Kbyte)　･･･③

■単位ジョブ出力結果のDISK使用容量の総和

単位ジョブオブジェクト数(m) × (推定の平均出力結果サイズ)　･･･④

■単位ジョブエラー出力結果のDISK使用容量の総和

単位ジョブオブジェクト数(m) × (推定の平均エラー出力結果サイズ)　･･･⑤

上記に加えて、上記のトラッカ生成の元になるジョブネットワーク定義を保存する領域として①と同じ容量が別途必要になります。

従って、1つのジョブネットワークがn回動作するために必要とするDISK使用容量の概算は次の通りです。

1つのジョブネットワークがn回動作するために必要とするDISK使用容量 ＝ ① ＋ n × (①＋②＋③＋④＋⑤)

1つのジョブネットワークがn回動作してn個のトラッカが生成されると、JobCenterワーキング領域にDISK使用容量(n × (①+②+③+④+⑤))が確保され、それぞれのトラッカの実行が終了してアーカイブ待ち時間の経過後にアーカイブされると解放されます。

23.3.2. トラッカアーカイブのDISK使用容量概算算出方法

「23.3.1.2 概算の算出方法」に従い、トラッカアーカイブの保存期間内に投入されるそれぞれのジョブネットワークのDISK使用容量の概算を算出し、それぞれのジョブネットワークがトラッカアーカイブの保存期間中に投入される累積回数を乗算し、総和することでトラッカアーカイブのDISK使用容量の概算を算出できます。

なお、単位ジョブ出力結果・単位ジョブエラー出力結果の文字コードはUTF-8のため、スクリプトの出力する文字コードと異なる場合はDISK使用容量に影響を与える可能性があります。

■トラッカアーカイブの保存期間中に投入されたジョブネットワーク(JNW) のDISK使用容量の総和

JNWの数(i) × [(JNWのDISK使用容量) × (トラッカアーカイブ保存期間内のJNW投入回数)]

23.3.3. ジョブリクエストのNQS実行時のDISK使用容量概算算出方法

ジョブネットワークを実行すると、フローに含まれる単位ジョブが順次NQSのキューに投入されて実行されます。そのNQSスプール領域のDISK使用容量の概算を算出し、同時に実行中になるジョブリクエストの個数を乗算し、総和することでNQSのDISK使用容量の概算を算出できます。

■ジョブリクエストのNQS管理領域のDISK使用容量

同時実行ジョブリクエスト数(m) × (約10Kbyte)　・・・①

■ジョブスクリプトのスプールファイルのDISK使用容量の総和

同時実行ジョブリクエスト数(m) × (ジョブスクリプトの推定平均サイズ)　・・・②

■ジョブリクエストID管理領域のDISK使用容量

最大約12Mbyte　・・・③

■ジョブリクエスト中のコマンド標準出力結果スプールファイルのDISK使用容量の総和

同時実行ジョブリクエスト数(m) × (標準出力結果の推定平均サイズ)　・・・④

■ジョブリクエスト中のコマンド標準エラー出力結果スプールファイルのDISK使用容量の総和

同時実行ジョブリクエスト数(m) × (標準エラー出力結果の推定平均サイズ)　・・・⑤

従って、ジョブリクエストがNQSのキューで同時に複数動作するために必要とするスプール領域のDISK使用容量の概算は次の通りです。

NQS領域のDISK使用容量 = (①+②+③+④+⑤)

なお複数のリモートマシン(サイト)からジョブがリモート投入されるような状況で、かつジョブリクエストを短時間に大量に連続的に実行する負荷の高い状況では、上記の③はより増加する可能性がありますので、③はあくまで目安として考えてください。

単位ジョブスクリプトのサイズや、単位ジョブスクリプトに記述したコマンドが出力する標準出力や標準エラー出力をNQSで一時保持するためのスプールファイルのサイズは、業務内容により大幅に増減します。業務内容をよく検討の上で見積もってください。

23.3.4. jcdbsのDISK使用容量概算算出方法

jcdbsが必要とするDISK使用量の概算はjcdbsメモリ使用量の約3倍になります。

jcdbsメモリ使用量は「23.2.2　jcdbsが必要とするメモリ使用量」から算出できます。ただし、jcdbsメモリ使用量が少量である場合、少なくとも64MBを使用します。

DISKの空き容量が不足している場合、エラーが発生することがありますのでDISK領域の増設をするようにしてください。

23.4. Windows版JobCenterのプロセスが使用する資源

Windows版JobCenterのプロセスが使用する資源について説明します。

本文中に出てくるパラメータの意味は次のとおりです。

表23.3 本文中に出てくるパラメータの意味

パラメータ	パラメータの意味
maxThreads	プロセスが動いている間に同時に開くthreadの最大数
nProcesses_const	同時実行可能な、ジョブ数とともに変わることがないプロセス数
nProcesses_var	同時実行可能な、ジョブ数とともに変わることがあるプロセス数

JobCenterのプロセスのうち、マルチスレッド向けに設計しているのはcomagentとnetdaemonだけなので、comagentとnetdaemonだけに対してmaxThreadsパラメータを計算します。

Windows版JobCenterに関するカーネルパラメータの計算式は次のようになります。

この値はJobCenterが単独で動作する場合に最低限必要な値であり、システム全体の設定値としては、ジョブとして実行するアプリケーションが使用する資源についてもあわせて考慮する必要があります。

表23.4 カーネルパラメータと計算式一覧

パラメータ	計算式 （nProcesses：JobCenter実行時に、トータルのプロセス数 W：同時実行ジョブリクエスト数 X：同時接続CL/Win数 Y：同時実行WebAPI数 Z：同時到着したリモートジョブリクエスト）
maxThreads	comagent(1 × X) + jcwebserver(6 + 1 × Y) + netdaemon(1 × Z)
nProcesses	1 + 4 × W + 1 + 1 + 2 + 1 + 1 + 2 + 1 + 1 × X + 1 + 2 + 1 + 1 + 1 × Y = 4 × W + 1 × X + 15 + 1 × Y

Windows版JobCenterのプロセス構成は大きく分けて次の12個があります。

プロセス名	処理内容
nqsdaemon	リクエスト処理用プロセス
logdaemon	nqsdaemonとnetdaemonのためのログを記述
netdaemon	リモートから転送されてきたジョブリクエストを受信し処理
jnwengine	ジョブネットワーク制御実行
qwb	NQS中のジョブ状態変化を監視し、それをjnwengineに通知
trkrdr	jnwengineのコマンドを待ち合わせ、NQSからジョブ情報を取得
sclaunchd	スケジュール実行
comagent	他のマシンとの通信用
jnwlauncher	起動トリガの各デーモンの管理
jl_submit	起動トリガのジョブネットワーク投入の管理
jl_logmonitor	起動トリガのテキストログ監視実行
jcwebserver	WebAPI通信用モジュール

23.4.1. nqsdaemon（リクエスト実行）

リクエストは、単位ジョブ実行時に使用されます。例として単位ジョブ内でnotepadをcmdで実行した場合を説明します。

生成されるプロセスは次の4つです。

プロセス名	処理内容
nqsdaemon	リクエスト処理用プロセス
shepherd	リクエスト実行用子プロセス
cmd	ジョブスクリプト実行用コマンド
notepad	ユーザコマンドプロセス

<使用資源>

パラメータ	1リクエスト当たり使用数	備　考
nProcesses_const	1	nqsdaemon使用分
nProcesses_var	4	

23.4.2. logdaemon（nqsdaemonとnetdaemonのためのログを記述）

nqsdaemonとnetdaemonのためのログを書くことにだけ使用されます。使用する資源はジョブ数とは関係ありません。

<使用資源>

パラメータ	使用数
nProcesses_const	1
nProcesses_var	0

23.4.3. netdaemon（リモートから転送されるリクエスト受信処理）

使用する資源は、リモートから転送されてきたリクエストの数に依存します。

次の例はリクエストを1回実行する場合の資源使用状況です。

<使用資源>

パラメータ名	1リクエスト当たり使用数	備　考
maxThreads	1	リクエストを1回実行するときに生成されるスレッドのみを考慮
nProcesses_const	1	
nProcesses_var	0	

23.4.4. jnwengine（ジョブネットワーク制御実行）

ジョブの投入でqsubコマンドを発行します。qsubコマンドは逐次実行されるため、使用する資源はジョブ数の影響を受けません。

<使用資源>

パラメータ名	数　量
nProcesses_const	2
nProcesses_var	0

23.4.5. qwb（NQS中のジョブの状態変化監視とjnwengineへの通知）

qwbが使用する資源はジョブ数とは関係ありません。

<使用資源>

パラメータ名	数　量
nProcesses_const	1
nProcesses_var	0

23.4.6. trkrdr（jnwengineのコマンド待ち合わせとジョブ情報取得）

trkrdrが使用する資源はジョブ数とは関係ありません。

<使用資源>

パラメータ名	数　量
nProcesses_const	1
nProcesses_var	0

23.4.7. sclaunchd（スケジュール実行）

スケジュールを起動してジョブネットワークを投入する場合、サブプロセスを起動する必要があります。ジョブネットワークの投入は逐次実行されるため、使用する資源はジョブネットワーク数の影響を受けません。

生成されるプロセスは次の2つです。

プロセス名	処理内容
sclaunchd	スケジュール管理プロセス
jnwman	JNW投入処理を実行

<使用資源>

パラメータ	1JNW投入処理当たり使用数
nProcesses_const	2
nProcesses_var	0

23.4.8. comagent（通信用モジュール）

JobCenter CL/Winからジョブの実行状態の収集や各種設定を行うときに使用されます（ジョブの転送、実行には使用されません）。このため、使用する資源はJobCenter CL/Winの同時接続数に依存します。

通信を受信するごとにjnwman，nmapmgrまたは他のサブプロセスを起動しなければなりません。これらのサブプロセスが単独で動いている場合に使用されるHandleの平均数は8と推定されています。

次の例は通信を1回実行する場合の資源使用状況です。

<使用資源>

パラメータ	1回の通信処理当たり使用数	備　考
maxThreads	1	リクエストを1回実行するときに生成されるスレッドのみを考慮
nProcesses_const	1	
nProcesses_var	1	サブプロセス使用分

23.4.9. jnwlauncher（起動トリガの各デーモンの管理）

起動トリガの各デーモンの起動・監視・停止に使用されます。使用する資源は起動トリガ定義の数には影響を受けません。

<使用資源>

パラメータ名	数　量
nProcesses_const	1
nProcesses_var	0

23.4.10. jl_submit（起動トリガのジョブネットワーク投入の管理）

起動トリガの各監視によるジョブネットワーク投入時に子プロセスを起動します。ジョブネットワークの投入処理は逐次実行されるため、使用する資源はジョブネットワーク数には影響を受けません。

生成されるプロセスは次の2つです。

プロセス名	処理内容
jl_submit	起動トリガのジョブネットワーク投入の管理プロセス
jnwman	JNW投入処理を実行

<使用資源>

パラメータ	1JNW投入処理当たり使用数
nProcesses_const	2
nProcesses_var	0

23.4.11. jl_logmonitor（起動トリガのテキストログ監視実行）

起動トリガのテキストログ監視時に使用されます。テキストログ監視は逐次実行されるため、使用する資源はテキストログ監視の数には影響を受けません。

<使用資源>

パラメータ名	数　量
nProcesses_const	1
nProcesses_var	0

23.4.12. jcwebserver（WebAPI通信用モジュール）

ブラウザからJobCenter　MG/SVのWebAPIの実行を行うときに使用されます。このため、資源はJobCenter MG/SVのWebAPIの同時実行数に依存します。

JobCenter MG/SVのWebAPIのリクエストを受信するごとにスレッドを起動してサブプロセス(jnwman等)を起動しますが、処理が終わると終了します。

次の例は通信を1回実行する場合の資源使用状況です。

<使用資源>

パラメータ	1回の通信処理当たり使用数	備　考
maxThreads	6 ＋ 1	jcwebserverの基本スレッド ＝ 6、リクエストを1回実行するときに生成されるスレッド ＝ 1

nProcesses_const	1	
nProcesses_var	1	サブプロセス使用分

23.5. メモリ使用量概算算出方法（Windows版）

Windows版JobCenterのメモリの算定方法について記述します。

JobCenterで使用するデスクトップヒープについて

Windows OSのデスクトップヒープのサイズがJobCenterの動作に影響する場合があります。デスクトップヒープの調整方法については Microsoft の KB184802 (User32.dll または Kernel32.dll の初期化に失敗する) 等を参照してください。

■OSのデスクトップヒープが枯渇して新しいデスクトップが作成できなくなると、JobCenterサービス（ローカルサイト)が起動できなくなります。その場合は非対話型のヒープサイズを小さくチューニングすることで、作成できるデスクトップの数を増やします。

■JobCenterサービスが作成したデスクトップが持つヒープの残りが少なくなり、新しいオブジェクトが作成できなくなると、ジョブリクエストからジョブを実行するためのプロセスを起動できなくなります。その場合は非対話型のヒープサイズを大きくチューニングすることで、JobCenterサービスのデスクトップに作成できるオブジェクトの数を増やします。

なお、JobCenterの動作に必要なデスクトップヒープは以下のとおりです。

■非対話型サービスのウィンドウステーションのデスクトップヒープの空きが5KB以上あること

■非対話型のデスクトップが新規に一つ以上(サイズは256KB以上)作成できること

■対話型のデスクトップヒープサイズは最低256KBの空きがあること(単位ジョブから実行するコマンドやアプリケーションによっては要調整)

運用中にメモリが大きく増減するのは以下の2プロセスです。

■jnwengine

■jcdbs

以降ではこれら2プロセスのメモリ使用量の見積り方について説明します。

23.5.1. jnwengineが必要とするメモリ使用量

jnwengineはジョブネットワークフローの実行制御を行っており、実行時には対象ジョブネットワークの情報をメモリ上に展開します。

1つのジョブネットワークが1回動作するために必要とするメモリの概算使用量は、「23.5.1.1 メモリ使用のイメージ」、「23.5.1.2 概算の算出方法」で説明するメモリ使用量の総和です。誤差と安全率を考慮して、概算使用量の2倍のメモリ空間が確保できる状態であれば円滑に動作できます。

不足している場合は、メモリ不足に陥りエラーが発生することがありますので、必要に応じてメモリの増設をするようにしてください。

23.5.1.1. メモリ使用のイメージ

ジョブネットワーク（JNW）メモリ使用域	ヘッダ部：約86Kbyte 情報部（JNW名/パラメータ/コメント/投入キュー名/環境変数/ステータス）：約4Kbyte 環境変数=2Kbyteと仮定しています。
サブジョブネットワーク(SubJNW）1 メモリ使用域 SubJNW2 メモリ使用域	ヘッダ部：約0.2Kbyte 情報部（SubJNW名）：約0.1Kbyte

・ ・ ・ SubJNWj メモリ使用域	SubJNW名は ［ROOTJNW:SubJNW1:・・・:JNWNAME］ の形式です。
ジョブ部品1 メモリ使用域 ジョブ部品2 メモリ使用域 ・ ・ ・ ジョブ部品k メモリ使用域	ヘッダ部：約6.5Kbyte 情報部（ジョブ名/パラメータ/スクリプト/ジョブ名@所属JNW名）：約2Kbyte 所属JNW名は ［ROOTJNW:SubJNW1:・・・:JNWNAME］ の形式です。
分岐ノード1 メモリ使用域 分岐ノード2 メモリ使用域 ・ ・ ・ 分岐ノードm メモリ使用域	ヘッダ部：約0.1Kbyte 情報部：約0.125Kbyte 条件分岐/カレンダ分岐/並列処理において分岐したそれぞれの流れを1として加算します。
ジョブネットワーク待ち合わせ or ジョブ待ち合わせ or 時間待ち合わせ1 メモリ使用域 ジョブネットワーク待ち合わせ or ジョブ待ち合わせ or 時間待ち合わせ2 メモリ使用域 ・ ・ ・ ジョブネットワーク待ち合わせ or ジョブ待ち合わせ or 時間待ち合わせo メモリ使用域	ヘッダ部：約0.8Kbyte 情報部：約0.1Kbyte
ファイル待ち合わせ1 メモリ使用域 ファイル待ち合わせ2 メモリ使用域 ・ ・ ・ ファイル待ち合わせp メモリ使用域	ヘッダ部：約2.2Kbyte 情報部（パスの長さ）：約0.25Kbyte パスの長さ=0.125Kbyteとして仮定しています。
イベント送受信1 メモリ使用域 イベント送受信2 メモリ使用域 ・ ・ ・ イベント送受信q メモリ使用域	ヘッダ部：約2.5Kbyte 情報部：約0.5Kbyte

23.5.1.2. 概算の算出方法

「23.5.1.1 メモリ使用のイメージ」から、概算の算出は次の数式の通りになります。なお数式は「23.5.1.1 メモリ使用のイメージ」の条件のまま作成してあります。

■ジョブネットワーク（JNW）ブロックのメモリ使用量

約90Kbyte　･･･①

■サブジョブネットワーク（SubJNW）オブジェクトブロックのメモリ使用量の総和

SubJNWオブジェクト数(j) × （約0.3Kbyte）　･･･②

■ジョブ部品オブジェクトブロックのメモリ使用量の総和

ジョブ部品オブジェクト数(k) × （約8.5Kbyte）　･･･③

■分岐ノードブロックのメモリ使用量の総和

分岐ノード数(m) × （約0.225Kbyte）　･･･④

■待ち合わせ部品ブロックのメモリ使用量の総和

- ジョブネットワーク待ち合わせ/ジョブ待ち合わせ/時間待ち合わせ

  待ち合わせオブジェクト数(o) × （約0.9Kbyte）　･･･⑤

- ファイル待ち合わせ

  待ち合わせオブジェクト数(p) × （約2.45Kbyte）　･･･⑥

■イベント送受信部品ブロックのメモリ使用量の総和

イベント送信または受信オブジェクト数(q) × （約3Kbyte）　･･･⑦

従って、1つのジョブネットワークが動作するために必要とするメモリ使用量の概算は次の通りです。

1つのジョブネットワークが円滑に動作するメモリ使用量 = 2 × （①+②+③+④+⑤+⑥+⑦）

23.5.2. jcdbsが必要とするメモリ使用量

jcdbsは、JobCenterの様々なデータをメモリ上で管理するデータベースとしての役割を担います。

以下に、メモリ上に格納されるデータとその使用量をまとめます。

データ名	メモリ使用量
システム環境設定	0.8Kbyte
トラッカ表示色	0.8Kbyte
パーミッション設定	0.6Kbyte
カスタムジョブ定義 (1定義あたり)	1Kbyte
稼働日カレンダ定義 (1定義あたり)	1.8Kbyte
ユーザ環境設定	1.5Kbyte
環境変数データ	(対象ユーザがもつ環境変数のサイズ)
デフォルトパラメータ	8KByte
JNWグループ	ヘッダ情報: 1Kbyte 1ノードごとに0.1Kbyte
JNW定義 (1JNWあたり)	JNW情報: 2.5Kbyte 部品情報: 1部品ごとに0.1Kbyte

スケジュールグループ	ヘッダ情報: 1Kbyte 1ノードごとに0.1Kbyte
スケジュール定義 (1スケジュールあたり)	1.8Kbyte
起動トリガ(テキストログ監視) (1定義あたり)	2.9Kbyte
監視対象テキストログ (1定義あたり)	2.1Kbyte
実行中のトラッカデータ (1トラッカあたり)	JNW情報: 2.5Kbyte 部品情報: 1部品ごとに0.4Kbyte

上記データのうち、JNW定義とトラッカデータのメモリ使用量が構築・運用において大きく変わりますので、予め見積りを行ってください。

23.6. DISK使用容量の概算算出方法（Windows版）

Windows版JobCenterのDISK容量の算定方法について記述しています。(表示上の容量はファイルシステムのセクタサイズにより上下する可能性があります)

■ジョブネットワークを自マシンに設定・定義して自マシンに投入して実行する場合は、以下の「23.6.1 ジョブネットワークが必要とするDISK使用容量」～「23.6.3 ジョブリクエストのNQS実行時のDISK使用容量概算算出方法」のDISK使用容量の合計で見積もってください。

■ジョブネットワーク定義が他のマシンにあり、そこから自マシンにジョブリクエストをリモート転送して実行する場合は、以下の「23.6.3 ジョブリクエストのNQS実行時のDISK使用容量概算算出方法」のDISK使用容量で見積もってください。

Windowsの場合、HDDのクラスタサイズに依存して実使用量より大きいサイズのディスク容量を必要とする場合があります。多数の単位ジョブを含むジョブネットワークを作成する場合はその点に注意してください。

23.6.1. ジョブネットワークが必要とするDISK使用容量

1つのジョブネットワークがｎ回動作してn個のトラッカを生成するために必要とするDISK使用容量の概算の算出方法です。

1つのジョブネットワークがｎ回動作するために必要とするDISK使用容量の概算は「23.6.1.1 DISK使用のイメージ(1トラッカ当たり)」、「23.6.1.2 概算の算出方法」で説明するDISK使用容量の総和です。このDISK空間が確保できる状態であれば動作できます。

DISKの空き容量が不足している場合、エラーが発生することがありますのでDISK領域の増設をするようにしてください。

なお、それぞれの使用容量については、ジョブネットワーク内に作りこむジョブの処理内容の複雑さや、単位ジョブの実行結果として得られるジョブ出力結果・エラー出力結果のデータ量により大幅に増減します。業務に即してジョブ実行内容をよく検討した上で算出してください。

23.6.1.1. DISK使用のイメージ(1トラッカ当たり)

ジョブネットワーク （JNW）フロー関連	単位ジョブ部品定義1	約3,000Kbyte 1Kbyteのジョブスクリプトを含む単位ジョブが直列で2,000個定義されていると仮定しています。
	単位ジョブ部品定義2	
	・ ・	
	単位ジョブ部品定義m	
JNWログ JNW START/ENDログ		約0.4Kbyte
JNWログ 単位ジョブ1 START/ENDログ JNWログ 単位ジョブ2 START/ENDログ ・ ・ ・ JNWログ 単位ジョブm START/ENDログ		約0.4Kbyte 約0.4Kbyte ・ ・ ・ 約0.4Kbyte
JNW詳細情報		約0.4Kbyte
単位ジョブ詳細情報1 単位ジョブ詳細情報2 ・		約0.4Kbyte 約0.4Kbyte ・

・ ・ 単位ジョブ詳細情報m	・ ・ 約0.4Kbyte
単位ジョブ出力結果1 単位ジョブ出力結果2 ・ ・ ・ 単位ジョブ出力結果m	バッチスクリプトに依存 バッチスクリプトに依存 ・ ・ ・ バッチスクリプトに依存 （スクリプトの標準出力結果のDISK使用容量は業務内容に依存しますので、お客様ご自身で計算してください）
単位ジョブエラー出力結果1 単位ジョブエラー出力結果2 ・ ・ ・ 単位ジョブエラー出力結果m	バッチスクリプトに依存 バッチスクリプトに依存 ・ ・ ・ バッチスクリプトに依存 （スクリプトの標準エラー出力結果のDISK使用容量は業務内容に依存しますので、お客様ご自身で計算してください）

23.6.1.2. 概算の算出方法

「23.6.1.1　DISK使用のイメージ(1トラッカ当たり)」から、概算の算出は次の数式の通りになります。なお数式は「23.6.1.1　DISK使用のイメージ(1トラッカ当たり)」の条件のまま作成してあります。(トラッカ生成の元になるジョブネットワークの定義・設定データを格納するための容量も別途必要になりますのでご注意ください)

■ジョブネットワーク（JNW）フロー関連のDISK使用容量(定義・設定データの容量に同じ)

約3,000Kbyte ･･･①

■ジョブネットワーク（JNW）ログのDISK使用容量の総和

約0.4Kbyte ＋ 単位ジョブオブジェクト数(m) ×（約0.4Kbyte）　･･･②

■ジョブネットワーク（JNW）詳細情報のDISK使用容量の総和

約0.4Kbyte ＋ 単位ジョブオブジェクト数(m) ×（約0.4Kbyte）　･･･③

■単位ジョブ出力結果のDISK使用容量の総和

単位ジョブオブジェクト数(m) ×（推定の平均出力結果サイズ）　･･･④

■単位ジョブエラー出力結果のDISK使用容量の総和

単位ジョブオブジェクト数(m) ×（推定の平均エラー出力結果サイズ）　･･･⑤

上記に加えて、上記のトラッカ生成の元になるジョブネットワーク定義を保存する領域として①と同じ容量が別途必要になります。

従って、1つのジョブネットワークがn回動作するために必要とするDISK使用容量の概算は次の通りです。

1つのジョブネットワークがn回動作するために必要とするDISK使用容量 =
① ＋ n × (①＋②＋③＋④＋⑤)

1つのジョブネットワークがn回動作してn個のトラッカが生成されると、JobCenterワーキング領域にDISK使用容量(n × (①＋②＋③＋④＋⑤))が確保され、それぞれのトラッカの実行が終了してアーカイブ待ち時間の経過後にアーカイブされると解放されます。

23.6.2. トラッカアーカイブのDISK使用容量概算算出方法

「23.6.1.2 概算の算出方法」に従い、トラッカアーカイブの保存期間内に投入されるそれぞれのジョブネットワークのDISK使用容量の概算を算出し、それぞれのジョブネットワークがトラッカアーカイブの保存期間中に投入される累積回数を乗算し、総和することでトラッカアーカイブのDISK使用容量の概算を算出できます。

なお、単位ジョブ出力結果・単位ジョブエラー出力結果の文字コードはUTF-8のため、スクリプトの出力する文字コードと異なる場合はDISK使用容量に影響を与える可能性があります。

■トラッカアーカイブの保存期間に投入されたジョブネットワーク（JNW）のDISK使用容量の総和

JNW数(i) × [(JNWのDISK使用容量) × (トラッカアーカイブ保存期間内のJNW投入回数)]

23.6.3. ジョブリクエストのNQS実行時のDISK使用容量概算算出方法

ジョブネットワークを実行すると、フローに含まれる単位ジョブが順次NQSのキューに投入されて実行されます。そのNQSスプール領域のDISK使用容量の概算を算出し、同時に実行中になるジョブリクエストの個数を乗算し、総和することでNQSのDISK使用容量の概算を算出できます。

■ジョブリクエストのNQS管理領域のDISK使用容量

同時実行ジョブリクエスト数(m) × (約10Kbyte)　･･･①

■ジョブスクリプトのスプールファイルのDISK使用容量の総和

同時実行ジョブリクエスト数(m) × (ジョブスクリプトの推定平均サイズ)　･･･②

■ジョブリクエストID管理領域のDISK使用容量

最大約24Mbyte (MG側)　･･･③

最大約12Mbyte (SV側)　･･･③

■ジョブリクエスト中のマンド標準出力結果スプールファイルのDISK使用容量の総和

同時実行ジョブリクエスト数(m) × (標準出力結果の推定平均サイズ)　･･･④

■ジョブリクエスト中のマンド標準エラー出力結果スプールファイルのDISK使用容量の総和

同時実行ジョブリクエスト数(m) × (標準エラー出力結果の推定平均サイズ)　･･･⑤

従って、ジョブリクエストがNQSのキューで同時に複数動作するために必要とするスプール領域のDISK使用容量の概算は次の通りです。

NQS領域のDISK使用容量 = (①＋②＋③＋④＋⑤)

なお複数のリモートマシン(サイト)からジョブがリモート投入されるような状況で、かつジョブリクエストを短時間に大量に連続的に実行する負荷の高い状況では、上記の③はより増加する可能性がありますので、③はあくまで目安として考えてください。

単位ジョブスクリプトのサイズや、単位ジョブスクリプトに記述したコマンドが出力する標準出力や標準エラー出力をNQSで一時保持するためのスプールファイルのサイズは、業務内容により大幅に増減します。業務内容をよく検討の上で見積もってください。

23.6.4. jcdbsのDISK使用容量概算算出方法

jcdbsが必要とするDISK使用量の概算はjcdbsメモリ使用量の約3倍になります。

jcdbsメモリ使用量は「23.5.2　jcdbsが必要とするメモリ使用量」から算出できます。ただし、jcdbsメモリ使用量が少量である場合、少なくとも64MBを使用します。

DISKの空き容量が不足している場合、エラーが発生することがありますのでDISK領域の増設をするようにしてください。

23.7. 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

アップロード・ダウンロード先のJobCenter MG/SVが使用するメモリの算出方法について説明します。

アップロード・ダウンロード先のJobCenter　MG/SVにおいて、メモリ不足を起因とするエラーが発生した場合、アップロード・ダウンロードする定義を分割して実行するようにしてください。

円滑なアップロード・ダウンロードを行うには、誤差と安全率を考慮して、概算使用量の2倍のメモリ空間が確保できる状態にしてください。

23.7.1. Windows版JobCenter MG/SVのアップロード・ダウンロード時のメモリ使用量の概算算出方法

■アップロード時のメモリ使用量の概算算出方法

アップロード1回に必要とするメモリ使用量の概算は次の通りです。各定義のアップロード時のメモリ使用量については、表23.5「各定義のアップロード時のメモリ使用量」を参照してください。

アップロード時に必要とするメモリ使用量 = [1] + [2] + [3] + [4] + [5] + [6] + [7]

- [1]:ジョブネットワーク数×ジョブネットワークのメモリ使用量
- [2]:部品数×部品のメモリ使用量
- [3]:スケジュール数×スケジュールのメモリ使用量
- [4]:起動トリガ(テキストログ監視)数×起動トリガ(テキストログ監視)のメモリ使用量
- [5]:監視対象テキストログ数×監視対象テキストログのメモリ使用量
- [6]:カレンダー数×カレンダーのメモリ使用量
- [7]:カスタムジョブテンプレート数×カスタムジョブテンプレートのメモリ使用量

表23.5 各定義のアップロード時のメモリ使用量

定義種別名	1定義あたりのメモリ使用量
ジョブネットワーク	20Kbyte
部品(単位ジョブ等)	25Kbyte
スケジュール	20Kbyte
起動トリガ(テキストログ監視)	10Kbyte
監視対象テキストログ	8Kbyte
カレンダー	40Kbyte
カスタムジョブテンプレート	4Kbyte

■ダウンロード時のメモリ使用量の概算算出方法

ダウンロード1回に必要とするメモリ使用量の概算は次の通りです。各定義のダウンロード時のメモリ使用量については、表23.6「各定義のダウンロード時のメモリ使用量」を参照してください。

ダウンロード時に必要とするメモリ使用量 = [1] + [2] + [3] + [4] + [5] + [6] + [7]

- [1]:ジョブネットワーク数×ジョブネットワークのメモリ使用量
- [2]:部品数×部品のメモリ使用量

- [3]:スケジュール数×スケジュールのメモリ使用量
- [4]:起動トリガ(テキストログ監視)数×起動トリガ(テキストログ監視)のメモリ使用量
- [5]:監視対象テキストログ数×監視対象テキストログのメモリ>使用量
- [6]:カレンダー数×カレンダーのメモリ使用量
- [7]:カスタムジョブテンプレート数×カスタムジョブテンプレートのメモリ使用量

表23.6 各定義のダウンロード時のメモリ使用量

定義種別名	1定義あたりのメモリ使用量
ジョブネットワーク	60Kbyte
部品(単位ジョブ等)	38Kbyte
スケジュール	9Kbyte
起動トリガ(テキストログ監視)	10Kbyte
監視対象テキストログ	7Kbyte
カレンダー	4Kbyte
カスタムジョブテンプレート	8Kbyte

23.7.2. UNIX版JobCenter MG/SVのアップロード・ダウンロード時のメモリ使用量の概算算出方法

■アップロード時のメモリ使用量の概算算出方法

アップロード1回に必要とするメモリ使用量の概算は次の通りです。各定義のアップロード時のメモリ使用量については、表23.7「各定義のアップロード時のメモリ使用量」を参照してください。

アップロード時に必要とするメモリ使用量 = [1] + [2] + [3] + [4] + [5] + [6] + [7]

- [1]:ジョブネットワーク数×ジョブネットワークのメモリ使用量
- [2]:部品数×部品のメモリ使用量
- [3]:スケジュール数×スケジュールのメモリ使用量
- [4]:起動トリガ(テキストログ監視)数×起動トリガ(テキストログ監視)のメモリ使用量
- [5]:監視対象テキストログ数×監視対象テキストログのメモリ>使用量
- [6]:カレンダー数×カレンダーのメモリ使用量
- [7]:カスタムジョブテンプレート数×カスタムジョブテンプレートのメモリ使用量

表23.7 各定義のアップロード時のメモリ使用量

定義種別名	1定義あたりのメモリ使用量
ジョブネットワーク	10Kbyte
部品(単位ジョブ等)	10Kbyte
スケジュール	10Kbyte
起動トリガ(テキストログ監視)	7Kbyte
監視対象テキストログ	5Kbyte

カレンダー	30Kbyte
カスタムジョブテンプレート	6Kbyte

■ダウンロード時のメモリ使用量の概算算出方法

ダウンロード1回に必要とするメモリ使用量の概算は次の通りです。各定義のダウンロード時のメモリ使用量については、表23.8「各定義のダウンロード時のメモリ使用量」を参照してください。

ダウンロード時に必要とするメモリ使用量 = [1] + [2] + [3] + [4] + [5] + [6] + [7]

- [1]:ジョブネットワーク数×ジョブネットワークのメモリ使用量
- [2]:部品数×部品のメモリ使用量
- [3]:スケジュール数×スケジュールのメモリ使用量
- [4]:起動トリガ(テキストログ監視)数×起動トリガ(テキストログ監視)のメモリ使用量
- [5]:監視対象テキストログ数×監視対象テキストログのメモリ>使用量
- [6]:カレンダー数×カレンダーのメモリ使用量
- [7]:カスタムジョブテンプレート数×カスタムジョブテンプレートのメモリ使用量

表23.8 各定義のダウンロード時のメモリ使用量

定義種別名	1定義あたりのメモリ使用量
ジョブネットワーク	30Kbyte
部品(単位ジョブ等)	10Kbyte
スケジュール	10Kbyte
起動トリガ(テキストログ監視)	7Kbyte
監視対象テキストログ	5Kbyte
カレンダー	8Kbyte
カスタムジョブテンプレート	8Kbyte

24. OS要塞化

この章ではSELinux設定およびユーザ環境をセキュアにしたサーバ環境におけるJobCenter　MG/SVの運用に関して説明します。対象となるOSは以下となります。

■Red Hat Enterprise Linux 6

■Red Hat Enterprise Linux 7

■Red Hat Enterprise Linux 8

SELinux設定をセキュアにしたサーバ環境とは、/etc/selinux/configファイルにSELINUX=enforcingを設定した環境を指します。

■SELinux設定をセキュアにしたサーバ環境におけるJobCenter　MG/SVの運用に関する留意点を以下に示します。

- JobCenter　MG/SV用に提供されるSELinuxポリシーファイル(jobcenter.pp)は、JobCenterの動作のベースとなるものとして提供しており、JobCenter運用環境において必要に応じてSELinuxポリシーをカスタマイズして頂くことを想定しています。そのため、JobCenter運用環境においてSELinuxに関する修正や設定を適切に行う体制が無い場合は、SELinux設定ファイルのSELINUX(/etc/selinux/configファイルのSELINUX=の設定)にpermissive(あるいはdisabled)を選択してください。

- SELinux設定ファイルのSELINUXTYPE(/etc/selinux/configファイルのSELINUXTYPE=の設定)は、既定値のtargetedのまま変更しないでください。

- JobCenterの動作に関連するOSのログインユーザのSELinuxコンテキストは、既定値のunconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 のまま変更しないでください。

ユーザ環境をセキュアにしたサーバ環境とは、ユーザにおけるアクセス権が制限されている環境（umask設定を既定値から077等に設定変更）を指します。

■ユーザ環境をセキュアにしたサーバ環境におけるJobCenter　MG/SVの運用に関する留意点を以下に示します。

- （JobCenter自体の動作には問題ありませんが、）umask設定が077等に変更された環境では、JobCenterのコマンドを実行してファイル出力する場合、出力(保存)されるファイルの権限が変更後のumask設定の影響を受けます。

25. トラブルシューティング

JobCenterのトラブルシューティングに関する情報を記載しています。

25.1. トラブルシューティングQ&A

Q.1	定義ファイルをコピーしても一覧画面に表示されない。
A.1	定義ファイルを単純にコピーしただけでは使用できません。別のマシン、ユーザから環境を移行する場合は、jdh_download、jdh_uploadコマンドを使用して定義の移行を行ってください。 マシン名やユーザ名の変換が必要な場合は移行時に変換を行う必要があります。詳細は 14章「環境移行」 を参照して下さい。

Q.2	JobCenterから起動したジョブのプロセスが強制終了させられることがある。
A.2	シェルの中でコマンドをバックグランドで起動し、シェルは終了してもプロセスが残っているという状況では、バックグランド起動したプロセスが強制終了することがあります。 JobCenterは、ジョブのプロセスをプロセスグループという単位で管理しています。ジョブの終了を検出する（＝シェルの終了）と、ジョブのプロセスグループに対してSIGKILLを送信して残っているプロセスを強制終了させます。

Q.3	コマンドプロンプトから実行すると正常終了するが、JobCenterのジョブとして実行するとエラーになる。
A.3	このような現象には次のような原因が考えられます。 ■ジョブの実行ユーザが異なります。 ジョブの実行ユーザは通常、ジョブの投入ユーザと一致しますが、ユーザマッピングの機能を使用しているとジョブを投入したユーザと実行するユーザが異なることがあります。 ■ジョブ実行時に設定されている環境変数が異なります。 起動されるプロセスには特定の環境変数の設定を必要とするものがあります。問題となっているプロセスを起動する前に、環境変数を表示させるsetコマンド等を挿入し、必要な環境変数が設定されているか確認してください。 ■ジョブ実行時のカレントディレクトリが異なります。 コマンドやファイルのパスを相対パスで記述していると参照できないことがあります。そのような場合はジョブ実行時のカレントディレクトリを確認するか、絶対パスで記述してください。 ■標準入力を使用するプロセスJobCenterは標準入力をクローズした状態でジョブを起動しますので、標準入力がオープンされた状態で実行されることを期待しているプロセスは動作しないことがあります。 この場合、次のどちらかの対処が必要です。設定方法については<NQS機能利用の手引き>を参照してください。 ▪ 標準入力を読み込まないようにする ▪ JobCenterのShell Strategyの設定をFREEにする。

Q.4	JobCenterから起動したジョブを強制停止、スキップなどを行っても、ジョブから起動したプロセスが終了しないことがある。
A.4	JobCenterはジョブのプロセスをプロセスグループという単位で管理しています。強制停止やスキップなどを行うと、JobCenterはジョブのプロセスグループ対してSIGKILLを送信し、プロセスグループ内の全プロセスを強制終了させます。 しかし、新しいプロセスグループを生成するようなプロセスを実行している場合はJobCenterの管理対象外となりますので、そのプロセスを終了させることはできません。

Q.5	JobCenterからシステムを停止するコマンドを実行したい。
A.5	JobCenterはデフォルトの設定の場合、マシンリブート時にマシン停止前に実行中であったジョブを再実行します。 shutdownコマンドなどのコマンドをJobCenterのジョブとして実行すると、マシン起動するたびにJobCenterからshutdownコマンドを含むジョブが再実行されて永久にシステムが起動完了しないという状態に陥る危険性がありますので、このようなコマンドはJobCenterのジョブとしては実行してはいけません。 どうしても実行する必要がある場合は、最低限次の1.～3.の対処を行ってください。 1. 単位ジョブのパラメータでリスタート属性をDISABLEに設定し、OSがリブートした後に同じジョブが再実行されないようにしてください。 2. 新しいプロセスグループを生成し、そのプロセスグループでshutdownコマンドを実行してください。また、shutdownコマンド起動後、ジョブは速やかに終了させてください。 3. shutdownコマンド起動後、実際にshutdown停止処理が開始されるまで十分に待機時間をとるようスクリプトに記述してください。

Q.6	rebootコマンドでシステムを停止させると、再起動時、ジョブのステータスが異常になることがある。
A.6	rebootコマンドは、通常のシステム停止の手順を行わず、動作中のプロセスを強制終了させて、システムを再起動させることがあります。 このような場合、JobCenterは停止処理を行えず強制終了させられますので、ジョブのステータスファイルを更新できないため、再起動時に異常なステータスで表示されることがあります。

Q.7	スクリプトの正しい記述方法が判らない
A.7	単位ジョブから起動するスクリプトはJobCenter自身が解釈しているのではなく、別途起動したOSのシェル（コマンドインタプリタ）プログラムが構文解析し、実行しています。 スクリプトの記述方法や文法については、各シェルやコマンドインタプリタのマニュアルを参照してください。

Q.8	ジョブ実行時に設定される環境変数（.profile, .cshrcなど）
A.8	JobCenterはジョブ実行開始時にシェルを起動しますが、これはログインシェルではありません。従ってcshを使用した場合は~/.cshrcは読み込まれますが、shを使用した場合は~/.profile は読み込まれない、という現象が発生します。 この違いはシェルプログラム自身の仕様によるもですので、JobCenterの設定でこれら挙動を変更することはできません。 ジョブを実行するシェルを単位ジョブパラメータ設定で変更するか、ジョブのスクリプトに直接必要な内容を記述してください。

Q.9	スケジュール投入時に設定される環境変数
A.9	ジョブネットワークをスケジュール起動させる場合、JobCenter内部に保存されているユーザ毎の環境変数が復元され、ジョブ実行時に設定されます。JobCenterでユーザの環境変数を保存する契機は以下のとおりです。 ■CL/Win, CL/Web によるログイン時 ■GUI操作によるスケジュールの作成/更新時 ■アップロード機能によるスケジュールの作成/更新時

Q.10	ACOSやSXのNQSとJobCenterとの間でジョブ連携を行おうとしているが、JobCenterからACOSまたはSX側へのNQSへのジョブ投入がうまく行えない。
A.10	JobCenterをACOSまたはSXのNQSとジョブ連携を行えるようにするためは、JobCenter側のNQSのモードをNEC拡張モードからCOSMICモードに変更する必要があります。 JobCenterの導入されているマシン上で、nmapmgrコマンドを使用して、JobCenter側のNMAPLISTに登録されている連携対象のホスト（ACOSまたはSX）のマシンタイプを変更してください。 echo "set type <マシンID> cos" | nmapmgr <マシンID>は、ACOSまたはSXのマシンIDです。 詳細については<コマンドリファレンス>の「3.12 nmapmgr ネットワークの構成管理」を参照してください。

Q.11	HP-UX上でのグループ権限に依存するジョブ実行
A.11	JobCenterからのジョブ実行は内部で initgroups というシステムコールを用いていますが、HP-UXでは当該システムコールが /etc/group ではなく /etc/logingroup を参照します。 そのため、ジョブ実行時に /etc/group の内容を反映させるためには ■/etc/group をコピーして /etc/logingroup を作成する ■/etc/logingroup -> /etc/group のシンボリックリンクを作成する 等の準備を行う必要があります。

Q.12	CL/Winの描画が遅い。
A.12	Aeroが有効の場合、CL/Winの描画が遅くなります。 CL/Winの描画遅延の対処方法は、以下の手順になります。 1. CL/WinのNscl.exeの右クリックメニューのプロパティをクリックする 2. 互換性タブの[設定]-[デスクトップコンポジションを無効にする]にチェックをつける 3. OKボタン、または適用ボタンを押し、プロパティの設定を保存する

25.2. エラーログファイルの設定を変更する

JobCenterの起動時や終了時、警告・異常発生時など様々な契機によってログが記録されます。本章ではこれらエラーログファイルのサイズやバックアップ数の変更方法について説明します。

エラーログファイルはJobCenterの常駐プロセス及び内部コマンドが記録するもので、サポート窓口へお問い合わせいただく際に、そのお問い合わせの内容によっては採取していただく必要があります。ただし、それぞれのエラーログファイルを個別に採取していただく必要はなく、情報採取コマンドを実行することで一括して採取が可能です。採取方法については「25.3 JobCenter MG/SV の障害発生時、原因究明に必要な情報を採取する」を参照してください。

なお、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。

25.2.1. Windows版

Windows版の場合、サーバの環境設定ウィンドウから行います。

1. JobCenter管理者ユーザでWindowsにログオンしてください。

2. スタートメニューから、［すべてのプログラム］－［JobCenter］－[SV]－［サーバの環境設定］をクリックします。

3. ［JobCenterサーバの環境設定］ウィンドウが表示されますので、左ペインの「サイト」から対象となるサイトをダブルクリックします。

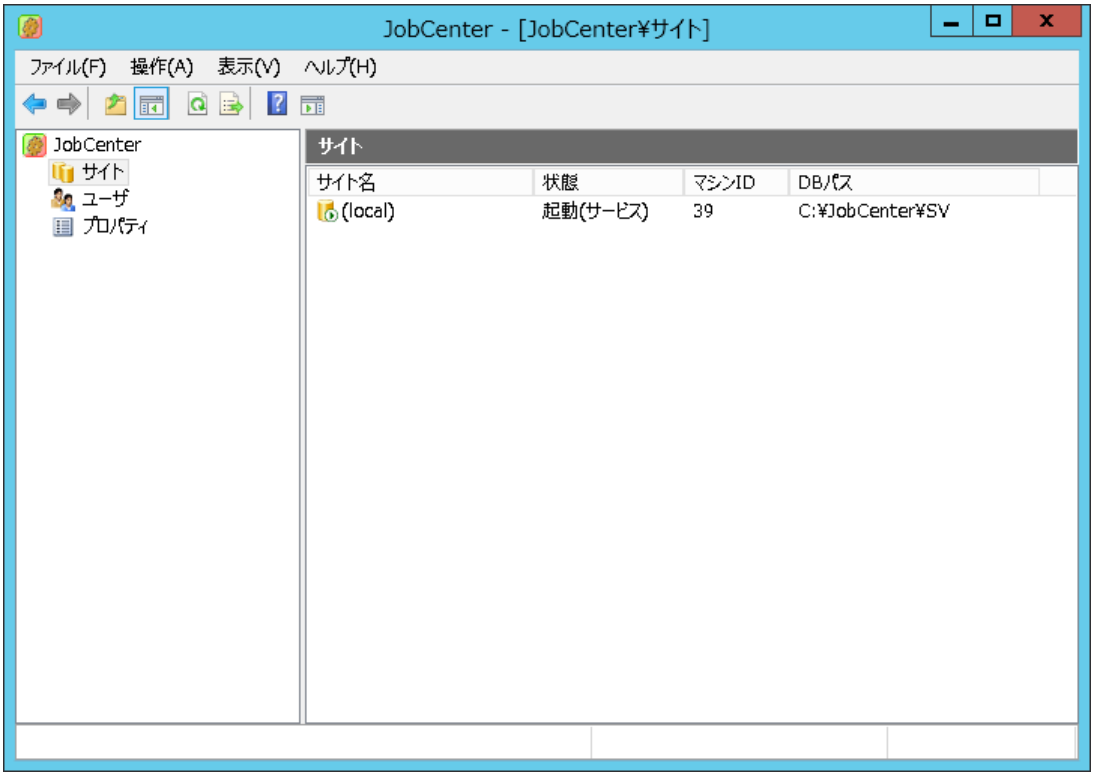

図25.1 JobCenterサーバの環境設定の画面例

4. サイトのプロパティから[デバッグログ]タブを選択します。

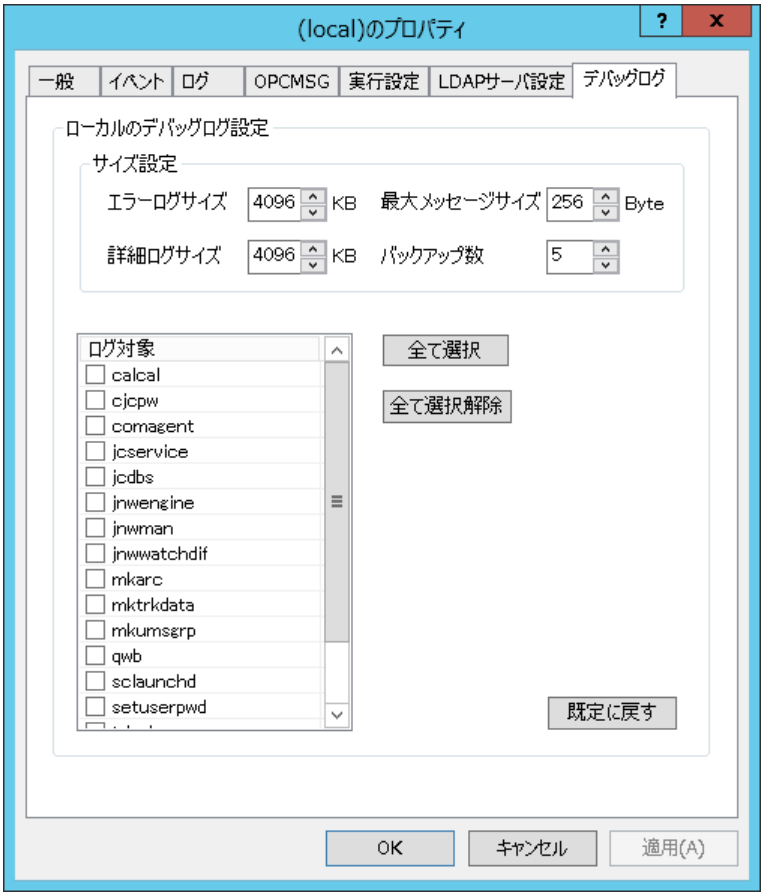

図25.2 ［デバッグログ］タブの画面例

設定項目は以下のとおりです。

［サイズの設定］

■エラーログサイズ

エラーログのファイルサイズを設定します。

表25.1「エラーログの名称とファイルパス」が設定対象になります。以下の範囲で設定してください。

既定値	4096 KB
設定範囲	64～4096 KB

表25.1 エラーログの名称とファイルパス

ログ名称	ファイルパス
calcalErr.log	%InstallDirectory%\spool\log\error\calcalErr.log
cjcpwErr.log	%InstallDirectory%\spool\log\error\cjcpwErr.log
comagentErr.log	%InstallDirectory%\spool\log\error\comagentErr.log
jcserviceErr.log	%InstallDirectory%\spool\log\error\jcserviceErr.log
jcdbsErr.log	%InstallDirectory%\spool\log\error\jcdbsErr.log
jnwengineErr.log	%InstallDirectory%\spool\log\error\jnwengineErr.log
jnwmanErr.log	%InstallDirectory%\spool\log\error\jnwmanErr.log

jnwwatchdifErr.log	%InstallDirectory%\spool\log\error\jnwwatchdifErr.log
mkarcErr.log	%InstallDirectory%\spool\log\error\mkarcErr.log
mktrkdataErr.log	%InstallDirectory%\spool\log\error\mktrkdataErr.log
mkumsgrpErr.log	%InstallDirectory%\spool\log\error\mkumsgrpErr.log
qwbErr.log	%InstallDirectory%\spool\log\error\qwbErr.log
sclaunchdErr.log	%InstallDirectory%\spool\log\error\sclaunchdErr.log
setuserpwdErr.log	%InstallDirectory%\spool\log\error\setuserpwdErr.log
trkrdrErr.log	%InstallDirectory%\spool\log\error\trkrdrErr.log
trksrchErr.log	%InstallDirectory%\spool\log\error\trksrchErr.log
umsclientErr.log	%InstallDirectory%\spool\log\error\umsclientErr.log
jnwlauncherErr.log	%InstallDirectory%\spool\log\error\jnwlauncherErr.log
jl_submitErr.log	%InstallDirectory%\spool\log\error\jl_submitErr.log
jl_logmonitorErr.log	%InstallDirectory%\spool\log\error\jl_logmonitorErr.log

%InstallDirectory%はJobCenterのインストールディレクトリです。デフォルトはC:\JobCenter\SVになります。また、クラスタ環境の場合は%InstallDirectory% の部分を <JobCenterDBパス>と読み替えてください。

■詳細ログサイズ

詳細ログ（デバッグログ）のファイルサイズを設定します。

表25.2「詳細ログの名称とファイルパス」が設定対象になります。以下の範囲で設定してください。

既定値	4096 KB
設定範囲	64～4096 KB

表25.2 詳細ログの名称とファイルパス

ログ名称	ログファイル名
calcalDbg.log	%InstallDirectory%\spool\log\debug\calcalDbg.log
cjcpwDbg.log	%InstallDirectory%\spool\log\debug\cjcpwDbg.log
comagentDbg.log	%InstallDirectory%\spool\log\debug\comagentDbg.log
jcserviceDbg.log	%InstallDirectory%\spool\log\debug\jcserviceDbg.log
jcdbsDbg.log	%InstallDirectory%\spool\log\debug\jcdbsDbg.log
jnwengineDbg.log	%InstallDirectory%\spool\log\debug\jnwengineDbg.log
jnwmanDbg.log	%InstallDirectory%\spool\log\debug\jnwmanDbg.log
jnwwatchdifDbg.log	%InstallDirectory%\spool\log\debug\jnwwatchdifDbg.log
mkarcDbg.log	%InstallDirectory%\spool\log\debug\mkarcDbg.log
mktrkdataDbg.log	%InstallDirectory%\spool\log\debug\mktrkdataDbg.log
mkumsgrpDbg.log	%InstallDirectory%\spool\log\debug\mkumsgrpDbg.log
qwdDbg.log	%InstallDirectory%\spool\log\debug\qwdDbg.log
sclaunchdDbg.log	%InstallDirectory%\spool\log\debug\sclaunchdDbg.log
setuserpwdDbg.log	%InstallDirectory%\spool\log\debug\setuserpwdDbg.log
trkrdrDbg.log	%InstallDirectory%\spool\log\debug\trkrdrDbg.log

trksrchDbg.log	%InstallDirectory%\spool\log\debug\trksrchDbg.log
umsclientDbg.log	%InstallDirectory%\spool\log\debug\umsclientDbg.log
jnwlauncherDbg.log	%InstallDirectory%\spool\log\debug\jnwlauncherDbg.log
jl_submitDbg.log	%InstallDirectory%\spool\log\debug\jl_submitDbg.log
jl_logmonitorDbg.log	%InstallDirectory%\spool\log\debug\jl_logmonitorDbg.log

%InstallDirectory%はJobCenterのインストールディレクトリです。デフォルトはC:\JobCenter\SVになります。

また、クラスタ環境の場合は %InstallDirectory% の部分を <JobCenterDBパス>と読み替えてください。

■最大メッセージサイズ

ログファイルに記述されるメッセージ1つあたりの最大長を設定します。以下の範囲で設定してください。

既定値	256 Byte
設定範囲	128～10240 Byte

■バックアップ数

エラーログおよび詳細ログがバックアップされるファイル数を設定します。

既定値	5 世代
設定範囲	1～10 世代

■ログ対象

詳細ログを取得する対象を設定します。採取するログ対象のチェックボックスにチェックしてください。

詳細ログ（デバッグログ）は、お客様からのお問い合わせの内容によって、サポート窓口から採取を依頼させていただく場合があります。

詳細ログを採取する場合、JobCenterの正常系動作も含めた内部処理のトレースを取得するため、実行速度が遅くなりますので、サポート窓口から採取依頼がある場合以外には採取しないようにしてください。

■全て選択

ログ対象の全てのチェックボックスをチェック状態にします。

■全て選択解除

ログ対象の全てのチェックボックスのチェックを外します。

■既定に戻す

既定値に戻します。

■「OK」または「適用」

設定が終了したら「OK」または「適用」ボタンを押して設定を有効にしてください。この際、JobCenterの再起動は必要ありません。

25.2.2. UNIX版

UNIX版の場合、以下の設定ファイルを変更します。

ローカルサイト	/usr/spool/nqs/gui/log.conf
クラスタサイト	<JobCenterDBパス>/nqs/gui/log.conf

log.confのサンプルは以下の通りです。

```
#
# JobCenter log configuration
#
#   XXXX_LOG_SIZE: log size (KB)
#   XXXX_LOG_NUM:  number of backup log files
#

JNWENGINE_LOG_SIZE=4096
JNWENGINE_LOG_NUM=5

JNWCASTER_LOG_SIZE=4096
JNWCASTER_LOG_NUM=5

COMAGENT_LOG_SIZE=4096
COMAGENT_LOG_NUM=5

SCLAUNCHD_LOG_SIZE=4096
SCLAUNCHD_LOG_NUM=5

JCDBS_LOG_SIZE=4096
JCDBS_LOG_NUM=5

CJCPW_LOG_SIZE=4096
CJCPW_LOG_NUM=5

JNWMAN_LOG_SIZE=4096
JNWMAN_LOG_NUM=5

JNWLAUNCHER_LOG_SIZE=4096
JNWLAUNCHER_LOG_NUM=5

JL_SUBMIT_LOG_SIZE=4096
JL_SUBMIT_LOG_NUM=5

JL_LOGMONITOR_LOG_SIZE=4096
JL_LOGMONITOR_LOG_NUM=5
```

モジュール毎に以下の設定を行うことが可能です。

パラメータ名	説明	既定値	範囲
<モジュール名>_LOG_SIZE	ログサイズ(KB)	4096	64～10240
<モジュール名>_LOG_NUM	バックアップ世代数	5	1～10

■設定変更後はJobCenterの再起動が必要です。

■無効な値を指定した場合は既定値で動作します。

各モジュールのログファイルは以下のディレクトリに格納されています。

ローカルサイト	/usr/spool/nqs/log/
クラスタサイト	<JobCenterDB/パス>/nqs/log/

各モジュールのログファイル名は以下の通りです。

表25.3 ログファイル名

モジュール名	ログファイル名
jnwengine	jeerr.log
jnwcaster	jnwcsterr.log
comagent	comagerr.log
sclaunchd	sclerr.log
jcdbs	jcdbserr.log
cjcpw	cjcpw.log
jnwman	jnwmanerr.log
jnwlauncher	jnwlauncherr.log
jl_submit	jl_submiterr.log
jl_logmonitor	jl_logmonitorerr.log

25.2.3. NQS関連のログ

NQS関連のログファイル設定は他のモジュールとは異なり、NQS固有の形式で設定します。設定方法はWindows版/UNIX版ともに共通であり、qmgrコマンドを利用します。

■UNIX版

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

■Windows版

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

■qmgrコマンドはJobCenter管理者ユーザで実行する必要があります。

■クラスタ環境の場合、事前に環境変数NQS_SITEの設定が必要です。

■qmgrコマンドの詳細については<コマンドリファレンス>の「3.13　qmgr　構成管理および運用管理」を参照してください。

ログの設定には、以下のサブコマンドを実行します。

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

設定項目は以下の通りです。

パラメータ名	説明	既定値	範囲
logfile_size	ログサイズを設定します。単位を指定しない場合はbyteとなり、単位(kb,mb)を指定することも可能です。	1048576(1mb)	65536(64kb)～4194304(4mb)
backup_number	バックアップ世代数を指定します。	10	1～10

実行例) ログサイズを4MB、バックアップ数を5に変更する場合

```
Mgr: set error_log size=(4mb) backup=5
NQS manager[TCML_COMPLETE  ]: Transaction complete at local host.
```

現在のログの設定を確認するにはshow parametersサブコマンドを利用します。

```
Mgr: show parameters

  Maximum global batch run_limit = 1000
  Maximum global network run_limit = 1000
  Maximum global pipe run_limit = 1000
  Debug level =  0
  Default batch_request priority = 31
  Default batch_request queue = NONE
  Default destination_retry time = 16 seconds
  Default destination_retry wait = 300 seconds
  Default device_request priority = 31
  No default print forms
  Default print queue = NONE
  (Pipe queue request) Lifetime = 0 hours
  Default network_retry time = 16 seconds
  Default network_retry wait = 0 seconds
  Default network_retry time_out = 300 seconds
  Default stage_retry time = 259200 seconds
  Default stage_retry wait = 300 seconds
  Default expire time = 259200 seconds
  Log_file = /dev/null
  Error_log size = 4194304 bytes
  Error_log backup number = 5
  Mail account = root
  Maximum number of print copies = 2
  Maximum failed device open retry limit = 2
  Maximum print file size = 1000000 bytes
  Netdaemon = /usr/lib/nqs/netdaemon
  Netclient = NONE
  Netserver = /usr/lib/nqs/netserver
  (Failed device) Open_wait time = 5 seconds
  NQS daemon is not locked in memory
  Next available sequence number = 1400
  Batch request shell choice strategy = FREE
  Mapping mode = TYPE3
  Maximum batch request priority = 0
  Maximum global group submit limit = Unlimited
  Maximum global user submit limit = Unlimited
  Maximum global group run limit = Unlimited
  Maximum global user run limit = Unlimited
  Maximum IDC connection number = 32
  Qwatch event spool size = 2097152
  Qwatch event expier time = 3600
  Inter Queue Scheduling mode = TYPE0
```

"Error_log size"がサイズ、"Error_log backup number"がバックアップ世代数を表しています。

25.2.4. jcwebserver関連のログ

jcwebserver関連のログファイル設定は他のモジュールとは異なり、jcwebserver固有の形式で設定します。設定方法はWindows版/UNIX版ともに共通であり、jcwebserver.confファイルにて設定します。

詳細については「5.7 jcwebserverの動作設定について」を参照してください。

また、jcwebserver関連のログファイルは、以下のディレクトリに格納されます。

Windows	ローカルサイト	%InstallDirectory%\spool\log\error\
	クラスタサイト	<JobCenterDBパス>\spool\log\error\
UNIX	ローカルサイト	/usr/spool/nqs/log/
	クラスタサイト	<JobCenterDBパス>/nqs/log/

jcwebserver関連のログファイル名は以下の通りです。

表25.4 ログファイル名

ログの種類	ログファイル名
サーバログ	jcwebserver.log
アクセスログ	jcwebserverAcs.log
エラーログ	jcwebserverErr.log

25.3. JobCenter MG/SV の障害発生時、原因究明に必要な情報を採取する

JobCenter MG/SV で障害が発生しサポート窓口へお問い合わせいただく際に、お問い合わせの内容によっては原因究明に必要な情報を採取していただく必要があります。本章では情報の採取方法について説明します。

情報の採取方法は以下の2つがあります。どちらの手順も採取する情報に違いはありませんので、実施しやすい方法を選択してください。

採取方法	説明
「25.3.1 JobCenter CL/Winで採取する場合」	JobCenter CL/Win で障害が発生したサイトに接続し、情報を採取します。 JobCenter MG/SV が起動している必要があります。
「25.3.2 jc_getinfoコマンドで採取する場合」	障害が発生したサイトが存在するマシン上でjc_getinfoコマンドを実行し、情報を採取します。 障害によりJobCenter MG/SV が起動できない場合は、本手順で実施してください。

25.3.1. JobCenter CL/Winで採取する場合

JobCenter CL/Win から情報を採取する手順は以下のとおりです。

■情報の採取はJobCenter管理者ユーザのみ実行できます。

■JobCenter MG/SV を起動できない場合は、JobCenter CL/Win で情報を採取することはできません。「25.3.2 jc_getinfoコマンドで採取する場合」の手順で採取してください。

1. JobCenter CL/Winから障害が発生したサイトにJobCenter管理者ユーザで接続します。

2. [ファイル]→[jc_getinfoの実行]をクリックします。

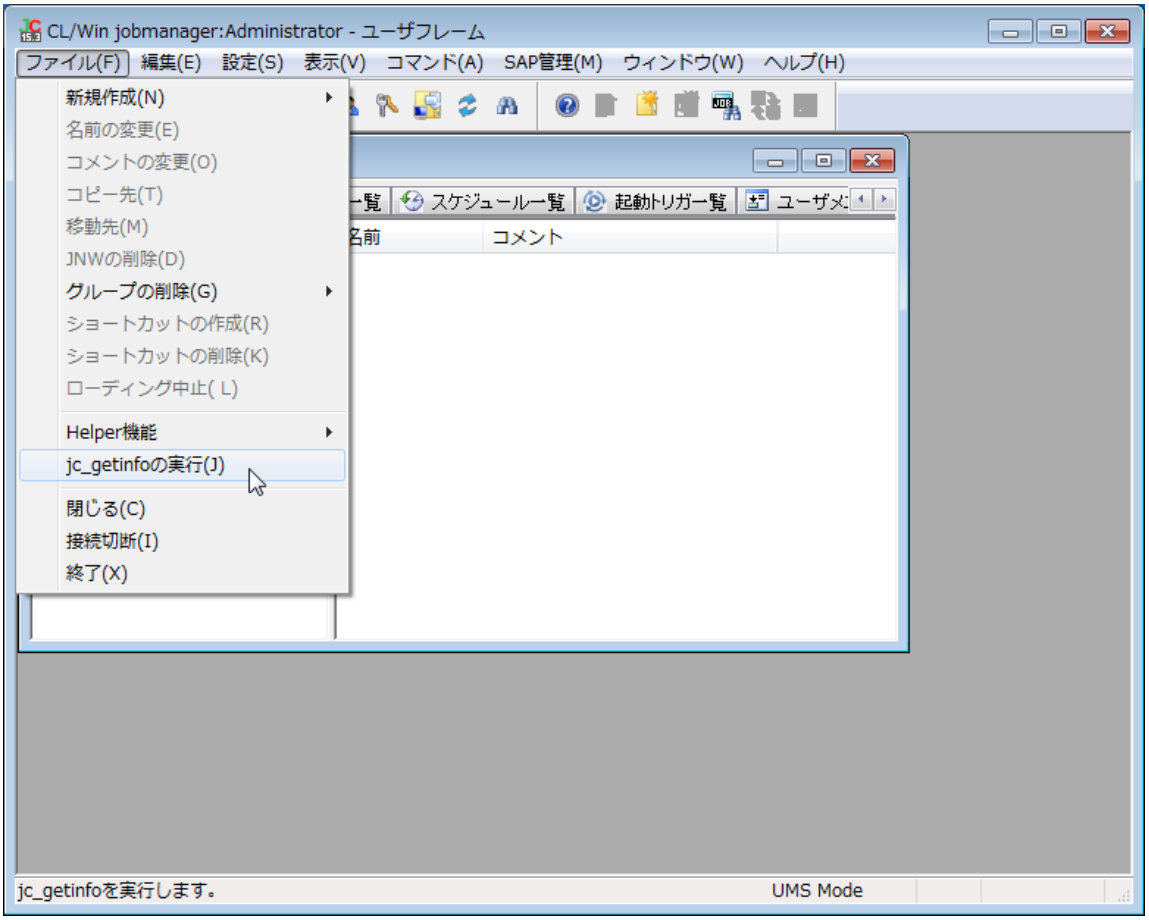

図25.3 JobCenter CL/Winからjc_getinfoを実行

[jc_getinfoの実行]画面が表示されます。

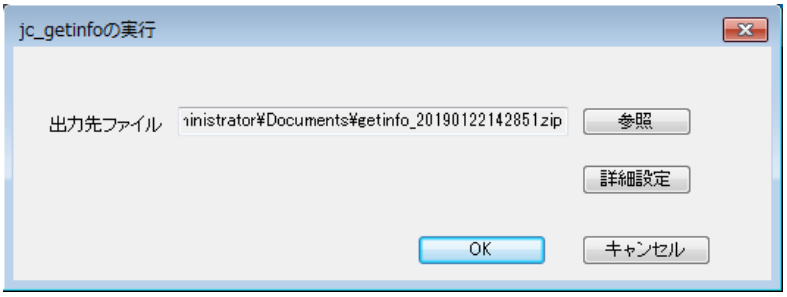

図25.4 [jc_getinfoの実行]画面

3. 採取する情報の保存先を変更したい場合は、[参照]ボタンから設定してください。

デフォルトではWindowsのログインユーザのドキュメント配下に以下のファイル名で保存されます。

```
getinfo_YYYYMMDDhhmmss.zip
```

■YYYYMMDDhhmmssは[jc_getinfoの実行]画面表示時の日時になります。

■ファイルはzip形式で保存されますので、拡張子は変更しないでください。

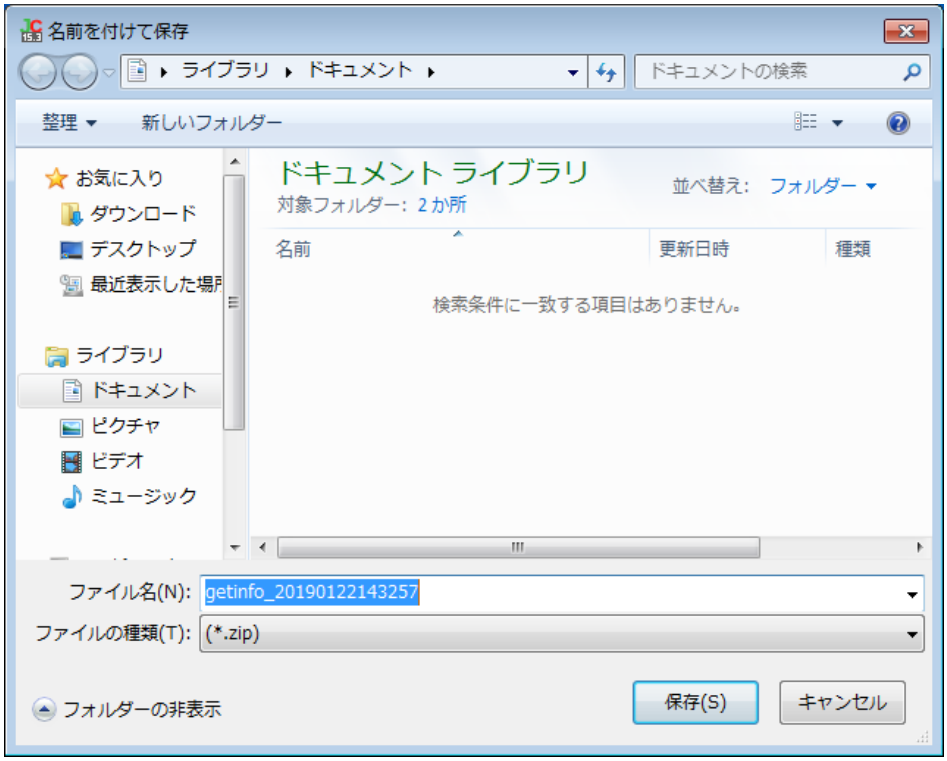

図25.5 保存先の設定画面

4. サポート窓口より情報採取時の設定変更の指示があった場合は、[詳細設定]ボタンから設定してください。

特に指示がない場合は、設定を変更する必要はありません。

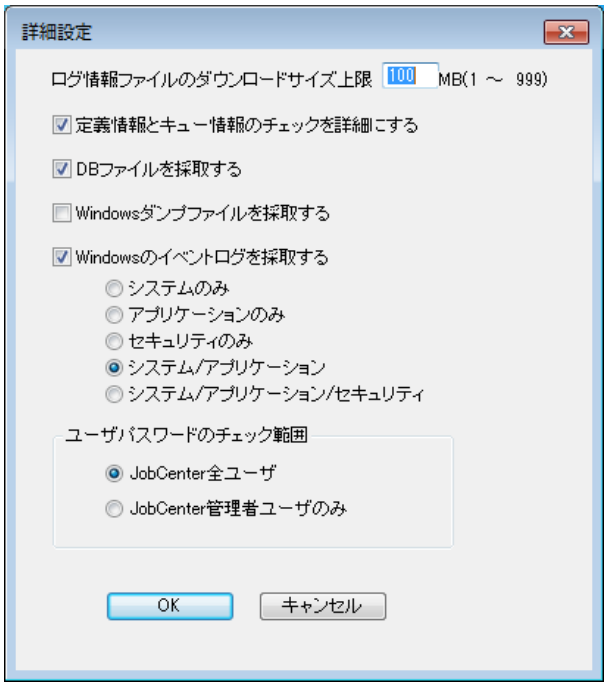

図25.6 jc_getinfoの詳細設定画面 (Windows版JobCenter MG/SV の場合)

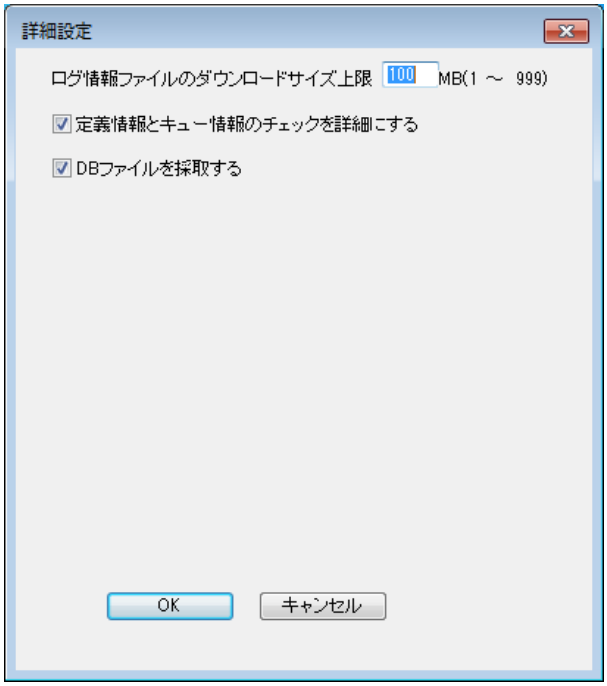

図25.7 jc_getinfoの詳細設定画面 (UNIX版JobCenter MG/SV の場合)

表25.5 [詳細設定]画面の設定

名称	説明	デフォルト値
ログ情報ファイルのダウンロードサイズ上限	採取する情報のファイルサイズの上限値(MB)を設定します。 1～999の数値を指定可能です。	100MB
定義情報とキュー情報のチェックを詳細にする	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、キューの詳細情報を取得します。	チェックあり
DBファイルを採取する	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を取得します。	チェックあり
Windowsダンプファイルを採取する	Windowsのエラー報告を取得します。 Windows版のみ表示されます。	チェックなし
Windowsのイベントログを採取する	採取するWindows OSイベントログの種類を指定します。チェックしない場合はイベントログを採取しません。 Windows版のみ表示されます。 ■システムのみ ■アプリケーションのみ ■セキュリティのみ ■システム/アプリケーション ■システム/アプリケーション/セキュリティ	チェックあり(システム/アプリケーション)

名称	説明	デフォルト値
ユーザパスワードのチェック範囲	ユーザのパスワードチェックの範囲を指定します。 Windows版のみ表示されます。 ■JobCenter全ユーザ JobCenter MG/SV の全ユーザのパスワードをチェックします。 ■JobCenter管理者ユーザのみ JobCenter管理者ユーザのみパスワードをチェックします。	JobCenter全ユーザ

5. [jc_getinfoの実行]画面の[OK]ボタンをクリックすると、[採取設定の確認]画面が表示されます。

   設定内容に問題なければ、[OK]ボタンをクリックしてください。

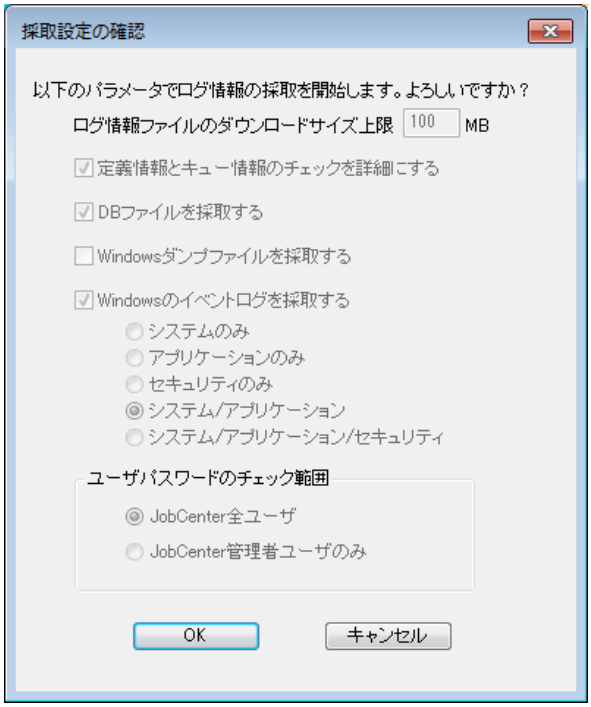

図25.8 [採取設定の確認]画面

6. [採取状況]画面が表示されます。情報の採取が完了するまでお待ちください。

   完了すると「ファイルのダウンロードが完了しました。」と表示されますので、[閉じる]ボタンで終了してください。

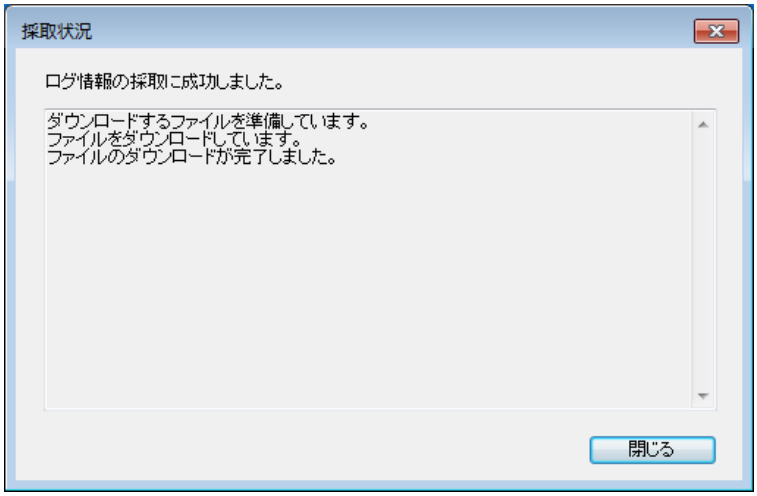

図25.9 [採取状況]画面

情報の採取に時間がかかると通信タイムアウトで失敗する場合があります。<基本操作ガイド>の「2.3.4　サーバへ接続する際のタイムアウト時間を設定する」で通信タイムアウト時間を大きくするか、「25.3.2 jc_getinfoコマンドで採取する場合」の手順で採取してください。

「ログ情報ファイルのダウンロードサイズ上限」で設定したファイルサイズを超えた場合は、ダウンロードせずに接続先のサイトのスプールディレクトリ配下に保存され、[採取状況]画面に保存先のパスが表示されます。

「ログ情報ファイルのダウンロードサイズ上限」の値を大きな値に設定して再度ダウンロードするか、手動でファイルを取得してください。

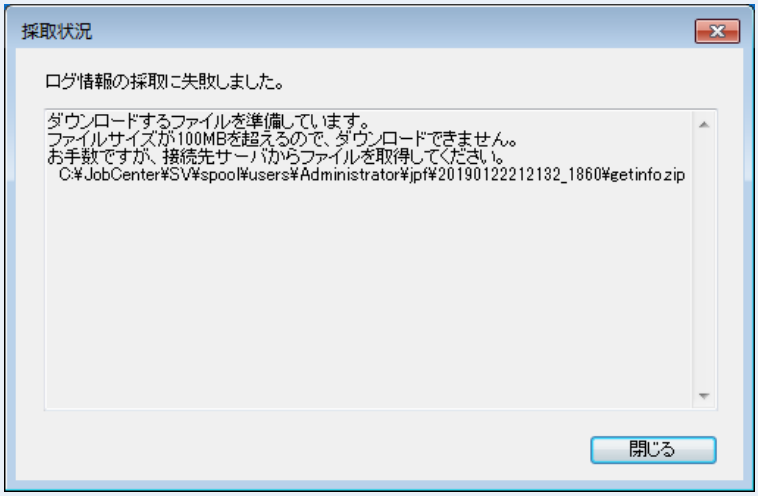

図25.10 「ログ情報ファイルのダウンロードサイズ上限」で設定したファイルサイズを超えた場合の[採取状況]画面

7. 複数のマシンと連携している場合は、全てのマシンで同様の手順で採取してください。

25.3.2. jc_getinfoコマンドで採取する場合

jc_getinfoコマンドで採取する手順は以下のとおりです。

1. 障害が発生したマシン上でjc_getinfoコマンドを実行し、情報を採取します。

   jc_getinfoコマンドの詳細は<コマンドリファレンス>の「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」を参照してください。

2. 複数のマシンと連携している場合は、全てのマシンで同様の手順で採取してください。

発行年月 Oct 2023
NEC Corporation　2023