

MQSeries® for Compaq NonStop™ Kernel



システム管理ガイド

バージョン 5.1

MQSeries® for Compaq NonStop™ Kernel



システム管理ガイド

バージョン 5.1

ご注意!

本書、および本書がサポートする製品をご使用になる前に、513ページの『付録Q. 特記事項』にある一般的な情報を必ずお読みください。

本書は、MQSeries for Compaq NSK、バージョン 5.1 および新版において特に断りのない限り、それ以降のすべてのリリースおよびモディフィケーション・レベルに適用されます。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

原典： SC34-5886-00
MQSeries® for Compaq NonStop™ Kernel
System Administration
Version 5 Release 1

発行： 日本アイ・ビー・エム株式会社

担当： ナショナル・ランゲージ・サポート

第1刷 2001.5

この文書では、平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、
平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 1993, 2001. All rights reserved.

Translation: © Copyright IBM Japan 2001

目次

図	xi
表	xiii
本書について	xv
本書の対象読者	xv
本書を理解する上での前提事項	xv
本書の使い方	xvi
インターネット上の MQSeries についての情報	xvi
MQSeries for Compaq NSK V5.1 の新機能	xvii
パフォーマンスの拡張	xvii
アップグレードされた MQSeries 機能	xviii
相互通信	xx
Compaq NSK 固有の使いやすさ	xx
第1部 手引き	1
第1章 概要	7
MQSeries とメッセージ・キューイング	7
時間に依存しないアプリケーション	8
メッセージ・ドリブン処理	8
メッセージとキュー	8
メッセージとは	8
キューとは	10
オブジェクト	11
オブジェクト名	12
オブジェクトの管理	12
MQSeries キュー・マネージャー	13
MQSeries キュー	14
プロセス定義	18
チャンネル	19
クラスター	19
名前リスト	20
システム・デフォルト・オブジェクト	20
管理	21
ローカル管理およびリモート管理	21
クライアントおよびサーバー	22

クライアント / サーバー環境での MQI アプリケーション	22
キュー・マネージャー機能の拡張	23
ユーザー出口	23
インストール可能なサービス	23
セキュリティ	24
オブジェクト権限マネージャー (OAM) 機能	24
トランザクション・サポート	25
パフォーマンスのチューニング、信頼性、スケラビリティ、およびサイジング	26

第2章 MQSeries for Compaq NSK V5.1 のアーキテクチャー	27
キュー・マネージャー・プロセスの概要	27
プロダクト・パッケージ	30
実行可能プログラム	30
バージョン 5.1 とバージョン 2.2.0.1 のアーキテクチャーの比較	31
キュー・マネージャー - 機能の概観	32
キュー・マネージャー・プロセス・モデル	34
MQSeries のファイルとサブボリューム	34

第3章 MQSeries コマンド・セットの使用	37
制御コマンドを使用した管理の実行	37
制御コマンドの使用	38
MQSC コマンドを使用した管理の実行	38
MQSC コマンドの実行	38
PCF コマンドを使用した管理の実行	39
MQSC および PCF での属性	40
エスケープ PCF	40
TS/MP (PATHWAY) 管理	41
TCP/IP リスナーの指定および制御	41
コマンド・サーバーの制御	41
チャンネル・イニシエーターの指定および制御	42
トリガー・モニターの指定および制御	42
CPU 間でのプロセスの分配の指定	43
新しい状況サーバー・プロセスの追加	45
新しいキュー・サーバー・プロセスの追加	45

MQM モニター・パネルを最新表示する頻度の指定	45	デフォルト以外の状況サーバーの追加と除去	74
キュー・マネージャーの PATHWAY 構成	46	デフォルトのキュー・サーバー名	76
PATHWAY サーバー・クラスのパラメーターの変更	54	デフォルト以外のキュー・サーバーの追加と除去.	77
MQSeries PATHWAY へのユーザー定義サーバー・クラスの追加.	55	ボリューム構造	79
第4章 キュー・マネージャーの管理	57	キュー・マネージャー FFST サブボリューム	80
はじめに	57	キュー・マネージャー・データ・ファイル・サブボリューム	80
キュー・マネージャー作成のガイドライン	58	キュー・マネージャー・エラー・ログ・サブボリューム.	81
キュー・マネージャーの作成後の構成ファイルのバックアップ	61	キュー・マネージャー・メッセージ・キュー・サブボリューム	81
キュー・マネージャーのプロパティの変更	61	キュー・マネージャー・チャンネル同期サブボリューム	82
キュー・マネージャーのホーム・ボリューム	62	オブジェクト名の変換.	83
キュー・マネージャーごとの EC プロセスの数.	62	キュー・マネージャーでの作業.	83
キュー・マネージャーのホーム端末	63	デフォルト・キュー・マネージャーの作成	83
キュー・マネージャーの PATHMON プロセス名	64	MQSeries プリンシパルの作成	85
キュー・マネージャーの CCSID	65	RDF 使用可能になっているキュー・マネージャーでの cleanrdf の実行.	85
EBCDIC データ変換の制御	65	キュー・マネージャーの開始	86
キュー・マネージャーの EMS コレクター	67	デフォルトおよびシステム・オブジェクトの復元.	86
キュー・マネージャー内の各 EC によって作動可能な状態に保持されるエージェントのプール	67	オブジェクト・ファイルの表示.	87
アイドル状態のエージェントとプロセスの再利用の最大数.	68	キュー・マネージャーの停止	87
キュー・マネージャー・プロセスのプロセス優先順位	68	問題がある場合	87
キュー・マネージャーのチャンネルの最大数	68	即時および優先キュー・マネージャー・シャットダウン.	87
キュー・マネージャーのアクティブ・チャンネルの最大数.	69	キュー・マネージャーの再始動.	88
MQSeries で使用する Guardian セグメント ID	69	既存のキュー・マネージャーのデフォルト・キュー・マネージャーへの指定.	89
デフォルト TCP/IP ポート	70	キュー・マネージャーの削除	89
キュー・マネージャーが listen する TCP/IP ポート	70	メッセージ・キュー管理 (MQM) 機能の使用	90
キュー・マネージャーで使用する TCP/IP プロセス	71	「Queue Manager Menu」の使用	92
デフォルト以外の TCP/IP プロセスやポートのための MQS-TCPLISnn サーバー・クラスの再構成.	71	「Queues」メニューの使用	95
スワップ・スペースの割り振り.	72	「Channels」メニューの使用	103
デフォルトの状況サーバー名	73	第5章 ローカル MQSeries オブジェクトの管理	117
		MQI を使用したアプリケーション・プログラムのサポート	117
		MQSC コマンドを使用してローカル管理タスクを実行する	118
		開始する前に	119

MQSC 機能を対話式で使用する	120	PCF コマンド	151
MQSC からのフィードバック	121	MQSC および PCF での属性	152
対話式での MQSC への入力を終了する	122	エスケープ PCF	152
キュー・マネージャーの属性を表示する	122	MQAI を使用して PCF の使い方を単純化	
デフォルトではないキュー・マネージャー		する	153
を使用する	123	リモート管理でコマンド・サーバーを管理す	
キュー・マネージャーの属性の更新	124	る	154
テキスト・ファイルから MQSCコマンドを実		コマンド・サーバーを開始する	154
行する	124	コマンド・サーバーの状況を表示する	154
MQSC コマンド・ファイルの使用	126	コマンド・サーバーを停止する	155
MQSC レポートの使用	126		
システムに提供された MQSC コマンド・		第7章 リモート MQSeries オブジェクトの	
ファイルを実行する	128	管理	157
runmqsc を使用してコマンドを確認する	128	チャンネル、クラスターおよびリモート・キュー	
MQSC で起こった問題の解決	129	イング	157
ローカル・キューの取り扱い	131	クラスターを使用するリモート管理	159
ローカル・キューを定義する	131	リモート・キュー・マネージャーの管理	160
キューの物理ファイル・サイズの変更	132	リモート管理のためのキュー・マネージャ	
送達不能キューを定義する	132	ーを作成する	160
デフォルト・オブジェクトの属性を表示す		リモート管理のためにチャンネルおよび伝送	
る	133	キューを作成する	161
ローカル・キュー定義をコピーする	134	チャンネルと伝送キューを定義する	162
ローカル・キューの属性を変更する	135	チャンネルを開始する	164
ローカル・キューのボリュームを変更する	136	リモートから MQSC コマンドを発行する	165
ローカル・キューのオプションを変更する	136	リモート・キューイングについての勧告	167
状況サーバーおよびキュー・サーバーにオ		リモートからの MQSC の使用に問題がある	
ブジェクトを再割り当てする	137	場合	167
ローカル・キューをクリアする	138	リモート・キューのローカル定義の作成	168
ローカル・キューを削除する	138	リモート・キューのローカル定義の働きに	
キューをブラウズする	138	ついて理解する	168
別名キューの取り扱い	142	リモート・キューにメッセージを書き込む代	
別名キューを定義する	143	替方法	170
キュー別名に関してその他のコマンドを使		リモート・キューに関してその他のコマンド	
用する	144	を使用する	170
モデル・キューの取り扱い	145	伝送キューを作成する	171
モデル・キューを定義する	145	デフォルト伝送キュー	171
モデル・キューに関してその他のコマンド		リモート・キュー定義を別名として使用する	172
を使用する	146	キュー・マネージャー別名	172
トリガー操作のためのオブジェクトの管理	146	応答先キュー別名付け	173
トリガー操作のためにアプリケーション・キュー		データ変換	173
を定義する	146	キュー・マネージャーがメッセージを組み	
開始キューを定義する	148	込み形式に変換できない場合	173
プロセス定義を作成する	148	ファイル CCSID	174
プロセス定義を表示する	149	ユーザー定義形式でのメッセージの変換	175
		キュー・マネージャー CCSID の変更	175
第6章 管理用タスクの自動化	151		

第8章 MQSeries オブジェクトの保護	177	DLQ ハンドラーのルール・テーブル	206
MQSeries リソースを保護する理由	177	制御データ	207
MQM ユーザー・グループのユーザー ID を理解する	178	ルール (パターンと処理)	209
その他の情報を表示する	179	パターン・マッチング・キーワード	209
オブジェクト権限マネージャー (OAM) を理解する	179	アクション・キーワード	211
OAM の機能	180	ルール・テーブルのルール	213
ユーザー・グループを介してアクセスを管理する	180	ルール・テーブルの処理方法	215
OAM でリソースを保護する	181	すべての DLQ メッセージを確実に処理する	216
許可にグループを使用する	182	DLQ ハンドラーのルール・テーブルの例	217
オブジェクト権限マネージャー (OAM) を使用不可にする	183		
オブジェクト権限マネージャー (OAM) コマンドを使用する	184	第10章 インストゥルメンテーションおよび EMS イベント	221
OAM コマンドの使用時に指定するもの	184	MQSeries インストゥルメンテーション・イベント	221
altmqusr コマンドを使用する	185	イベントのタイプ	222
dspmqusr コマンドを使用する	186	イベント・キューによるイベント通知	223
setmqaut コマンドを使用する	186	トリガー・イベント・キューを使用する	223
アクセス許可	187	インストゥルメンテーション・イベントの有効化	223
権限の表示コマンド	188	イベント・メッセージ	224
オブジェクト権限マネージャー (OAM) のガイドライン	188	Event Management Service (EMS) イベント	224
ユーザー ID	188	MQSeries for Compaq NSK に付属する EMS テンプレート・ファイル	224
キュー・マネージャー・ボリューム	189	MQSeries EMS イベント・テンプレートの統合	225
キュー	189	PARAM MQEMSEVENTS の定義	228
代替ユーザー権限	189	代替コレクターの使用	229
コンテキスト権限	190	MQSeries EMS イベントを処理するためのプログラムの作成	229
リモート・セキュリティに関する考慮事項	191		
チャンネル・コマンド・セキュリティ	192	第11章 トランザクション・サポートとメッセージングについての理解	231
許可指定テーブルについての理解	193	NonStop TM/MP (トランザクション・マネージャー) の使用	231
MQI 許可	193	同期点の制限	232
管理許可	197	持続メッセージでの非同期点操作	233
エスケープ PCF 中の MQSC コマンドに関する許可	198	非持続メッセージでの同期操作	234
許可ファイルの理解	201	TM/MP および MQSeries for Compaq NSK の構成要件	234
プリンシパル・データベース	201	モニター	234
OAM データベース	201	監査証跡のサイズ	235
クラス許可レコード	204	リソース・マネージャーの構成	235
全クラス許可レコード	204	トラブルシューティング	235
第9章 MQSeries 送達不能キュー・ハンドラー	205	第12章 リカバリーおよび再始動	237
DLQ ハンドラーの呼び出し	205		

フォールト・トレランスとリカバリー	238	アプリケーションまたは MQSeries for	
MQSeries のバックアップと復元	238	Compaq NSK の実行速度は低速ですか。	267
MQSeries のバックアップ	238	アプリケーション設計の考慮事項	269
MQSeries の復元	239	メッセージ長の影響	269
状況サーバーとキュー・サーバーのリカバリー		特定のメッセージの検索	269
ーおよび再始動	239	長さが異なるメッセージを含んでいるキュー	
RDF を使用した災害時回復	240	ー	269
第13章 構成ファイル	243	同期点の頻度	270
構成ファイルとは	243	MQPUT1 呼び出しの使用	270
MQSeries 構成ファイル (MQSINI)	244	不正出力	270
MQSeries 構成ファイルの内容	244	キューにメッセージが現れない場合	270
キュー・マネージャー構成ファイル (QMINI)	245	メッセージが予期しない情報または破壊さ	
キュー・マネージャー構成ファイルの内容	246	れた情報を含んでいる場合	272
キュー・マネージャー構成ファイルの例	249	分散キューを使用している場合の不正出力	
構成ファイルを編集する	254	の問題	273
構成ファイルへの変更のインプリメント	255	エラー・ログ	275
構成ファイルに関する推奨事項	255	ログ・ファイル	276
第14章 問題判別	257	早期エラー	276
予備的な検査の作成	257	オペレーター・メッセージ	277
以前に MQSeries は正常に実行されたか	257	エラー・ログの例	277
エラー・メッセージが出されたか	258	EMS イベント	279
問題について説明した戻りコードが戻され		送達不能キュー	279
ていないか	258	構成ファイルと問題判別	279
問題を再現できるか	258	MQSeries トレースの使用	280
最後に正常に実行された後、変更が行われ		トレース・ファイル名	280
たか	259	トレース・データのサンプル	280
アプリケーションは以前に正常に実行され		第 1 障害サポート・テクノロジー™ (FFST)	281
たか	259	FFST の読み方	281
アプリケーションが以前に正常に実行され		第15章 スケーラビリティおよびパフォー	
なかった場合	260	マンス	283
問題がネットワークの特定部分に影響する		概要	283
か	261	パフォーマンスおよびスケーラビリティ	
1 日の特定の時刻に問題が起こるか	262	を向上させる新規アプリケーションを設計	
問題は断続的に起こるか	262	する	283
サービス更新を適用したか	262	共用リソースの使用を最小限にしたり、使	
共通プログラミング・エラー	263	用しなくてもすむように設計する	284
コマンドに関する問題	263	パフォーマンス・チューニングは本質的に	
次に何を行うか	263	対話式	284
不正な出力を得たか	264	持続メッセージ	285
PCF コマンドからの応答を受け取れなか		非持続メッセージ	285
ったか	264	非持続メッセージおよびチャンネル	286
キューの一部が障害を起こしているか	266	キュー・サーバーおよびキュー・ファイル	287
問題はリモート・キューにのみ影響するか	267	持続メッセージのストレージ	287
		非持続メッセージのストレージ	288
		キュー・サーバーの CPU 分散	288

altmqfls を使用してキュー・サーバーにキューを再割り当てする	288	altmqfls (キュー・ファイル属性の更新).	327
クラスター伝送キュー:		altmqusr (MQSeries ユーザー情報の更新)	333
SYSTEM.CLUSTER.TRANSMIT.QUEUE	289	cleanrdf (RDF ハウスキーピングの実行)	335
altmqfls を使用してキュー・ファイルの配置を変更する	289	cnvclchl (クライアント・チャンネル定義の変換).	337
キュー・ファイルの区画化.	290	crtmqcvx (データ変換)	339
メッセージ・オーバーフロー・ファイル	290	crtmqm (キュー・マネージャーの作成).	342
ブラウザ中にメッセージをバッファーに入		dltmqm (キュー・マネージャーの削除).	347
れる	291	dspmqaout (権限の表示)	349
その他のキュー・サーバー・オプション	292	dspmqsrv (コマンド・サーバーの表示)	354
CPU の割り当て	293	dspmqlfs (MQSeries ファイル属性の表示)	356
FASTPATH バインディング・アプリケーション・プログラム.	295	dspmqrtrc (MQSeries 定様式トレース出力の表示).	360
バックグラウンド.	295	dspmqsrv (MQSeries ユーザー情報の表示)	361
MQI オーバーヘッドを削減する	296	endmqsv (コマンド・サーバーの終了).	363
FASTPATH バインディングの使用可能化	296	endmqm (キュー・マネージャーの終了)	365
FASTPATH バインディングの使用時の制限	297	endmqtrc (MQSeries トレースの終了)	368
第16章 データ保全性と可用性	299	instmqm (MQSeries for Compaq NSK のインストール)	370
データ保全性	299	runmqchi (チャンネル・イニシエーターの実行)	371
可用性	300	runmqchl (チャンネルの実行).	372
持続データと非持続データ.	302	runmqdlq (送達不能キュー・ハンドラーの実行).	373
持続メッセージ	302	runmqslr (リスナーの実行)	375
非持続メッセージ.	305	runmqsc (MQSeries コマンドの実行).	377
データベースの一貫性	305	runmqtrm (トリガー・モニターの開始).	381
内部データベースの一貫性.	306	setmqaut (権限の設定 / リセット)	382
外部データベースの一貫性.	306	strmqsv (コマンド・サーバーの開始)	389
OpenTMF	307	strmqm (キュー・マネージャーの開始).	390
NonStop Tuxedo	308	strmqtrc (MQSeries トレースの開始).	392
インターリーブド・アプリケーション・トランザクション	308	upgmqm (V2.2.0.1 キュー・マネージャーのアップグレード).	396
MQSeries の重要なデータベース・ファイル	309		
重要なプロセス	310	第3部 付録	399
クラスター	318	付録A. MQSeries for Compaq NSK の概要	401
可用性に関する構成上の考慮事項	318	プログラムと部品番号	401
データ保全性に関する構成上の考慮事項	319	ハードウェア要件	401
		ソフトウェア要件	401
		セキュリティ	402
		メンテナンス機能	402
		互換性	403
		サポートされるコンパイラー	403
		ライセンス管理	404
		言語の選択	404
第2部 リファレンス	321		
第17章 MQSeries 制御コマンド	323		
制御コマンドの要約	323		
名前の使用	324		
構文図の見方	325		

国際化対応	404	データ変換	435
付録B. システム・デフォルト	405	付録I. アプリケーションの構築および実行	437
付録C. MQSeries for Compaq NSK 用の TACL 環境の設定	409	アプリケーションの作成	437
キュー・サーバーの調整パラメーター	411	MQGET 待機インターバルとチャネル DISCONT および HBINT の使用	437
付録D. コマンド・セットの比較.	413	作業単位 (トランザクション) 管理	438
付録E. 手動によるキュー・マネージャーの 停止および除去	417	一般設計の考慮事項	440
手動によるキュー・マネージャーの停止	417	XA インターフェース	440
手動によるキュー・マネージャーの除去	418	MQGMO_LOCK を指定した MQGMO_BROWSE_*	440
付録F. MQSeries および Compaq NonStop Server for Java	419	起動されるアプリケーション	441
トランザクションに関する考慮事項.	420	サポートされる言語および環境	443
付録G. MQSeries for Compaq NSK によ ってサポートされる MQSC	421	スレッドを使用したアプリケーションの作 成における考慮事項	444
MQSC の属性	424	アプリケーションのコンパイルおよびバイン ド	444
チャンネル状況情報 (DISPLAY CHSTATUS)	424	FASTPATH バインディングと STANDARD バインディングとの比較	445
MAXUMSGS と MAXHANDS	425	アプリケーションの実行	446
HARDENBO と NOHARDENBO	425	付録J. MQSeries 管理インターフェース (MQSeries Administration Interface (MQAI)).	447
CONNNAME	425	付録K. MQSeries for Compaq NSK サン プル・プログラム	449
起動されたプログラムの USERDATA	425	C サンプル・プログラムの作成	450
出口名のオブジェクトの属性としての使用	426	非ネイティブ (非ネイティブ静的ライブラ リー MQMLIB の使用)	450
付録H. アプリケーション・プログラミン グ・リファレンス	427	ネイティブ (ネイティブ静的ライブラリー MQMLIBN の使用)	451
構造データ・タイプ	427	ネイティブ (SRL MQSRLLIB の使用)	452
MQCNO - 接続オプション.	428	C++ サンプル・プログラムの作成	453
MQGMO - Get メッセージ・オプション	429	ネイティブ (ネイティブ静的ライブラリー MQMLIBN の使用)	453
MQMD - メッセージ記述子	430	ネイティブ (SRL MQSRLLIB の使用)	454
MQPMO - Put メッセージ・オプション	431	COBOL サンプル・プログラムの作成	454
MQI 呼び出し	431	非ネイティブ (非ネイティブ静的ライブラ リー MQMLIB の使用)	454
MQCLOSE - オブジェクトのクローズ	432	ネイティブ (ネイティブ静的ライブラリー MQMLIBN の使用)	455
MQDISC - キュー・マネージャーの切断	433	ネイティブ (SRL MQSRLLIB の使用)	456
MQINQ - オブジェクト属性に関する問い 合せ	433	TAL サンプル・プログラムの作成	457
MQOPEN - オブジェクトを開く	433	非ネイティブ (非ネイティブ静的ライブラ リー MQMLIB の使用)	457
MQSET- オブジェクト属性の設定	434		
MQSeries の属性	434		
すべてのキューの属性	434		
ローカルおよびモデル・キューの属性	434		
キュー・マネージャーの属性	435		

OSS でのサンプル・プログラムの作成 (ネイティブ・モード専用)	458	イベント・メッセージの形式	498
付録L. ユーザー出口	459	プログラム式コマンド形式 (PCF)	500
サポートされるユーザー出口	459	PCF メッセージ記述子	501
出口名の形式	460	PCF ヘッダー (MQCFH)	501
MQ_LOAD_ENTRY_POINT_EXIT - ユーザー出口のロード	460	PCF ストリング・パラメーター (MQCFST)	502
MQLXP -		PCF 整数リスト・パラメーター (MQCFIL)	502
MQ_LOAD_ENTRY_POINT_EXIT パラメーター構造	461	PCF 整数 (MQCFIN)	503
ユーザー出口のインストール	465	PCF ストリング・リスト (MQCFSL)	503
MQSeries プライベート SRL への出口のインストール	465	MQSeries for Compaq NSK がサポートする PCF コマンド	504
MQSeries ネイティブ静的ライブラリーへの出口のインストール	467	PCF コマンド応答	505
MQSeries 非ネイティブ静的ライブラリーへの出口のインストール	467	インストール可能なサービス	506
		許可サービス・インターフェース	506
		名前サービス・インターフェース	506
付録M. 通信のセットアップ	471	付録P. MQSeries for Compaq NSK で使用する EMS イベント・テンプレート	507
SNA チャンネル	471	付録Q. 特記事項	513
LU 6.2 応答側プロセス	472	商標	515
TCP/IP チャンネル	479	参考文献	517
通信の例	480	MQSeries 複数のプラットフォームに共通する資料	517
SNAX 通信の例	480	MQSeries プラットフォーム固有の資料	518
ICE 通信の例	487	ソフトコピー資料	519
TCP/IP 通信の例	491	HTML 形式	519
		PDF	519
付録N. MQSeries クライアント	495	インターネットで利用できる MQSeries 情報	520
クライアント・サポート	495	参考資料	520
セキュリティに関する考慮事項	495	用語集および略語集	523
付録O. プログラム式システム管理	497	索引	539
インスツルメンテーション・イベント	497		
MQSeries for Compaq NSK がサポートするイベント・タイプ	497		



1. MQSeries for Compaq NSK のプロセス	28	27. 「Monitor Channels」パネル	111
2. MQSeries for Compaq NSK V5.1 のコンポーネント	33	28. 「Channel Status」パネル	112
3. PATHWAY 構成の例	46	29. 「Start/Stop Channel」パネル	113
4. MQSeries for Compaq NSK MQM Main Menu	91	30. 「Reset Channel」パネル	114
5. 「Queue Manager Menu」パネル	92	31. 「Resolve Channel」パネル	115
6. 「Queue Manager Menu」パネル 2	93	32. 「Copy Channel」パネル	116
7. QUEUE MANAGER TRACE MENU	94	33. キュー、メッセージ、およびアプリケーション	118
8. 「Search Criteria」パネル (キュー)	96	34. QMGR ALL の出力例	123
9. 「Queue Menu」	97	35. MQSC コマンド・ファイル mymqscin からの抽出	126
10. 「Create Queue」パネル	98	36. MQSC レポート・ファイル mymqscou からの抽出	128
11. 「Create Local Queue」パネル	99	37. リモート管理	161
12. 「Create Local Queue」パネル 2	99	38. リモート管理のためのチャンネルとキューのセットアップ	162
13. 「Create Remote Queue」パネル	100	39. SAFEGUARD を設定したグループを使用して許可を与える	183
14. 「Copy Queue」パネル	101	40. 制御データの例	208
15. 「Display/Modify Local Queue」パネル	102	41. ルールの例	209
16. 「Monitor Local Queues」パネル	103	42. MQSeries 構成ファイル (MQSINI) の例	245
17. 「Search Criteria」パネル (チャンネル)	104	43. キュー・マネージャー構成ファイル (QMINI) の例	249
18. 「Channel Menu」	105	44. トレースのサンプル	281
19. 「Display/Modify Sender Channel」パネル (1)	106	45. 第 1 症状レポートのサンプル	282
20. 「Display/Modify Sender Channel」パネル (2)	106	46. サンプル MQLOADEXIT	463
21. 「Create Channel」パネル	107	47. MQSeries SNAX セットアップ・ファイルのサンプル	475
22. 「Create Sender Channel」パネル	108	48. MQSeries SNA セットアップ・ファイル (ICE 用) のサンプル	478
23. 「Create Receiver Channel」パネル	108		
24. 「Create Server Connection Channel」パネル	109		
25. 「Create Cluster Sender Channel」パネル	109		
26. 「Create Cluster Receiver Channel」パネル	110		

表

1. MQSeries の実行可能プログラム	30	17. コマンド・サーバーの管理用のコマンド	413
2. 使用できる ISO CCSID のリスト	66	18. キューの管理用のコマンド	414
3. MQI 呼び出しに必要なセキュリティー 許可	194	19. プロセスの管理用のコマンド	414
4. MQSC コマンドと必要なセキュリティー 許可	198	20. チャンネルの管理用のコマンド	415
5. PCF コマンドと必要なセキュリティー 許可	199	21. その他の制御コマンド	415
6. キュー・オーバーフローとメッセージ・ オーバーフローの比較	303	22. Java 言語インターフェース	420
7. 重要な監査データベース・ファイル	309	23. MQSeries for Compaq NSK によってサ ポートされる MQSC	421
8. 重要なプロセスで使用される保護方式	310	24. サポートされる言語および環境の要約	443
9. dspmqaut コマンドからのセキュリティー 権限	350	25. MQI ライブラリーの正しいバージョン の使用	445
10. 種々のオブジェクト・タイプについての 許可の指定	385	26. MQSeries for Compaq NSK でサポート されるユーザー出口	459
11. アップグレード後に削除できるファイル の例	396	27. MQSeries for Compaq NSK がサポート するイベント・タイプ	497
12. キューのシステムおよびデフォルト・オ ブジェクト	405	28. イベント・メッセージの MQMD 構造	499
13. チャンネルのシステムおよびデフォルト・ オブジェクト	406	29. イベント・ヘッダー構造 (MQCFH)	500
14. 名前リストのシステムおよびデフォル ト・オブジェクト	407	30. PCF メッセージ記述子	501
15. プロセスのシステムおよびデフォルト・ オブジェクト	407	31. PCF ヘッダー	502
16. キュー・マネージャーの管理用のコマン ド	413	32. PCF ストリング・パラメーター	502
		33. PCF 整数リスト	503
		34. PCF 整数	503
		35. PCF ストリング・リスト	503
		36. MQSeries for Compaq NSK がサポート する PCF コマンド	504

本書について

MQSeries for Compaq NonStop Kernel バージョン 5 リリース 1 (本書では文脈に応じて MQSeries for Compaq NSK または MQSeries と呼ばれることもある) は、MQSeries プロダクト・ファミリーの一部です。これらのプロダクトは、アプリケーション・プログラムがメッセージ・キューを使用して相互に通信できるようにするアプリケーション・プログラミング・サービスを提供します。この形式の通信は、コマーシャル・メッセージングと呼ばれています。関係するアプリケーションは、広範囲のタイプのマシンおよびオペレーティング・システムの種々のノードに存在できます。アプリケーションは、メッセージ・キューイング・インターフェースまたは MQI と呼ばれる共通アプリケーション・プログラミング・インターフェースを使用するので、あるプラットフォームで開発されたプログラムは容易に別のプラットフォームに移すことができます。

本書は、MQSeries for Compaq NSK バージョン 5 リリース 1、および Compaq NSK 環境でのコマーシャル・メッセージングをサポートするために MQSeries for Compaq NSK が提供するサービスについて、システム管理の面から説明します。その中には、アプリケーションがメッセージを受け取るために使用するキューの管理や、必要とするキューにアプリケーションが確実にアクセスできるようにすることなども含まれています。

本書の対象読者

本書の主な対象読者は、システム管理者、および MQSeries の構成タスクと管理タスクを管理するシステム・プログラマーです。本書はまた、MQSeries の管理作業について理解しておく必要のあるアプリケーション・プログラマーにとっても役立つものです。

本書を理解する上での前提事項

本書を使用するためには、本書に記載されている Compaq NSK オペレーティング・システム、および関連したユーティリティについてよく理解しておく必要があります。メッセージ・キューイング・プロダクトを使用した経験はなくても構いませんが、メッセージ・キューイングの基本概念は理解しておく必要があります。

本書について

本書の使い方

本書の主部は、次のことを扱っています。

- MQSeries の紹介
- MQSeries for Compaq NonStop Kernel システムの日常の管理の説明。ローカルおよびリモート MQSeries オブジェクトの管理、セキュリティー、トランザクション・サポート、および問題判別について検討します。

インターネット上の MQSeries についての情報

MQSeries の URL

MQSeries プロダクト・ファミリーの URL は、次のとおりです。

<http://www.ibm.com/jp/software/mqseries/>

MQSeries for Compaq NSK V5.1 の新機能

MQSeries for Compaq NSK V5.1 システム管理ガイド のこの版では、以下の新機能について説明します。

パフォーマンスの拡張

新規キュー・サーバー・プロセス

(キュー・マネージャーに) 導入された新規キュー・サーバー・プロセスは、1 つまたは複数のローカル・キューにメッセージ・ストレージを提供し、それらのキューに対するすべての GET および PUT 操作を管理します。これは、非持続メッセージングを効率的にインプリメントし、バージョン 5.1 の新規メッセージング機能をサポートします。詳細については、27ページの『第2章 MQSeries for Compaq NSK V5.1 のアーキテクチャー』を参照してください。

状況サーバー操作の変更

チャンネル状況へのファイル・ベースのアプローチは、状況サーバーに置き換えられました。これは、ローカル・キューでないこれらのオブジェクトの状況情報をサポートし、チャンネル状況情報に効率的にアクセスできるようにします。詳細については、27ページの『第2章 MQSeries for Compaq NSK V5.1 のアーキテクチャー』を参照してください。

非持続メッセージ

非持続メッセージによるパフォーマンス向上を利用できるようになりました。

トラステッド・アプリケーションのための FASTPATH バインディング・サポート アプリケーションが対応していれば、FASTPATH バインディングを用いてキュー・マネージャーに接続して、パフォーマンスを大幅に向上させることができます。FASTPATH アプリケーションは特定の点で制限されており、この形式のバインディングではキュー・マネージャーの重大な内部データの保護が弱くなるため、正しく動作する必要があります。詳細については、437ページの『付録I. アプリケーションの構築および実行』を参照してください。

持続メッセージのディスク・ストレージの改善

新規キュー・サーバー・アーキテクチャーの一部として、ディスク上の持続メッセージのストレージが変更され、すべてのサイズのメッセージ

でパフォーマンスが拡張されました。キュー・ファイル用の代替キー・ファイルは必要なく、非常に大きなメッセージ用に、メッセージのストレージの効率を (最大で 100 MB のサイズまで) 最大化する、新規タイプのディスク・ストレージが導入されています。

アップグレードされた MQSeries 機能

MQSeries キュー・マネージャー・クラスター

複数の MQSeries キュー・マネージャーを結合することによって、キュー・マネージャーのクラスターをつくることができます。同じクラスター内では、どのキュー・マネージャーも、別のキュー・マネージャーに属するキューを使用することができます。キュー・マネージャーが同じクラスター内の他のキュー・マネージャーにメッセージを送る場合、特別なチャンネル定義やリモート・キュー定義、各宛先用の伝送キューの定義はいずれも必要ありません。MQSeries クラスターの主な利点を次に挙げます。

- システム管理タスクの減少
- 可用性の向上
- ワークロード・バランシング

この機能の詳細については、MQSeries キュー・マネージャー・クラスター を参照してください。

MQSeries 管理インターフェース (MQSeries Administration Interface (MQAI))

MQSeries for Compaq NSK V5.1 は、MQSeries 管理インターフェース (MQAI) をサポートするようになりました。これは、MQSeries を構成するための PCF メッセージの使用を単純化するプログラミング・インターフェースです。コマンドの完全な説明を含めた MQAI の詳細については、MQSeries® 管理インターフェースのプログラミングの手引き および解説書 を参照してください。

拡張された MQI サポート

MQSeries for Compaq NSK V5.1 は、他のプラットフォーム上のバージョン 5.0 およびバージョン 5.1 MQSeries リリースで提供されている拡張メッセージ機能をサポートしています。これには、配布リスト処理、参照メッセージ、セグメント化メッセージ、および他の多くの新規オプションが含まれます。詳細については、MQSeries アプリケーション・プログラミング・ガイド と MQSeries アプリケーション・プログラミング・リファレンス を参照してください。

メッセージおよびメッセージ・キューのサイズの増加

メッセージの最大サイズは、100 MB です。メッセージ・キューの最大容量は、4 GB です。

デフォルト・オブジェクトの自動作成

crtmqm コマンドを使用してキュー・マネージャーを作成すると、システム・デフォルト・オブジェクトが自動的に作成されます。

キュー・マネージャーの制御同期シャットダウン

endmqm コマンドに新しいオプションが追加されました。このオプションを指定すると、キュー・マネージャーを同期制御シャットダウンすることができます。

Java™ のサポート

MQSeries for Compaq NSK V5.1 では Java コンパイラーを使用して作業することができ、アプリケーションを Java でコード化することができます。詳細については、419ページの『付録F. MQSeries および Compaq NonStop Server for Java』を参照してください。

OSS アプリケーション・サポート

MQSeries for Compaq NSK V5.1 では C、C++、COBOL および Java を使用する NSK OSS アプリケーションを使用して作業することができます。

Web 管理

MQSeries for Compaq NSK V5.1 では、Microsoft® Windows NT® システムで HTML ブラウザー (Netscape Navigator や Microsoft Internet Explorer など) を使用して、以下のタスクを実行できます。

- MQSeries 管理者としてログオンする。
- キュー・マネージャーを選択し、キュー・マネージャーに対して MQSC コマンドを発行する。
- MQSC スクリプトを作成、編集および削除する。

改善されたユーザー出口メカニズム

MQSeries 用のユーザー出口をバインディングして構成するメカニズムが大幅に改善され、標準に近いメカニズムと、すべての出口で共通のメカニズムが提供されます。詳細については、459ページの『付録L. ユーザー出口』を参照してください。

相互通信

TCP/IP

MQSeries for Compaq NSK V5.1 では、複数の Guardian TCP/IP サーバー・プロセスを 1 つのキュー・マネージャーが使用できるようになりました。これにより、ネットワーク・ハードウェアにおけるロード・บาลancingの構成が改善され、キュー・マネージャーおよびアプリケーション用のネットワーク接続における冗長度が実現します。詳細については、471ページの『付録M. 通信のセットアップ』を参照してください。

SNA

MQSeries for Compaq NSK V5.1 には、SNA トランスポート・プロトコル用のチャンネルのリモート開始を管理して制御するための改善されたメカニズムがあります。この新規メカニズムは、PATHWAY の下で実行されて SNAX および InSession ICE 製品の両方でサポートされる、リスナー・プロセスを使用します。標準でないチャンネル属性 AUTOSTART はもはやサポートされていません。詳細については、471ページの『付録M. 通信のセットアップ』を参照してください。

チャンネル

チャンネルは、ハートビート と、作業単位の外部に非持続メッセージを送送する機能をサポートするようになり、パフォーマンスが向上しています。

MQSeries for Compaq NSK V5.1 では、他のキュー・マネージャーまたはクライアントからリモートに開始したチャンネル用に、オプションでチャンネルの自動定義がサポートされています。

Compaq NSK 固有の使いやすさ

runmqsc に組み込まれている Compaq NSK Fix Command

runmqsc に、Compaq NSK Fix Command 機能が組み込まれ、MQSC コマンドの再呼び出しと編集が可能になりました。詳細については、39ページの『Compaq NSK Fix Command の使用』を参照してください。

拡張された altmqfls ユーティリティー

ユーティリティー **altmqfls** は実質的に変更され、メッセージのストレージ・オプションを詳細に管理できるようになりました。詳細については、283ページの『第15章 スケーラビリティおよびパフォーマンス』と 356ページの『dspmqfls (MQSeries ファイル属性の表示)』を参照してください。

MQMC パネル

キュー・マネージャー PATHWAY 環境の一部として提供された MQMC 管理パネルは、拡張された MQSeries 機能をサポートするようにこのリリースでアップグレードされました。

第1部 手引き

第1章 概要	7
MQSeries とメッセージ・キューイング	7
時間に依存しないアプリケーション	8
メッセージ・ドリブン処理	8
メッセージとキュー	8
メッセージとは	8
メッセージ長	9
キューとは	10
アプリケーションによるキューの送信 / 受信方法	10
事前定義キューと動的キュー	10
キューからメッセージを取り出す	11
オブジェクト	11
オブジェクト名	12
オブジェクトの管理	12
オブジェクトの属性	12
MQSeries キュー・マネージャー	13
MQI 呼び出し	14
MQSeries キュー	14
キュー・オブジェクトの使用	15
MQSeries で使用されるローカル・キュー	16
プロセス定義	18
チャンネル	19
クラスター	19
名前リスト	20
システム・デフォルト・オブジェクト	20
管理	21
ローカル管理およびリモート管理	21
クライアントおよびサーバー	22
クライアント / サーバー環境での MQI アプリケーション	22
キュー・マネージャー機能の拡張	23
ユーザー出口	23
インストール可能なサービス	23
セキュリティ	24
オブジェクト権限マネージャー (OAM) 機能	24
トランザクション・サポート	25
パフォーマンスのチューニング、信頼性、スケラビリティ、およびサイジング	26

第2章 MQSeries for Compaq NSK V5.1 のアーキテクチャー	27
キュー・マネージャー・プロセスの概要	27
プロダクト・パッケージ	30
実行可能プログラム	30
バージョン 5.1 とバージョン 2.2.0.1 のアーキテクチャーの比較	31
キュー・マネージャー - 機能の概観	32
キュー・マネージャー・プロセス・モデル	34
MQSeries のファイルとサブボリューム	34
第3章 MQSeries コマンド・セットの使用	37
制御コマンドを使用した管理の実行	37
制御コマンドの使用	38
MQSC コマンドを使用した管理の実行	38
MQSC コマンドの実行	38
Compaq NSK Fix Command の使用	39
PCF コマンドを使用した管理の実行	39
MQSC および PCF での属性	40
エスケープ PCF	40
TS/MP (PATHWAY) 管理	41
TCP/IP リスナーの指定および制御	41
コマンド・サーバーの制御	41
チャンネル・イニシエーターの指定および制御	42
チャンネル・イニシエーターのデフォルトの開始キューの変更	42
トリガー・モニターの指定および制御	42
CPU 間でのプロセスの分配の指定	43
新しい状況サーバー・プロセスの追加	45
新しいキュー・サーバー・プロセスの追加	45
MQM モニター・パネルを最新表示する頻度の指定	45
キュー・マネージャーの PATHWAY 構成	46
PATHWAY サーバー・クラスのパラメーターの変更	54
MQSeries PATHWAY へのユーザー定義サーバー・クラスの追加	55
第4章 キュー・マネージャーの管理	57
始めに	57

キュー・マネージャー作成のガイドライン	58	キュー・マネージャー・データ・ファイル・サブボリューム	80
キュー・マネージャーの作成後の構成ファイルのバックアップ	61	キュー・マネージャー・エラー・ログ・サブボリューム	81
キュー・マネージャーのプロパティの変更	61	キュー・マネージャー・メッセージ・キュー・サブボリューム	81
キュー・マネージャーのホーム・ボリューム	62	キュー・マネージャー・チャンネル同期サブボリューム	82
キュー・マネージャーごとの EC プロセスの数	62	オブジェクト名の変換	83
システム・ロード・バランシング	63	キュー・マネージャーでの作業	83
キュー・マネージャーのホーム端末	63	デフォルト・キュー・マネージャーの作成	83
キュー・マネージャーの PATHMON プロセス名	64	MQSeries プリンシパルの作成	85
キュー・マネージャーの CCSID	65	RDF 使用可能になっているキュー・マネージャーでの cleanrdf の実行	85
EBCDIC データ変換の制御	65	キュー・マネージャーの開始	86
キュー・マネージャーの EMS コレクター	67	デフォルトおよびシステム・オブジェクトの復元	86
キュー・マネージャー内の各 EC によって作動可能な状態に保持されるエージェントのプール	67	オブジェクト・ファイルの表示	87
アイドル状態のエージェントとプロセスの再利用の最大数	68	キュー・マネージャーの停止	87
キュー・マネージャー・プロセスのプロセス優先順位	68	問題がある場合	87
キュー・マネージャーのチャンネルの最大数	68	即時および優先キュー・マネージャー・シャットダウン	87
キュー・マネージャーのアクティブ・チャンネルの最大数	69	キュー・マネージャーの再始動	88
MQSeries で使用する Guardian セグメント ID	69	既存のキュー・マネージャーのデフォルト・キュー・マネージャーへの指定	89
デフォルト TCP/IP ポート	70	キュー・マネージャーの削除	89
キュー・マネージャーが listen する TCP/IP ポート	70	メッセージ・キュー管理 (MQM) 機能の使用	90
キュー・マネージャーで使用する TCP/IP プロセス	71	「Queue Manager Menu」の使用	92
デフォルト以外の TCP/IP プロセスやポートのための MQS-TCPLISnn サーバー・クラスの再構成	71	キュー・マネージャーの属性の更新	93
スワップ・スペースの割り振り	72	MQSeries オブジェクトのトレース	93
デフォルトの状況サーバー名	73	「Queues」メニューの使用	95
デフォルト以外の状況サーバーの追加と除去	74	キューの作成	97
デフォルトのキュー・サーバー名	76	キューのコピー	100
デフォルト以外のキュー・サーバーの追加と除去	77	キューの変更	101
ボリューム構造	79	キューの削除	102
キュー・マネージャー FFST サブボリューム	80	キューのモニター	102
		「Channels」メニューの使用	103
		チャンネルの変更	105
		チャンネル定義の作成	107
		チャンネルのモニター	110
		チャンネルの削除	111
		チャンネル状況の表示	111
		チャンネルの開始および停止	112
		メッセージ・シーケンス番号 (MSN) のリセット	113
		チャンネルの解決	114

チャンネルのコピー	115	モデル・キューの取り扱い	145
第5章 ローカル MQSeries オブジェクトの管理	117	モデル・キューを定義する	145
MQI を使用したアプリケーション・プログラムのサポート	117	モデル・キューに関してその他のコマンドを使用する	146
MQSC コマンドを使用してローカル管理タスクを実行する	118	トリガー操作のためのオブジェクトの管理	146
開始する前に	119	トリガー操作のためにアプリケーション・キューを定義する	146
MQSeries オブジェクト名	119	開始キューを定義する	148
MQSC コマンドでの大文字小文字の区別	120	プロセス定義を作成する	148
MQSC 機能を対話式で使用する	120	プロセス定義を表示する	149
MQSC からのフィードバック	121	第6章 管理用タスクの自動化	151
対話式での MQSC への入力を終了する	122	PCF コマンド	151
キュー・マネージャーの属性を表示する	122	MQSC および PCF での属性	152
デフォルトではないキュー・マネージャーを使用する	123	エスケープ PCF	152
キュー・マネージャーの属性の更新	124	MQAI を使用して PCF の使い方を単純化する	153
テキスト・ファイルから MQSC コマンドを実行する	124	リモート管理でコマンド・サーバーを管理する	154
MQSC コマンド・ファイルの使用	126	コマンド・サーバーを開始する	154
MQSC レポートの使用	126	コマンド・サーバーの状況を表示する	154
システムに提供された MQSC コマンド・ファイルを実行する	128	コマンド・サーバーを停止する	155
runmqsc を使用してコマンドを確認する	128	第7章 リモート MQSeries オブジェクトの管理	157
MQSC で起こった問題の解決	129	チャンネル、クラスターおよびリモート・キューイング	157
ローカル・キューの取り扱い	131	クラスターを使用するリモート管理	159
ローカル・キューを定義する	131	リモート・キュー・マネージャーの管理	160
キューの物理ファイル・サイズの変更	132	リモート管理のためのキュー・マネージャーを作成する	160
送達不能キューを定義する	132	リモート管理のためにチャンネルおよび伝送キューを作成する	161
デフォルト・オブジェクトの属性を表示する	133	チャンネルと伝送キューを定義する	162
ローカル・キュー定義をコピーする	134	チャンネルを開始する	164
ローカル・キューの属性を変更する	135	チャンネルの自動定義	165
ローカル・キューのボリュームを変更する	136	リモートから MQSC コマンドを発行する	165
ローカル・キューのオプションを変更する	136	MVS/ESA でのキュー・マネージャーの取り扱い	166
状況サーバーおよびキュー・サーバーにオブジェクトを再割り当てする	137	リモート・キューイングについての勧告	167
ローカル・キューをクリアする	138	リモートからの MQSC の使用に問題がある場合	167
ローカル・キューを削除する	138	リモート・キューのローカル定義の作成	168
キューをブラウズする	138	リモート・キューのローカル定義の働きについて理解する	168
別名キューの取り扱い	142	例	168
別名キューを定義する	143		
キュー別名に関してその他のコマンドを使用する	144		

処理内容	168	オブジェクト権限マネージャー (OAM) のガ イドライン	188
リモート・キューにメッセージを書き込む代 替方法	170	ユーザー ID	188
リモート・キューに関してその他のコマンド を使用する	170	キュー・マネージャー・ボリューム	189
伝送キューを作成する	171	キュー	189
デフォルト伝送キュー	171	代替ユーザー権限	189
リモート・キュー定義を別名として使用する	172	コンテキスト権限	190
キュー・マネージャー別名	172	リモート・セキュリティーに関する考慮事 項	191
応答先キュー別名付け	173	チャンネル・コマンド・セキュリティー	192
データ変換	173	PCF コマンド	192
キュー・マネージャーがメッセージを組み 込み形式に変換できない場合	173	MQSC チャンネル・コマンド	192
ファイル CCSID	174	許可指定テーブルについての理解	193
デフォルトのデータ変換	174	MQI 許可	193
ユーザー定義形式でのメッセージの変換	175	管理許可	197
キュー・マネージャー CCSID の変更	175	エスケープ PCF 中の MQSC コマンド に関する許可	198
第8章 MQSeries オブジェクトの保護	177	PCF コマンドについての許可	199
MQSeries リソースを保護する理由	177	許可ファイルの理解	201
MQM ユーザー・グループのユーザー ID を 理解する	178	プリンシパル・データベース	201
その他の情報を表示する	179	OAM データベース	201
オブジェクト権限マネージャー (OAM) を理 解する	179	クラス許可レコード	204
OAM の機能	180	全クラス許可レコード	204
ユーザー・グループを介してアクセスを管 理する	180	第9章 MQSeries 送達不能キュー・ハンドラ ー	205
ユーザーが複数のユーザー・グループ に属している場合	180	DLQ ハンドラーの呼び出し	205
グループ・セットと 1 次グループ	181	DLQ ハンドラーのルール・テーブル	206
OAM でリソースを保護する	181	制御データ	207
許可にグループを使用する	182	ルール (パターンと処理)	209
オブジェクト権限マネージャー (OAM) を 使用不可にする	183	パターン・マッチング・キーワード	209
オブジェクト権限マネージャー (OAM) コマ ンドを使用する	184	アクション・キーワード	211
OAM コマンドの使用時に指定するもの	184	ルール・テーブルのルール	213
許可リスト	184	ルール・テーブルの処理方法	215
altmqusr コマンドを使用する	185	すべての DLQ メッセージを確実に処理す る	216
dspmqusr コマンドを使用する	186	DLQ ハンドラーのルール・テーブルの例	217
setmqaut コマンドを使用する	186	第10章 インストールメンテーションおよび EMS イベント	221
権限コマンドとインストール可能なサ ービス	187	MQSeries インストールメンテーション・イベ ント	221
アクセス許可	187	イベントのタイプ	222
権限の表示コマンド	188	イベント・キューによるイベント通知	223
		トリガー・イベント・キューを使用する	223

インストゥルメンテーション・イベントの有効化	223
イベント・メッセージ	224
Event Management Service (EMS) イベント	224
MQSeries for Compaq NSK に付属する EMS テンプレート・ファイル	224
MQSeries EMS イベント・テンプレートの統合	225
PARAM MQEMSEVENTS の定義	228
代替コレクターの使用	229
MQSeries EMS イベントを処理するためのプログラムの作成	229
第11章 トランザクション・サポートとメッセージングについての理解	231
NonStop TM/MP (トランザクション・マネージャー) の使用	231
同期点の制限	232
持続メッセージでの非同期点操作	233
非持続メッセージでの同期操作	234
TM/MP および MQSeries for Compaq NSK の構成要件	234
モニター	234
監査証跡のサイズ	235
リソース・マネージャーの構成	235
トラブルシューティング	235
第12章 リカバリーおよび再始動	237
フォールト・トレランスとリカバリー	238
MQSeries のバックアップと復元	238
MQSeries のバックアップ	238
MQSeries の復元	239
状況サーバーとキュー・サーバーのリカバリーおよび再始動	239
RDF を使用した災害時回復	240
第13章 構成ファイル	243
構成ファイルとは	243
MQSeries 構成ファイル (MQSINI)	244
MQSeries 構成ファイルの内容	244
キュー・マネージャー構成ファイル (QMINI)	245
キュー・マネージャー構成ファイルの内容	246
キュー・マネージャー構成ファイルの例	249
構成ファイルを編集する	254
構成ファイルへの変更のインプリメント	255
構成ファイルに関する推奨事項	255

第14章 問題判別	257
予備的な検査の作成	257
以前に MQSeries は正常に実行されたか	257
エラー・メッセージが出されたか	258
問題について説明した戻りコードが戻されていないか	258
問題を再現できるか	258
最後に正常に実行された後、変更が行われたか	259
アプリケーションは以前に正常に実行されたか	259
アプリケーションが以前に正常に実行されなかった場合	260
問題がネットワークの特定部分に影響するか	261
1 日の特定の時刻に問題が起こるか	262
問題は断続的に起こるか	262
サービス更新を適用したか	262
共通プログラミング・エラー	263
コマンドに関する問題	263
次に何を行うか	263
不正な出力を得たか	264
PCF コマンドからの応答を受け取れなかったか	264
キューの一部が障害を起こしているか	266
問題はリモート・キューにのみ影響するか	267
アプリケーションまたは MQSeries for Compaq NSK の実行速度は低速ですか	267
アプリケーション設計の考慮事項	269
メッセージ長の影響	269
特定のメッセージの検索	269
長さが異なるメッセージを含んでいるキュー	269
同期点の頻度	270
MQPUT1 呼び出しの使用	270
不正出力	270
キューにメッセージが現れない場合	270
メッセージが予期しない情報または破壊された情報を含んでいる場合	272
分散キューを使用している場合の不正出力の問題	273
エラー・ログ	275
ログ・ファイル	276
早期エラー	276
オペレーター・メッセージ	277

MQERRLG ファイル内の EC 番号の複 合	277	FASTPATH バインディング・アプリケーシ ョン・プログラム	295
エラー・ログの例	277	バックグラウンド	295
EMS イベント	279	MQI オーバーヘッドを削減する	296
送達不能キュー	279	FASTPATH バインディングの使用可能化	296
構成ファイルと問題判別	279	FASTPATH バインディングの使用時の制 限	297
MQSeries トレースの使用	280		
トレース・ファイル名	280		
トレース・データのサンプル	280		
第 1 障害サポート・テクノロジー™ (FFST)	281		
FFST の読み方	281		
第15章 スケーラビリティおよびパフォー マンス	283	第16章 データ保全性と可用性	299
概要	283	データ保全性	299
パフォーマンスおよびスケーラビリティ を向上させる新規アプリケーションを設計 する	283	可用性	300
共用リソースの使用を最小限にしたり、使 用しなくてもすむように設計する	284	持続データと非持続データ	302
パフォーマンス・チューニングは本質的に 対話式	284	持続メッセージ	302
持続メッセージ	285	非持続メッセージ	305
非持続メッセージ	285	データベースの一貫性	305
非持続メッセージおよびチャネル	286	内部データベースの一貫性	306
キュー・サーバーおよびキュー・ファイル	287	外部データベースの一貫性	306
持続メッセージのストレージ	287	OpenTMF	307
非持続メッセージのストレージ	288	NonStop Tuxedo	308
キュー・サーバーの CPU 分散	288	インターリーブド・アプリケーション・ト ランザクション	308
altmqfls を使用してキュー・サーバーにキ ューを再割り当てる	288	MQSeries の重要なデータベース・ファイ ル	309
クラスター伝送キュー:		重要なプロセス	310
SYSTEM.CLUSTER.TRANSMIT.QUEUE	289	クラスター	318
altmqfls を使用してキュー・ファイルの配 置を変更する	289	可用性に関する構成上の考慮事項	318
キュー・ファイルの区画化	290	データ保全性に関する構成上の考慮事項	319
メッセージ・オーバーフロー・ファイル	290		
ブラウザ中にメッセージをバッファァに入 れる	291		
その他のキュー・サーバー・オプション	292		
始動時のロード	292		
キャッシュ内にロックする	292		
チェックポイント NPM	293		
測定カウンター	293		
CPU の割り当て	293		

第1章 概要

この章では、管理者の観点から MQSeries for Compaq NonStop Kernel バージョン 5.1 (MQSeries for Compaq NSK V5.1) を紹介し、MQSeries およびメッセージングの基本概念について説明します。この章には、次の節があります。

- 『MQSeries とメッセージ・キューイング』
- 8ページの『メッセージとキュー』
- 11ページの『オブジェクト』
- 14ページの『MQSeries キュー』
- 18ページの『プロセス定義』
- 19ページの『チャンネル』
- 20ページの『システム・デフォルト・オブジェクト』
- 21ページの『管理』
- 22ページの『クライアントおよびサーバー』
- 23ページの『キュー・マネージャー機能の拡張』
- 24ページの『セキュリティ』
- 25ページの『トランザクション・サポート』
- 26ページの『パフォーマンスのチューニング、信頼性、スケーラビリティ、およびサイジング』

MQSeries とメッセージ・キューイング

MQSeries により、アプリケーション・プログラムは、メッセージ・キューイングを使用して、メッセージ・ドリブン処理を行うことができます。適切なメッセージ・キューイング・ソフトウェア・プロダクトを使用すれば、プラットフォームが異なっても、アプリケーション・プログラム相互間で通信することができます。たとえば、Compaq NSK アプリケーションと MVS/ESA™ アプリケーションは、それぞれ MQSeries for Compaq NSK と MQSeries for OS/390® を介して通信を行うことができます。アプリケーションは、基礎となる通信機構からは保護されています。

MQSeries プロダクトでは、アプリケーションがどのプラットフォームで実行される場合でも、共通アプリケーション・プログラミング・インターフェース (メッセージ・キュー・インターフェース: MQI) が実現されます。このため、アプリケーションを、あるプラットフォームから別のプラットフォームに移植するのが容易になります。

MQSeries とメッセージ・キューイング

MQI については、*MQSeries アプリケーション・プログラミング・リファレンス* で詳しく説明されています。

時間に依存しないアプリケーション

メッセージ・キューイングを使用すれば、送信側プログラムと受信側プログラム間のメッセージ交換は、時間に依存しなくなります。つまり、送信側と受信側のアプリケーションの相互の結び付きがなくなり、送信側は、受信側からの受信の通知を待たずに処理を続行することができます。そればかりか、ターゲット・アプリケーションが、メッセージの送信時に、実行中である必要もありません。ターゲット・アプリケーションは、始動後にメッセージを取り出すことができます。

メッセージ・ドリブン処理

トリガー操作 と呼ばれるメカニズムを使用して、メッセージがキューに到着したときに自動的にアプリケーションを開始できます。必要に応じて、メッセージの処理が完了したときに、アプリケーションを停止させることもできます。

メッセージとキュー

メッセージとキューは、メッセージ・キューイング・システムの基本コンポーネントです。

メッセージとは

メッセージ とは、そのメッセージを使用するアプリケーションにとってなんらかの意味を持つ、何バイトかの長さのストリングです。メッセージは、ある 1 つのアプリケーションから別のアプリケーションに（または、同じアプリケーションの異なる部分に）情報を転送するために使用されます。アプリケーションが実行されるのは、同じプラットフォーム上でも、異なるプラットフォーム上でも構いません。

MQSeries のメッセージは次の 2 つの部分で構成されます。

- アプリケーション・データ

アプリケーション・データの内容と構造は、それらのデータを使用するアプリケーション・プログラムによって定義されます。

- メッセージ記述子

メッセージ記述子はそのメッセージを識別するものですが、この中には、そのメッセージのタイプや送信側のアプリケーションによって与えられた優先順位などの他の制御情報も格納されています。

メッセージ記述子の形式は、MQSeries によって定義されます。メッセージ記述子の詳細については、MQSeries アプリケーション・プログラミング・リファレンス を参照してください。

メッセージ記述子の形式は、MQSeries によって定義されます。メッセージ記述子の詳細については、MQSeries アプリケーション・プログラミング・リファレンス を参照してください。

メッセージには 2 つのタイプ (持続メッセージと非持続メッセージ) があります。これらは次の点で異なります。

- 持続メッセージは、キュー・マネージャーを再始動したときに持続します。非持続メッセージは、キュー・マネージャーを再始動したときに持続しません。
- 通常、非持続メッセージはディスクに書き込まれないため、キューに追加されたりキューから除去されたりするリソースが持続メッセージよりも少なくなります。
- 何らかの障害条件において、非持続メッセージは持続メッセージより信頼性が低くなります。
- 持続メッセージは一時動的キューに置くことができません。

メッセージ長

MQSeries では、最大メッセージ長は 100 MB です (1 MB は 1 048 576 バイトとします)。メッセージ長は次のものによって制限されます。

- 受信側のキュー用に定義した最大メッセージ長
- キュー・マネージャー用に定義した最大メッセージ長
- 送信側のアプリケーションまたは受信側のアプリケーションで定義した最大メッセージ長
- メッセージ用として使用可能なストレージ量

1 つのアプリケーションが必要とするすべての情報を送るには、複数のメッセージが必要な場合があります。

最大メッセージ長を大きくすると、良くない影響が生じる可能性があります。また、メッセージがキューやキュー・マネージャーにとって大きくなりすぎる可能性もあります。このような場合は、メッセージをセグメントに分割して、論理メッセージにグループ化することができます。メッセージの論理グループを行えば、類似のメッセージをグループ化し、メッセージの順序を保証することができます。メッセージのセグメント化とグループ化について詳しくは、MQSeries アプリケーション・プログラミング・ガイド を参照してください。

キューとは

キューとは、メッセージを保管するためのデータ構造体です。メッセージは、アプリケーション・プログラムまたはキュー・マネージャーによって、通常の操作の一部としてキューに書き込まれます。

キューはすべて、キュー・マネージャーに属しています。キュー・マネージャーは、所有するキューを管理し、受信したすべてのメッセージを適切なキューに格納します。

キューの最大サイズは、4 GB です。キューに必要な記憶容量の計画についての情報は、*MQSeries Planning Guide* を参照してください。また、次の Web サイトには、プラットフォーム別のパフォーマンス・レポートがあります。

<http://www.ibm.com/software/mqseries/txppacs/txpml.html>

アプリケーションによるキューの送信 / 受信方法

アプリケーションでは、*MQI* 呼び出しを使用してメッセージを送受信します。たとえば、キューにメッセージを書き込むために、アプリケーションでは以下の処理が行われます。

1. *MQI MQOPEN* 呼び出しを発行して必要なキューをオープンする。
2. *MQI MQPUT* 呼び出しを発行してキューにメッセージを書き込む。
3. 別のアプリケーションがそのキューからメッセージを取り出すときには、*MQI MQGET* 呼び出しを発行する。

MQI 呼び出しの詳細については、*MQSeries アプリケーション・プログラミング・リファレンス* を参照してください。

事前定義キューと動的キュー

キューは、その作成方法によって特性をもたせることができます。

- **事前定義キュー** は、管理者が該当のコマンド・セットを使用して作成するものです。たとえば、*MQSC* コマンド *DEFINE QLOCAL* は、事前定義ローカル・キューを作成します。事前定義キューは、永続キューであって、それらを使用するアプリケーションとは無関係に存在し、*MQSeries* が再始動しても存続します。
- **動的キュー** は、アプリケーションがモデル・キュー の名前を指定してオープン要求を出したときに作成されるものです。作成されたキューは、モデル・キューであるテンプレート・キュー定義に基づいています。モデル・キューは、*MQSC DEFINE QMODEL* を使用して作成できます。モデル・キューの属性 (たとえば、キューに保管できるメッセージの最大数) は、そのモデル・キューから作成される動的キューが継承します。

モデル・キューは、作成される動的キューが永続キューになるか一時キューになるかを指定する属性を持っています。永続キューは、アプリケーションやキュー・マネージャーが再始動しても存続しますが、一時キューは、再始動すると失われます。

キューからメッセージを取り出す

MQSeries では、アプリケーションに適切な許可が与えられている場合、次の取り出しアルゴリズムに従ってキューからメッセージを取り出すことができます。

- 先入れ先出し法 (FIFO)
- メッセージ記述子に定義されたメッセージ優先順位。同じ優先順位を持つメッセージは、FIFO 順に取り出されます。
- 特定のメッセージについてのプログラム要求。

アプリケーションからの MQGET 要求によって、使用される方式が決まります。

オブジェクト

本書で説明しているタスクの多くには、MQSeries オブジェクト の操作が含まれています。MQSeriesバージョン 5.1 の場合、オブジェクトのタイプには、キュー・マネージャー、キュー、プロセス定義、チャンネル、クラスター、名前リストがあります。

オブジェクトの操作または管理では、以下のことが行われます。

- キュー・マネージャーの開始および停止。
- アプリケーション用のオブジェクト、特にキューの作成。
- チャンネルを使用して、他の (リモート) システムにあるキュー・マネージャーへの通信パスを作成。これについては、*MQSeries 相互通信* で詳しく説明されています。
- キュー・マネージャーのクラスターを作成することによって、管理プロセス全体を簡易化する、またはワークロードのバランスをとる。

本書では、以下の各章で管理についての詳細情報を掲載しています。

- 37ページの『第3章 MQSeries コマンド・セットの使用』
- 57ページの『第4章 キュー・マネージャーの管理』
- 117ページの『第5章 ローカル MQSeries オブジェクトの管理』
- 157ページの『第7章 リモート MQSeries オブジェクトの管理』

オブジェクト

オブジェクト名

キュー・マネージャーの各インスタンスにはオブジェクト名があります。メッセージが送信されるターゲット・キュー・マネージャーを正しく識別するため、このオブジェクト名はキュー・マネージャーのネットワーク内で固有のものでなければなりません。

オブジェクト名は、1 つのキュー・マネージャーおよびオブジェクト・タイプ内において固有のものである必要があります。たとえば、同じ名前のついたキューとプロセスを持つことはできますが、同じ名前の 2 つのキューを持つことはできません。

オブジェクト名は最大 48 文字までです。ただし、チャンネルは例外で、最大 20 文字までです。名前についての詳細は、324ページの『名前の使用』を参照してください。

オブジェクトの管理

MQSeries には、オブジェクトの作成、変更、表示、および削除を行う機能が備わっています。これには次のものが含まれます。

- MQSC コマンド (MQSC)。これは、キーボードからの入力とファイルからの読み取りの両方が可能です。
- MQM (画面ベースのインターフェース)
- プログラム式コマンド形式 (PCF) のコマンド。プログラムで使用することができます。
- 制御コマンド。オペレーティング・システムのコマンド行から対話式に入力することができます。

詳細については、37ページの『第3章 MQSeries コマンド・セットの使用』を参照してください。

オブジェクトの属性

オブジェクトの特性は、そのオブジェクトの属性によって定義されます。あるオブジェクト属性は指定したり変更したりできますが、他のオブジェクト属性は表示することしかできません。たとえば、キューが収容できる最大メッセージ長は、*MaxMsgLength* 属性によって定義されます。このオブジェクト属性は、キューの作成時に指定できます。*DefinitionType* 属性は、キューが作成された方法を指定します。*DefinitionType* 属性は表示のみできます。

MQSeriesでは、オブジェクト属性を参照する方法には次の 2 とおりがあります。

- 属性の PCF 名 (たとえば、*MaxMsgLength*) を使用する方法属性の PCF 名は、その属性の正式の名前です。
- 属性の MQSC 名 (たとえば、MAXMSGL) を使用する方法

属性の正式の名前は、その属性の PCF 名です。MQSC 機能の使用は、本書の重要な部分を占めているので、本書の例では、与えられている属性について PCF 名よりも MQSC 名の方が多く使用されています。

MQSeries キュー・マネージャー

キュー・マネージャーは、アプリケーションにメッセージ・キューイング・サービスを提供します。キュー・マネージャーによって、次のことが確実に行われます。

- オブジェクトの属性は、受け取ったコマンドに応じて変更されます。
- 該当の条件が満たされたときに、トリガー・イベントやインスツルメンテーション・イベントなどの特殊イベントが生成されます。
- メッセージが、MQPUT 呼び出しを行ったアプリケーションの要求により、正しいキューに書き込まれます。正しいキューに入れられなかった場合には、アプリケーションに通知され、該当の理由コードが戻されます。

それぞれのキューは、1 つのキュー・マネージャーに属しており、そのキュー・マネージャーに対してローカル・キュー であるといえます。アプリケーションが接続されているキュー・マネージャーは、そのアプリケーションに対してローカル・キュー・マネージャーであるといえます。アプリケーションのローカル・キュー・マネージャーに属しているキューは、そのアプリケーションのためのローカル・キューです。

リモート・キュー とは、単に別の管理プログラムに属しているキューのことです。

リモート・キュー・マネージャー とは、ローカル・キュー・マネージャー以外の任意のキュー・マネージャーのことです。リモート・キュー・マネージャーは、ネットワーク内のリモート・マシン上にある場合と、ローカル・キュー・マネージャーと同じマシン上にある場合があります。

MQSeries では、1 つのマシン上で複数のキュー・マネージャーを使用することができます。

オブジェクト

MQI 呼び出し

キュー・マネージャー・オブジェクトはさまざまな MQI 呼び出しで使用できます。たとえば、オブジェクト属性について、MQI 呼び出しの MQINQ を使用して問い合わせることができます。

注: メッセージは、必ずキュー・オブジェクトに書き込まれ、キュー・マネージャー・オブジェクトには書き込まれません。メッセージをキュー・マネージャー・オブジェクトに書き込むことはできません。

MQSeries キュー

MQSeries for Compaq NSK に対してキューを定義するには、次のコマンドを使用します。

- MQSC DEFINE コマンド
- MQSeries for Compaq NSK のメッセージ・キュー管理 (MQM) 機能
- PCF コマンドの「キューの作成」
- MQAI コマンド

これらのコマンドは、キューのタイプおよびキューのオブジェクト属性を指定します。たとえば、ローカル・キューは、アプリケーションがそのキューを MQI 呼び出しで参照するときに指定するオブジェクト属性を持っています。オブジェクト属性には次のものがあります。

- アプリケーションがメッセージをキューから取り出せるかどうか (読み取り (GET) 可能)
- アプリケーションがメッセージをキューに書き込めるかどうか (書き込み (PUT) 可能)
- キューへのアクセスが、1 つのアプリケーション専用になるか、または複数のアプリケーションで共用されるか
- 同時にキューに保管できるメッセージの最大数 (キューの最大サイズ)
- キューに書き込むことのできる最大メッセージ長

詳細については、次の資料を参照してください。

- MQSC については、*MQSeries MQSC コマンド・リファレンス*
- MQM については、90ページの『メッセージ・キュー管理 (MQM) 機能の使用』
- PCF コマンドについては、*MQSeries プログラム式システム管理*

キュー・オブジェクトの使用

MQSeries には、さまざまなタイプのキュー・オブジェクトがあります。オブジェクトのそれぞれのタイプは、プロダクト・コマンドで操作することができ、それぞれ異なる方法で実際のキューと関連付けられます。

• ローカル・キュー・オブジェクト

ローカル・キュー・オブジェクトは、アプリケーションが接続されているキュー・マネージャーに属するローカル・キューを識別します。それぞれのキューが 1 つのキュー・マネージャーに属しているという意味では、すべてのキューがローカル・キューであり、そのキュー・マネージャーについては、それに属しているキューがローカル・キューです。

• リモート・キュー・オブジェクト

リモート・キュー・オブジェクトは、他のキュー・マネージャーに属するキューを識別します。このキューは、そのキュー・マネージャーのローカル・キューとして定義する必要があります。リモート・キュー・オブジェクトを定義したときに指定した情報により、ローカル・キュー・マネージャーはリモート・キュー・マネージャーを見つけることができ、それにより、そのリモート・キュー宛てのすべてのメッセージは、正しいキュー・マネージャーに渡されるようになります。

別のキュー・マネージャー上のキューにメッセージを送信するには、あらかじめ伝送キューと宛先のキュー・マネージャーとの間のチャンネルを定義しておく必要があります。ただし、複数のキュー・マネージャーをクラスターにグループ化してある場合は、その必要はありません。クラスターについての詳細は、159ページの『クラスターを使用するリモート管理』を参照してください。

• 別名キュー・オブジェクト

別名キュー・オブジェクトにより、アプリケーションは、MQI 呼び出しで間接的にキューを参照することにより、そのキューをアクセスすることができます。別名キュー名が MQI 呼び出しで使用されると、その名前は、実行時にローカル・キューまたはリモート・キューの名前に解決されます。これにより、アプリケーションを変更しなくても、単に別名キュー定義を変更して、別名によって決定される新しいキューの名前になるようにするだけで、アプリケーションが使用するキューを変更することができます。

別名キューは、キューではありませんが、他のキューへのアクセスに使用できるオブジェクトです。

• モデル・キュー・オブジェクト

モデル・キュー・オブジェクトは、動的キュー作成のテンプレートとして使用するキュー属性のセットを定義します。動的キューは、モデル・キューの

名前であるキュー名を指定した MQOPEN 要求をアプリケーションが出す時に、キュー・マネージャーによって作成されます。このようにして作成された動的キューはローカル・キューであり、その属性は、モデル・キュー定義から継承されます。動的キュー名は、アプリケーションで指定するか、あるいは、キュー・マネージャーがその名前を生成して、アプリケーションにそれを戻すことができます。

このようにして定義された動的キューは、プロダクトの再始動後に残存しない一時キューである場合もあれば、再始動後も残存する永続キューである場合もあります。

MQSeries で使用されるローカル・キュー

MQSeries は、さまざまなローカル・キューを、その操作に関連する特定の目的のために使用します。これらのキューは、MQSeries が使用する前に必ず 定義する必要があります。

アプリケーション・キュー: アプリケーションが (MQI を介して) 使用するキューのことをアプリケーション・キューと言います。このキューは、アプリケーションが接続されているキュー・マネージャー上のローカル・キューの場合もあれば、別のキュー・マネージャーが所有するリモート・キューの場合もあります。

アプリケーションは、メッセージを、ローカル・キューやリモート・キューに入れることができます。ただし、メッセージを読み取ることができるのは、ローカル・キューからのみです。

開始キュー: 開始キューは、トリガー操作に使用するキューです。キュー・マネージャーは、トリガー・イベントが発生すると、開始キューにトリガー・メッセージを書き込みます。トリガー・イベントとは、キュー・マネージャーによって検出される条件の論理的組み合わせのことです。たとえば、キュー上のメッセージの数が、あらかじめ定義されたキューのサイズに達したときにトリガー・イベントが生成される場合があります。このイベントが発生すると、キュー・マネージャーは指定の開始キューにトリガー・メッセージを入れることになります。このトリガー・メッセージは、開始キューをモニターする特殊アプリケーションであるトリガー・モニターによって取り出されます。次に、トリガー・モニターは、トリガー・メッセージに指定されているアプリケーション・プログラムを始動します。

キュー・マネージャーでこのトリガー操作を使用する場合は、少なくとも 1 つの開始キューを、そのキュー・マネージャー用に定義する必要があります。

146ページの『トリガー操作のためのオブジェクトの管理』を参照してください。トリガー操作の詳細については、*MQSeries* アプリケーション・プログラミング・ガイド を参照してください。

伝送キュー: 伝送キューは、リモートのキュー・マネージャー宛てのメッセージを一時的に保管するキューです。ローカル・キュー・マネージャーがメッセージを直接送信する各リモート・キュー・マネージャーごとに、少なくとも 1 つの伝送キューを定義する必要があります。これらのキューは、リモート管理にも使用されます。160ページの『リモート・キュー・マネージャーの管理』を参照してください。分散キューイングでの伝送キューの使用法の詳細については、*MQSeries* 相互通信 を参照してください。

クラスター伝送キュー: クラスター内のそれぞれのキュー・マネージャーには、`SYSTEM.CLUSTER.TRANSMIT.QUEUE` というクラスター伝送キューがあります。このキューはすべてのキュー・マネージャーに対してデフォルトで定義されます。

クラスター内のキュー・マネージャーは、同じクラスター内のすべてのキュー・マネージャーに、伝送キュー上のメッセージを送ることができます。

クラスター内のキュー・マネージャーは、同一クラスターには属さないキュー・マネージャーと通信することができます。そのためには、従来の分散キューイング環境での場合と同様に、キュー・マネージャーで、他のキュー・マネージャーに対するチャンネルと伝送キューを定義する必要があります。

ネーム・レゾリューション時には、クラスター伝送キューがデフォルト伝送キューより優先されます。クラスター外のキュー・マネージャーがリモート・キューにメッセージを書き込む場合、宛先のキュー・マネージャーと同名の伝送キューがなければ、デフォルトでは、デフォルト伝送キューが使用されます。

クラスター内のキュー・マネージャーの場合は、宛先のキュー・マネージャーが同じクラスターに属さない限り、デフォルトでは `SYSTEM.CLUSTER.TRANSMIT.QUEUE` が使用されます。

送達不能キュー: 送達不能キューは、正しい宛先に渡すことができないメッセージを保管します。このようなことが起きるのは、たとえば、宛先キューが満杯である場合です。システムに提供された送達不能キューは、`SYSTEM.DEAD.LETTER.QUEUE` と呼ばれます。このキューは、他のプラットフォームでは、未配布メッセージ・キューとも呼ばれています。

分散キューイングでは、アクティブな各キュー・マネージャーごとに 1 つの送達不能キューを定義する必要があります。

コマンド・キュー: SYSTEM.ADMIN.COMMAND.QUEUE というコマンド・キューは、許可を与えられたアプリケーションが処理のために MQSeries for Compaq NSK コマンドを送ることができる送り先のローカル・キューです。次に、これらのコマンドは、コマンド・サーバーと呼ばれる MQSeries コンポーネントによって取り出されます。コマンド・サーバーは、それらのコマンドを検査し、正しいものをキュー・マネージャーの処理用に渡し、該当の応答先キューに応答を戻します。

応答先キュー: あるアプリケーションが要求メッセージを送信した場合、そのメッセージを受信するアプリケーションは、送信側のアプリケーションに応答メッセージを送信することができます。この応答メッセージは、応答先キューと呼ばれるキューに書き込まれます。このキューは、通常は送信側のアプリケーションのローカル・キューです。応答先キューの名前は、送信側のアプリケーションによってメッセージ記述子の一部として指定されます。

イベント・キュー: MQSeries for Compaq NSK バージョン 5.1 は、MQI アプリケーションとは無関係にキュー・マネージャーをモニターするのに使用できるインスツルメンテーション・イベントをサポートします。インスツルメンテーション・イベントが生成される条件には、次のようなものがあります。

- 使用不能または存在しないキューに、アプリケーションがメッセージを書き込もうとした。
- キューが満杯になった。
- チャンネルが開始された。

インスツルメンテーション・イベントが発生すると、キュー・マネージャーはイベント・キューにイベント・メッセージを書き込みます。書き込まれたこのメッセージは、イベントが問題を提示すると、管理者に通知したり何らかの矯正処置を開始するモニター・アプリケーションにより、読み取られます。

注: トリガー・イベントは、同じ条件では発生しない、イベント・メッセージが生成されない、という 2 点で、インスツルメンテーション・イベントとは異なります。

インスツルメンテーション・イベントの詳細については、*MQSeries* プログラム式システム管理 を参照してください。

プロセス定義

プロセス定義オブジェクト は、MQSeries for Compaq NSK キュー・マネージャーでのトリガー・イベントに反応して開始されるアプリケーションを定義します。詳細については、16 ページの『開始キュー』を参照してください。

プロセス定義の属性には、アプリケーション ID、アプリケーション・タイプ、およびアプリケーション特有のデータがあります。

プロセス定義を作成するには、MQSC コマンド DEFINE PROCESS または PCF コマンド Create Process を使用します。

チャネル

チャネル とは、あるキュー・マネージャーと別のキュー・マネージャーを結ぶ通信パスを提供するオブジェクトのことです。チャネルは、あるキュー・マネージャーから別のキュー・マネージャーにメッセージを移すために、分散メッセージ・キューイングで使用されます。チャネルは、基礎をなす通信プロトコルからアプリケーションを保護します。複数のキュー・マネージャーが、同じプラットフォームまたは異なるプラットフォーム上に存在できます。キュー・マネージャーが相互に通信できるようにするには、メッセージを送信するキュー・マネージャーに 1 つのチャネル・オブジェクトを定義し、メッセージを受信するキュー・マネージャーに別のチャネル・オブジェクトを定義する必要があります。

MQSeries for Compaq NSK は、TCP/IP および SNA LU6.2 (SNAX または ICE) トランSPORT・プロトコルをサポートしています。

チャネルおよびその用法については、*MQSeries 相互通信* および 161 ページの『リモート管理のためにチャネルおよび伝送キューを作成する』を参照してください。

クラスター

分散キューイングを使用する従来の MQSeries ネットワークでは、すべてのキュー・マネージャーは独立しています。1 つのキュー・マネージャーが別のキュー・マネージャーにメッセージを送信する必要がある場合、伝送キュー、リモート・キュー・マネージャーへのチャネル、メッセージの送信先であるすべてのキューのリモート・キュー定義を定義しておく必要があります。

クラスターは、複雑な伝送キュー、チャネル、およびキューを定義しなくても、複数のキュー・マネージャーが単一のネットワーク上で互いに直接通信できる方法でセットアップされたキュー・マネージャーのグループです。

クラスターについては、157 ページの『第7章 リモート MQSeries オブジェクトの管理』および *MQSeries* キュー・マネージャー・クラスター を参照してください。

名前リスト

名前リストは、MQSeries オブジェクトの 1 つで、他の MQSeries オブジェクトのリストが格納されています。通常、名前リストはトリガー・モニターなどのアプリケーションによって、キュー・グループの識別に使用されます。名前リストを使用した場合の利点は、アプリケーションとは別個に管理できるということです。したがって、更新時に、その名前リストを使用しているアプリケーションを停止する必要はありません。また、あるアプリケーションで障害が起こった場合でも、名前リストには影響はなく、他のアプリケーションは引き続きその名前リストを使用できます。

名前リストは、キュー・マネージャーのクラスター用にも使用できます。これによって、複数の MQSeries オブジェクトが参照するクラスターのリストを管理することができます。

システム・デフォルト・オブジェクト

システム・デフォルト・オブジェクト とは、キュー・マネージャーの作成時に各キュー・マネージャーごとに自動的に作成される 1 組のオブジェクト定義のことです。

デフォルト・オブジェクト名には、語幹 `SYSTEM.DEFAULT` が付いています。たとえば、デフォルト・ローカル・キューは `SYSTEM.DEFAULT.LOCAL.QUEUE` であり、デフォルト受信チャネルは `SYSTEM.DEFAULT.RECEIVER` です。これらのオブジェクトの名前を変更することはできません。これらの名前を持つデフォルト・オブジェクトは必須です。

オブジェクトを定義する際に、明示的に指定されなかった属性は該当するデフォルト・オブジェクトからコピーされます。たとえば、ローカル・キューを定義する場合、指定しなかった属性は、デフォルト・キュー `SYSTEM.DEFAULT.LOCAL.QUEUE` から取られます。

キュー・マネージャーが作成されたら、`runmqsc` コマンドを使用して、デフォルトを他の定義に置き換えることができます。

システム・デフォルトの詳細については、405ページの『付録B. システム・デフォルト』を参照してください。

管理

MQSeries では、コマンド を発行することによって管理タスクを実行します。4 つのコマンド・セットが備わっています。どのセットを使用するかは、実行するタスクやタスクを実行する方法によります。これらのコマンド・セットについては、

37ページの『第3章 MQSeries コマンド・セットの使用』で説明しています。管理タスクには次のものが含まれます。

- キュー・マネージャーの開始および停止。
- アプリケーション用のオブジェクト、特にキューの作成。
- チャネルを使用して、他の (リモート) システムにあるキュー・マネージャーへの通信パスを作成。このプロセスについては、*MQSeries 相互通信* で詳しく説明されています。

ローカル管理およびリモート管理

ローカル管理とは、ローカル・システムに定義したキュー・マネージャーで管理タスクを実行することです。たとえば、TCP/IP の端末エミュレーション・プログラム **telnet** を介して、他のシステムにアクセスし、そこで管理作業を行うことができます。MQSeries では、これをローカル管理と考えることができます。なぜなら、チャネルは無関係であり、通信はオペレーティング・システムによって管理されるからです。

MQSeries は、リモート管理 という管理方法による、1 つの地点からの管理をサポートします。これにより、別のシステムで処理されるコマンドを、ユーザーのローカル・システムから出すことができます。この場合、別のシステムにログオンする必要はありませんが、適切なチャネルを定義しておく必要があります。ターゲット・システム上のキュー・マネージャーおよびコマンド・サーバーは、実行中である必要があります。たとえば、リモート・コマンドを出して、リモート・キュー・マネージャー上のキュー定義を変更することができます。

一部のコマンドは、このような方法では発行することができません。特に、キュー・マネージャーの作成や開始、およびコマンド・サーバーの開始などの際には、このような方法では発行できません。この種のタスクを実行するためには、リモート・システムにログオンしてそこからコマンドを発行するか、あるいはユーザーの代わりにコマンドを発行するプロセスを作成する必要があります。

クライアントおよびサーバー

MQSeries for Compaq NSK は、MQI アプリケーション用のクライアント / サーバー構成をサポートします。MQSeries for Compaq NSK クライアントは存在せず、MQSeries for Compaq NSK サーバーだけが存在します。ただし、他のプラットフォーム上のクライアントが MQSeries for Compaq NSK サーバーに接続することができます。

MQI クライアント は、マシン上にインストールされる MQSeries プロダクトの一部であり、アプリケーションからの MQI 呼び出しを受け入れて、それを MQI サーバー・マシンに渡します。これらは、キュー・マネージャーによって処理されます。クライアントとサーバーは、通常は異なるマシン上にありますが、同一マシン上に存在する場合があります。

MQI サーバー とは、キューイング・サービスを 1 つまたは複数のクライアントに提供するキュー・マネージャーのことです。たとえばキューなどの MQSeries オブジェクトのすべては、キュー・マネージャー・システム上、つまり、MQI サーバー・マシン上にのみ存在します。サーバーは、通常のローカル MQI アプリケーションもサポートできます。

詳細については、*MQSeries 相互通信* と *MQSeries クライアント* を参照してください。

クライアント / サーバー環境での MQI アプリケーション

サーバーにリンクされていれば、MQI クライアント・アプリケーションは、ローカル・アプリケーションの場合と同様にして MQI 呼び出しを発行することができます。クライアント・アプリケーションは、MQCONN 呼び出しを発行して、指定のキュー・マネージャーに接続します。接続要求から戻された接続ハンドルを指定した追加の MQI 呼び出しは、このキュー・マネージャーによって処理されます。

ユーザーのアプリケーションを、該当のクライアント・ライブラリーにリンクする必要があります。詳細については、*MQSeries アプリケーション・プログラミング・ガイド* を参照してください。Compaq NSK 用の MQI クライアントは現在提供されていません。ただし、Compaq NSK は MQI サーバーなので、他のプラットフォーム上で実行されている任意の MQSeries MQI クライアントからの接続を受け入れます。

キュー・マネージャー機能の拡張

キュー・マネージャーが提供する機能は、次のものによって拡張できます。

- ユーザー出口
- インストール可能なサービス

ユーザー出口

ユーザー出口を使用すれば、独自のプログラミング・コードをキュー・マネージャー機能に挿入することができます。2つのタイプのユーザー出口がサポートされています。

- チャンネル出口

この出口では、チャンネルの動作方法を変更できます。チャンネル出口については、*MQSeries 相互通信* で説明されています。

- データ変換出口

この出口では、アプリケーション・プログラムに書き込んでデータの形式を変換するための部分ソース・コードを作成できます。データ変換出口については、*MQSeries アプリケーション・プログラミング・ガイド* で説明されています。

- クラスタ・ワークロード出口

この出口を使用すれば、クラスタ内のキュー・マネージャーが複数のリモート・キューのインスタンスから選択する方法を変更することができます。呼び出し定義情報については、*MQSeries キュー・マネージャー・クラスタ* で説明されています。

注: *MQSeries* でユーザー出口を使用可能にするためのメカニズムは、バージョン 5.1 で変更されました。以前のバージョンの *MQSeries for Compaq NSK* からユーザー出口コードをマイグレーションする場合は、出口メカニズムの説明を注意深く検討して、必要な変更点を判別してください。

これらの出口の詳細については、459ページの『付録L. ユーザー出口』を参照してください。

インストール可能なサービス

インストール可能なサービスは、複数のエントリー・ポイントを持つ形式化されたアプリケーション・プログラミング・インターフェース (API) があるという点で、ユーザー出口よりもさらに利用範囲の広いものです。

インストール可能なサービスは、サービス・コンポーネント と呼ばれるもので実現されます。プロダクトで提供されているコンポーネントを使用することも

できますし、必要な機能を実行するコンポーネントを独自に作成することもできます。現在、次のようなインストール可能なサービスが用意されています。

- **許可サービス**

許可サービスにより、独自のセキュリティー機能を構築することができます。

このサービスを実現するデフォルト・サービス・コンポーネントは、オブジェクト権限マネージャー (OAM) です。これはプロダクトと共に提供されています。デフォルトでは、OAM は使用可能になっています。許可サービス・インターフェースを使用して別のコンポーネントを作成し、OAM を置き換えたり OAM を補強したりすることができます。OAM の詳細については、177ページの『第8章 MQSeries オブジェクトの保護』を参照してください。

- **名前サービス**

これにより、アプリケーションはリモート・キューをローカル・キューであるかのように認識することができ、キュー・マネージャーはキューを共有できます。

独自の名前サービス・コンポーネントを作成することもできます。詳細については、*MQSeries プログラム式システム管理* を参照してください。

セキュリティー

MQSeries for Compaq NSK は、オブジェクト権限マネージャー (OAM) 機能によってセキュリティーを提供します。

オブジェクト権限マネージャー (OAM) 機能

MQI 呼び出しとコマンドの使用、およびオブジェクトへのアクセスのための許可は、オブジェクト権限マネージャー (OAM) によって提供されます。OAM はデフォルトでは使用可能になっています。MQSeries エンティティーへのアクセスは、MQSeries for Compaq NSK のプリンシパルのユーザー・グループ、および OAM によって制御されます。OAM がサポートするプリンシパル名とグループ名は、Compaq NSK のユーザー名とグループ名に解決されます。バージョン 5.1 では、MQSeries のすべてのユーザーは、Compaq NSK ユーザー名にマップされるプリンシパル名を持つ必要があります。これは、OAM が使用可能になっているかどうかにかかわらず必要です。必要に応じてプリンシパルの追加や削除、および許可の認可や取り消しを行えるよう、コマンド行のインターフェースが提供されています。

加えて、Compaq NSK セキュリティー機能を使用すれば、MQSeries のコマンドやデータベース・ファイルへのアクセスを制御することができます。

SAFEGUARD がインストールされている場合、MQSeries は SAFEGUARD が備える拡張機能の一部と互換性を持ち、それを利用することができます。詳細については、177ページの『第8章 MQSeries オブジェクトの保護』を参照してください。

トランザクション・サポート

アプリケーション・プログラムは、一連の更新内容を 1 つの作業単位としてグループ化します。作業単位としてグループ化される更新は相互に論理的に関連しているため、データの一貫性を維持できるように正常に処理されなければなりません。ある更新処理が正常に行われ、一方で他の更新処理が失敗すると、データの一貫性が失われます。

処理が正常に完了すると、作業単位はコミットされます。この時点で作業単位内のすべての更新内容が永久的なものとなり、これ以降は取り消し不可能になります。作業単位が失敗した場合は、すべての更新がバックアウトされます。同期点調整とは、データの一貫性を維持しながら作業単位をコミットまたはバックアウトするプロセスです。

ローカル作業単位では、更新されるリソースのみが MQSeries キュー・マネージャーに属するリソースとなります。この場合、キュー・マネージャー自体が単一フェーズ・コミットによって同期点を調整します。

グローバル作業単位では、XA 準拠データベースなどの他のリソース・マネージャーにより管理されているリソースも更新されます。この場合、2 フェーズ・コミット手順を実行する必要があるため、作業単位はキュー・マネージャー自体によって調整されるか、XA に準拠する別のトランザクション・マネージャー (IBM® CICS®, Transarc Encina, BEA Tuxedo など) によって外部で調整されます。

MQSeries for Compaq NSK では、すべての作業単位は TM/MP によって調整されるか、それ自体が TM/MP を使用する、互換性のある階層化プロダクト (たとえば、OSS 環境における NonStop Tuxedo) によって調整されます。アプリケーションは TM/MP を使用して、MQSeries メッセージや TM/MP と互換性のあるリソースの更新を含む作業単位を調整することができます。Compaq NSK 環境には XA 準拠データベースはありませんが、類似のインターフェースがインプリメントされています。たとえば、NonStop SQL データベースや ENSCRIBE ファイルの更新は、データベースの健全性を維持するため、MQSeries メッセージング操作によって調整されます。

セキュリティー

詳細については、231ページの『第11章 トランザクション・サポートとメッセージングについての理解』を参照してください。

パフォーマンスのチューニング、信頼性、スケーラビリティ、およびサイジング

MQSeries for Compaq NSK には、スケーラブルでハイパフォーマンスかつ信頼性のあるアプリケーション・インフラストラクチャーを実現する Compaq NSK ハードウェアおよびソフトウェアの機能を利用する機能が備わっています。この機能には次のものが含まれます。

- 複数の CPU とディスクにまたがって複数のキュー・マネージャーを構成し、並列性を向上させる柔軟性
- 最悪でも一時的なサービスへの割り込みと完全なサービスへの自動リカバリーによって単一障害ポイントを解決できることを保証する、ソフトウェア機能
- ソフトウェアと基礎となるトランザクション・ファイル・システムによって保護される、データベースの保全性
- 災害時回復保護を備えるプロダクトとの互換性

詳細については、283ページの『第15章 スケーラビリティおよびパフォーマンス』を参照してください。

第2章 MQSeries for Compaq NSK V5.1 のアーキテクチャ

この章では、MQSeries for Compaq NSK V5.1 の全体的なアーキテクチャーについて説明します。この章には、次の節があります。

- 『キュー・マネージャー・プロセスの概要』
- 30ページの『プロダクト・パッケージ』
- 30ページの『実行可能プログラム』
- 31ページの『バージョン 5.1 とバージョン 2.2.0.1 のアーキテクチャーの比較』
- 32ページの『キュー・マネージャー — 機能の概観』
- 34ページの『キュー・マネージャー・プロセス・モデル』
- 34ページの『MQSeries のファイルとサブボリューム』

この情報は、ご使用の操作環境に合わせてプロダクトを最適に構成するのに役立ちます。

283ページの『第15章 スケーラビリティおよびパフォーマンス』と 299ページの『第16章 データ保全性と可用性』では、このアーキテクチャー情報に関連した具体的な構成の指示も示しています。

キュー・マネージャー・プロセスの概要

下の図は、MQSeries キュー・マネージャーを構成するプロセスを示しています (キュー・マネージャーは PATHWAY で実行されるので、図を完全なものとするために PATHMON が示されています)。また、プロセス IPC 対話の概要も示しています。

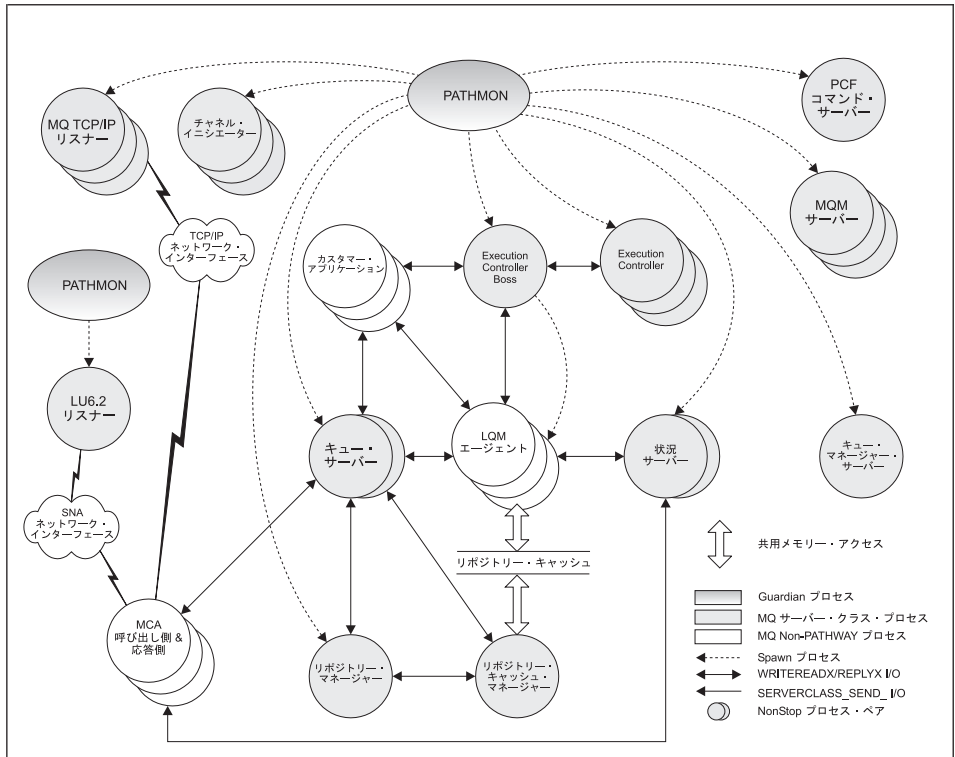


図1. MQSeries for Compaq NSK のプロセス

キュー・マネージャーが開始すると、PATHWAY はかぎとなるキュー・マネージャー・プロセスを作成します。デフォルト構成では、PATHMON は Execution Controller Boss (ECBoss)、Execution Controller (EC)、状況サーバー、キュー・サーバー、およびリポジトリ・マネージャーを開始します。

ECBoss は、すべての着信 MQCONN 要求を処理し、使用可能な Execution Controller に分散して、最も少ない数の接続を処理する EC を選択することによるロード・バランシング機能を提供します。

EC は、他のキュー・マネージャー・プロセスと MQI アプリケーションの管理とモニターを行います。

ローカル・キュー・マネージャー・エージェント (LQMA) は、アプリケーションのために MQI 呼び出しを処理するために必要な操作を実行します。このエージェントは、MQI をサポートする大量のコードを実行します。このエージェントの主な目的は、アプリケーション・プログラムをキュー・マネージャーのクリティカル・リソースから分離し、危険または有害なアプリケーションから保護することです。

エージェント・プロセスの数は、接続されているアプリケーションの数と、QMINI ファイルで指定されている MinIdleLQMAgents によって決まります。キュー・マネージャーが開始すると、それぞれの Execution Controller は指定された数のアイドル・エージェントを開始します。エージェントがアクティブになると、Execution Controller は新しいエージェントを開始して、最小数のアイドル・エージェントを維持します。

バージョン 5.1 の MQSeries アーキテクチャーの中心をなすのは、キュー・サーバー です。キュー・サーバーは、ローカル・キューのすべてのメッセージング操作をサポートする NonStop プロセス・ペアです。最初に開始したとき、キュー・マネージャーにはデフォルトのキュー・サーバーが 1 つあります。ご使用のシステム構成およびパフォーマンス要件に応じて、追加のキュー・サーバーを構成し、それらにローカル・キューを割り当てることもできます。

状況サーバーは、ローカル・キューを除くすべてのオブジェクトの状況情報を処理します。このサーバーには、バージョン 2.2.0.1 で使用された MQSS サーバーの機能が組み込まれています。加えて、デフォルトの状況サーバーは、キュー・マネージャーのチャンネル状況情報を処理します。最初に開始したとき、キュー・マネージャーにはデフォルトの状況サーバーが 1 つあります。ご使用のシステム構成およびパフォーマンス要件に応じて、追加の状況サーバーを構成し、それらにMQSeries オブジェクトを割り当てることもできます。

キュー・マネージャーは、開始時にチャンネル・イニシエーター・サーバー・クラスを自動的に開始します。これは、チャンネル・イニシエーターを手作業で開始しなくてもクラスター化操作が機能できるようにするためのものです。

バージョン 5.1 に新しく追加されたのは、クラスター・キューを処理するリポジトリー・マネージャー およびリポジトリー・キャッシュ・サーバー・プロセスです。Execution Controller が実行されている各 CPU には、1 つのリポジトリー・マネージャー・プロセスがあります。最初に開始したプロセスがリポジトリー・マネージャーの役割を担い、キュー・マネージャーにおけるリポジトリー・アクティビティを調整します。それ以降のプロセスは、自身が実行されている CPU のリポジトリー・キャッシュを含む共用メモリー・セグメントを管理します。リポジトリー・マネージャーは、CPU ごとに 1 つしか存在できません。リポジトリー・キャッシュには、クラスター化された MQSeries オブジェクト (クラスター内のキューと他のキュー・マネージャーを含む) に関する情報が入っています。

バージョン 5.1 で新しく追加された別のものは、キュー・マネージャー・サーバー です。このサーバーは、有効期限が切れたメッセージとレポートを処理します。

メッセージ・チャネル・エージェント (MCA) は、他のキュー・マネージャーとの間でメッセージをやり取りします。MQSeries for Compaq NSK V5.1 では、MCA は効率化のために FASTPATH によって制限されています。FASTPATH アプリケーションの詳細については、295ページの『FASTPATH バインディング・アプリケーション・プログラム』を参照してください。

LU6.2 リスナー (別個の PATHWAY 環境で実行される) もバージョン 5.1 で新しく追加されたものです。471ページの『付録M. 通信のセットアップ』には、リスナーとその環境の操作および構成に関する詳細情報が載せられています。

プロダクト・パッケージ

MQSeries for Compaq NSK V5.1 は、3 つのタイプのバインディングを提供します (ネイティブ動的、ネイティブ静的、および非ネイティブ)。

ネイティブ動的バインディング用に、MQSeries はネイティブ・モードの共用リソース・ライブラリー (MQSRLLIB) を備えています。MQSeries for Compaq NSK V5.1 の設計目標の 1 つは、できるだけ多くのプロダクトを共用リソース・ライブラリー (SRL) に組み込み、プロダクトのフットプリントを最小にすることでした。MQSeries を使用する際は、ネイティブ動的バインディングの使用が推奨されています。こうすれば、システム・リソースを最も効率的に使用できます。OSS 環境の場合、このバインディングだけがサポートされています。

ネイティブ静的バインディング用に、MQSeries for Compaq NSK は再リンク可能ライブラリー (MQSRLLNK) を備えています。このライブラリーは、すでに SRL を使用していて、MQSeries の SRL を使用できないお客様のために提供されています。

非ネイティブ静的ライブラリーは、ネイティブ・モードのバインディングを使用できないレガシー・アプリケーションとの互換性を得るために提供されています。

実行可能プログラム

次の表は、MQSeries の実行可能プログラムを示しています。

表 1. MQSeries の実行可能プログラム

MQSeries の実行可能プログラム	名前
Execution Controller	MQEC

表1. MQSeries の実行可能プログラム (続き)

MQSeries の実行可能プログラム	名前
Execution Controller Boss	MQECBOSS
ローカル・キュー・マネージャー・エージェント	MQLQMAG
状況サーバー	MQSTSVR
キュー・サーバー	MQQSSVR
リポジトリ・マネージャー	MQREPSVR
キュー・マネージャー・サーバー	MQMGRSVR
コマンド・サーバー	MQCMDSVR
MQSeries 管理サーバー	MQMSVR
チャンネル・イニシエーター	RUNMQCHI
呼び出し側 MCA	MQMCACAL
TCP/IP 応答側 MCA	MQTCPRES
LU6.2 応答側 MCA	MQLU6RES
トリガー・モニター	RUNMQTRM

バージョン 5.1 とバージョン 2.2.0.1 のアーキテクチャーの比較

MQSeries バージョン 2.2.0.1 とバージョン 5.1 の主な構造上の変更点は、キュー・サーバーが導入されたことです。各キュー・サーバーは、1 つまたは複数のローカル・キュー・オブジェクトに関連したデータおよびファイルを保守しています。最初に開始したとき、キュー・マネージャーにはデフォルトのキュー・サーバーが 1 つあり、これはキュー・マネージャーに関連したすべてのローカル・キューを担当します。追加のキュー・サーバーを (PATHWAY を使用して) 追加し、**altmqfls** を使用してそのキュー・サーバーにキューを割り当てることもできます。

キュー・サーバーは NonStop プロセス・ペアであり、ローカル・キューのすべてのメッセージング操作をサポートします。キュー・サーバーはまた、純粋にメモリー・ベースの非持続メッセージもサポートしています。これは、バージョン 2.2.0.1 に対して大きなパフォーマンス上の利点となっています。

MQSeries では、TM/MP への内部インターフェースにより、これらのメモリー・ベースのメッセージがトランザクションに統合されているため、非持続メッセージをディスクに書き込む際のオーバーヘッドを伴わずに、持続メッセージと非持続メッセージの両方を同じトランザクションに含めることができま

す。キュー・サーバーは、ローカル・キューに関してのみ、バージョン 2.2.0.1 の MQSS サーバーの機能を引き継いでいます。

MQSS サーバーは状況サーバーに取って代わられました。状況サーバーは、ローカル・キューを除くオブジェクトの状況情報を保守する NonStop プロセス・ペアです。

アーキテクチャーにおける 2 番目の主な変更点は、共用メモリー・セグメントが導入されたことです。キュー・マネージャーの初期設定ファイルは、読み取り専用共用メモリーを使用するキュー・マネージャーに分散されます。これにより、キュー・マネージャーに接続する際のパフォーマンスが向上します。

MQSeries リポジトリ・キャッシュ (クラスター化機能をインプリメントするために使用される) は、MQSeries プロセスが実行されている各 CPU で、読み取り / 書き込み共用メモリー・セグメントを使用します。

バージョン 5.1 では、LU6.2 チャンネルがより透過的にサポートされています。このバージョンでは、LU6.2 応答側の開始を担当する LU6.2 リスナーが導入されています。この結果、LU6.2 チャンネルの "AUTOSTART" 属性は除去されました。

キュー・マネージャー - 機能の概観

キュー・マネージャーには次の主なコンポーネントがあります。

アプリケーション・インターフェース

MQI 呼び出しを実行するための環境とメカニズムを提供します。

キュー・マネージャー・カーネル

MQI のほとんどの機能を提供します。たとえば、トリガーはここでインプリメントされています。

オブジェクト権限マネージャー (OAM)

キュー・マネージャーとそのリソースのアクセス制御を提供します。ユーザーやグループが実行できる操作や、対象となるリソースを指定することができます。

データ抽象および持続 (DAP)

キュー・マネージャーが保持するデータの保管とリカバリーを提供します。DAP はメッセージを保持します。

メッセージ・チャンネル・エージェント

これらは、大多数の操作で MQI を使用する特殊なアプリケーションです。キュー・マネージャー間での信頼できるメッセージの伝送を担当します。MCA は FASTPATH で制限されています。

コマンド・サーバー

コマンド・サーバーは、キュー・マネージャーを管理するコマンドを含むメッセージの処理を担当する、特殊な MQI アプリケーションです。

共通サービス

これは、キュー・マネージャーの残りの部分をオペレーティング・システムから分離し、オペレーティング・システムと類似のサービス (ストレージ管理、シリアライゼーション、およびプロセス管理など) のセットを提供します。

図2 は、コンポーネント間の関係を示しています。

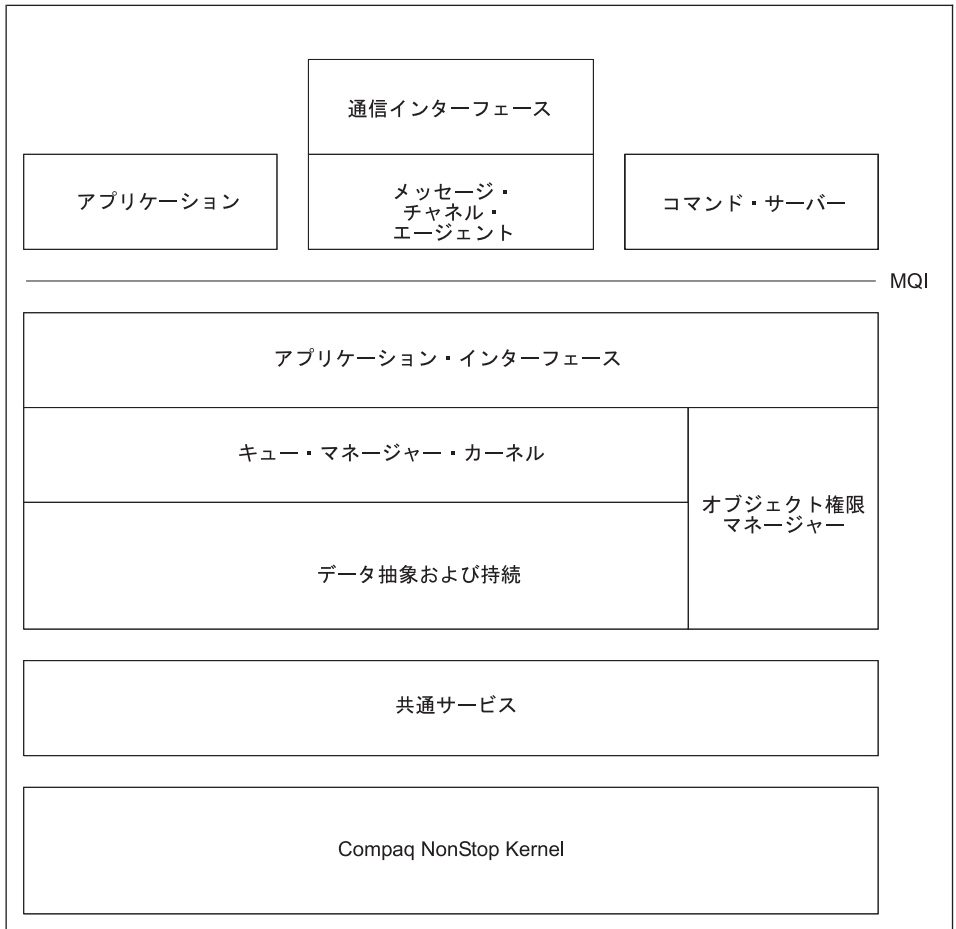


図2. MQSeries for Compaq NSK V5.1 のコンポーネント

キュー・マネージャー・プロセス・モデル

アプリケーションは、エージェントと接続する必要があると、Execution Controller Boss (ECBoss) と通信します。ECBoss は最も使用されていない (接続されているアプリケーションが最も少ない) Execution Controller (EC) を選択し、接続要求をその EC に転送します。EC は、アイドル状態のローカル・キュー・マネージャー・エージェント (LQMA) を選択します。EC が ECBoss を介して応答をアプリケーションに戻すと、ECBoss は選択された LQMA に接続します。

アプリケーション・インターフェースは 2 つの部分に分けられます。

- MQI アプリケーション・スタブは、MQ 要求をパックし、プロセス間通信コンポーネント (IPCC) を使用してそれをエージェント・プロセスに渡します。
- IPCC は、MQI アプリケーション、エージェント、EC、および ECBoss の間のメッセージ受け渡しインターフェースを提供します。MQSeries IPCC コンポーネントの基礎を成しているのは標準の GUARDIAN IPC です。

アプリケーションは、IPCC を介してエージェント・プロセスと通信します。エージェント・プロセスは、アプリケーションのために MQI 呼び出しを実行します。IPCC は、アプリケーションとエージェントの間で、同期する要求応答メッセージを交換します。エージェントと通信することに加え、アプリケーションは、MQPUT および MQGET 操作を実行するときに、そのキューを担当するキュー・サーバーにメッセージ・データを直接転送します。

FASTPATH によって制限されるアプリケーションの場合、MQSeries コードはユーザー・アプリケーションに直接リンクされます。これによりパフォーマンスは向上しますが、危険なアプリケーションがキュー・マネージャーの操作を混乱させ、データの消失や他の問題を引き起こす可能性があります。

MQSeries のファイルとサブボリューム

MQSeries は、いくつかの Compaq NonStop ENSCRIBE ファイルを使用します。ファイルの位置と名前について以下に要約します。

MQSeries キュー・マネージャーに関連したファイルは、いくつかのサブボリュームに分散されます。

<qmgr> D

データ・サブボリューム。キュー・マネージャー全体に関する情報を保持するファイルに使用します。

<qmgr> M

メッセージ・データ・サブボリュームメッセージ・データに関連したファイルに使用します。

<qmgr> L

エラー・ログ

<qmgr> S

チャンネル同期ファイル。

<qmgr>

FFST™ ファイル

オブジェクト・カタログ

オブジェクト・カタログ (OBJCAT) には、MQSeries オブジェクトに関する情報が含まれます。オブジェクト・カタログは、キュー・マネージャーのデータ・サブボリュームにあります。

キュー・ファイル

各ローカル・キューには、関連したキュー・ファイル、キュー・オーバーフロー・ファイル、およびタッチ・ファイルがあります。これらのファイルにはそれぞれ Q、O、および T の接頭部が付いています。名前の残りの部分は、オブジェクトの MQSeries 名またはシステム生成名の一部となっています。特定のキューに関連したファイルを検索するには、**dspmqls** ユーティリティを使用します。キューが作成されると、キュー・マネージャーは、キュー・マネージャーのメッセージ・サブボリューム (<qmgr>M) に関連するファイルを作成します。

altmqfls は、パフォーマンスやスペースの理由によりこれらのファイルの位置を別のボリュームに変更しなければならない場合にそれを行う機能を提供します。

キュー・マネージャーは、キューについて構成されたメッセージ・オーバーフローしきい値より大きい各メッセージについて、メッセージ・オーバーフロー・ファイルを作成します。メッセージ・オーバーフロー・ファイルは、デフォルトでキュー・サーバーのサブボリュームに入っています。キュー・サーバーがメッセージ・オーバーフロー・ファイルを作成するサブボリュームを変更するには、**altmqfls** ユーティリティを使用します。メッセージ・オーバーフロー・ファイルは、特定のメッセージに特化された、構造化されておらず、監査されないファイルです (キュー・ファイルの構成について詳しくは、287ページの『キュー・サーバーおよびキュー・ファイル』を参照してください)。効率的なデータ転送を保証するため、データは大量転送モード (SETMODE 141) を

使用してメッセージ・オーバーフロー・ファイルに書き込まれるので、アプリケーション・プロセス・メモリーからの 56 KB セグメントでの直接転送が可能となっています。

約 3 KB より小さい持続メッセージの場合、メッセージ・データはキュー・ファイルに保管されます。大きさが 3 KB とそのキューに構成されたメッセージ・オーバーフローしきい値の間のメッセージの場合、キュー・サーバーは追加のレコードをキュー・オーバーフロー・ファイルに書き込みます。メッセージ・オーバーフローしきい値を超える持続メッセージの場合、キュー・サーバーはメッセージを保管するメッセージ・オーバーフロー・ファイルを作成します。

サイズまたはパフォーマンス上の理由でキュー・ファイルやキュー・オーバーフロー・ファイルを区分しなければならない場合は、290ページの『キュー・ファイルの区画化』にこれを行う方法が説明されています。ファイルを区分すると、MQSeries は区画間にメッセージ・データを均等に分散して、パフォーマンスを最適化します。

283ページの『第15章 スケーラビリティおよびパフォーマンス』には、キューおよびキュー・ファイルに関連したパフォーマンス・チューニング・オプションの詳細が載せられています。

別名およびリモート

別名およびリモート・キューには、関連したタッチ・ファイルがあります。タッチ・ファイルは接頭部 T が付けられており、キュー・マネージャーのデータ・サブボリュームにあります。

Namelist Files

それぞれの名前リストには、関連した非構造化ファイルがあります。このファイルは接頭部 L が付けられており、キュー・マネージャーのデータ・サブボリュームにあります。ファイル名の残りの部分は、オブジェクトの MQSeries 名の一部またはシステム生成の値となっています。特定の名前リストに関連したファイルの名前を検索するには、**dspmqls** ユーティリティを使用します。

第3章 MQSeries コマンド・セットの使用

この章では、MQSeries オブジェクトに対するシステム管理タスクで使用できるコマンドについて説明します。管理タスクには、キュー・マネージャー、キュー、プロセス、プロセス、およびチャネルの作成、起動、変更、表示、停止、および削除があります。これらのタスクを実行するには、適切なコマンドを選択しなければなりません。

MQSeries for Compaq NSK V5.1 は、管理用タスクを実行するための以下の管理コマンド・セットを提供しています。

- MQSC (MQSeries コマンド)
- PCF (プログラム式コマンド形式) コマンド
- 制御コマンド
- MQAI (MQSeries 管理インターフェース)

さらに、

- いくつかの TS/MP (PATHWAY) コマンドが管理目的で使用されます。
- MQM (メッセージ・キュー管理) 機能がいくつかの管理タスクをサポートしています。MQM については、90ページの『メッセージ・キュー管理 (MQM) 機能の使用』を参照してください。

この章では、413ページの『付録D. コマンド・セットの比較』で扱われている、MQSC、PCF、および制御コマンド・セットについて紹介し、それぞれのコマンド・セットがサポートしている機能を要約しています。TS/MP コマンドの使用方法については、41ページの『TS/MP (PATHWAY) 管理』を参照してください。

制御コマンドを使用した管理の実行

以下のタイプの制御コマンドを使用できます。

- キュー・マネージャーを作成、開始、停止、および削除するためのコマンド。
- コマンド・サーバーを開始、停止、および表示するためのコマンド。
- MQSC コマンドの実行、MQSeries オブジェクトへのアクセスの管理、MQSeries トレースの開始および停止、トリガー・モニターの実行などに関連するユーティリティ・コマンド。

制御コマンド

制御コマンドの使用

制御コマンドは Compaq TACL プロンプトから実行します。コマンド名には大文字小文字の区別はありません。(ただし、キュー・マネージャー名には大文字小文字の区別があるので注意してください。)たとえば、次のようにします。

```
runmqsc
```

323ページの『第17章 MQSeries 制御コマンド』では、各コマンドの構文および目的が説明されています。

MQSC コマンドを使用した管理の実行

キュー・マネージャー、チャンネル、キュー、およびプロセス定義などのキュー・マネージャー・オブジェクトを管理するために、MQSC コマンドを使用します。たとえば、MQSC コマンドを使用して、指定したキューの定義、更新、表示、および削除を行えます。

DISPLAY QUEUE コマンドを使用してキューを表示する場合、キュー属性を表示します。たとえば、MAXMSGL 属性はキューに入れることのできるメッセージの最大長を指定します。コマンドはキュー上のメッセージの表示は行いません。これらのコマンドについては、413ページの『付録D. コマンド・セットの比較』で要約されています。各 MQSC コマンドの詳細については、*MQSeries MQSC コマンド・リファレンス* を参照してください。

MQSC コマンドの実行

Compaq TACL プロンプトから制御コマンド **runmqsc** を呼び出すことにより、またはローカル・キュー・マネージャーを実行する際にスクリプトを実行することにより、MQSC を対話式に実行できます。コマンドに設定したフラグによって、**runmqsc** コマンド自体を次の 3 とおりのモードで実行できます。

- **検証モード**。このモードでは、MQSC コマンドはローカル・キュー・マネージャー上で検査されますが、実行される訳ではありません。
- **直接モード**。このモードでは、MQSC コマンドはローカル・キュー・マネージャー上で実行されます。
- **間接モード**。このモードでは、MQSC コマンドはリモート・キュー・マネージャー上で実行されます。

MQSC 機能およびテキスト・ファイルの使い方の詳細については、120ページの『MQSC 機能を対話式で使用する』を参照してください。 `runmqsc` コマンドの詳細については、377ページの『runmqsc (MQSeries コマンドの実行)』を参照してください。 MQSC コマンドについては、413ページの『付録D. コマンド・セットの比較』にその概要があります。

Compaq NSK Fix Command の使用

`runmqsc` を対話式に実行する (NSK TACL プロンプトから) 場合、MQSC コマンドの呼び出しおよび編集を行える、Compaq NSK Fix Command 機能を使用することもできます。たとえば、次のようにします。

- `history` または `h` を入力すると、現在からさかのぼって 10 個のコマンドのリストが生成されます。
- `!n` (ここで、*n* はコマンド番号) を入力すると、そのコマンドが再実行されます
- `h n` または `history n` (ここで、*n* は数字) を入力すると、現在からさかのぼって *n* 個のコマンドがリストされます。
- `fc` を入力すると、編集のために入力された最後のコマンドが表示されます。`fc n` (ここで、*n* はコマンド番号) を入力すると、番号を指定した編集のためのコマンドが表示されます。`fc string` (ここで、*string* は直前に入力されたコマンドの先頭部分または全体) を入力すると、指定した編集のためのコマンドの最後の出現が表示されます。構文は NSK 標準です。たとえば、文字を削除する場合は `d`、文字を挿入する場合は `i`、文字を置換する場合は `r` を入力します。

PCF コマンドを使用した管理の実行

PCF コマンドを使用すれば、管理用タスクをアプリケーションまたは管理プログラムにプログラミングすることができます。PCF コマンドは、MQSC 機能が提供する機能と同様の処理を行います。単一のノードからネットワーク内の任意のキュー・マネージャーへ PCF コマンドを発行するプログラムを作成することができます。また、管理タスクを中央集中方式にしたり自動化することができます。

各 PCF コマンドは、MQSeries メッセージのアプリケーション・データ部分に組み込まれたデータ構造です。各コマンドは、MQI 機能 `MQPUT` を使用して宛先キュー・マネージャーに送られます。メッセージを受信するキュー・マネージャー上のコマンド・サーバーは、そのコマンドをコマンド・メッセージとして解釈し、実行します。応答を入手するには、アプリケーションが `MQGET` 呼び出しを発行します。すると、応答データが MQSeries メッセージのアプリ

PCF コマンド

ケーション・データ部分のデータ構造として戻されます。次に、アプリケーションはその応答を処理し、その応答に応じてアクションを実行します。

注: MQSC コマンドとは異なり、PCF コマンドおよびそれらの応答は、テキスト形式では読み取ることはできません。

PCF コマンド・メッセージを作成するには、以下の項目を指定しなければなりません。

メッセージ記述子

これは、標準 MQSeries メッセージ記述子です。この中で、次のように指定します。

メッセージ・タイプ (*MsgType*) は MQMT_REQUEST です。

メッセージ形式 (*Format*) には、MQFMT_ADMIN を指定します。

アプリケーション・データ

これには、PCF ヘッダーを含む PCF メッセージが入ります。この中で、

PCF メッセージ・タイプ (*Type*) には MQCFT_COMMAND を指定します。

コマンド ID には、*ChangeQueue* (MQCMD_CHANGE_Q) などのコマンドを指定します。

PCF データ構造およびそれをインプリメントする方法の詳細な説明については、MQSeries プログラム式システム管理 を参照してください。

MQSC および PCF での属性

MQSC に指定するオブジェクト属性は、大文字 (たとえば、RQMNAME) で示されます。ただし、大文字小文字の区別はありません。これらの属性名は 8 文字までに制限されています (たとえば、QDPHIEV)。PCF のオブジェクト属性はイタリック体で表示され、8 文字までに制限されていません。RQMNAME と同等の PCF は *RemoteQMgrName* で、QDPHIEV と同等の PCF は *QDepthHighEvent* です。

エスケープ PCF

エスケープ PCF は、メッセージ・テキスト内に MQSC コマンドを含んでいる PCF コマンドです。PCF を使用して、リモート・キュー・マネージャーにコマンドを送信することができます。エスケープ PCF の使い方の詳細については、MQSeries プログラム式システム管理 を参照してください。

MQAI は、Compaq NSK 上で現在使用できる MQSeries に対する管理インターフェースです。これは、MQSeries を管理する簡単な方法です。独自の PCF

メッセージを書き込む必要がないので、複雑なデータ構造に関連する問題を回避することができます。MQAI の使用に関する詳細については、153ページの『MQAI を使用して PCF の使い方を単純化する』を参照してください。

TS/MP (PATHWAY) 管理

キュー・マネージャーに対するほとんどの操作は、TACL から MQSeries 制御コマンドを実行することによって行われます。ただし、PATHCOM を使用して TS/MP サーバー・クラスから直接操作しなければならない操作もあります。また、システム構成の変更が原因で、TS/MP 構成自体にいくつかの管理アクションを実行しなければならない場合もあります。

この節では、それらのアクティビティを要約しています。

TCP/IP リスナーの指定および制御

TCP/IP リスナーを開始するには、MQSeries 制御コマンド **runmqtsr** (375ページの『runmqtsr (リスナーの実行)』で説明されている) を使用するか、または PATHCOM コマンド **THAW SERVER** および **START SERVER** を使用できます。TCP/IP リスナーを停止するには、PATHCOM コマンド **FREEZE SERVER** および **STOP SERVER** を使用します。実行している TCP/IP コマンドの数、およびそれらのプロセス名を表示するには、PATHCOM コマンド **STATUS SERVER** を使用します。

デフォルトでは、リスナー (サーバー・クラス MQS-TCPLIS00 内にある) はキュー・マネージャーごとに 1 つだけです。複数の TCP/IP ポートを扱うために TCP/IP リスナー・サーバー・クラスを作成するには、PATHCOM コマンド **ADD SERVER** を使用します。柔軟性を最大にするために、TCP/IP リスナーはそれ自体の TS/MP サーバー・クラス内で構成してください。TCP/IP リスナーを追加する場合、70ページの『キュー・マネージャーが listen する TCP/IP ポート』で説明されているように、TCP/IP ポート定義をキュー・マネージャー初期設定ファイル (QMINI) に追加することも必要です。開始する最初のリスナーは QMINI 内で定義されている最初のリスナー・ポートを使用し、2 番目のリスナーは 2 番目のリスナー・ポートを使用して、以下も同様に使用していきます。複数のリスナーをサポートするのに必要な QMINI エントリーの例については、494ページの『複数の TCP/IP リスナーをサポートする QMINI を構成する』を参照してください。

コマンド・サーバーの制御

コマンド・サーバーは TS/MP サーバー・クラス MQS-CMDSERV00 として作成されます。制御コマンド **strmqcsv**、**endmqcsv**、および **dspmqcsv** の代

わりとして、PATHCOM コマンド THAW SERVER、START SERVER、FREEZE SERVER、STOP SERVER、および STATUS SERVER を使用できません。

チャンネル・イニシエーターの指定および制御

デフォルトのチャンネル・イニシエーターは TS/MP サーバー・クラス MQS-CHANINIT00 として作成されます。 **runmqchi** 制御コマンド (371ページの『runmqchi (チャンネル・イニシエーターの実行)』で説明されている) を使用する代わりに、PATHCOM コマンド THAW SERVER、START SERVER、FREEZE SERVER、STOP SERVER、および STATUS SERVER を使用して、チャンネル・イニシエーターの状況を制御および表示できます。デフォルトのチャンネル・イニシエーターは、デフォルトのチャンネル開始キュー SYSTEM.CHANNEL.INITQ を処理します。

チャンネル・イニシエーターのデフォルトの開始キューの変更

バージョン 5.1 では、キュー・マネージャーは自動的にデフォルトのチャンネル・イニシエーターを開始します。クラスター操作をサポートするために、デフォルトのチャンネル・イニシエーターを実行しなければなりません。

デフォルト (SYSTEM.CHANNEL.INITQ) 以外の開始キューを使用する場合、PATHWAY 構成を変更しなければなりません。

注: キュー・マネージャーがクラスターの一部である場合、開始キューは変更しないでください。デフォルトのチャンネル・イニシエーターのデフォルトの開始キューを変更すると、クラスターをサポートできなくなります。

キュー・マネージャーの実行中に、デフォルトの開始キューは変更できませんが、チャンネル・イニシエーターのサーバー・クラス自体は停止しなければなりません。PATHCOM で、キュー・マネージャーの PATHWAY 構成に対して以下のコマンドを発行してください。

```
ALTER SERVER MQS-CHANINIT00, STARTUP "-q<init-queue>"
```

ここで、<init-queue> は代替りの開始キューの名前です。それから、チャンネル・イニシエーターを開始し、PATHCOM を終了します。

トリガー・モニターの指定および制御

単一のデフォルトのトリガー・モニターは、TS/MP サーバー・クラス MQS-TRIGMON00 として作成されます。このサーバー・クラスを管理するた

めに、PATHCOM コマンド THAW SERVER、START SERVER、FREEZE SERVER、STOP SERVER、および STATUS SERVER を使用できます。追加トリガー・モニターが必要な場合、テンプレートとして MQS-TRIGMON00 を使用し、追加サーバーとしてそれらを構成できます。柔軟性を最大にするために、別々のサーバー・クラス・オブジェクトを使用するようにお勧めします。トリガー・モニターを制御するために、TS/MP を使用する必要はありません。たとえば、制御コマンド `runmqtrm` を使用して、TACL からトリガー・モニターを実行できます。

デフォルトのトリガー・モニターは、デフォルトの開始キュー `SYSTEM.DEFAULT.INITIATION.QUEUE` を処理します。これを変更するには、トリガー・モニターを保持するサーバー・クラスの `STARTUP` メッセージを追加または変更します。キュー・マネージャー用に複数のトリガー・モニターが構成されている場合、このような変更が必要です。STARTUP 属性を追加または変更するには、PATHCOM `ALTER SERVER` コマンドを使用します。

CPU 間でのプロセスの分配の指定

CPU 間での作業の分配に関する重要な面は、キュー・マネージャー内の各 EC への CPU の割り当てです。各 EC が作成および管理するのは、それ自体の CPU 内のエージェント・プロセスのセットだけです。したがって、EC プロセスがシステムの CPU 間で分散している場合、エージェント・プロセスも同様に分散しています。

デフォルトでは、複数の EC が指定されている場合、EC プロセス (個別のサーバー・クラスごとの) はシステム上の使用可能な CPU 間で、可能な限り均等に分散されます。キュー・マネージャー内の EC プロセスの数に関する制限は組み込まれていません。必要な数は、キュー・マネージャーが扱う負荷に完全に依存しています。デフォルトでは、キュー・マネージャー内の EC プロセスは 1 つです。

デフォルトの EC サーバーは `MQS-EC00` という名前です。複数の EC を持つキュー・マネージャーを作成するには、`crtmqm` コマンドで `-e` フラグを指定します。EC プロセスの数は、キュー・マネージャーを作成した後も変更できます。この変更を行うには、EC プロセス・サーバー・クラスを追加または削除し、`QMINI` ファイル内の `ECBoss` スタンザの `ExpectedNumECs` エントリに、対応する変更を行います。

各 EC プロセスは必ずそれ自体のサーバー・クラス内になければなりません。追加 EC プロセスを手動で作成しなければならない場合、テンプレートとして `MQS-EC00` サーバー・クラスを使用してください。

EC を扱う各 CPU は、リポジトリ・マネージャーも扱わなければなりません。キュー・マネージャーが作成されると、MQSeries は EC ごとにリポジトリ・マネージャーのサーバー・クラスを作成 (MQS-REPSVR00 形式の名前で) します。それまで EC を扱ったことのない CPU に、EC サーバー・クラスを手動で追加する場合、新しい CPU 内の新しいリポジトリ・マネージャーのサーバー・クラスを作成するために、テンプレートとして MQS-REPSVR00 サーバー・クラスを使用してください。

MQSeries では、以下のいずれかが当てはまる場合に、CPU ごとにリポジトリ・マネージャーのサーバー・クラスが必要です。

- CPU が EC を扱う
- CPU が MQS-QMGRSVR00 サーバー・クラスを扱う
- CPU が FASTPATH にバインドされたアプリケーションを実行する
- CPU のユーザーが MQSeries アプリケーションである、**runmqsc**、**runmqchi**、**runmqchl**、または **mqrepdmp** のいずれかを実行する

EC プロセス (または他のサーバー・クラス) への CPU のデフォルトの割り当ては、CPU 属性を指定して PATHCOM ALTER SERVER コマンドを使用することにより、変更できます。

デフォルトの状況サーバーは、サーバー・クラス MQS-STATUS00 内の **crtmqm** によって自動的に作成されます。デフォルトでは、1 次を CPU 0 に、バックアップを CPU 1 に (つまり CPUS(0:1)) という CPU の割り当てだけが行われます。PATHCOM ALTER SERVER コマンドを使用して、MQS-STATUS00 サーバーの CPU の割り当てを変更できます。CPUS(2:12) のように、2 つの CPU 番号をコロンで区切って指定することにより、状況サーバーが使用する特定のバックアップ CPU を指定できます。この場合、PATHMON は 1 次を CPU 2 に、バックアップを CPU 12 に作成します。特定のバックアップ CPU を指定しない場合、バックアップを作成する場所は、Compaq NSK オペレーティング・システムが決定します。

デフォルトのキュー・サーバーも、サーバー・クラス MQS-QUEUE00 内の **crtmqm** によって自動的に作成されます。デフォルトでは、1 次を CPU 0 に、バックアップを CPU 1 に (つまり CPUS(0:1)) という CPU の割り当てだけが行われます。PATHCOM ALTER SERVER コマンドを使用して、MQS-QUEUE00 サーバーの CPU の割り当てを変更できます。CPUS(2:12) のように、2 つの CPU 番号をコロンで区切って指定することにより、キュー・サーバーが使用する特定のバックアップ CPU を指定できます。この場合、PATHMON は 1 次を CPU 2 に、バックアップを CPU 12 に作成します。特

定のバックアップ CPU を指定しない場合、バックアップを作成する場所は、Compaq NSK オペレーティング・システムが決定します。

新しい状況サーバー・プロセスの追加

キュー・マネージャーに状況サーバーをさらに追加するには、テンプレートとしてデフォルトの状況サーバー・クラス MQS-STATUS00 を使用し、サーバー・クラスを作成します。

新しい状況サーバー・クラスの名前の先頭は、文字ストリング MQS-STATUS にしなければなりません。サーバー・クラス名がこの命名規則に従っていないと、キュー・マネージャーの開始時に **strmqm** がそれらを自動的に開始せず、これらの状況サーバー・クラス用に構成されているあらゆるオブジェクトへのアクセスが使用できなくなります。

追加状況サーバーを構成する場合、それぞれに固有のプロセス名が必要です。また、状況サーバーのアーキテクチャーが提供する拡張容易性を活用できるように、これらが別々の CPU で実行されるように構成することをお勧めします。

新しいキュー・サーバー・プロセスの追加

キュー・マネージャーにキュー・サーバーをさらに追加するには、テンプレートとしてデフォルトのキュー・サーバー・クラス MQS-QUEUE00 を使用し、サーバー・クラスを作成します。

新しいキュー・サーバー・クラスの名前の先頭は、文字ストリング MQS-QUEUE にしなければなりません。サーバー・クラス名がこの命名規則に従っていないと、キュー・マネージャーの開始時に **strmqm** がそれらを自動的に開始せず、これらのキュー・サーバー・クラス用に構成されているあらゆるオブジェクトへのアクセスが使用できなくなります。

追加キュー・サーバーを構成する場合、それぞれに固有のプロセス名が必要です。また、キュー・サーバーのアーキテクチャーが提供する拡張容易性を活用できるように、これらが別々の CPU で実行されるように構成することをお勧めします。

MQM モニター・パネルを最新表示する頻度の指定

MQS-MQMSVR00 の MQMQMREFRESHINT PATHWAY パラメーターで、チャンネルおよびキューのモニター画面が最新表示される頻度を決定します。デフ

ォルトの頻度は 30 秒ごとです。たとえば、頻度を 10 秒ごとに変更する場合、以下のように、キュー・マネージャーの PATHWAY から入力します。

```
ALTER SERVER MQS-MQMSVR00, PARAM MQMQMREFRESHINT 10
```

キュー・マネージャーの PATHWAY 構成

以下に、キュー・マネージャーの PATHWAY 構成の例を示します。この例は、デフォルトのキュー・マネージャー構成のオブジェクトに、一連の INFO コマンドを発行することによって生成されたものです。

```
TCP MQS-TCP-01
  AUTORESTART 0
  CHECK-DIRECTORY OFF
  CODEAREALEN 80000
  CPUS 0:1
  DEBUG OFF
  DUMP OFF
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  INSPECT OFF
  MAXINPUTMSGLEN 6000
  MAXINPUTMSGS 0
  MAXPATHWAYS 0
  MAXREPLY 32000
  MAXSERVERCLASSES 1
  MAXSERVERPROCESSES 10
  MAXTERMDATA 500000
  MAXTERMS 10
  NONSTOP 0
  POWERONRECOVERY ON
  PRI 175
  PROGRAM ¥RAPTOR.$SYSTEM.SYSTEM.PATHTCP2
  SERVERPOOL 32000
  STATS OFF
  TCLPROG ¥RAPTOR.$DEV.ZMQSEXE.POBJ
  TERMBUF 1500
  TERMPPOOL 10000
```

図 3. PATHWAY 構成の例 (1/13)

```

PROGRAM MQMC
  ERROR-ABORT OFF
  OWNER ¥RAPTOR.44,1
  SECURITY "N"
  TCP MQS-TCP-01
  TMF ON
  TYPE T16-6520 (BREAK OFF,ECHO ON,EXCLUSIVE OFF,INITIAL MAINC,IOPROTOCOL
0,MAXINPUTMSG 0,TRAILINGBLANKS ON)

```

図 3. PATHWAY 構成の例 (2/13)

```

SERVER MQS-CHANINIT00
  PROCESSTYPE GUARDIAN
  AUTORESTART 10
  CPUS (0,1)
  CREATEDELAY 1 MINS
  DEBUG OFF
  DELETEDELAY 10 MINS
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  LINKDEPTH 255
  MAXSERVERS 1
  NUMSTATIC 1
  OUT ¥RAPTOR.$ZTN0.#PTY001C
  OWNER ¥RAPTOR.44,1
  PARAM MQQUEMGRNAME "p101"
  PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
  PARAM MQDEFAULTPREFIX "$DEV"
  PRI 175
  PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.RUNMQCHI
  SECURITY "N"
  TMF ON
  VOLUME ¥RAPTOR.$DATA1.P101D

```

図 3. PATHWAY 構成の例 (3/13)

```
SERVER MQS-CMDSERV00
  PROCESSTYPE GUARDIAN
  AUTORESTART 10
  CPUS (0,1)
  CREATEDELAY 1 MINS
  DEBUG OFF
  DELETEDELAY 10 MINS
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  LINKDEPTH 255
  MAXSERVERS 1
  NUMSTATIC 1
  OUT ¥RAPTOR.$ZTN0.#PTY001C
  OWNER ¥RAPTOR.44,1
  PARAM MQQUEMGRNAME "p101"
  PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
  PARAM MQDEFAULTPREFIX "$DEV"
  PRI 175
  PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.MQCMSVR
  SECURITY "N"
  TMF ON
  VOLUME ¥RAPTOR.$DATA1.P101D
```

図3. PATHWAY 構成の例 (4/13)

```
SERVER MQS-EC00
  PROCESSTYPE GUARDIAN
  AUTORESTART 10
  CPUS (0,1)
  CREATEDELAY 1 MINS
  DEBUG OFF
  DELETEDELAY 10 MINS
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  LINKDEPTH 255
  MAXSERVERS 1
  NUMSTATIC 1
  OUT ¥RAPTOR.$ZTN0.#PTY001C
  OWNER ¥RAPTOR.44,1
  PARAM MQQUEMGRNAME "p101"
  PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
  PARAM MQDEFAULTPREFIX "$DEV"
  PRI 175
  PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.MQEC
  SECURITY "N"
  TMF ON
  VOLUME ¥RAPTOR.$DATA1.P101D
```

図3. PATHWAY 構成の例 (5/13)


```

SERVER MQS-ECBOSS
  PROCESSTYPE GUARDIAN
  AUTORESTART 10
  CPUS (0,1)
  CREATEDELAY 1 MINS
  DEBUG OFF
  DELETEDELAY 10 MINS
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  LINKDEPTH 255
  MAXSERVERS 1
  NUMSTATIC 1
  OUT ¥RAPTOR.$ZTN0.#PTY001C
  OWNER ¥RAPTOR.44,1
  PARAM MQQUEMGRNAME "p101"
  PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
  PARAM MQDEFAULTPREFIX "$DEV"
  PRI 175
  PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.MQECBOSS
  SECURITY "N"
  TMF ON
  VOLUME ¥RAPTOR.$DATA1.P101D

```

図 3. PATHWAY 構成の例 (6/13)

```

SERVER MQS-MQMSVR00
  PROCESSTYPE GUARDIAN
  AUTORESTART 0
  CPUS (0,1)
  CREATEDELAY 1 MINS
  DEBUG OFF
  DELETEDELAY 10 MINS
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  LINKDEPTH 255
  MAXSERVERS 1
  NUMSTATIC 1
  OUT ¥RAPTOR.$ZTN0.#PTY001C
  OWNER ¥RAPTOR.44,1
  PARAM MQQUEMGRNAME "p101"
  PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
  PARAM MQDEFAULTPREFIX "$DEV"
  PARAM MQMPAGESTORETRIEVE "20"
  PRI 175
  PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.MQMSVR
  SECURITY "N"
  TMF ON
  VOLUME ¥RAPTOR.$DATA1.P101D

```

図 3. PATHWAY 構成の例 (7/13)

```
SERVER MQS-STATUS00
PROCESSTYPE GUARDIAN
AUTORESTART 0
CPUS (0:1)
CREATEDDELAY 1 MINS
DEBUG OFF
DELETEDDELAY 10 MINS
HIGHPIN ON
HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
LINKDEPTH 255
MAXSERVERS 1
NUMSTATIC 1
OUT ¥RAPTOR.$ZTN0.#PTY001C
OWNER ¥RAPTOR.44,1
PARAM MQQUEMGRNAME "p101"
PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
PARAM MQDEFAULTPREFIX "$DEV"
PRI 176
PROCESS $P01S
PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.MQSTSVR
SECURITY "N"
TMF ON
VOLUME ¥RAPTOR.$DATA1.P101D
```

図3. PATHWAY 構成の例 (8/13)

```
SERVER MQS-TCPLIS00
PROCESSTYPE GUARDIAN
AUTORESTART 10
CPUS (0,1)
CREATEDELAY 1 MINS
DEBUG OFF
DELETEDELAY 10 MINS
HIGHPIN ON
HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
LINKDEPTH 255
MAXSERVERS 1
NUMSTATIC 1
OUT ¥RAPTOR.$ZTN0.#PTY001C
OWNER ¥RAPTOR.44,1
PARAM MQQUEMGRNAME "p101"
PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
PARAM MQDEFAULTPREFIX "$DEV"
PRI 175
PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.RUNMQLSR
SECURITY "N"
TMF ON
VOLUME ¥RAPTOR.$DATA1.P101D
```

図 3. PATHWAY 構成の例 (9/13)

```
SERVER MQS-TRIGMON00
  PROCESSTYPE GUARDIAN
  AUTORESTART 10
  CPUS (0,1)
  CREATEDELAY 1 MINS
  DEBUG OFF
  DELETEDELAY 10 MINS
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  LINKDEPTH 255
  MAXSERVERS 1
  NUMSTATIC 1
  OUT ¥RAPTOR.$ZTN0.#PTY001C
  OWNER ¥RAPTOR.44,1
  PARAM MQQUEMGRNAME "p101"
  PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
  PARAM MQDEFAULTPREFIX "$DEV"
  PRI 175
  PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.RUNMQTRM
  SECURITY "N"
  TMF ON
  VOLUME ¥RAPTOR.$DATA1.P101D
```

図3. PATHWAY 構成の例 (10/13)

```
SERVER MQS-QUEUE00
  PROCESSTYPE GUARDIAN
  AUTORESTART 10
  CPUS (0:1)
  CREATEDELAY 1 MINS
  DEBUG OFF
  DELETEDELAY 10 MINS
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  LINKDEPTH 255
  MAXSERVERS 1
  NUMSTATIC 1
  OUT ¥RAPTOR.$ZTN0.#PTY001C
  OWNER ¥RAPTOR.44,1
  PARAM MQQUEMGRNAME "p101"
  PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
  PARAM MQDEFAULTPREFIX "$DEV"
  PRI 176
  PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.MQSSVR
  SECURITY "N"
  TMF ON
  VOLUME ¥RAPTOR.$DATA1.P101D
```

図3. PATHWAY 構成の例 (11/13)

```

SERVER MQS-QMGRSVR
  PROCESSTYPE GUARDIAN
  AUTORESTART 10
  CPUS (0,1)
  CREATEDELAY 1 MINS
  DEBUG OFF
  DELETEDEDELAY 10 MINS
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  LINKDEPTH 255
  MAXSERVERS 1
  NUMSTATIC 1
  OUT ¥RAPTOR.$ZTN0.#PTY001C
  OWNER ¥RAPTOR.44,1
  PARAM MQQUEMGRNAME "p101"
  PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
  PARAM MQDEFAULTPREFIX "$DEV"
  PRI 175
  PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.MQMGRSVR
  SECURITY "N"
  TMF ON
  VOLUME ¥RAPTOR.$DATA1.P101D

```

図 3. PATHWAY 構成の例 (12/13)

```

SERVER MQS-REPSVR00
  PROCESSTYPE GUARDIAN
  AUTORESTART 10
  CPUS (0,1)
  CREATEDELAY 1 MINS
  DEBUG OFF
  DELETEDEDELAY 10 MINS
  HIGHPIN ON
  HOMETERM ¥RAPTOR.$ZTN0.#PTY001C
  LINKDEPTH 255
  MAXSERVERS 1
  NUMSTATIC 1
  OUT ¥RAPTOR.$ZTN0.#PTY001C
  OWNER ¥RAPTOR.44,1
  PARAM MQQUEMGRNAME "p101"
  PARAM MQMACHINIFILE "$DATA1.p101D.UMQSINI"
  PARAM MQDEFAULTPREFIX "$DEV"
  PARAM MQREPMANAGER "YES"
  PRI 175
  PROGRAM ¥RAPTOR.$DEV.ZMQSEXE.MQREPSVR
  SECURITY "N"
  TMF ON
  VOLUME ¥RAPTOR.$DATA1.P101D

```

図 3. PATHWAY 構成の例 (13/13)

PATHWAY サーバー・クラスのパラメーターの変更

PATHWAY サーバー・クラスのパラメーターを変更するには、以下を行います。

1. **endmqm** コマンドを発行して、キュー・マネージャーを停止します。これで、PATHMON プロセスが停止します。
2. サブボリューム *queue managerD* に移動します。ここでは、PATHCTL ファイルが入っています。たとえば、次のようにします。

```
>VOLUME $DATA2.MT01D
```

3. キュー・マネージャーの PATHMON と同じ名前を使用し、NOWAIT オプションを指定して、PATHMON を開始します。たとえば、次のようにします。

```
>PATHMON /NAME $MT01, NOWAIT/
```

4. 新しい PATHMON に対して PATHCOM を開始します。たとえば、次のようにします。

```
>PATHCOM $MT01
```

5. PATHCOM プロンプトから以下のコマンドを発行して、キュー・マネージャーの既存の PATHWAY 構成をロードします。

```
>START PATHWAY COOL
```

6. PATHCOM コマンドを使用して、必要な変更を行います。
7. 以下のコマンドを発行して、PATHWAY システムをシャットダウンします。

```
>SHUTDOWN2
```

8. **strmqm** コマンドを使用して、キュー・マネージャーを開始します。

MQSeries PATHWAY へのユーザー定義サーバー・クラスの追加

PATHCOM を使用して、独自のサーバー・クラス定義を MQSeries PATHWAY 構成に追加できます。ただし、これは推奨されてはいません。サーバーは必ず正常に機能しなければなりません。さもないと、**endmqm** が正しく機能しません。また、キュー・マネージャーが削除されると、ユーザー定義サーバー・クラスもなくなるので注意してください。不都合を最小限にとどめるために、再使用可能スクリプトを作成するようにお勧めします。

第4章 キュー・マネージャーの管理

この章では、MQSeries キュー・マネージャーの管理のあらゆる面について説明します。

この章には、次の節があります。

- 『はじめに』
- 58ページの『キュー・マネージャー作成のガイドライン』
- 61ページの『キュー・マネージャーのプロパティの変更』
- 76ページの『デフォルトのキュー・サーバー名』
- 77ページの『デフォルト以外のキュー・サーバーの追加と除去』
- 79ページの『ボリューム構造』
- 83ページの『キュー・マネージャーでの作業』
- 83ページの『デフォルト・キュー・マネージャーの作成』
- 85ページの『MQSeries プリンシパルの作成』
- 85ページの『RDF 使用可能になっているキュー・マネージャーでの `cleanrdf` の実行』
- 86ページの『キュー・マネージャーの開始』
- 86ページの『デフォルトおよびシステム・オブジェクトの復元』
- 87ページの『オブジェクト・ファイルの表示』
- 87ページの『キュー・マネージャーの停止』
- 88ページの『キュー・マネージャーの再始動』
- 89ページの『既存のキュー・マネージャーのデフォルト・キュー・マネージャーへの指定』
- 89ページの『キュー・マネージャーの削除』
- 90ページの『メッセージ・キュー管理 (MQM) 機能の使用』

はじめに

メッセージやキューを使用するためには、まず、少なくとも 1 つキュー・マネージャーを作成する必要があります。それで、インストールのプロセスが完了したなら、MQSeries 制御コマンドを使用してキュー・マネージャーを作成し、MQSeries プリンシパルを作成し、それからそのキュー・マネージャーを開始してください。

キュー・マネージャーの作成時には、必要なデフォルト・オブジェクトとシステム・オブジェクトが自動的に作成されます。デフォルト・オブジェクトは、

始めに

作成されるすべてのオブジェクト定義の基礎となるものです。また、システム・オブジェクトはキュー・マネージャーの操作に必要です。デフォルト・システム・オブジェクトの復元については、86ページの『デフォルトおよびシステム・オブジェクトの復元』を参照してください。

MQSeries for Compaq NSK で使用できるコマンドと、それら呼び出すための様々なメソッドについては、37ページの『第3章 MQSeries コマンド・セットの使用』を参照してください。

キュー・マネージャー作成のガイドライン

キュー・マネージャーは、自身が所有しているキューなど、関連付けられているリソースを管理します。キュー・マネージャーは、メッセージ・キューイング・インターフェース (MQI) 呼び出しやコマンドによって MQSeries オブジェクトの作成、修正、表示、および削除を行えるよう、アプリケーションに対してキューイング・サービスを提供します。キュー・マネージャーの作成には、**crtmqm** コマンドを使用します。ここでは、キュー・マネージャーを作成する際に考慮できる点をリストします。

- キュー・マネージャーには固有な名前を指定してください。

キュー・マネージャーを作成するときには、必ず、同じ名前を持つキュー・マネージャーがネットワーク内に他に存在していないことを確かめてください。キュー・マネージャーの名前が作成時に検査されることはありません。固有でない名前が使用された場合は、分散キューイング用のチャンネルが作成できなくなります。

名前が確実に固有なものとなるようにするには、各キュー・マネージャー名の先頭にそれぞれのノード名を付けることができます。たとえば、ノードの名前が `accounts` の場合は、キュー・マネージャーの名前を `accounts.saturn.queue.manager` にすることもできます。ここで、`saturn` は特定のキュー・マネージャーを識別し、`queue.manager` はすべてのキュー・マネージャーに与えることができる拡張子です。この拡張子は省略することもできます。ただしその場合、`accounts.saturn` と `accounts.saturn.queue.manager` とは異なる キュー・マネージャー名になります。

注: 制御コマンドでは、キュー・マネージャー名の太文字小文字が区別されます。たとえば、`jupiter.queue.manager` と `JUPITER.queue.manager` という 2 つのキュー・マネージャーを作成できる、ということです。

- キュー・マネージャーの数には制限を設けてください。

MQSeries for Compaq NSK では、リソースに余裕がある限り、いくつでもキュー・マネージャーを作成できます。しかし、各キュー・マネージャーはそれぞれが専用のリソースを必要とするため、一般的には、それぞれが 10 個のキューを持つ 10 個のキュー・マネージャーを作成するよりも、ノードに 100 個のキューを持つ 1 個のキュー・マネージャーを作成する方がより効率的です。多くのノードは 1 つのキュー・マネージャーで稼働させられますが、大きなサーバーの場合は、複数のキュー・マネージャーが使用される場合もあります。また、パフォーマンスや機能上の特殊な要件によって、複数のキュー・マネージャーが必要とされる場合もあります。

- **デフォルトのキュー・マネージャーを指定してください。**

デフォルト・キュー・マネージャーを持たない MQSeries をノード上に構成することは不可能ではありませんが、各ノードにデフォルト・キュー・マネージャーを 1 つずつ持つのが望ましいかたちです。

デフォルト・キュー・マネージャーとは、アプリケーションが MQCONN 呼び出しでキュー・マネージャー名を指定しなかった場合に、そのアプリケーションが接続されるキュー・マネージャーのことです。また、これは、キュー・マネージャー名を指定せずに `runmqsc` コマンドを呼び出したときに MQSC コマンドを処理するキュー・マネージャーのことでもあります。

あるキュー・マネージャーをデフォルト・キュー・マネージャーとして指定すると、ノードについての既存のデフォルト・キュー・マネージャーの指定は置き換えられます。

デフォルト・キュー・マネージャーを変更する場合、他のユーザーまたはアプリケーションに影響を与える可能性があります。変更しても、現在接続されているアプリケーションに影響はありません。それらのアプリケーションは、以後の MQI 呼び出しで元の接続呼び出しからのハンドルを使用できるためです。このハンドルにより、呼び出しは確実に同じキュー・マネージャーに対して出されます。変更後に接続されるアプリケーションは、新しいデフォルト・キュー・マネージャーに接続されます。

デフォルト・キュー・マネージャーを作成するには、`-q` フラグを指定して `crtmqm` コマンドを実行します。このコマンドとそのパラメーターについての詳細は、342ページの『`crtmqm` (キュー・マネージャーの作成)』を参照してください。

- **送達不能キューを指定してください。**

送達不能キューとは、正しい宛先に送ることができなかったメッセージを入れるローカル・キューのことです。

重要: 送達不能キューは、ネットワーク内の各キュー・マネージャーごとに必要です。このような障害は、アプリケーション・プログラムのエラーになる場合があります。この場合、チャンネルはクローズされ、管理コマンドへの

キュー・マネージャーの作成

応答は失敗します。たとえば、アプリケーションが別のキュー・マネージャーのキューにメッセージを入れようとしたときに、誤ったキュー名が与えられると、チャンネルが停止し、メッセージは伝送キューに残ったままになります。そうなると、他のアプリケーションは、このチャンネルを使用して自分のメッセージを送ることができなくなります。

しかし、キュー・マネージャーに送達不能キューがあれば、チャンネルは影響を受けません。未送達メッセージは受信側の送達不能キューに入れられ、チャンネルとその伝送キューはそのまま使用できます。

したがって、キュー・マネージャーを作成するときには、`-u` フラグを使用して送達不能キューの名前を指定する必要があります。MQSC コマンドを使用してキュー・マネージャーの属性を更新し、使用する送達不能キューを指定することもできます。MQSC ALTER コマンドの例については、124ページの『キュー・マネージャーの属性の更新』を参照してください。

送達不能キュー上にメッセージが見つかった場合、MQSeries と共に提供されている送達不能キュー・ハンドラーを使用して、それらのメッセージを処理することができます。送達不能キュー・ハンドラーの詳細と、送達不能キューを指定していれば送達不能キューに入れられるようなメッセージの数を少なくする方法については、205ページの『第9章 MQSeries 送達不能キュー・ハンドラー』を参照してください。

- **デフォルト伝送キューを指定してください。**

伝送キューとは、リモート・キュー・マネージャーへ送られる途中のメッセージが、伝送されるまでの間キューイングされるローカル・キューのことです。デフォルト伝送キューは、その伝送キューが明示的に定義されなかった場合に使用されるキューです。各キュー・マネージャーには、デフォルト伝送キューを 1 つ割り当てることができます。

キュー・マネージャーを作成するときには、`-d` フラグを使用してデフォルト伝送キューの名前を指定します。ただし、この `-d` フラグによって実際にキューが作成されるわけではないので、後でこれを作成する必要があります。詳細については、131ページの『ローカル・キューの取り扱い』を参照してください。

- **キュー・マネージャーを作成した後は、構成ファイルのバックアップを作成してください。**

MQSeries の構成ファイル (MQSINI) は、MQSeries のインストール時に作成されます。このファイルには、キュー・マネージャーを作成したり削除したりするたびに更新される、キュー・マネージャーのリストが含まれます。

MQSINI ファイルは、インストール単位につき 1 つ作成されます。デフォルトでは、MQSINI は \$SYSTEM.ZMQSSYSにあります。

キュー・マネージャー構成ファイル (QMINI) は、新しいキュー・マネージャーが作成されるときに自動的に作成されます。このファイルには、そのキュー・マネージャーのための構成パラメーターが含まれています。

これらのファイルには、バックアップを作成するようにしてください。問題の原因となる別のキュー・マネージャーを作成した場合、問題の原因を取り除いたときにバックアップを回復させることができます。新しいキュー・マネージャーを作成するたびに、構成ファイルのバックアップを取る必要があります。

構成ファイルについての詳細は、243ページの『第13章 構成ファイル』を参照してください。

キュー・マネージャーの作成後の構成ファイルのバックアップ

バックアップの必要な構成ファイルには、MQSINI と QMINI の 2 つがあります。

1. MQSeries の構成ファイル (MQSINI) は、MQSeries のインストール時に作成されます。このファイルには、キュー・マネージャーを作成したり削除したりするたびに更新される、キュー・マネージャーのリストが含まれます。MQSINI ファイルは、インストール単位につき 1 つ作成されます。デフォルトでは、MQSINI は \$SYSTEM.ZMQSSYS にあります。
2. キュー・マネージャー構成ファイル (QMINI) は、新しいキュー・マネージャーが作成されるときに自動的に作成されます。このファイルには、そのキュー・マネージャーの構成パラメーターが含まれます。

これらのファイルには、バックアップを作成するようにしてください。問題の原因となる別のキュー・マネージャーを作成した場合、問題の原因を取り除いたときにバックアップを回復させることができます。新しいキュー・マネージャーを作成するたびに、構成ファイルのバックアップを取る必要があります。

キュー・マネージャーのプロパティーの変更

キュー・マネージャーのプロパティーの多くは、キュー・マネージャーの作成時に変更できます。また、大抵は変更を有効にするためにキュー・マネージャーを一度停止させてから再始動する必要があるものの、いくつかのプロパティーについては、キュー・マネージャーを作成した後でこれを変更することも可能です。

この節の残りの部分では、変更されることがあるいくつかのプロパティーについて説明します。

キュー・マネージャーの作成

キュー・マネージャーのホーム・ボリューム

キューを含むすべてのデータベースは、このボリュームで作成されます。(ただし、327ページの『altmqfls (キュー・ファイル属性の更新)』で説明されているように、個々のキューは、作成した後で **altmqfls** 制御コマンドを使って別のボリュームに移せます。)

デフォルト値は、MQSINI ファイルの AllQueueManagers スタンザにある QMDefaultVolume エントリーからとられます。ただしこれは、**crtmqm** コマンドに `-p DefaultPrefix` パラメーターが指定されていれば、それによって指定変更されます。

ホーム・ボリュームは、キュー・マネージャーを作成するときのみ 指定できます。作成の後でこれを変更することはできません。

キュー・マネージャーごとの EC プロセスの数

デフォルトでは、キュー・マネージャーごとに 1 つの EC プロセスがあります。キュー・マネージャーごとの EC プロセスの数は、**crtmqm** コマンドの `-e NumECs` パラメーターで指定します。

各 EC は、その EC そのものと同じ CPU を使用するアプリケーションやチャネルに対してメッセージングやキューイングを実行する、サーバー・プロセスのサブセットを扱います。EC の数は、EC ごとのアプリケーションの数が多い限りは CPU につき 1 つにすることが勧められていますが、EC ごとのアプリケーションの数が多い場合は、同じ CPU 上にさらに別の EC を稼働させたほうがよい場合もあります。たとえば、大規模なインストールでは、たくさんのアプリケーションやチャネルが同時に扱われることがあるため、より多くの EC プロセスが必要です (複数の CPU 間に分散させられることが多い)。

キュー・マネージャーの作成時には、**crtmqm** コマンドで指定された EC ごとに TS/MP サーバー・クラスが作成されます。指定された EC サーバー・クラスは、ラウンドロビン式にシステム内の全 CPU に分散されます。たとえば、4 つの CPU があるシステムに 8 つの EC プロセスが指定された場合、デフォルトでは、各 CPU に 2 つずつ EC プロセスが分散されます。

キュー・マネージャーの QMINI ファイルにある ECBoss スタンザの ExpectedNumECs フィールドには、作成時に指定された EC プロセスの数が設定されます。この値は、常に TS/MP 構成と一致していなければなりません。

キュー・マネージャー内の EC プロセスの数は、作成の後でも、TS/MP サーバー・クラスを追加または削除することによって、また QMINI ファイル内の

ECBoss スタンザの `ExpectedNumECs` エントリーに修正を加えることによって変更できます。ただしこれは、キュー・マネージャーが停止しているときにしか行えません。

システム・ロード・バランシング

EC Boss は、キュー・マネージャーのワークロードを EC 間で分散させる役割を果たします。キュー・マネージャーの処理の負荷は、EC の構成が適切であれば、複数の CPU にバランスよく分散させることができます。

ローカル・アプリケーションから新しい接続の要求を受け取ったときや、チャンネルが開始されるときは、この EC Boss が、アクティブになる LQMA や MCA ができるだけ少なくなるように要求を EC に割り振ります。

キュー・マネージャーのホーム端末

キュー・マネージャーのサーバー・プロセスを含むすべての Compaq NSK プロセスには、ホーム端末があります。端末は、必ず存在していなければならない、かつ休止状態になっていなければならない。一般的に、キュー・マネージャーのホーム端末は出力用には使用しません。Compaq Virtual Hometerm Service (VHS) 製品を含め、有効な端末装置であればどの端末でもホーム端末にすることができます。

Compaq NonStop Kernel で使用できる物理端末の基本オープンナーは、最大で 255 までです。したがって、この限界を超えることがないように、慎重な計画が必要です。255 を超えるオープンナーのサポートが必要な場合は、Compaq の VHS 製品を使用することが勧められています。

キュー・マネージャーのホーム端末は、`crtmqm` コマンドの `-o` `HomeTerminalName` パラメーターで識別します。デフォルトはありません。このパラメーターは必須です。

すべての TS/MP サーバー・クラスの HOMETERM 属性と OUT 属性は、指定された端末装置に設定されます。これらの属性は、サーバー・クラスが休止状態にあるとき (通常は、キュー・マネージャーが停止しているとき) であれば、いつでも調整できます。

キュー・マネージャーのホーム端末を変更するためには、QMINI ファイルの Configuration スタンザにある `HomeTerminalName` エントリーにも修正を加える必要があります。

キュー・マネージャーの作成

キュー・マネージャーの PATHMON プロセス名

各キュー・マネージャーは、それぞれ独自の TS/MP (Pathway) 構成の下で稼働します。そしてこれを制御するプロセスのことを、PATHMON プロセスといいます。このプロセスには、キュー・マネージャーごとに固有な名前を指定する必要があります。さらに、この名前はシステム内でも固有のものでなければなりません。

PATHMON プロセス名は、**crtmqm** コマンドの `-n PATHMONProcessName` パラメーターで指定します。デフォルトはありません。このパラメーターは必須です。

システム内で固有であり、かつ制御するキュー・マネージャーと結び付けやすいプロセス名を指定してください。

キュー・マネージャーの PATHMON プロセス名は、次のようにして変更できます。

1. キュー・マネージャーを停止させます。
2. デフォルトのボリュームとサブボリュームを、キュー・マネージャー・データ・ファイル (通常は <QMgr name>D) の位置に、設定します。
3. キュー・マネージャーの QMINI ファイルで PathmonProcName エントリーを変更して、新しいプロセス名を指定します。
4. その新しい プロセス名を用いて、そのキュー・マネージャーの PATHMON を作成します。

TACL から、次のコマンドを実行します。

```
PATHMON /name $<newname>, nowait/
```

5. 新しく名前を付けた PATHMON に対して PATHCOM を実行します。

TACL から、次のコマンドを実行します。

```
PATHCOM $<newname>
```

6. キュー・マネージャーの Pathway 構成をロードし、PATHMON プロセスの名前が変更されたことを確認します。

PATHCOM から、次のコマンドを実行します。


```
START PATHWAY COOL
```

構成がロードされる際、新しい PATHMON プロセスの名前が構成ファイルに保管されているものと異なっていることを知らせる警告が出されます。そしてその後、処理を進めてもよいかどうかを確認するプロンプトが出されません。プロンプトで `y` を入力すると、構成のロードが完了します。

7. 新しい PATHWAY 構成情報をデータベースに保管します。
PATHCOM から、次のコマンドを実行します。

```
SHUTDOWN2  
EXIT
```

QMINI ファイルの Configuration スタンザでは、PathmonProcName エントリが変更されているはずですが。

これで、PATHMON プロセス名の変更は完了しました。次に **strmqm** でキュー・マネージャーが開始されるときからは、新しい PATHMON プロセス名が使用されます。

キュー・マネージャーの CCSID

これは、キュー・マネージャーがメッセージに関する情報を保管する際に使用する文字セットのコード化文字セット ID (CCSID) です。

CCSID は、**crtmqm** コマンドの `-I CCSID` パラメーターで指定します。デフォルトは 819 です。

キュー・マネージャーの CCSID は、キュー・マネージャーを作成した後でも、**runmqsc**、MQM 機能、または PCF コマンドを使用していつでも変更できます。

EBCDIC データ変換の制御

変換の際に EBCDIC 改行 (NL) 文字をどのように扱うかは、MQSINI 構成ファイルの AllQueueManagers スタンザにある、ConvEBCDICNewline の項目を使って制御できます。

ConvEBCDICNewline=NL_TO_LF|TABLE|ISO

EBCDIC コード・ページには、ASCII コード・ページではサポートされない改行 (NL) 文字が含まれています。ただし、ISO 仕様の ASCII

キュー・マネージャーの作成

には、この改行文字と同等の文字が含まれているものもいくつかあります。ConvEBCDICNewline 属性には、EBCDIC の NL 文字を ASCII 形式に変換するときに MQSeries が使用するメソッドを指定します。

NL_TO_LF

EBCDIC から ASCII へのすべての変換について、EBCDIC の NL 文字 (X'15') が ASCII の改行 (LF) 文字 (X'0A') に変換されるようにするには、NL_TO_LF を指定します。

NL_TO_LF はデフォルトです。

TABLE

ご使用のプラットフォームで EBCDIC から ASCII への変換に使用している変換テーブルに従って、EBCDIC の NL 文字が変換されるようにするには、TABLE を使用します。このタイプの変換の結果は、プラットフォームと言語によって異なる場合があるので、注意が必要です。同じプラットフォーム上でも、使用する CCSID が異なれば動作が違ってくる場合があります。

ISO 次の場合は ISO を指定します。

- TABLE メソッドを使用して ISO CCSID を変換したい場合
- NL_TO_CF メソッドを使用してその他のすべての CCSID を変換したい場合

使用できる ISO CCSID は、表2に示されています。

表2. 使用できる ISO CCSID のリスト

CCSID	コード・セット
819	ISO8859-1
912	ISO8859-2
915	ISO8859-5
1089	ISO8859-6
813	ISO8859-7
916	ISO8859-8
920	ISO8859-9
1051	roman8

ASCII CCSID が ISO サブセットの 1 つではない場合は、デフォルトにより、ConvEBCDICNewline は NL_TO_LF に設定されます。

データ変換の詳細については、*MQSeries アプリケーション・プログラミング・ガイド*を参照してください。

キュー・マネージャーの EMS コレクター

キュー・マネージャーは、必要に応じて、代替のコントローラーを使用するように構成できます。EMS イベントは、デフォルトで \$0 に送信されます。QMINI ファイルの Configuration スタンザにある EMSCollectorName エントリーに、このキュー・マネージャーの EMS コレクターの名前を指定してください。

EMS コレクターは、このエントリーの値を変えることによっていつでも変更できますが、変更を有効にするためにはキュー・マネージャーを再始動する必要があります。

キュー・マネージャー内の各 EC によって作動可能な状態に保持されるエージェントのプール

EC は、アイドル状態、つまり新しい作業に割り当てられる状態にあるエージェント・プロセスのプールを、4 つの基本的なエージェントのタイプごとに保守できます。これらのプールのサイズを構成することによって、新しい作業への応答時間とリソース使用率との間のバランスを適切なものにできます。アイドル状態で保持するプロセスの数は、QMINI ファイルの Configuration スタンザで以下のフィールド値を変更することによって変更できます。

MinIdleMCALU62Responders

アイドル状態で保守する SNA LU 6.2 の応答側 MCA の最小数を指定します。デフォルト値は 0 です。

MinIdleMCATCPResponders

アイドル状態で保守する TCP/IP の応答側 MCA の最小数を指定します。デフォルト値は 0 です。

MinIdleMCACallers

アイドル状態で保守する呼び出し側 MCA (プロトコル固有ではない) の最小数を指定します。デフォルト値は 0 です。

MinIdleLQMAgents

アイドル状態で保守するローカル・キュー・マネージャー・エージェント (LQMA) の最小数を指定します。デフォルト値は 1 です。

これらのフィールドで指定されたプロセスの数は、各キュー・マネージャーではなく、各 EC に適用されます。ですから、キュー・マネージャーに 2 つの

キュー・マネージャーの作成

EC がある場合、そのキュー・マネージャーにはデフォルトで最低 2 つの LQMA がアイドル状態で保守されることになります。

これらの値はいつでも変更できますが、変更を有効にするためにはキュー・マネージャーを再始動する必要があります。

アイドル状態のエージェントとプロセスの再利用の最大数

デフォルトでは、キュー・マネージャーは、各タイプのエージェント・プロセスを 10 個までアイドル状態に保つことができます。この値は、QMINI ファイルの Configuration スタンザにある MaxIdleAgents エントリーで制御されます。

MaxIdleAgentReuse エントリーは、何回エージェント・プロセスを再利用したらそれを新しいエージェント・プロセスに置き換えるかを決定します。デフォルトでは、MaxIdleAgentReuse は 10 に設定されています。

キュー・マネージャー・プロセスのプロセス優先順位

MQSeries と他のアプリケーションとの間でリソースをバランスよく配分するため、優先順位の変更が必要になる場合があります。TS/MP サーバー・クラスのプロセス優先順位は、キュー・マネージャーが停止しているときに TS/MP オブジェクトに調整を加えることで変更できます。

エージェント・プロセスのプロセス優先順位は、QMINI ファイルの EC スタンザにある MCAAgentPriority フィールドと LQMAgentPriority フィールドを使って変更できます。

状況サーバーとキュー・サーバー (デフォルトのプロセス優先順位が 176) は別にして、TS/MP によって構成されるプロセスのデフォルトの優先順位はすべて 175 になっています。デフォルトでは、MCA と LQMA 両方のプロセス優先順位も 165 です。

状況サーバーとキュー・サーバーの優先順位を一番高くし、その次に EC Boss と EC、そしてさらにその次に MCA と LQMA という優先順位が保たれるようにしてください。

キュー・マネージャーのチャンネルの最大数

1 つのキュー・マネージャーのために一度に制御できるチャンネルの数には、限度があります。この限度が高すぎると、このパラメーターがチャンネル状況テーブル (多くの検索操作がここで実行される) のサイズを指示する際にパフォーマンスに影響が出ます。しかし、逆にこの限度が低すぎると、今度はアプリケ

ーションに必要なだけのチャンネルを制御できなくなる可能性があります。同時に制御できるチャンネルの最大数は、QMINI ファイルの Channels スタンザにある MaxChannels フィールドで定義されます。

作成時のデフォルトは 10 です。作成時のデフォルトを変更することはできません。

QMINI ファイルにある Channels スタンザの MaxChannels エントリーはいつでも変更できますが、変更を有効にするためにはキュー・マネージャーを再始動する必要があります。

キュー・マネージャーのアクティブ・チャンネルの最大数

キュー・マネージャー内で同時にアクティブ (実行中) にできるチャンネルの数には制限があります。この制限を用いて、チャンネルによるシステム・リソース要求の限度を制限できます。キュー・マネージャーのアクティブ・チャンネルの最大数は、QMINI ファイルの Channels スタンザにある MaxActiveChannels エントリーで定義します。

作成時のデフォルトは 10 です。作成時のデフォルトを変更することはできません。

QMINI ファイルの MaxActiveChannels エントリーはいつでも変更できますが、変更を有効にするためにはキュー・マネージャーを再始動する必要があります。

MQSeries で使用する Guardian セグメント ID

MQSeries は、自身のプロセスとアプリケーション・プログラムのプロセスの両方に Guardian メモリー・セグメントを割り振ります。Guardian セグメント ID というのは、MQSeries がこれらのセグメントに使用する ID で、MQSINI 構成ファイルで定義された範囲から割り振られます。MQSeries がメモリー・セグメントを割り振る際に使用するこのセグメント ID の範囲は、MQSINI 構成ファイルの AllQueueManagers スタンザにある NSKSegidRange の項目で定義されます。

この範囲は、MQSeries のインストール時にデフォルトで
・NSKSegidRange=10-20 に設定されます。

これらのメモリー・セグメントはアプリケーション・プログラムのプロセスにも割り振られるため、アプリケーションがこの範囲にセグメントを割り振ることのないようにしてください。ただし、MQSINI ファイルの NSKSegidRange 項目に変更を加えれば、MQSerie の方に別の範囲にセグメントを割り振らせるこ

キュー・マネージャーの作成

ともできます。この項目を変更した後は、新しい値を有効にするために、キュー・マネージャーを一度停止させてから再始動してください。

NSKSegidRange パラメーターに定義される範囲は、0-1023 という Guardian セグメント ID の制限に収まっていなければなりません。また、NSKSegidRange パラメーターには、MQSeries が使用するために少なくとも 10 個のセグメント ID を含む範囲を定義してください。

デフォルト TCP/IP ポート

QMINI の TCPConfig スタンザにある TCPPort エントリーでは、発信チャンネルのデフォルト・ポート番号を定義します。デフォルトでは、ポート番号 1414 が使用されます。このデフォルトは、チャンネルの CONNAME フィールドで指定されたポート番号によって指定変更されます。

キュー・マネージャーが listen する TCP/IP ポート

TCP/IP チャンネルを使用するキュー・マネージャーは、1 つまたは複数の TCP/IP ポートで着信接続を listen するように構成できます。listen するポートの数やこのキュー・マネージャーに割り当てるポートの数は、TCPNumListenerPorts や、QMINI ファイルの TCPConfig スタンザにあるエントリーで定義します。実際にどのような QMINI エントリーがあるかについては、491ページの『TCP/IP 通信の例』を参照してください。

キュー・マネージャーは、1 つのシステムに複数置くことができます。それで、システム上の各キュー・マネージャーに listen 用の TCP/IP ポートのセットを割り当てるときには、ポートが重複しないようにしなければなりません。各キュー・マネージャーに割り当てる TCP/IP ポートは、TCP/IP 接続要求の着信率が低ければ 1 つだけでも構いませんし、構成が大規模なら複数にすることもできます。デフォルトの TCP/IP ポートは 1414 です。そしてデフォルトでは、キュー・マネージャーはこのポートでだけ listen を行うように作成されません。

listen 用ポートのリストは、QMINI ファイルの TCPConfig スタンザを編集することによって変更できます。変更後は、キュー・マネージャーを再始動してください。なお、複数のポートで listen を行うためには、キュー・マネージャーも、TS/MP を使って付加的な TCP/IP リスナー・サーバー・クラスで構成される必要があります。この操作は、PATHCOM を使用して手動で行います。

別の方法として、**runmqlsr** コマンドでポート番号を指定する方法もあります (375ページの『runmqlsr (リスナーの実行)』に説明があります)。リスナー・サ

サーバー・クラス・プログラムの環境にパラメーター `MQLISTENPORTNUM` が定義されている場合、`TCPListenerPort` の値はそのリスナー・プログラムによって指定変更されます。

キュー・マネージャーで使用する TCP/IP プロセス

Compaq TCP/IP 製品へのインターフェースは、TCP/IP プロセスとして知られるサーバー・プロセスを介します。デフォルトでは、システムのデフォルト `$ZTC0` が使用されます。キュー・マネージャーの作成時にこのデフォルトを指定変更することはできません。システムのデフォルトでは用が足りないときは、特定の TCP/IP プロセスを使用するためにキュー・マネージャーのチャンネルを構成できます。システムのデフォルト TCP/IP プロセス `$ZTC0` が適切でないのであれば、TCP/IP リスナーの TS/MP サーバー・クラス構成を手動で変更する必要があります。なお、このサーバー・クラス構成は、手動で変更すれば、リスナーに特定のポートで `listen` させ、`QMINI` ファイルで定義されたポートを指定変更することもできます。

サーバー・クラス `MQS-TCPLIS00` には、必要な代替 TCP/IP プロセスの名前を示す `DEFINE TCP/IP^PROCESS^NAME` を加えなければなりません。複数の EC を使用する場合は、それらすべてを更新する必要がありますので注意してください。詳細については、Compaq NSK TCP/IP 製品のマニュアルを参照してください。

TS/MP サーバー・クラスに変更を加えられるのは、キュー・マネージャーが停止しているときだけです。

MQSeries for Compaq NSK は、キュー・マネージャーごとに複数の TCP/IP プロセスをサポートできます。また、複数の TCP/IP ポートやリスナー・プロセスもサポートします。

デフォルト以外の TCP/IP プロセスやポートのための MQS-TCPLISnn サーバー・クラスの再構成

デフォルト以外の TCP/IP プロセスやポートのために `MQS-TCPLISnn` サーバー・クラスを再構成する場合は、54ページの『PATHWAY サーバー・クラスのパラメーターの変更』にある一般的な指示に従ってください。

デフォルト以外の TCP/IP プロセスを指定するには、次の `PATHCOM` コマンドを使用します。

キュー・マネージャーの作成

```
ALTER SERVER MQS-TCPLIS00, (DEFINE =TCP/IP^PROCESS^NAME, FILE $ZZZZ)
```

ここで、\$ZZZZ には、必要な TCP/IP プロセスの名前が入ります。

特定のポートを指定するには、次の PATHCOM コマンドを使用します。

```
ALTER SERVER MQS-TCPLIS00, PARAM MQLISTENPORTNUM "nnnn"
```

この nnnn の部分には、listen に使用するポートの番号を入れます。

デフォルトの \$ZTC0 とは別の TCP/IP プロセス名を必要とする TACL (runmqlsr) からリスナーを稼働させる場合は、そのリスナーを稼働させる TACL 環境に以下を追加してください。

```
ADD DEFINE =TCP/IP^PROCESS^NAME, FILE $ZZZZ)
```

ここで、\$ZZZZ には、必要な TCP/IP プロセスの名前が入ります。

注: 個々の TCP/IP リスナー・サーバー・クラスでポートごとに別の TCP/IP プロセスを使用したい場合は、それぞれのサーバー・クラスのインスタンスでポートと TCP/IP プロセスの両方を定義する必要があります。これが定義されない限り、QMINI ファイルに定義されたポートでどの TCP/IP プロセスが使用されるという確証はありません。

なお、リスナー・プログラムの環境にパラメーター MQLISTENPORTNUM が定義されていない場合は、QMINI ファイルからポートが取得されます。

スワップ・スペースの割り振り

MQSeries for Compaq NSK は、QMINI 構成ファイル内の様々な実行可能ノードの ExtPoolSize 値に応じてスワップ・スペースを割り振ります。したがって、デフォルトの QMINI ファイルを使用するキュー・マネージャーの場合であれば、10 の発信チャンネル、20 のエージェント、そして 10 の TCP 応答側が稼働し、そのそれぞれにデフォルトで 300 KB が割り振られるので、最低 12 MB (40 * 300 KB) のスワップ・スペースが必要になります。この所要量を減らすためには、QMINI ファイル内の値を低くできます。この ExtPoolSize 値は、初めに割り振られたメモリーが使い果たされたときに割り振られる、最

低限の予備のメモリーです。この値は、キュー・マネージャーの最大メッセージ・サイズより大きくする必要はありません。

MQSeries 実行可能ノードのスワップ割り振りを制御する方法は、以下を追加することで MQ Pathway サーバー・クラスを調整するというものです。

```
DEFINE =_DEFAULTS, CLASS DEFAULTS, VOLUME volume.qmD subvolume, SWAP volume
```

DEFINE を追加すると、EC サーバーによって作成されるエージェント・プロセスにもここで定義されたスワップの量が継承されるため、複数の EC サーバー・クラスが使用されるときでもスワップの使用率にいくらかのスケラビリティが生まれます。

デフォルトの状況サーバー名

キュー・マネージャーを作成する際には、デフォルトの状況サーバー・プロセスのペアに固有のプロセス名を指定する必要があります。デフォルトの状況サーバー名は、**crtmqm** コマンドの必須パラメーター **-s StatusServerName** で指定します。このパラメーターには、デフォルト値はありません。システム内で固有であり、かつそれが属しているキュー・マネージャーと結び付けやすいプロセス名を指定してください。

キュー・マネージャーのデフォルトの状況サーバー・プロセス名は、次のようにして変更できます。

1. キュー・マネージャーを停止させます。
2. デフォルトのボリュームとサブボリュームを、キュー・マネージャー・データ・ファイル (通常は <QMgrName>D) の位置に、設定します。
3. 新しいプロセス名を記録するには、キュー・マネージャーの QMINI ファイルにある Configuration スタンザの DefaultStatusServerName エントリーを変更します。
4. TACL から次のコマンドを入力し、キュー・マネージャーに対して PATHMON を実行します。

```
PATHMON /name $<pmon>, nowait/
```

ここで、<pmon> には、キュー・マネージャーの PATHMON プロセス名が入ります。

キュー・マネージャーの作成

5. TACL から次のコマンドを入力し、PATHMON プロセスに対して PATHCOM を実行します。

```
PATHCOM $<pmon>
```

6. PATHCOM から次のコマンドを入力し、キュー・マネージャーの Pathway 構成をロードします。

```
START PATHWAY COOL
```

7. PATHCOM から次のコマンドを入力し、サーバー MQS-STATUS00 を調整してプロセスをリセットします。

```
ALTER SERVER MQS-STATUS00, DELETE PROCESS $<oldname>  
ALTER SERVER MQS-STATUS00, PROCESS $<newname>
```

8. PATHCOM から次のコマンドを入力して、調整した構成を保管します。

```
SHUTDOWN2  
EXIT
```

これで、デフォルトの状況サーバー・プロセス名の変更が完了しました。次に **strmqm** コマンドでキュー・マネージャーが開始されるときからは、新しいデフォルトの状況サーバー・プロセス名が使用されます。

デフォルト以外の状況サーバーの追加と除去

付加的な状況サーバーを追加する場合は、すでにあるデフォルトの状況サーバーをテンプレートとして用います。新しい状況サーバーを追加するためにキュー・マネージャーを停止させる必要はありません。

次のような手順で新しい状況サーバーを追加します。

注: この手順は、キュー・マネージャーが稼働している場合を想定したものです。キュー・マネージャーが稼働していない場合は、この手順を開始する前にまず PATHMON を開始し、PATHWAY 構成をロードしてください。その場合は、

78ページの5 のステップを省略し、最後に SHUTDOWN2 コマンドを使用して Pathway 構成をディスクに保管します。

1. TACL から次のコマンドを入力し、PATHMON プロセスに対して PATHCOM を実行します。

```
PATHCOM $<pmon>
```

2. PATHCOM から次のコマンドを入力し、デフォルトの状況サーバー・クラスをテンプレートとしてベースに置いた属性の作業セットを作成してください。

```
RESET SERVER  
SET LIKE MQS-STATUS00
```

3. PATHCOM から次のコマンドを入力し、新しいサーバーの作業セットに変更を加えます。

```
SET SERVER CPUS(n:m)  
RESET SERVER PROCESS $<default status server process name>  
SET SERVER PROCESS $<new status server name>
```

4. PATHCOM から次のコマンドを入力し、新しいサーバー・クラス名で新しいサーバーを追加します。

```
ADD SERVER MQS-STATUSxx
```

5. 次のコマンドを入力して、新しい状況サーバーを始動させて使用できるようにした後、PATHCOM を終了します。

```
START SERVER MQS-STATUSxx  
EXIT
```

追加のサーバー・クラスに推奨されている命名規則は、MQS-STATUS01 や MQS-STATUS02 などです。ただし、この規則を使用しなければいけないとい

キュー・マネージャーの作成

うわけではありません。サーバー・クラス名の先頭に `MQS-STATUS` という文字ストリングがあれば、サーバー・クラスは `strmqm` によって開始されません。

状況サーバーが追加され、開始された (明示的に `PATHCOM` を使用した場合も、暗黙的に `strmqm` で開始された場合も含む) なら、`altmqfls` を使用して新しい状況サーバーにオブジェクトを再割り当てできます。オブジェクトの再割り当てについては、137ページの『状況サーバーおよびキュー・サーバーにオブジェクトを再割り当てする』を参照してください。

状況サーバーを除去するときは、まず、この状況サーバーに対して構成されたすべてのオブジェクトが、削除されたか別の状況サーバーに再割り当てされていることを確認してください。なお、デフォルトの状況サーバーは絶対に削除しないでください。これを削除すると、キュー・マネージャーは作動しなくなります。

デフォルトのキュー・サーバー名

キュー・マネージャーを作成する際には、デフォルトのキュー・サーバー・プロセスのペアに固有のプロセス名を指定する必要があります。デフォルトのキュー・サーバー名は、`crtmqm` の必須パラメーター `-v QueueServer` で指定します。このパラメーターには、デフォルト値はありません。システム内で固有であり、かつそれが属しているキュー・マネージャーと結び付けやすいプロセス名を指定してください。

キュー・マネージャーのデフォルトのキュー・サーバー・プロセス名は、次のようにして変更できます。

1. キュー・マネージャーを停止させます。
2. デフォルトのボリュームとサブボリュームを、キュー・マネージャー・データ・ファイル (通常は `<QMgrName>D`) の位置に、設定します。
3. 新しいプロセス名を記録するには、キュー・マネージャーの `QMINI` ファイルにある `Configuration` スタンザの `DefaultQueueServerName` エントリーを変更します。
4. `TACL` から次のコマンドを入力し、キュー・マネージャーに対して `PATHMON` を実行します。

```
PATHMON /name $<pmon>, nowait/
```

ここで、<pmon> には、キュー・マネージャーの PATHMON プロセス名が入ります。

5. TACL から次のコマンドを入力し、PATHMON プロセスに対して PATHCOM を実行します。

```
PATHCOM $<pmon>
```

6. PATHCOM から次のコマンドを入力し、キュー・マネージャーの Pathway 構成をロードします。

```
START PATHWAY COOL
```

7. PATHCOM から次のコマンドを入力し、サーバー `mqs-queue n` を調整してプロセスをリセットします。

```
ALTER SERVER MQS-QUEUE $n$ , DELETE PROCESS $<oldname>  
ALTER SERVER MQS-QUEUE $n$ , PROCESS $<newname>
```

8. PATHCOM から次のコマンドを入力して、調整した構成を保管します。

```
SHUTDOWN2  
EXIT
```

これで、デフォルトの状況サーバー・プロセス名の変更が完了しました。次に **strmqm** コマンドでキュー・マネージャーが開始されるときからは、新しいデフォルトの状況サーバー・プロセス名が使用されます。

デフォルト以外のキュー・サーバーの追加と除去

付加的なキュー・サーバーを追加する場合は、すでにあるデフォルトのキュー・サーバーをテンプレートとして用います。新しいキュー・サーバーを追加するためにキュー・マネージャーを停止させる必要はありません。

次のような手順で新しいキュー・サーバーを追加します。

キュー・マネージャーの作成

注: この手順は、キュー・マネージャーが稼働している場合を想定したものです。キュー・マネージャーが稼働していない場合は、この手順を開始する前にまず PATHMON を開始し、PATHWAY 構成をロードしてください。その場合は、

5 のステップを省略し、最後に SHUTDOWN2 コマンドを使用して PATHWAY 構成をディスクに保管します。

1. TACL から次のコマンドを入力し、PATHMON プロセスに対して PATHCOM を実行します。

```
PATHCOM $<pmon>
```

2. PATHCOM から次のコマンドを入力し、デフォルトのキュー・サーバー・クラスをテンプレートとしてベースに置いた属性の作業セットを作成してください。

```
RESET SERVER  
SET LIKE MQS-QUEUEnn
```

3. PATHCOM から次のコマンドを入力し、新しいサーバーの作業セットに変更を加えます。

```
SET SERVER CPUS(n:m)  
RESET SERVER PROCESS $<default queue server process name>  
SET SERVER PROCESS $<new queue server name>
```

4. PATHCOM から次のコマンドを入力し、新しいサーバー・クラス名で新しいサーバーを追加します。

```
ADD SERVER MQS-QUEUEnn
```

5. 次のコマンドを入力して、新しいキュー・サーバーを始動させて使用できるようにした後、PATHCOM を終了します。

```
START SERVER MQS-QUEUEnn
EXIT
```

追加のサーバー・クラスに推奨されている命名規則は、MQS-QUEUE01 や MQS-QUEUE02 などです。ただし、この規則を使用しなければいけないというわけではありません。サーバー・クラス名の先頭に MQS-QUEUE という文字ストリングがあれば、サーバー・クラスは **strmqm** によって開始されます。

キュー・サーバーが追加され、開始された (明示的に PATHCOM を使用した場合も、暗黙的に **strmqm** で開始された場合も含む) なら、**altmqfls** を使用して新しいキュー・サーバーにオブジェクトを再割り当てできます。オブジェクトの再割り当てについては、137ページの『状況サーバーおよびキュー・サーバーにオブジェクトを再割り当てする』を参照してください。

キュー・サーバーを除去するときは、まず、このキュー・サーバーに対して構成されたすべてのオブジェクトが、削除されたか別のキュー・サーバーに再割り当てされていることを確認してください。なお、デフォルトのキュー・サーバーは絶対に削除しないでください。これを削除すると、キュー・マネージャーは作動しなくなります。

ボリューム構造

MQSeries for Compaq NSK のファイルは、いくつかのサブボリュームに分散されます。これらのサブボリュームが常駐するボリュームは、キュー・マネージャーの作成時に選択されますが、これは、MQSINI 内のデフォルトのボリュームの値からとられるか、**crtmqm** コマンドの **-p DefaultPrefix** パラメーターで指定されます。

サブボリュームは、キュー・マネージャーごとに 5 つあります。サブボリュームの内容は、そのサブボリュームの名前の末尾の文字から判別できます。たとえば、ボリューム \$DATA に常駐する QMGR というキュー・マネージャーの場合、サブボリュームは次のようになります。

\$DATA.QMGR	FFST サブボリューム
\$DATA.QMGRD	キュー・マネージャー・データ・ファイル・サブボリューム
\$DATA.QMGRL	キュー・マネージャー・エラー・ログ・サブボリューム
\$DATA.QMGRM	メッセージ・キュー・サブボリューム
\$DATA.QMGRS	チャンネル同期サブボリューム

ボリューム構造

キュー・マネージャーの名前が 7 文字以上ある場合、サブボリューム名は変形または省略されます。これらのサブボリュームの位置や名前の記録には、キュー・マネージャーの MQSINI ファイルのスタンザ QMVolume と QMSubvolume が使用されます。

キュー・マネージャー FFST サブボリューム

FFST サブボリュームには、第 1 障害サポート・ファイルが含まれています。これらのファイルには、接頭部に FD の文字が付いています。これらは、リソースの不足や MQSeries 内部エラー、Compaq NSK システムの問題といった、MQSeries システムで発生する重大な問題を指示します。

キュー・マネージャー・データ・ファイル・サブボリューム

AMQRFN_{xx}	各 CPU のリポジトリ・キャッシュ共用メモリー・ファイル (_{xx} は数)。
CCHDEFS	クライアント接続チャネル定義ファイル。
CCSIDMEM	データ変換 CCSID サポートのための読み取り専用共用メモリー・ファイル。
CHDEFS	チャネル定義ファイル。このファイルには、キュー・マネージャーに定義されるチャネルの構成情報が含まれます。
Lxxxxxxx	名前リスト・ファイル。Lxxxxxxx はオブジェクト名に由来します。そうでないとすれば、それは生成された値です。
OAMDB	許可データベース。
OBJCAT	オブジェクト・カタログ。
ABJCAT	オブジェクト名による索引を含む、OBJCAT の代替キー・ファイル。
PATHCTL	PATHWAY 制御ファイル。
PRIDB	MQSeries プリンシパル (ユーザー) ・データベース。
PRIDBA	PRIDB の代替キー・ファイル。
QMINI	キュー・マネージャー初期設定ファイル。
QMINIMEM	キュー・マネージャー規模の静的構成情報を扱う読み取り専用共用メモリー・ファイル。
REPMGR	複数のリポジトリ・サーバー・プロセスの始動を調整する際に使用されるファイル。

RDFPURGE	論理上で削除された (最終的には cleanrdf によって除去される) ファイルを記録するためにキュー・マネージャーが使用するデータベース。
RUNTIME	ECBOSS および EC リカバリー調整のためのファイル。
SHUTDOWN	endmqm を制御するファイル。
STATABLE	チャンネル状況テーブル・ファイル。このファイルは、チャンネル状況に関連した動的情報を保持します。キュー・マネージャーがシャットダウンされてから再始動されるまでの間チャンネル状況の情報を保管するのに使用されます。
TRACEOPT	キュー・マネージャーの現在のトレース設定を、不定様式のビットマップ・レコードで記録するファイル。このファイルの内容は、制御コマンド strmqtrc および endmqtrc によって変更されます。更新のプロセスはすべて、CONTROL 27 メカニズムによって通知されます。
UMQSINI	キュー・マネージャー始動時の、構造化されていない MQSINI ファイルのスナップショット。

キュー・マネージャー・エラー・ログ・サブボリューム

エラー・ログ・サブボリュームには、エラー・ログ・ファイルやトレース・ログ・ファイルが含まれます。トレース・ファイルは、TR という接頭部によって識別されます。(この接頭部は、QMINI ファイルの TracePrefix エントリで変更できます。) トレース・ファイルには診断情報が含まれています。このファイルは、MQM 機能か **strmqtrc** 制御コマンドによってトレースのスイッチがオンになっている場合にだけ作成されます。

エラー・ログには、MQERRLGn という形式 (n は 1、2、または 3) で名前が付けられます。現行エラー・ログは常に MQERRLG1 です。この MQERRLG1 がいっぱいになると、その内容が MQERRLG2 に移されます。そして、次に MQERRLG1 の内容が移されるときは、MQERRLG2 の内容が MQERRLG3 に移されます。MQERRLG3 は、必要に応じて上書きされます。エラー・ログは、3 つまでしか作成されません。それで、有用なエラー情報が失われることがないように、サイズを適切に設定する必要があります。

キュー・マネージャー・メッセージ・キュー・サブボリューム

メッセージのサブボリュームには、ローカル・キュー上にあるメッセージのストレージに関連したファイルが含まれます。ファイル名は、次のような形式で付けられます。

ボリューム構造

Qxxxxxxx	キュー・ファイルそのもの。持続メッセージが含まれます。
Txxxxxxx	Touch ファイル。オブジェクトに変更が加えられると、Touch ファイルがオブジェクトの日付スタンプを変更します。 Txxxxxxx はオブジェクト名に由来します。そうでないとすれば、それは生成された値です。
Oxxxxxxx	キュー・オーバーフロー・ファイル。
xxxxxxx	固有な短い名前であれば、おそらくキューの名前。そうでないとすれば、生成された値。(83ページの『オブジェクト名の変換』を参照してください。)

上で説明したファイルに加えて、メッセージ・キュー・サブボリュームには、デフォルトで、固有なファイル名の接頭部を持つメッセージ・オーバーフロー・ファイルが作成されます。メッセージ・オーバーフロー・ファイルは、大きなメッセージを保管するために作成されます。このファイルは、固有なファイル名の接頭部が付いていなくても、未監査の構造化されていないファイルなので容易に識別できます。新しいメッセージ・オーバーフロー・ファイルを作成する位置は、

--msgofsubvol パラメーターを指定することによって、キュー単位で **altmqfls** を使用して変更できます。(詳細については、356ページの『dspmqfls (MQSeries ファイル属性の表示)』を参照してください。)

キュー・マネージャー・チャネル同期サブボリューム

キュー・マネージャー・チャネル同期サブボリュームには、内部データベースが含まれていて、キュー・マネージャーが所有するチャネル上で伝送または受信される作業単位 (またはメッセージのバッチ) の状況が記録されます。

キュー・マネージャーでチャネルが使用されると、サブボリュームには次のファイルが作成されます。

SYNCHIDX

同期索引ファイル。キュー・マネージャーによって作成される各同期ファイルのエントリーが含まれています。

Sxxxxxxx

個々の同期ファイル。キュー・マネージャーで使用された、ローカル・チャネルとリモート・チャネルの固有な組み合わせごとに 1 つのファイルが作成されます。これらのファイルには、メッセージのバッチ内で伝送または受信されたメッセージの識別情報が記録されます。この情報は、障害や未確定チャネルの解決の後に行われるチャネルの再同期に使用されます。

オブジェクト名の変換

オブジェクト名は、必ずしも有効なファイル・システム名になっていません。そのため、オブジェクト名を変換しなければならない場合があります。使用される変換方法は、キュー・マネージャー名に使用されるものとは異なります。つまり、1 つのシステムにつきキュー・マネージャー名はわずかしかないかもしれませんが、各キュー・マネージャーごとに相当数の他のオブジェクトが存在する可能性があるためです。ファイル・システム内の個別のファイルを使用して表されるのは、プロセス定義、名前リスト、およびキューだけです。チャンネルや他のオブジェクトは、複数のオブジェクト定義を保持するデータベース内にレコードとして保管されるので、これらの考慮事項に影響されません。

変換プロセスで新しい名前が生成される場合、元のオブジェクト名とは関係ありません。 **dspmqls** コマンドを使用して、本当のオブジェクト名と変換されたオブジェクト名の間で変換を行うことができます。 **dspmqls** は MQSeries オブジェクトと関係があるメインファイルの名前を表示します。

キュー・マネージャーでの作業

MQSeries には、キュー・マネージャーの作成、開始、編集、および削除を行うためのコマンドが用意されています。また、MQSC コマンド **DISPLAY QMGR** を使用してキュー・マネージャーの属性を表示したり、**ALTER QMGR** を使用してそれらを変更することもできます。 122ページの『キュー・マネージャーの属性を表示する』および 124ページの『キュー・マネージャーの属性の更新』を参照してください。

制御コマンドの使用を試行する前に、環境変数 **PMSEARCHLIST** で MQSeries の場所が指定されているか確認してください。環境変数の詳細については、409ページの『付録C. MQSeries for Compaq NSK 用の TA CL 環境の設定』を参照してください。

デフォルト・キュー・マネージャーの作成

crtmqm コマンドを使用して、デフォルト・キュー・マネージャーを作成します。 **-q** フラグを使用して指定した **crtmqm** コマンドは、以下を実行します。

- **saturn.queue.manager** というデフォルト・キュー・マネージャーを作成する。
- デフォルトおよびシステム・オブジェクトを作成する。
- デフォルト伝送キューおよび送達不能キュー両方の名前を指定する。

キュー・マネージャーでの作業

```
crtmqm -q -d MY.DEF.XMITQ -u SYSTEM.DEAD.LETTER.QUEUE -n $PMON -o $TRM01 -s $MQSS  
-v $MQQS saturn.queue.manager
```

説明

-q このキュー・マネージャーがデフォルト・キュー・マネージャーであることを示します。

-d MY.DEF.XMIT.Q
デフォルト伝送キューの名前です。

-u SYSTEM.DEAD.LETTER.QUEUE
送達不能キューの名前です。

-n \$PMON
キュー・マネージャーの PATHMON のプロセス名です。

-o \$TRM01
ホーム端末名です (一時停止しなければなりません)。

-s \$MQSS
デフォルトの状況サーバーのプロセス名です。

-v \$MQQS
デフォルトのキュー・サーバーのプロセス名です。

saturn.queue.manager
このキュー・マネージャーの名前です。 **crtmqm** の場合、この名前はコマンドの最後のパラメーターでなければなりません。

指定した名前およびオプションが設定されて、キュー・マネージャーが作成されます。キュー・マネージャーを作成すると、自動的に以下も作成されます。

- キュー・マネージャーの作成者の MQSeries デフォルト・プリンシパル
- キュー・マネージャーの状況サーバー
- キュー・マネージャーのキュー・サーバー
- キュー・マネージャーの構成ファイル

これで、キュー・マネージャーを開始できます。詳細については、86ページの『キュー・マネージャーの開始』を参照してください。

キュー・マネージャーの属性のいくつかを変更することもできます。これは、MQM または制御コマンド **strmqm** を使用して行えます。詳細については、92ページの図5または 390ページの『strmqm (キュー・マネージャーの開始)』を参照してください。

キュー・マネージャーの作成時に作成された 2 つの構成ファイルのバックアップを作成してください。作成した別のキュー・マネージャーによって問題が発生した場合、それらのバックアップを使用して復元できます。詳細については、61ページの『キュー・マネージャーの作成後の構成ファイルのバックアップ』を参照してください。

MQSeries プリンシパルの作成

crtmqm は、キュー・マネージャーを作成したユーザーのプリンシパルを自動的に作成します。このプリンシパル (デフォルト・プリンシパルとも呼ばれる) は、他の MQSeries インプリメンテーションと互換性を持たせるために、必ず **mqm** と呼ばれます。

いったんキュー・マネージャーを作成したら、MQSeries の他のユーザーのプリンシパルを定義できます。このステップはいつでも (キュー・マネージャーが開始されているかどうかに関係なく) 実行できます。キュー・マネージャーで他のユーザーが必要ない場合、このステップは省略できます。

Compaq NSK ユーザー MQM.MQUSER に対応する、MQPRINCIPAL という名前の MQSeries プリンシパルを作成する場合、以下のコマンドを入力します。

```
altmqusr -m saturn.queue.manager -p MQPRINCIPAL -u MQM.MQUSER
```

現在作成されているすべてのプリンシパルを表示するには、以下のコマンドを入力します。

```
dspmqsusr -m saturn.queue.manager
```

ユーザー用のプリンシパルのエントリーを作成しないと、ユーザーがキュー・マネージャーへのアクセスを試行した場合に (OAM が使用可能かどうかに関係なく)、許可エラーになります。この変更は MQSeries for Compaq NSK バージョン 2.2.0.1 で導入され、以降のすべてのバージョンに含まれています。

RDF 使用可能になっているキュー・マネージャーでの cleanrdf の実行

RDF 環境でキュー・マネージャーを実行しており、MQRDF PARAM を使用して RDF 特有の振る舞いを使用可能にしている場合、以下のように **cleanrdf** ユーティリティーを定期的に使用可能にしてください。

キュー・マネージャーでの作業

- 何らかの構成変更 (オブジェクトの作成または削除や QMINI ファイルへの変更など) を行ったら、**cleanrdf** を実行しなければなりません。
- アプリケーションが通常の操作の一部としてオブジェクトを作成および削除する場合、特に動的キューを使用する場合は、通常の操作の間に **cleanrdf** を実行しなければなりません。この頻度はオブジェクトの削除の率によって決まります。NetBatch または他のスケジューリング・ソフトウェアを使用しなければなりません。

キュー・マネージャーの開始

キュー・マネージャーを作成しても、それが開始されるまでコマンドまたは MQI 呼び出しを処理できません。以下のコマンドを入力して、キュー・マネージャーを開始します。

```
strmqm saturn.queue.manager
```

strmqm コマンドは、キュー・マネージャーが開始して、接続要求を受け入れる用意ができるまで、制御を戻しません。

デフォルトおよびシステム・オブジェクトの復元

デフォルトおよびシステム・オブジェクトは、キュー・マネージャーの作成時に自動的に作成されます。ただし、オブジェクトは他のオブジェクト定義によっていつでも置換できます。デフォルトおよびシステム・オブジェクトを `saturn.queue.manager` というキュー・マネージャーに復元する場合、以下のよう
に **-c** オプションを指定して **strmqm** コマンドを発行します。

```
strmqm -c saturn.queue.manager
```

キュー・マネージャーが開始され、キュー・マネージャーの作成時に作成されたデフォルトおよびシステム・オブジェクトが復元されます。それから、キュー・マネージャーが停止されます。

オブジェクト・ファイルの表示

MQSeries キュー、キュー・マネージャー、またはプロセス・オブジェクトは、それぞれファイルによって表されます。これらのオブジェクトの名前は必ずしも有効なファイル名ではないので、キュー・マネージャーは、必要に応じてそのオブジェクト名を有効なファイル名に変換します。このプロセスについては、83ページの『オブジェクト名の変換』を参照してください。

キュー・マネージャーの停止

キュー・マネージャーを停止するには、**endmqm** コマンドを使用します。たとえば、`saturn.queue.manager` という名前のキュー・マネージャーを停止するには、以下のコマンドを入力します。

```
endmqm saturn.queue.manager
```

デフォルトでは、このコマンドは、指定されたキュー・マネージャーの制御された または静止 シャットダウンを実行します。これは完了するまでに多少の時間がかかります。制御シャットダウンは、接続されたアプリケーションすべてが切断されるまで、および実行されているすべてのチャンネルが停止するまで待機するためです。

『即時および優先キュー・マネージャー・シャットダウン』では、シャットダウンが行われる方法を指定する、**endmqm** コマンドのオプション・フラグについて説明しています。

問題がある場合

キュー・マネージャーのシャットダウンにおける問題は、アプリケーションによって頻繁に引き起こされます。たとえば、次のような場合です。

- アプリケーションが MQI 戻りコードを正しく検査しない場合。
- アプリケーションが静止の通知を要求しない場合。

即時および優先キュー・マネージャー・シャットダウン

キュー・マネージャーのシャットダウンに時間がかかる場合、またはキュー・マネージャーが停止しない場合は、**BREAK** の後に **STOP** を使用して、**endmqm** コマンドを終了できます。その後、別の **endmqm** コマンドを発行できますが、この場合は、即時シャットダウンまたは優先シャットダウンを指定するフラグを付加します。

キュー・マネージャーでの作業

即時シャットダウンの場合、現在の MQI 呼び出しは完了することができますが、新しい呼び出しは失敗します。このタイプのシャットダウンは、アプリケーションがキュー・マネージャーに接続中でも実行されます。即時シャットダウンの場合、コマンドは以下のようになります。

```
endmqm -i saturn.queue.manager
```

即時シャットダウンが機能しない場合は、`-p` フラグを指定して優先 シャットダウンを試行してください。たとえば、次のようにします。

```
endmqm -p saturn.queue.manager
```

重要

この方法は、**endmqm** コマンドを使ってもキュー・マネージャーを停止できなかったときにのみ使用してください。この方法は、接続されているアプリケーションに予測不能な結果を及ぼす可能性があります。

このメソッドがまだ機能しない場合、別の方法について、417ページの『手動によるキュー・マネージャーの停止』を参照してください。

endmqm コマンドおよびそのオプションの詳細については、365ページの『endmqm (キュー・マネージャーの終了)』を参照してください。

キュー・マネージャーの再始動

キュー・マネージャーを再始動するには、次のコマンドを使用します。

```
strmqm saturn.queue.manager
```

既存のキュー・マネージャーのデフォルト・キュー・マネージャーへの指定

デフォルト・キュー・マネージャーを作成するときに、デフォルト・キュー・マネージャーの名前が MQSeries 構成ファイル (MQSINI) 中の *DefaultQueueManager* スタンザのに挿入されます。スタンザとその内容は、スタンザが存在していない場合には自動的に作成されます。

次の場合、このスタンザを編集しなければならないことがあります。

- 既存のキュー・マネージャーをデフォルト・キュー・マネージャーにする場合。このタスクを実行するには、このスタンザのキュー・マネージャー名を、新しいデフォルト・キュー・マネージャーの名前に変更する必要があります。このステップは、テキスト・エディターを使用し、手動で実行しなければなりません。
- ノード上にデフォルト・キュー・マネージャーがなく、既存のキュー・マネージャーをデフォルト・キュー・マネージャーにする場合。このタスクを実行するには、必要な名前を挿入した *DefaultQueueManager* スタンザをユーザー自身で作成する必要があります。
- 別のキュー・マネージャーを誤ってデフォルト・キュー・マネージャーにしてしまったため、元のデフォルト・キュー・マネージャーに戻りたい場合。このタスクを実行するには、MQSeries 構成ファイル内の *DefaultQueueManager* スタンザを編集し、誤ってデフォルト・キュー・マネージャーにしたキュー・マネージャーの名前を、希望するデフォルト・キュー・マネージャーの名前に置き換えます。

デフォルト・キュー・マネージャーが、指定されたものに変更します。アプリケーションがキュー・マネージャーを指定しない場合、以後のすべての接続試行では、新しいデフォルト・キュー・マネージャーに接続します。

構成ファイルについては、243ページの『第13章 構成ファイル』を参照してください。

スタンザに必須情報を入れたら、キュー・マネージャーを停止して再始動します。

キュー・マネージャーの削除

キュー・マネージャーを削除するには、まずキュー・マネージャーを停止してから、その後次のコマンドを使用します。

```
dltmqm saturn.queue.manager
```

重要

キュー・マネージャーを削除する場合、それに関連するすべてのリソース (すべてのキューとそのメッセージやすべてのオブジェクト定義など) も削除されるので、注意してください。また、キュー・マネージャー内のすべてのファイルも除去される場合があります (MQSeries によって作成されていない場合)。

dltmqm コマンドとそのオプションの詳細については、347ページの『dltmqm (キュー・マネージャーの削除)』を参照してください。信頼できる管理者のみにこのコマンドの使用許可を与えるようにします。

キュー・マネージャーを削除する通常の方式がうまくいかない場合、代替の方法について、418ページの『手動によるキュー・マネージャーの除去』を参照してください。

メッセージ・キュー管理 (MQM) 機能の使用

MQSeries for Compaq NSK V5.1 のメッセージ・キュー管理 (MQM) 機能は、Terminal Control Process (TCP) による PATHWAY SCOBOL として実行されます。これは、C 言語 API を呼び出す、MQM サーバー・クラス・サーバーを使用します。

各キュー・マネージャーは独自の PATHWAY 構成で制御されているので、システム上の構成されているキュー・マネージャーごとに、MQM のそれぞれ別個のインスタンスがあります。したがって、MQM はそれが属しているキュー・マネージャーの管理に限定されます。

注: デフォルトでは、最大で 10 人までのユーザーが MQM 機能を同時に使用できます。たとえば、この制限を 20 に変更するには、キュー・マネージャーの PATHWAY から以下のように入力します。

```
ALTER TCP MQS-TCP-01, MAXTERMS 20
```

詳細については、37ページの『第3章 MQSeries コマンド・セットの使用』を参照してください。

注:

MQM を呼び出すには、キュー・マネージャーの PATHCOM プロンプトから run mqmc を入力します。

「MQM Main Menu」は以下のようになります。

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1

      ** Main Menu **

Enter Choice:  _

1. Queue Manager
2. Queues
3. Channels

F1 - Enter                                F16 - Return

(C) Copyright IBM Corp. 1993, 2001 All Rights Reserved.
```

図4. MQSeries for Compaq NSK MQM Main Menu

「MQM Main Menu」から以下のサブメニューを選択できます。

1. キュー・マネージャー
2. キュー
3. チャネル

これらのサブメニューについては、この章の残りの部分で説明します。 Alt+F6 を押せば、いつでも「MQM Main Menu」に戻れます。「Return」キー (F16) を押せば、直前の画面に戻れます。「MQM Main Menu」から選択する場合、F16 を押すと MQM 機能を終了します。

メッセージ・キュー管理 (MQM)

「Queue Manager Menu」の使用

「Queue Manager」オプションを選択するには、「MQM Main Menu」の「Enter Choice」フィールドに 1 を入力してから、「Enter」キー (F1) を押します。以下の「Queue Manager Menu」パネルが表示されます。

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1

      ** Queue Manager Menu **

Name           : MT01
Description    : _____

Command Level   :      510 Trigger Interval : 999999999
Coded Char Set :      819 Platform          : NSK
Max Handles    :      256 Max Uncommitted Msg: 10000
Max Message    : 4194304 Max Priority      :      9
Dead Letter Queue Name : SYSTEM.DEAD.LETTER.QUEUE
Command Input Queue Name : SYSTEM.ADMIN.COMMAND.QUEUE
Default Xmit Queue Name : _____

Authority Event Enabled Y/N? : N      Inhibit Event Enabled Y/N? : N
Local Event Enabled Y/N?    : N      Remote Event Enabled Y/N?   : N
Start/Stop Event Enabled Y/N?: N    Performance Event Enabled Y/N?: N

                        FORCE Y/N? _

F1 - Modify   F2 - Trace   PGDN - Next Page                F16 - Return
```

図 5. 「Queue Manager Menu」パネル

情報の 2 番目のパネルを表示するには、「PGDN」キーを押します。

```

IBM MQSeries for Compaq NonStop Kernel Version 5.1

      ** Queue Manager Menu **

Queue Manager Id      : ROBERT_2001-01-15_10.28.40 _____
Channel Auto Definition: N   Channel Auto Definition Events Enabled Y/N?: N
Auto Definition Exit  : _____
Cluster Workload Data : _____
Cluster Workload Exit : _____
Cluster Workload Length:      100           Distribution List Support: Y
Repository Name       : _____
Repository Name List  : _____

                                     PGUP - Return
    
```

図6. 「Queue Manager Menu」パネル 2

「Queue Manager Menu」パネルを使用して、以下を行うことができます。

- キュー・マネージャーのいくつかの属性の更新
- MQSeries オブジェクトのトレースの制御

キュー・マネージャーの属性の更新

「Queue Manager Menu」パネルで目的の値を上書き入力し、「Modify」キー (F1) を押します。変更できない値は、上書き入力できないようになっています。

MQSeries オブジェクトのトレース

「Trace」キー (F2) を押すと、以下のような QUEUE MANAGER TRACE MENU が表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1

** QUEUE MANAGER TRACE MENU **

_ API          : MQI.
_ COMMS       : Communications networks processing flow.
_ CSFLOWS    : Common services processing flow.
_ LQMFLOWS   : Local queue manager processing flow.
_ REMOTEFLOWS : Communications component processing flow.
_ ADMINFLOW  : Administrative processing flow.
_ OTHERFLOWS : Other components processing flow.
_ CSDATA     : Common services data buffers.
_ LQMDATA    : Local queue manager internal data buffers.
_ REMOTEDATA : Communications component internal data buffers.
_ ADMINDATA  : Administrative internal data buffers.
_ OTHERDATA  : Other components internal data buffers.
_ VERSIONDATA : Output version of MQSeries running.
_ COMMENTARY : Output program comments in the MQSeries components.

_ All         : Select all options.

F1-Start Trace  F2-Stop Trace                               F16-Return
```

図 7. QUEUE MANAGER TRACE MENU

以下のトレース・オプションが使用可能です。

- | | |
|--------------------|---|
| API | MQI および主なキュー・マネージャー・コンポーネントに関連するトレース・ポイントについてデータを出力します。 |
| COMMS | 通信ネットワークを介して流れるデータに関連するトレース・ポイントについてデータを出力します。 |
| CSFLOWS | 共通サービス内の処理フローに関連するトレース・ポイントについてデータを出力します。 |
| LQMFLOWS | ローカル・キュー・マネージャー内の処理フローに関連するトレース・ポイントについてデータを出力します。 |
| REMOETFLOWS | 通信コンポーネント内の処理フローに関連するトレース・ポイントについてデータを出力します。 |
| ADMINFLOW | 管理内部データ・バッファに関連するトレース・ポイントについてデータを出力します。 |
| OTHERFLOWS | 他のコンポーネントの処理フローに関連するトレース・ポイントについてデータを出力します。 |

- CSDATA** 共通サービス内の内部データ・バッファーに関連するトレース・ポイントについてデータを出力します。
- LQMDATA** ローカル・キュー・マネージャー内の内部データ・バッファーに関連するトレース・ポイントについてデータを出力します。
- REMOTEDATA** 通信コンポーネント内の内部データ・バッファーに関連するトレース・ポイントについてデータを出力します。
- ADMINDATA** 通信コンポーネント内の内部データ・バッファーに関連するトレース・ポイントについてデータを出力します。
- OTHERDATA** 他のコンポーネントの内部データ・バッファーに関連するトレース・ポイントについてデータを出力します。
- VERSIONDATA** 実行中の MQSeries のバージョンに関連するトレース・ポイントについてデータを出力します。
- COMMENTARY** MQSeries コンポーネント内の注釈に関連するトレース・ポイントについてデータを出力します。
- ALL** トレース・ポイントが使用可能になり、完全なトレースが生成されます。

トレースを開始 (または停止) するコンポーネントの名前に対する何らかの文字を入力します。

選択したコンポーネントのトレースを開始するには、「Start Trace」キー (F1) を押します。選択したコンポーネントのトレースを停止するには、「Stop Trace」キー (F2) を押します。

「Queues」メニューの使用

「Queues」オプションを選択するには、「MQM Main Menu」の「Enter Choice」フィールドに 2 を入力してから、「Enter」キー (F1) を押します。以下の「Search Criteria」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1

** Search Criteria **

Queue Name: _____
Enter a queue name or part of one:

Queue Type: _
choose one or leave blank:      1. Local
                                2. Model
                                3. Remote
                                4. Alias

F1 - Enter                                F16 - Return
```

図 8. 「Search Criteria」パネル (キュー)

「Search Criteria」パネルの「Queue Name」フィールドに、キュー名の一部または全体を入力します。検索を 1 つのタイプのキューに限定する場合、「Queue Type」ID も指定します。「Enter」キー (F1) を押します。「Queue Menu」が表示されます。これは、MQSeries キューの表示、変更、作成、コピー、削除、およびモニターに使用します。


```

IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Queue Menu **

Queue Name                                     Type
- ANNE.ET01.RQSD.LOCAL                        QLOCAL
- ANNE.ET01.RQSD.REMOTE                       QREMOTE
- ANNE.ET01.RQSV.LOCAL                        QLOCAL
- ANNE.ET01.RQSV.REMOTE                       QREMOTE
- ANNE.ET01.SDRCL.LOCAL                       QLOCAL
- ANNE.ET01.SDRCL.REMOTE                      QREMOTE
- ANNE.M401.RQSD.LOCAL                        QLOCAL
- ANNE.M401.RQSD.REMOTE                       QREMOTE
- ANNE.M401.RQSV.LOCAL                        QLOCAL
- ANNE.M401.RQSV.REMOTE                       QREMOTE
- ANNE.M401.SDRCL.LOCAL                       QLOCAL
- ANNE.M401.SDRCL.REMOTE                      QREMOTE

F1 - Enter/Display/Modify      F2 - Create      F3 - Copy      F4 - Delete
F5 - Monitor                   PGDN            PGUP          F16 - Return
    
```

図9. 「Queue Menu」

注: 使用している MQM 要求側に関連したキュー・マネージャーだけでしか、キューの作成、変更、および削除は行えません。

キューのリストをスクロールするには、PGUP および PGDN キーを使用します。

キューの作成

「Queue Menu」から「Create」キー (F2) を押すと、以下のような「Create Queue」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Queue **

Queue Type: _  1=Local, 2=Model, 3=Remote, 4=Alias,

Name: _____

Replace [Y/N]: _

F1 - Enter                                F16 - Return
```

図 10. 「Create Queue」パネル

新しいキュー定義を作成するには、以下を行います。

1. 「Queue Type」フィールドで、1 (ローカル・キューの場合)、2 (モデル・キューの場合)、3 (リモート・キューの場合)、または 4 (別名キューの場合) を入力します。
2. 「Name」フィールドにキュー・マネージャー名を入力します。
3. キューが、名前とタイプの同じ既存のキューを置換する場合、「Replace」フィールドに Y を入力します。
4. 「Enter」キー (F1) を押します。

ローカル・キューを作成する場合、以下のような「Create Local Queue」パネルが表示されます。

```

IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Local Queue **

Queue Name : TEST
Description:
-----
Default Msg Priority : 0          Put Enabled [Y/N]      : _
Default Persistence : _          Get Enabled [Y/N]      : _

Retention Interval  : 0          Queue Definition Type  : _____
Max Queue Depth     : 0          Priority/FIFO [P/F]    : _
Max Message Length  : 0          Share [Y/N]           : _
Backout Threshold   : 0          Usage [N/X]           : _
Backout Requeue Name : _____
Init. Queue         : _____
Process Name        : _____
Trigger Type [N/E/F/D]: _        Trigger/NoTrigger [Y/N] : _
Trigger Depth       : 0          Trigger Priority       : 0
Trig. Data :
-----
Q Depth Max Event   : _          Q Serv. Int. Event[H/O/N]: _
Q Depth High Limit  : 0          Q Depth High Event      : _
Q Depth Low Limit   : 0          Q Depth Low Event       : _
Q Service Interval  : 0          Scope                   : _____
F1 - Enter          PGDN - Next Page          F16 - Return
    
```

図 11. 「Create Local Queue」パネル

情報の 2 番目のパネルを表示するには、「PGDN」キーを押します。

```

IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Local Queue **

Cluster Name       : _____
Cluster Name List  : _____

Distribution List Y/N : _          Default Binding [O/N]   : _

PGUP - Return
    
```

図 12. 「Create Local Queue」パネル 2

パネルを完成させ、「Enter」キー (F1) を押します。

リモート・キューを作成する場合、以下のような「Create Remote Queue」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Remote Queue **

Queue Name : TEST_REMOTE
Description: _____
Default Msg Priority : 0          Put Enabled (Y/N) : _
Default Persistence : _          Default Binding [0/N] : _

Scope      :
Remote Queue Name : _____
Remote Queue Manager : _____
Transmit Queue Name : _____
Cluster Name : _____
Cluster Name List : _____

F1 - Enter                                F16 - Return
```

図 13. 「Create Remote Queue」パネル

パネルを完成させ、「Enter」キー (F1) を押します。

キューのコピー

既存の定義をコピーすることによって新しいキューを定義するには、「Queue Menu」から「Copy」キー (F3) を押します。以下のような「Copy Queue」パネルが表示されます。

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Copy Queue **

Name: ANNE.ET01.RQSD.LOCAL.2_____

Replace [Y/N]: _

Like Queue: ANNE.ET01.RQSD.LOCAL_____

Queue Type: QLOCAL__

F1 - Enter                                F16 - Return
```

図 14. 「Copy Queue」パネル

「Name」フィールドに新しいキュー定義の名前を入力します。新しいキューが、名前とタイプと同じ既存のキューを置換する場合、「Replace」フィールドに Y を入力します。コピーする定義の名前を「Like Queue」フィールドに入力します。キュー・タイプを「Queue Type」フィールドに入力します。

「Enter」キー (F1) を押します。

キューの変更

「Queue Menu」で「Modify」キー (F1) を押すと、以下のような「Display/Modify Local Queue」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Display/Modify Local Queue **
Queue Name : ANNE.ET01.RQSD.LOCAL
Description: Local queue ET01 receiver
Default Msg Priority : 0          Put Enabled [Y/N]      : Y
Default Persistence : N          Get Enabled [Y/N]     : Y

Retention Interval : 999999999   Queue Definition Type : PREDEFINED
Max Queue Depth    : 5000        Priority/FIFO [P/F]   : P
Max Message Length : 1024        Share [Y/N]          : Y
Backout Threshold  : 0           Usage [N/X]           : N
Backout Requeue Name : _____
Init. Queue       : _____
Process Name      : _____
Trigger Type [N/E/F/D]: F        Trigger/NoTrigger [Y/N] : N
Trigger Depth    : 1            Trigger Priority       : 0
Trig. Data : _____
Q Depth Max Event : Y           Q Serv. Int. Event[H/O/N]: N
Q Depth High Limit : 80         Q Depth High Event      : N
Q Depth Low Limit  : 20         Q Depth Low Event       : N
Q Service Interval : 999999999  Scope                   : QMGR
F1 - Modify      PGDN - Next Page                               F16 - Return
```

図 15. 「Display/Modify Local Queue」パネル

変更する値を上書き入力して、「Modify」キー (F1) を押します。変更できない値は、上書き入力できないようになっています。

キューの削除

「Queue Menu」で、削除するキューの名前に対する何らかの文字を入力します。「Delete」キー (F4) を押してから、もう一度 F4 を押して削除を確認します。

キューのモニター

「Queue Menu」で「Monitor」キー (F5) を押すと、以下のような「Monitor Local Queues」パネルが表示されます。

```

IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Monitor Local Queues **
Queue          OPEN INPUT  OPEN OUTPUT  DEPTH
-----
ANNE_M401_RQSD_LOCAL
ANNE_M401_RQSV_LOCAL
ANNE_M401_SDRS_LOCAL
ANNE_MA02_RQSD_LOCAL
ANNE_MA02_RQSV_LOCAL
ANNE_MA02_SDRS_LOCAL
ANNE_MD01_RQSD_LOCAL
ANNE_MD01_RQSV_LOCAL
ANNE_MD01_SDRS_LOCAL
ANNE_MD01_SVRS_LOCAL
ANNE_ME02_RQSD_LOCAL
ANNE_ME02_RQSV_LOCAL
ANNE_ME02_SDRS_LOCAL          10
ANNE_ME02_SVRS_LOCAL

F12 - Refresh          PGDN          PGUP          F16 - Return
    
```

図 16. 「Monitor Local Queues」パネル

この例では、キューは入力も出力も行われません。

ANNE_ME02_SDRS_LOCAL という 1 つのキューに 10 個のメッセージが含まれています。

MQS-MQMSVR00 の MQMQMREFRESHINT PATHWAY パラメーターで、チャネルおよびキューのモニター画面が最新表示される頻度を決定します。デフォルトの頻度は 30 秒ごとです。たとえば、頻度を 10 秒ごとに変更する場合、以下のように、キュー・マネージャーの PATHWAY から入力します。

```

ALTER SERVER MQS-MQMSVR00, PARAM MQMQMREFRESHINT 10
    
```

「Channels」メニューの使用

「Channels」オプションを選択するには、「MQM Main Menu」の「Enter Choice」フィールドに 3 を入力してから、「Enter」キー (F1) を押します。以下のようなチャネルの「Search Criteria」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1

** Search Criteria **

Channel Name: _____
Enter a channel name or part of one:

Channel Type: _
choose one or leave blank:
1. Sender
2. Server
3. Receiver
4. Requester
5. SvrConn
6. Cluster Sender
7. Cluster Receiver

F1 - Enter                                F16 - Return
```

図 17. 「Search Criteria」パネル (チャンネル)

「Channel Name」フィールドにチャンネル名の一部または全体を入力します。
「Channel Type」フィールドには、目的のチャンネルのタイプを識別するために、1 から 5 までの番号を入力できます。「Enter」キー (F1) を押すと、以下のような「Channel Menu」が表示されます。


```

IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Channel Menu **

Channel Name          TYPE          STATUS
-
MA02.MT01.SDRC.0001  RECEIVER
MA02_MT01_RQSD_0001  REQUESTER
MA02_MT01_RQSV_0001  REQUESTER
MA02_MT01_SDRC_0001  RECEIVER
MD01_MT01_RQSD_0001  REQUESTER
MD01_MT01_RQSV_0001  REQUESTER
MD01_MT01_SDRC_0001  RECEIVER
MD01_MT01_SVRC_0001  RECEIVER
ME02_MT01_RQSD_0001  REQUESTER
ME02_MT01_RQSV_0001  REQUESTER
ME02_MT01_SDRC_0001  RECEIVER
ME02_MT01_SVRC_0001  RECEIVER

F1 - Enter/Display/Modify   F2 - Create   F3 - Copy   F4 - Delete
F5 - Monitor               F6 - Resolve  F7 - Reset MSN F8 - Start/Stop F10 - Status
F12 - Refresh              PGDN          PGUP        F16 - Return

```

図 18. 「Channel Menu」

「Channel Menu」には、使用した検索基準に一致するチャンネルのリストが表示されます。「Channel Menu」では以下を行えます。

- チャンネル状況の表示および変更。
- 新しいチャンネル定義の作成。
- チャンネル定義のコピー。
- チャンネル定義の削除。
- チャンネル状況のモニター。
- チャンネルの解決。
- メッセージ・シーケンス番号 (MSN) のリセット。
- チャンネルの開始または停止。

チャンネルの変更

「Channel Menu」で、変更するチャンネルに対する何らかの文字を入力し、「Enter/Display/Modify」キー (F1) を押します。適当なパネルが表示されます。たとえば、送信側チャンネルを選択すると、「Display/Modify Sender Channel」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Display/Modify Sender Channel **

Channel Name      : MT01.M401.SDRC.0001_
Description       : Sender to M401_____

Xmit Queue Name   : M401.TQ.SDRC.0001_____
Data Conversion Y/N: N   NonPersistent Msg Speed [FAST/NORMAL]: FAST__
User Id           : _____ Password : _____
MCA Name          : _____ MCA UserID : _____
Batch Size        : _____ 50 Max Message Size : _____ 4194304
MSN Wrap Count    : _____ 9999999 Disconnect Interval: _____ 60
Short Retry Count : _____ 10 Short Timer : _____ 60
Long Retry Count  : _____ 9999999 Long Timer : _____ 1200
Heartbeat Interval : _____ 300 Batch Interval : _____ 0
Transport Protocol : 1 (1=Lu6.2/ 2=TCP/IP) TCP/IP Port Number : _____
TCP/IP Address    : _____
TCPIP/SNA Process : $ZTC1_____
Local LU Name     : IYAHT080 Remote LU Name : IYAFT110
Local TP Name     : INTCRS6A_____ Mode Name : LU62PS__
Remote TP Name    : _____

F1 - Modify PGDN - Exits F16 - Return
```

図 19. 「Display/Modify Sender Channel」 パネル (1)

情報の 2 番目のパネルを表示するには、「PGDN」キーを押します。

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Display/Modify Sender Channel **

Scrty Data: _____
Scrty Exit: _____

NOTE: RUNMQSC must be used to update the Send, Receive and Message
Data and Exit attributes of channels

PGUP - Return
```

図 20. 「Display/Modify Sender Channel」 パネル (2)

変更する値を上書き入力して、「Modify」キー (F1) を押します。変更できない値は、上書き入力できないようになっています。

チャンネル定義の作成

「Channel Menu」で「Create」キー (F2) を押すと、以下のような「Create Channel」パネルが表示されます。

```

IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Channel **

Channel Type: 1 1=Sender, 2 = Server, 3=Receiver,
                4=Requester, 5 = Server Connection
                6 = Cluster Sender, 7 = Cluster Receiver
Name: _____
Replace [Y/N]: _

F1 - Enter                                F16 - Return

```

図 21. 「Create Channel」パネル

新しいチャンネル定義を作成するには、以下を行います。

1. 「Channel Type」フィールドで、1 (送信側チャンネルの場合)、2 (サーバー・チャンネルの場合)、3 (受信側チャンネルの場合)、4 (要求側チャンネルの場合)、または 5 (サーバー接続の場合) を入力します。
2. 「Name」フィールドにチャンネル定義の名前を入力します。
3. 「Enter」キー (F1) を押します。
4. 定義が、名前とタイプと同じ既存の定義を置換する場合、「Replace」フィールドに Y を入力します。

「Channel Type」フィールドに 1 を入力すると、以下のような「Create Sender Channel」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Sender Channel **
Channel Name      : Compaq_TO_SOLARIS___
Description      : _____
                  _____

Xmit Queue Name  : _____
Data Conversion Y/N: NonPersistent Msg Speed [FAST/NORMAL]: FAST__
User Id         : _____ Password      : _____
MCA Name       : _____ MCA UserID    : _____
Batch Size     : _____ 50 Max Message Size : 4194304
MSN Wrap Count : 999999999 Disconnect Interval: 6000
Short Retry Count : 999999 Short Timer      : 60
Long Retry Count : 999999 Long Timer       : 1200
Heartbeat Interval : 300 Batch Interval    : 0
Transport Protocol : _ (1=Lu6.2/ 2=TCP/IP) TCP/IP Port Number : _____
TCP/IP Address   : _____
TCPIP/SNA Process : $ZTC1_____
Local LU Name    : _____ Remote LU Name : _____
Local TP Name    : _____ Mode Name     : _____
Remote TP Name   : _____

F1 - Enter PGDN - Exits F16 - Return
```

図 22. 「Create Sender Channel」 パネル

「Channel Type」フィールドに 3 を入力すると、以下のような「Create Receiver Channel」パネルが表示されます。

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Receiver Channel **
Channel Name      : SOLARIS_TO_Cmpaq___
Description      : _____
                  _____

Put Authority D/C : _ NonPersistent Msg Speed [FAST/NORMAL]: FAST__
User Id         : _____ MCA UserID    : _____
Batch Size     : _____ 50 Max Message Size : 4194304
Msg Retry Count : _____ 10 Msg Retry Interval : 1000
Heartbeat Interval : 300 MSN Wrap Count : 999999999
Transport Protocol : _ (1=Lu6.2/ 2=TCP/IP)
F1 - Enter PGDN - Exits F16 - Return
```

図 23. 「Create Receiver Channel」 パネル

「Channel Type」フィールドに 5 を入力すると、以下のような「Create Server Connection Channel」パネルが表示されます。

```

IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Server Connection Channel **

Channel Name      : WINDOWS_CLIENT_____
Description       : _____
                  _____

MCA UserID        : _____
Max Message Size  :      4194304
Heartbeat Interval :      300
Transport Protocol : _ (1=Lu6.2/ 2=TCP/IP)

F1 - Enter      PGDN - Exits                      F16 - Return
    
```

図 24. 「Create Server Connection Channel」 パネル

新しいチャネル定義を作成するには、要求されたパネルを完成させ、「Enter」キー (F1) を押します。

「Channel Type」フィールドに 6 を入力すると、以下のような「Create Cluster Sender Channel」パネルが表示されます。

```

IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Cluster Sender Channel **

Channel Name      : CLUSTER_SENDER_____
Description       : _____
                  _____

Cluster Name      : _____
Cluster Name List : _____
Data Conversion Y/N : _ NonPersistent Msg Speed [FAST/NORMAL]: FAST__
User Id           : _____ Password           : _____
MCA Name          : _____ MCA UserID         : _____
Batch Size        :      50 Max Message Size     :      4194304
MSN Wrap Count    :      999999999 Disconnect Interval :      6000
Short Retry Count :      999999 Short Timer        :      60
Long Retry Count  :      999999 Long Timer         :      1200
Heartbeat Interval :      300 Batch Interval      :      0
Transport Protocol : _ (1=Lu6.2/ 2=TCP/IP) TCP/IP Port Number : _____
TCP/IP Address    : _____
TCP/IP/SNA Process : _____
Local LU Name     : _____ Remote LU Name     : _____
Local TP Name     : _____ Mode Name         : _____
Remote TP Name    : _____

F1 - Enter      PGDN - Exits                      F16 - Return
    
```

図 25. 「Create Cluster Sender Channel」 パネル

「Channel Type」フィールドに 7 を入力すると、以下のような「Create Cluster Receiver Channel」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Create Cluster Receiver Channel **
Channel Name      : CLUSTER_RECEIVER____
Description       : _____ Put Authority D/C: _
                  : _____ Network Priority : _
Cluster Name     : _____
Cluster Name List : _____
Data Conversion Y/N: _ NonPersistent Msg Speed [FAST/NORMAL]: FAST_
User Id          : _____ Password           : _____
MCA Name         : _____ MCA UserID        : _____
Batch Size       : _____ 50 Max Message Size : _____ 4194304
MSN Wrap Count   : _____ 999999999 Disconnect Interval: _____ 6000
Short Retry Count : _____ 999999 Short Timer      : _____ 60
Long Retry Count  : _____ 999999 Long Timer       : _____ 1200
Heartbeat Interval : _____ 300 Batch Interval    : _____ 0
Msg Retry Count   : _____ 0 Msg Retry Interval : _____ 0
Transport Protocol : _ (1=Lu6.2/ 2=TCP/IP) TCP/IP Port Number : _____
TCP/IP Address    : _____
TCP/IP/SNA Process : _____ Local LU Name      : _____
Local TP Name     : _____ Remote LU Name     : _____
Remote TP Name    : _____ Mode Name          : _____
F1 - Enter       PGDN - Exits                      F16 - Return
```

図 26. 「Create Cluster Receiver Channel」パネル

チャンネルのモニター

「Channel Menu」パネルで「Monitor」キー (F5) を押すと、以下のような「Monitor Channels」パネルが表示されます。

```

IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Monitor Channels **
Channel Name      Status   Curr MSN  Last MSN  MCA Status  Stop
=====
MT01.MH01.SDRC.0002  BINDING
MT01.VM03.SDRC.0002  RUNNING      6266      6266  RUNNING    NO
=====

F12 - Refresh          PGDN          PGUP          F16 - Return

Refreshing.....
    
```

図 27. 「Monitor Channels」 パネル

MQS-MQMSVR00 の MQMQMREFRESHINT PATHWAY パラメーターで、チャンネルおよびキューのモニター画面が最新表示される頻度を決定します。デフォルトの頻度は 30 秒ごとです。たとえば、頻度を 10 秒ごとに変更する場合、以下のように、キュー・マネージャーの PATHWAY から入力します。

```
ALTER SERVER MQS-MQMSVR00, PARAM MQMQMREFRESHINT 10
```

チャンネルの削除

「Channel Menu」でチャンネル名に対する何らかの文字を入力することにより、削除するチャンネルを選択します。チャンネルを削除するには、「Delete」キー (F4) を押してから、もう一度 F4 を押して削除要求を確認します。

チャンネル状況の表示

「Channel Menu」パネルで「Status」キー (F10) を押すと、以下のような「Channel Status」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
Channel Status

Channel Name      : MT01.VM03.SDR.C.0002_
Xmit Queue Name  : VM03NCM.TQ.SDR.C.0001_____
Connection Name  : $BP01.IYAHT080.IYCNVM03_____

Channel Status   : RUNNING_____      In Doubt           : NO_
Start Date      : 2001-02-03           Start Time         : 15.07.14
Last Msg Date   : 2001-02-03           Last Msg Time      : 16.34.04

MCA Job Name     : 000069AA_____
Current LUW ID   : 03544240E28B0277
Last LUW ID     : 03544240E28B0277      Current Messages   :          0
MCA Status      : RUNNING_____      Current Seq Num    :        6266
Stop Requested  : NO______          Last Seq Num       :        6266
Number of Batches :          6          Number of Messages :          6
Number of Buffers Sent:        14      Number of Buffers Recvd:          7
Number of Bytes Sent :        3204     Number of Bytes Recvd :         196
Num of Long Retry Left: 9999999       Num of Short Retry Left:          10

F12 - Refresh                                     F16 - Return
```

図 28. 「Channel Status」 パネル

チャネルの開始および停止

「Channel Menu」 パネルで「Start/Stop」キー (F8) を押すと、以下のような「Start/Stop Channel」パネルが表示されます。


```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
Start/Stop Channel

Name: MT01_MA02_SDRC_0001_

Status:

Action: _  choose one of the following:
          1. Start Channel
          2. Stop Immediate
          3. Stop Quiesce

F1 - Enter                                F16 - Return
```

図 29. 「Start/Stop Channel」パネル

「Name」フィールドにチャンネルの名前を入力して、「Action」フィールドに 1 から 3 までの番号を入力します。「Enter」キー (F1) を押します。

メッセージ・シーケンス番号 (MSN) のリセット

「Channel Menu」で「Reset MSN」キー (F7) を押すと、以下のような「Reset Channel」パネルが表示されます。

メッセージ・キュー管理 (MQM)

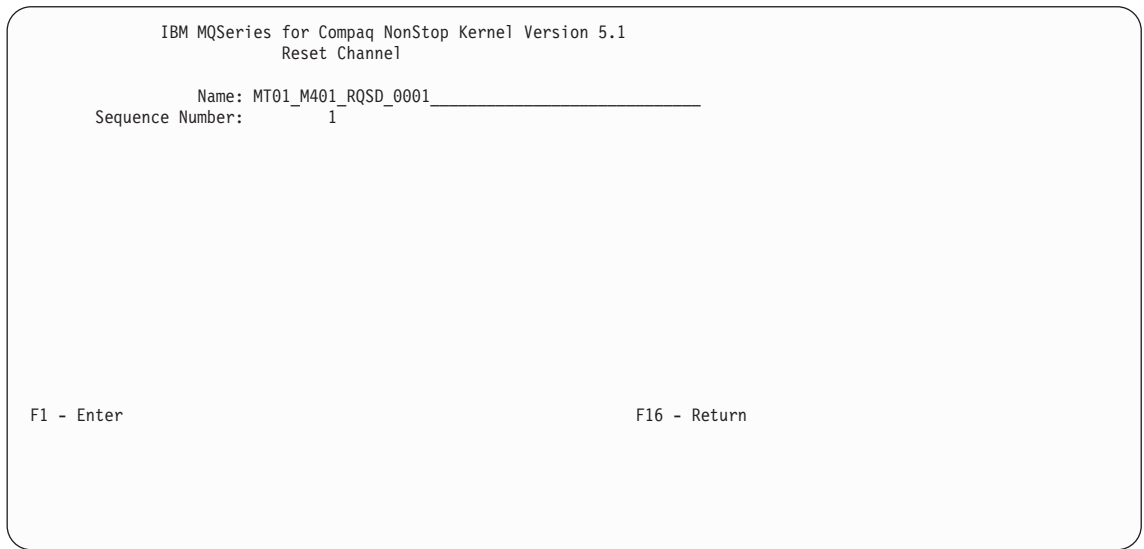


図 30. 「Reset Channel」パネル

MSN を使用することにより、メッセージの重複を確実に避けることができ、メッセージが送信された順序と同じ順序で必ず保管されるようになります。この画面では、必要に応じてチャンネルのシーケンス番号をリセットできます。

チャンネルの解決

「Channel Menu」で「Resolve」キー (F6) を押すと、「Resolve Channel」パネルが表示されます。

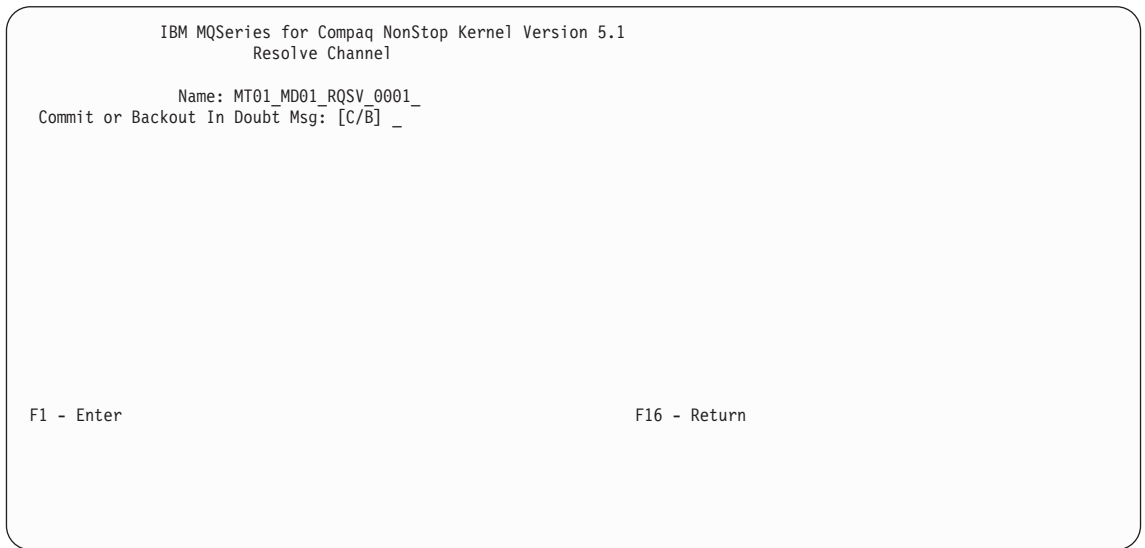


図 31. 「Resolve Channel」パネル

以下を行えます。

- 未確定メッセージ・バッチのバックアウト (B)
- 未確定メッセージ・バッチのコミット (C)

チャネルのコピー

「Channel Menu」で「Copy」キー (F3) を押します。以下のような「Copy Channel」パネルが表示されます。

メッセージ・キュー管理 (MQM)

```
IBM MQSeries for Compaq NonStop Kernel Version 5.1
** Copy Channel **

Name: _____
Replace [Y/N]: _
Like Name: MT01_M401_RQSV_0001_
Channel Type: SERVER

F1 - Enter                                F16 - Return
```

図 32. 「Copy Channel」パネル

「Name」フィールドに新しいチャネルの名前を入力します。「Like Name」フィールドには、コピーするチャネル定義の名前を入力します。「Channel Type」フィールドには、チャネルのタイプを入力します。チャネル定義をコピーするには、「Enter」キー (F1) を押します。

第5章 ローカル MQSeries オブジェクトの管理

この章では、メッセージ・キュー・インターフェース (MQI) を使用したアプリケーション・プログラムをサポートするための、ローカル MQSeries オブジェクトの管理方法を説明します。MQI によって、アプリケーション・プログラムはメッセージ・キュー・サービスにアクセスすることができます。

ローカル管理は、MQSeries オブジェクトを作成、表示、変更、コピー、および削除する場合に行われます。

この章には、次の節があります。

- 『MQI を使用したアプリケーション・プログラムのサポート』
- 118ページの『MQSC コマンドを使用してローカル管理タスクを実行する』
- 124ページの『テキスト・ファイルから MQSCコマンドを実行する』
- 129ページの『MQSC で起こった問題の解決』
- 131ページの『ローカル・キューの取り扱い』
- 142ページの『別名キューの取り扱い』
- 145ページの『モデル・キューの取り扱い』
- 146ページの『トリガー操作のためのオブジェクトの管理』

MQI を使用したアプリケーション・プログラムのサポート

MQI のアプリケーション・プログラムを正常に実行するためには、一定のオブジェクトが必要です。MQI アプリケーションは、キューからメッセージを除去し、それらを処理し、その結果を同じキュー・マネージャー上の別のキューに送信することができます。

アプリケーションのサポート

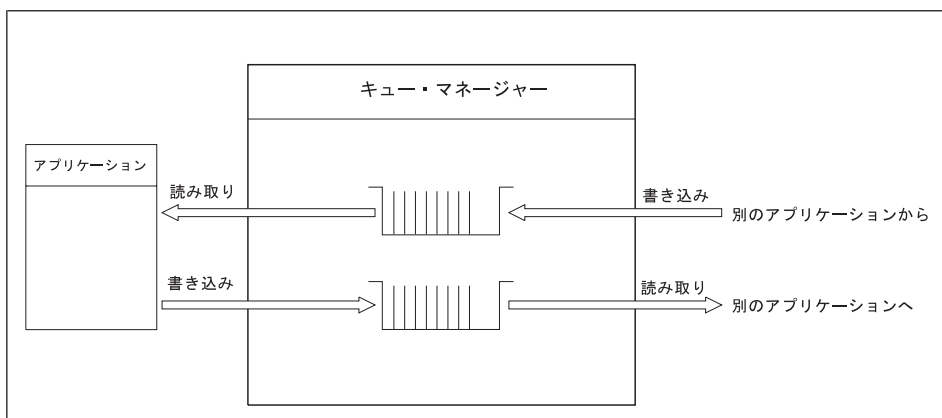


図 33. キュー、メッセージ、およびアプリケーション

アプリケーションは、ローカル・キューまたはリモート・キューにメッセージを (MQPUT を使用して) 書き込むことができますが、ローカル・キューからしかメッセージを (MQGET を使用して) 直接読み取れません。

このアプリケーションが実行されるには、次の条件が満たされている必要があります。

- キュー・マネージャーが存在しており、実行されている。
- 最初のアプリケーション・キュー (ここからメッセージが取り出される) が定義されている。
- 2 番目のキュー (アプリケーションはここにメッセージを書き込む) も定義されている (動的キューでない限り)。
- アプリケーションが、キュー・マネージャーに接続可能である。このタスクを実行するためには、アプリケーションがプロダクト・コードにリンクされている必要があります。詳細については、437ページの『付録I. アプリケーションの構築および実行』を参照してください。
- 最初のキューにメッセージを書き込むアプリケーションも、キュー・マネージャーに接続する。また、それらのアプリケーションがリモートである場合も、伝送キューとチャンネルは共にセットアップされている。

MQSC コマンドを使用してローカル管理タスクを実行する

MQSeries コマンド (MQSC) により、MQSeries オブジェクトを操作することができます。コマンド・プロンプトで **runmqsc** コマンドを使用してコマンドを発行することができます。

MQSeries for Compaq NSK 環境での MQSC の使用の詳細については、421ページの『付録G. MQSeries for Compaq NSK によってサポートされる MQSC』を参照してください。

MQSeries スクリプト・コマンド (MQSC) を使用して、キュー・マネージャー自体、クラスター、チャンネル、キュー、およびプロセス定義などのキュー・マネージャー・オブジェクトを管理することができます。この節では、キュー・マネージャー、キューおよびプロセス定義を扱います。チャンネル・オブジェクトの管理に関する詳細については、*MQSeries 相互通信* の『DQM のインプリメンテーション』を参照してください。

runmqsc コマンドを使用して、キュー・マネージャーに対して、MQSC コマンドを発行します。このことは、対話式で行うことができますし (キーボードからコマンドを発行)、標準入力をリダイレクトして ASCII テキスト・ファイルから一連のコマンドを実行することもできます。どちらの場合も、コマンドの形式は同じです。

コマンドに設定したフラグによって、**runmqsc** コマンドを次の 3 とおりのモードで実行できます。

- **検証モード**。このモードでは、MQSC コマンドはローカル・キュー・マネージャー上で検査され、実際には実行されません。
- **直接モード**。このモードでは、MQSC コマンドはローカル・キュー・マネージャー上で実行されます。
- **間接モード**。このモードでは、MQSC コマンドはリモート・キュー・マネージャー上で実行されます。

MQSC に指定するオブジェクト属性は、本書では大文字 (たとえば、RQMNAME) で示されます。ただし、大文字小文字の区別はありません。MQSC 属性名は 8 文字までに制限されています。

開始する前に

開始する前に、キュー・マネージャーを作成して始動する必要があります。そして、MQSC コマンドを実行します。詳細については、83ページの『デフォルト・キュー・マネージャーの作成』を参照してください。

MQSeries オブジェクト名

例では、いくつかの長い名前をオブジェクトに使用しています。これは、取り扱うオブジェクトのタイプを識別するのに役立ちます。

MQSC コマンドの発行

MQSC コマンドを出す場合、必要なことはキューのローカル名を指定することのみです。例では、次のようなキュー名を使用しています。

```
ORANGE.LOCAL.QUEUE
```

この名前の LOCAL.QUEUE という部分は、このキューがローカル・キューであることを示すためにのみ使用しています。一般に、ローカル・キューの名前に、この部分は必要ではありません。

キュー・マネージャー名として saturn.queue.manager という名前も使用しています。

この名前の queue.manager の部分は、このオブジェクトがキュー・マネージャーであることを示すために使用しているだけです。一般に、キュー・マネージャーの名前に、この部分は必要ではありません。

このような名前を使用する必要はありませんが、使用しない場合には、これらの名前を指定している例の中のコマンドを修正する必要があります。

MQSC コマンドでの大文字小文字の区別

MQSC コマンドおよびその属性は、大文字にも小文字にもなることができ、大文字小文字の区別はありません。それとは逆に、オブジェクト名は大文字小文字の区別があります (つまり、input-queue と INPUT-QUEUE は異なるオブジェクトです)。MQSC コマンドの中のオブジェクト名は、単一引用符で名前を囲まない限り、自動的に大文字に変換されます (つまり、QUEUE と queue は区別されません)。引用符で囲まないと、オブジェクト名には大文字が使用されます。詳細については、*MQSeries MQSC コマンド・リファレンス* を参照してください。

ただし、MQSC 機能呼び出す **runmqsc** コマンドの引き数によっては、大文字小文字を区別するものがあります。38ページの『制御コマンドの使用』を参照してください。

MQSC 機能を対話式で使用する

コマンドを対話式に入力するには、TACL セッションをオープンし、次のように入力します。

```
runmqsc
```


この例では、キュー・マネージャー名が指定されていません。したがって、MQSC は、デフォルト・キュー・マネージャーによって処理されます。MQSC コマンドを入力することができます。たとえば、次のようにします。

```
MQSC>DEFINE QLOCAL (ORANGE.LOCAL.QUEUE)
```

コマンドが次の行に続くことを示すためには連結文字を使用する必要があります。

- 負符号 (-) は、コマンドが次の行の先頭に続くことを示します。
- 正符号 (+) は、コマンドが次の行の最初の空白でない文字に続くことを示します。

コマンドの入力は、連結文字でなく空白でもない行の最後の文字で終了します。

runmqsc コマンドは、標準の Compaq NSK ヒストリーおよび修正コマンド機能もサポートします。たとえば、次のようにします。

- **history** または **h** を入力すると、現在からさかのぼって 10 個のコマンドのリストが生成されます。
- **!n** (ここで、*n* はコマンド番号) を入力すると、そのコマンドが再実行されます
- **h n** または **history n** (ここで、*n* は数字) を入力すると、現在からさかのぼって *n* 個のコマンドがリストされます。
- **fc** を入力すると、編集のために入力された最後のコマンドが表示されます。**fc n** (ここで、*n* はコマンド番号) を入力すると、番号を指定した編集のためのコマンドが表示されます。**fc string** (*string* は前もって入力されたコマンドの開始部分またはすべての部分) をタイプすると、編集用のコマンドの最後の出現箇所が表示されます。構文は NSK 標準です。たとえば、文字を削除する場合は **d**、文字を挿入する場合は **i**、文字を置換する場合は **r** を入力します。

MQSC からのフィードバック

MQSC を発行すると、キュー・マネージャーは、確認メッセージまたはエラー・メッセージを表示します。たとえば、次のようにします。

MQSC コマンドの発行

```
AMQ8006: MQSeries queue created
.
.
.
AMQ8405: Syntax error detected at or near end of command segment below:-
```

最初のメッセージは、キューが作成されたことを確認するものですが、2 番目のメッセージは、構文エラーがあったことを示すものです。コマンドを正しく入力していなかった場合は、*MQSeries MQSC コマンド・リファレンス* を参照して正しい構文を確認してください。

対話式での MQSC への入力を終了する

MQSC を対話的に使用している場合は、EOF 文字の CTRL+Y を入力するか、exit、quit、または end とタイプしてから Enter を押すと、終了できます。

テキスト・ファイルなどの他の送信元から入力をリダイレクトしている場合は、MQSC は、ファイルの終わりに到達するときに終了します。

キュー・マネージャーの属性を表示する

runmqsc コマンドで指定されたキュー・マネージャーの属性を表示するには、次の MQSeries コマンドを使用します。

```
MQSC>DISPLAY QMGR ALL
```

一般的な出力を 123ページの図34 に示してあります。

```

1 : dis qmgr all
AMQ8408: Display Queue Manager details.
DESCR( )
DEFXMITQ(MY.DEFAULT.XMIT.QUEUE)
CLWLXIT( )
REPOS( )
COMMANDQ(SYSTEM.ADMIN.COMMAND.QUEUE)
CRDATE(2001-03-12)
ALTDATE(2001-03-12)
QMID(SIMONW_2001-03-12_09.24.30)
MAXHANDS(256)
AUTHOREV(ENABLED)
LOCALEV(ENABLED)
PERFMEV(ENABLED)
CHAD(DISABLED)
CLWLEN(100)
CCSID(819)
CMDLEVEL(510)
SYNCP
DEADQ(SYSTEM.DEAD.LETTER.QUEUE)
CHADEXIT( )
CLWLDATA( )
REPOSNL( )
QMNAME(saturn.queue.manager)
CRTIME(09.24.30)
ALTTIME(09.26.27)
TRIGINT(10000)
MAXUMSGS(10000)
INHIBTEV(ENABLED)
REMOOTEV(ENABLED)
STRSTPEV(ENABLED)
CHADDEV(ENABLED)
MAXMSGL(100000000)
MAXPRTY(9)
PLATFORM(NSK)
DISTL(YES)

```

図 34. QMGR ALL の出力例

DISPLAY QMGR コマンドに ALL パラメーターを指定すると、キュー・マネージャーのすべての属性が表示されます。出力では、キュー・マネージャー名 (saturn.queue.manager)、送達不能キューの名前 (SYSTEM.DEAD.LETTER.QUEUE)、およびコマンド・キューの名前 (SYSTEM.ADMIN.COMMAND.QUEUE) が示されます。 **crtmqm** コマンドで送達不能キューの名前を指定しない場合、送達不能キューをキュー・マネージャーと関連付けるようにキュー・マネージャーを更新しなければなりません。

次のコマンドを入力して、これらのキューが作成されることを確認しなければなりません。

```
DISPLAY QUEUE (*)
```

デフォルトではないキュー・マネージャーを使用する

runmqsc コマンドの実行時にキュー・マネージャー名を指定して、ローカル・キュー・マネージャー (デフォルト以外) に対して MQSC を実行することができます。たとえば、キュー・マネージャー jupiter.queue.manager に対して MQSC を実行するには、次のコマンドを使用します。

MQSC コマンドの発行

```
runmqsc jupiter.queue.manager
```

このキュー・マネージャーが同じノード上にあって、すでに実行されている場合、入力するすべての MQSC コマンドはこのキュー・マネージャーによって処理されます。

リモート・キュー・マネージャーに対しても MQSC コマンドを実行できません。

165ページの『リモートから MQSC コマンドを発行する』を参照してください。

キュー・マネージャーの属性の更新

runmqsc コマンドに指定されたキュー・マネージャーの属性を更新するには、変更したい属性および値を指定した MQSC コマンド **ALTER QMGR** を使用します。たとえば、`jupiter.queue.manager` の属性を更新するには、次のコマンドを使用します。

```
runmqsc jupiter.queue.manager  
ALTER QMGR DEADQ (ANOTHERDLQ) INHIBTEV (ENABLED)
```

ALTER QMGR コマンドにより、使用されている送達不能キューが変更され、禁止イベントが使用可能になります。

テキスト・ファイルから MQSCコマンドを実行する

MQSC を対話式で実行する方法は迅速なテストの実施には適していますが、長いコマンドや、繰り返し実行したい一連のコマンドの場合には、テキスト・ファイルから入力するようにしてください。

このタスクを実行するには、テキスト・エディターを使用して MQSC を含むテキスト・ファイルを作成してください。 **runmqsc** コマンドを使用する際に、**runmqsc** で **TACL IN** および **OUT** リダイレクト演算子、またはフラグ **-i** および **-o** を使用してください。たとえば、次のコマンドは、テキスト・ファイル `mymqscin` に含まれている一連のコマンドを実行します。

```
runmqsc /IN mymqscin/
```

または

```
runmqsc -i mymqscin
```

同様に、出力をファイルにリダイレクトすることもできます。入力用の MQSC が格納されているファイルを *MQSC* ファイル と呼びます。キュー・マネージャーからの応答が格納されている出力ファイルを *レポート・ファイル* と呼びます。

runmqsc コマンドで入力と出力の両方をリダイレクトするためには、次のコマンドを使用します。

```
runmqsc /IN mymqscin, OUT mymqscou/
```

または

```
runmqsc -i mymqscin -o mymqscou
```

このコマンドにより、ファイル *mymqscin* に含まれている MQSC コマンドが呼び出されます。キュー・マネージャー名を指定していないので、MQSC コマンドはデフォルト・キュー・マネージャーに対して実行されます。出力は、レポート・ファイル *mymqscou* に送られます。126ページの図35は、MQSC コマンド・ファイル *mymqscin* からの抜粋を示しており、128ページの図36は、*mymqscou* 中の対応する部分を示しています。

デフォルトのものではないキュー・マネージャー (*saturn.queue.manager*) について、**runmqsc** コマンドで入力と出力をリダイレクトするためには、次のコマンドを使用します。

```
runmqsc /IN mymqscin, OUT mymqscou/ saturn.queue.manager
```

または

MQSC コマンドの実行

```
runmqsc -i mymqscin -o mymqscou saturn.queue.manager
```

MQSC コマンド・ファイルの使用

MQSC コマンド・ファイルは EDIT ファイルとして作成されます (Compaq ファイル・タイプ・コード 101)。図35 は、MQSC ファイルからの抜粋で、属性が指定された MQSeries コマンド (DEFINE QLOCAL) を示しています。MQSeries MQSC コマンド・リファレンス には、各 MQSC コマンドとその構文が説明されています。

```
.  
. .  
DEFINE QLOCAL(ORANGE.LOCAL.QUEUE) REPLACE +  
  DESCR(' ') +  
  PUT(ENABLED) +  
  DEFPRTY(0) +  
  DEFPSIST(NO) +  
  GET(ENABLED) +  
  MAXDEPTH(5000) +  
  MAXMSGL(1024) +  
  DEFSOPT(SHARED) +  
  NOHARDENBO +  
  USAGE(NORMAL) +  
  NOTRIGGER  
.  
. .  
.
```

図 35. MQSC コマンド・ファイル mymqscin からの抽出

各行を最大で 72 文字に制限する必要があります。正符号 (+) は、コマンドが次の行に続くことを示します。正符号の前にスペースを入れなければならないことに注意してください。

MQSC レポートの使用

runmqsc コマンドはレポートを戻し、これは現行の OUT ストリームに送られます。レポートには、次のものが含まれます。

- MQSC をレポートの出所として識別するヘッダー。

Starting MQSeries Commands.

- 発行された MQSC コマンドの番号付きリスト (オプション)。デフォルトでは、入力のテキストが出力にエコーされます。この出力の中では、128ページの図36 に示されているように、各コマンドには順序番号が先頭に付けられています。ただし、**runmqsc** コマンドに **-e** フラグを指定すると、この出力を抑制することができます。
- エラーのあったコマンドについての構文エラー・メッセージ。
- 各コマンドの実行結果を示すオペレーター・メッセージ。たとえば、**DEFINE QLOCAL** コマンドの正常終了を示すオペレーター・メッセージは、次のとおりです。

AMQ8006: MQSeries queue created.

- スクリプト・ファイルの実行時の一般エラーのために出されるその他のメッセージ。
- 読み取られたコマンドの数、構文エラーのあったコマンドの数、処理できなかったコマンドの数を示す、レポートの簡単な統計の要約。

注: キュー・マネージャーが処理を試みるのは、構文エラーのなかったコマンドのみです。

MQSC コマンドの実行

```
Starting MQSeries Commands.
:
:
12:      DEFINE QLOCAL('RED.LOCAL.QUEUE') REPLACE +
:          DESCR(' ') +
:          PUT(ENABLED) +
:          DEFPRTY(0) +
:          DEFPSIST(NO) +
:          GET(ENABLED) +
:          MAXDEPTH(5000) +
:          MAXMSGL(1024) +
:          DEFSOPT(SHARED) +
:          USAGE(NORMAL) +
:          NOTRIGGER
AMQ8006: MQSeries queue created.
:
:
15 MQSC commands read.
0 commands have a syntax error.
0 commands cannot be processed.
```

図 36. MQSC レポート・ファイル *mymqscou* からの抽出

システムに提供された MQSC コマンド・ファイルを実行する

MQSeries for Compaq NSK をインストールすると、**AMQSCOS0** という MQSC ファイルが提供されます。このファイルには、サンプル・プログラムが使用するオブジェクトの定義が含まれています。このファイルは、デフォルトでは、サンプル・サブボリューム `$SYSTEM.ZMQSSMPL` にあります。

runmqsc を使用してコマンドを確認する

runmqsc コマンドを使用すると、MQSC コマンドを実際に実行しなくても、ローカル・キュー・マネージャーでそれらのコマンドを確認することができます。このステップを実行するには、**runmqsc** コマンドに `-v` フラグを設定してください。たとえば、次のようにします。

```
runmqsc -i mymqscin -o mymqscou -v
```

MQSC コマンド・ファイルに対して **runmqsc** を呼び出すと、キュー・マネージャーは、実際に MQSC コマンドを実行することなく、各コマンドを確認し

てレポートを戻します。これによって、コマンド・ファイル内のすべてのコマンドの構文を検査できます。このステップは、次のことを行う場合に重要です。

- コマンド・ファイルから多数のコマンドを実行する。
- MQSC コマンド・ファイルを何回も使用する。

このレポートは、128ページの図36 に示されているものと同様です。

リモートから MQSC コマンドを確認するには、この方法を用いることはできません。たとえば、次のコマンドを試行するとします。

```
runmqsc -i mymqscin -o mymqscou -w 30 -v jupiter.queue.manager
```

キュー・マネージャーがリモートであることを示すために使用する `-w` フラグは無視され、コマンドは検証モードでローカルで実行されます。

MQSC で起こった問題の解決

MQSC コマンドを実行できない場合は、次のチェックリストに記載された共通問題に該当するものがないか確認してください。

runmqsc コマンドを使用するとき、次の点に注意してください。

- `$$SYSTEM.ZMQSEXE` が `TACLCLSTM` 内の `PMSEARCH` にあることを検査します。
- ファイルから入力をリダイレクトする際に `IN` 演算子または `-i` フラグを使用します。そうしないと、キュー・マネージャーはそのファイル名をキュー・マネージャー名として解釈し、次のエラー・メッセージを発行します。

```
AMQ8118: MQSeries queue manager does not exist.
```

- 出力をファイルにリダイレクトする場合には、`OUT` 演算子または `-o` フラグを使用します。デフォルトでは、出力ファイルは、コマンドの発行時に効力のある `TACL` デフォルト値を使用して作成されます。出力を特定のファイルに送るには、完全修飾ファイル名を指定してください。
- コマンドを実行するキュー・マネージャーが作成されているかどうか確認します。これを行うためには、構成ファイル `MQSINI` を調べます。このファイルは、デフォルトではインストール・サブボリューム

MQSC で起こった問題

`$$SYSTEM.ZMQSSYS` にあります。このファイルには、キュー・マネージャーの名前と、すでに作成されている場合はデフォルト・キュー・マネージャーの名前が入っています。

- キュー・マネージャーはすでに開始されている必要があります。キュー・マネージャーが開始されていない場合には、86ページの『キュー・マネージャーの開始』で説明されているように開始します。すでに開始されている場合に開始しようとする、エラー・メッセージが出されます。
- デフォルト・キュー・マネージャーが定義されていない場合、**runmqsc** コマンドにキュー・マネージャー名を指定します。これを定義しなければ、次のようなエラー・メッセージが出されます。

```
AMQ8146: MQSeries queue manager not available.
```

この種の問題を訂正するには、89ページの『既存のキュー・マネージャーのデフォルト・キュー・マネージャーへの指定』を参照してください。

- MQSCコマンドを **runmqsc** パラメーターとして指定することはできません。たとえば、次のものは無効です。

```
runmqsc DEFINE QLOCAL(FRED)
```

- **runmqsc** コマンドを発行する前に、TACL から MQSC コマンドを入力することはできません。たとえば、次のようにします。

```
DEFINE QLOCAL(Queue1)
```

```
* Error Name of Variable, built-in, or file needed.
```

- **runmqsc** から制御コマンドを実行することはできません。たとえば、MQSC を対話式で実行すると、キュー・マネージャーを開始できません。

```
runmqsc
(C) Copyright IBM Corp. 1993, 2001. All Rights Reserved
Starting MQSeries Commands.

strmqm saturn.queue.manager
  1 : strmqm saturn.queue.manager
AMQ8405: Syntax error detected at or near end of command segment below:
```

167ページの『リモートからの MQSC の使用に問題がある場合』も参照してください。

ローカル・キューの取り扱い

この節では、使用できる MQSC コマンドの例をいくつか示します。これらのコマンドの詳細については、*MQSeries MQSC コマンド・リファレンス* を参照してください。

ローカル・キューを定義する

アプリケーションにとって、ローカル・キュー・マネージャーとは、アプリケーションが接続されているキュー・マネージャーです。ローカル・キュー・マネージャーによって管理されるキューは、そのキュー・マネージャーに対してローカルであるといいます。

ローカル・キューの定義を作成するため、またキューと呼ばれるデータ構造を作成するためには、MQSC コマンド `DEFINE QLOCAL` を使用します。デフォルト・ローカル・キューの特性からのキュー特性を修正することもできます。

この例では、`ORANGE.LOCAL.QUEUE` は、次のような特性を持つものとして指定します。

- 読み取りは可能、書き込みは不可、先入れ先出し法 (FIFO) で操作が行われる。
- 「通常の」キュー。つまり、開始キューや伝送キューではなく、トリガー・メッセージを生成しない。
- キューの最大サイズは、1000 個のメッセージで、最大メッセージ長は、2000 バイトである。

次の MQSC コマンドは、このアクションを実行します。

ローカル・キューの取り扱い

```
DEFINE QLOCAL (ORANGE.LOCAL.QUEUE) +
  DESCR('Queue for messages from other systems') +
  PUT (DISABLED) +
  GET (ENABLED) +
  NOTRIGGER +
  MSGDLVSQ (FIFO) +
  MAXDEPTH (1000) +
  MAXMSGL (2000) +
  USAGE (NORMAL)
```

注:

1. これらの属性のほとんどは、プロダクトで提供されているデフォルト値です。ただし、ここでは、これらは説明の目的で示してあります。デフォルト値が自分の希望するものであるか、デフォルト値が変更されていないことが確実ならば、これらは省略することができます。
2. USAGE (NORMAL) は、このキューが伝送キューではないことを示します。
3. 名前が ORANGE.LOCAL.QUEUE である同じキュー・マネージャーにすでにローカル・キューがあると、このコマンドは失敗します。既存のキューの定義を上書きする場合には、REPLACE 属性を使用してください。ただし、135ページの『ローカル・キューの属性を変更する』も参照してください。

キューの物理ファイル・サイズの変更

デフォルトでは、キュー・マネージャーは、最大 100 MB のデータをサポートするキュー・データ・ファイルを作成します。この限度に到達すると、アプリケーションは戻りコード MQRC_Q_SPACE_NOT_AVAILABLE を受け取ります。キューに割り振られた最大ストレージを変更するには、まず **dspmqls** を使用してキューのデータを保留にする物理ファイルを識別し、次に **altmqfls** コマンドを使用して、1 次および 2 次エクステン・サイズと最大エクステン・サイズを変更します。ファイルがすでに区画に分割されている場合には、**altmqfls** はすべての区画をサイズ変更します。詳細については、327ページの『altmqfls (キュー・ファイル属性の更新)』を参照してください。さらにストレージが必要な場合には、複数のボリュームにわたってファイルを区画に分割することができます。

送達不能キューを定義する

正しい宛先に送達できないメッセージを後で取り出すために保管することができるよう、各キュー・マネージャーは、送達不能キューとして使用されるローカル・キューを持っている必要があります。

送達不能キューについては、キュー・マネージャーに通知する必要があります。このことは、送達不能キューを `crtmqm` コマンドに指定することにより行えます。あるいは `ALTER QMGR` コマンドを使用して後でそれを指定することができます。送達不能キューを使用するためには、その前にそれを定義しておくことも必要です。

`SYSTEM.DEAD.LETTER.QUEUE` という名前のサンプル送達不能キューが、プロダクトと共に提供されています。このキューは、キュー・マネージャーが作成されるときに自動的に作成されます。必要ならば、この定義を修正できます。その名前を変更する必要はありません。

送達不能キューには、以下に示すものを除いて、特別な要件はありません。

- ローカル・キューでなければならない。
- その `MAXMSGL` (最大メッセージ長) 属性は、キュー・マネージャーが取り扱う最大メッセージおよび送達不能ヘッダー (`MQDLH`) をキューに収容できるようにしておく必要があります。

MQSeries には送達不能キュー・ハンドラーがあり、これによって、送達不能キュー上で見つかったメッセージの処理方法または除去方法を指定できます。詳細については、205ページの『第9章 MQSeries 送達不能キュー・ハンドラー』を参照してください。

デフォルト・オブジェクトの属性を表示する

MQSeries オブジェクトを定義した場合、指定していない属性はデフォルト・オブジェクトから取られます。たとえば、ローカル・キューを定義すると、このキューは、定義の中で省略された属性を、`SYSTEM.DEFAULT.LOCAL.QUEUE` と呼ばれるデフォルト・ローカル・キューから継承します。これらの属性を正確に知りたい場合には、次のコマンドを使用します。

```
DISPLAY QUEUE (SYSTEM.DEFAULT.LOCAL.QUEUE)
```

注: このコマンドの構文は、対応する `DEFINE` コマンドの構文とは異なっています。

属性を個別に指定すると、属性を選択的に表示できます。たとえば、次のようにします。

デフォルト・オブジェクトの属性を表示する

```
DISPLAY QUEUE (ORANGE.LOCAL.QUEUE) +  
MAXDEPTH +  
MAXMSGL +  
CURDEPTH
```

このコマンドにより、次のような 3 つの指定の属性が表示されます。

```
AMQ8409: Display Queue details.  
QUEUE(ORANGE.LOCAL.QUEUE)  
MAXDEPTH(1000)  
MAXMSGL(2000)  
CURDEPTH(0)
```

CURDEPTH は、現行キュー項目数、つまりキュー上のメッセージ数です。これは、表示すると便利な属性です。キュー項目数を監視することによって、キューが満杯にならないようにすることができます。

ローカル・キュー定義をコピーする

DEFINE コマンドに LIKE 属性を指定すると、キュー定義をコピーできます。たとえば、次のようにします。

```
DEFINE QLOCAL (MAGENTA.QUEUE) +  
LIKE (ORANGE.LOCAL.QUEUE)
```

このコマンドにより、システム・デフォルト・ローカル・キューの属性ではなく、コピー元のキュー ORANGE.LOCAL.QUEUE と同じ属性を持つキューが作成されます。

この同じ形式の DEFINE コマンドを使用して、キュー定義をコピーし、元のキューの属性をいくつか変更することもできます。たとえば、次のようにします。

```
DEFINE QLOCAL (THIRD.QUEUE) +  
LIKE (ORANGE.LOCAL.QUEUE) +  
MAXMSGL(1024)
```

このコマンドにより、キュー ORANGE.LOCAL.QUEUE の属性がキュー THIRD.QUEUE にコピーされ、新しいキューの最大メッセージ長は、2000 バイトではなく、1024 バイトになるように指定されます。

注:

1. DEFINE コマンドの LIKE 属性を使用した場合、キューの属性だけをコピーします。キュー上のメッセージはコピーしません。
2. LIKE を指定せずにローカル・キューを定義する場合、それは DEFINE LIKE(SYSTEM.DEFAULT.LOCAL.QUEUE) と同じになります。

ローカル・キューの属性を変更する

キューの属性は 2 とおりの方法で変更できます。つまり、ALTER QLOCAL コマンドを使用するか、あるいは REPLACE 属性を指定した DEFINE QLOCAL コマンドを使用する方法です。131ページの『ローカル・キューを定義する』では、キュー ORANGE.LOCAL.QUEUE を定義しました。ここで、たとえば、このキューの最大メッセージ長を 10 000 バイトに増やしたいとします。

- ALTER コマンドを使用

```
ALTER QLOCAL (ORANGE.LOCAL.QUEUE) MAXMSGL(10000)
```

このコマンドにより、1 つの属性、つまり最大メッセージ長の属性は変更されますが、他の属性はすべて変更されません。

- REPLACE オプションを指定した DEFINE コマンドを使用。たとえば、次のように指定します。

```
DEFINE QLOCAL (ORANGE.LOCAL.QUEUE) MAXMSGL(10000) REPLACE
```

このコマンドにより、最大メッセージ長だけでなく、他のすべての属性も変更されます。他のすべての属性にはデフォルト値が与えられます。このキューは、以前は書き込み禁止でしたが、これで書き込み可能になります。キュー SYSTEM.DEFAULT.LOCAL.QUEUE で指定されているとおり、変更されていない限り、書き込み可能はデフォルト値です。

既存のキューの最大メッセージ長を短くしても、既存のメッセージは影響を受けません。ただし、新しいメッセージはこの新しい基準に適合する必要があります。

ローカル・キューのポリシーを変更する

altmqfls コマンドを使用して、ローカルな事前定義キューが保管されるポリシーを変更します。これは、最適なパフォーマンスを得られるようにシステムのバランスを保つため、複数のポリシーにまたがってディスク入出力を広げるのに必要とされる場合があります。このコマンドが発行される前にキュー・マネージャーが開始されていなければならず、キュー自体がオープンしてはなりません。どの **altmqfls** コマンドでも 1 つのキューだけを指定することができます。 **altmqfls** コマンドの構文については、327ページの『altmqfls (キュー・ファイル属性の更新)』を参照してください。

ローカル・キューのオプションを変更する

altmqfls コマンドを使用して、次の事柄を変更します。

- キュー・マネージャーが開始するときにキューがディスクからキャッシュにロードされるかどうか。 **--qsoptions S** を使用します。
- キューが最初にロードされた後、キュー・サーバーが実行している間にキューがメモリー内に残されるかどうか。これが設定されると、キュー・サーバーは、キューの構造およびデータを、すでにロードされているメモリー内に保存します。設定されない場合、キューに関連したストレージは、アクセスされなくなるとキュー・サーバーのアドレス・スペースから除去されることがあります。 **--qsoptions L** を使用します。
- メッセージの保管に必要な余分のチェックポイント、余分の IPC メッセージ、および余分のメモリーを処理するために CPU の負荷が求められるにもかかわらず、フォールト・トレランスを提供して、非持続メッセージのチェックポイントがバックアップ・キュー・サーバーに対してとられるかどうか。 **--qsoptions C** を使用します。

注: コマンドを発行するたびに、**すべての --qsoptions SLC** が設定されます。たとえば、**--qsoptions S** は **L** と **C** の設定を解除します。

--qsoptions オプションはコマンド行で 1 回限り指定することができます。

- それぞれの持続メッセージが (ディスク上だけでなく) キュー・サーバーのキャッシュ内に保持できるデータの最大バイト数。設定されるバイト数がメモリーに保持され、ブラウザ操作によりディスクにアクセスしなくてもこのデータがアプリケーションに戻されます。これを使用すると、キュー・サーバーによって使用されるメモリー・リソースを増やすことができます。**--browse** パラメーターを使用します。
- メッセージ・データを保管するためにメッセージ・オーバーフロー・ファイルによって使用される最小メッセージ・サイズ。このサイズよりも小さい持

続メッセージは、キュー・オーバーフロー・ファイルに保管されます。このサイズよりも大きい持続メッセージは、専用のメッセージ・オーバーフロー・ファイルに大量のデータを保管します。 --msgofthresh パラメーターを使用します。

- キュー・サーバーが新規のメッセージ・オーバーフロー・ファイルを作成するときに、その作成場所であるキューが存在するボリューム上のサブボリューム。 --msgofsubvol パラメーターを使用します。
- 現行の項目数に初期設定され、その後、メッセージがキューに追加またはキューから除去される際にキューを担当するキュー・サーバーによって増減される測定カウンターの名前 (実行中の測定の一部である場合)。 --meascount パラメーターを使用します。
- キュー・ファイルの 1 次および 2 次エクステント・サイズおよび最大エクステント数。 --qsize (primaryextent,secondaryextent,maxextents) パラメーターを使用します。 --qsize パラメーターを使用して、すべての値が指定されなければなりません。
- キュー・オーバーフロー・ファイルの 1 次および 2 次エクステント・サイズおよび最大エクステント数。 --osize (primaryextent,secondaryextent,maxextents) パラメーターを使用します。 --osize パラメーターを使用して、すべての値が指定されなければなりません。

状況サーバーおよびキュー・サーバーにオブジェクトを再割り当てる

状況サーバーは、ローカル・キューおよびモデル・キューを除くすべてのオブジェクトを処理します。キュー・サーバーは、すべてのローカル・キューおよびモデル・キューのオブジェクトを処理します。

最初に、すべてのオブジェクトは、それぞれのオブジェクトに応じてデフォルトのキュー・サーバーまたは状況サーバーを使用するために作成されます。オブジェクトを作成した後に **altmqfls** を使用して、デフォルト以外の適切なサーバーを使用するようにオブジェクトを構成することができます。オブジェクトを使用するには、その前にサーバー用に新しい **PATHWAY** サーバー・クラスを構成しており、それを開始していなければなりません。

altmqfls のコマンド行にプロセス名または **DEFAULT** という語のいずれかを指定することができます。オブジェクトの再構成時に、新しいサーバーがアクティブであるか、または構成されているかどうかを調べる検査は実行されません。

オブジェクトの再割り当て

dspmqlfs コマンドを使用すると、オブジェクト用の現行状況サーバーを表示することができます。

ローカル・キューおよびモデル・キュー用のキュー・サーバーを設定することに注意してください。動的キューは、それらを作成するために使用されるモデル・キューからキュー・サーバーを継承します。

ローカル・キューをクリアする

MAGENTA.QUEUE という名前のローカル・キューからすべてのメッセージを削除するためには、次のコマンドを使用します。

```
CLEAR QLOCAL (MAGENTA.QUEUE)
```

次の場合には、キューの内容をクリアすることができません。

- 同期点でコミットされていないメッセージで、そのキューに書き込まれているものがある場合
- アプリケーションがそのキューを現在オープンしている場合

ローカル・キューを削除する

ローカル・キューを削除するには、MQSC コマンド **DELETE QLOCAL** を使用します。キュー上にコミットされていないメッセージがある場合、そのキューは削除できません。ただし、キューがコミットされたメッセージを 1 つまたは複数持っているが、コミットされていないメッセージがない場合は、**PURGE** オプションを指定した場合にのみ削除することができます。たとえば、次のようになります。

```
DELETE QLOCAL (PINK.QUEUE) PURGE
```

PURGE の代わりに **NOPURGE** を指定すると、コミットされたメッセージがキューに含まれている場合、そのキューが削除されることはありません。

キューをブラウズする

キュー上のメッセージの内容を調べる必要がある場合、MQSeries for Compaq NSK では、サンプル・キュー・ブラウザーが用意されています。ブラウザーは、ソースとしても実行可能なモジュールとしても提供されています。デフォルトでは、ファイル名とパスは次のとおりです。

ソース \$SYSTEM.ZMQSSMPL.AMQSBCG0

実行可能モジュール

\$SYSTEM.ZMQSSMPL.AMQSBCG

サンプルは、次の 2 つのパラメーターをとります。

キュー名 たとえば、SYSTEM.ADMIN.RESPQ.tpp01。

キュー・マネージャー名

たとえば、snooker。

たとえば、次のようにします。

```
AMQSBCG SYSTEM.ADMIN.RESPQ.tpp01 snooker
```

デフォルトはありません。パラメーターは両方とも必須です。このコマンドの一般的な結果を次に示します。

AMQSBCG - starts here

MQCONN to snooker

MQOPEN - 'SYSTEM.ADMIN.RESPQ.tpp01'

MQGET of message number 1

Message descriptor

```
StrucId : 'MD ' Version : 1
Report : 0 MsgType : 8
Expiry : -1 Feedback : 0
Encoding : 273 CodedCharSetId : 850
Format : 'AMQMRESP'
Priority : 5 Persistence : 1
MsgId : X'414D5120736E6F6F6B65722020202020ED47690071A6D00'
CorrelId : X'000000000000000000000000000000000000000000000000'
BackoutCount : 0
ReplyToQ : '
ReplyToQMgr : 'snooker'
** Identity Context
UserIdentifier : 'tpp01'
AccountingToken :
X'043730373000000000000000000000000000000000000000000000000000'
ApplIdentityData : '
** Origin Context
PutApplType : '6'
PutApplName : '
PutDate : '19941124' PutTime : '11184015'
ApplOriginData : '

```


MQGET of message number 3

****Message descriptor****

```

StrucId : 'MD ' Version : 1
Report  : 0 MsgType : 8
Expiry  : -1 Feedback : 0
Encoding : 273 CodedCharSetId : 850
Format  : 'AMQMRESP'
Priority : 5 Persistence : 1
MsgId   : X'414D5120736E6F6F6B6572202020202020ED477D62A9EA100'
CorrelId : X'0000000000000000000000000000000000000000000000000000000000000000'
BackoutCount : 0
ReplyToQ      : '
ReplyToQMgr   : 'snooker
** Identity Context
UserIdentifier : 'trevor
AccountingToken :
X'04373037300000000000000000000000000000000000000000000000000000000000'
ApplIdentityData : '
** Origin Context
PutApplType    : '6'
PutApplName    : '
PutDate       : '19941124' PutTime : '11240678'
ApplOriginData : '

```

**** Message ****

length - 188 bytes

```

00000000: 736E 6F6F 6B65 7220 2020 2020 2020 2020 'snooker
00000010: 2020 2020 2020 2020 2020 2020 2020 2020 '
00000020: 2020 2020 2020 2020 2020 2020 2020 2020 '
00000030: 534E 4F4F 4B45 522E 5749 4748 542E 5443 'SNOOKER.WIGHT.TC'
00000040: 5020 2020 2020 2020 2020 2020 2020 2020 'P
00000050: 2020 2020 2020 2020 2020 2020 2020 2020 '
00000060: 0000 0001 0000 0024 0000 0001 0000 0015 '.....$......'
00000070: 0000 0001 0000 0001 0000 0000 0000 0000 '.....'
00000080: 0000 0002 0000 0004 0000 0028 0000 0DAD '.....(.....'
00000090: 0000 0000 0000 0014 534E 4F4F 4B45 522E '.....SNOOKER.'
000000A0: 5749 4748 542E 5443 5020 2020 0000 0003 'WIGHT.TCP ....'
000000B0: 0000 0010 0000 05E7 0000 0001 '.....'

```

MQGET of message number 4

****Message descriptor****

```

StrucId : 'MD ' Version : 1
Report  : 0 MsgType : 2
Expiry  : -1 Feedback : 0
Encoding : 273 CodedCharSetId : 850
Format  : 'MQADMIN '
Priority : 8 Persistence : 1
MsgId   : X'414D5120736E6F6F6B65722020202020ED477D63826C000'
CorrelId : X'414D5120736E6F6F6B65722020202020ED477D62A9EA100'

```

キューのブラウズ

```
BackoutCount : 0
ReplyToQ      : '
ReplyToQMgr   : 'snooker'
** Identity Context
UserIdentifier : 'tiger'
AccountingToken :
X'04373037300000000000000000000000000000000000000000000000000000000000'
ApplIdentityData : '
** Origin Context
PutApplType   : '6'
PutApplName   : '
PutDate       : '19941124'   PutTime : '11240694'
ApplOriginData : '

**** Message ****

length - 36 bytes

00000000: 0000 0002 0000 0024 0000 0001 0000 0015 '..$..'
00000010: 0000 0001 0000 0001 0000 0000 0000 0000 '..'
00000020: 0000 0000                                     '....'

No more messages
MQCLOSE
MQDISC
```

別名キューの取り扱い

別名キュー (キュー別名としても知られています) は、MQI 呼び出しのリダイレクトの方法を提供します。別名キューは、実際のキューではなく、実際のキューに解決される定義です。別名キュー定義は、TARGQ 属性 (PCF では *BaseQName*) で指定される宛先キュー名を含んでいます。アプリケーションが MQI 呼び出しの中で別名キューを指定すると、キュー・マネージャーは実行時に実際のキュー名に解決します。

たとえば、MY.ALIAS.QUEUE という名前のキューにメッセージを入れるようなアプリケーションが開発されたとします。このアプリケーションは、MQOPEN 要求を出すときにこのキューの名前を指定し、メッセージをこのキューに書き込む場合には、間接的にこのキューの名前を指定します。アプリケーションは、キューが別名キューであることに気付きません。この別名を使用した各 MQI 呼び出しについて、キュー・マネージャーは実際のキュー名に解決します。このキュー名はローカル・キューか、このキュー・マネージャーに定義されたリモート・キューのいずれかです。

TARGQ 属性の値を変更することにより、MQI 呼び出しを別のキュー（おそらく別のキュー・マネージャー上の別のキュー）にリダイレクトできます。これは、保守、マイグレーション、および負荷平衡に役立ちます。

別名キューを定義する

次のコマンドにより、別名キューが作成されます。

```
DEFINE QALIAS (MY.ALIAS.QUEUE) TARGQ (YELLOW.QUEUE)
```

このコマンドは、MQI 呼び出し (MY.ALIAS.QUEUE を指定している) をキュー YELLOW.QUEUE にリダイレクトします。このコマンドは、ターゲット・キューを作成しないので、キュー YELLOW.QUEUE が実行時に存在しなければ、MQI 呼び出しは失敗します。

別名定義を変更すると、MQI 呼び出しを別のキューにリダイレクトできます。たとえば、次のようにします。

```
DEFINE QALIAS (MY.ALIAS.QUEUE) TARGQ (MAGENTA.QUEUE) REPLACE
```

このコマンドは、MQI 呼び出しを別のキュー MAGENTA.QUEUE にリダイレクトします。

別名キューを使用すると、単一のキュー（ターゲット・キュー）が、異なるアプリケーションについては異なる属性を持っているように見えるようにすることもできます。これは、アプリケーションごとに 1 つの別名、つまり合計 2 つの別名を定義すると行えます。2 つのアプリケーションがあるとします。

- アプリケーション ALPHA は、メッセージを YELLOW.QUEUE に書き込むことができますが、そこからメッセージを読み取ることはできません。
- アプリケーション BETA は、YELLOW.QUEUE からメッセージを読み取ることはできますが、そこにメッセージを書き込むことはできません。

これは、次のコマンドを使用して行います。

別名キューを定義する

```
* This alias is put enabled and get disabled for application ALPHA

DEFINE QALIAS (ALPHAS.ALIAS.QUEUE) +
    TARGQ (YELLOW.QUEUE) +
    PUT (ENABLED) +
    GET (DISABLED)

* This alias is put disabled and get enabled for application BETA

DEFINE QALIAS (BETAS.ALIAS.QUEUE) +
    TARGQ (YELLOW.QUEUE) +
    PUT (DISABLED) +
    GET (ENABLED)
```

ALPHA は、MQI 呼び出しの中でキュー名 ALPHAS.ALIAS.QUEUE を使用しますが、BETAは、キュー名 BETAS.ALIAS.QUEUE を使用します。これらはいずれも同じキューをアクセスしますが、その方法は異なっています。

キュー別名を定義する際には、ローカル・キューの場合と同様にして、LIKE 属性および REPLACE 属性を使用することができます。

キュー別名に関してその他のコマンドを使用する

該当の MQSC コマンドを使用すると、キュー別名の属性を表示または更新したり、キュー別名オブジェクトを削除したりできます。たとえば、次のようにします。

```
* Display the queue alias' attributes
* ALL = Display all attributes

DISPLAY QUEUE (ALPHAS.ALIAS.QUEUE) ALL

* ALTER the base queue name, to which the alias resolves.
* FORCE = Force the change even if the queue is open.

ALTER QALIAS (ALPHAS.ALIAS.QUEUE) TARGQ(ORANGE.LOCAL.QUEUE) FORCE

* Delete this queue alias, if you can.

DELETE QALIAS (ALPHAS.ALIAS.QUEUE)
```

キュー別名は、たとえば、アプリケーションがそのキューを現在オープンしていたり、このキューに解決されるキューをオープンしている場合には、削除す

ることができません。このキュー別名およびその他のキュー別名コマンドの詳細については、*MQSeries MQSC コマンド・リファレンス* を参照してください。

モデル・キューの取り扱い

キュー・マネージャーは、モデル・キューとして定義されているキュー名を指定した MQI 呼び出しをアプリケーションから受け取ると、動的キューを作成します。新しい動的キューの名前は、そのキューの作成時にキュー・マネージャーによって生成されます。モデル・キューとは、動的キューの属性を指定しているテンプレートのことで、動的キューはこのモデル・キューから作成されます。

モデル・キューは、アプリケーションがキューを必要とするときにそのキューを作成するための便利な方法を提供します。

モデル・キューを定義する

ローカル・キューを定義するのと同じ方法で、1組の属性を持つモデル・キューを定義します。モデル・キューには、作成される動的キューが一時キューとなるか持続キューとなるかを指定できること以外は、モデル・キューとローカル・キューは同じ1組の属性を持っています。(持続キューはキュー・マネージャーが再始動しても維持されますが、一時キューは維持されません。)たとえば、次のようにします。

```
DEFINE QMODEL (GREEN.MODEL.QUEUE) +
  DESCR('Queue for messages from application X') +
  PUT (DISABLED) +
  GET (ENABLED) +
  NOTRIGGER +
  MSGDLVSQ (FIFO) +
  MAXDEPTH (1000) +
  MAXMSGL (2000) +
  USAGE (NORMAL) +
  DEFTYPE (PERMDYN)
```

このコマンドにより、モデル・キュー定義が作成されます。DEFTYPE 属性により、このテンプレートから作成される実際のキューは、持続動的キューになります。

注: 指定されていない属性は、SYSYSTEM.DEFAULT.MODEL.QUEUE デフォルト・キューから自動的にコピーされます。

モデル・キューを定義する

モデル・キューを定義する際には、ローカル・キューの場合と同様にして、LIKE 属性および REPLACE 属性を使用することができます。

モデル・キューに関してその他のコマンドを使用する

該当の MQSC コマンドを使用すると、モデル・キューの属性を表示または更新したり、モデル・キュー・オブジェクトを削除したりできます。たとえば、次のようにします。

```
* Display the model queue's attributes
* ALL = Display all attributes

DISPLAY QUEUE (GREEN.MODEL.QUEUE) ALL

* ALTER the model to enable puts on any
* dynamic queue created from this model.

ALTER QMODEL (BLUE.MODEL.QUEUE) PUT(ENABLED)

* Delete this model queue:

DELETE QMODEL (RED.MODEL.QUEUE)
```

トリガー操作のためのオブジェクトの管理

MQSeries には、キューで特定の条件が満たされると自動的にアプリケーションを開始するための機能があります。その条件の一例として、キュー上のメッセージ数が指定の数に達した場合があります。この機能は、トリガー操作と呼ばれていて、詳細は、*MQSeries アプリケーション・プログラミング・ガイド* に記載されています。この節では、MQSeries for Compaq NSK でのトリガー操作をサポートするために必要なオブジェクトのセットアップ方法を説明します。

トリガー操作のためにアプリケーション・キューを定義する

アプリケーション・キューとは、アプリケーションが MQI を介してメッセージング用に使用するローカル・キューのことです。トリガー操作では、いくつかのキュー属性をアプリケーション・キューに定義する必要があります。トリガー操作自体は、*Trigger* 属性 (MQSC の中の TRIGGER) によって使用可能になります。

以下に示す例では、トリガー・イベントは、MOTOR.INSURANCE.QUEUE というローカル・キューに優先順位 5 以上のメッセージが 100 個入れられたときに生成されます。

```
DEFINE QLOCAL (MOTOR.INSURANCE.QUEUE) +  
        PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) +  
        MAXMSGL (2000) +  
        DEFPSIST (YES) +  
        INITQ (MOTOR.INS.INIT.QUEUE) +  
        TRIGGER +  
        TRIGTYPE (DEPTH) +  
        TRIGDPTH (100)+  
        TRIGMPRI (5)
```

説明

QLOCAL (MOTOR.INSURANCE.QUEUE)

定義するアプリケーション・キューの名前を指定します。

PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)

トリガー・モニター・プログラムによって開始するアプリケーションの名前を指定します。

MAXMSGL (2000)

キューに入れるメッセージの最大長を指定します。

DEFPSIST (YES)

メッセージがこのキューでは持続であることを指定します。

INITQ (MOTOR.INS.INIT.QUEUE)

キュー・マネージャーがトリガー・メッセージを入れる開始キューの名前です。

TRIGGER

トリガー属性値です。

TRIGTYPE (DEPTH)

要求した優先順位 (TRIMPRI) を持つメッセージの数が TRIGDPTH で指定した数に達したときにトリガー・イベントを生成するように指定します。

TRIGDPTH (100)

トリガー・イベントを生成するのに必要なメッセージ数を指定します。

TRIGMPRI (5)

トリガー・イベントを生成するかどうかを決める際に、キュー・マネー

トリガー操作のためのアプリケーション・キュー

ジャーが考慮に入れるメッセージの優先順位です。優先順位 5 以上のメッセージだけが考慮に入れます。

開始キューを定義する

トリガー・イベントが発生すると、キュー・マネージャーは、アプリケーション・キュー定義に指定された開始キューにトリガー・メッセージを入れます。開始キューには特別の設定値はありませんが、参考として以下に示す `MOTOR.INS.INIT.QUEUE` というローカル・キューの定義を使用することができます。

```
DEFINE QLOCAL(MOTOR.INS.INIT.QUEUE) +
  GET (ENABLED) +
  NOSHARE +
  NOTRIGGER +
  MAXMSGL (2000) +
  MAXDEPTH (10)
```

プロセス定義を作成する

プロセス定義を作成するには、`DEFINE PROCESS` コマンドを使用します。プロセス定義は、アプリケーション・キューを、キューからのメッセージを処理するアプリケーションと関連付けます。この関連付けは、アプリケーション・キュー `MOTOR.INSURANCE.QUEUE` の `PROCESS` 属性によって行われます。次の `MQSC` コマンドは、この例で識別されている必須プロセス `MOTOR.INSURANCE.QUOTE.PROCESS` を定義します。

```
DEFINE PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) +
  DESCR ('Insurance request message processing') +
  APPLTYPE (NSK) +
  APPLICID ('$DATA1.TEST.IRMP01') +
  USERDATA ('open, close, 235')
```

説明

MOTOR.INSURANCE.QUOTE.PROCESS

プロセス定義の名前です。

DESCR ('Insurance request message processing')

キーワードに続く、定義が関連付けられているアプリケーション・プログラムの説明テキストです。このテキストは、`DISPLAY PROCESS` コマンドを使用すると表示されます。これは、プロセスが行う事柄を識別

するのに役立ちます。ストリングの中でスペースを使用する場合は、ストリングを単一引用符で囲む必要があります。

APPLTYPE(NSK)

Compaq NSK 上で実行されるアプリケーションのタイプです。

APPLICID ('\$DATA1.TEST.IRMP01')

ローカル・システム上のアプリケーション実行可能プログラムの名前です。

USERDATA ('open, close, 235')

アプリケーションで使用できるユーザー定義のデータです。

プロセス定義を表示する

定義の結果を調べるには、ALL キーワードを指定して DISPLAY PROCESS コマンドを使用します。たとえば、次のようにします。

```
DISPLAY PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) ALL

24 : DISPLAY PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) ALL
AMQ8407: Display Process details.
DESCR (Insurance request message processing)
APPLICID ($DATA1.TEST.IRMP01)
ENVRDATA ( )
USERDATA (open, close, 235)
PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)
APPLTYPE (NSK)
```

USERDATA は、トリガー操作されたアプリケーションに渡される引き数を表すストリングです。トリガー・モニターおよびトリガー操作されたアプリケーションを作成する方法の例については、サンプル・プログラム AMQSTRG0 および AMQINQA (ZMQSSMPL サブボリュームにある) を参照してください。

MQSC コマンド ALTER PROCESS を使用して既存のプロセス定義を更新したり、DELETE PROCESS を使用してプロセス定義を削除することもできます。

第6章 管理用タスクの自動化

この章の内容を理解するためには、MQSeries オブジェクト管理の経験が必要です。

タスクのモニターおよび一部の管理を自動化することがインストールに役立つと判断する場合があります。プログラム式コマンド形式 (PCF) コマンドを使用して、ローカル・キュー・マネージャーおよびリモート・キュー・マネージャー両方の管理タスクを自動化することができます。

この章は次のことについて説明します。

- プログラム式コマンド形式を使用して、PCF コマンドを使用した管理の実行の管理用タスクを自動化する方法。
- 154ページの『リモート管理でコマンド・サーバーを管理する』におけるコマンド・サーバーの使用方法

PCF コマンド

MQSeries プログラム式コマンド形式 (PCF) コマンドの目的は、管理タスクを管理プログラムに組み込めるようにすることです。これにより、プログラムから、キュー、プロセス定義、チャンネル、および名前リストを作成したり、キュー・マネージャーを変更したりすることができます。

PCF コマンドは、MQSC 機能が提供する機能と同様の処理を行います。

したがって、単一のノードからネットワーク内の任意のキュー・マネージャーへ PCF コマンドを発行するプログラムを作成することができます。これにより、管理タスクを中央集中方式にすると同時に自動化することができます。

各 PCF コマンドは、MQSeries メッセージのアプリケーション・データ部分に組み込まれたデータ構造です。各コマンドは、他のメッセージの場合と同様に、MQI 機能 MQPUT を使用して宛先キュー・マネージャーに送られます。メッセージを受信するキュー・マネージャー上のコマンド・サーバーは、そのコマンドをコマンド・メッセージとして解釈し、実行します。応答を入手するには、アプリケーションが MQGET 呼び出しを発行します。すると、応答データが別のデータ構造に戻されます。次に、アプリケーションはその応答を処理し、その応答に応じてアクションを実行します。

PCF コマンド

注: MQSC コマンドとは異なり、PCF コマンドおよびそれらの応答は、テキスト形式では読み取ることはできません。

次に、PCF コマンド・メッセージを作成するためにアプリケーション・プログラマーが指定する必要がある事項のいくつか簡単に示します。

メッセージ記述子

これは、標準 MQSeries メッセージ記述子です。この中で、次のように指定します。

メッセージ・タイプ (*MsgType*) には、MQMT_REQUEST を指定します。

メッセージ形式 (*Format*) には、MQFMT_ADMIN を指定します。

アプリケーション・データ

これには、PCF ヘッダーを含む PCF メッセージが入ります。この中で、

PCF メッセージ・タイプ (*Type*) には MQCFT_COMMAND を指定します。

コマンド ID には、*Change Queue* (MQCMD_CHANGE_Q) などのコマンドを指定します。

PCF データ構造およびそれをインプリメントする方法の詳しい説明については、*MQSeries プログラム式システム管理* を参照してください。

MQSC および PCF での属性

MQSC に指定するオブジェクト属性は、本書では大文字 (たとえば、RQMNAME) で示されます。ただし、大文字小文字の区別はありません。MQSC 属性名は 8 文字までに制限されています。

PCF でのオブジェクト属性はイタリックで示され、8 文字までに制限されていません。たとえば、PCF では、RQMNAME に相当するものは *RemoteQMgrName* です。

エスケープ PCF

エスケープ PCF は、メッセージ・テキスト内に MQSC コマンドを含んでいる PCF コマンドです。PCF を使用して、リモート・キュー・マネージャーにコマンドを送信することができます。エスケープ PCF の使い方の詳細については、*MQSeries プログラム式システム管理* を参照してください。

MQAI を使用して PCF の使い方を単純化する

MQAI は、Compaq NSK 上で現在使用できる MQSeries に対する管理インターフェースです。

これは、データ・バッグを使用することにより、キュー・マネージャー上の管理タスクを実行します。データ・バッグを使用すると、PCF を使用するよりも簡単な方法で、オブジェクトのプロパティ（またはパラメーター）を処理することができます。

MQAI は、次のことを実行するために使用できます。

- **PCF メッセージの使い方を単純化する。** MQAI は、MQSeries を管理する簡単な方法です。独自の PCF メッセージを書き込む必要がないので、複雑なデータ構造に関連する問題を回避することができます。
MQI 呼び出しを使用して書き込まれたプログラムにおいてパラメーターを渡すには、PCF メッセージにコマンドおよびストリングまたは整数データの詳細を入れる必要があります。これを行うには、すべての構造についてプログラムに複数のステートメントを指定する必要があり、メモリー・スペースを割り振る必要があります。このタスクは、時間がかかる大変な作業です。
一方、MQAI を使用して書き込まれたプログラムはパラメーターを適切なデータ・バッグに渡し、構造ごとに1つのステートメントしか必要ありません。MQAI データ・バッグを使用すると、配列を処理したり、ストレージを割り振る必要がなく、PCF の詳細を入れる必要がある程度なくなります。
- **エラー条件をより簡単に処理する。** MQSC コマンドから戻りコードを戻すのは難しいですが、MQAI によりエラー条件の処理が簡単になります。

データ・バッグを作成し入力した後、mqExecute 呼び出しを使用して、管理コマンド・メッセージをキュー・マネージャーのコマンド・サーバーに送信することができます。mqExecute 呼び出しはどの応答メッセージも待ちます。mqExecute 呼び出しは、コマンド・サーバーとの交換を処理し、応答を応答バッグに返します。

MQAI の使い方の詳細については、MQSeries 管理インターフェースのプログラミングの手引きおよび解説書を参照してください。

一般的な PCF の詳細については、MQSeries プログラム式システム管理を参照してください。

リモート管理でコマンド・サーバーを管理する

各キュー・マネージャーには、それぞれに関連付けられた 1 つのコマンド・サーバーがあります。コマンド・サーバーは、リモート・キュー・マネージャーからの着信コマンド、またはアプリケーションからの PCF コマンドを処理します。コマンド・サーバーは処理のために、そのコマンドをキュー・マネージャーに渡し、コマンドの発信元に応じて、完了コードやオペレーター・メッセージを戻します。

コマンド・サーバーは、PCF、MQAI に関するすべての管理およびリモート管理にも必須です。

注: リモート管理では、宛先キュー・マネージャーを確実に実行しているようにする必要があります。実行していないと、コマンドを含んだメッセージは、メッセージの発信元のキュー・マネージャーから出ていくことができません。代わりに、それらのメッセージは、リモート・キュー・マネージャーが使用しているローカル伝送キューに保持されます。こうした状況は、できるだけ回避する必要があります。

コマンド・サーバーを開始する

コマンド・サーバーを開始するには、次のコマンドを使用します。

```
strmqcsv saturn.queue.manager
```

ここで、`saturn.queue.manager` は、開始されるコマンド・サーバーに関連したキュー・マネージャーです。

コマンド・サーバーは、そのサーバー・クラスを解凍して開始することによっても、PATHCOM で開始できます。

コマンド・サーバーの状況を表示する

リモート管理では、宛先キュー・マネージャー上でコマンド・サーバーが実行中であることを確認します。これが実行されていないと、リモート・コマンドを処理できません。コマンドを含んだメッセージは、宛先キュー・マネージャーのコマンド・キューに入れられます。

キュー・マネージャー（ここでは `saturn.queue.manager`）のコマンド・サーバーの状況を表示するには、次のコマンドを出します。

```
dspmqcsv saturn.queue.manager
```

このコマンドは、ターゲット・マシンに対して出す必要があります。コマンド・サーバーが実行されていると、次のメッセージが戻されます。

```
AMQ8027    MQSeries Command Server Status ...: Running
```

コマンド・サーバーを停止する

コマンド・サーバーを停止するには、前の例を使用して、次のコマンドを出します。

```
endmqcsv saturn.queue.manager
```

コマンド・サーバーを停止するには、次の 2 つの方法があります。

- 制御された停止の場合、`-c` フラグを指定した **endmqcsv** コマンドを使用します。これはデフォルトです。
- 即時停止の場合、`-i` フラグを指定した **endmqcsv** コマンドを使用します。

注: キュー・マネージャーを停止すると、そのキュー・マネージャーと関連付けられているコマンド・サーバーも (開始されている場合) 終了します。

第7章 リモート MQSeries オブジェクトの管理

この章では、別のキュー・マネージャー上の MQSeries オブジェクトを管理する方法について説明します。また、リモート・キュー・オブジェクトを使用して、メッセージおよび応答メッセージの宛先を制御する方法についても説明します。

この章には、次の節があります。

- 160ページの『リモート・キュー・マネージャーの管理』
- 168ページの『リモート・キューのローカル定義の作成』
- 172ページの『リモート・キュー定義を別名として使用する』

チャンネル、その属性およびセットアップ方法の詳細については、*MQSeries 相互通信* を参照してください。

チャンネル、クラスターおよびリモート・キューイング

キュー・マネージャーはメッセージを送信し、必要な場合には、戻される応答を受信して他のキュー・マネージャーと通信します。受信側キュー・マネージャーは、次のようになります。

- 同じマシン上
- 同じ場所または別の場所にある別のマシン上
- ローカル・キュー・マネージャーと同じプラットフォームで実行
- MQSeries がサポートする別のプラットフォームで実行

メッセージの発信元としては次のようなものがあります。

- あるノードから別のノードにデータを転送するユーザー作成アプリケーション・プログラム
- PCF または MQAI を使用するユーザー作成管理アプリケーション。
- キュー・マネージャーは、次のものを送信します。
 - 別のキュー・マネージャーにインストゥルメンテーション・イベント・メッセージを送信する。
 - **runmqsc** コマンドから発行される MQSC コマンドを間接モードで送信する (この場合、コマンドは別のキュー・マネージャーで実行されます)。

リモート・オブジェクトの管理

メッセージをリモート・キュー・マネージャーに送信する前に、ローカル・キュー・マネージャーには、メッセージの到着を検出したり、メッセージを転送するメカニズムが必要です。そのメカニズムは次のもので構成されます。

- 少なくとも 1 つのチャンネル
- 伝送キュー
- メッセージ・チャンネル・エージェント (MCA)
- チャンネル・リスナー
- チャンネル・イニシエーター

チャンネルは、2 つのキュー・マネージャー間の単方向通信リンクであり、リモート・キュー・マネージャーの多数のキューにメッセージを伝送することができます。

チャンネルの各端は、個別に定義されます。たとえば、一方の端が送信側、つまりサーバーであれば、他方の端は受信側、つまり要求側にする必要があります。簡単なチャンネルは、ローカル・キュー・マネージャーの端の送信側チャンネル定義 およびリモート・キュー・マネージャーの端の受信側チャンネル定義 で構成されます。2 つの定義の名前は同じにして、共に単一のチャンネルを構成する必要があります。

リモート・キュー・マネージャーを、ローカル・キュー・マネージャーが送信したメッセージに対して応答するよう設定する場合、2 番目のチャンネルは、ローカル・キュー・マネージャーに応答を戻すようにセットアップする必要があります。

チャンネルは `MQSC DEFINE CHANNEL` コマンドを使用して定義されます。この章では、特に断りがない限り、チャンネルに関連した例はデフォルト・チャンネル属性を使用しています。

チャンネルの各端にはメッセージ・チャンネル・エージェント (MCA) があり、メッセージの送受信を制御します。伝送キューからメッセージを受け取り、キュー・マネージャー間の通信リンクにメッセージを書き込むジョブは、MCA が担当します。それとは逆に、メッセージを通信リンクから取り出して、ターゲット・キューに書き込むジョブは、受信側 MCA が担当します。

伝送キューは、専用ローカル・キューであり、メッセージを MCA が受信し、リモート・キュー・マネージャーに送信する前に、一時的にメッセージを保管します。リモート・キュー定義 上の伝送キューの名前を指定します。

161ページの『リモート管理のためにチャンネルおよび伝送キューを作成する』は、リモート管理をセットアップするためにこれらの定義をどのように使用するかを示しています。

分散キューイングのセットアップに関する詳細については、*MQSeries 相互通信*を参照してください。

クラスターを使用するリモート管理

分散キューイングを使用する従来の *MQSeries* ネットワークでは、すべてのキュー・マネージャーは独立しています。1つのキュー・マネージャーが別のキュー・マネージャーにメッセージを送信する必要がある場合、伝送キュー、リモート・キュー・マネージャーへのチャンネル、メッセージの送信先であるすべてのキューのリモート・キュー定義を定義しておく必要があります。

クラスターは、複雑な伝送キュー、チャンネル、およびキューを定義しなくても、複数のキュー・マネージャーが単一のネットワーク上で互いに直接通信できる方法でセットアップされたキュー・マネージャーのグループです。クラスターは、簡単にセットアップでき、通常は、一部論理的に関連付けられるキュー・マネージャーを含み、データまたはアプリケーションを共用する必要があります。

クラスターを作成すると、クラスター内のキュー・マネージャーは、複雑なチャンネルまたはリモート・キューを定義しなくても互いに通信することができます。最小クラスターでも、システム管理のオーバーヘッドが短縮します。

クラスター内のキュー・マネージャーのネットワークを確立する場合、従来の分散キューイング環境を確立するのに比べて、定義が少なくてすみます。作成する定義が少ないので、ネットワークを素早く、簡単にセットアップまたは変更することができます。定義にエラーが発生するリスクが軽減されます。

クラスターをセットアップするには、通常、キュー・マネージャーにつき1つのクラスター送信側 (CLUSSDR) 定義と1つのクラスター受信側 (CLUSRCVR) 定義が必要です。伝送キュー定義またはリモート・キュー定義は必要ありません。リモート管理の基本は、クラスター内で使用されるときは同じですが、定義自体は大幅に単純化されます。

クラスター、その属性およびセットアップ方法の詳細については、*MQSeries* キュー・マネージャー・クラスターを参照してください。

リモート・キュー・マネージャーの管理

この節では、ローカル・キュー・マネージャーからリモート・キュー・マネージャーを管理する方法について説明します。次のものを使用して、ローカル・ノードからリモート管理を実現できます。

- MQSC コマンド
- PCF コマンド

キューおよびチャネルの作成方法は、基本的にはどちらの方法についても同じです。本書では、理解しやすいように各例で MQSC コマンドを示しています。ただし、必要に応じて、それらの例を PCF に変換することができます。PCF を使用した管理プログラムの作成方法に関する詳細については、*MQSeries プログラム式システム管理* を参照してください。

リモート管理では、対話式で、あるいはコマンドを含んでいるテキスト・ファイルから、MQSC コマンドをリモート・キュー・マネージャーに送ります。リモート・キュー・マネージャーは、同じマシン上に存在する場合がありますが、通常は、異なったマシン上に存在します。キュー・マネージャーは、UNIX[®]、Compaq NSK、AS/400[®]、MVS/ESA、Windows[®] 2000、および OS/390 など、さまざまな MQSeries 環境でリモートから管理することができます。

リモート管理を実施するには、特定のオブジェクトを作成する必要があります。特殊な要件がない限り、デフォルト値（メッセージ長など）で十分です。

リモート管理のためのキュー・マネージャーを作成する

161ページの図37 は、リモート管理に必要なキュー・マネージャーおよびチャネルの構成を示しています。source.queue.manager は、MQSC コマンドを発行できる ソース キュー・マネージャーであり、それらのコマンド（オペレーター・メッセージ）の結果も、可能ならここに戻されます。

target.queue.manager は、コマンドを処理し、オペレーター・メッセージを生成する宛先キュー・マネージャーです。

注: source.queue.manager は、使用しているマシン上のデフォルト・キュー・マネージャーでなければなりません。キュー・マネージャーの作成についての詳細は、342ページの『crtmqm (キュー・マネージャーの作成)』を参照してください。

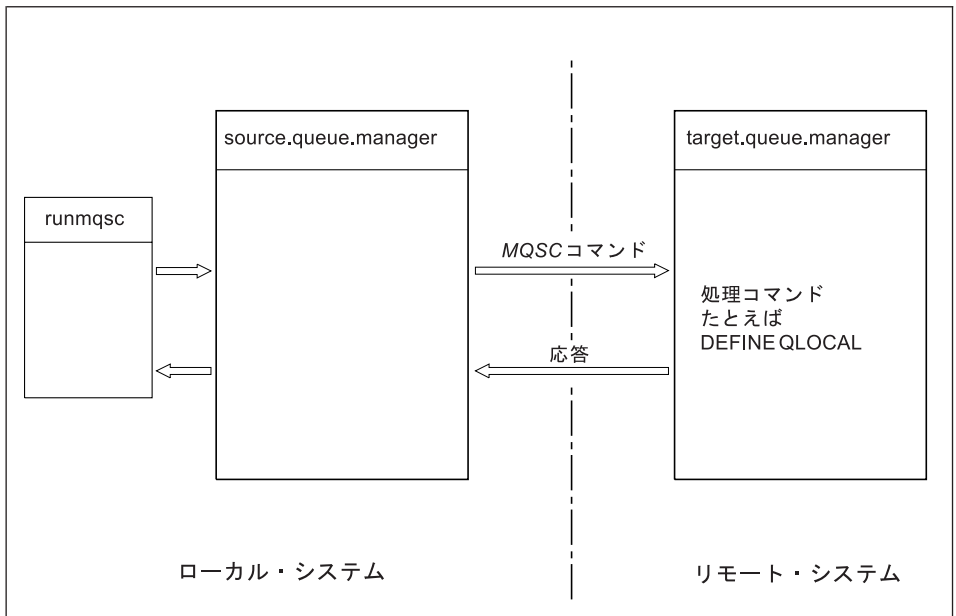


図 37. リモート管理

両方のシステムで、次の事柄を行う必要があります (まだ行っていない場合)。

- **crtmqm** コマンドを使用して、キュー・マネージャーを作成します。
- **strmqm** コマンドを使用して、キュー・マネージャーを開始します。

このステップの詳細については、86ページの『デフォルトおよびシステム・オブジェクトの復元』を参照してください。これらのコマンドは、ローカルで実行するか、Telnet などのネットワーク機能を介して実行する必要があります。

ターゲット・キュー・マネージャーに対して、次のようにします。

- コマンド・キュー `SYSTEM.ADMIN.COMMAND.QUEUE` が存在している。
このキューは、キュー・マネージャーが作成されるときにデフォルトとして作成されます。
- **strmqcsv** コマンドを使用してコマンド・サーバーを開始する。

リモート管理のためにチャンネルおよび伝送キューを作成する

リモートから MQSC コマンドを実行するには、2 つのチャンネル (各方向ごとに 1 つ) およびそれらに関連した伝送キューをセットアップする必要があります。この例では、TCP/IP がトランスポート・タイプとして使用されていること、および TCP/IP アドレスをユーザーが把握していることが前提条件です。

チャンネル `source.to.target` は、MQSC コマンドをソース・キュー・マネージャーから宛先に送るためのものです。送信側は `source.queue.manager` であ

リモート・キュー・マネージャーの管理

り、受信側はキュー・マネージャー `target.queue.manager` です。チャンネル `target.to.source` は、コマンドの出力および生成されたオペレーター・メッセージをソース・キュー・マネージャーに戻すためのものです。各送信側ごとに伝送キューを定義する必要があります。このキューは、受信側のキュー・マネージャーの名前が付けられたローカル・キューです。キュー・マネージャー別名を使用していない限り、リモート管理を行うために、`XMITQ` 名がリモート・キュー・マネージャー名と一致している必要があります。図38にこの構成が示されています。ただし、`SYSTEM.MQSC.REPLY.QUEUE` は、独自の動的応答キューを作成するために `MQSC` によって使用されるモデル・キューの名前である点に注意する必要があります。このキュー名はさまざまで、`MQSC` に対して内部的なものです。

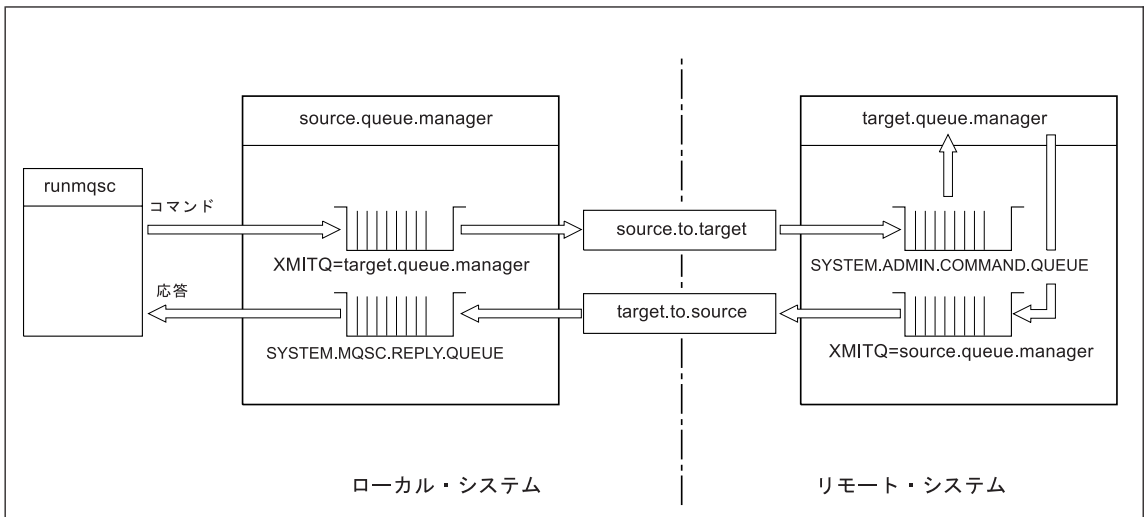


図 38. リモート管理のためのチャンネルとキューのセットアップ

リモート・チャンネルのセットアップに関する詳細については、*MQSeries 相互通信* を参照してください。

チャンネルと伝送キューを定義する

送信元キュー・マネージャーに対して次の `MQSC` コマンドを出して、チャンネルと伝送キューを定義します。

```

* Define the sender channel at the source queue manager
DEFINE CHANNEL ('source.to.target') +
  CHLTYPE(SDR) +
  CONNNAME ('198.210.60.37(1414)') +
  XMITQ ('target.queue.manager') +
  TRPTYPE(TCP)

* Define the receiver channel at the source queue manager

DEFINE CHANNEL ('target.to.source') +
  CHLTYPE(RCVR) +
  TRPTYPE(TCP)

* Define the transmission queue on the source

DEFINE QLOCAL ('target.queue.manager') +
  USAGE (XMITQ)

```

これらのコマンドを宛先キュー・マネージャー (target.queue.manager) に対して発行し、そこにチャンネルと伝送キューを作成します。

```

* Define the sender channel on the destination queue manager

DEFINE CHANNEL ('target.to.source') +
  CHLTYPE(SDR) +
  CONNNAME ('198.210.60.37(1414)') +
  XMITQ ('source.queue.manager') +
  TRPTYPE(TCP)

* Define the receiver channel on the destination queue manager

DEFINE CHANNEL ('source.to.target') +
  CHLTYPE(RCVR) +
  TRPTYPE(TCP)

* Define the transmission queue on the destination queue manager

DEFINE QLOCAL ('source.queue.manager') +
  USAGE (XMITQ)

```

注: 送信側チャンネル定義の CONNNAME 属性に指定された TCP/IP 接続名は、説明のためだけに示したものです。これは、接続の他方側のマシンの IP アドレスまたはネットワーク名です。各自ネットワークに合った値を使用してください。

チャンネルを開始する

次の説明は、両側のチャンネルが MQSeries for Compaq NSK 上で実行されていることを想定しています。このようになっていない場合は、Compaq NSK ではない側のチャンネルの該当する資料を参照してください。

2 つのチャンネルを開始するには、両方のノードで Compaq NSK TCP リスナー・プロセスが MQSeries 用に構成されていて、接続の両側で実行されていることをまず確認してください。その後、**runmqsc** でチャンネルを開始します。

- 送信元キュー・マネージャーに対して、次のように入力します。

```
start channel ('source.to.target')
```

- 宛先キュー・マネージャーに対して、次のように入力します。

```
start channel ('target.to.source')
```

チャンネルの自動定義

チャンネルの自動定義は、MQSeries バージョン 5.1 以降のプロダクトにおいてターゲット・宛先キュー・マネージャーが実行している場合にだけ適用します。インバウンド接続要求を受信して、チャンネル定義ファイル (CDF) に適切な受信側またはサーバー接続定義がない場合、MQSeries は自動的に定義を作成し、その定義を CDF に追加します。自動定義は MQSeries で提供される 2 つのデフォルト定義 SYSTEM.AUTO.RECEIVER および SYSTEM.AUTO.SVRCONN に基づいています。

MQSC コマンド、ALTER QMGR (またはキュー・マネージャーの変更 PCF コマンド) を使用してキュー・マネージャー・オブジェクトを更新することによって、受信側の自動定義およびサーバー接続定義を使用可能にします。

チャンネル定義の自動作成に関する詳細については、*MQSeries 相互通信* を参照してください。

クラスターのためのチャンネルの自動定義に関する詳細については、*MQSeries キュー・マネージャー・クラスター* を参照してください。

リモートから MQSC コマンドを発行する

MQSC コマンドをリモートから処理する場合には、コマンド・サーバーが宛先キュー・マネージャー上で実行されている必要があります。(ソース・キュー・マネージャーで実行されている必要はありません)。

- 宛先キュー・マネージャーに対して、次のように入力します。

```
strmqcsv target.queue.manager
```

- 次に、送信元キュー・マネージャーに対して次のように入力することにより、キュー・モードで対話的に MQSC を実行できます。

```
runmqsc -w 30 target.queue.manager
```

この形式の **runmqsc** コマンド (-w フラグ付き) は、キュー・モードで MQSC コマンドを実行します。このモードでは、コマンドがコマンド・サーバーの入力キューに (修正された形式で) 書き込まれ、順次に行われます。

MQSC コマンドを入力すると、このコマンドはリモート・キュー・マネージャー (ここでは、target.queue.manager) にリダイレクトされます。タイムアウト

リモート・キュー・マネージャーの管理

トは 30 秒に設定されます。したがって、30 秒以内に応答がなければ、ローカル (送信元) キュー・マネージャーで次のメッセージが生成されます。

```
AMQ8416: MQSC timed out waiting for a response from the command server.
```

MQSC セッションの終了時に、ローカル・キュー・マネージャーは、到着したタイムアウト応答を表示します。MQSC セッションが終了すると、それ以降の応答は破棄されます。

キュー・モードでは、MQSC コマンドもリモート・キュー・マネージャー上で実行することができます。たとえば、次のようにします。

```
runmqsc /IN mycmds, OUT report/ -w 60 target.queue.manager
```

ここで、mycmds は MQSC コマンドを含んでいるファイルであり、report はレポート・ファイルです。

MVS/ESA でのキュー・マネージャーの取り扱い

MQSeries for Compaq NSK キュー・マネージャーから、MVS/ESA キュー・マネージャーに対して MQSC コマンドを発行することができます。ただし、これを行うには、送信側の **runmqsc** コマンドとチャネル定義を修正する必要があります。

特に、Compaq NSK ノード側で **runmqsc** コマンドに **-x** フラグを追加します。

```
runmqsc -w 30 -x QMRI
```

チャネル定義は、次のようになります。

```
* Define the sender channel at the source
queue manager on Compaq NSK
```

```
DEFINE CHANNEL ('source.to.target') +
  CHLTYPE(SDR) +
  CONNAME ('198.210.60.37(1414)') +
  XMITQ (QMRI) +
  TRPTYPE(TCP) +
```

前と同じように、送信元キュー・マネージャーに受信側チャンネルと伝送キューを定義する必要もあります。この例でも、TCP/IP が伝送プロトコルとして使用されていることを想定しています。

リモート・キューイングについての勧告

リモート・キューイングを実施する際に、次のようにしてください。

1. リモート・システムで実行する MQSC コマンドは、コマンド・ファイルに入れます。
2. **runmqsc** コマンドに **-v** フラグを指定することにより、MQSC コマンドをローカルで確認します。
runmqsc を使用して別のキュー・マネージャー上の MQSC コマンドを確認することはできません。
3. コマンド・ファイルがローカルでエラーなしで実行されることを可能な限り確認します。
4. 最後に、そのコマンド・ファイルをリモート・システムに対して実行します。

リモートからの MQSC の使用に問題がある場合

リモートから MQSC コマンドを実行する際に問題が生じた場合、次のリストに従ってチェックしてください。

- 宛先キュー・マネージャーのコマンド・サーバーを開始しましたか。
- 有効な伝送キューを定義しましたか。
- 次の両方について、メッセージ・チャンネルの両端を定義しましたか。
 - コマンドが送信されるチャンネル
 - 応答が戻されるチャンネル
- チャンネル定義に正しい結合名 (CONNAME) を指定しましたか。
- メッセージ・チャンネルを開始する前に、リスナーを開始しましたか。

リモートからの MQSC の使用の問題

- 切断時間の間隔が期限切れでないかどうか (チャンネルが開始したが、しばらくしてシャットダウンされた場合など) について検査しましたか。これは、チャンネルを手動操作で開始した場合に特に重要です。
- 意味をなさない要求 (新しいパラメーターを含んだ要求など) を発信元キュー・マネージャーからターゲット・キュー・マネージャーに送信しないようにしてください。

129ページの『MQSC で起こった問題の解決』も参照してください。

リモート・キューのローカル定義の作成

リモート・キュー定義は、リモート・キューのローカル定義として使用できません。別のキュー・マネージャー上のローカル・キューを識別するために、自分のローカル・キュー・マネージャーにリモート・キュー・オブジェクトを作成します。

リモート・キューのローカル定義の働きについて理解する

アプリケーションは、ローカル・キュー・マネージャーに接続し、その後 MQOPEN 呼び出しを出します。オープン呼び出しで指定されるキュー名は、ローカル・キュー・マネージャー上のリモート・キュー定義のキュー名です。リモート・キュー定義は、宛先キューの名前、宛先キュー・マネージャーの名前、および伝送キューの名前 (オプション) を提供します。リモート・キューにメッセージを書き込むためには、アプリケーションは、MQOPEN 呼び出しから戻されたハンドルを指定して、MQPUT 呼び出しを出します。キュー・マネージャーは、リモート・キュー名およびリモート・キュー・マネージャー名を、メッセージの伝送ヘッダーに追加します。この情報は、ネットワーク内の正しい宛先にメッセージを転送するために使用されます。

管理者は、リモート・キュー定義を更新することにより、メッセージの宛先を制御できます。

例

アプリケーションは、リモート・キュー・マネージャーが所有しているキューにメッセージを書き込む必要があります。

処理内容

アプリケーションはキュー・マネージャー (`saturn.queue.manager`など) に接続します。宛先キューは、別のキュー・マネージャーが所有しています。

MQOPEN 呼び出しで、アプリケーションは MQOD に次のフィールドを指定します。

フィールド値	説明
<i>ObjectName</i> CYAN.REMOTE.QUEUE	リモート・キュー・オブジェクトのローカル名を指定します。これは、宛先キューと宛先キュー・マネージャーを定義します。
<i>ObjectType</i> (キュー)	このオブジェクトをキューとして識別します。
<i>ObjectQmgrName</i> ブランク または saturn.queue.manager	このフィールドの指定はオプションです。 ブランクの場合、ローカル・キュー・マネージャーの名前と見なされます。(これは、リモート・キュー定義が作成され、アプリケーションが接続されているキュー・マネージャーです。) ブランクでない場合、ローカル・キュー・マネージャーの名前を指定する必要があります。

この後、アプリケーションはこのキューにメッセージを書き込むために、MQPUT 呼び出しを出します。

ローカル・キュー・マネージャーでは、次の MQSC コマンドを使用してリモート・キューのローカル定義を作成することができます。

```
DEFINE QREMOTE ('CYAN.REMOTE.QUEUE') +
DESCR ('Queue for auto insurance requests from the branches') +
RNAME ('AUTOMOBILE.INSURANCE.QUOTE.QUEUE') +
RQMNAME ('jupiter.queue.manager') +
XMITQ ('INQUOTE.XMIT.QUEUE')
```

説明

QREMOTE ('CYAN.REMOTE.QUEUE')

リモート・キュー・オブジェクトのローカル名です。これは、このキュー・マネージャーに接続されたアプリケーションが、リモート・キュー・マネージャー `jupiter.queue.manager` 上のキュー `AUTOMOBILE.INSURANCE.QUOTE.QUEUE` をオープンするために、MQOPEN 呼び出しに指定する必要がある名前です。

DESCR ('Queue for auto insurance requests from the branches')

キューの用途を説明する追加テキストです。

リモート・キューのローカル定義の作成

RNAME ('AUTOMOBILE.INSURANCE.QUOTE.QUEUE')

リモート・キュー・マネージャーの宛先キューの名前です。これは、アプリケーションが送信する、キュー名 'CYAN.REMOTE.QUEUE' を指定したメッセージの実際の宛先キューです。キュー 'AUTOMOBILE.INSURANCE.QUOTE.QUEUE' を、ローカル・キューとしてリモート・キュー・マネージャーに定義する必要があります。

RQMNAME ('jupiter.queue.manager')

宛先キュー 'AUTOMOBILE.INSURANCE.QUOTE.QUEUE' を所有するリモート・キュー・マネージャーの名前です。

XMITQ ('INQUOTE.XMIT.QUEUE')

伝送キューの名前です。この指定はオプションです。この名前を指定しないと、リモート・キュー・マネージャーと同じ名前のキューが使用されます。

いずれの場合でも、伝送キューであることを指定する *Usage* 属性 (MQSC に USAGE(XMITQ) を指定) を持つローカル・キューとして、該当の伝送キューを定義する必要があります。

リモート・キューにメッセージを書き込む代替方法

リモート・キューのローカル定義を使用する方法以外にも、リモート・キューにメッセージを書き込む方法があります。アプリケーションは、リモート・キュー・マネージャー名を含んでいる完全なキュー名を、MQOPEN 呼び出しの一部として指定することができます。この場合、リモート・キューのローカル定義は不要です。ただし、この代替方法の場合、アプリケーションがリモート・キュー・マネージャーの名前を認識しているか、実行時にリモート・キュー・マネージャーの名前にアクセスできなければなりません。

リモート・キューに関してその他のコマンドを使用する

該当の MQSC コマンドを使用すると、リモート・キュー・オブジェクトの属性を表示または変更したり、リモート・キュー・オブジェクトを削除したりすることができます。たとえば、次のようにします。

```

* Display the remote queue's attributes.
* ALL = Display all attributes

DISPLAY QUEUE (CYAN.REMOTE.QUEUE) ALL

* ALTER the remote queue to enable puts.
* This does not affect the destination queue,
* only applications that specify this remote queue.

ALTER QREMOTE (CYAN.REMOTE.QUEUE) PUT(ENABLED)

* Delete this remote queue
* This does not affect the destination queue
* only its local definition

DELETE QREMOTE (CYAN.REMOTE.QUEUE)

```

注: リモート・キューを削除する場合、削除するのはリモート・キューのローカル表示のみです。リモート・キューやリモート・キュー上のメッセージは削除されません。

伝送キューを作成する

伝送キューとは、キュー・マネージャーがメッセージ・チャンネルを介してメッセージをリモート・キュー・マネージャーに転送する際に使用されるローカル・キューのことです。チャンネルは、リモート・キュー・マネージャーへの片方向リンクを提供します。メッセージは、チャンネルがメッセージを受け入れることができるまで、伝送キューにキューイングされます。チャンネルを定義する際には、メッセージ・チャンネルの送信側に伝送キュー名を指定してください。

Usage 属性 (MQSC 内の USAGE) は、キューが伝送キューであるか、通常キューであるかを定義します。

デフォルト伝送キュー

*XmitQName*属性 (MQSC 内の XMITQ) を任意で使用して、伝送キューをリモート・キュー・オブジェクトに指定できます。伝送キューを定義しなければ、デフォルト値が使用されます。アプリケーションがメッセージをリモート・キューに書き込む際に、宛先キュー・マネージャーと同じ名前を持つ伝送キューが存在している場合、そのキューが使用されます。このキューが存在していなければ、ローカル・キュー・マネージャーの *DefaultXmitQ* 属性 (MQSC 内の DEFXMITQ) によって指定されたキューが使用されます。

伝送キューの作成

たとえば、次の MQSC コマンドでは、`target.queue.manager` に送られるメッセージ用として、デフォルト伝送キューが `source.queue.manager` に作成されます。

```
DEFINE QLOCAL ('target.queue.manager') +
DESCR ('Default transmission queue for target qm') +
USAGE (XMITQ)
```

アプリケーションは、伝送キューに直接メッセージを書き込んだり、間接的に（たとえば、リモート・キュー定義を介して）書き込んだりすることができます。

168ページの『リモート・キューのローカル定義の作成』も参照してください。

リモート・キュー定義を別名として使用する

キューを別のキュー・マネージャーに置くだけでなく、リモート・キューのローカル定義を次の 2 つの作業に使用することもできます。

- キュー・マネージャー別名付け
- 応答先キュー別名付け

いずれのタイプの別名も、リモート・キューのローカル定義を使用して解決されます。

リモート・キューイングで通常行われているように、メッセージがその宛先に到着するためには、該当のチャンネルをセットアップする必要があります。

キュー・マネージャー別名

別名付けとは、宛先キュー・マネージャーの名前（メッセージ内に指定されている）をメッセージ経路上のキュー・マネージャーによって変更するためのプロセスです。キュー・マネージャー別名は重要です。キュー・マネージャーのネットワーク内でメッセージの宛先を制御するのに、この別名を使用できるためです。

これを行うには、制御点でキュー・マネージャーのリモート・キュー定義を変更します。送信アプリケーションは、指定されたキュー・マネージャー名が別名であることを認識しません。

キュー・マネージャー別名に関する詳細については、*MQSeries 相互通信* を参照してください。

応答先キュー別名付け

オプションとして、アプリケーションは、要求メッセージをキューに入れる際に、応答先キューの名前を指定することができます。メッセージを処理するアプリケーションは、その応答先キューの名前を取り出すときに、必要に応じて応答メッセージの送り先を確認します。

応答先キュー別名付けとは、応答先キューの名前（要求メッセージ内で指定されている）をメッセージ経路上のキュー・マネージャーによって変更するためのプロセスです。送信アプリケーションは、指定された応答先キュー名が別名であることを認識しません。

応答先キュー別名を使用すると、応答先キューの名前を更新でき、オプションでそのキュー・マネージャーを更新することもできます。これによって、応答メッセージに使用される経路を制御することができます。

要求メッセージ、応答メッセージおよび応答先キューの詳細については、*MQSeries アプリケーション・プログラミング・リファレンス* を参照してください。応答先キュー別名に関する詳細については、*MQSeries 相互通信* を参照してください。

データ変換

MQSeries で定義された形式（組み込み形式とも呼ばれる）のメッセージ・データは、キュー・マネージャーによって 1 つのコード化文字セットからもう 1 つのコード化文字セットに変換することができます。ただし、2 つのコード化文字セットが、1 つの言語または類似する言語グループに関連付けられていることが必要です。

たとえば、ID (CCSID) がそれぞれに 850 と 500 であるコード化文字セット間の変換は、両方とも西欧の言語に該当するため、サポートされます。

ASCIIへの EBCDIC 改行 (NL) 文字変換については、244ページの『MQSeries 構成ファイルの内容』を参照してください。

サポートされている変換は、*MQSeries アプリケーション・プログラミング・リファレンス* の付録 F「コード・ページ変換表」で定義されています。

キュー・マネージャーがメッセージを組み込み形式に変換できない場合

CCSID が別の各国語グループを表している場合には、キュー・マネージャーはメッセージを組み込み形式に自動的に変換することはできません。たとえば、CCSID 850 と CCSID 1025 (キリル文字スクリプトを使用する言語用の

EBCDIC コード化文字セット) 間の変換はサポートされていません。これは、一方のコード化文字セットの文字の多くが、もう一方のコード化文字セットで表現できないためです。さまざまな各国語で稼働しているキュー・マネージャーのネットワークがあり、一部のコード化文字セット間でのデータ変換がサポートされていない場合に、デフォルト変換を使用することができます。デフォルトのデータ変換については、『デフォルトのデータ変換』で説明しています。

ファイル CCSID

ファイル CCSID は、その他のコード・セットおよびデフォルトのデータ変換があれば、それらを指定します。CCSID ファイルに記録されている情報は更新可能です。たとえば、使用しているオペレーティング・システムの将来のリリースで追加のコード化文字セットがサポートされる場合に、更新が必要となる場合があります。追加のコード・セットを指定するには、この CCSID ファイルを編集する必要があります。そのための方法については、ファイル内に記されています。

デフォルトのデータ変換

デフォルトのデータ変換を実施するには、CCSID ファイルを編集して、デフォルト EBCDIC CCSID とデフォルト ASCII CCSID を指定し、さらにデフォルト CCSID も指定します。この方法に関する指示は、このファイルに入っています。

CCSID を更新してデフォルト・データ変換を実施する場合は、変更内容が有効になる前にキュー・マネージャーを再始動する必要があります。

デフォルトのデータ変換プロセスは、次のようになります。

- ソース CCSID とターゲット CCSID の間の変換がサポートされていなくても、CCSID のソース環境およびターゲット環境の両方が EBCDIC または ASCII のいずれかである場合は、文字データは変換されずにターゲット・アプリケーションに渡されます。
- 一方の CCSID が ASCII コード化文字セットを表し、もう一方の CCSID が EBCDIC コード化文字セットを表す場合、MQSeries は CCSID で定義されているデフォルトのデータ変換機構 CCSIDS を使用してデータを変換します。

注: メッセージ用として指定されたコード化文字セットとデフォルトのコード化文字セット中で同じコード値を持つ文字に、変換対象文字を制限してください。MQSeriesオブジェクト名にとって有効な文字のセットのみを使用する場合には、通常、この要件を満たします。日本で使用されている

EBCDIC CCSID 290、930、1279、および 5026 では例外が発生します。この場合、小文字は他の EBCDIC CCSID で使用されるものとは異なるコードを持ちます。

ユーザー定義形式でのメッセージの変換

ユーザー定義形式のメッセージを、キュー・マネージャーによって 1 つのコード化文字セットから別のコード化文字セットに変換することはできません。ユーザー定義形式のデータが変換を必要とする場合は、それぞれの形式ごとにデータ変換出口が必要となります。ユーザー定義形式の文字データの変換のために、デフォルト CCSID を使用することはできますが、お勧めできません。ユーザー定義形式のデータ変換およびデータ変換出口の作成に関する詳細については、*MQSeries アプリケーション・プログラミング・ガイド* を参照してください。

キュー・マネージャー CCSID の変更

キュー・マネージャーの CCSID の変更時には、ALTER QMGR コマンドの CCSID 属性を使用して、キュー・マネージャーを停止して再始動することをお勧めします。

これによって、コマンド・サーバーとチャネル・プログラムを含むすべての実行中のアプリケーションが、停止して再始動するようになります。

キュー・マネージャー CCSID の変更時に実行中であったアプリケーションは、既存の CCSID を使用し続けるので、このことが必要です。

第8章 MQSeries オブジェクトの保護

この章では、MQSeries for Compaq NSK でのセキュリティー管理について、またセキュリティー管理をインプリメントする方法について説明します。

この章には、次の節があります。

- 178ページの『MQM ユーザー・グループのユーザー ID を理解する』
- 『MQSeries リソースを保護する理由』
- 179ページの『オブジェクト権限マネージャー (OAM) を理解する』
- 184ページの『オブジェクト権限マネージャー (OAM) コマンドを使用する』
- 187ページの『アクセス許可』
- 188ページの『権限の表示コマンド』
- 188ページの『オブジェクト権限マネージャー (OAM) のガイドライン』
- 193ページの『許可指定テーブルについての理解』
- 201ページの『許可ファイルの理解』

MQSeries リソースを保護する理由

MQSeries キュー・マネージャーは潜在的価値のある情報の転送を取り扱うため、権限システムによる安全対策が必要になります。これによって、キュー・マネージャーが所有して取り扱うリソースが無許可アクセスから保護されることになり、データの消失または開示のおそれなくなります。システム・セキュリティーでは、無許可ユーザーまたは無許可アプリケーションが次のものをアクセスしたり、変更できないようにすることが重要です。

- キュー・マネージャーとの接続。
- キュー、クラスター、チャネルおよびプロセスなどの MQSeries オブジェクトへのアクセス。
- MQSCs コマンドおよび PCF コマンドを含むキュー・マネージャー管理用のコマンド。
- MQSeries メッセージへのアクセス。
- メッセージに関連したコンテキスト情報。

どのユーザーにどのリソースをアクセスさせるかについて、独自の方針を確立してください。

MQM ユーザー・グループのユーザー ID を理解する

すべてのキュー・マネージャー・リソースは、次のグループ ID MQM を使用して実行されます。

MQSeries for Compaq NSK へアクセスできるようにするため、それぞれのユーザー ID が MQSeries プリンシパルに対応していなければなりません。初めは、キュー・マネージャーを作成したユーザー ID だけが、MQSeries プリンシパル `mqm` を持っています。MQSeries へアクセスするユーザーごとにプリンシパルを作成するには、**altmqusr** コマンドを使用します。プリンシパルとユーザーは、どちらも固有でなければなりません。キュー・マネージャーのプリンシパルとそのプロパティを表示するには、**dspmqusr** コマンドを使用します。

ユーザー ID が MQSeries for Compaq NSK グループ MQM に属していて、そのユーザー ID 向けに MQSeries プリンシパルが作成されている場合、すべての MQSeries リソースに対するすべての権限を持っています。すべての MQSeries for Compaq NSK 制御コマンド (**crtmqcvx** を除く) を使用できるようにするには、ユーザー ID が MQM グループに属していなければなりません。特に、次の権限が必要です。

- **runmqsc** ユーティリティを使用して MQSC コマンドを実行する。
- **setmqaut** コマンドを使用して、MQSeries for Compaq NSK に関する権限を管理する。

Compaq NSK リモート・システムのキュー・マネージャーにチャンネル・コマンドを送信する場合は、ユーザー ID はターゲット・システムの Compaq NSK グループ MQM のメンバーである必要があります。PCF および MQSC チャンネル・コマンドについては、192ページの『チャンネル・コマンド・セキュリティ』のリストを参照してください。

次のものを発行する場合、ユーザー ID がグループ MQM に属していなくても構いません。

- 管理プログラムから PCF コマンド (エスケープ PCF を含む) を発行する場合
- アプリケーション・プログラムから MQI 呼び出しを発行する場合。

注: `mqm` プリンシパルは、保護されたリソースへアクセスする目的で内部キュー・マネージャー・コンポーネントによって使用されるため、`mqm` プリンシパル (および MQM グループ) を許可することは重要です。キュー・マネージャー内のオブジェクトから `mqm` プリンシパルまたは MQM の許可

を削除する、あるいは mqm プリンシパル自体を削除すると、管理できないキュー・マネージャーで終了してしまうか、最悪の場合にはまったく使用できなくなる可能性があります。

その他の情報を表示する

次のものの詳細については、示されている資料を参照してください。

- MQSeries for Compaq NSK コマンド・セットについては、37ページの『第3章 MQSeries コマンド・セットの使用』を参照してください。
- MQSeries for Compaq NSK 制御コマンドについては、323ページの『第17章 MQSeries 制御コマンド』を参照してください。
- PCF コマンドおよびエスケープ PCF については、*MQSeries プログラム式システム管理* を参照してください。
- MQI 呼び出しについては、*MQSeries アプリケーション・プログラミング・ガイド* および *MQSeries アプリケーション・プログラミング・リファレンス* を参照してください。

オブジェクト権限マネージャー (OAM) を理解する

デフォルトでは、キュー・マネージャー・リソースへのアクセスは、許可サービスのインストール可能なコンポーネントを介して管理されます。MQSeries for Compaq NSK に付属する権限サービス・コンポーネントは、OAM と呼ばれており、特別の指定をしないう限り、自動的にインストールされ、作成されるキュー・マネージャーごと使用可能になります。この章で使用される OAM とは、このプロダクトと共に提供されるオブジェクト権限マネージャーのことです。

OAM は、許可サービスのインストール可能なコンポーネントです。OAM がインストール可能なコンポーネントとして提供されるため、次のような柔軟な対応が可能です。

- 提供されているインターフェースを使用して、この OAM を独自の許可サービス・コンポーネントで置き換える。
- 提供されているインターフェースを再度使用して、独自の許可サービス・コンポーネントにより、この OAM の機能を強化する。
- OAM を除去するか、使用不可にして、許可サービスなしで実行する。

インストール可能サービスの詳細については、*MQSeries プログラム式システム管理* を参照してください。

オブジェクト権限マネージャー

OAM は、キュー、プロセス定義、およびチャネルなどの MQSeries オブジェクトを操作するためのユーザーの許可を管理します。また、OAM は、特定のオブジェクトへのアクセス権限を特定のグループのユーザーに与えたり、取り消したりするためのコマンド・インターフェースを提供します。あるリソースへのアクセスを認める決定は OAM が行い、キュー・マネージャーはその決定に従います。OAM が決定できない場合は、キュー・マネージャーは該当のリソースへのアクセスを妨げます。

OAM の機能

OAM は、ユーザーおよびグループ ID と、Compaq NSK オペレーティング・システムのセキュリティー機能を使用します。ユーザーは、必要な権限を持っている場合にだけ、キュー・マネージャー・オブジェクトにアクセスできません。

ユーザー・グループを介してアクセスを管理する

MQSeries リソースへのアクセス許可の管理は、Compaq NSK グループに基づいています。OAM は、グループ・レベルで許可を保守します。

コマンド・インターフェースでは、ユーザー ID よりも MQSeries プリンシパルが使用されます。その理由は、ユーザー ID に与えられる権限は、別のエンティティーにも与えられるためです。たとえば、権限は MQI 呼び出しを発行するアプリケーション・プログラムや PCF コマンドを発行する管理プログラムにも与えられます。これらの場合、プログラムに関連付けられたプリンシパルは、プログラムの開始時に使用されたユーザー ID とは限りません。

Compaq NSK ユーザー ID は <group>.<name> の形式とすることができます。ここで group も name もそれぞれ 8 文字まで可能ですが、MQSeries プリンシパル名は 12 文字までです。さらに、他のプラットフォームのユーザー ID では、ピリオド (.) は使用できません。MQSeries for Compaq NSK では、プリンシパル・データベースには、12 文字以下の MQSeries プリンシパル名に対する Compaq NSK ユーザー ID の割り当てが収められています。

ユーザーが複数のユーザー・グループに属している場合

ユーザーが持っている許可は、ユーザーが属する全グループの許可と、全ユーザーのデフォルトの許可とを兼ね備えたものです。制御コマンド **setmqaut** を使用すると、特定のグループの許可を設定できます。

注: **setmqaut** コマンドを使用して行われたすべての変更の影響は、オブジェクトが使用中でない限り即時に現れます。オブジェクトが使用中の場合は、オブジェクトが次にオープンされると変更が行われます。

グループ・セットと 1 次グループ

MQSeries リソースに対するアクセス許可の管理は、Compaq NSK ユーザー・グループに基づいています。SAFEGUARD を実行している場合、Compaq NSK ユーザー ID を複数のグループに関連付けられるため、対応する MQSeries プリンシパルもこれらのグループに関連付けられます。1 次グループは、常に Compaq 管理グループになります。2 次グループは、SAFEGUARD ファイル共有グループを作成することにより構成され、Compaq NSK ユーザー ID が該当するファイル共有グループに関連付けられます。

OAM は、個々のプリンシパルではなくグループのレベルで許可を保守します。グループ名に対するプリンシパルの割り当ては、プリンシパル・データベースと Compaq NSK および SAFEGUARD 機能を使用して OAM 内で実行されます。OAM 操作は、グループ・レベルで実行されます。ただし、個々のプリンシパルの許可を表示することはできません。

OAM でリソースを保護する

OAM を使用することによって次の事柄が制御できます。

- MQI を介する MQSeries オブジェクトへのアクセス。アプリケーション・プログラムがオブジェクトにアクセスしようとする時、OAM は、要求された操作に関する (そのユーザー・グループを介した) 許可を要求元のユーザー ID が持っているかどうかを調べます。

特に、これはキューおよびキュー上のメッセージを無許可アクセスから保護することを意味します。

- MQSC コマンドを使用する許可。ユーザー・グループ `mqm` のメンバーか、`setmqaut` を介して許可されたメンバーだけが、たとえば、キューを作成するために、キュー・マネージャー管理コマンドを実行できます。
- 制御コマンドを使用する許可。ユーザー・グループ `mqm` のメンバーだけが、たとえば、キュー・マネージャーの作成や、コマンド・サーバーの開始などの制御コマンドを実行できます。
- PCF コマンドを使用する許可。

同じオブジェクトに対して、ユーザー・グループごとに異なる種類のアクセス権限を与えることができます。たとえば、特定のキューに対して、あるグループには書き込み操作と読み取り操作の両方を許可し、別のグループにはキューのブラウズ (ブラウズ・オプションによる MQGET) のみを許可することができます。また、あるグループには、あるキューの読み取りおよび書き込みは権限があるが、そのキューの変更または削除は認めないということもできます。

許可にグループを使用する

個別のプリンシパルでなく、グループを使用することにより、必要な管理タスクの量を減らすことができます。特定の種類のアクセスを複数のプリンシパルが必要とするのが普通であるためです。たとえば、特定のアプリケーションの実行を希望するエンド・ユーザーからなるグループを定義できます。新しいユーザーの場合は、適切なグループをそのユーザーの Compaq NSK ユーザー ID に追加してアクセスを許可できます。SAFEGUARD を使用して MQSeries をシステムヘインストールしてデータ共有グループを作成しない限り、各ユーザー ID は、1 つの 1 次グループだけに関連付けることができます。

SAFEGUARD を使用しない場合、グループあたり 255 のプリンシパルに制限されます。SAFEGUARD ファイル共有を使用すると、グループあたり 65535 のプリンシパルの制限になります。また、SAFEGUARD ファイル共有を使用することにより、プリンシパルが独自の許可を持てるようにもなります。たとえば、SAFEGUARD 管理グループに許可が与えられていない場合、そのグループに属する各ファイル共有メンバーは独自の権限を持つことができます (プリンシパル A でグループ MQSEC のメンバーである TESTSEC1 は PUT だけ、プリンシパル B でグループ MQSEC のメンバーである TESTSEC2 は GET だけを持っています)。

SAFEGUARD を使用しない場合、グループの許可はプリンシパル (Compaq NSK ユーザー ID MQM.MANAGER グループ MQM (connect) プリンシパル mqm および Compaq NSK ユーザー ID MQM.APPS グループ MQM プリンシパル apps (get/put)) で結合されてしまうので、2 つのグループ (別々のグループにある 2 つの Compaq NSK ユーザー ID (たとえば MQ.TEST (connect/put) と MQ1.TEST (connect/get))) を作成し、それぞれが独自の許可を持てるようにする必要があります。グループの権限セット (mqm の場合は connect) とプリンシパルの権限セット (apps の場合は get/put) は、相互に追加されます (connect/get/put)。その後、結合したデータに対してユーザー権限の妥当性検査が行われます。ユーザー ID MQ.TEST を使用してログオンしているときに、アプリケーションが MQGET を実行しようとする、その操作は拒否されて MQRC_NOT_AUTHORIZED (2035) が示されます。MQ1.TEST としてのログオンはパスされます。Compaq ユーザー ID で MQM グループにログオンしたユーザーがいれば、パスされます。プリンシパルに権限が与えられれば、そのグループ内の全プリンシパルに同じ権限が与えられます。

Principal	Compaq Userid	Group	GroupType	Security
mqm	MQM.MANAGER	MQM		connect
apps	MQM.APPS	MQM		get/put
inq	MQM.INQ	MQM		inq
mq	MQ.TEST	MQ		connect/put
mq1	MQ1.TEST	MQ1		connect/get
A	MQSEC.FRED	MQSEC	safeguard admin	none
		TESTSEC1	safeguard file sharing	connect/put
B	MQSEC.JOE	MQSEC	safeguard admin	none
		TESTSEC2	safeguard file sharing	connect/get
C	MQSEC.FRANK	MQSEC	safeguard admin	none
		TESTSEC3	safeguard file sharing	none

図 39. SAFEGUARD を設定したグループを使用して許可を与える

図39 では、グループ MQM にログオンしたユーザーがいれば、connect、get、put および inq 権限を持っています。ユーザー MQ.TEST は connect と put で、MQ1.TEST は connect と get です。MQSEC グループのユーザーには、safeguard ファイル共有メンバーの権限が付与されています。

グループの数はできるだけ少なくします。たとえば、ユーザーを 1 グループのアプリケーション・ユーザーと 1 グループのアドミニストレーターとに分けることができます。

オブジェクト権限マネージャー (OAM) を使用不可にする

デフォルトでは、OAM は使用可能になっています。キュー・マネージャーを作成する前に、次のように Compaq NSK 環境変数 MQSNOAUT を設定すると、OAM を使用不可にすることができます。

```
PARAM MQSNOAUT 1
```

ただし、キュー・マネージャーの OAM を使用不可にする場合、後で OAM を再始動することはできません。OAM を使用可能にして、すべてのユーザーおよびアプリケーションが適切なユーザー ID を介してアクセスできるようにしておくことができます。また、キュー・マネージャー構成ファイル (QMINI) の許可サービス・スタンザを除去するか、245ページの『キュー・マネージャー構成ファイル (QMINI)』で説明されているように、QMINI の Authority スタンザにある MQAUTH をオフにするかして、テスト目的だけに OAM を使用不可にすることができます。

注: PARAM MQSNOAUT 0 を指定しても、OAM は使用可能になりません。OAM を再び使用可能にする予定であれば、その環境に環境変数がある必要はありません。

オブジェクト権限マネージャー (OAM) コマンドを使用する

OAM は、権限を与えたり、取り消したりするためのコマンド・インターフェースを提供しています。これらのコマンドを使用するには、そのための許可が必要です。つまり、ユーザー ID は Compaq NSK MQM グループに属している必要があります。(MQSeries for Compaq NSK をインストールする前に、このグループを設定しておく必要があります。)

ユーザー ID がグループ MQM のメンバーである場合は、キュー・マネージャーに対する「スーパーユーザー」権限があります。ユーザー ID からすべての MQI 要求または制御コマンドを発行する許可が与えられています。

OAM は、ユーザーの許可を管理するために TACL から起動できる 4 つのコマンドを提供しています。そのコマンドは次のとおりです。

- **altmqusr** (MQSeries プリンシパルの作成、削除、または更新)
- **dspmqsar** (プリンシパルの表示)
- **setmqaut** (権限の設定またはリセット)
- **dspmqaat** (権限の表示)

権限検査は、MQCONN、MQOPEN、MQPUT1、および MQCLOSE の呼び出しの際に発生します。権限検査は、これらすべての呼び出しの最初のインスタンスだけで実行され、リセットする (つまり、クローズして再オープンする) まで権限は修正されません。したがって、**setmqaut** を使用してオブジェクトの権限に加えられた変更は、いずれもオブジェクトをリセットするまでは有効になりません。

OAM コマンドの使用時に指定するもの

OAM コマンドは指定したキュー・マネージャーに適用されます。キュー・マネージャーを指定しないと、デフォルト・キュー・マネージャーが使用されます。これらのコマンドには、オブジェクトを固有に指定する必要があります。つまり、オブジェクト名とそのタイプを指定する必要があります。また、権限を適用するユーザーまたはグループ名も指定する必要があります。

許可リスト

setmqaut コマンドを使用して、許可のリストを指定します。この方法では、許可の付与または取り消しの指定や、許可が適用されるリソースの指定が簡単に行えます。リストの各許可は、小文字のキーワードの前に正符号 (+) または負符号 (-) を付けて指定します。指定した許可を追加するときには正符号を使用し、許可を取り消すときには負符号を使用します。単一のコマンドで、許可をいくつでも指定できます。たとえば、次のようにします。


```
+browse -get +put
```

altmqusr コマンドを使用する

必要な許可を持っている場合は、**altmqusr** コマンドを使用すると、MQSeries プリンシパルを作成し、Compaq NSK ユーザー ID (または SAFEGUARD 別名) に関連付けることができます。次の例は、**altmqusr** コマンドの使い方を示しています。

```
altmqusr -m saturn.queue.manager -p MQPRINCIPAL -u MQM.MQUSER
```

この例のそれぞれの指定の意味は次のとおりです。

語句...	指定内容...
saturn.queue.manager	キュー・マネージャー名
MQPRINCIPAL	作成するプリンシパル名
MQM.MQUSER	Compaq NSK ユーザー ID

このコマンドの説明については、333ページの『altmqusr (MQSeries ユーザー情報の更新)』を参照してください。

altmqusr コマンドを使用しても、プリンシパルを削除できます (したがって、MQSeries へのすべてのアクセス権限を取り消せます)。たとえば、次のようにします。

```
altmqusr -m saturn.queue.manager -p MQPRINCIPAL -remove
```

この例のそれぞれの指定の意味は次のとおりです。

語句...	指定内容...
saturn.queue.manager	キュー・マネージャー名
MQPRINCIPAL	削除するプリンシパル名
-remove	プリンシパルを削除するための命令

OAM コマンドの使用

dspmqsqr コマンドを使用する

dspmqsqr コマンドを使用して、各 MQSeries プリンシパルに対応する該当ユーザー ID が属する Compaq NSK 管理およびファイル共有グループだけでなく、プリンシパル・データベースの内容も表示できます。-p パラメーターを指定すると、表示される情報が指定したプリンシパルに限定されます。たとえば、次のようにします。

```
dspmqsqr -m saturn.queue.manager -p MQPRINCIPAL
```

この例のそれぞれの指定の意味は次のとおりです。

語句...	指定内容...
saturn.queue.manager	キュー・マネージャー名
MQPRINCIPAL	表示するプリンシパル名

このコマンドの説明については、361ページの『dspmqsqr (MQSeries ユーザー情報の表示)』を参照してください。

setmqaut コマンドを使用する

必要な許可を持っている場合は、**setmqaut** コマンドを使用すると、特定のオブジェクトをアクセスする許可をプリンシパルまたはユーザー・グループに与えたり、取り消したりできます。次の例は、**setmqaut** コマンドの使い方を示しています。

```
setmqaut -m saturn.queue.manager -t queue -n RED.LOCAL.QUEUE -g GroupA +browse -get +put
```

この例のそれぞれの指定の意味は次のとおりです。

語句....	指定内容....
saturn.queue.manager	キュー・マネージャー名
キュー	オブジェクト・タイプ
RED.LOCAL.QUEUE	オブジェクト名
GroupA	許可を与えるグループの ID
+browse -get +put	指定したキューの許可リスト。「+」または「-」符号とキーワードの間にはスペースを入れません。

許可リストは、付与される許可を指定します。次のとおりです。

語句...	指定内容...
+browse	ブラウズする (ブラウズ・オプションを指定した MQGET) 許可を追加します。
-get	キューからメッセージを読み取る (MQGET) 許可を取り消します。
+put	キューにメッセージを書き込む (MQPUT) 許可を追加します。

Compaq NSK ユーザー・グループ GroupA に属するユーザー ID で開始したアプリケーションには、これらの許可があります。

次のコマンドはキュー MyQueue に対する書き込み権限を、グループ GroupA および GroupB に対して取り消します。

```
setmqaut -m saturn.queue.manager -t queue -n MyQueue -g GroupA -g GroupB -put
```

このコマンドの形式的定義と構文については、382ページの『setmqaut (権限の設定 / リセット)』を参照してください。

権限コマンドとインストール可能なサービス

setmqaut コマンドは、更新が適用される許可サービス・コンポーネントの名前を指定する追加パラメーターを取ります。同時に実行される複数の許可コンポーネントがある場合は、このパラメーターを指定する必要があります。デフォルトでは、このパラメーターは使用されません。このパラメーターを省略した場合は、最初に見つかった許可コンポーネント (存在する場合) が更新されます。デフォルトでは、これはシステムに提供された OAM です。

アクセス許可

setmqaut コマンドに関連した許可リストで定義される許可は、次のように類別できます。

- MQI 呼び出しに関する許可
- 管理コマンドに関する許可
- コンテキスト許可
- 一般許可、すなわち、MQI 呼び出しまたはコマンドに関するもの、あるいはその両方に関するもの

アクセス許可

各許可は、`setmqaut` および `dspmqaut` コマンドで使用するキーワードで指定します。これらのコマンドについては、382ページの『`setmqaut` (権限の設定 / リセット)』で説明しています。

権限の表示コマンド

コマンド `dspmqaut` を使用して、特定のオブジェクトに関して特定のプリンシパルまたはグループが持つ許可を表示することができます。フラグの意味は、`setmqaut` コマンドの場合と同じです。一度に 1 つのグループまたはプリンシパルの許可のみが表示されます。このコマンドの形式的な指定については、349ページの『`dspmqaut` (権限の表示)』を参照してください。

たとえば、次のコマンドは、キュー・マネージャー `QueueMan1` にある `Annuities` という名前のプロセス定義について `GpAdmin` が持っている許可を表示しています。

```
dspmqaut -m QueueMan1 -t process -n Annuities -g GpAdmin
```

このコマンドの結果として表示されるキーワードは、アクティブな許可を示しています。

オブジェクト権限マネージャー (OAM) のガイドライン

一部の操作は特に重要度が高いので、その実行は特権ユーザーに限る必要があります。たとえば、次のようにします。

- キュー・マネージャーの作成、削除、開始および停止。
- 伝送キューまたはコマンド・キュー `SYSTEM.ADMIN.COMMAND.QUEUE` などの特殊キューへのアクセス。
- 完全な MQI コンテキスト・オプションを使用するプログラム。
- アプリケーション・キューの作成とコピー。

ユーザー ID

作成した `MQM` という特別なグループは、プロダクトのアドミニストレーターだけが使用するよう意図されています。特権のないユーザーは使用できません。

キュー・マネージャー・ボリューム

キューおよびその他のキュー・マネージャー・データを入れるボリュームは、プロダクト専用です。このディレクトリーの中のオブジェクトには、OAM 許可に関連する Compaq NSK ユーザー許可があります。Compaq NSK の標準コマンドは、MQI リソースへの許可を与えたり、取り消したりするために使用しないでください。理由は次のとおりです。

- MQSeries オブジェクトは、対応するシステム・オブジェクトと必ずしも同じではないからです。この点に関する詳細については、79ページの『ボリューム構造』を参照してください。
- MQSeries オブジェクトは、必ずしもオブジェクトの NSK セキュリティー設定に割り当てられているわけではないためです。

キュー

動的キューに対してアクセスする権限は、それが派生したモデル・キューに対する権限に基づきます (ただし、必ずしも同じではありません)。

別名キューまたはリモート・キューの場合、許可はオブジェクト自体に関するものであり、別名キューまたはリモート・キューが解決されるキューの許可ではありません。したがって、プリンシパルに別名キューへのアクセスを許可し、その解決先のローカル・キューへのアクセスは認めないという場合もあります。

キューを作成する権限は、特権ユーザーに限定してください。権限を限定しないと、ユーザーが別名を作成して通常のアクセス管理を逃れる事態になりかねません。

代替ユーザー権限

代替ユーザー権限は、1 つのユーザー ID が MQSeries オブジェクトにアクセスしている間に、別のユーザー ID の権限を使用できるかどうかを制御するものです。この方法は、サーバーがプログラムから要求を受け取り、その要求に必要な権限をプログラムに与えなければならない場合に重要です。サーバーは、要求に必要な権限があっても、要求したアクションに関する権限がプログラムにあるかどうかを確認する必要があります。

たとえば、次のようにします。

- ユーザー ID PAYSERV のもとで実行中のサーバーが、キューから要求メッセージを取り出したとします。この要求メッセージは、ユーザー ID USER1 によってキューに置かれます。

OAM のガイドライン

- サーバー・プログラムは、要求メッセージを読み取ると、要求を処理し、要求メッセージで指定されている応答先キューに応答を書き戻します。
- サーバーは、サーバーのユーザー ID (PAYSERV) を使用して応答先キューのオープンを許可する代わりに、別のユーザー ID (この場合は USER1) を指定することができます。この例では、PAYSERV が応答先キューをオープンするときに代替ユーザー ID として USER1 を指定できるかどうかを制御するために、代替ユーザー権限を使用することができます。

代替ユーザー ID は、オブジェクト記述子の *AlternateUserId* フィールドに指定します。

ユーザー ID と代替ユーザー ID はどちらも、許可を与えるために、Compaq NSK ユーザー ID と関連付けられたプリンシパル・データベースのエントリーに対応するプリンシパルとして指定する必要があります。

注: 代替ユーザー ID は、どの MQSeries オブジェクトでも使用できます。代替ユーザー ID を使用しても、別のリソース・マネージャーが使用するユーザー ID には影響しません。

コンテキスト権限

コンテキストは、特定のメッセージに適用される情報であって、メッセージの一部であるメッセージ記述子 MQMD に含まれています。コンテキスト情報は 2 つのセクションからなります。

ID セクション

この部分は、メッセージの発信者を示します。次のフィールドからなります。

- *UserIdentifier*
- *AccountingToken*
- *ApplIdentityData*

起点セクション

このセクションは、メッセージの発信元のアプリケーションと、キューに書き込まれた日時を示します。次のフィールドからなります。

- *PutApplType*
- *PutApplName*
- *PutDate*
- *PutTime*
- *ApplOriginData*

アプリケーションは、MQOPEN 呼び出しまたは MQPUT 呼び出しのいずれかを出すときにコンテキスト・データを指定することができます。このデータ

は、アプリケーションによって生成されたり、別のメッセージから渡されたり、デフォルトでキュー・マネージャーによって生成されたりします。たとえば、コンテキスト・データはサーバー・プログラムによって、要求側の ID の検査、メッセージの発信元が許可ユーザー ID のもとで実行中のアプリケーションであるかどうかのテストに使用されることがあります。

サーバー・プログラムは、*UserIdentifier* を使用して、代替ユーザーのユーザー ID を判別することができます。*UserIdentifier* は、プリンシパル・データベースのエントリーに対応するプリンシパルとして指定する必要があります。

コンテキスト許可は、ユーザーが MQOPEN 呼び出しまたは MQPUT1 呼び出しにコンテキスト・オプションを使用できるかどうかを制御するのに使用できます。コンテキスト・オプションの詳細については、*MQSeries アプリケーション・プログラミング・ガイド* を参照してください。コンテキストに関連するメッセージ記述子フィールドの説明については、*MQSeries アプリケーション・プログラミング・リファレンス* を参照してください。

リモート・セキュリティーに関する考慮事項

リモート・セキュリティーについては、次のことを考慮する必要があります。

書き込み権限

複数のキュー・マネージャーにまたがるセキュリティーについては、チャネルが別のキュー・マネージャーから送られたメッセージを受け取ったときに使用する書き込み権限を指定することができます。

チャネル属性 PUTAUT は次のように指定します。

DEF デフォルト・ユーザー ID。メッセージ・チャネル・エージェントが実行されるユーザー ID です。

CTX メッセージ・コンテキスト内のユーザー ID。

どちらの場合も、ユーザー ID は、プリンシパル・データベースのエントリーに対応するプリンシパルとして指定する必要があります。

伝送キュー

キュー・マネージャーは、伝送キューにリモート・メッセージを自動的に書き込みます。特殊権限は必要ありません。しかし、メッセージを伝送キューに直接書き込むには、特殊な許可が必要です。194ページの表 3 を参照してください。

チャネル出口

チャネル出口は、追加されたセキュリティーに使用されます。

OAM のガイドライン

詳細については、*MQSeries 相互通信* を参照してください。

チャンネル・コマンド・セキュリティー

チャンネル・コマンドは、MQAI、MQSC コマンドおよび制御コマンドにより、PCF コマンドとして発行することができます。

PCF コマンド

リモート Compaq NSK システムの SYSTEM.ADMIN.COMMAND.QUEUE に PCF メッセージを送ることによって、PCF チャンネル・コマンドを発行できます。PCF メッセージのメッセージ記述子に指定するユーザー ID は、ターゲット・システムの mqm グループに属する Compaq NSK ユーザー ID と関連付けられたプリンシパル・データベースのエントリーに対応するプリンシパルとして指定する必要があります。これらのコマンドは、次のとおりです。

- ChangeChannel
- CopyChannel
- CreateChannel
- DeleteChannel
- PingChannel
- ResetChannel
- StartChannel
- StopChannel
- ResolveChannel

PCF セキュリティー要件の詳細については、*MQSeries プログラム式システム管理* を参照してください。

MQSC チャンネル・コマンド

MQSC チャンネル・コマンドをリモート Compaq NSK システムに出すには、PCF エスケープ・メッセージでコマンドを直接送信するか、**runmqsc**を使用して間接モードでコマンドを発行するか、の 2 つの方法があります。PCF メッセージのメッセージ記述子に指定するユーザー ID は、ターゲット・システムの mqm グループに属する Compaq NSK ユーザー ID と関連付けられたプリンシパル・データベースのエントリーに対応するプリンシパルとして指定する必要があります。(PCF コマンドは、間接モードで **runmqsc** から出された MQSC コマンドに暗黙に含まれています。) これらのコマンドは、次のとおりです。

- ALTER CHANNEL
- DEFINE CHANNEL
- DELETE CHANNEL
- PING CHANNEL
- RESET CHANNEL

- START CHANNEL
- START CHINIT
- STOP CHANNEL
- RESOLVE CHANNEL

runmqsc コマンドから発行された MQSC コマンドの場合、PCF メッセージ内のユーザー ID は通常、現行ユーザーの ID です。

許可指定テーブルについての理解

194ページ以降の許可指定表には、許可の機能と、適用される制限が正確に定義されています。これらの表は、次のような状態に適用されます。

- MQI 呼び出しを発行するアプリケーション。
- MQSC コマンドをエスケープ PCF として発行する管理プログラム。
- PCF コマンドを発行する管理プログラム。

この節では、次のものを指定する 1 組のテーブルとして情報を示します。

実行するアクション	MQI オプション、MQSC コマンド、または PCF コマンド
アクセス制御オブジェクト	キュー、プロセス、またはキュー・マネージャ
必要な許可	「MQZAO_」定数で表す

テーブルの中で、接頭部が MQZAO_ の定数は、特定のエンティティーに関する **setmqaut** コマンドの許可リストのキーワードに対応します。たとえば、MQZAO_BROWSE はキーワード **+browse** に対応します。また、キーワード MQZAO_SET_ALL_CONTEXT はキーワード **+setall** に対応します。これらの定数は、プロダクトと共に提供されるサブボリューム ZMQSLIB 内のヘッダー・ファイル CMQZCH に定義されています。

MQI 許可

MQI 呼び出しおよびオプションのいくつかは、アプリケーションを実行するユーザー ID (またはアプリケーションが許可を想定できるユーザー ID) が適切な許可を与えられている場合にのみ、アプリケーションから発行できます。

許可検査を必要とする MQI 呼び出しは、MQCONN、MQOPEN、MQPUT1、MQCLOSE の 4 つです。

MQOPEN および MQPUT1 の場合、許可検査は、名前が解決された結果の 1 つまたは複数の名前についてではなく、オープンされるオブジェクトの名前について行われます。たとえば、アプリケーションが別名キューをオープンする

許可指定テーブル

権限を与えられていても、別名が解決される基本キューをオープンする権限は与えられていない場合があります。検査の規則は次のとおりです。ネーム・レゾリューションの過程で最初に検出された定義について検査が行われます。この定義は、キュー・マネージャー別名定義が直接オープンされる場合以外はキュー・マネージャー別名ではない定義です。つまり、オブジェクト記述子の *ObjectName* フィールドに現れた名前について検査が行われます。特定のオブジェクトをオープンするには必ず権限が必要です。さらに、キューに依存しない追加の権限 (キュー・マネージャー・オブジェクトに関する許可を介して取得する) が必要なこともあります。

表3 は、各呼び出しに必要な許可を要約したものです。

表3. MQI 呼び出しに必要なセキュリティー許可

必要な許可	キュー・オブジェクト	プロセス・オブジェクト	キュー・マネージャー	名前リスト
MQCONN オプション	適用しない	適用しない	MQZAO_CONNECT	適用しない
MQOPEN オプション				
MQOO_INQUIRE	MQZAO_INQUIRE (2)	MQZAO_INQUIRE (2)	MQZAO_INQUIRE (2)	MQZAO_INQUIRE (2)
MQOO_BROWSE	MQOO_BROWSE	適用しない	検査しない	適用しない
MQOO_INPUT_*	MQZAO_INPUT	適用しない	検査しない	適用しない
MQOO_SAVE_ALL_CONTEXT (3)	MQZAO_INPUT	適用しない	検査しない	適用しない
MQOO_OUTPUT (通常キュー) (4)	MQOO_OUTPUT	適用しない	検査しない	適用しない
MQOO_PASS_IDENTITY_CONTEXT (5)	MQZAO_PASS_IDENTITY_CONTEXT	適用しない	検査しない	適用しない
MQOO_PASS_ALL_CONTEXT (5、6)	MQZAO_PASS_ALL_CONTEXT	適用しない	検査しない	適用しない
MQOO_SET_IDENTITY_CONTEXT (5、6)	MQZAO_SET_IDENTITY_CONTEXT	適用しない	MQZAO_SET_IDENTITY_CONTEXT	適用しない
MQOO_SET_ALL_CONTEXT (5、8)	MQZAO_SET_ALL_CONTEXT	適用しない	MQZAO_SET_ALL_CONTEXT (7)	適用しない

表 3. MQI 呼び出しに必要なセキュリティ許可 (続き)

必要な許可	キュー・ オブジェクト	プロセス・ オブジェクト	キュー・ マネージャー	名前リスト
MQOO_OUTPUT (伝送キュー) (9)	MQZAO_SET_ ALL_CONTEXT	適用しない	MQZAO_SET_ ALL_CONTEXT (7)	適用しない
MQOO_SET	MQZAO_SET	適用しない	検査しない	適用しない
MQOO_ ALTERNATE _USER _AUTHORITY	(10)	(10)	MQZAO_ ALTERNATE_ USER_ AUTHORITY (10、 11)	適用しない
MQPUT1 オプション				
MQPMO_PASS_ IDENTITY_ CONTEXT	MQZAO_PASS_ IDENTITY_ CONTEXT (12)	適用しない	検査しない	適用しない
MQPMO_PASS_ ALL_CONTEXT	MQZAO_PASS_ ALL_CONTEXT (12)	適用しない	検査しない	適用しない
MQPMO_SET_ IDENTITY_ CONTEXT	MQZAO_SET_ IDENTITY_ CONTEXT (12)	適用しない	MQZAO_SET_ IDENTITY_ CONTEXT (7)	適用しない
MQPMO_SET_ ALL_CONTEXT	MQZAO_SET_ ALL_CONTEXT (12)	適用しない	MQZAO_SET_ ALL_CONTEXT (7)	適用しない
(伝送キュー) (9)	MQZAO_SET_ ALL_CONTEXT	適用しない	MQZAO_SET_ ALL_CONTEXT (7)	適用しない
MQPMO_ ALTERNATE _ USER_ AUTHORITY	(13)	適用しない	MQZAO _ALTERNATE_ USER_ AUTHORITY (11)	適用しない
MQCLOSE オプション				
MQCO_DELETE	MQZAO_DELETE (14)	適用しない	適用しない	適用しない
MQCO_DELETE_ PURGE	MQZAO_DELETE (14)	適用しない	適用しない	適用しない

特別な注:

1. モデル・キューがオープンされる場合は、次のようになります。

許可指定テーブル

- モデル・キューの場合、指定したオープン・オプションに必要な他のすべての権限に加えて (モデル・キューのための権限も)、MQZAO_DISPLAY 権限が必要です。
 - 動的キューを作成する場合、MQZAO_CREATE 権限は必要ありません。
 - モデル・キューのオープンに使用したユーザー ID には、作成された動的キューに関するキュー特有のあらゆる権限が自動的に与えられます (MQZAO_ALL と同等)。
2. オープンされるオブジェクトのタイプに応じて、キュー、プロセス、名前リスト、またはキュー・マネージャー・オブジェクトのいずれかが検査されます。
 3. MQOO_INPUT_* も指定する必要があります。これは、ローカル・キュー、モデル・キュー、または別名キューの場合に有効です。
 4. この検査は、注9 に示した場合以外は、すべての場合の出力において実行されます。
 5. MQOO_OUTPUT も指定する必要があります。
 6. このオプションは、MQOO_PASS_IDENTITY_CONTEXT も暗黙的に指定されます。
 7. この権限は、キュー・マネージャー・オブジェクトと個々のキューの両方に対して必要です。
 8. MQOO_PASS_IDENTITY_CONTEXT、MQOO_PASS_ALL_CONTEXT、および MQOO_SET_IDENTITY_CONTEXT も、このオプションによって暗黙的に指定されます。
 9. この検査は、Usage キュー属性として MQUS_TRANSMISSION を持ち、出力のために直接オープンされているローカル・キューまたはモデル・キューについて実行されます。リモート・キューがオープンされる場合 (リモート・キュー・マネージャーとリモート・キューの名前を指定するか、リモート・キューのローカル定義の名前を指定して) は、この検査は適用されません。
 10. MQOO_INQUIRE (あらゆるオブジェクト・タイプの場合)、または (キューの場合) MQOO_BROWSE、MQOO_INPUT_*、MQOO_OUTPUT、または MQOO_SET の中から、少なくとも 1 つを指定する必要があります。検査は他の指定されたオプションの場合と同じで、提供されている代替ユーザー ID を使用し、特有の名前のあるオブジェクト権限と、MQZAO_ALTERNATE_USER_IDENTIFIER 検査の現行アプリケーション権限を調べます。
 11. この許可では、任意の *AlternateUserId* を指定できます。

12. MQUS_TRANSMISSION の Usage キュー属性がないキューの場合は、MQZAO_OUTPUT 検査も行われます。
13. 検査は他の指定されたオプションの場合と同じで、提供されている代替ユーザー ID を使用し、特有の名前のあるキュー権限と、MQZAO_ALTERNATE_USER_IDENTIFIER 検査の現行アプリケーション権限を調べます。
14. この検査は、次の 2 つの条件が両方とも満たされた場合にのみ行われます。
 - 永続動的キューがクローズされて削除中である。
 - 使用中のオブジェクト・ハンドルを戻した MQOPEN が作成したキューではない。
 上記以外の場合は、検査は行われません。

一般的な注:

1. 特殊許可 MQZAO_ALL_MQI には、オブジェクト・タイプに関係する次の許可がすべて含まれます。
 - MQZAO_CONNECT
 - MQZAO_INQUIRE
 - MQZAO_SET
 - MQZAO_BROWSE
 - MQZAO_INPUT
 - MQZAO_OUTPUT
 - MQZAO_PASS_IDENTITY_CONTEXT
 - MQZAO_PASS_ALL_CONTEXT
 - MQZAO_SET_IDENTITY_CONTEXT
 - MQZAO_SET_ALL_CONTEXT
 - MQZAO_ALTERNATE_USER_AUTHORITY
2. MQZAO_DELETE (注14 を参照) および MQZAO_DISPLAY は、管理許可として分類されます。したがって、MQZAO_ALL_MQI には含まれません。
3. 「検査しない」は、許可検査が行われないことを意味します。
4. 「適用しない」は、その検査がこの操作には該当しないことを意味します。たとえば、プロセス・オブジェクトに MQPUT 呼び出しを発行できません。

管理許可

これらの許可を持つユーザーは、管理コマンドを発行できます。これは、エスケープ PCF メッセージとしての MQSC コマンドの場合と、PCF コマンドそ

許可指定テーブル

れ自体の場合があります。こうした方法を使用して、プログラムは、管理コマンドを管理ユーザーに代わって実行させるためにメッセージとしてキュー・マネージャーに送ることができます。

エスケープ PCF 中の MQSC コマンドに関する許可

表4 は、エスケープ PCF に含まれる各 MQSC コマンドに必要な許可を要約したものです。

表4. MQSC コマンドと必要なセキュリティ許可

(2) 必要な許可	キュー・オブジェクト	プロセス・オブジェクト	キュー・マネージャー・オブジェクト	名前リスト
ALTER オブジェクト	MQZAO_CHANGE	MQZAO_CHANGE	MQZAO_CHANGE	MQZAO_CHANGE
CLEAR QLOCAL	MQZAO_CLEAR	適用しない	適用しない	適用しない
DEFINE オブジェクト NOREPLACE (3)	MQZAO_CREATE (4)	MQZAO_CREATE (4)	適用しない	MQZAO_CREATE (4)
DEFINE オブジェクト REPLACE (3, 5)	MQZAO_CHANGE	MQZAO_CHANGE	適用しない	MQZAO_CHANGE
DELETE オブジェクト	MQZAO_DELETE	MQZAO_DELETE	適用しない	MQZAO_DELETE
DISPLAY オブジェクト	MQZAO_DISPLAY	MQZAO_DISPLAY	MQZAO_DISPLAY	MQZAO_DISPLAY

特別な注:

1. コマンドを実行依頼するプログラム (たとえば、`runmqsc`) を実行させるユーザー ID は、キュー・マネージャーに関する `MQZAO_CONNECT` 権限も持っている必要があります。
2. オブジェクトのタイプに応じて、キュー、プロセス、名前リスト、またはキュー・マネージャー・オブジェクトのいずれかが検査されます。
3. DEFINE コマンドでは、LIKE オブジェクトが指定されている場合は LIKE オブジェクトに関する、また LIKE が省略されている場合は適切な `SYSTEM.DEFAULT.xxx` オブジェクトに関する、`MQZAO_DISPLAY` 権限も必要です。
4. `MQZAO_CREATE` 権限は、特定のオブジェクトまたはオブジェクト・タイプに特有のものではありません。 `SETMQAUT` コマンドで `QMGR` のオブ

ジェクト・タイプを指定すれば、指定したキュー・マネージャーのすべてのオブジェクトについて作成権限が与えられます。

- これは、置き換えられるオブジェクトが実際に存在している場合に適用されます。存在していない場合は、DEFINE オブジェクト NOREPLACE の検査になります。

一般的な注:

- PCF コマンドを実行するためには、キュー・マネージャーに DISPLAY 権限を持っている必要があります。
- エスケープ PCF を実行する権限は、エスケープ PCF メッセージのテキスト内の MQSC コマンドに応じて決まります。
- 「適用しない」は、その検査がこの操作には該当しないことを意味します。たとえば、キュー・マネージャー・オブジェクトに CLEAR QLOCAL を発行することはできません。

PCF コマンドについての許可

表5 は、各 PCF コマンドに必要な許可を要約したものです。

表5. PCF コマンドと必要なセキュリティ許可

(2) 必要な許可	キュー・オブジェクト	プロセス・オブジェクト	キュー・マネージャー・オブジェクト	名前リスト
オブジェクトの変更	MQZAO_CHANGE	MQZAO_CHANGE	MQZAO_CHANGE	MQZAO_CHANGE
キューのクリア	MQZAO_CLEAR	適用しない	適用しない	適用しない
オブジェクトのコピー (置き換えなし) (3)	MQZAO_CREATE (4)	MQZAO_CREATE (4)	適用しない	MQZAO_CREATE (4)
オブジェクトのコピー (置き換えあり) (3、6)	MQZAO_CHANGE	MQZAO_CHANGE	適用しない	MQZAO_CHANGE
オブジェクトの作成 (置き換えなし) (5)	MQZAO_CREATE (4)	MQZAO_CREATE (4)	適用しない	MQZAO_CREATE (4)
オブジェクトの作成 (置き換えあり) (5、6)	MQZAO_CHANGE	MQZAO_CHANGE	適用しない	MQZAO_CHANGE
オブジェクトの削除	MQZAO_DELETE	MQZAO_DELETE	適用しない	MQZAO_DELETE

許可指定テーブル

表 5. PCF コマンドと必要なセキュリティ許可 (続き)

(2) 必要な許可	キュー・オブジェクト	プロセス・オブジェクト	キュー・マネージャー・オブジェクト	名前リスト
オブジェクトの照会	MQZAO_DISPLAY	MQZAO_DISPLAY	MQZAO_DISPLAY	MQZAO_DISPLAY
オブジェクト名の照会	検査しない	検査しない	検査しない	検査しない
キュー統計のリセット	MQZAO_DISPLAY および MQZAO_CHANGE	適用しない	適用しない	適用しない

特別な注:

1. コマンドを実行依頼するプログラムを実行させるユーザー ID は、そのローカル・キュー・マネージャーに接続する許可と、出力のために管理コマンド・キューをオープンする許可も持っている必要があります。
2. オブジェクトのタイプに応じて、キュー、プロセス、またはキュー・マネージャー・オブジェクトのいずれかが検査されます。
3. Copy コマンドでは、From オブジェクトに関する MQZAO_DISPLAY 許可も必要です。
4. MQZAO_CREATE 権限は、特定のオブジェクトまたはオブジェクト・タイプに特有のものではありません。SETMQAUT コマンドで QMGR のオブジェクト・タイプを指定すれば、指定したキュー・マネージャーのすべてのオブジェクトについて作成権限が与えられます。
5. Create コマンドでは、適切な SYSTEM.DEFAULT.* オブジェクトに関する MQZAO_DISPLAY 権限も必要です。
6. これは、置き換えようとするオブジェクトがすでに存在している場合に適用されます。存在しない場合は、Copy または Create (置き換えなし) と同じ検査になります。

一般的な注:

1. PCF コマンドを実行するためには、キュー・マネージャーに DISPLAY 権限を持っている必要があります。
2. 特殊許可 MQZAO_ALL_ADMIN には、オブジェクト・タイプに関係する次の許可がすべて含まれます。
 - MQZAO_CHANGE
 - MQZAO_CLEAR
 - MQZAO_DELETE

- MQZAO_DISPLAY

MQZAO_CREATE は、個々のオブジェクトまたはオブジェクト・タイプ固有の許可ではないため、これには含まれません。

3. 「検査しない」は、許可検査が行われないことを意味します。
4. 「適用しない」は、その検査がこの操作には該当しないことを意味します。たとえば、プロセス・オブジェクトに Clear Queue コマンドは使用できません。

許可ファイルの理解

MQSeries for Compaq NSK V5.1 では、許可情報はすべて、\$VOL.<QMGrSubVol>D: にある以下の TM/MP 監査ファイルに保管されます。

OAMDB	OAM データベース
PRIDB	基本データベース
PRIDBA	プリンシパル・データベース代替キー・ファイル

プリンシパル・データベース

プリンシパル・データベースの各レコードは、Compaq NSK ユーザー ID をプリンシパル名に割り当てます。プリンシパル・データベースは、ENSCRIBE キー・シーケンス・ファイルで、OAM プリンシパルと Compaq NSK ログオン ID を相互に割り当てます。

OAM プリンシパル	MQPRINCIPAL
Compaq ログオン ID	0x2CFF

1 次キーは、OAM プリンシパルです (12 文字)。代替キーは、Compaq ログオン ID です (2 バイトの短整数)。OAM プリンシパルは、常に大文字小文字を区別します。ログオン ID フィールドのバイトは、<group>.<user> です。したがって、上記の例は Compaq NSK ID (44,255) 向けになります。

OAM データベース

新しい OAM 許可データベース内の各レコードは、特定のキュー・マネージャー・オブジェクト、またはオブジェクトのクラスを参照します。1 次キーは、オブジェクト名とオブジェクト・タイプを足したものです。レコードは可変長で、レコード・レイアウトは次のようになります。

オブジェクト名	タイプ	#許可エントリ	許可エントリ
QUEUE.AUTH	1	2	PAYROLL 0x00000004, ADMIN 0xFFFFFFFF

許可ファイル

「オブジェクト名」フィールドには、完全な 48 文字の、ブランクのないオブジェクト名が入ります。「タイプ」フィールド (4 バイト) は、MQSeries オブジェクトのタイプと、OAM で必要なオブジェクトのクラスとを区別します。

「タイプ」フィールドには、次の値が入ります。

- 1 キュー名
- 2 プロセス名
- 4 キュー・マネージャー名
- 128 クラス

「#許可エントリー」フィールド (4 バイト) は、このレコードの「許可エントリー」フィールドに個々の許可数を指定します。各「許可エントリー」は、このオブジェクト向けに、グループ名とそのグループの許可を指定します。

グループ名	PAYROLL
許可	0x00000004

「グループ名」フィールドは、長さが 12 バイトで、ブランクなしの Compaq NSK 管理または SAFEGUARD ファイル共有グループ名 (最初の 12 文字だけ) を含みます。「許可」フィールドは、4 バイト (ULONG) のビット・マスクで、そのグループの許可が含まれています。それぞれのレコードには、250 までの個別のグループおよび許可のペアを含めることができます。

250 よりも多くのグループの許可情報を保持する必要がある場合、同じオブジェクトに複数のレコードを使用します。

権限指定は、次に示す割り当てに基づく個別のビット・パターンユニオンです。

Authorization keyword	Formal name	Hexadecimal Value
connect	MQZAO_CONNECT	0x00000001
browse	MQZAO_BROWSE	0x00000002
get	MQZAO_INPUT	0x00000004
put	MQZAO_OUTPUT	0x00000008
inq	MQZAO_INQUIRE	0x00000010
set	MQZAO_SET	0x00000020
passid	MQZAO_PASS_IDENTITY_CONTEXT	0x00000040
passall	MQZAO_PASS_ALL_CONTEXT	0x00000080
setid	MQZAO_SET_IDENTITY_CONTEXT	0x00000100
setall	MQZAO_SET_ALL_CONTEXT	0x00000200
altusr	MQZAO_ALTERNATE_USER_AUTHORITY	0x00000400
allmqi	MQZAO_ALL_MQI	0x000007FF
crt	MQZAO_CREATE	0x00010000
dlt	MQZAO_DELETE	0x00020000
dsp	MQZAO_DISPLAY	0x00040000
chg	MQZAO_CHANGE	0x00080000
clr	MQZAO_CLEAR	0x00100000
chgaut	MQZAO_AUTHORIZE	0x00800000
alladm	MQZAO_ALL_ADMIN	0x009E0000
none	MQZAO_NONE	0x00000000
all	MQZAO_ALL	0x009E07FF

これらの定義は、ヘッダー・ファイル `cmqzc h` 内で行われます。次の例では、16 進数 `0x40007` に基づいて、`groupB` に許可が与えられています。これは次に対応します。

MQZAO_CONNECT	0x00000001
MQZAO_BROWSE	0x00000002
MQZAO_INPUT	0x00000004
MQZAO_DISPLAY	0x00040000

Authority is:	0x00040007

これらのアクセス権限は、`groupB` のユーザーであれば、次の MQI 呼び出しを発行できることを意味します。

- MQCONN
- MQGET (ブラウズ付き)

また、この許可ファイルに関連付けられたオブジェクトに関する `DISPLAY` 権限を持っていることを意味します。

クラス許可レコード

クラス許可レコードは、クラス全体に関連した許可を保持します。「オブジェクト名」および「タイプ」フィールドの対応は、次のとおりです。

オブジェクト名	タイプ
@QMGRCLASS	0x80
@PROCESSCLASS	0x80
@QUEUECLASS	0x80

「許可」フィールドの項目 MQZAO_CRT は、このクラスにオブジェクトを作成する許可を与えます。これは唯一のクラス権限です。

全クラス許可レコード

全クラス許可レコードは、キュー・マネージャー全体に適用される許可を保持します。「オブジェクト名」および「タイプ」フィールドの対応は、次のとおりです。

オブジェクト名	タイプ
@ALLCLASSES	0x80

次の許可は、キュー・マネージャー全体に適用され、全クラス許可ファイルに保持されます。

項目	与える許可
MQZAO_ALTUSER	MQSeries オブジェクトと対話するときに別のユーザーと一致すると想定します。
MQZAO_SET_ALL_CONTEXT	MQPUT を発行するときにメッセージのコンテキストを設定します。
MQZAO_SET_IDENTITY_CONTEXT	MQPUT を発行するときにメッセージの一致コンテキストを設定します。

第9章 MQSeries 送達不能キュー・ハンドラー

MQSeries for Compaq NSK には、未配布メッセージ・キューとも呼ばれる送達不能キュー (DLQ) が備えられています。これは、宛先キューに送達できないメッセージ用の保留キューです。ネットワーク内のすべてのキュー・マネージャーが、DLQ を持つ必要があります。

キュー・マネージャー、メッセージ・チャンネル・エージェント (MCA)、およびアプリケーションは、メッセージを DLQ に書き込みます。DLQ 上のすべてのメッセージの先頭に、送達不能ヘッダー 構造体 MQDLH を付ける必要があります。キュー・マネージャーまたはメッセージ・チャンネル・エージェントが DLQ に書き込むメッセージには、常にこのヘッダー構造体があります。メッセージを DLQ に書き込むアプリケーションが MQDLH を提供することを推奨します。MQDLH 構造体の *Reason* フィールドは、メッセージが DLQ 上にある理由を識別する理由コードを含みます。

DLQ 上のメッセージを処理するために定期的に行われるルーチンがなければなりません。MQSeries は、送達不能キュー・ハンドラー (DLQ ハンドラー) と呼ばれるデフォルト・ルーチンを提供しています。DLQ ハンドラーは、**runmqdlq** コマンドを使用して呼び出します。

DLQ 上のメッセージを処理する命令は、ユーザー作成ルール・テーブル を介して DLQ ハンドラーに提供されます。つまり、DLQ ハンドラーは DLQ 上のメッセージとルール・テーブルの項目の突き合わせを行います。DLQ メッセージがルール・テーブルの項目と一致すると、DLQ ハンドラーはその項目に関連付けられたアクションを実行します。

この章には次の節があります。

- 『DLQ ハンドラーの呼び出し』
- 206ページの『DLQ ハンドラーのルール・テーブル』
- 215ページの『ルール・テーブルの処理方法』
- 217ページの『DLQ ハンドラーのルール・テーブルの例』

DLQ ハンドラーの呼び出し

DLQ ハンドラーは、**runmqdlq** コマンドを使用して呼び出します。処理したい DLQ の名前および使用したいキュー・マネージャーの名前を指定するには、次の方法があります。

DLQ ハンドラーの呼び出し

- パラメーターを使用してコマンド・プロンプトから指定する。たとえば、次のようにします。

```
runmqdlq /IN qrule/ ABC1.DEAD.LETTER.QUEUE ABC1.QUEUE.MANAGER
```

- ルール・テーブルの中で指定する。たとえば、次のようにします。

```
INPUTQ(ABC1.DEAD.LETTER.QUEUE) INPUTQM(ABC1.QUEUE.MANAGER)
```

上記の例は、キュー・マネージャー `ABC1.QUEUE.MANAGER` が所有する `ABC1.DEAD.LETTER.QUEUE` という DLQ に適用されます。

DLQ またはキュー・マネージャーを上記の例のように指定しなかった場合は、インストール先のデフォルト・キュー・マネージャーと共にそのキュー・マネージャーが所有する DLQ が使用されます。

runmqdlq コマンドは、標準 IN ファイルに収められているルール・テーブルから入力を読み取ります。IN をルール・ファイルへ転送することにより、ルール・テーブルと **runmqdlq** とを関連付けます。

DLQ ハンドラーを実行するためには、DLQ 自体、および DLQ 上のメッセージの転送先となるあらゆるメッセージ・キューの両方へのアクセスが許可されていることが必要です。さらに、DLQ ハンドラーがメッセージ・コンテキスト中のユーザー ID の権限を使用してキューにメッセージを書き込むことが可能になっている場合には、DLQ ハンドラーを実行するユーザーは他のユーザーの ID を借用する権限を持っている必要があります。

runmqdlq コマンドの詳細については、373ページの『**runmqdlq** (送達不能キュー・ハンドラーの実行)』を参照してください。

DLQ ハンドラーのルール・テーブル

DLQ ハンドラーのルール・テーブルは、DLQ に到着したメッセージを DLQ ハンドラーがどのように処理するかを定義するものです。ルール・テーブルには、次の 2 種類の項目があります。

- テーブルの最初の項目は**制御データ**で、この項目はオプションです。
- テーブルの中の他のすべての項目は、DLQ ハンドラーが従う**ルール**です。各ルールは、**パターン** (1 組のメッセージ特性) と**アクション** からなりま

す。メッセージはパターンと突き合わされ、指定のパターンと DLQ 上の一致したメッセージにはアクションが実行されます。ルール・テーブルには、ルールが少なくとも 1 つ必要です。

ルール・テーブルの各項目は、1 つまたは複数のキーワードからなります。

ルール・テーブルに適用できる構文規則の詳細は、213ページの『ルール・テーブルのルール』を参照してください。

制御データ

この節では、DLQ ハンドラーのルール・テーブルの制御データ項目に入れることができるキーワードについて説明します。次の点に注意してください。

- キーワードのデフォルト値 (ある場合) には、下線が引いてあります。
- 縦線 (|) によって、代替の値を区切っています。値のうちの 1 つのみを指定できます。
- キーワードはすべてオプションです。

INPUTQ (*QueueName*|'_')

このキーワードは、ルール・テーブルを適用する DLQ の名前です。処理したい DLQ の名前を指定します。

1. **runmqdlq** コマンドに *QName* パラメーターを指定した場合は、ルール・テーブル内の INPUTQ 値が指定変更されます。
2. **runmqdlq** コマンドに *QName* パラメーターを指定しないで、ルール・テーブルの中に値を指定すると、ルール・テーブルの INPUTQ 値が使用されます。
3. DLQ を指定しなかった場合、またはルール・テーブルの中で INPUTQ(' ') を指定した場合、**runmqdlq** コマンドの *QMgrName* パラメーター、またはルール・テーブルの INPUTQM キーワードで名前を指定したキュー・マネージャーが属する DLQ が処理されます。

INPUTQM (*QueueManagerName*|'_')

このキーワードは、DLQ を所有するキュー・マネージャーの名前です。INPUTQ キーワードで指定した DLQ を所有するキュー・マネージャーの名前を指定します。

1. **runmqdlq** コマンドに *QMgrName* パラメーターを指定した場合は、ルール・テーブル内の INPUTQM 値が指定変更されます。
2. **runmqdlq** コマンドに *QMgrName* パラメーターを指定しない場合、ルール・テーブルの INPUTQM 値が使用されます。

ルール・テーブル

3. キュー・マネージャーを指定しない場合、または INPUTQM(' ') をルール・テーブルの中に指定した場合は、インストール・システムのデフォルト・キュー・マネージャーが使用されます。

RETRYINT (*Interval*|60)

このキーワードは、最初の試みで処理できなかった DLQ 上のメッセージについて、試みの反復が要求されている場合に DLQ ハンドラーが再処理を試みる間隔 (秒数) です。デフォルトでは、再試行間隔は 60 秒です。

WAIT (YES|NO|*nnn*)

このキーワードは、DLQ ハンドラーが処理できるメッセージがこれ以上ないことを DLQ ハンドラーが検出したとき、DLQ にメッセージが新たに到着するのを DLQ ハンドラーが待機するかどうかを指示します。

YES このキーワードでは、DLQ ハンドラーは無限に待機します。

NO このキーワードでは、DLQ ハンドラーは、DLQ が空になるか、あるいは処理できるメッセージがなくなったことを検出すると終了します。

nnn このキーワードでは、キューが空であるか、または DLQ ハンドラーが処理できるメッセージがキューにないことを DLQ ハンドラーが検出したあと、メッセージが新たに到着するのを DLQ ハンドラーが *nnn* 秒間待機するようにします。

使用頻度の高い DLQ については WAIT (YES)、使用頻度の低い DLQ については WAIT (NO) または WAIT (*nnn*) を指定するようにします。

DLQ ハンドラーが終了するようにした場合は、トリガー操作によって DLQ ハンドラーを再び呼び出す必要があります。

図40 の制御データは、ルール・テーブルがキュー・マネージャー QM1 に属する DLQ に適用されることを示しています。1 行目の最後にある正符号 (+) は、制御データが 2 行目の最初のブランクでない文字に続くことを示します。

```
INPUTQ' ' +  
INPUTQM'QM1'
```

図40. 制御データの例

ルール・テーブルに制御データを組み込む代わりに、**runmqdlq** コマンドの入力パラメーターとして DLQ とそのキュー・マネージャーの名前を指定することができます。ルール・テーブルに値を指定した場合や **runmqdlq** コマンドの入力で値を指定した場合は、**runmqdlq** コマンドで指定した値が優先されません。

注: 制御データ項目をルール・テーブルに組み込む場合、その項目は、ルール・テーブルの最初の項目でなければなりません。

ルール (パターンと処理)

図41 は、DLQ ハンドラーのルール・テーブルのルールの一例です。このルールは、MQPUT および MQPUT1 の使用が禁止されたため DLQ に書き込まれた持続メッセージをその宛先キューに送達する試みを 3 回行うように、DLQ ハンドラーに指示しています。

```
PERSIST(MQPER_PERSISTENT) REASON (MQRC_PUT_INHIBITED) +
ACTION (RETRY) RETRY (3)
```

図41. ルールの例. 1 行目の最後にある正符号 (+) は、ルールが 2 行目の最初の空白でない文字に続くことを示します。

ルールに使用できるすべてのキーワードについては、この節の後の方で説明します。次の点に注意してください。

- キーワードのデフォルト値 (ある場合) には、下線が引いてあります。ほとんどのキーワードの場合、デフォルト値は* (アスタリスク) であり、これはどの値でも構いません。
- 縦線 (|) によって、代替の値を区切っています。値のうちの 1 つのみを指定できます。
- ACTION 以外のキーワードはすべてオプションです。

この節では、まず、パターン・マッチング・キーワード (DLQ 上のメッセージを突き合わせるキーワード) について説明し、次に、処理キーワード (一致するメッセージを DLQ ハンドラーが処理する方法を決定するキーワード) について説明します。

パターン・マッチング・キーワード

DLQ 上で突き合わせられたメッセージに対して値を指定するために使用するパターン・マッチング・キーワードについて、これから説明します。パターン・マッチング・キーワードのすべてがオプションです。

APPLIDAT (*ApplIdentityData*| *)

このキーワードは、DLQ 上のメッセージのメッセージ記述子 (MQMD) に指定した *ApplIdentityData* 値です。

APPLNAME (*PutApplName*|*)

このキーワードは、DLQ 上のメッセージのメッセージ記述子 (MQMD) の

ルール・テーブル

PutApplName フィールドに指定した、MQPUT 呼び出しまたは MQPUT1 呼び出しを出したアプリケーションの名前です。

APPLTYPE (*PutApplType**)

このキーワードは、DLQ 上のメッセージのメッセージ記述子 (MQMD) に指定した *PutApplType* 値です。

DESTQ (*QueueName**)

このキーワードは、メッセージの宛先であるメッセージ・キューの名前です。

DESTQM (*QueueManagerName**)

このキーワードは、メッセージの宛先であるメッセージ・キューのキュー・マネージャーの名前です。

FEEDBACK (*Feedback**)

MsgType 値が MQFB_REPORT の場合、キーワード *Feedback* はレポートの性質を記述します。

記号名を使用できます。たとえば、記号名 MQFB_COA を使用して、DLQ 上のメッセージのうち、宛先キューへの到着の確認が必要なものを識別することができます。

FORMAT (*Format**)

このキーワードは、メッセージ・データの形式を記述するためにメッセージの送信側が使用する名前です。

MSGTYPE (*MsgType**)

このキーワードは、DLQ 上のメッセージのメッセージ・タイプです。

記号名を使用できます。たとえば、記号名 MQMT_REQUEST を使用して、DLQ 上のメッセージのうち応答が必要なものを識別することができます。

PERSIST (*Persistence**)

このキーワードは、メッセージの持続性の値です。(メッセージの持続性によって、キュー・マネージャーの再始動後もメッセージが存続するかどうかが決まります。)

記号名を使用できます。たとえば、記号名 MQPER_PERSISTENT を使用して、DLQ 上のメッセージのうち持続するものを識別することができます。

REASON (*ReasonCode**)

このキーワードは、メッセージが DLQ に書き込まれた理由を記述する理由コードです。

記号名を使用できます。たとえば、記号名 MQRC_Q_FULL を使用して、宛先キューが満杯だったために DLQ に置かれたメッセージを識別することができます。

REPLYQ (QueueName|*)

このキーワードは、DLQ 上のメッセージのメッセージ記述子 (MQMD) に指定した応答先キューの名前です。

REPLYQM (QueueManagerName|*)

このキーワードは、DLQ 上のメッセージのメッセージ記述子 (MQMD) に指定した応答先キューのキュー・マネージャーの名前です。

USERID (UserIdentifier|*)

このキーワードは、メッセージ記述子 (MQMD) に指定した DLQ 上のメッセージの発信元のユーザーのユーザー ID です。

アクション・キーワード

一致するメッセージの処理方法の記述に使用するアクション・キーワードについて、次に説明します。

ACTION (DISCARD|IGNORE|RETRY|FWD)

このキーワードは、このルールに定義したパターンに一致する DLQ 上のメッセージに関して実行されるアクションです。

DISCARD このキーワードでは、メッセージは、DLQ から削除されません。

IGNORE このキーワードでは、メッセージは DLQ 上に残されません。

RETRY このキーワードでは、DLQ ハンドラーは、メッセージを宛先キューに入れることをもう一度試みます。

FWD このキーワードでは、FWDQ キーワードに指定したキューへのメッセージの転送を、DLQ ハンドラーに実行させません。

ACTION キーワードを指定する必要があります。アクションを実行するための試行の回数は、RETRY キーワードで制御します。試行相互間の間隔は、制御データの RETRYINT キーワードで制御します。

FWDQ (QueueName|&DESTQ|&REPLYQ)

このキーワードは、ACTION (FWD) を要求したときのメッセージの転送先のメッセージ・キューの名前です。

ルール・テーブル

QueueName

このキーワードは、メッセージ・キューの名前です。 FWDQ(' ') は無効です。

&DESTQ

このキーワードでは、MQDLH 構造体の *DestQName* フィールドからキュー名が入手されるようにします。

&REPLYQ

このキーワードでは、メッセージ記述子 MQMD の *ReplyToQ* フィールドからキュー名が入手されるようにします。

FWDQ(&REPLYQ) を指定したルールが、ブランクの *ReplyToQ* フィールドを持つメッセージと一致したときにエラー・メッセージを避けるために、メッセージ・パターンで REPLYQ(?*) を指定することができます。

FWDQM (QueueManagerName|&DESTQM|&REPLYQM|'')

このキーワードは、メッセージの転送先のキューのキュー・マネージャーを識別します。

QueueManagerName

このキーワードは、ACTION (FWD) を要求したときの、メッセージの転送先のキューのキュー・マネージャーの名前です。

&DESTQM

このキーワードでは、MQDLH 構造体の *DestQMgrName* フィールドからキュー・マネージャー名が入手されるようにします。

&REPLYQM

このキーワードでは、メッセージ記述子 (MQMD) の *ReplyToQMgr* フィールドからキュー・マネージャー名が入手されるようにします。

' ' FWDQM(' ') はデフォルト値で、ローカル・キュー・マネージャーを識別します。

HEADER (YES|NO)

このキーワードは、ACTION (FWD) が要求されたメッセージに MQDLH を残すかどうかを指定します。デフォルトでは、MQDLH はメッセージに残ります。 HEADER キーワードは、FWD 以外の処理に関しては無効です。

PUTAUT (DEF|CTX)

このキーワードは、DLQ ハンドラーがメッセージを書き込む際の権限を定義します。

- DEF** このキーワードでは、メッセージは DLQ ハンドラー自体の権限で書き込まれます。
- CTX** このキーワードでは、メッセージはメッセージ・コンテキストの中のユーザー ID の権限で書き込まれます。PUTAUT (CTX) を指定する場合、他のユーザーの ID を使用することが許可されている必要があります。

RETRY (RetryCount[1])

RETRY は、1 から 999 の範囲の数で、(制御データの RETRYINT キーワードに指定されている間隔で) アクションを試みる回数です。

注: DLQ ハンドラーが特定のルールを実行するために行う試みのカウントは、DLQ ハンドラーの現行インスタンスに特有のものであり、再始動後には持ちこされません。DLQ ハンドラーが再始動すると、あるルールに適用された試行のカウントはゼロにリセットされます。

ルール・テーブルのルール

ルール・テーブルは、その構文、構造、内容に関して次のルールに従う必要があります。

- ルール・テーブルには、ルールが少なくとも 1 つはなければなりません。
- キーワードは、任意の順序で指定して構いません。
- キーワードは、どのルールにも 1 回のみ指定できます。
- キーワードには大文字小文字の区別はありません。
- キーワードとそのパラメーター値は、1 つ以上のブランクまたはコンマによって他のキーワードから分離する必要があります。
- ルールの始めまたは終わり、およびキーワード、句読点、値の間には、ブランクをいくつ入れても構いません。
- 各ルールは、新しい行で始まらなければなりません。
- 移植性のために、行の有効長は 72 文字を超えないようにする必要があります。
- 次行の最初の非ブランク文字にルールが継続するよう指示するには、行の最後の非ブランク文字として正符号 (+) を使用します。次行の先頭にルールが継続するよう指示するには、行の最後の非ブランク文字として負符号 (-) を使用します。連結文字がキーワードおよびパラメーターの内部に現れても構いません。
- 注釈行は、アスタリスク (*) で始まり、ルール・テーブルのどこにでも指定することができます。

ルール・テーブルのルール

- ブランク行は無視されます。
- DLQ ハンドラーのルール・テーブルの各項目は、1 つまたは複数のキーワードと、それらに関連付けられたパラメーターからなります。そのパラメーターは、次の構文ルールに従う必要があります。

- 各パラメーター値には、有効な文字を少なくとも 1 個は含める必要があります。引用符で囲んだ値の区切り用の引用符は無効と見なされます。たとえば、次のパラメーターは有効です。

FORMAT('ABC') 有効な文字は 3 個

FORMAT(ABC) 有効な文字は 3 個

FORMAT('A') 有効な文字は 1 個

FORMAT(A) 有効な文字は 1 個

FORMAT(' ') 有効な文字は 1 個

次のパラメーターは、有効な文字が 1 個も含まれていないため無効です。

FORMAT('')

FORMAT()

FORMAT()

FORMAT

- ワイルドカード文字がサポートされます。後書きブランク以外の 1 文字の代わりに疑問符 (?) を使用することができます。0 個以上の隣接した文字の代わりにアスタリスク (*) を使用することができます。アスタリスク (*) および疑問符 (?) は、パラメーター値の中では常に ワイルドカード文字と解釈されます。
- 次のキーワードのパラメーターの中には、ワイルドカード文字を含めることはできません。ACTION、HEADER、RETRY、FWDQ、FWDQM、および PUTAUT。
- パラメーター値の中の後書きブランク、および DLQ 上のメッセージ内のそれに対応するフィールドの中の後書きブランクは、ワイルドカード突き合わせの実行時には無効です。しかし、引用符で囲んだストリングの中の先行ブランクと組み込みブランクは、ワイルドカード突き合わせでも有効です。
- 数値パラメーターには、疑問符 (?) のワイルドカード文字を含むことはできません。1 個の数値パラメーター全体の代わりにアスタリスク (*) を使用できますが、数値パラメーターの一部として含めることはできません。たとえば、次の数値パラメーターは有効です。

MSGTYPE(2) 応答メッセージのみが適格

MSGTYPE(*) どのメッセージ・タイプも適格

MSGTYPE('*') どのメッセージ・タイプも適格

しかし、MSGTYPE('2*')は無効です。数値パラメーターの一部分としてアスタリスク (*) が含まれているためです。

- 数値パラメーターの有効範囲は、0 ~ 999 です。パラメーター値がこの範囲内であるなら、キーワードが関連するフィールドで現在無効であっても、パラメーター値は受け入れられます。数値パラメーターには、記号名を使用できます。
- キーワードが関連する MQDLH または MQMD 内のフィールドよりもストリング値が短い場合、そのストリング値は、フィールドの長さになるまでブランクが埋め込まれます。ストリング値 (アスタリスクを除外して) がフィールドより長い場合は、エラーの診断が下されます。たとえば、次のストリング値は、8 文字のフィールドに関してすべて有効です。

'ABCDEFGH' 8 文字

'A*C*E*G*I' アスタリスクを除外して 5 文字

'*A*C*E*G*I*K*M*O*'

アスタリスクを除外して 8 文字

- ブランク、小文字、もしくはピリオド (.), 斜線 (/), 下線 (_), およびパーセント記号 (%) を除く特殊文字が含まれているストリングは、単一引用符で囲む必要があります。引用符で囲んでいない小文字は、大文字に変換されます。ストリングに引用符を含む場合は、その引用符の始めと終わりの両方を示すために、2 個の単一引用符を使用する必要があります。ストリングの長さが計算されるとき、二重引用符の各オカレンスは、1 個の文字としてカウントされます。

ルール・テーブルの処理方法

DLQ ハンドラーはルール・テーブルを検索し、DLQ 上のメッセージとそのパターンが一致するルールを求めます。検索は、ルール・テーブルの最初のルールから始まって、表中を順番に進んでいきます。一致するパターンを持つルールが見つかったら、そのルールの処理が試行されます。DLQ ハンドラーは、そのルールの適用を試行するたびに、ルールの再試行カウントを 1 つずつ増分します。最初の試行が失敗すると、なされた試行のカウントが、RETRY キーワードに指定した数と一致するまで、試行は繰り返されます。試行がすべて失敗すると、DLQ ハンドラーは、ルール・テーブルの中の次に一致するルールを検索します。

このプロセスは、アクションが正常に実行されるまで、一致するルールについて順番に繰り返されます。一致するルールがそれぞれ RETRY キーワードで指定されている回数試みられ、その試みがすべて失敗した場合は、ACTION

ルール・テーブルの処理

(IGNORE) であると見なされます。一致するルールがまったく見つからないときにも、ACTION (IGNORE) が想定されます。

注:

1. 一致するルールのパターンは、接頭部が MQDLH の DLQ 上のメッセージについてのみ検索されます。接頭部が MQDLH 以外のメッセージは、エラーとして定期的に報告され、DLQ 上にいつまでも残ります。
2. すべてのパターン・キーワードを、デフォルトにして、ルールがアクションのみで構成されるようにすることができます。ただし、そのキューにおいて、テーブル内のその他のルールに従って処理されておらず、MQDLH があるキューの全メッセージにアクションのみのルールが適用されます。
3. ルール・テーブルは、DLQ ハンドラーの開始時に検査され、その時点でエラーのフラグが付けられます。(DLQ ハンドラーによって発行されたエラー・メッセージについては、MQSeries メッセージで説明されています。) ルール・テーブルにはいつでも変更を加えることができますが、DLQ ハンドラーが再始動されないと、その変更は有効になりません。
4. DLQ ハンドラーは、メッセージ、MQDLH、メッセージ記述子のいずれの内容も更新しません。DLQ ハンドラーは、常にメッセージ・オプション MQPMO_PASS_ALL_CONTEXT でメッセージを他のキューに書き込みます。
5. DLQ ハンドラーは、オプション MQOO_INPUT_AS_Q_DEF を持つ DLQ をオープンします。
6. 同じルール・テーブルを使用して同じキューに DLQ ハンドラーの複数インスタンスが並行して実行される場合もあります。しかし、DLQ と DLQ ハンドラーの間には、1 対 1 の関係があるのが普通です。

すべての DLQ メッセージを確実に処理する

DLQ ハンドラーは、すでに表示されたが除去されていない DLQ 上のすべてのメッセージを記録しておきます。DLQ からメッセージの小さいサブセットを抽出するためのフィルターとして DLQ ハンドラーを使用する場合にも、DLQ ハンドラーは、DLQ 上にある未処理のメッセージの記録を保持し続けます。また、DLQ が先入れ先出し (FIFO) として定義されても、DLQ に到着する新規メッセージが表示されることを DLQ ハンドラーは保証できません。したがって、キューが空でない限り、すべてのメッセージを検査するために DLQ が定期的に再走査されます。以上の点から、必ず DLQ にはできるだけ少数のメッセージを入れるようにしてください。廃棄したり他のキューに転送したりできない (その理由が何であっても) メッセージをキュー上に累積させると、DLQ ハンドラーのワークロードが増大し、DLQ 自体が満杯になる恐れがあります。

特定の対策を取って、DLQ ハンドラーに DLQ を空にさせることができます。たとえば、ACTION (IGNORE) は、DLQ 上のメッセージを放置するだけなので、使用しないでください。テーブルの中の他のルールによって明示的に処理されないメッセージには、ACTION (IGNORE) が適用されます。その代わり、無視するメッセージに関して、別のキューにそのメッセージを移動するアクションを実行してください。たとえば、次のようにします。

```
ACTION (FWD) FWDQ (IGNORED.DEAD.QUEUE) HEADER (YES)
```

また、テーブルの最後のルールは、テーブルの中のそれまでのルールから漏れたメッセージを処理可能なものにする必要があります。たとえば、テーブルの中の最後のルールは、次のようなものにすることができます。

```
ACTION (FWD) FWDQ (REALLY.DEAD.QUEUE) HEADER (YES)
```

テーブルの中の最後のルールが適用されることになったメッセージは、このアクションによってキュー REALLY.DEAD.QUEUE に転送されます。このキューで、そのメッセージを手動によって処理できます。このようなルールがない場合は、メッセージが DLQ に無期限に残ることになります。

DLQ ハンドラーのルール・テーブルの例

1 つの制御データ項目といくつかのルールが含まれているルール・テーブルの例を以下に示します。

```
*****
*           An example rules table for the runmqdlq command           *
*****
* Control data entry
* -----
* If no queue manager name is supplied as an explicit parameter to
* runmqdlq, use the default queue manager for the machine.
* If no queue name is supplied as an explicit parameter to runmqdlq,
* use the DLQ defined for the local queue manager.
*
inputqm(' ') inputq(' ')

* Rules
* -----
* We include rules with ACTION (RETRY) first to try to
* deliver the message to the intended destination.

* If a message is placed on the DLQ because its destination
* queue is full, attempt to forward the message to its
```

ルール・テーブルの例

- * destination queue. Make 5 attempts at approximately
- * 60-second intervals (the default value for RETRYINT).

REASON(MQRC_Q_FULL) ACTION(RETRY) RETRY(5)

- * If a message is placed on the DLQ because of a put inhibited
- * condition, attempt to forward the message to its
- * destination queue. Make 5 attempts at approximately
- * 60-second intervals (the default value for RETRYINT).

REASON(MQRC_PUT_INHIBITED) ACTION(RETRY) RETRY(5)

- * The AAAA corporation are always sending messages with incorrect
- * addresses. When we find a request from the AAAA corporation,
- * we return it to the DLQ (DEADQ) of the reply-to queue manager
- * (&REPLYQM).
- * The AAAA DLQ handler attempts to redirect the message.

MSGTYPE(MQMT_REQUEST) REPLYQM(AAAA.*) +
ACTION(FWD) FWDQ(DEADQ) FWDQM(&REPLYQM)

- * The BBBB corporation never do things by half measures. If
- * the queue manager BBBB.1 is unavailable, try to
- * send the message to BBBB.2

DESTQM(bbbb.1) +
action(fwd) fwdq(&DESTQ) fwdqm(bbbb.2) header(no)

- * The CCCC corporation considers itself very security
- * conscious, and believes that none of its messages
- * will ever end up on one of our DLQs.
- * Whenever we see a message from a CCCC queue manager on our
- * DLQ, we send it to a special destination in the CCCC organization
- * where the problem is investigated.

REPLYQM(CCCC.*) +
ACTION(FWD) FWDQ(ALARM) FWDQM(CCCC.SYSTEM)

- * Messages that are not persistent run the risk of being
- * lost when a queue manager terminates. If an application
- * is sending nonpersistent messages, it should be able
- * to cope with the message being lost, so we can afford to
- * discard the message.

PERSIST(MQPER_NOT_PERSISTENT) ACTION(DISCARD)

- * For performance and efficiency reasons, we like to keep
- * the number of messages on the DLQ small.
- * If we receive a message that has not been processed by
- * an earlier rule in the table, we assume that it
- * requires manual intervention to resolve the problem.
- * Some problems are best solved at the node where the
- * problem was detected, and others are best solved where
- * the message originated. We don't have the message origin,
- * but we can use the REPLYQM to identify a node that has

- * some interest in this message.
- * Attempt to put the message onto a manual intervention
- * queue at the appropriate node. If this fails,
- * put the message on the manual intervention queue at
- * this node.

```
REPLYQM('?*') +  
  ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION) FWDQM(&REPLYQM)  
  
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION)
```

ルール・テーブルの例

第10章 インストゥルメンテーションおよび EMS イベント

この章の内容は次のとおりです。

- キュー・マネージャーの操作をモニターするために使用することのできる、MQSeries インストゥルメンテーション・イベントの簡単な紹介。『MQSeries インストゥルメンテーション・イベント』を参照してください。インストゥルメンテーション・イベントの詳細については、MQSeries プログラム式システム管理 を参照してください。
- MQSeries for Compaq NSK による Event Management Service (EMS) イベントの使用に関する説明。224ページの『Event Management Service (EMS) イベント』を参照してください。

MQSeries インストゥルメンテーション・イベント

キュー・マネージャーが事前定義された 1 組の条件を検出すると、インストゥルメンテーション・イベントによって、イベント・メッセージ が生成されます。たとえば、キュー満杯 イベントは、次のような条件が原因となって発生します。

- キュー満杯イベントは、指定されたキューについて有効です。
- アプリケーションは、そのキューにメッセージを書き込むために MQPUT 呼び出しを発行しますが、キューが満杯であるため、その呼び出しは失敗します。

インストゥルメンテーション・イベントを引き起こす可能性のあるその他の条件には、次のようなものがあります。

- キューのメッセージ数に関する限度に達した。
- 指定した時間内にキューの運用が実行されない。
- チャネル・インスタンスが開始または停止した。
- アプリケーションが、許可されていないユーザー ID を指定してキューをオープンしようとした。

チャネル・イベントの例外として、すべてのインストゥルメンテーション・イベントは、生成される前に有効な状態になっている必要があります。

インスツルメンテーション・イベント

イベント・メッセージには、そのイベントを引き起こした条件に関する情報が含まれます。これは、イベント・キュー 書き込まれます。アプリケーションは、分析のためにこのキューからイベント・メッセージを取り出すことができます。

イベント・キューをリモート・キューとして定義する場合、すべてのイベント・キューを (インスツルメンテーション・イベントをサポートする複数のノード用として) 1 つのキュー・マネージャーに置くことができます。これによって、生成されたイベントを使用して、単一のノードからキュー・マネージャーのネットワークをモニターできます。

イベントのタイプ

インスツルメンテーション・イベントには、次の 4 つのタイプがあります。

キュー・マネージャー・イベント

キュー・マネージャー・イベントは、キュー・マネージャー内のリソースの定義に関係したものです。キュー・マネージャー・イベントは、たとえば、アプリケーションが、存在しないキューにメッセージを書き込もうとする場合などに生成されます。

パフォーマンス・イベント

パフォーマンス・イベントは、リソースのしきい値に達したことを通知するものです。パフォーマンス・イベントは、たとえば、キューの制限項目数に達した場合や、あるいは MQGET コール後に、事前定義された時間制限内にキューが運用されなかった場合に生成されます。

チャンネル・イベント

チャンネル・イベントは、チャンネル操作時に検出された条件の結果として、チャンネルによって報告されます。チャンネル・イベントは、たとえば、チャンネル・インスタンスが停止する場合に生成されます。

トリガー・イベント

トリガー・イベントは、トリガー・イベントの条件が満たされたことをキュー・マネージャーが検出したときに起こります。たとえば、メッセージが到着するたびにトリガー・イベントを生成するようキューを構成できます。(トリガー・イベントとインスツルメンテーション・イベントの条件はまったく異なります。)

トリガー・イベントが起きると、トリガー・メッセージが開始キューに入れられ、オプションで、アプリケーション・プログラムが開始されます。

イベント・キューによるイベント通知

イベントが起こると、キュー・マネージャーはイベント・メッセージを該当のイベント・キュー (定義されている場合) に書き込みます。イベント・メッセージには、次のような適切な MQI アプリケーション・プログラムを作成すれば取り出すことができるイベントの情報が入っています。

- キューからメッセージを取得するアプリケーション。
- イベント・データを取り出すメッセージを処理するアプリケーション。イベント・メッセージの形式については、*MQSeries* プログラム式システム管理を参照してください。

イベントの各カテゴリーには、それぞれのイベント・キューがあります。そのカテゴリーのイベントが起こると、すべて同じキューにイベント・メッセージが入れられることとなります。

イベント・キュー	入れられるメッセージの発生元
SYSTEM.ADMIN.QMGR.EVENT	キュー・マネージャー・イベント
SYSTEM.ADMIN.PERFM.EVENT	パフォーマンス・イベント
SYSTEM.ADMIN.CHANNEL.EVENT	チャンネル・イベント

トリガー・イベント・キューを使用する

イベントが生成されると、イベント・メッセージがイベント・キューに書き込まれ、それによってユーザー作成のモニター・アプリケーションが開始されるように、トリガー付きのイベント・キューをセットアップできます。このアプリケーションはイベント・メッセージを処理し、適切なアクションを実行できます。たとえば、オペレーターに通知する必要のあるイベントもあれば、種々の管理タスクを実行するアプリケーションを自動的に開始するイベントもあります。

インストルメンテーション・イベントの有効化

インストルメンテーション・イベントを有効化する方法は、イベント・タイプによって決まります。

- キュー・マネージャー・イベントは、キュー・マネージャー上の属性を設定することにより、有効化されます。
- パフォーマンス・イベントは全体として、キュー・マネージャー上で有効化する必要があります。適切なキュー属性を設定することによって特定のパフォーマンス・イベントを有効化し、キュー項目数の上限などの、イベントを引き起こす条件を指定する必要もあります。

インスツルメンテーション・イベント

- チャネル・イベントは自動的に起こるので、有効化する必要はありません。チャネル・イベントをモニターしない場合には、チャネル・イベント・キューを書き込み禁止にすることができます。

以下のいずれかを使用して、インスツルメンテーション・イベントの生成を有効化および無効化することができます。

- MQSC コマンド。詳細については、*MQSeries MQSC コマンド・リファレンス* を参照してください。
- キュー・マネージャーの PCF コマンド。詳細については、*MQSeries プログラム式システム管理* を参照してください。
- MQAI コマンド。詳細については、*MQSeries 管理インターフェースのプログラミングの手引きおよび解説書* を参照してください。

イベント・メッセージ

イベント・メッセージには、イベントの発生元に関係した情報（イベントのタイプ、イベントを引き起こしたアプリケーションの名前など）、さらにパフォーマンス・イベントの場合にはキューの短い統計要約も含まれています。

イベント・メッセージの形式は、PCF 応答メッセージの形式と類似しています。メッセージ・データは、*MQSeries プログラム式システム管理* に説明されているデータ構造を使用したユーザー作成の管理プログラムによって、イベント・メッセージから取り出すことができます。

Event Management Service (EMS) イベント

MQSeries for Compaq NSK は、MQSeries のキュー・マネージャー・イベント、チャネル・イベント、およびパフォーマンス・イベントに対応する Event Management Service (EMS) イベント・メッセージを生成します。また、MQSeries ログ内のメッセージ項目および FFST に対応する EMS メッセージが生成されることもあります。これらのイベント・メッセージは、システム・オペレーターおよびアドミニストレーターに対して、MQSeries 操作環境に悪影響を及ぼす可能性のあるソフトウェア条件についての警告を与えます。

MQSeries for Compaq NSK に付属する EMS テンプレート・ファイル

サブボリューム ZMQSSYS では、次のファイルが提供されています。

ZMQSTMPL (ファイル・コード 839)

MQSeries が生成する EMS イベントの書式設定テンプレートが入った、EMS テンプレート・オブジェクト・ファイル。

ZMQSDDL (ファイル・コード 101)

MQSeries が生成する EMS イベントのデータ定義言語スキーマ。

ZMQSC (ファイル・コード 101)

この製品が生成する EMS イベントの定義の DDL コンパイラーからコンパイルされた出力 (C)

ZMQSCOB (ファイル 101)

この製品が生成する EMS イベントの定義の DDL コンパイラーからコンパイルされた出力 (COBOL)

ZMQSPAS (ファイル・コード 101)

この製品が生成する EMS イベントの定義の DDL コンパイラーからコンパイルされた出力 (PASCAL)

ZMQSTACL (ファイル・コード 101)

この製品が生成する EMS イベントの定義の DDL コンパイラーからコンパイルされた出力 (TACL)

ZMQSTAL (ファイル・コード 101)

この製品が生成する EMS イベントの定義の DDL コンパイラーからコンパイルされた出力 (TAL)

サブボリューム ZMQSSYS には、EMS テンプレート・ファイル SMQSTMPL が含まれ、これを元にしてテンプレート・ファイル ZMQSTMPL が生成されます。ファイル ZMQSTMPL は、COUP および SYSGEN を使って、ご使用のシステムのイベント・テンプレートに組み込むことができます。イベント・テンプレートのソースが提供されているため、ご使用の環境でイベントを使用する時に、そのイベントの書式設定を変更することができます。

たとえば、あるイベントに含まれている情報をすべては表示したくない場合や、そのイベント内の情報に付随して表示されるテキストを追加または変更したい場合があるかもしれません。EMS イベント・テンプレートのソース言語の説明、および代替 ZMQSTMPL ファイルを作成するために定義をコンパイルする手順については、Compaq の資料を参照してください。

MQSeries EMS イベント・テンプレートの統合

テンプレート・オブジェクト・ファイルを、ご使用のシステムの常駐および非常駐の EMS テンプレート・ファイルに組み込んで、VIEWPOINT および EMSDIST などのプログラムが MQSeries EMS イベントの書式設定や表示を行えるようにする必要があります。

EMS イベント

この節の残りの部分では、MQSeries EMS テンプレートをシステム・テンプレートに組み込むための手順を説明します。システムによっては、別の手順のほうが望ましい場合もあるということに注意してください。

1. COUP コマンド INFO ALLPROCESSORS を使用して、現行システム・テンプレートの名前を判別します。EMS[^]TEMPLATES パラメーターに表示される値をご覧ください。たとえば、次のようにします。

```
$DEV2 ZMQSSYS 425> coup
CONFIGURATION UTILITY PROGRAM - T9023D30 - (26MAY95)      SYSTEM
#RAPTOR
COPYRIGHT Compaq COMPUTERS INCORPORATED 1987-1994
CONFIG $SYSTEM.SYS06.OSCONFIG
1) info allprocessors
   EMS^TEMPLATES ( RESIDENT $SYSTEM.SYS01.RTMLATE,
                  NONRESIDENT $SYSTEM.SYS01.RTMLATE )
   SYSTEM^ID      ( NAME #RAPTOR, NUMBER 001 )
   SYSTEM^TIME   ( GMT^OFFSET -05:00, DST USA66 )
   DP2_UPSOPTION ( OFF )
2) exit
```

2. SCF コマンド ASSUME SUBSYS \$ZZKRN; INFO を使用して、G Series オペレーティング・システムの現行システム・テンプレートの名前を判別します。NONRESIDENT_TEMPLATES と RESIDENT_TEMPLATES に表示される値をご覧ください。たとえば、次のようにします。

```
SCF;ASSUME SUBSYS $ZZKRN;INFO
NONSTOP KERNEL - Info SUBSYS #HAWK.$ZZKRN
Current Settings
*DAYLIGHT_SAVING_TIME ..... USA66
*NONRESIDENT_TEMPLATES..... $SYSTEM.SYS01.TEMPLATE
*POWERFAIL_DELAY_TIME..... 30
*RESIDENT_TEMPLATES..... $SYSTEM.SYS01.RTMLATE
SUPER_SUPER_IS_UNDENIABLE..... OFF
*SYSTEM_NAME..... #HAWK
*SYSTEM_NUMBER..... 2
SYSTEM_PROCESSOR_TYPE ..... NSR-W
*TIME_ZONE_OFFSET..... -05:00

Pending Changes (will take effect at next system load)
None Total Errors = 0 Total Warnings = 0
```

3. TEMPLI コンパイラーを実行して、現行システム・テンプレートと新しい MQSeries テンプレートを結合した、新規のシステム・テンプレート・ファイルを作成します。これは、次のような 2 段階の処理になります。

- a. 以下のコマンドを含むテキスト・ファイルを作成します。

```
FILE <current NONRESIDENT system template file>
FILE <MQSeries install volume>.ZMQSSYS.ZMQSTMPL
EXIT
```

たとえば、次のようにします。

```
FILE $SYSTEM.SYS06.TEMPLATE
FILE $DEV2.ZMQSSYS.ZMQSTMPL
EXIT
```

- b. 新規テキスト・ファイルを入力データとして指定して、次のように TEMPLI コンパイラを実行します。

```
TEMPLI /IN <command file>/<new resident template file>, <new
nonresident template file>
```

たとえば、作成したコマンド・ファイルが TEMGUIDE という名前であり、\$SYSTEM.EMS に新規のテンプレート・ファイルを作成する場合は、次のようになります。

```
TEMPLI /IN TEMGUIDE/$SYSTEM.EMS.NEWRES, $SYSTEM.EMS.NEWRES
```

システムで必要とされるすべての EMS イベント・テンプレートが処理されるので、新規のテンプレート・ファイルのコンパイルには、数秒かかることがあります。

4. COUP コマンドを使って、現行テンプレートの代わりに新規の EMS イベント・テンプレートを使用するように、システムを次のように構成します。

```
ASSUME ALLPROCESSORS
ALTER EMS^TEMPLATES(RESIDENT <new resident template file>,
NONRESIDENT <new nonresident template file>)
EXIT
```

EMS イベント

G Series の場合は、SCF コマンドを使用して、システムが新規の EMS イベント・テンプレートを使用するように構成します。

```
ALTER $ZZKRN, RESIDENT_TEMPLATES $SYSTEM.SYS01.NEWRES
ALTER $ZZKRN, NONRESIDENT_TEMPLATES $SYSTEM.SYS01.NEWNRES
EXIT
```

注: この変更を永続的なものにするには、SYSGEN を使用してシステムを更新する必要があります。

EMS テンプレートの詳細については、*DSM Template Services Manual* を参照してください。この資料は、SYSGEN を使用してこのタスクを実行する方法についても説明しています。

PARAM MQEMSEVENTS の定義

MQSeries EMS イベントの有効化を完了するために、PARAM MQEMSEVENTS が正しく定義されていることを確かめる必要があります。この値はビットマップとして解釈される 4 文字ストリングで、次のようになります。

EMS メッセージ	ビットマップ・エントリー	MQEMSEVENT 値
FFST	0x00000001	1
START / STOP	0x00000002	2
PERFORMANCE	0x00000004	4
CHANNEL	0x00000008	8
QUEUE MANAGER	0x00000010	16
MESSAGE	0x00000020	32
ERROR	0x00000040	64
ALL	0x0000007F	127

したがって、MQSeries のすべての EMS イベントのスイッチをオンにするには、管理コマンドの発行元となる TACL 環境で次の PARAM を定義しなければなりません。

```
PARAM MQEMSEVENTS 127
```

この定義は、MQSeries のすべてのサーバー・クラスのサーバー・クラス定義でも必要です。各サーバー・クラスは、様々なオプションを指定して構成することができます。詳細については、54ページの『PATHWAY サーバー・クラスのパラメーターの変更』を参照してください。

デフォルトでは、EMS イベントは生成されません (つまり、PARAM は定義されていません)。

代替コレクターの使用

Compaq NSK システムには、\$0 と呼ばれるデフォルトの EMS イベント・コレクターが常時存在しています。MQSeries キュー・マネージャーが生成する EMS イベントはすべて、このデフォルト・コレクターに送信されます。別のコレクターでキュー・マネージャーの EMS イベントを収集する場合は、QMINI ファイル内の Configuration スタンザの中の EMSCollector エントリを変更して、キュー・マネージャーを再始動してください。各キュー・マネージャーごとに、別々の EMS イベント・コレクターを指定することもできます。

MQSeries EMS イベントを処理するためのプログラムの作成

EMS イベント・メッセージを処理することによって、MQSeries キュー・マネージャーをモニターするアプリケーションを作成することができます。このようなアプリケーションは、生成された EMS イベント・メッセージに対する応答として PCF コマンドを発行することによって、キュー・マネージャーの操作に影響を与えることもあります。

ZMQSSYS サブボリューム内の MQSeries for Compaq NSK に付属しているファイル ZMQSC、ZMQSTAL、ZMQSCOB、ZMQSPAS、および ZMQSTACL は、C、TAL、COBOL、PASCAL、TACL で MQSeries EMS イベント・メッセージに含まれるトークンを定義します。これらの定義は、管理プログラムがメッセージの書式を理解するために使用されます。

MQSeries によって生成される EMS イベントの詳細については、507ページの『付録P. MQSeries for Compaq NSK で使用する EMS イベント・テンプレート』を参照してください。

第11章 トランザクション・サポートとメッセージングについての理解

メッセージ・キュー・インターフェース (MQI) を使用するアプリケーションでは、同期点制御のもとで、書き込み操作や読み取り操作が実行されます。MQSeries for Compaq NSK には、次の 2 つのトランザクション操作があります。

- コミット — トランザクションを完了することにより、データベースへの変更を記録して永続的なものにする行為。トランザクションがコミットされた後に、保護リソースが解放される。
- バックアウト — 現行のリカバリー単位または作業単位中に行われたすべての変更内容を取り消す操作。この操作の完了後、新規のリカバリー単位または作業単位が開始される。

コミットとバックアウトは、Compaq NSK 上の TM/MP (または TMF) トランザクション環境の一部として提供されています。MQSeries for Compaq NSK では、MQPUT、MQGET、および MQPUT1 はデフォルトでは同期点操作です。つまり、非同期点がアプリケーションによって明示的に要求されない限り、TMF トランザクションは進行中でなければならず、そうしないと MQI 呼び出しは失敗します。

アプリケーションは、TM/MP トランザクションの開始には TM/MP BEGINTRANSACTION プロシージャを使用し、トランザクションのコミットには ENDTRANSACTION を使用し、さらにトランザクションのバックアウトには ABORTTRANSACTION を使用します。

さらに、MQSeries for Compaq NSK は、TM/MP を介して同期点操作を提供する製品 (たとえば NonStop Tuxedo) もサポートしています。

NonStop TM/MP (トランザクション・マネージャー) の使用

MQSeries for Compaq NSK V5.1 は、Compaq の NonStop TM/MP のトランザクション管理機能に依存して、トランザクションの整合性の保守を行います。

NonStop TM/MP トランザクション・システムは、トランザクションの保護と並行性、およびオブジェクト・カタログとメッセージの整合性を提供しています。

TM/MP の使用

TM/MP トランザクションは、MQSeries の書き込み操作および読み取り操作を、ENSCRIBE や NonStop SQL データベース・ファイルに対してアプリケーションが行う更新に合わせて調整することができます。

MQSeries は、TM/MP トランザクションを透過的に処理します。MQI 関数が呼び出される時に進行中の TM/MP トランザクションがあると、同期点オブションが指定された書き込み操作や読み取り操作はすべて、この同じトランザクションの一部になります。つまり、そのトランザクションがコミットされる時に、キューの更新が行われます。障害の発生時に、TM/MP は、コミットされたすべてのトランザクションがデータベース・ファイルに適用され、コミットされていないトランザクションがバックアウトされるようにします。トランザクション・バックアウトは、事前イメージをデータベース・レコードに再度適用して、取り消されたトランザクションの影響を元に戻します。コミット操作が完了するまでは、変更は行われません。

ユーザー・アプリケーションに進行中のトランザクションがあるときに、これが同期点の外部で MQI 呼び出しを試行する場合、MQSeries は現行の継承したトランザクションを保留にし、独自のトランザクションの 1 つを開始してそのトランザクションをコミットしてから、元のトランザクションを再開し、最後にユーザー・アプリケーションに制御を返します。書き込みおよび読み取り操作の結果として、キューへの変更が即座に行われます。

TM/MP トランザクションは、持続メッセージだけでなく、非持続メッセージの書き込みおよび読み取り操作の調整にも使用されます。MQSeries for Compaq NSK は、*OpenTMF* と呼ばれる TM/MP への特殊なインターフェースを使用して、この調整の実行を可能にします。

同期点の制限

ファイル・システムは、持続メッセージ・データを保持する物理ファイル上のレコード・ロックの数を制限することによって、1 つのトランザクション内で書き込みや読み取りを行うことのできる持続メッセージ・データの量を制限することができます。

ファイル・システムの、トランザクション当たりのデフォルトのロック制限は、ディスク・ボリューム当たり 5000 ロックです。これは、SCF を使用して変更することができます。たとえば、次のコマンドは、トランザクション当たりのロック制限を、ディスク・ボリューム当たり 10000 ロックに制限します。


```
ALTER DISK $DISK01, MAXLOCKSPERTCP 10000
```

キュー・オーバーフロー・ファイルに保管されるメッセージの場合、(サイズがメッセージ・オーバーフロー・ファイルを使用するためのしきい値サイズより小さいために) メッセージの数とサイズは制限されます。

メッセージ・オーバーフローしきい値サイズは、200 KB 以下に設定することをお勧めします。このメッセージ・サイズであれば、デフォルトのレコード・ロック制限で、1 つのトランザクション内に 100 メッセージ (ほとんどのアプリケーションの必要を満たす数) を収容することができます。加えて、メッセージ・オーバーフロー・ファイルを使用することによるパフォーマンス上の利益は、このメッセージ・サイズの時に意味のあるものとなります。

また、キュー・オーバーフロー・ファイルの代わりにメッセージ・オーバーフロー・ファイルを使用すると、TM/MP 監査証跡の使用が大幅に削減されるという点にも注意してください。

メッセージ・オーバーフロー・ファイルとキュー・オーバーフロー・ファイルの相違点や利点についての詳細は、290ページの『メッセージ・オーバーフロー・ファイル』を参照してください。

持続メッセージでの非同期点操作

持続メッセージでは、メッセージ・データを保持する ENSCRIBE ファイルを更新するために、TM/MP トランザクションがキュー・サーバーによって内部で開始されなければなりません。1 つのプロセスで開始される並行トランザクションの数は、NSK ファイル・システムによって 100 個と制限されています。そのため、1 つのキュー・サーバーがサポートできる非同期点持続メッセージの PUT または GET 操作の数は 100 だけです。これが起きる可能性があるのは、複数の大きなメッセージ (32 KB を超える) が、同じキュー・サーバーを介して同時にエンキューまたはデキューされる場合のみです。

この状態が生じると、MQPUT または MQGET は理由コード MQRC_SYNCPOINT_LIMIT_REACHED によって終了されます。複数のプロセス間に負荷を分散させるために、代替キュー・サーバーにキューの再割り当てを行うか、または別のキュー・サーバーによってホストされた別のキューを使用するようにアプリケーションを変更してください。

非持続メッセージでの同期操作

非持続メッセージは、監査済みディスク・ファイルではなく、メモリーに保管されるため、それ自体は監査証跡スペースを必要としません。MQSeries は、TM/MP の内部インターフェースを使用して、同期点内でエンキューまたはデキューされる非持続メッセージの可用性を制御します。1 つの同期点操作に、持続メッセージと非持続メッセージが混在していることもあります。MQSeries は、TM/MP トランザクションの完了時に、すべての非持続メッセージの操作がトランザクションの実際の結果に基づいて、持続メッセージの操作と同時に論理的にコミットまたはバックアウトされるようにします。

TM/MP および MQSeries for Compaq NSK の構成要件

NSK システムは、キュー・マネージャーまたはキューを入れるすべてのボリュームに対して TMF (TM/MP) 監査を使用可能にして構成する必要があります。システム上のボリュームの監査状況を判別するには、TMFCOM コマンド `status datavols` を使用します。(TMFCOM を使用するためには、ユーザーが SUPER.SUPER でなければなりません。) さらに、キュー・マネージャーをサポートするデータ・ボリューム用に構成される TMF 監査証跡は、これらのボリュームを使用するすべてのキュー・マネージャーにおいて予測されるメッセージ・トラフィックのピークの比率とサイズに十分見合う大きさでなければなりません。

動作不良のあるアプリケーションは持続期間の長いトランザクションを引き起こすことがあるため、TMF システムは、持続期間の長いトランザクションを自動的に取り消すように構成する必要があります。監査証跡のサイズと、持続時間の長いトランザクションの時間制限は、アプリケーション依存の調整パラメーターです。MQSeries 用に構成される監査証跡は、テープにダンプするように構成する必要はありません。

モニター

MQSeries が稼働している状態で TMF の状況をモニターするには、TMFCOM インターフェースを使用します。 `status tmf` および `status datavols` コマンドを使用して、TMF の一般状況と個々のデータ・ボリュームの状況を調査します。

重大な TM/MP イベントのシステム・イベント・ログ (EMS) もモニターする必要があります。これは、TM/MP 内に、MQSeries やそれを使用するアプリケーションに影響を及ぼすかもしれない問題が将来発生する可能性があることを示します。TM/MP は、MQSeries のクリティカル・リソースであり、MQSeries が適切に機能するために継続的に作動している必要があります。

監査証跡のサイズ

TM/MP 監査証跡のおおよそのサイズは、次の指針をもとに計算することができます。

- 監査証跡スペースは、持続メッセージ操作（書き込みおよび破壊的な読み取り）にのみ必要である。
- 監査証跡スペースのおおよそのサイズは、メッセージ・データ・サイズの合計に 1500 バイトを加えたものである。
- メッセージ・オーバーフローしきい値を超えるメッセージを含む持続メッセージ操作には、1 つの書き込み操作または読み取り操作につき（そのサイズには関係なく）4 KB だけの監査証跡が必要である。

リソース・マネージャーの構成

MQSeries を使って処理されることが予想されるトランザクションのボリューム用に、TM/MP の内部インターフェースを適切に構成する必要があります。

TM/MP サブシステムに関する限り、MQSeries キュー・サーバーがキュー・マネージャーの役割を果たします。また、TM/MP サブシステムには、リソース・マネージャーに適用される様々なしきい値や制限があります。必須構成は、使用するキュー・サーバーの数と、CPU 上でのキュー・サーバーの分散、および同時に進行する並行同期点操作の数によって異なります。値を変更するには、TMFCOM の ALTER BEGINTRANS コマンドを使用します。これについてはこの後で説明します。

RMOPENPERCPU - 少なくとも、CPU で実行するキュー・サーバーの最大数の 2 倍が必要です。通常は、デフォルト値の 128 で十分です。

BRANCHESPERRM - 少なくとも、1 つのキュー・サーバーが処理できる並行同期点操作の最大数が必要です。通常は、デフォルト値の 128 で十分ですが、もし不十分であれば、このパラメーターを最大値の 1024 にまで増やすことができます。あるいは、キューを他のキュー・サーバーに割り当てて、キュー・サーバーが処理する並行同期点操作の最大数を減らすこともできます。

これらのパラメーターの新規値を有効にするには、TM/MP サブシステムを停止して再始動する必要があります。

トラブルシューティング

BEGINTRANSACTION コマンドが TMF によって使用不可にされていることを示す EMS イベントまたは FFST レポートは、通常は監査証跡が満杯であることを意味します。この発生原因として、監査証跡が小さすぎるか、動作不良

TM/MP 構成要件

のアプリケーションが持続時間の長いトランザクションを保留にしているために、TMF が時間内に終了していないことが考えられます。

この場合は次のようにします。

- 監査証跡のサイズを大きくする。
- 持続時間の長いトランザクションの原因を判明して、それを訂正する。
- 持続時間の長いトランザクションがもっと短期間で終了するように、TMF を再構成する。

TMF が稼働していないことを示す EMS イベントと FFST レポートは、TMF に構成上の問題があり、キュー・マネージャーを再び稼働する前にそれを訂正しなければならないことを示しています。一般的に、MQSeries キュー・マネージャーは、どの容量でも作動するように、TMF が正しく稼働していることを必要とします。メッセージが失われたり、破壊されたりすることはありませんが、TMF がなければキュー・マネージャーは作動できません。

第12章 リカバリーおよび再始動

メッセージング・システムでは、システムに入ってきたメッセージが確実にその宛先に配布されます。また、メッセージング・システムには、システム内のメッセージを追跡したり、何らかの理由でシステムに障害が起きた場合にメッセージをリカバリーしたりする手段が備わっていなければなりません。

MQSeries for Compaq NSK は、Compaq NonStop トランザクション・マネージャー (TM/MP) を使用することにより、持続メッセージが失われることがないようにしています。TM/MP は、トランザクション保護、キューとファイル整合性、およびキュー・ファイル・リカバリーを提供しています。

また、MQSeries for Compaq NSK は NonStop プロセスのペア・テクノロジーを使用して、非持続メッセージであっても障害からの強い回復力を備えるようにしています。メッセージのストレージの役割を担うキュー・サーバーは、別の CPU で実行しているそれらのバックアップ・プロセスに非持続メッセージのチェックポイントをとります。

非持続メッセージのチェックポイントは、**altmqfls** を使用して構成できるキューごとのオプションです。非持続メッセージのチェックポイントは、デフォルトでは使用可能になっています。

TM/MP サブシステムは、現行トランザクションとデータベース整合性のための複合操作である、ユーザー操作と MQSeries 操作の両方を管理し、これらの操作がユーザーとアプリケーション・プログラムの両方に対して透過的になるようにします。

リカバリーによってキュー・マネージャーは、停止した時の状態に復元されます。進行中のトランザクションはすべてロールバックされ、キュー・マネージャーの停止時にコミットされていなかったメッセージはキューから除去されます。リカバリーによって、持続メッセージはすべて復元され、非持続メッセージはこのプロセスの間に失われることとなります。

この章の残りの部分では、リカバリーと再始動の概念をさらに詳しく紹介し、問題が発生したときにリカバリーする方法を説明します。この章では、次の事柄を説明します。

- 238ページの『フォールト・トレランスとリカバリー』
- 238ページの『MQSeries のバックアップと復元』

フォールト・トレランスとリカバリー

MQSeries バージョン 5.1 製品および Compaq NSK のシステム・ソフトウェアとハードウェアが正しく構成されている (たとえば、Compaq の規定どおりに、すべてのコンポーネントが冗長デバイス、二重化デバイス、またはプロセス・ペアとして構成されている) 場合は、ハードウェアまたはソフトウェアの 1 つのコンポーネントの障害が原因となってデータの消失、重複、破壊、あるいはシステムのいずれかの機能の永久的な消失 (復元するには外部からの介入が必要) が生じることはありません。MQSeries for Compaq NSK V5.1 は、上記で言及したデータ保全性を維持しながら、単一ポイントでの障害からリカバリーを行うことができます。

同じソフトウェア・コンポーネントの繰り返される連続障害 (たとえば、障害とリカバリーのループ) は、構成されている障害の最大数を超えると、トラップされます。このような場合や、複数ポイントでの障害の場合には、MQSeries 製品はキューの保全性を維持することはできません。

データ保全性と可用性を保つためのキュー・マネージャーのセットアップに関する詳細は、299ページの『第16章 データ保全性と可用性』を参照してください。

MQSeries のバックアップと復元

定期的にキュー・マネージャーのデータのバックアップを作成することによって、ハードウェア障害によるデータの破壊からシステムを保護することができます。

MQSeries のバックアップ

キュー・マネージャーのデータをバックアップするには、次の作業が必要です。

1. キュー・マネージャーが実行中でないことを確認します。

キュー・マネージャーが実行されている場合は、**endmqm** コマンドでそれを停止してください。

注: 実行中のキュー・マネージャーのバックアップを作成しようとする時、ファイルがコピーされたときに更新が進行中であるため、バックアップが一貫していない可能性があります。

2. キュー・マネージャーがデータを保管しているボリュームとサブボリュームを見付けます。

これらのディレクトリーを判別するために、構成ファイル中の情報を使用できます。詳細については、243ページの『第13章 構成ファイル』を参照してください。

注: ディレクトリーに表示されている名前がよく分からない場合は、MQSeries を使用するプラットフォームとの互換性を持たせるために名前が変換されているからです。名前の変換の詳細については、79ページの『ボリューム構造』を参照してください。

3. すべてのキュー・マネージャーのデータおよびログ・ファイルのサブボリュームのコピーを作成します。

どのファイルも見落とさないようにしてください。

MQSeries の復元

キュー・マネージャーのデータのバックアップを復元するには、次の作業が必要です。

1. キュー・マネージャーが実行中でないことを確認します。
2. キュー・マネージャーがデータを保管しているサブボリュームを見付けます。この情報は、構成ファイルの中にあります。
3. バックアップ・データを入れるサブボリュームを空にします。
4. バックアップされたキュー・マネージャーのデータを、正しい場所にコピーします。

できあがったディレクトリー構造を検査して、必要なディレクトリーがすべて入っていることを確認してください。

MQSeries が復元データを見つけるために正しい場所を調べられるように、MQSeries およびキュー・マネージャーの構成ファイルが一貫しているかどうか検査してください。

データが正しくバックアップされ復元されていれば、キュー・マネージャーは開始します。

状況サーバーとキュー・サーバーのリカバリーおよび再始動

状況サーバーとキュー・サーバーのプロセスは、Compaq NSK プロセスのペアです。これは、これらが、単一の CPU またはプロセスそのものの障害の発生時にも、そのサービスを提供し続けるように設計されていることを意味します。単一障害（たとえば、1 次状況サーバーまたはキュー・サーバーのプロセスを含む CPU に障害が起きたか、1 次状況サーバーまたはキュー・サーバー

状況サーバーとキュー・サーバーのリカバリーおよび再始動

のプロセスそのものが失敗した) の場合は、キュー・マネージャーの処理を中断しなくても、バックアップの状況サーバーまたはキュー・サーバーのプロセスが新規 1 次サーバーとして引き継ぎます。

通常の単一ポイント障害状況では、状況サーバーやキュー・サーバーに固有なリカバリー・アクションは必要とされていません。バックアップが 1 次によって再始動される必要が生じるたびに、あるいはバックアップが 1 次として引き継ぐたびに、メッセージが、状況サーバーまたはキュー・サーバーによって、ホーム端末とメッセージ・ログ・ファイルに記録されます。

より深刻な障害 (たとえば、1 次またはバックアップの状況サーバーやキュー・サーバーの初期化を妨げる環境障害) の場合は、TS/MP PATHMON プロセスが、状況サーバーまたはキュー・サーバーの再始動を 10 回まで試行します。

状況サーバーまたはキュー・サーバーは、TM/MP によって保護されるデータベースだけを使用するため、ディスクへのアクセスに影響する障害が発生した場合には、TM/MP と DP2 ディスク・システムが提供する保護をあてにすることができます。

通常は必要ではありませんが、状況サーバーやキュー・サーバーは、TS/MP PATHCOM コマンドを使用することにより、必要に応じて、個別に停止および再始動することができます。 **strmqm** コマンドは、文字ストリング **MQS-STATUS** または **MQS-QUEUE** で始まる名前を持つ、すべての状況サーバーまたはキュー・サーバーを自動的に開始します。キュー・マネージャーのシャットダウン時に、アクティブなすべての状況サーバーまたはキュー・サーバーのクラスがそれらのシャットダウンを調整し、アクティブなすべてのキュー・マネージャー接続が終了する時には、その接続に関係する状況サーバーまたはキュー・サーバーがすべてシャットダウンします。

Compaq NSK は、NonStop プロセス・ペアの 1 次プロセスの障害発生時に、特定の状況のもとではトランザクションを打ち切ります。これによって、1 次プロセスの障害発生時に進行している MQPUT または MQGET 操作は失敗します。詳細については、299ページの『第16章 データ保全性と可用性』を参照してください。

RDF を使用した災害時回復

災害によって 1 次サイトが使用できなくなった場合は、次の手順を使用して、キュー・マネージャーをバックアップ・サイトで作動させる必要があります。

1. RDF がデータベースの更新を完了していることを確認します。

状況サーバーとキュー・サーバーのリカバリーおよび再始動

2. FUP を使用して、以下のファイルの監査フラグをオンにセットします。
 - a. <qmgr>M サブボリューム内のすべてのファイル
 - b. QMINI、AMQRFNxx、CCSIDMEM、QMINIMEM、STATABLE、UMQSINI、PATHCTL、TRACEOPT、SHUTDOWN を除いた <qmgr>D サブボリューム内のすべてのファイル
 - c. <qmgr>S サブボリューム内のすべてのファイル
3. デフォルト・ボリュームを <qmgr>D サブボリュームに設定します。PATHMON を手操作で実行し、それに対して PATHCOM を実行して、キュー・マネージャーの PATHWAY 構成をロードします。
4. すべてのオブジェクトに対して INFO コマンドを実行します。
 - a. ノード名のすべてのインスタンスが "※*" として表示されているか、あるいはバックアップ・システム用の正しいノード名を持っているかを検査します。
 - b. サーバー・クラスに割り当てられている CPU 番号が、このバックアップ・サイトでも有効であるかどうかを検査します。
 - c. ホーム端末と Out ファイル名が、バックアップ・サイトで有効であるかどうかを検査します。有効でなければ、それらを変更してください。
 - d. 指定された代替 TCP/IP プロセス名が、このシステムで有効であるかどうかを検査します。
 - e. バックアップ・サイトそのものが RDF 操作用に構成されていない場合は、PARAM MQRDF の設定をすべて EC サーバー・クラス定義から除去します。
 - f. PATHWAY 構成を検査したら、shutdown2 コマンドを使用してこれを再度ディスクに保存し、PATHCOM を終了します。ホーム端末名が変更されている場合は、キュー・マネージャー用の QMINI ファイルを変更して PATHWAY 構成と一致させてください。必要に応じて、QMINI ファイル内の TCP/IP リスナー・ポート構成も変更します。
 - g. **strmqm** を使用してキュー・マネージャーを始動します。
 - h. **runmqsc** を使用してチャンネル構成を検査し、必要に応じて調整します。
 - i. 以前に実行していたものと同じチャンネルを使用しようとしている場合は、次のようにならなければ、リモート・キュー・マネージャーのチャンネル構成も変更しなければならないことがあります。バックアップ・システムを、同じホスト名の IP アドレスを使用するように再構成できる (TCP/IP チャンネルの場合) か、あるいは同じ SNAX/APC および ICE リソース名 (たとえば、プロセス名や LU 名など) を使用するように再構成できる (SNA チャンネルの場合)。
 - j. 災害時に 1 次サイト・チャンネルが実行していた場合は特に、チャンネル同期エラーまたはシーケンス・エラーに備えてください。RDF は、バ

状況サーバーとキュー・サーバーのリカバリーおよび再始動

ックアップ・サイトのデータベースが最新のものである (1 次との差異がない) かどうかを確認しないため、破壊的な災害の結果としてデータが失われることもあり得ます。この危険を最小限にするためには、RDF 構成がメッセージ・フローと関連したデータベース更新のボリュームを処理できるように構成されていることを確認してください。

第13章 構成ファイル

MQSeries for Compaq NSK は、構成ファイルを使用して、基本的な製品構成情報を保持します。この章では、構成ファイルとは何か、また、それらを用いてどのようにキュー・マネージャーの作動方法を変更できるのかについて説明します。この章には、次の節があります。

- 『構成ファイルとは』
- 244ページの『MQSeries 構成ファイル (MQSINI)』
- 245ページの『キュー・マネージャー構成ファイル (QMINI)』
- 254ページの『構成ファイルを編集する』

構成ファイルとは

構成ファイルは、個々のキュー・マネージャーとノード上の MQSeries に関するオプション値を全体として定義します。これらのファイルを、*ini* ファイルまたは *スタンザ・ファイル*と言います。構成ファイルには、1 つまたは複数のスタンザが含まれています。スタンザとはファイル内の行のグループで、それらの行は全体として 1 つの共通機能を持っているか、またはシステムの一部を定義します。たとえば、スタンザには、ログ、チャンネル、インストール可能サービスに関連付けられているものがあります。

構成ファイルは、ノード上のキュー・マネージャーの構成を変更するコマンドによって自動的に変更することも、または、手操作でそれらを編集することによって変更することもできます。ただし、通常では、キュー・マネージャーの実行中は、構成ファイルを手操作で変更しないでください。

構成ファイルには次の 2 つのタイプがあります。

- *MQSeries* 構成ファイル、MQSINI。これはノード上の MQSeries の値を全体として指定します。通常、1 つのノードにつき 1 つの MQSeries 構成ファイルがあります。
- キュー・マネージャー構成ファイル、QMINI。特定のキュー・マネージャーの値を指定します。ノード上の各キュー・マネージャーごとに、1 つのキュー・マネージャー構成ファイルがあります。

MQSeries 構成ファイル (MQSINI)

MQSeries 構成ファイル MQSINI は、MQSeries インストール・ノード上のすべてのキュー・マネージャーに関連する情報を含んでいます。これは、インストール時に自動的に作成されます。特に、MQSeries 構成ファイルは、各キュー・マネージャーに関連したデータを見つけるために使用されます。MQSeries 構成ファイルは、ZMQSSYS サブボリューム (デフォルトでは、`$SYSTEM.ZMQSSYS.MQSINI`) にあります。環境変数 `MQMACHINIFILE` は、MQSeries 構成ファイルがデフォルトの名前と位置を持たないシステム上で使用するために提供されています。

MQSeries 構成ファイルの内容

MQSINI ファイルには、インストール・システム全体のデフォルト、キュー・マネージャーの名前、デフォルト・キュー・マネージャーの名前、およびそれぞれのキュー・マネージャーに関連したファイルの位置が入っています。

MQSINI には、次のスタンザが表示されます。

AllQueueManagers

インストール・システム全体のファイル位置とボリュームの値を指定します。

DefaultQueueManager

インストール用のデフォルトのキュー・マネージャーを指定します。キュー・マネージャーの名前が明示的に指定されていない場合は、このデフォルトのキュー・マネージャーが `MQSC` コマンドを実行します。スタンザは、新規のデフォルト・キュー・マネージャーを作成すると、自動的に更新されます。誤ってデフォルト・キュー・マネージャーを作成した場合に、それを元に戻したいときは、このスタンザを手動で変更する必要があります。

QueueManager

このスタンザは、各キュー・マネージャーごとに 1 つずつあります。QueueManager スタンザは、キュー・マネージャーの名前と、そのキュー・マネージャーに関連したファイルの位置を指定します。これらのファイルの名前はキュー・マネージャー名に基づいていますが、キュー・マネージャー名が有効なファイル名でない場合は、変換されます。

245ページの図42に MQSINI ファイルの例を示します。

```

#*****#
## Module Name: MQSINI                                     ##
## Type:          MQSeries machine-wide ini file          ##
## Function:      Define configuration data for all queue managers ##
##               ##
#*****#
## Notes :                                               ##
## 1) This file defines configuration data for all queue managers ##
##               ##
#*****#
AllQueueManagers:
    MQSVolume=$DATA00                                     /Volume for the installation
    MQSExePath=$DATA00.ZMQSEXE                          /Location of product executables
    QMDefaultVolume=$DATA00                             /Default volume for queue manager creation
    ConvEBCDICNewline=NL_TO_LF                          /Data Conversion EBCDIC Newline
    NSKSegidRange=10-20                                  /Segment Id Range
QueueManager:
    Name=MT01                                             /A queue manager called MT01
    QMVolume=$DATA00                                     /Volume of the queue manager
    QMSubvolume=MT01                                     /Subvolume prefix for the queue manager
DefaultQueueManager:
    Name=MT01                                             /Name of the default queue manager (optional)

```

図 42. MQSeries 構成ファイル (MQSINI) の例。MQSINI ファイルは、ユーザーが提供するボリュームおよびサブボリューム情報を使用してインストール中に初期化されます。

注: MQSeries 構成ファイルは、キュー・マネージャーに関連したデータを見つけるために使用されるので、構成ファイルが存在しなかったり、正しくなかったりすると、一部または全部の MQSeries コマンドが失敗します。また、アプリケーションは、MQSeries 構成ファイルに定義されていないキュー・マネージャーには接続できません。

キュー・マネージャー構成ファイル (QMINI)

キュー・マネージャー構成ファイル QMINI は、特定のキュー・マネージャーに関係のある情報を含んでいます。各キュー・マネージャーごとに、1 つのキュー・マネージャー構成ファイルがあります。これは、関連したキュー・マネージャーが作成されるときに、自動的に作成されます。

このファイルは、キュー・マネージャーのサブボリュームに保持されます。たとえば、QMNAME という名前のキュー・マネージャーの構成ファイルのパスと名前は、\$VOLUME.QMNAME.QMINI です。

注: キュー・マネージャーの名前の長さは 48 文字までです。サブボリューム名は、キュー・マネージャー名に基づいて生成されます。このプロセスは名前変換と呼ばれ、名前が必ず有効で固有なものになります。

キュー・マネージャー構成ファイルの内容

キュー・マネージャー構成ファイル QMINI には、次のスタンザがあります。

Configuration

このスタンザは、キュー・マネージャーのグローバル構成を定義します。

変更できる項目は次のとおりです。

```
DefaultQueueServerName  
DefaultStatusServerName  
HomeTerminalName  
PathmonProcName  
EMSCollectorName  
MinIdleMCALU62Responders  
MinIdleMCATCPResponders  
MinIdleMCACallers  
MinIdleLQMAgents
```

これらの項目の詳細については、61ページの『キュー・マネージャーのプロパティーの変更』を参照してください。このスタンザ内の他の項目は、変更しないでください。

DefaultProcess

このスタンザは、MQSeries プロセスで使用されるデフォルト値を定義します。このスタンザ内の項目は、変更しないでください。

ECBoss

このスタンザは、MQSeries EC Boss プロセスの構成を定義します。ExpectedNumECs の項目は、このキュー・マネージャーの EC プロセスの数を定義します。この値は、キュー・マネージャーの PATHWAY 構成と一致している必要があります。詳細については、61ページの『キュー・マネージャーのプロパティーの変更』を参照してください。このスタンザ内の他の項目は、変更しないでください。

EC EC スタンザの MCAAgentPriority および LQMAgentPriority 項目は、エージェント・プロセスのプロセス優先順位を制御するものであり、変更が可能です。詳細については、61ページの『キュー・マネージャーのプロパティーの変更』を参照してください。このスタンザ内の他の項目は、変更しないでください。

次のスタanzasは、各 MQSeries プロセス・タイプごとの特定の操作パラメーターを定義します。通常、これらのパラメーターの値を変更する必要はありませんが、249ページの『キュー・マネージャー構成ファイルの例』を参照してください。

MCACaller**MCATCPResponder****MCALU62Responder****MQIServer****LQMAgent****ChannellInitiator****TCPListener****Authority**

キュー・マネージャーの OAM を有効化および無効化するための推奨メカニズムを提供しています。OAM を使用可能化または使用不能化するには、MQAUTH フラグを On または Off に設定すればよく、Service スタanzasや Service Component スタanzasを追加したり除去したりする必要はありません。

Service

インストール可能なサービスの名前、およびそのサービスへの入り口点の数を指定します。スタanzasは、各サービスごとに 1 つずつ存在します。次のサービスが提供されています。

- 許可サービス
- 名前サービス

Object Authority Manager (OAM) はデフォルトでは使用可能になっています。QMIMI には、デフォルトで、許可サービス・スタanzasとこれに関連した ServiceComponent スタanzasが存在します。

OAM を使用不可にするには、Authority スタanzasの MQAUTH フラグを Off に設定してから、キュー・マネージャーを再始動します。あるいは、次のようにすることもできます。

1. キュー・マネージャーを削除する (**dltmqm** コマンドを使用)。
2. MQSNOAUT 環境変数を設定して再度キュー・マネージャーを作成する (**crtmqm** コマンドを使用)。
3. QMIMI から許可サービス・スタanzasを削除する。

キュー・マネージャー構成ファイル

提供されている名前サービスを使用可能にするには、名前サービス・スタンザを QMINI に手動で追加する必要があります。

ServiceComponent

これらのスタンザは、特定のサービスに関連したサービス・コンポーネントを定義します。各サービスごとに複数のサービス・コンポーネント・スタンザがあってもかまいませんが、各サービス・コンポーネント・スタンザはそれぞれ対応するサービス・スタンザに一致している必要があります。詳細については、*MQSeries* プログラム式システム管理を参照してください。

TuningParameters

このスタンザは、ローカル・キュー・マネージャー・エージェントが使用する内部調整パラメーターを定義します。これらの値は変更しないでください。

Channels

このスタンザにはチャンネルに関する情報が入っています。キュー・マネージャーに対して定義することのできるチャンネルの最大数 (MaxChannels) を定義することに加えて、2 番目の項目 (MaxActiveChannels) では、同時にアクティブにすることのできるチャンネルの数を制限します。MaxActiveChannels は、MaxChannels より大きくなってはなりません。また、チャンネル・スタンザには、チャンネル・イニシエーターのパフォーマンスの調整に使用できる項目 (ChanInitDiscInterval) が含まれています。これらの項目の詳細については、61ページの『キュー・マネージャーのプロパティーの変更』を参照してください。このスタンザ内の他の項目は、変更しないでください。

チャンネルの詳細については、*MQSeries* 相互通信 を参照してください。

TCPConfig

ネットワーク・プロトコル構成パラメーターを指定します。これらのスタンザは、チャンネルのデフォルト・パラメーターを指定変更します。変更されたデフォルト値を示すスタンザだけが、実際に存在しています。

パラメーター MQLISTENPORTNUM がリスナー・プロセスの環境に存在している場合は、TCPListenerPort 値がリスナー・プログラムによって指定変更されます。

詳細については、*MQSeries* 相互通信 を参照してください。

TCPPort、TCPNumListenerPorts、および TCPListenerPort の項目の変更方法については、61ページの『キュー・マネージャーのプロパティーの変更』を参照してください。

キュー・マネージャー構成ファイルの例

図43 は、キュー・マネージャー構成ファイル (QMINI) のサンプルです。

```

*****#
** Module Name: QMINI                               **
** Type      : MQSeries queue manager configuration file **
** Function  : Define the configuration of a single queue manager **
**          : **
**          : **
*****#
** Notes : **
** 1) This file defines the configuration of the queue manager **
**          : **
*****#
Configuration:
  PathmonProcName=$p01p
  DefaultStatusServerName=$p01s
  ServerClassName=MQS-ECBOSS
  EMSCollectorName=$0
  HomeTerminalName=$ZTN0.#PTY001C
  ShutdownFileName=SHUTDOWN
  TraceOptionsFileName=TRACEOPT
  RuntimeFileName=RUNTIME
  StatableFileName=STATABLE
  ChannelDefFileName=CHDEFS
  DefaultCCSID=819
  DefaultTraceOptions=0
  MaxIdleAgents=10
  MinIdleMCALU62Responders=0
  MinIdleMCATCPResponders=0
  MinIdleMCAcallers=0
  MinIdleLQMAgents=1
  MaxIdleAgentReuse=10
DefaultProcess:
  ExeFileName=DEFAULT
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=256000
  IniPoolSize=256000
  Priority=175

```

図43. キュー・マネージャー構成ファイル (QMINI) の例 (1/6)

キュー・マネージャー構成ファイル

```
ECBoss:
  ExeFileName=MQECBOSS
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=300000
  IniPoolSize=256000
  Priority=175
  ExpectedNumECs=1
EC:
  ExeFileName=MQEC
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=256000
  IniPoolSize=256000
  Priority=175
  LQMAgentExe=MLQMAG
  MCACallerExe=MQMCACAL
  MCATCPResponderExe=MQTCPRES
  MCALU62ResponderExe=MLU6RES
  MCAAgentPriority=165
  LQMAgentPriority=165
  StopProcessTimer=3000
  IdleProcessTimer=3000
```

図 43. キュー・マネージャー構成ファイル (QMINI) の例 (2/6)

```
MCACaller:
  ExeFileName=MQMCACAL
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=300000
  IniPoolSize=256000
  Priority=175
MCATCPResponder:
  ExeFileName=MQTCPRES
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=300000
  IniPoolSize=256000
  Priority=175
MCALU62Responder:
  ExeFileName=MLU6RES
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=300000
  IniPoolSize=256000
  Priority=175
```

図 43. キュー・マネージャー構成ファイル (QMINI) の例 (3/6)

キュー・マネージャー構成ファイル

```
MQIServer:
  ExeFileName=MQMQISER
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=300000
  IniPoolSize=256000
  Priority=175
LQMAgent:
  ExeFileName=MLQLMAG
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=50
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=120000
  IniPoolSize=200000
  Priority=175
ChannelInitiator:
  ExeFileName=RUNMQCHI
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=256000
  IniPoolSize=256000
  Priority=175
TCPLListener:
  ExeFileName=RUNMQLSR
  TraceVolSubvol=$DATA1.p101L
  TracePrefix=TR
  ErrorVolSubvol=$DATA1.p101L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=256000
  IniPoolSize=256000
  Priority=175
```

図 43. キュー・マネージャー構成ファイル (QMINI) の例 (4/6)

```
Queue Manager Server:
  ExeFileName=MQMGRSVR
  TraceVolSubvol=$DATA01.MV1L
  TracePrefix=TR
  ErrorVolSubvol=$DATA01.MV1L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=300000
  IniPoolSize=256000
  Priority=175
Repository Server:
  ExeFileName=MQREPSVR
  TraceVolSubvol=$DATA01.MV1L
  TracePrefix=TR
  ErrorVolSubvol=$DATA01.MV1L
  ErrorPrefix=ER
  DebugMode=0
  IPCCTimeOut=10000
  IPCCMemSetSize=32000
  MemSetSize=16000
  ExtPoolSize=256000
  IniPoolSize=256000
  Priority=175
```

図 43. キュー・マネージャー構成ファイル (QMINI) の例 (5/6)

構成ファイルの編集

```
Authority:
  MQAUTH=On
Service:
  Service=AuthorizationService
  EntryPoints=9
ServiceComponent:
  Service=AuthorizationService
  Name=MQSeries.Compaq.auth.service
  Module=MQOAM
  ComponentDataSize=0
  ComponentID=0
TuningParameters:
  KernelMemSetSize=32000
  ObjCatMemSetSize=32000
  QueueMemSetSize=16000
  MQGETActiveQPoll=50
  MQGETInactiveQPoll=1000
Channels:
  RetryAll=1
  MaxChannels=10
  MaxActiveChannels=10
  MaxTries=3
  MaxTriesInterval=10
  ChanInitDiscInterval=10
  AdoptNewMCA=NO
  AdoptNewMCATimeout=60
  AdoptNewMCACheck=NAME,ADDRESS,QM
TCPConfig:
  TCPPort=1414
  TCPNumListenerPorts=1
  TCPListenerPort=1414
  TCPKeepAlive=1
```

図 43. キュー・マネージャー構成ファイル (QMINI) の例 (6/6)

構成ファイルを編集する

システム・デフォルトを変更するには、デフォルト構成ファイルを編集します。ただし、構成ファイルを編集する前に、必要に応じて復元できるように必ずバックアップを行い、関係するキュー・マネージャーが停止していることを確認してください。

構成ファイルは、たとえば次のような場合に編集する必要があります。

- 構成ファイルが失われた場合 (可能であれば、バックアップから回復します)。
- CPU 上でのキュー・マネージャーの分散を変更する必要がある場合。
- デフォルトのキュー・マネージャーを変更する必要がある場合 (たとえば、既存のキュー・マネージャーを誤って削除してしまった場合)。
- S.E. から編集するよう勧められた場合。

詳細については、61ページの『キュー・マネージャーのプロパティの変更』を参照してください。

構成ファイルへの変更のインプリメント

構成ファイルを編集する場合は、キュー・マネージャーにより変更が即時にインプリメントされることはありません。MQSeries 構成ファイル (MQSINI) に対する変更は、MQSeries が作成または開始されたときにのみ有効になります。キュー・マネージャー構成ファイル (QMINI) に対する変更は、キュー・マネージャーが開始されたときに有効になります。変更の際にキュー・マネージャーが実行している場合には、この変更をシステムが認識するように、このキュー・マネージャーを停止してから再始動しなければなりません。

構成ファイルに関する推奨事項

新規のキュー・マネージャーを作成するときは、必ず次の操作をしてください。

- MQSeries 構成ファイル (MQSINI) をバックアップする
- 新規のキュー・マネージャー構成ファイル (QMINI) をバックアップする

第14章 問題判別

この章では、MQSeries for Compaq NSK のトラブルシューティング情報について説明します。問題を判別するには、症状をリストして、それらをトレースして原因を突き止めてください。

ハードウェアの制限により生じるパフォーマンス上の問題は、即時に解決できません。問題が MQSeries コードにあると考えられる場合は、IBM サポート・センターに連絡してください。この章には、次の節があります。

- 『予備的な検査の作成』
- 263ページの『共通プログラミング・エラー』
- 263ページの『次に何を行うか』
- 269ページの『アプリケーション設計の考慮事項』
- 269ページの『メッセージ長の影響』
- 275ページの『エラー・ログ』
- 279ページの『送達不能キュー』
- 279ページの『構成ファイルと問題判別』
- 280ページの『MQSeries トレースの使用』
- 281ページの『第 1 障害サポート・テクノロジー™ (FFST)』

予備的な検査の作成

問題の原因が、次のものにある場合があります。

- MQSeries
- ネットワーク
- アプリケーション
- Compaq システム・ソフトウェア

以下の節では、考慮できる質問が示されています。質問に答えて、問題と関係がありそうな点を書き留めてください。

以前に MQSeries は正常に実行されたか

MQSeries が以前に正常に実行されなかった場合、正しくセットアップができていなかったのかもしれませんが、*MQSeries for Compaq NSK* スタートアップ・ガイドを参照して、すべてのステップを正しく行ったか確認してください。

エラー・メッセージが出されたか

MQSeries は、エラー・ログを使用して、MQSeries 自体の操作、ユーザーが開始したキュー・マネージャー、および使用中のチャネルからのエラー・データに関係したメッセージを取り込みます。問題に関連したメッセージが記録されていないかどうか、エラー・ログを検査してください。

エラー・ログの内容と位置については、275ページの『エラー・ログ』を参照してください。

問題について説明した戻りコードが戻されていないか

メッセージ・キュー・インターフェース (MQI) 呼び出しが失敗したことを示す戻りコードをアプリケーションが受け取っている場合、MQSeries アプリケーション・プログラミング・リファレンス で、戻りコードの説明を参照してください。

問題を再現できるか

問題を再現できる場合は、以下の質問を考慮してください。

- 問題は、コマンドまたはそれと同等の管理要求によって引き起こされますか。
別の方法で入力した場合に、操作はうまくいきますか。コマンド行から入力した場合にはコマンドは作動するが、コマンド行以外からの場合には作動しない場合、コマンド・サーバーが停止していないかどうか検査してください。またキュー定義の `SYSTEM.ADMIN.COMMAND.QUEUE` が変更されていないかどうか検査してください。
- 問題はプログラムによって引き起こされますか。
- 問題が生じる時にいつもシステムで実行されていると思われるアプリケーションを識別することができますか。識別できる場合、そのアプリケーションにエラーがないかどうか調べてください。
- キュー・マネージャー・データベースが存在するボリュームは、TM/MP 保護されていますか。 `TMFCOM; STATUS DATAVOLS` を発行して、ボリュームが TM/MP 保護されているか確認してください。
- 409ページの『付録C. MQSeries for Compaq NSK 用の TAACL 環境の設定』で説明されている必須 TAACL 環境パラメーターは追加されていますか。TAACL プロンプトで `PARAM` を発行して、現在設定されているパラメーターを表示してください。
- TM/MP 監査証跡は、負荷を処理できるようにサイズ変更されていますか。(231ページの『第11章 トランザクション・サポートとメッセージングについて』)

ての理解』を参照してください。) TMFCOM; INFO AUDITTRAIL を発行して、監査証跡構成を表示してください。

- Compaq ファイル・セキュリティーは、qmD データベース・ファイルへのアクセスを許可していますか。MQM グループ外のユーザーがファイルにアクセスできるようにするには、最低限の読み取りアクセスが必要です。Compaq ファイル・セキュリティー属性がアクセスを除外すると、MQCONN 要求に対して戻りコード 2035 (MQRC_NOT_AUTHORIZED) が戻されま

す。

最後に正常に実行された後、変更が行われたか

最近行われた可能性のある変更について考慮する場合は、MQSeries システムについて考えてください。また、MQSeries システムがインターフェースする他のプログラムや、ハードウェアおよび新しいアプリケーションについても考えてください。さらに、ユーザーの気付いていない新しいアプリケーションがシステムで実行されている可能性はないかどうかも考慮してください。

- キュー定義を変更、追加、または削除しましたか。
- チャネル定義を変更または追加しましたか。変更は、MQSeries チャネル定義、またはアプリケーションが必要とする基本的な通信定義に行われた可能性があります。
- アプリケーションは、ユーザーが行った変更の結果としてアプリケーションが受け取る可能性のある戻りコードを処理しますか。
- MQSeries PATHWAY に変更が加えられましたか。
- MQSeries インストール・ファイルに何らかの変更 (たとえば、ファイル・セキュリティーの変更など) が加えられましたか。
- MQSeries Compaq NSK データベース・ファイルに何らかの変更 (たとえば、ファイル・セキュリティーの変更、TM/MP 監査の更新など) が加えられましたか。
- キュー・マネージャー QMINI ファイルは、変更を適用しましたか。

アプリケーションは以前に正常に実行されたか

ある特定のアプリケーションが関係する問題であると思われる場合には、そのアプリケーションが以前に正常に実行されたかどうか考えてください。

以下の質問を考慮してください。

- そのアプリケーションが最後に正常に実行された後、そのアプリケーションに何らかの変更が行われましたか。

予備的な検査

変更が加えられた場合、アプリケーションの新規または変更部分にエラーが存在する可能性があります。変更部分を調べて、問題の明白な理由があるかどうか見てください。アプリケーションの前のバージョンを使用して再試行できますか。

- アプリケーションのすべての機能が以前に完全に動作したか。

以前に呼び出されたことのないアプリケーションの一部が初めて使用されたときに問題が起きたということはありませんか。もしそうなら、アプリケーションのその部分にエラーがある可能性があります。障害が起きたときにアプリケーションが何を実行していたかを分析し、プログラムのその部分のソース・コードにエラーがないかどうか検査してください。

以前はプログラムが正常に実行されたという場合は、エラーが起きたときに処理されていた現行のキューの状態およびファイルを検査してください。プログラムでめったに使用されないパスを呼び出すようなごくまれなデータ値が、それに含まれている可能性があります。

- アプリケーションはすべての戻りコードを検査しますか。

MQSeries システムは、変更されている可能性があります。その変更の結果出される戻りコードをアプリケーションは検査しません。たとえば、アプリケーションは自分がアクセスするキューは共用できると想定しています。もしキューが排他的として再定義された場合には、そのアプリケーションは、そのキューがアクセス不能になったことを示す戻りコードを処理できませんか。

- アプリケーションは他の MQSeries システムで実行されますか。

この MQSeries システムのセットアップの仕方に異なった点があり、それが問題を引き起こしている可能性がありますか。たとえば、キューは同じメッセージ長または優先順位で定義されていますか。

- PARAM SAVE-ENVIRONMENT ON を設定しましたか。

そうしていない場合、MQCONN 呼び出しで MQRC 2058 を受け取ります。409ページの『付録C. MQSeries for Compaq NSK 用の TAQL 環境の設定』の説明に従って、アプリケーション環境の PARAM を設定してください。

アプリケーションが以前に正常に実行されなかった場合

アプリケーションがこれまでに正常に実行されたことがない場合は、そのアプリケーションにエラーがないかどうか注意深く調べる必要があります。

コードを調べる前に、コードに使用されているプログラム言語に応じて、エラーが報告されていないかどうか、変換プログラムからの出力、またはコンパイラとリンケージ・エディター（該当する場合）からの出力を調べてください。

アプリケーションが、ロード・ライブラリーに変換、コンパイル、またはリンク・エディットされない場合には、そのアプリケーションは実行できません。アプリケーションの作成の詳細については、*MQSeries アプリケーション・プログラミング・リファレンス* を参照してください。

出力文書が、それぞれのステップがエラーなしで行われたことを示している場合は、アプリケーションのコーディング論理を検討する必要があります。問題の症状は、機能の障害を示しており、その原因がコード内のエラーである場合があります。MQSeries アプリケーションで問題の原因となる共通エラーの例については、263ページの『共通プログラミング・エラー』を参照してください。

問題がネットワークの特定部分に影響するか

問題の影響を受けるネットワークの特定の部分（たとえば、リモート・キュー）を識別できる場合があります。リモート・メッセージ・キュー・マネージャーへのリンクが働いていない場合、メッセージはリモート・キューに送ることができません。

2 つのシステム間の接続が使用可能かどうか、また MQSeries の相互通信コンポーネントが始動しているかどうかを検査してください。

メッセージが伝送キューに到達しているかどうかを検査し、伝送キューのローカル・キュー定義およびリモート・キューを検査してください。

ネットワーク関連の変更を行ったり、問題の理由の説明となる可能性がある MQSeries 定義を変更しましたか。

MQSeries PATHWAY TCP リスナー・サーバーの MQS-TCPLISxx 始動しているか確認してください。サーバーが始動できていない場合、SCF（たとえば、SCF; STATUS PROCESS \$ZTC0）を使用して、終了中の接続（たとえば、FIN-WAIT 状態）がないかポートを確認してください。リモート・パートナーは、停止してから再始動して、ポートを解放する必要があります。SNA の場合、MQSeries SNA リスナーが SNAX/APC または ICE PATHWAY 用に構成されているか確認するには、472ページの『LU 6.2 応答側プロセス』を参照してください。

チャンネル用の接続名に、TCPIP プロセス名、ホスト名または IP アドレス、およびポートが正しく使用されているか、またリモート・チャンネル定義と一致しているか確認してください。

1 日の特定の時刻に問題が起こるか

1 日の特定の時刻に問題が起こる場合は、問題にシステム負荷が関与している可能性があります。普通、システム負荷のピークは、午前中ごろと午後の中ごろです。そのため、この時間帯に負荷に依存する問題が起こる可能性が最も高くなります (MQSeries ネットワークが複数の時間帯にまたがっている場合には、システム負荷のピークは、1 日の別の時間帯に起こる可能性があります)。

問題は断続的に起こるか

断続的な問題は、プロセスが互いに独立して実行できるということを考慮に入れておかなかったことが原因となっている可能性があります。たとえば、プログラムは、前のプロセスが完了する前に、待機オプションを指定せずに MQGET 呼び出しを出すことができます。メッセージを書き込む呼び出しが未確定であるとき (つまり、コミットされるかバックアウトされる前) に、アプリケーションがキューからメッセージを読み取ろうとする場合にも、断続的な問題が起こる可能性があります。

時折、ENDMQM の実行中に PATHWAY エラーが記録されることがあります。キュー・マネージャーのシャットダウン時に、エラー *1018* SERVER FILE (6006) または *1018* SERVER FILE (7006) が表示されることがあります。これらは、MQS-QMGRSVR00、MQS-STATUS00、または MQS-QUEUE00 のそれぞれのサーバーの終了中に予期されます。

サービス更新を適用したか

サービス更新が MQSeries に適用された場合、更新処置が正常に完了したかどうか、またエラー・メッセージが生成されていないかどうかを検査してください。

- 更新には、特別の指示がありませんでしたか。
- 更新が正しく完全に適用されたことを確認するためのテストが行われましたか。
- MQSeries が前のサービス・レベルに復元されても、まだ問題が解消しませんか。
- インストールが正常に行われている場合、パッチ・エラーがないかどうか、IBM の S.E. に問い合わせてください。
- パッチが他のプログラムに適用されている場合には、MQSeries がそのプログラムとインターフェースをとる際の影響を考慮してください。
- サービス・レベルを検査してください。ZMQSSYS.MEMOPTF 読み取り専用を編集してください。CSD HISTORY の項目 A) を書き留めてください。TACL プロンプトで、VPROC \$vol.ZMQSLIB.MQSRLLIB と入力して

ください。\$vol は、MQ のインストール・ボリュームです。これは、MEMOPTF の VPROC 情報と一致するはずです。あるいは、efix が適用された場合は、これよりも新しくなっている可能性もあります。更新された VPROC 情報については、efix で提供されている MEMOEFIX ファイルを参照してください。

共通プログラミング・エラー

次のリストのエラーでは、MQSeries プログラムの実行中に検出される問題に共通する原因の大部分が示されています。MQSeries システムの問題が、次のエラーの 1 つまたは複数によって引き起こされる可能性があることを考慮する必要があります。

- キューが共用可能であると想定しているが、実際は排他的である場合
- MQI 呼び出しで誤ったパラメーターを渡す場合
- MQI 呼び出しで不十分なパラメーターを渡す場合。これは、MQI が、アプリケーションが処理できるような完了コードおよび理由コードをセットアップできないことを意味します。
- MQI 要求からの戻りコードを検査できない場合
- 誤った長さを指定した変数を渡す場合
- 間違った順序でパラメーターを渡す場合
- *MsgId* および *CorrelId* を正しく初期化しなかった場合
- MQPUT コマンドで MQPMO_SYNCPOINT が指定されたときに、BEGINTRANSACTION を発行できなかった場合

コマンドに関する問題

コマンドの説明テキストで円記号 (¥) や二重引用符 (") などの特殊文字を使用する場合は、注意する必要があります。説明テキストでこの 2 つの文字のいずれかを使用する場合、その前に ¥ を付けてください。つまり、テキストで ¥ または " を入力したい場合は、¥¥ または ¥" と入力します。

次に何を行うか

システムに変更が行われていないこと、アプリケーション・プログラムに問題がないことを確認したら、問題の症状を最も的確に表している項目を選んでください。

- 264ページの『不正な出力を得たか』
- 264ページの『PCF コマンドからの応答を受け取れなかったか』
- 267ページの『問題はリモート・キューにのみ影響するか』

次に

- 267ページの『アプリケーションまたは MQSeries for Compaq NSK の実行速度は低速ですか。』

不正な出力を得たか

本書で「不正出力」とは、次のことを指します。

- アプリケーションが、予期していたメッセージを受け取らないこと
- アプリケーションが、予期しない情報または破壊された情報を含んでいるメッセージを受け取ること
- アプリケーションが、予期していなかったメッセージ（たとえば、別のアプリケーションに向けられたメッセージ）を受け取ること
- 情報を表示するために使用している構造は正確ですか。たとえば、それは送達不能キュー上にあるメッセージの、メッセージ・テキストの先頭に追加された送達不能キュー・ヘッダーを表示するために使用する MQDLH 構造ですか。
- それは、異なるオペレーティング・システム (OS) レベル、または同じオペレーティング・システム・レベルの別のマシンで MQSeries をインストールするときに再現することができますか。再現できない場合、Compaq からの Interim Program Maintenance (IPM) で修正される OS エラーがある可能性があります。それを適用する必要があります。Compaq サポートに確認してください。

いずれの場合にも、アプリケーションが使用しているキューまたはキュー・マネージャーの別名が正しく指定されているかどうか検査し、ネットワークに対して行われた変更を配慮してください。

MQSeries エラー・メッセージ（すべて先頭に「AMQ」の文字が付いている）が生成されたら、エラー・ログを見てください。詳細については、275ページの『エラー・ログ』を参照してください。

PCF コマンドからの応答を受け取れなかったか

コマンドを出したが応答を受け取れなかった場合には、次の質問を考慮してください。

- コマンド・サーバーが実行されていますか。
dspmqcsv コマンドを使用して、コマンド・サーバーの状況を検査してください。このコマンドに対する応答が、コマンド・サーバーが実行されていないことを示したら、**strmqcsv** コマンドを使用してそれを開始してください。コマンドに対する応答が、MQGET 要求に対して

SYSTEM.ADMIN.COMMAND.QUEUE が使用できないことを示したなら、そのキューを MQGET 要求に対して使用できるようにしてください。

- 応答は送達不能キューへ送られましたか。
送達不能キューのヘッダー構造は、問題を説明する理由またはフィードバック・コードを含んでいます。送達不能キューのヘッダー構造 (MQDLH) については、*MQSeries アプリケーション・プログラミング・リファレンス* を参照してください。
送達不能キューにメッセージが入っている場合は、提供されているブラウザのサンプル・アプリケーション (AMQSBCG) を使用し、MQGET 呼び出しを用いてメッセージをブラウザできます。サンプル・アプリケーションは、命名されたキュー・マネージャーの指定されたキューのすべてのメッセージを処理し、指定されたキューのすべてのメッセージのメッセージ記述子フィールドとメッセージ・コンテキスト・フィールドの両方を表示します。
- メッセージはエラー・ログに送られましたか。
詳細については、275ページの『エラー・ログ』を参照してください。
- キューは、読み書き操作に対して使用可能ですか。
- *WaitInterval* の長さは充分ですか。
MQGET 呼び出しがタイムアウトの場合は、完了コード MQCC_FAILED および理由コード MQRC_NO_MSG_AVAILABLE が戻されます。
(*WaitInterval* フィールドについて、および MQGET からの完了コードと理由コードについては、*MQSeries アプリケーション・プログラミング・リファレンス* を参照してください。)
- ユーザー自身のアプリケーション・プログラムを使用して、コマンドを SYSTEM.ADMIN.COMMAND.QUEUE に入れる場合には、トランザクションをコミットする必要がありますか。
要求メッセージを同期点から特別に除外しない限り、応答メッセージを受け取ろうとする前にトランザクションをコミットする必要があります。
- キューの MAXDEPTH 属性および MAXMSGL 属性は、十分高い値に設定されていますか。
- *CorrelId* フィールドおよび *MsgId* フィールドを正しく使用していますか。
キューからのすべてのメッセージを受け取ることができるように、アプリケーションで *MsgId* および *CorrelId* の値を設定してください。

コマンド・サーバーを停止し、その後再始動するよう試みて、生成されたエラー・メッセージに回答してください。

次に

それでもシステムが応答しなければ、問題は、キュー・マネージャーに関係している可能性があります。キュー・マネージャーを停止した後、再始動を試みてください。再始動後にもまだ問題が起きる場合は、IBM サポート・センターに連絡してください。

キューの一部が障害を起こしているか

キューのサブセットのみで問題が起きている疑いがある場合には、疑いのあるローカル・キューに問題があるかどうか検査してください。

1. 各キューについての情報を表示します。MQSC コマンド `DISPLAY QUEUE` を使用すればこの情報を表示できます。
2. 表示されたデータを使用して、次の検査を行います。
 - `CURDEPTH` が `MAXDEPTH` になっている場合、それは、そのキューが処理されていないことを示します。すべてのアプリケーションが正常に実行されているかどうか検査してください。
 - `CURDEPTH` が `MAXDEPTH` になっていない場合、次のキュー属性を検査してそれらが正しいかどうか確認してください。
 - トリガー操作が使用されている場合
 - トリガー・モニターは実行されていますか。
 - トリガーのサイズが大き過ぎませんか。つまり、トリガー操作によってトリガー・イベントは十分な頻度で生成されますか。
 - プロセス名は正しいですか。
 - プロセスは入手でき、操作可能ですか。
 - キューは共用可能ですか。共用可能でなければ、別のアプリケーションがすでにそのキューを入力のためにオープンしている可能性があります。
 - キューは、読み取り (GET) および書き込み (PUT) が適切に行えるようになっていますか。
 - キューからメッセージを入手するアプリケーション・プロセスがない場合、その理由を判別してください。アプリケーションを開始する必要がある、接続が中断している、あるいは `MQOPEN` 呼び出しが何らかの理由で障害を起こしているなどの可能性があります。

キュー属性の `IPPROCS` および `OPPROCS` を検査してください。これらの属性は、キューが入力および出力のためにオープンされているかどうかを示します。値がゼロの場合、該当するタイプの操作は行われなことを示します。それらの値は変更されている可能性があること、およびオープンされていたキューがクローズされてしまうことに注意してください。

メッセージの書き込みまたは読み取りを予期している時の状況を検査する必要があります。

問題を解決できない場合、IBM サポート・センターに連絡してください。

問題はリモート・キューにのみ影響するか

問題がリモート・キューにのみ影響する場合には、次の検査を行ってください。

- 必要とされるチャンネルは開始されているかどうか、トリガーが可能かどうか、および必要とされる開始プログラムが実行されているかどうかを検査します。
- リモート・キューにメッセージを書き込む必要のあるプログラムが問題を報告していないかを検査します。
- トリガー操作を使用して分散キューイング・プロセスを開始する場合、伝送キューのトリガー操作がオンに設定されているかを検査します。また、トリガー・モニターが実行しているかを検査します。
- チャンネル・エラーや問題を示すようなメッセージがないかエラー・ログを検査します。
- 必要ならば、チャンネルを手動で開始します。この方法については、*MQSeries 相互通信* を参照してください。

チャンネルの定義方法については、471ページの『付録M. 通信のセットアップ』および *MQSeries 相互通信* を参照してください。

アプリケーションまたは MQSeries for Compaq NSK の実行速度は低速ですか。

アプリケーションの実行速度が低速の場合、これはループに入っているか、または利用できないリソースを待機していることを示しています。

strmqtrc か、MQMC キュー・マネージャー・パネルのいずれかを使用して、MQSeries トレースが使用可能になっていますか。これは、パフォーマンスの低下を引き起こします。qmL サブボリュームのすべての TR ファイルがオープンしているか確認してください。**endmqtrc** か MQMC キュー・マネージャー・パネルのいずれかを使用して、トレースを使用不可にしてください。

これは、パフォーマンス上の問題によって生じる可能性もあります。システムが、能力の限界付近で作動しているのかもしれませんが。この種の問題は、システム負荷のピーク時 (通常は、午前の中ごろおよび午後の中ごろ) に最悪の状

次に

態になることが予想されます。(ネットワークが複数の時間帯にまたがっている場合には、システム負荷のピークは、別の時間帯に起こる可能性があります)。

STATUS コマンドを使用して、アプリケーションおよびキュー・マネージャーのプロセスの優先順位を調べてください。ループが原因で、NSK によってプロセスの優先順位が下げられ、次第にゼロになってしまいます。

NSK システムの各 CPU が完全に利用されているか確認してください。少ししか負荷がかかっていないプロセッサがある場合、NSK システムの平衡化を計る必要があります。別のプロセッサに EC を追加して MQSeries ワークロードの分散することを考慮してください。

パフォーマンス上の問題は、ハードウェアの制限により生じる場合があります。

注: MQSeries の新規インストールまたは Compaq NSK システムのコールド・ロードの後最初に MQSeries 実行ファイルを呼び出すとき、予想よりも実行に長い時間がかかる場合があります。これは、Compaq NSK オペレーティング・システムが、『修正』フェーズを経るからです。このフェーズで、すべての外部宣言が確実に解決されます。

パフォーマンスの低下にシステム負荷は関係ないことが分かり、しかしシステムの負荷が軽い時にパフォーマンスの低下が生じる場合は、おそらくアプリケーション・プログラムの設計の悪さに原因があります。これは、特定のキューにアクセスされた場合にのみ起きる問題として、それそのものをはっきり示すことができます。

以下の症状は、MQSeries の実行が低速であることを示すものです。

- システムの MQSeries コマンドに対する反応が低速である。
- キュー項目数が繰り返し表示される場合、これは、大量のキュー・アクティビティが予期されるアプリケーションで、キューが低速に処理されていることを示しています。

上記の可能性のある原因を検討してもシステムのパフォーマンスが低下する場合、問題は MQSeries for Compaq NSK そのものにあるのかもしれませんが。その疑いがある場合、IBM サポート・センターに連絡し、援助を求める必要があります。

アプリケーション設計の考慮事項

設計の悪いプログラムは、いろいろな方法でパフォーマンスに影響を与える可能性があります。どのような方法で影響するかを見つけるのは難しい場合があります。そのプログラムはうまく実行されているように見えても、他のタスクのパフォーマンスに影響していることがあるためです。MQSeries 呼び出しを行うプログラムに特有のいくつかの問題を、以降の各節で解説します。

アプリケーション設計の詳細については、*MQSeries アプリケーション・プログラミング・ガイド* を参照してください。

メッセージ長の影響

MQSeries では、100 MB のデータを保持できるようになっていますが、メッセージ内のデータの量はメッセージを処理するアプリケーションのパフォーマンスに影響します。アプリケーションの最高のパフォーマンスを達成するためには、メッセージ内の重要なデータのみを送信する必要があります。たとえば、銀行預金口座の借方勘定に記入する要求では、クライアントからサーバー・アプリケーションに渡す必要のある情報は、口座番号と借方の金額だけです。

特定のメッセージの検索

MQGET 呼び出しでは、通常、キューの最初のメッセージが取り出されます。しかし、メッセージ記述子内のメッセージ ID と相関 ID (MsgId および CorrelId) を使用して特定のメッセージを指定すると、キュー・マネージャーは、そのメッセージを見つけるまでキューを検索しなければなりません。このような方法で MQGET 呼び出しを使用すると、アプリケーションのパフォーマンスに影響します。

長さが異なるメッセージを含んでいるキュー

キュー上のメッセージの長さが様々である場合、メッセージのサイズを決めるために、アプリケーションでは *BufferLength* フィールドをゼロに設定した MQGET 呼び出しを使用できます。このようにすると、その呼び出しは失敗するが、メッセージ・データのサイズが戻されます。その後、アプリケーションは最初の呼び出しで調べたメッセージの ID と正しいサイズのバッファーを指定して、呼び出しをもう一度行うことができます。ただし、同じキューをサービスしている別のアプリケーションがある場合には、ユーザーのアプリケーションのパフォーマンスは低下します。2 番目に出す MQGET 呼び出しは、最初の呼び出しと 2 番目の呼び出しの間に別のアプリケーションによって取り出されたメッセージを見つけるために時間を費やしてしまうからです。

アプリケーション設計の考慮事項

アプリケーションが固定長のメッセージを使用できない場合、この問題の別の解決方法は、MQINQ 呼び出しを使用して、キューが受け入れることのできるメッセージの最大サイズを調べ、その後 MQGET 呼び出しでその値を使用することです。キューのメッセージの最大サイズは、キューの *MaxMsgLength* 属性に保管されています。ただし、このキュー属性の値は 100 MB (MQSeries for Compaq NSK で許容されている最大値) という大きな値である場合があるので、この方法は大きなストレージを使用する可能性があります。

同期点の頻度

1 つの同期点内で多数の MQPUT 呼び出しをコミットなしで出すプログラムは、パフォーマンス効率の問題を起こす可能性があります。影響を受けるキューは、現在アクセス不能なメッセージで満杯になり、他のタスクはそれらのメッセージを読み取るために待機することがあります。これには、ストレージの観点、TMF 監査証跡の使用の観点、およびメッセージを読み取ろうとしているタスクと関連付けられているプロセスの観点から、考慮すべき点があります。

MQPUT1 呼び出しの使用

キューに入れるメッセージが 1 つしかない場合、MQPUT1 呼び出しを使用します。複数のメッセージを書き込みたい場合は、MQOPEN 呼び出しと、その後続けて一連の MQPUT 呼び出しおよび 1 つの MQCLOSE 呼び出しを使用します。

不正出力

「不正出力」という用語は、見方により異なった解釈ができます。本書で問題判別するため、その意味は 264 ページの『不正な出力を得たか』で説明しています。

この節では、次の 2 つのタイプの不正な出力について解説します。

- 予期していたときに現れないメッセージ
- 不正な情報または破壊された情報を含んでいるメッセージ

アプリケーションが分散キューの使用を含んでいる場合に起こる可能性のある別の問題についても解説します。

キューにメッセージが現れない場合

予期していたときにメッセージが現れない場合、次の点を検査してください。

- メッセージはキューに正常に書き込まれましたか。

- キューは正しく定義されていますか。たとえば、MAXMSGL の大きさは充分ですか。
- キューは書き込みが行えるようになっていませんか。
- キューが満杯になっていませんか。満杯になっていると、アプリケーションは必要なメッセージをキューに書き込めません。
- メッセージをキューから読み取ることができますか。
- 同期点をとる必要がありますか。

同期点内でメッセージが書き込まれたり取り出されたりしている場合、リカバリ単位がコミットされるまで他のタスクはそれらのメッセージを使用できません。

- 待機間隔の長さは十分ですか。
待機間隔は、MQGET のオプションとして設定できます。応答を待つ時間を十分に長くする必要があります。
- メッセージ識別子または相関 ID (*MsgId* または *CorrelId*) で識別された特定のメッセージを待っていますか。

正しい *MsgId* または *CorrelId* のメッセージを待っているかどうか検査してください。正常な MQGET 呼び出しでは、これらの値は取り出されたメッセージの値に設定されます。したがって、別のメッセージを正常に読み取るためにこれらの値をリセットする必要があります。

他のメッセージをそのキューから取得できるかどうかも検査してください。

- 他のアプリケーションは、キューからメッセージを取得できますか。
- 予期しているメッセージは、持続メッセージとして定義されましたか。
持続メッセージとして定義されていなくて、MQSeries が再始動した場合には、メッセージは失われています。
- 別のアプリケーションがそのキューを排他的にアクセスしていましたか。

キューに問題が見いだされず、MQSeries が実行されている場合には、ユーザーが考えていたキューへのメッセージの書き込み処理について次の検査を行ってください。

- アプリケーションは開始されていましたか。
トリガーで開始されたと思われる場合には、正しいトリガー・オプションが指定されていたかどうか検査してください。
- アプリケーションは停止しましたか。
- トリガー・モニターは実行されていますか。
- トリガー・プロセスは正しく定義されていましたか。

- アプリケーションは正しく完了しましたか。
ジョブ・ログに異常終了の記録がないかどうか調べてください。
- アプリケーションは変更をコミットしましたか、あるいは変更はバックアウトされましたか。

複数のトランザクションがキューをサービスしている場合、それらは互いに対立する可能性があります。たとえば、あるトランザクションは、バッファ長ゼロを指定した MQGET 呼び出しを出してメッセージの長さを調べ、その後、そのメッセージの *MsgId* を指定した特定の MQGET 呼び出しを出すします。しかし、一方で、別のトランザクションは、そのメッセージについて正常な MQGET 呼び出しを出すため、最初のアプリケーションは理由コード MQRC_NO_MSG_AVAILABLE を受け取ることになります。複数サーバー環境で実行されることが予期されるアプリケーションは、この状況に対処できるよう設計されている必要があります。

メッセージは受信されたが、アプリケーションがある点でそれを処理できなかった場合を考えてください。たとえば、予期した形式のメッセージにエラーがあったためプログラムがそれを拒否しましたか。そのような場合は、『メッセージが予期しない情報または破壊された情報を含んでいる場合』を参照してください。

メッセージが予期しない情報または破壊された情報を含んでいる場合

メッセージに含まれている情報が、アプリケーションの予期していたものではない場合、あるいはその情報がある点で破壊されていた場合、次の点を考慮してください。

- ユーザーのアプリケーション、つまりメッセージをキューに入れるアプリケーションは変更されましたか。

変更はすべて、その変更を認識している必要のあるすべてのシステムで同時に反映されるようにしてください。

たとえば、メッセージ・データの形式が変更された可能性がある場合、どちらのアプリケーションもその変更を取り入れるために再コンパイルする必要があります。一方のアプリケーションが再コンパイルされていなければ、他方のアプリケーションにとってはデータが破壊されているように見えます。

- アプリケーションは、誤ったキューにメッセージを送信していますか。
ユーザーのアプリケーションが受け取っているメッセージは、実際には別のキューをサービスしているアプリケーションに向けられたものではないかと

うか検査してください。無許可のアプリケーションが誤ったキューにメッセージを書き込むのを防ぐために、必要に応じてセキュリティー定義を変更してください。

アプリケーションが別名キューを使用していた場合は、別名が正しいキューを指し示しているかどうか検査してください。

- トリガー情報は、このキューについて正しく指定されていますか。
ユーザーのアプリケーションが開始されていたのか、または別のアプリケーションが開始されていたのかを検査してください。

上記の検査を行っても問題を解決できない場合には、メッセージを送信するプログラムとメッセージを受信するプログラムの両方のアプリケーション論理を検査しなければなりません。

分散キューを使用している場合の不正出力の問題

アプリケーションが分散キューを使用する場合には、次の点を考慮してください。

- MQSeriesは、送信側のシステムと受信側のシステムの両方に正しくインストールされ、分散キューイング用に正しく構成されていますか。
- 2つのシステム間でリンクは使用できますか。

両方のシステムが使用可能で、かつ MQSeries に接続されているかどうか検査してください。2つのシステムの間接続、および2つのキュー・マネージャーの間のチャンネルがアクティブかどうかを検査してください。

- トリガー操作が、送信側のシステムでオンに設定されていますか。
- 待機の対象となっているメッセージは、リモート・システムからの応答メッセージですか。

トリガー操作が、リモート・システムでアクティブにされているかを検査してください。

- キューが満杯になっていませんか。

満杯になっていると、アプリケーションは必要なメッセージをキューに書き込めません。その場合は、メッセージが送達不能キューに書き込まれているかを検査してください。

送達不能キューのヘッダーは、メッセージが宛先キューに書き込まれなかった理由を示す理由コードまたはフィードバック・コードを含んでいます。送達不能キューのヘッダー構造については、MQSeries アプリケーション・プログラミング・リファレンス を参照してください。

- 送信側のキュー・マネージャーと受信側のキュー・マネージャーの間に不整合がありませんか。

たとえば、メッセージ長が、受信側のキュー・マネージャーの取り扱える長さより長い場合があります。

- 送信側のチャンネルと受信側のチャンネルのチャンネル定義に互換性がありますか。

たとえば、順序番号折り返しに不整合があると、分散キューイング・コンポーネントは停止します。分散キューイングの詳細については、*MQSeries 相互通信* を参照してください。

- TCP/IP リスナーを始動しましたか。

MQSeries と Compaq の間の通信の通信プロトコルに TCP/IP を使用している場合、TCP/IP リスナーを実行していなければなりません。詳細については、41ページの『TCP/IP リスナーの指定および制御』を参照してください。

- TCP/IP リスナーは、正しい TCP/IP ポートで listen していますか。

TCP/IP リスナーは、キュー・マネージャーの QMINI ファイルの TCPConfig スタンザにある、TCPListenerPort 項目で定義されたポートで listen します。詳細については、70ページの『キュー・マネージャーが listen する TCP/IP ポート』を参照してください。

- TCP/IP プロセス名は正しいですか。

TCP/IP 通信プロトコルを使用している場合、Compaq システムは TCP/IP プロセス用のデフォルト・プロセス名 (\$ztc0) を使用していますか。使用していない場合は、MQSeries パスにあるいくつかのサーバー・クラスを変更して、MQSeries チャンネルが正しいプロセス名を使用できるようにする必要があります。詳細については、71ページの『デフォルト以外の TCP/IP プロセスやポートのための MQS-TCPLISnn サーバー・クラスの再構成』を参照してください。

- MQSeries SNA リスナーは構成されていますか。

次のような場合があるとします。

- 通信プロトコルに SNA を使用する MQSeries チャンネルを実行している。
- Compaq のチャンネル・タイプが、リモート MQSeries システム (たとえば、RECEIVER) から開始されるのを待っている。
- リモート・システムが、チャンネルの始動に関する問題を抱えている。

このような場合、PATHWAY MQSeries SNA リスナーはキュー・マネージャーに対して実行できない可能性があります。MQSeries SNA リスナーがキュー・マネージャー SNAX/APC または ICE PATHWAY に対して構成されているか確認してください。詳細については、472ページの『LU 6.2 応答側プロセス』を参照してください。

- データ変換が関係していますか。送信側のアプリケーションと受信側のアプリケーションの間でデータ形式が異なっている場合には、データ変換が必要です。データ形式が、組み込まれている形式の 1 つとして認識される場合は、MQGET が出された時に変換が自動的に行われます。

データ形式が組み込まれている形式として認識されない場合には、ユーザー独自のルーチンで変換を行うために、データ変換出口を使用できます。ルーチンが正しくロードされているか確認してください。

データ変換についての詳細は、*MQSeries アプリケーション・プログラミング・ガイド* を参照してください。

エラー・ログ

MQSeries for Compaq NSKは、MQSeries 自体の操作、ユーザーが開始したキュー・マネージャー、および使用中のチャンネルからのエラー・データに関係したメッセージを取り込むために、多数のエラー・ログを使用します。

エラー・ログが保管される位置は、キュー・マネージャー名が知られているかどうか、およびエラーがクライアントに関連しているかどうかによって異なります。

- キュー・マネージャー名が知られていて、キュー・マネージャーが使用可能である場合は、次の位置になります。

```
<QMVOL>.<SUBVOL>L.MQERRLG1
```

- キュー・マネージャーが使用可能でない場合には、次の位置になります。

```
<MQSVOL>.<ZMQSSYS>L.MQERRLG1
```

- 第 1 障害サポート・テクノロジーは、以下にあります。

```
<QMVOL>.<SUBVOL>L.FDnnnnn
```

- 281ページの『FFST の読み方』を参照してください。

ログ・ファイル

このエラー・ログ・サブボリュームには、次のような名前のエラー・ログ・ファイルを 3 つまで入れることができます。

- MQERRLG1
- MQERRLG2
- MQERRLG3

キュー・マネージャーの作成後、キュー・マネージャーが必要とするときに 3 つのエラー・ログ・ファイルが作成されます。これらのファイルはそれぞれ、MQERRLG1、MQERRLG2、MQERRLG3 となり、作成した各キュー・マネージャーのサブボリュームに入れられます。

エラーまたはログ・メッセージが生成されるたびに、MQERRLG1 に入れられます。MQERRLG1 が満杯になると、これは MQERRLG2 にコピーされます。そのコピーの前に、MQERRLG2 が MQERRLG3 にコピーされます。MQERRLG3 の前の内容 (ある場合) は、破棄されます。

このようにして、最新のエラー・メッセージは常に MQERRLG1 に入れられ、他のファイルは、エラー・メッセージの履歴を保持する目的で使用されます。

チャンネルに関連したすべてのメッセージも、該当のキュー・マネージャーのエラー・ファイルに入れられます。ただし、キュー・マネージャーの名前が知られていない場合や、キュー・マネージャーが使用できない場合は別です。キュー・マネージャー名が入手できない場合やその名前を判別できない場合には、チャンネル関連メッセージは、システム・エラー・ログ (ZMQSSYS.MQERRLG1) に入ります。

エラー・ログ・ファイルの内容を調べるためには、`fup copy` コマンドか、通常の Compaq NSK エディター (読み取り専用モード) のいずれかを使用できます。(エラー・ログを更新モードで開くと、エラー・メッセージは失われてしまう場合があります。)

早期エラー

上記のエラー・ログがまだ確立されていないときにエラーが起こるような特別のケースがあります。MQSeriesは、そのようなエラーをエラー・ログに記録しようとしています。ログの位置は、どの程度のキュー・マネージャーが確立されたかによって異なります。

構成ファイルの破壊などのために位置情報を判別できない場合は、エラーは、インストール時に MQERRLG1 ファイルの ZMQSSYS サブボリュームに作成されたエラー・ファイルにログ記録されます。

構成ファイルの詳細については、243ページの『第13章 構成ファイル』を参照してください。

オペレーター・メッセージ

MQSeries for Compaq NSK では、オペレーター・メッセージの通常エラーを識別します。通常エラーは、普通、有効でないパラメーターをコマンドに指定するといった事柄をユーザーが行うと直接生成されます。これらのメッセージは、関連したウィンドウ (ある場合) に書き込まれ、キュー・マネージャー・サブボリュームにあるファイルにも書き込まれます。

特定のキュー・マネージャーに関連したエラーは、キュー・マネージャーのログ・サブボリュームの MQERRLG1 にログ記録されます。定義済みで操作可能なキュー・マネージャーにリンクできないエラーは、サブボリューム ZMQSSYS にある MQERRLG1 ファイルにログ記録されます。

MQERRLG ファイル内の EC 番号の複合

MQERRLG1 ファイルにログ記録されたメッセージの EC 番号は、EC プロセスの追跡のために ECBOSS によって割り当てられた番号です。。

MQERRLG1 で ECBOSS によって割り当てられた番号と、MQS-ECxx PATHWAY サーバーの名前で使用される番号に、直接的な相関はありません。たとえば、MQS-EC00 は EC 番号 0 に割り当てられません。割り当てられる EC 番号は、初期化完了メッセージの EC プロセス名と関連付けられます。EC プロセス名を使用して、MQS-ECxx PATHWAY サーバーを判別したり、問題分析を行ったりします。

エラー・ログの例

以下の例は、MQSeries for Compaq NSK エラー・ログの一部を示します。

```
...
02/01/01 11:41:56 AMQ8003: MQSeries queue manager started.
EXPLANATION: MQSeries queue manager Janet started.
ACTION: None.
-----
02/01/01 11:56:52 AMQ9002: Channel program started.
EXPLANATION: Channel program 'JANET' started.
ACTION: None.
-----
02/01/01 11:57:26 AMQ9208: Error on receive from host 'camelot
(9.20.12.34)'.
EXPLANATION: An error occurred receiving data from 'camelot
(9.20.12.34)' over TCP/IP. This may be due to a communications failure.
ACTION: Record the TCP/IP return code 232 (X'E8') and tell the
systems administrator.
-----
02/01/01 11:57:27 AMQ9999: Channel program ended abnormally.
EXPLANATION: Channel program 'JANET' ended abnormally.
ACTION: Look at previous error messages for channel program
'JANET' in the error files to determine the cause of the failure.
-----
02/01/01 14:28:57 AMQ8004: MQSeries queue manager ended.
EXPLANATION: MQSeries queue manager Janet ended.
ACTION: None.
-----
02/02/01 15:02:49 AMQ9002: Channel program started.
EXPLANATION: Channel program 'JANET' started.
ACTION: None.
-----
02/02/01 15:02:51 AMQ9001: Channel program ended normally.
EXPLANATION: Channel program 'JANET' ended normally.
ACTION: None.
-----
02/02/01 15:09:27 AMQ7030: Request to quiesce the queue manager
accepted. The queue manager will stop when there is no further
work for it to perform.
EXPLANATION: You have requested that the queue manager end when
there is no more work for it. In the meantime, it will refuse
new applications that attempt to start, although it allows those
already running to complete their work.
ACTION: None.
-----
02/02/01 15:09:32 AMQ8004: MQSeries queue manager ended.
EXPLANATION: MQSeries queue manager Janet ended.
ACTION: None.
...

```

EMS イベント

EMS イベントは、MQERRLG1 ファイルに作成される各エラー項目ごとに生成されます。EMS イベントの詳細については、224ページの『Event Management Service (EMS) イベント』を参照してください。

送達不能キュー

何らかの理由で送達できないメッセージは、送達不能キューに入れられます。MQSC DISPLAY QUEUE コマンドを出すことにより、キューが何らかのメッセージを含んでいるかどうかを検査できます。キューがメッセージを含んでいる場合、提供されているブラウザのサンプル・アプリケーション (MQSBCG0E) を使用し、MQGET 呼び出しを使用してキュー上のメッセージをブラウザできます。サンプル・アプリケーションは、命名されたキュー・マネージャーの指定されたキューのすべてのメッセージを処理し、指定されたキューのすべてのメッセージのメッセージ記述子フィールドとメッセージ・コンテキスト・フィールドの両方を表示します。

送達不能キューにメッセージが入れられた理由に応じて、そのキューにあるメッセージの後処理の方法を決める必要があります。

使用しているそれぞれのキュー・マネージャーに送達不能エラーがない場合に、問題が発生することがあります。この送達不能キューを作成したら、**runmqsc** を使用してキュー・マネージャーの DEADQ 属性を変更しなければなりません。

構成ファイルと問題判別

構成ファイルにエラーがあると、通常、キュー・マネージャーが見つからず、「キュー・マネージャーは利用不能」というタイプのエラーが起こります。

構成ファイルで実行する検査がいくつかあります。

- 構成ファイルが存在することを確認します。
- 適切な許可があるかどうかを確認します。
- MQSeries 構成ファイルが確実に正しいキュー・マネージャーとディレクトリーを参照するようにします。

MQSeries トレースの使用

MQSeries for Compaq NSK は、トレース機能に以下のコマンドを使用します。

- **strmqtrc** - 392ページの『strmqtrc (MQSeries トレースの開始)』を参照
- **dspmqrct** - 360ページの『dspmqrct (MQSeries 定様式トレース出力の表示)』を参照
- **endmqtrc** - 368ページの『endmqtrc (MQSeries トレースの終了)』を参照

トレース機能では、トレースされているそれぞれのエンティティごとに 1 つのファイルを使用し、トレース情報は適切なファイルに記録されます。

トレース・オプションは、QMINI オプションで指定されます。

注: MQSeries for Compaq NSK を使用すると、メッセージ・キュー管理 (MQM) 機能の「Queue Manager」メニューを経由して制御を行うことも可能です。

トレース・ファイル名

トレース・ファイル名は、以下のように構成されます。

TRccpppp

この *ccpppp* は、トレースを生成するプロセスのプロセス ID (PID) です。PID は、以下で構成されます。

cc。CPU の番号。

pppp。プロセスの番号。

トレース機能が、削除されていない同一のプロセス ID のトレース・ファイルを検出する場合、プロセス番号の最後の文字が英文字に置換されます。これにより、同じ PID を持つ 26 のプロセスに、出力を作成する機会が与えられます。たとえば、PID 00, 0315 の最初のトレース・ファイルは、TR000315 になります。同じ PID を持つプロセス 00, 0315 の完了時に開始する 2 番目のプロセスの場合、トレース・ファイルは、TR00031A になります。

注: この制限により、トレース・ファイルを調査した後すぐに、システムから除去する必要があります。

トレース・データのサンプル

以下のサンプルは、トレースの抜粋です。


```

...
ID      ELAPSED_MSEC   DELTA_MSEC     APPL    SYSCALL KERNEL  INTERRUPT
30d     0 0 MQS CEI Exit!. 12484.1 xcsWaitEventSem rc=10806020
30d     0 0 MQS CEI Exit! 12484.1 zcpReceiveOnLink rc=20805311
30d     0 0 MQS FNC Entry 12484.1 zxcProcessChildren
30d     0 0 MQS CEI Entry. 12484.1 xcsRequestMutexSem
30d     1 0 MQS CEI Entry.. 12484.1 xcsHSHMEMBtoPTR
30d     1 0 MQS CEI Exit... 12484.1 xcsHSHMEMBtoPTR rc=00000000
30d     1 0 MQS FNC Entry.. 12484.1 xllSemGetVal
30d     1 0 MQS FNC Exit... 12484.1 xllSemGetVal rc=00000000
30d     1 0 MQS FNC Entry.. 12484.1 xllSemReq
30d     1 0 MQS FNC Exit... 12484.1 xllSemReq rc=00000000
30d     1 0 MQS CEI Exit.. 12484.1 xcsRequestMutexSem rc=00000000
30d     2 0 MQS CEI Entry. 12484.1 xcsReleaseMutexSem
30d     2 0 MQS CEI Entry.. 12484.1 xcsHSHMEMBtoPTR
30d     2 0 MQS CEI Exit... 12484.1 xcsHSHMEMBtoPTR rc=00000000
30d     2 0 MQS FNC Entry.. 12484.1 xllSemRel
30d     2 0 MQS FNC Exit... 12484.1 xllSemRel rc=00000000
30d     2 0 MQS CEI Exit.. 12484.1 xcsReleaseMutexSem rc=00000000
30d     2 0 MQS CEI Entry. 12484.1 xcsHSHMEMBtoPTR
...

```

図 44. トレースのサンプル

注:

1. この例では、データの切り捨てが行われていますが、実際のトレースには完全な機能名と戻りコードが提示されます。
2. 戻りコードはリテラルではなく値として戻されます。

第 1 障害サポート・テクノロジー™ (FFST)

FFST エラーは、通常、重大なエラーであって、システム関連の構成問題か MQSeries の内部エラーを示します。大抵の場合、キュー・マネージャーは引き続き作動可能ですが、いくつかの、あるいはすべてのアプリケーションに対するサービスが少しの間中断してしまうことがあります。FFST は、ZMQSSYS.MQSYSLOG ファイルで参照されます。

FFST の読み方

ファイル名の形式は `FDnnnnn` です。ここで、`nnnnn` エラーを報告するプロセス ID です。

プロセスが FFST レポートを作成すると、EMS イベントも生成します。

一般的な FFST レポートを 282ページの図45 に示してあります。

```

MQSeries First Failure Symptom Report
=====
Date/Time      :- February 6 12:23:26 2001
Host Name      :- ¥HURSLEY
PIDS           :- 5724A39
LVLS           :- 510
Product Long Name :- MQSeries for Compaq NonStop Kernel
Vendor         :- IBM
Probe Id       :- RM020011
Application Name :- MQM
Component      :- rrxOpenSync
Build Date     :- Feb 5 2001
Exe File Name  :- ¥HURSLEY.$DATA0.ZMQSEXE.MQMCACAL
UserID         :- MQM.MANAGER
Process File Name :- ¥HURSLEY.$Z734:15441941
Node number    :- 1
CPU            :- 0
PIN            :- 339
QueueManager   :- MT01
Major Errorcode :- xecF_E_UNEXPECTED_RC
Minor Errorcode :- Unknown(A)
Probe Type     :- MSGAMQ6118
Probe severity  :- Severity 2: error
Probe Description :- AMQ6118: An internal MQSeries error has occurred.
Text           :- Error creating synch file

Arith1         :- 10 (0xa)
Comment1       :- error 0000000010 in function 0000000020

```

図 45. 第 1 症状レポートのサンプル

ただし、解決可能な問題もあります。FFST に「out of resource」または「out of space on device」が示された場合は、該当するシステムの限界を超えたためと考えられます。

この問題を解決するには、該当する限界値を増やし、キュー・マネージャーを再始動します。

第15章 スケーラビリティおよびパフォーマンス

この章では、MQSeries アプリケーション・プログラムのパフォーマンスおよびスケーラビリティを最大限に発揮する技法について解説します。この章には、次の節があります。

- 『概要』
- 285ページの『持続メッセージ』
- 285ページの『非持続メッセージ』
- 287ページの『キュー・サーバーおよびキュー・ファイル』
- 293ページの『CPU の割り当て』
- 295ページの『FASTPATH バインディング・アプリケーション・プログラム』

概要

パフォーマンスおよびスケーラビリティのチューニングは、2つの主要なリソース、すなわち CPU とディスク・サブシステムの使用を最小限に抑えるために行われます。使用する CPU およびディスク入出力が少ないほど、アプリケーションのパフォーマンスとスケーラビリティは向上します（たとえば、ハードウェアとシステム・ソフトウェアを最大限に使用することにより、ワークロードが大きくなっても処理を行うように構成できます）。

この章では、アプリケーション・プログラム自体と MQSeries のパフォーマンスを向上させる技法について解説します。続く節では、アプリケーションのパフォーマンスを向上させるための一般的な原則を要約します。

パフォーマンスおよびスケーラビリティを向上させる新規アプリケーションを設計する

新規アプリケーションの設計段階の初期では、MQSeries とその他のサブシステムをどのように使用するか考慮する必要があります。業務上の必要によって、どの MQSeries 機能が必要か、またはアプリケーションごとにどの MQSeries 機能が関係しているかを判別しなければなりません。MQSeries 機能（メッセージ持続性など）によっては、強力な整合性と送達保証を備えているものがあります。このため、提供する CPU およびディスク入出力の量を増やす必要があります。特定のアプリケーションでこれらの保証が必要ではない場合、それ

スケーラビリティおよびパフォーマンスの概要

に応じて MQSeries を構成すると、大きなパフォーマンス向上につながります。業務上の必要が切迫していることが認められる場合には、設計の初期段階でこの判断を行うのが最善です。

共用リソースの使用を最小限にしたり、使用しなくてもすむように設計する

メッセージ・トラフィックの拡大を緩和するには、基礎となるハードウェアおよびシステム・ソフトウェアを最大限使用することが求められます。通常は、共用リソースが負荷の増加などの障害となります。この障害は、共用の度合いが大きくなったり (たとえば、同じ CPU を共用するユーザーの増加)、消費されるリソースが増えたり (たとえば、各ユーザーが実行する作業の増加) することによって、大きくなります。

パフォーマンスおよびスケーラビリティの観点では、CPU およびディスク・サブシステムは、最も頻繁に共用されるリソースを表します。したがって、最大の注意を払う必要があります。

MQSeries プロセスは、必要な数の CPU に分散させることができます。複数の CPU を持つシステムでは、使用可能な CPU に MQSeries プロセスを分散すると、デフォルトの CPU 割り当てを使用する場合よりもパフォーマンスが向上します。同様に、ディスク・サブシステムの使用率を最大にするために、可能であれば、それぞれのキュー・ファイルをそれぞれのディスク・ボリューム上に配置し、別々の Compaq NSK ディスク・プロセスによってサービスを提供することが賢明です。

パフォーマンス・チューニングは本質的に対話式

パフォーマンスを客観的に向上させるには、測定・チューニング・再測定のサイクルが必要です。各チューニングのサイクルには、主要な変数の影響を他の変数の影響と比較できるように、主要な変数を 1 つだけ変更することが含まれていなければなりません。通常、同時に複数の設定を更新することは逆効果です。一部の変更がそれ以外の変更よりも総合的なパフォーマンスを向上させることがある一方、それ以外の変更がパフォーマンスを低下させてしまうことがあるからです。アプリケーション・システムのパフォーマンスは通常、1 つの限られた障害によって判別されます。通常、システムのチューニング変更を行うと、他のいくつかのリソースが障害になります。この性質を持つ変更の間での相互作用は、パフォーマンスのチューニングを行う際に、測定・チューニング・再測定の方式に注意深く従う必要があることを強調するものです。

たとえば、あるアプリケーションが大量の小さい持続メッセージをキューに書き込むとします。この場合、すべてのメッセージを非持続メッセージにすることの効果や、同量のデータを少量の大きいメッセージの内部に書き込むことの

効果をテストするのが役立ちます。どちらの変更も (通常は) パフォーマンスを向上させますが、各変更ごとに別々に測定・チューニング・再測定のサイクルを実行しないと、どちらの方法が大きい向上につながるのかわかりません。

持続メッセージ

持続メッセージには、送達と回復可能性の点で、MQSeries が提供する最強の保証を有しています。持続メッセージは常に、固定されたメディアに保管されるため、キュー・マネージャー始動時にも有効です。Nonstop TM/MP はキュー・ファイルを監査して、持続メッセージの読み取り / 書き込みを行うときに、キュー・ファイル自体と TM/MP 監査ファイルの両方に対してディスク・アクティビティーが行われるようにします。システムまたはハードウェアの障害時であっても、持続メッセージのトランザクション整合性を保守するためには、TM/MP 監査ログが必要です。MQSeries アプリケーション設計のパフォーマンスを評価する際に、持続メッセージに関連した TM/MP 監査ログを考慮する必要があります。

持続とはメッセージのプロパティーのことであり、それが保管されるキューではありません。管理者は書き込み側アプリケーションが指定していなくても、新規メッセージが持続かどうかを指定することができるので、キューには持続メッセージも非持続メッセージも保管することができます。(MQSeries アプリケーション・プログラミング・リファレンス で説明されている『キューの属性 DefPersistence』を参照してください。)

非持続メッセージ

持続メッセージとは異なり、非持続メッセージはディスクに固定されないため、キュー・マネージャーの再始動時に無効になります。キューのキュー・サーバー・オプションによっては、バックアップ・キュー・サーバーに非持続メッセージのチェックポイントがとられることがあります。非持続メッセージを使用する基本的な理由は、パフォーマンスです。持続メッセージは送達と回復可能性の点で強力な保証を備えているため、それらの読み取り / 書き込みを行うには、キュー・ファイル自体と TM/MP 監査ファイルの両方に対するディスク・アクティビティーが必要になります。このディスク・アクティビティーが、持続メッセージの読み取り / 書き込みを行うアプリケーションと、持続メッセージを他のキュー・マネージャーに移動させる MQSeries チャネルのパフォーマンスを低下させます。

非持続メッセージ

非持続メッセージをできる限り使用すれば、パフォーマンスが格段に向上します。なぜなら、非持続メッセージはディスクに固定されない代わりに、Nonstop キュー・サーバーによって管理されるメモリーにキャッシュされるからです。

持続とはメッセージのプロパティーのことであり、それが保管されるキューにはありません。管理者は書き込み側アプリケーションが指定していなくても、新規メッセージが持続かどうかを指定することができるので、キューには持続メッセージも非持続メッセージも保管することができます (『キューの属性 DefPersistence』を参照)。

非持続メッセージおよびチャネル

メッセージ・チャネルは、ネットワーク上で送信されるメッセージの送達を 1 回限り確認するために、送信側と受信側の両方で同期ロギングを使用します。この同期ロギングは、チャネルがキューとの間でメッセージの読み取り / 書き込みを行う際に、(キュー・マネージャーの代わりに) TM/MP によって実行される監査ロギングを補足するものです。非持続メッセージを送受信する際には、NPMSPEED 属性を FAST に設定することによって、同期ロギングを実行しないようにメッセージ・チャネルを構成することができます。NPMSPEED チャネル属性は、非持続メッセージを処理する送信側チャネルと受信側チャネルの両方の動作を制御します。あるチャネルについて NPMSPEED が NORMAL に設定されている場合、非持続メッセージはチャネルのメッセージ・バッチの一部となり (BATCHINT および BATCHSZ 属性の定義のとおり)、持続メッセージと同じ方法での同期ロギングが必要になります。さらに、NPMSPEED が NORMAL に設定されている場合、チャネルは同期点の制御のもとで非持続メッセージの読み取り / 書き込みを行います。これによって、トランザクションの最初と最後に TM/MP 監査ファイルの少量のアクティビティが行われることとなります。

NPMSPEED が FAST に設定されている場合、メッセージはチャネルの現行バッチの一部ではないため、同期点の制御外でキューとの間の読み取り / 書き込みが行われます。したがって、NPMSPEED(FAST) を使用すると、チャネル・ディスク・アクティビティの 2 つのソースが除去されます。それは、チャネル・バッチの同期化メカニズムと、TM/MP 監査ロギング (本来なら同期点の制御のもとでメッセージの読み取り / 書き込みを行うために実行される) です。

NPMSPEED(FAST) は、障害の発生後に非持続メッセージの回復可能性と引き換えに、かなり高度なパフォーマンスを獲得するためのパフォーマンス・オプションです。NPMSPEED(FAST) を使用すると、チャネルまたはネットワークに障害が発生した場合に非持続メッセージが失われます。

NPMSPEED のデフォルト値は FAST です。

キュー・サーバーおよびキュー・ファイル

キュー・サーバーは、メッセージの読み取りと書き込み、およびそれらのメッセージの保管を仲介する MQSeries プロセスです。このため、キュー・サーバーは、注意すべき主要な MQSeries コンポーネントを表します。キュー・サーバー構成は、使用中の MQSeries システムのパフォーマンスに大きな影響を与ることがあります。

キュー・サーバーには、メッセージの物理ストレージをキューに保持します。キュー・サーバーは持続メッセージと非持続メッセージを基本的に別個の方法で管理するため、持続メッセージと非持続メッセージのストレージを別々に調べることは役立ちます。

キュー・サーバーは 1 つ以上のキューを管理することができます。キューの作成時に、キューはデフォルト・キュー・サーバーによって管理されます。そのため、デフォルトでは、キューが他のキュー・サーバーに割り当てられていない場合には、すべてのキューはデフォルト・キュー・サーバーによって管理されます。

キュー・サーバーによって管理されるキューごとに、次のファイルがあります。

- キュー・ファイル
- キュー・オーバーフロー・ファイル
- タッチ・ファイル

さらに、キュー内にある大きいメッセージごとに、メッセージ・オーバーフロー・ファイルもあります (キューのメッセージ・オーバーフローしきい値で定義されているとおり)。メッセージ・オーバーフロー・ファイルについては、後の節で説明されます。

持続メッセージのストレージ

持続メッセージは常にディスクに固定されます。持続メッセージの保管方法は、大抵、メッセージのサイズによって決まります。

メッセージ・サイズ	保管方法
< 3000 バイト (約)	全体がキュー・ファイルに保管される
> 3000 バイト (約) <= メッセージ・オーバーフローしきい値	最初の 3000 バイト (約) がキュー・ファイル保管され、残りはキュー・オーバーフロー・ファイルに保管される

メッセージ・サイズ	保管方法
> メッセージ・オーバーフローしきい値	最初の 3000 バイト (約) はキュー・ファイルに保管され、残りはメッセージ・オーバーフロー・ファイルに保管される

非持続メッセージのストレージ

非持続メッセージは、キュー・サーバーのプロセスでメモリー・バッファーに保管されます。この理由で、キュー・サーバーは非持続メッセージの読み取りおよび書き込み中に、ディスク入出力を実行しません。

キュー・サーバーの CPU 分散

キュー・サーバーは PATHWAY サーバー・クラスです。そのため、キュー・マネージャーの PATHWAY で定義されます。キュー・マネージャーが作成されると、デフォルト・キュー・サーバーが PATHWAY に定義され、キュー・サーバーの 1 次プロセスが CPU 0 で実行されます。通常、CPU 0 には高優先順位のシステム・プロセスが含まれているため、キュー・サーバーは CPU 0 で実行しない方が賢明です。むしろ、相対的に静止しているたくさんの CPU を識別してから、それぞれの CPU にキュー・サーバーを作成してください。キューが明示的に割り当てられるまで、新規のキュー・サーバーはどのキューも管理しません。

altmqfls を使用してキュー・サーバーにキューを再割り当てする

キューを作成する際に (`runmqsc` を使用)、MQSeries はそれをデフォルト・キュー・サーバーに割り当てます。新規のキュー・サーバーを作成して、それらにキューを割り当てない場合、すべてのキューは単一のデフォルト・キュー・サーバーによって管理されます。

使用中の MQSeries システムでは、すべてのキューを単一のキュー・サーバーに割り当てることは効率的でもありませんし、拡張が容易でもありません。新規のキュー・サーバーを作成して、それらにキューを割り当てる主要な理由は、キュー・サーバーの CPU 負荷を使用可能な CPU 間に均等に分散させることです。

`altmqfls` を使用して、キューを別のキュー・サーバーに割り当てることができます。たとえば、次のようにします。


```
altmqfls --qmgr QMGR --type QLOCAL --server $QS01 TEST.QUEUE
```

altmqfls の使用の詳細については、327ページの『altmqfls (キュー・ファイル属性の更新)』を参照してください。

クラスター伝送キュー: SYSTEM.CLUSTER.TRANSMIT.QUEUE

MQSeries はすべてのクラスター化操作に単一のクラスター伝送キューを使用します。ご使用のキュー・マネージャーが使用中のクラスターの一部である場合、クラスター化操作のパフォーマンスを最大限にし、その他のアプリケーションに与える影響を最小限にするために、このキューは専用のキュー・サーバーに割り当てる必要があります。

altmqfls を使用してキュー・ファイルの配置を変更する

デフォルトでは、3 つの 1 次キュー・ファイル (キュー・ファイル、キュー・オーバーフロー・ファイル、およびタッチ・ファイル) はキュー・マネージャーの M サブボリュームに保管されます。メッセージ・オーバーフロー・ファイルは、キュー・サーバーのサブボリュームに保管されます。

キュー・ファイルを他のディスク・ボリュームに再配置することを考慮することには、次の 2 つの理由があります。

- ディスク入出力負荷をより均等に分散させる
- ディスク・ボリュームごとにオープンしているファイルの数に関する OS の限界を克服する

altmqfls を次のように使用して、既存のキュー・ファイルを別のディスク・ボリュームに移動させることができます。

```
altmqfls --qmgr QMGR --type QLOCAL --volume $DATA01 TEST.QUEUE
```

altmqfls を使用してキューを移動させるには、キューがオープンしてはなりません。

既存のメッセージ・オーバーフロー・ファイルを移動させることはできません。しかし、**altmqfls** の `--msgofsubvol` オプションを次のように使用して、新規のメッセージ・オーバーフロー・ファイルを任意の位置に作成することができます。

キュー・サーバーおよびキュー・ファイル

新規のメッセージ・オーバーフロー・ファイルを新しい位置に作成するには、次のようにします。

```
altmqfls --qmgr QMGR --type QLOCAL --msgofsubvol $DATA01.TESTMOF TEST.QUEUE
```

altmqfls の使用の詳細については、327ページの『altmqfls (キュー・ファイル属性の更新)』を参照してください。

キュー・ファイルの区画化

ファイル区画化は、ファイルを複数のディスク・ボリュームに分割する技法です。ファイルは論理的に複数の区画を構成し、各ディスク・ボリュームに1つの区画があります。

キュー・ファイルの区画化の利点は、次のとおりです。

- 区画化により、単一キューのための1つのディスク入出力負荷が複数のディスク・ボリュームに分散され、したがって複数の入出力パスに分散されることになる。
- 区画化により、論理キュー・ファイルを、ディスク・ボリューム上の物理ファイルの最大サイズよりも大きくすることができる。

既存のキュー・ファイルを区画化するには、標準の TAACL コマンド (FUP ユーティリティ) を使用できます。MQSeries では、FUP を使用して既存のキュー・ファイルを区画化する方法を示す、PARTIT という TAACL スクリプトを提供しています。

キュー・ファイルが区画化されると、管理側のキュー・サーバーは新規メッセージを使用可能なすべての区画に均等に分散させようとしています。これにより、区画化の利点がさらに広がります。

注: **altmqfls** は、区画化されたキュー・ファイルの移動を行えません。

メッセージ・オーバーフロー・ファイル

メッセージ・オーバーフロー・ファイルは、定義されたメッセージ・オーバーフローしきい値より大きいメッセージごとに作成されます。メッセージ・オーバーフロー・ファイルは TM/MP によって監査されない、未構造化ファイルです。非常に大きいメッセージの場合、メッセージの大部分をメッセージ・オーバーフロー・ファイル (TM/MP によって監査されない) に保管すると、より効率的です。比較的小さいメッセージの場合、メッセージ全体を TM/MP 監査済

みキュー・ファイル (キューおよびキュー・オーバーフロー・ファイル) に保管すると、より効率的です。クロス・ポイントは、過去の記録に基づいて、約 200 KB と判別されています。

メッセージ・オーバーフローしきい値は、キューの作成時にデフォルトの 200 KB に設定されます。しきい値は、**altmqfls** の `--oflowsize` を使用して変更できます。

altmqfls コマンドを次のように使用して、メッセージ・オーバーフローしきい値を新しい値に設定できます。

```
altmqfls --qmgr QMGR --type QLOCAL --oflowsize 400000 TEST.QUEUE
```

ブラウズ中にメッセージをバッファーに入れる

キュー・サーバーは、キューのすべての持続メッセージの最初の n バイトをメモリーに保持することができます。この機能には、キュー内の持続メッセージをブラウズするアプリケーションのパフォーマンスに非常に大きな影響を与えます。アプリケーションがブラウズするメッセージがこのブラウズしきい値より小さい場合、それぞれの持続メッセージをブラウズするためにディスク入出力は必要ありません。

必要であれば、たとえば、メッセージの内容を識別する持続メッセージの最初の 100 バイトに情報を含めることもできます。対象となるメッセージを見つけるために、各メッセージごとにデータの最初の 100 バイトをブラウズする作業は、ディスク入出力を伴わずに行うことができます。メッセージ・データ全体が見つかったら、それを読み取り、必要に応じてデキューすることができます。

altmqfls の `--browse` オプションを使用して、キューのブラウズしきい値を設定することができます。デフォルトでは、持続メッセージ・データはメモリーに保持されません (たとえば、ブラウズしきい値がゼロの場合)。最大値は 25000 です。

```
altmqfls --qmgr QMGR --type QLOCAL --browse 100 TEST.QUEUE
```

その他のキュー・サーバー・オプション

その他のキュー・サーバー・オプション (たとえば、キュー・マネージャーの始動時にキューがディスク・キャッシュからロードされるかどうか、または非持続メッセージのチェックポイントがバックアップ・キュー・サーバーに対してとられるかどうか、など) は、`--qsoptions` パラメーターを **L**、**S**、および **C** オプション付きで使用します。これらのオプションを使用して (単独でも組み合わせでも可)、キューの検索可能性および信頼性を微調整します。

注: コマンドを発行するたびに、**すべての** `--qsoptions SLC` が設定されます。たとえば、`--qsoptions S` は **L** と **C** の設定を解除します。 `--qsoptions` オプションはコマンド行で 1 回限り指定することができます。

始動時のロード

altmqfls `--qsoptions S` スイッチによって制御されます。このオプションにより、キュー・サーバーはキュー・ファイルを読み取り、キューが最初にオープンされるときではなく、キュー・サーバーの始動時 (一般にキュー・マネージャーの始動時) にその内部メッセージデータ構造を構築します。

```
altmqfls --qmgr QMGR --type QLOCAL --qsoptions S TEST.QUEUE
```

このオプションを使用すると、キューが最初にオープンされるときの CPU およびディスク入出力アクティビティーが少なくなる一方で、キュー・マネージャーの始動時にキュー・サーバー・アクティビティー (CPU およびディスク入出力) が増加します。

キャッシュ内にロックする

altmqfls `--qsoptions L` スイッチによって制御されます。このオプションにより、キュー・サーバーは、キューに関連したデータ構造およびチェーンをメモリー内にロックします。他のキューのためのスペースを確保するため、キューのメモリー・データ構造はディスクにアンロードされません。デフォルトの動作では、必要なときにキューのデータ構造をディスクにアンロードします。

```
altmqfls --qmgr QMGR --type QLOCAL --qsoptions L TEST.QUEUE
```

これらのオプションを使用すると、他のキューをできる限り犠牲にして、特定のキューのメモリー・データ構造に高速でアクセスします。

チェックポイント NPM

altmqfls --qsoptions C スイッチによって制御されます。このオプションにより、キュー・サーバーはノンストップのバックアップ・プロセスを行って、非持続メッセージのチェックポイントをとります。デフォルトの動作では、非持続メッセージ・データのチェックポイントをとります。

このオプションを使用すると、1 次キュー・サーバー・プロセスとバックアップ・キュー・サーバー・プロセスの両方の IPC トラフィックの増加と、CPU およびメモリー使用率の増加を抑えることにより、非持続メッセージの信頼性が向上します。

```
altmqfls --qmgr QMGR --type QLOCAL --qsoptions C TEST.QUEUE
```

測定カウンター

altmqfls --meascount スイッチによって制御されます。このオプションにより、キュー・サーバーはキューの CURDEPTH を使用して、ユーザー定義の測定カウンターを保守します。測定カウンターは、システムの総合的なパフォーマンス上の動作を評価するためのデータを収集する際に役立ちます。このオプションを使用すると、キューに対するメッセージの読み取り / 書き込みアクティビティー (MQGET および MQPUT) を、CPU やディスク入出力などのシステム関連の変数に相関させることができます。

```
altmqfls --qmgr QMGR --type QLOCAL --meascount TESTCOUNT1 TEST.QUEUE
```

このオプションにより、MQSeries を使用するシステムのパフォーマンスおよびスケーラビリティを正確に評価できるため、チューニングの結果が改善されます。

CPU の割り当て

キュー・マネージャーを始動すると、MQSeries for Compaq NSK はたくさんのプロセスを作成します。これらのプロセスの一部はコア・メッセージング操作を実行し、その他のプロセスは、それら一部のプロセスを間接的にサポートする機能を実行します。アプリケーションが MQI メッセージング呼び出し (MQGET および MQPUT) を実行しているときには、MQSeries のコア・メッセージング機能を提供するプロセスは使用中になります。

CPU の割り当て

次の MQSeries および Compaq NSK プロセスには、コア・メッセージング操作が関係しています。

- LQMA (ローカル・キュー・マネージャー・エージェント)
- キュー・サーバー
- 状況サーバー
- NSK ディスク・プロセス

次の MQSeries および NSK プロセスには、分散キューイング操作が関係しています。

- MCA
- デフォルト状況サーバー
- NSK TCPIP または SNA プロセス

次の MQSeries プロセスには、サポート操作または管理操作が関係していません。

- キュー・マネージャー・サーバー
- リポジトリ・サーバー
- Execution Controller (ECs および ECBoss)
- チャネル・イニシエーター
- トリガー・モニター
- リスナー

過負荷のかかった MQSeries システムは通常、上述の最初の 2 つのカテゴリのプロセス (コア・メッセージング・プロセスと分散キューイング・プロセス) では、高い CPU 使用率を示します。

したがって、MQSeries の CPU 負荷を分散させることには、キュー・マネージャーの LQMA、キュー・サーバーおよび MCA をできるだけ多くの CPU に分散させることが関係しています。

新規のキュー・マネージャーのデフォルト構成は、CPU 0 でプロセスを実行することです (Nonstop プロセスのペアの場合、1 次プロセスは CPU 0 で実行され、バックアップ・プロセスは CPU 1 で実行されます)。これは実稼働環境には適していません。そのため、使用中のプロセスをできるだけ多くの CPU に効率よく分散するように、キュー・マネージャーの PATHWAY を再構成する準備をする必要があります。CPU 0 には通常、高優先順位オペレーティング・システム・プロセスが多く含まれているため、使用中の MQSeries プロセスの実行には適していません。

EC の CPU 分散は、キュー・マネージャーの PATHWAY で定義されますが、これは LQMA と MCA の実行を制御します。キュー・サーバーもキュー・マネージャーの PATHWAY で定義されます。

MQSeries プロセスについて考慮すると、NSK オペレーティング・システムのディスク・プロセスは、メッセージング操作の重要なコンポーネントです (特にキュー・サーバーが持続メッセージを処理する場合)。ネットワーク上で分散キューイングを過剰に使用すると、必然的に、対応する TCPIP または SNA プロセスが CPU を消費することになります。MQSeries のインストールで総合的なパフォーマンスのプロファイルを評価する際に、それらのプロセスの数と CPU の配置を考慮する必要があります。

CPU へのプロセスの割り当て、ディスク・プロセスまたは TCPIP の構成については、関連する Compaq NSK システムの資料を参照してください。SNA プロセスの構成については、関連する SNAX または ICE の資料を参照してください。TCPIP または SNA プロセスの構成については、471ページの『付録 M. 通信のセットアップ』を参照してください。

FASTPATH バインディング・アプリケーション・プログラム

FASTPATH バインディングは、MQSeries アプリケーションがより効率的に実行するように設計された MQI の機能です。FASTPATH バインディングを使用すると、MQSeries アプリケーション・プログラムによって発行されるすべての MQI 動詞において、固有のオーバーヘッドを減らすことができます。FASTPATH バインディングを使用するアプリケーションをトラステッド・アプリケーションと言います。なぜなら、キュー・マネージャーのソフトウェアおよびメモリーが、顧客のアプリケーション・ソフトウェアと密接に関連しているからです。トラステッド・アプリケーションにエラーが発生すると、MQSeries データ構造が損傷し、キュー・マネージャーの整合性に危険が及ぶ可能性があります。

バックグラウンド

アプリケーション・プログラムが MQCONN 動詞を実行すると、MQSeries はローカル・キュー・マネージャー・エージェント (LQMA) という特殊なプロセスを作成 (または再作成) します。LQMA は、接続ハンドルを使用して、アプリケーションが行う後続の MQI 呼び出しにサービスを提供します。MQSeries for Compaq NSK では、LQMA は接続側アプリケーションと同じ CPU か、または別の CPU で実行します。

FASTPATH バインディング・アプリケーション・プログラム

LQMA は別個のプロセスであるため、アプリケーション・プログラムは、MQSeries が使用するメモリーまたはファイルに対して直接アクセスを行いません。そのため、エラーのあるアプリケーション・プログラムが LQMA に損害を与えることはありません。このようにして、その操作に重要な MQSeries ソフトウェアおよびデータ構造は、顧客のアプリケーション・ソフトウェアおよびデータから分離されます。この分離には多くの犠牲が伴います。MQCONN 動詞は、新規の LQMA プロセスが作成されるまで（または既存の LQMA プロセスが再利用されるまで）完了できません。しかし、もっと大きな犠牲は、アプリケーションが MQI 動詞を発行するたびに LQMA に情報を渡さなければならないことです。

この MQI 情報は、プロセス間通信 (IPC) メカニズムを使用して LQMA に渡されます。IPC 要求は CPU 内 (LQMA が接続側アプリケーションと同じ CPU で実行してしまった場合) か、またはもっと犠牲の伴う CPU 間です。

MQI オーバーヘッドを削減する

アプリケーションの設計者は、各 MQI 動詞と関連したアプリケーションと LQMA 間の IPC オーバーヘッドを除去する（さらに、可能性のある LQMA プロセスの作成も避ける）方法として、FASTPATH バインディングを使用することができます。アプリケーションに対して FASTPATH バインディングが使用可能になっていると、別個の LQMA プロセスは使用されません。その代わりに、通常 LQMA に含まれている MQSeries のコンポーネントが、ユーザーのプロセス（たとえば、接続側アプリケーションのプロセス）にロードされます。

アプリケーションによって発行される後続の MQI 動詞では、LQMA を使用した IPC アプリケーションを必要としません。なぜなら、MQSeries ソフトウェアおよびデータ構造（通常は LQMA プロセスに保管される）は、アプリケーションのプロセス内にローカルに保持されるからです。キュー・マネージャーがキュー・サーバーなどの他の MQSeries プロセスと通信を行う必要がある場合には、その他の IPC アクティビティーが引き続き行われることに注意してください。FASTPATH バインディングはすべての IPC アクティビティーを除去するわけではありませんが、IPC アクティビティーの重要なソースは除去します。

FASTPATH バインディングの使用可能化

FASTPATH バインディングを使用するには、MQCONNX 動詞に MQCNO_FASTPATH_BINDING オプションを付けて使用し、キュー・マネージャーに接続します。MQCONNECTTYPE PARAM の値（それがあある場合）は、MQCONNX の動作に影響を与えます。MQCONNECTTYPE PARAM があある場

合、MQCONN が FASTPATH 接続をセットアップできるように、その値は FASTPATH でなければなりません。FASTPATH バインディング接続が確立されると、その他のすべての MQI 動詞の動作は標準バインディング接続のようになります。ただし、この後の節で記述される事柄は除きます。

FASTPATH バインディングの使用時の制限

STANDARD バインディング接続を使用すると、アプリケーション・ソフトウェアは内部 MQSeries データから分離されます。この分離は、トラステッド・アプリケーションが FASTPATH バインディング接続を確立すると除去されません。そのため、トラステッド・アプリケーションにエラーが発生すると、MQSeries データ構造が損傷し、キュー・マネージャーの整合性に危険が及ぶ可能性があります。特定のアプリケーションで FASTPATH アプリケーションを使用するかどうかを評価する際には、この点を考慮に入れる必要があります。

次の追加の考慮事項はトラステッド・アプリケーションに適用されます。

- トラステッド・アプリケーションはキュー・マネージャーから明示的に切断しなければならない (たとえば、MQDISC を発行)。
- トラステッド・アプリケーションは、**endmqm** コマンドが発行される前に停止しなければならない。トラステッド・アプリケーションは、アドミニストレーター・ユーザー ID (mqm プリンシパルに対応するユーザー ID) として実行しなければならない。
- トラステッド・アプリケーションは、稼働中の MQSeries リポジトリ・サーバーを含む CPU でのみ実行できる。

第16章 データ保全性と可用性

この章では、データ保全性と可用性の概念について説明し、これらの重要なシステムの性質を MQSeries for Compaq NSK V5.1 の管理や構成にどのように適用するか説明します。この章には、MQSeries から期待できるデータ保全性と可用性のレベルと、これらのレベルに影響を与える可能性のある構成の選択項目が説明されています。この章には、次の節があります。

- 『データ保全性』
- 300ページの『可用性』
- 302ページの『持続データと非持続データ』
- 305ページの『データベースの一貫性』
- 310ページの『重要なプロセス』
- 318ページの『クラスター』
- 318ページの『可用性に関する構成上の考慮事項』
- 319ページの『データ保全性に関する構成上の考慮事項』

この章の情報を正しく理解し活用するには、27ページの『第2章 MQSeries for Compaq NSK V5.1 のアーキテクチャー』を読んで理解する必要があります。

データ保全性

特にオンライン・トランザクション処理アプリケーションの場合に、データの記憶と管理に関して求められる以下の優れた性質について考慮すると、最善の仕方ですべてのデータ保全性の概念を理解できます。

- データのレコードがデータベースに書き込まれたり、レコードがデータベースから読み取られたりする場合に、転送時にエラー通知が出されずにデータが破壊されたり、重複したり、失われたりしてはならない。
- マルチプロセスでデータを同時アクセスする必要がある場合に、これらのプロセスにおけるデータ・ビューは同一でなければならず、データが破壊されたり、重複したり、失われたりすることから保護しなければならない。
- 複数のデータベース中のデータに一貫性のある変更を複数加える必要がある場合に、すべての変更を加えるか全く変更を加えないかのどちらかでなければならない。

MQSeries の場合、上記のデータ記憶に関するデータ保全性の要件は、キュー上のメッセージング操作 (MQPUT や MQGET など) だけに適用できます。複数

データ保全性

のデータベースに変更を加えた場合は、アプリケーション・データベースと MQSeries キューの間で、全体として一貫性が保持されなければならないので注意してください。

MQSeries for Compaq NSK V5.1 は、単一障害ポイント（ハードウェアまたはソフトウェア）を使用して、持続データの操作時のデータ保全性を保守するように設計されています。実際には、複数障害ポイントを使用してデータ保全性を保守できる場合もあります。これは、非持続メッセージが信頼できないことを暗示している訳ではありません。キュー・サーバー・アーキテクチャーには、壊滅的なシステム障害の場合を除いて、持続メッセージと同様に非持続メッセージを信頼できるようにする機能が備えられています。（キュー・サーバー・アーキテクチャーの詳細については、27ページの『第2章 MQSeries for Compaq NSK V5.1 のアーキテクチャー』を参照してください。）

MQSeries for Compaq NSK V5.1 では、Compaq NSK システム・ソフトウェアおよびハードウェアの基本機能と MQSeries ソフトウェア自体の組み合わせによって、データ保全性が備えられます。

以下のような MQSeries の構成に関する選択項目によって、データ保全性のレベルが影響を受けることがあります。

- アプリケーションによるメッセージ持続性に関する選択項目
- 持続メッセージの記憶技法に関する選択項目
- 非持続メッセージのチューニング・オプションに関する選択項目
- キュー・サーバー構成オプションに関する選択項目
- キュー・ファイルをサポートするハードウェアの構成
- NonStop TM/MP の使用法と構成方法
- リモート・データベース機能 (NonStop RDF) の使用法と構成方法

この章の後の方に個々の説明があります。

可用性

一般的な用法では、可用性とは、アプリケーションやサービスが操作可能で使用可能な時間を経過時間と比較して測定したものです。したがって、**連続可用性**とは、この種のシステムの最終目的を表現した用語です。もちろん、対応する時間枠がこの測定値に関連付けられていないと、この種の測定値は意味を成しません。ただし短時間の場合は 100% の可用性を求めるのは容易です。

現実的な操作可能時間間隔では、システムの可用性について、以下のようなさまざまなタイプの難題が生じます。

- ハードウェアおよびシステム・ソフトウェアの障害
- アプリケーション・ソフトウェア自体の障害
- 予防保守を行うためにシステムの性質に変更を加える必要性
- トラフィックやトランザクション負荷による設計の制約やリソースの制限の超過

データ保全性については、MQSeries for Compaq NSK V5.1 では、Compaq NSK システム・ソフトウェアおよびハードウェアの基本機能と MQSeries 自体の組み合わせによって、その可用性が備えられます。

MQSeries for Compaq NSK は (少なくとも現在の形式では)、ファイル・システムなどの Compaq NSK システム・ソフトウェアに備えられているものと同等のレベルの連続可用性を備えることは意図されていないことを覚えておくことは重要です。実際は、キュー・マネージャーのコンポーネントの中にはこのレベルの可用性は備えられているものもありますが、キュー・マネージャー全体としては備えられていません。

MQSeries for Compaq NSK V5.1 は、単一障害ポイント (ハードウェアまたはソフトウェア) のレベルの可用性を備えることを想定しています。

- サービスの中断や切断が起きるキュー・マネージャー接続は、障害の発生元のコンポーネントとの間の接続だけに限定される (たとえば、CPU 障害の場合は、その CPU で実行している LQMA プロセスによって設けられる接続だけに障害が起きる必要があります)。
- システム管理者が手操作による介入を行わなくても、新しい接続を試行する際に依然としてキュー・マネージャーが使用可能になっている。
- 直接障害の影響を受ける接続以外の接続から、キュー・マネージャー・オブジェクト (ローカル・キューなど) へアクセスできる。

上記のような MQSeries の Compaq NSK 固有の機能に加えて、すべての MQSeries バージョン 5.1 プラットフォームに共通の機能で、MQSeries for Compaq NSK の可用性を拡張するのに使用できるものがあります。以下のような MQSeries の構成に関する選択項目によって、可用性のレベルが影響を受けることがあります。

- アプリケーション設計に関する選択項目
- PATHWAY 構成オプションに関する選択項目
- 標準的な MQSeries 機能 (クラスターを含む) の使用法と構成方法
- MQSeries をサポートするハードウェアに関する選択項目
- 非持続メッセージの構成オプションに関する選択項目

可用性

- NonStop TM/MP の使用法と構成方法
- NonStop RDF の使用法と構成方法

持続データと非持続データ

持続性 という用語は、MQSeries に関連して使用される場合は、以下のようなデータの品質を暗示しています。

- 持続データに加えた変更内容は、キュー・マネージャーを再始動しても有効である。
- 持続データは不揮発性メディアに格納される。
- 持続データは、特定の操作環境で備えられるデータ保全性に関する最も高度な要件も満たす。
- 持続データ操作により高水準の保全性は得られるが、速度やリソースの使用効率は低下する。

MQSeries の可用性が著しく低下するリスクを最小限にするために、MQSeries の構成データベースのデータ保全性は最高レベルでなければならないので、ほとんどの管理操作は持続データに対して行われます。管理操作は頻繁に行われる訳ではないので、ほとんどの管理操作では速度やリソースの使用効率はあまり重要ではありません。

持続管理操作には次のものがあります。

- キューの属性に対する変更
- チャンネルの属性に対する変更
- 新しいキュー、プロセス、名前リスト、またはチャンネルの作成

非持続管理操作には次のものがあります。

- チャンネルの開始または停止
- キュー・マネージャー・オブジェクトの属性に関する照会

MQSeries for Compaq NSK にはメッセージの格納方法が複数備えられており、これらはメッセージのエンキューの際にアプリケーションで持続か非持続のどちらが選択されるか、およびシステム管理者が選択するキュー・レベルの構成を基にしています。

持続メッセージ

持続メッセージは常にディスクに格納されます。システム管理者は、キュー単位で持続メッセージの記憶技法を 2 種類のうちから選択できます。どちらのタイプの記憶技法を使用するかは、メッセージのサイズを基にして選択します。

その理由は、非常に大きなメッセージのメッセージング操作のパフォーマンスを向上させる目的で、複数の記憶技法が実装されているからです。

すべての持続メッセージについて、1つのレコードが監査キュー・ファイル中にあります。このレコードにはヘッダーとメッセージに関する重要な制御情報が含まれており、最大レコード・サイズの 4096 バイト以内に収まるだけのメッセージ・データも含まれます。

TM/MP 制御下で小規模または中規模の持続メッセージを記憶するのに最も速く効率の良い機構は、MQSeries for Compaq NSK バージョン 2.2.0.1 で使用されているものと同じ基本手法を使用して、複数レコードの形式でキュー・オーバーフロー・ファイル中にオーバーフロー・メッセージ・データを格納するというものです。これらのメッセージのデータはすべて TM/MP にログ記録されるので、必要に応じて監査証跡から完全な形でリカバリーすることができます。

大規模メッセージ (約 200 KB を超えるデータ) の場合、未監査の専用メッセージ・オーバーフロー・ファイルを使用することが最も効率的なメカニズムになります。したがって、メッセージ・オーバーフロー・ファイルに書き込まれるデータは、TM/MP 監査証跡には書き込まれないので、CPU とディスク入出力を節約でき、大規模ブロックの形式で転送できます。

上記の 2 つの機構の間におけるデータ保全性の性質のわずかな違いが、表6 に示されています。

表6. キュー・オーバーフローとメッセージ・オーバーフローの比較

データ保全性の性質	キュー・オーバーフロー方式	メッセージ・オーバーフロー方式
TM/MP によって監査されるデータの量	すべてのメッセージ・データが監査される	Q ファイル・レコードに適合するデータだけが監査される
複数の障害が起きて全ボリュームが失われた場合の、監査証跡からのデータ・リカバリーの可能性	監査証跡からメッセージ全体をリカバリーできる	監査証跡からメッセージ・ヘッダーとデータの先頭部分だけをリカバリーできる

持続データと非持続データ

表6. キュー・オーバーフローとメッセージ・オーバーフローの比較 (続き)

データ保全性の性質	キュー・オーバーフロー方式	メッセージ・オーバーフロー方式
メッセージのサイズの上限	ENSCRIBE によって生じるトランザクション当たりのボリューム当たりのレコード・ロック数によって限定される。非区分ファイルの場合、デフォルトの ENSCRIBE 構成では約 20 MB です。物理メモリの制限のために、実際の限度はこの値より小さくなることがあります。	使用可能なディスク・スペースまたは 100 MB (MQSeries のメッセージ・サイズの上限) だけに限定される
リモート・データベース機能 (RDF) との互換性	完全互換 (MQRDF 環境パラメーターを指定してください)	RDF とは非互換。メッセージ・データが監査されないため、RDF を使用して、メッセージ・ファイルを使ったメッセージ操作をバックアップ・システムに伝搬できません。
ディスク・ハードウェア障害に対する耐障害性	ミラーリングされたディスクを使用している場合は、すべての単一障害ポイントに対して完全に耐障害性あり。すべてのデータが監査されるので、ミラーリングされた対のディスクが両方も障害が生じた場合でもファイル・リカバリーを実行できます。ミラーリングされていない場合は、ディスクだけ耐障害性あり。	ミラーリングされたディスクを使用してメッセージ・オーバーフロー・ファイルを保持している場合は、すべての単一障害ポイントに対して完全に耐障害性あり。全ボリュームに障害が起きた場合はメッセージ・データをリカバリーできません。

要約すると、持続メッセージの場合は、どちらの記憶技法でも非常に高水準のデータ保全性を実現できますが、メッセージ・オーバーフロー・ファイルについては、監査されないメッセージ・データもあるので多少の制限が生じます。通常、この種の制限は非常に大きなメッセージの場合しか関係しませんが、セグメント化を使用して (アプリケーション・メッセージをさらに小さな物理メッセージに分割することにより)、この種の問題を処理することができます。

非持続メッセージ

非持続メッセージ (NPM) は通常はメモリー内に格納されます。キュー・サーバーが大量の非持続データを管理しており、しきい値に達して仮想メモリーが使い果たされる間際になると、キュー・サーバーにより強制的にディスクに NPM が入れられます。構成に対する変更がオンラインで行われている間は、特定の管理操作の場合にキュー・サーバーによりディスク・ファイルに NPM がコピーされることもあります。アプリケーションにより NPM がアクセスされる時点では、NPM は常にメモリー内にあります。

キュー・サーバーにより、作業対象のキューについてすべてのメッセージの記憶が管理され、NPM に適用されるデータ保全性のレベルを制御する構成オプションがキューごとに備えられます。キュー・サーバーによりバックアップ・プロセスに NPM のチェックポイントが指定されるかどうかは構成パラメーターで制御されるので、NPM が主キュー・サーバー・プロセスを含む CPU の障害に対して持続メッセージと同程度の耐性があるようにします。

NPM チェックポイント機能を使用すると、キュー・サーバーにより消費される CPU (主およびバックアップ) が増え、チェックポイント時にバックアップに転送されるデータが多くなり、メッセージを再度格納する必要が生じてからバックアップ・プロセスで消費されるメモリーが増えるという代償が生じます。

キューに NPM チェックポイント機能を使用できない場合に、主キュー・サーバー・プロセスが (プロセスか CPU の障害のために) 異常終了すると、引き継ぎ以前にそのキュー上にあった NPM は、主プロセス中だけに格納されていたので廃棄されます。

耐障害性ある非持続メッセージのオプションは他の MQSeries プラットフォーム上では使用できず、歴代の製品すべてに一貫してデフォルトでは NPM チェックポイント機能が使用可能になっています (この機能を使用不可にするには、**altmqfls** を使用します)。非持続メッセージがデータ保全性の必要に合うかどうか注意深く考慮する必要があります。合う場合、著しくリソースの使用率が節約され、パフォーマンスが向上するという利点を活用できるようにする必要があります。

データベースの一貫性

データベースの一貫性を、MQSeries によって内部的にも、また SYNCPOINT (同期点) オプションを使用してメッセージング操作を行う際には外部的にも保持しなければなりません。両方ともデータ保全性と可用性にとって重要です。

データベースの一貫性

内部データベースの一貫性

キュー・マネージャー中の重要なデータベース・ファイルはすべて TM/MP によって監査されます。したがって、キュー・マネージャー・プロセスでこれらのファイルに変更を加えるには、トランザクションを使用しなければなりません。変更内容はすべて TM/MP 監査証跡にログ記録され、(使用する場合は) RDF を使用して 1 つまたは複数の災害時回復システムに複製されます。したがって、高水準な内部データベースのデータ保全性を保証できます。

TM/MP を使用して内部データベースを保護すると、システム障害やソフトウェア障害が起きても、重要なデータベースの保全性が損なわれないようにするのに役立ちます。したがって、これらのデータベースを使用するプロセスやサービスの再始動 (自動または手動) が正常実行される可能性が高くなるので、高可用性を実現できます。

外部データベースの一貫性

TM/MP を MQSeries 内部とアプリケーションで使用すると、外部データベースに対する変更を調整したり、外部データベースと MQSeries データベースとの一貫性を調整したりできます。同期点制御下で MQSeries のメッセージング操作を行うこともできます。この場合、アプリケーションにアクティブな現行トランザクションが必要です (別のプロセスから継承するか、TM/MP BEGINTRANSACTION サービスを使用して開始する)。このトランザクションはキュー・マネージャーによって継承され、必要なストレージ更新や重要データベースの更新はこのトランザクションを使用して実行されます。TM/MP により、いかなるプロセスで更新が実行されても、このトランザクションの下で実行されるすべてのディスク入出力について、該当する監査証跡のエントリが確実に記録されます。

また更新が実行されたものの、すべてのデータベース・ファイル中の影響を受けるレコードのロックを保留してコミットされていない場合、TM/MP によりその更新のビューの一貫性が保たれます。たとえば、TM/MP の制御下で SQL テーブル中に挿入された行は、トランザクション・コミットの時点まで更新したり削除したりできません。

キュー・マネージャーからアプリケーション・プロセスに応答がある (および MQI から戻る) 際に、アプリケーションでは引き続きこのトランザクションの下で追加作業を行うこともできます。MQSeries を使用してさらにメッセージのエンキューかデキューを行うか、ENSCRIBE か NonStop SQL を使用して独自のデータベース更新を実行します。アプリケーションで MQSeries キューに変更を加えるようになっている場合に、他のデータベースが持続的なものであ

れば、TM/MP の ENDTRANSACTION サービスが呼び出され、すべてのデータベースに対する変更内容がシステム全体で同時にコミットされます。

アプリケーションでトランザクションの処理中にエラーが起きたことが判別された場合に、データベースに対して多少の更新が実行済みであれば、そのアプリケーションで ABORTTRANSACTION を呼び出して、TM/MP によりすべてのデータベースに対する変更内容をシステム全体で同時にバックアウトする必要があります。こうすると、たとえば同期点 MQGET 操作でメッセージがデキューされた後にキュー上でそのメッセージが置き換えられ、また SQL データベース・テーブル中に挿入される前に除去されます。アプリケーション設計の際にこの点に注意すると、これらのエラーに対処して一貫性を保ち、データ保全性とアプリケーションの可用性を拡張できます。

MQSeries で同期点 MQPUT または MQGET 操作が実行される際には、トランザクションが最終的にはコミットされるという前提で、この操作の時点でのキュー項目数が調整されます。したがって、キュー項目数には、キュー上のコミットされていないメッセージとコミット済みのメッセージの数が含まれます。トランザクションが最終的に打ち切られる（故意または障害のため）と、MQSeries でキュー項目数が調整され、完全に正確に値が保守されます。これは MQSeries on Compaq NSK V2.2 以前のリリースから向上した点です。以前のリリースでは、キュー・マネージャーでトランザクションの結果を判別できず、いかなる条件の下でも正確な項目数を保守することができませんでした。

『OpenTMF』には、この操作を実行できる機構について説明されています。

OpenTMF

OpenTMF とは、NonStop TM/MP 製品に対する新しい内部インターフェースの略式名です。この製品は、キュー・マネージャーで同期点メッセージング操作に使用されたトランザクションの結果を判別するために、MQSeries で Compaq の補助の下に使用されます。

この TM/MP の新機能は、NonStop TM/MP の異種トランザクション処理機能を導入するための基礎として、D42 で導入されました。OpenTMF を使用すると、MQSeries によってアプリケーションから継承されたトランザクション中で参加者として登録できます。登録後、TM/MP から MQSeries にトランザクションの完了時に通知が送信され、そのトランザクションが正常に完了したかバックアウトされたかが知らされます。MQSeries では、この TM/MP からの新しい通知を使用して以下のことを行えます。

- いかなる条件下でも正確なキュー項目数を保持する。
- ローカル・キューに関係した内部状況情報を正確に保つ。

データベースの一貫性

- 同期点操作に関係した非持続メッセージの可用性を制御する。
- 待機している MQGET 操作の効率と応答時間を改善する。

システム管理者の立場から OpenTMF の使用状況を認識できる方法は 1 つだけです。TMFCOM ユーティリティの STATUS RESOURCEMANAGER コマンドを使用して作成できるリソース・マネージャーのリスト中で、MQSeries プロセスを識別できます。このリスト中で、OpenTMF を使用するすべての MQSeries プロセスは、TM/MP によって自動的に名前を付けられて、VOLATILE リソース・マネージャーとして示されます。

この TM/MP の新しい使用法では特別な管理処置は必要ありません。MQSeries により自動的に使用され管理されます。TM/MP の RMOOPENPERCPU (CPU 当たりの VOLATILE および RECOVERABLE リソース・マネージャーの最大数) 構成パラメーターが、システム全体で単一の CPU で実行できるキュー・サーバーと状況サーバーの最大数より大きな集合に設定されていることを確認しなければなりません。これらのサーバーは NonStop プロセスのペアなので、Backup プロセスを行える必要があることに注意してください。通常、デフォルト値の 128 は、ほとんどのインストール状況に適しています。*Compaq NSK NonStop TM/MP Configuration and Planning Guide* には、リソース・マネージャーと異種トランザクション処理について説明されています。

NonStop Tuxedo

NonStop Tuxedo は、NonStop TM/MP を基にしており、MQSeries と同じ異種トランザクション処理機能を使用するので、MQSeries でこの製品を使用して OSS アプリケーションのメッセージング操作を調整できます。

Compaq NSK NonStop Tuxedo System Application Development Guide には、NonStop Tuxedo トランザクション環境の使用法と、TM/MP と相互作用する方法が説明されています。

インターリーブ・アプリケーション・トランザクション

MQSeries for Compaq NSK を使用すると、他のプラットフォームでは得られない固有トランザクション環境の利点をアプリケーションで活用できます。通常 Compaq NSK では、1 つのプロセスで複数のトランザクションを並行して管理できます。監査データベースに対する更新は常に現行トランザクションの下で実行され、アプリケーションで他のアクティブ・トランザクションの 1 つに切り替えてから、すべてのトランザクションをコミットできます。このようにして、アプリケーションで同期点メッセージング操作を並行して実行できます。

MQSeries の重要なデータベース・ファイル

表7 に、MQSeries の重要な監査データベース・ファイルについて説明されています。

表7. 重要な監査データベース・ファイル

記述名	場所 / 名前	使用法
オブジェクト・カタログ	データ・サブボリューム / OBJCAT および ABJCAT	キュー・マネージャー自体に加えて、個々のキュー、プロセス、および名前リスト・オブジェクトの属性も保持します。
非クライアント・チャンネル定義	データ・サブボリューム / CHDEFS	個々の非クライアント・チャンネル (SENDER、RECEIVER、SERVER、REQUESTER、および CLUSTER チャンネル・タイプ) の属性を保持します。
クライアント・チャンネル定義	データ・サブボリューム / CCHDEFS	個々のクライアント・チャンネル (SVRCONN チャンネル・タイプ) の属性を保持します。
OAM データベース	データ・サブボリューム / OAMDB	個々のオブジェクトに関する許可 (アクセス権) と、キュー・マネージャーへのアクセスを許可されている OAM プリンシパルを保持します。
基本データベース	データ・サブボリューム / PRIDB および PRIDBA	許可されている個々の OAM プリンシパルの名前と、そのプリンシパルに対応している Guardian ユーザー ID の名前を保持します。
EC 制御ファイル	データ・サブボリューム / RUNTIME	EC Boss および EC によって使用される情報を保持し、始動操作やリカバリー操作を調整します。

データベースの一貫性

表7. 重要な監査データベース・ファイル (続き)

記述名	場所 / 名前	使用法
名前リストの定義	データ・サブボリューム / Lxxxxxxx	定義されている個々の名前リスト・オブジェクトの内容を保持します。
キュー・ファイル	メッセージ・サブボリューム / Qxxxxxxx	ローカル・キュー上のすべての持続メッセージについて、1つのヘッダー情報と多少のデータを保持します。
キュー・オーバーフロー・ファイル	メッセージ・サブボリューム / Oxxxxxxx	ローカル・キュー上の中規模～大規模な持続メッセージすべてのデータを保持します。
オブジェクト・タッチ・ファイル	データおよびメッセージ・サブボリューム / Txxxxxxx	任意のオブジェクトの属性に対する管理変更を検出するのに使用します。
チャンネル同期ファイル	同期サブボリューム / Sxxxxxxx	チャンネル・インスタンスのチャンネル同期情報を保持します。

重要なプロセス

表8には、キュー・マネージャーの重要なプロセスが説明されており、ソフトウェアやシステムの障害から MQSeries が保護される方法やリカバリーする方法を示します。以下の表で、災害とは、複数のシステムの障害または全システムの破損を指します。

表8. 重要なプロセスで使用される保護方式

プロセス	使用される保護方式	リカバリー処理
キュー・サーバー	NonStop プロセスのペア。災害の場合を除いて、すべての時点のローカル・キューの正確な状況とメッセージを保守します。	壊滅的な障害の後に、監査データベースから再初期設定します。その他のリカバリーは必要ありません。災害の場合は NPM が失われます。

表 8. 重要なプロセスで使用される保護方式 (続き)

プロセス	使用される保護方式	リカバリー処理
状況サーバー	NonStop プロセスのペア。災害の場合を除いて、すべての時点の非ローカル・キュー・オブジェクトの正確な状況とチャンネルの状況を保守します。	壊滅的な障害の後に、監査データベースから再初期設定します。その他のリカバリーは必要ありません。
ローカル・キュー・マネージャー・エージェント (LQMA)	<p>アプリケーションの接続は切断としてマークが付けられます (2009)。</p> <p>リポジトリ・マネージャーのガーベッジ・コレクションでは、プロセスの障害の後に残ったクラスター・キャッシュの登録域がクリーンアップされます。</p> <p>EC、EC Boss、キュー・サーバー、および状況サーバーにより、NSK IPC 接続を介して障害が即時に認識され、状況データに該当する訂正 / 調整が加えられます。</p> <p>TM/MP により、障害の時点でアクティブになっておりプロセスに使用されていたすべてのアクティブ・トランザクションが打ち切られます。</p>	なし。1 つの LQMA のサービス対象は 1 つの接続です。障害の際に接続は除去されるので、アプリケーションによって新しい接続を開始しなければなりません。

重要なプロセス

表8. 重要なプロセスで使用される保護方式 (続き)

プロセス	使用される保護方式	リカバリー処理
メッセージ・チャンネル・エージェント	<p>状況サーバーにより NSK IPC 接続を介して即時に障害が通知され、該当するチャンネル状況のマークが付けられます。</p> <p>状況サーバーにより、障害が起こったアウトバウンド・チャンネルが確実に再試行されます。</p> <p>MCA 採用機能を使用して、障害が起こったチャンネルを再始動できます。</p> <p>チャンネル同期データは、TM/MP によって監査され、チャンネルの保全性を保持するために MQSeries で使用されます。まれな未確定状態の事例として自動的に解決できない場合がありますが、この場合には標準的な解決用の管理機能があります。</p> <p>その他の保護方式はすべて LQMA プロセスと同じです。</p>	<p>なし。1 つの MCA のサービス対象は 1 つの接続です。障害の際にチャンネルは停止し、チャンネルを再始動する MQSeries の自動化機能があります。</p> <p>MQSeries for Compaq NSK V5.1 には、チャンネル・ハートビートやクラスター化などの、チャンネルの可用性を向上させる新機構も導入されています。</p>
チャンネル・イニシエーター	<p>複数のイニシエーターを構成して、複数のチャンネル・イニシエーター間でチャンネル開始キューを分散させることにより、可用性を高めることができます。</p> <p>チャンネル・イニシエーターは、通常 PATHWAY サーバー・クラスとして実行されます。これは障害発生時に固定値の 10 分以内に複数回 (デフォルトは 10) AUTOSTART を実行するよう構成されています。</p> <p>標準的な PATHWAY 構成オプションを使用して、CPU 障害が起きたら代替 CPU を使用するよう構成できます。</p>	<p>なし。デフォルトの状況サーバーでは、すべての条件下の正確なチャンネル状況が保守され、チャンネル再試行が行われます。チャンネル・イニシエーターでは、MQSeries のトリガー機能を使用して開始操作が実行されます。</p>

表 8. 重要なプロセスで使用される保護方式 (続き)

プロセス	使用される保護方式	リカバリー処理
コマンド・サーバー	<p>コマンド・サーバーでは、同期点で管理コマンドが実行されるので、一貫性が保守されます。</p> <p>チャンネル・イニシエーターと同様の標準的な PATHWAY 機能。</p>	<p>なし。再始動すると、キュー・マネージャーに対して新しい接続が行われます。</p>
EC Boss	<p>前述の標準的な PATHWAY 機能。</p>	<p>EC Boss と EC により、監査済みの RUNTIME ファイルを使用してリカバリーが調整されます。EC により EC Boss が再登録され、処理が継続されます。</p>
EC	<p>前述の標準的な PATHWAY 機能。</p>	<p>EC と EC Boss により、監査済みの RUNTIME ファイルを使用してリカバリーが調整されます。EC により EC Boss が再登録され、処理が継続されます。</p>
TCP/IP リスナー	<p>複数の TCP/IP リスナー・プロセスを構成して、複数のポート、IP アドレス、または TCP/IP サーバー・プロセス間でチャンネルを分散させることにより、可用性を高めることができます。</p> <p>前述の標準的な PATHWAY 機能。</p>	<p>なし。TCP/IP リスナーにより、再始動時にキュー・マネージャー用に構成された使用可能なポートへの接続が試行されます。</p>

重要なプロセス

表 8. 重要なプロセスで使用される保護方式 (続き)

プロセス	使用される保護方式	リカバリー処理
<p>リポジトリ・マネージャーまたはリポジトリ・キャッシュ・マネージャー</p>	<p>リポジトリ・マネージャーまたはリポジトリ・キャッシュ・マネージャーに障害が起きても、同じ CPU 中のキャッシュの現行ユーザーや新規ユーザーは、キャッシュへのアクセスやクラスター化操作により中断させられることはありません。</p> <p>リポジトリ・マネージャーにより、常にリポジトリ・キュー上で一貫性があり強化されたクラスター・キャッシュが保守されます。変更を加える際には、変更を実行するクラスター・コマンド・キューからのメッセージの MQGET との同期点で加えます。したがって、強化中にリポジトリ・マネージャーに障害が起こっても、一貫性が保守されます。</p> <p>前述の標準的な PATHWAY 機能。</p>	<p>リカバリーの際に、キュー・マネージャー中にリポジトリ・マネージャーがない場合は、リポジトリ・プロセスがリポジトリ・マネージャーになります。ある場合は、実行場所の CPU のリポジトリ・キャッシュ・サーバーの役割が想定されます。</p> <p>リポジトリ・マネージャーまたはリポジトリ・キャッシュ・マネージャーが依然としてメモリー内にある場合は、キャッシュに再接続されます。ない場合は、ディスクがあればそのディスクからキャッシュが再ロードされます。ディスク・ファイルがない場合は、CPU 中に新しいキャッシュが作成され、リポジトリ・キューから初期設定されます。</p>

表 8. 重要なプロセスで使用される保護方式 (続き)

プロセス	使用される保護方式	リカバリー処理
キュー・マネージャー・サーバー	<p>キュー・マネージャー・サーバーでは、有効期限が切れたメッセージの検索や、有効期限切れレポートの生成が同期点で実行されるので、サーバーに障害が起きても一貫性は保たれます。</p> <p>前述の標準的な PATHWAY 機能。</p>	<p>キュー・サーバーでハウスキーピングを実行するたびに報告する必要がある現行の有効期限切れメッセージの集合が、それらのキュー・サーバーによって報告されると、再始動されたキュー・マネージャー・サーバーがそれらのキュー・サーバーと自動的に再同期されます。</p>

重要なプロセス

表 8. 重要なプロセスで使用される保護方式 (続き)

プロセス	使用される保護方式	リカバリー処理
アプリケーション - STANDARD 結合	<p>切断された接続か、または MQI 呼び出しから戻る予期しないエラーによって MQSeries 中の障害が検出されます。キュー・マネージャーとの接続を再確立するには、アプリケーションで MQDISC を呼び出してから再度 MQCONN を呼び出す必要があります。</p> <p>キュー・マネージャーにより、アプリケーション・プロセスの障害が即時に検出されます。キュー・マネージャーにより、すべてのオープン・リソースのクローズと解放を引き起こしたアプリケーションに代わって、暗黙的に MQDISC が実行されます。</p> <p>進行中の同期点操作はすべて TM/MP によって打ち切れ、MQSeries によりローカル・キューに該当する調整が実行されてやり直されます。</p> <p>アプリケーションのアドレス・スペース中の共用メモリーだけが読み取り専用になり、重要なデータベース・ファイルは直接アクセスされないため、エラーを含むアプリケーション・コードによりキュー・マネージャーの重要なデータベースや共用リソースが破壊されることはありません。</p>	なし。アプリケーションからキュー・マネージャーに新たに接続されます。

表 8. 重要なプロセスで使用される保護方式 (続き)

プロセス	使用される保護方式	リカバリー処理
<p>アプリケーション - FASTPATH 結合</p>	<p>MQI 呼び出しから戻る予期しないエラーによって MQSeries 中の障害が検出されます。キュー・マネージャーとの接続を再確立するには、アプリケーションで MQDISC を呼び出してから再度 MQCONNX を呼び出す必要があります。</p> <p>キュー・マネージャーにより、アプリケーション・プロセスの障害が即時に検出されます。キュー・マネージャーにより、すべてのオープン・リソースのクローズと解放を引き起こしたアプリケーションに代わって、暗黙的に MQDISC が実行されます。</p> <p>進行中の同期点操作はすべて TM/MP によって打ち切れ、MQSeries によりローカル・キューに該当する調整が実行されてやり直されます。</p> <p>エラーを含むアプリケーション・コードによりキュー・マネージャーの重要なデータベースや共用リソースが破壊されることがあります。アプリケーションには読み取り / 書き込みリポジトリ・キャッシュや内部キュー・マネージャーの構造へのアクセス権がありますが、これらはオブジェクト・カタログを破壊する可能性があるからです。キュー構造やメッセージ自体にはキュー・サーバーしかアクセスしないので、これらは安全であることに注意してください。</p>	<p>なし。アプリケーションの再始動時に新たに接続されます。</p>

クラスター

MQSeries for Compaq NSK V5.1 クラスターの目的は、MQSeries ネットワークの管理要件を減らし、MQSeries の分散サービスとしての総合的な可用性や拡張容易性を拡張することにあります。

クラスターに属するキュー・マネージャーは、クラスターにローカル・キューであるかのように公示されるキューに対して MQPUT を実行できます。

MQSeries クラスター化機能は、メッセージを宛先キューに転送するのに必要なすべての定義とチャネルの管理を扱います。

クラスター・キューを、クラスター中の複数のキュー・マネージャー上で定義することもできます。この場合、クラスター中の 1 つのキューのインスタンスが複数作成されます。クラスター・ワークロード・マネージャー (CWLM) という、MQSeries のコンポーネントの 1 つで選択されたキューの 1 つのインスタンスだけに、アプリケーションによる挿入が行われます。この選択は、キューのオープン時に行われるか、または挿入のたびに動的に行われます。

CWLM により、クラスター・キューのインスタンスのうち使用するのに最適なものが判別されます。この判別は、インスタンスに対するチャネルが実行中かどうか、およびネットワーク優先順位やクラスター・ワークロード管理出口を介したアプリケーションの考慮事項など他の特定の要因を基にして行われます。

したがって、クラスターにより MQSeries ネットワーク・レベルの可用性を拡張できます。Compaq NSK 上での MQSeries の操作の信頼性や拡張容易性を考えると、この種の MQSeries をクラスターのフル・リポジトリとして使用することは良いことです。

可用性に関する構成上の考慮事項

以下に、MQSeries for Compaq NSK V5.1 と Compaq NSK 上のそのアプリケーションの可用性を拡張する構成オプションについて要約します。

- すべてのサーバー・クラスを CPU 障害に対して保護するには、代替 CPU を指定して PATHWAY を構成してください。
- MQSeries ネットワーク・リソースの可用性を拡張するには、クラスター化の使用を考慮してください。
- パフォーマンスと可用性を高くするには、適切なメッセージ・タイプの場合に、チェックポイント機能を使用可能にして非持続メッセージを使用することを考慮してください。

- TM/MP 構成が、MQSeries とそのアプリケーションのピーク予見要求を処理できるサイズになっているか確認してください。
- 大規模メッセージの場合は、メッセージ・オーバーフロー・ファイルを使用して、監査証跡の必要性を小さくしてください。
- 災害時回復の必要がある場合は、RDF を使用して、MQSeries のバックアップ場所を作成して保守してください。
- 使用可能な物理メモリーが十分ある状態で CPU を実行して、MQSeries とそのアプリケーションのピーク要求を処理してください。
- MQSeries とそのアプリケーションを保持する CPU が使用できるスワップ・スペースが十分あるか確認してください。

データ保全性に関する構成上の考慮事項

以下に、MQSeries for Compaq NSK V5.1 と Compaq NSK 上のそのアプリケーションのデータ保全性を拡張する構成オプションについて要約します。

- MQSeries によってメッセージやトランザクションのタイプごとに、持続メッセージと非持続メッセージによって備えられるデータ保全性の必須レベルを判別してください。
- 使用する必要のある非常に大規模なメッセージを格納するのに、メッセージ・オーバーフロー・ファイルが、ストレージ機構として適切か判別してください。
- 非持続メッセージにチェックポイント機能が必要か、それとも一部のメッセージだけに必要か判別してください。さまざまなチェックポイント機能オプションを指定できる、さまざまなタイプのメッセージをさまざまなキューに挿入する必要があります。
- アプリケーションで最高のデータ保全性が必要な場合に、持続メッセージを使用した同期点操作が使用されているか確認してください。
- 災害時回復の必要がある場合は、RDF を構成して使用し、MQSeries 環境の複製バックアップを作成して保守してください。

データ保全性に関する構成上の考慮事項

第2部 リファレンス

第17章 MQSeries 制御コマンド	323
制御コマンドの要約	323
名前の使用	324
構文図の見方	325
altmqfls (キュー・ファイル属性の更新).	327
altmqusr (MQSeries ユーザー情報の更新)	333
cleanrdf (RDF ハウスキーピングの実行)	335
cnvclchl (クライアント・チャンネル定義の変換).	337
crtmqcvx (データ変換)	339
crtmqm (キュー・マネージャーの作成).	342
dltmqm (キュー・マネージャーの削除).	347
dspmqaout (権限の表示)	349
dspmqcsv (コマンド・サーバーの表示)	354
dspmqls (MQSeries ファイル属性の表示)	356
dspmqtrc (MQSeries 定様式トレース出力の表示).	360
dspmqusr (MQSeries ユーザー情報の表示)	361
endmqcsv (コマンド・サーバーの終了).	363
endmqm (キュー・マネージャーの終了)	365
endmqtrc (MQSeries トレースの終了)	368
instmqm (MQSeries for Compaq NSK のインストール)	370
runmqchi (チャンネル・イニシエーターの実行)	371
runmqchl (チャンネルの実行).	372
runmqdlq (送達不能キュー・ハンドラーの実行).	373
runmqlsr (リスナーの実行).	375
runmqsc (MQSeries コマンドの実行).	377
runmqtrm (トリガー・モニターの開始).	381
setmqaut (権限の設定 / リセット)	382
strmqcsv (コマンド・サーバーの開始)	389
strmqm (キュー・マネージャーの開始).	390
strmqtrc (MQSeries トレースの開始).	392
upgmqm (V2.2.0.1 キュー・マネージャーのアップグレード)	396

第17章 MQSeries 制御コマンド

この章は、MQSeries for Compaq NSK で使用する制御コマンドについての参照資料です。

制御コマンドの要約

以下の制御コマンドは、TACL マクロおよびコンパイル済みプログラム経由で、MQSeries for Compaq NSK によってサポートされています。

- altmqfls (キュー・ファイルの属性の更新)
- altmqusr (MQSeries ユーザー情報の更新)
- cleanrdf (RDF ハウスキーピング・ユーティリティー)
- cnvclchl (クライアント・チャンネル定義の変換)
- crtmcvqx (データ変換)
- crtmqm (キュー・マネージャーの作成)
- dlrmqm (キュー・マネージャーの削除)
- dspmqaut (権限の表示)
- dspmqcsv (コマンド・サーバーの表示)
- dspmqfls (MQSeries ファイル属性の表示)
- dspmqtrc (MQSeries 定様式トレース出力の表示)
- dspmqusr (MQSeries ユーザー情報の表示)
- endmqcsv (コマンド・サーバーの終了)
- endmqm (キュー・マネージャーの終了)
- endmqtrc (MQSeries トレースの終了)
- instmqm (MQSeries for Compaq NSK のインストール)
- runmqchi (チャンネル・イニシエーターの実行)
- runmqchl (チャンネルの実行)
- runmqdlq (送達不能キュー・ハンドラーの実行)
- runmqslr (TCP/IP リスナーの実行)
- runmqsc (MQSeries コマンドの実行)
- runmqtrm (トリガー・モニターの開始)
- setmqaut (権限の設定 / リセット)
- strmqcsv (コマンド・サーバーの開始)
- strmqm (キュー・マネージャーの開始)
- strmqtrc (MQSeries トレースの開始)
- upgmqm (V2.2.0.1 キュー・マネージャーのアップグレード)

制御コマンド

これらのコマンドについての詳細は、この章で後程説明します。

注:

1. ダッシュと単一文字の ID から成るフラグ (たとえば、**runmqsc** コマンドの **-v** など) は、小文字で指定しなければなりません。
2. 制御文字が **-?;** と共に呼び出される場合、またはパラメーターが予期されるときにパラメーター無しで呼び出される場合、使用法に関するメッセージが表示されます。

名前の使用

以下の MQSeries オブジェクト名の最大文字数は、48 文字です。

- キュー・マネージャー
- キュー
- プロセス定義

チャンネル名の最大長は 20 文字です。

すべての MQSeries の名前に使用できる文字は、次のとおりです。

- 英大文字 A - Z
- 英小文字 a - z
- 数値 0 - 9
- ピリオド (.)
- 下線 (_)
- スラッシュ (/)
- パーセント記号 (%)

注:

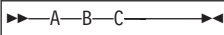
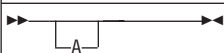
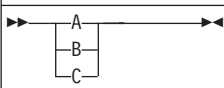
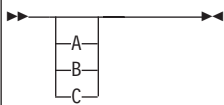
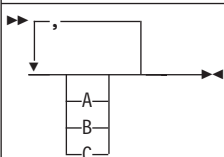
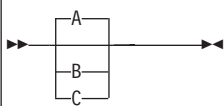
1. スラッシュとパーセント記号は、特殊文字です。名前の中でこの 2 つの文字のいずれかを使用する場合、名前は必ず二重引用符で囲む必要があります。
2. 先行ブランクも組み込みブランクも許されていません。
3. 各国語文字は許されていません。
4. 名前は二重引用符で囲むことができますが、これは、特殊文字が名前に含まれている場合にのみ必要です。

構文図の見方

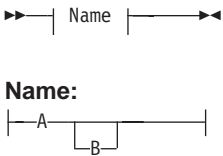
この章では、構文図（「幹線」図ともいう）を示しています。

各構文図は二重の右矢印で始まり、右矢印と左矢印の対で終了します。1つの右矢印で始まる行は、継続行です。構文図を見る場合は左から右、上から下へ、矢印の方向に従って進みます。

構文図で使用される他の規則は次のとおりです。

規則	意味
	値 A、B、および C を指定しなければなりません。必須の値は、構文図の主線上に示されます。
	値 A を指定できます。オプションの値は構文図の主線の下に示されます。
	値 A、B、および C は代替であり、いずれかの値を指定しなければなりません。
	値 A、B、および C は代替であり、いずれかの値を指定することができます。
	値 A、B、および C のいずれかまたは複数の値を指定することができます。複数值または繰り返し値の必須の区切り記号（この例ではコンマ (,)）はどれも矢印の上に示されます。
	値 A、B、および C は代替であり、いずれかの値を指定することができます。値をどれも指定していない場合は、デフォルト値 A（主線の上に示される値）が使用されます。

名前

規則	意味
 <p>Name:</p>	構文断片 Name は、主構文図とは別に示されます。

altmqfls (キュー・ファイル属性の更新)

目的

altmqfls コマンドは、キュー・ファイル属性を更新するために使用します。1 つの **altmqfls** コマンドが、以下の 3 つの操作のうちの 1 つだけを 1 回実行できます。

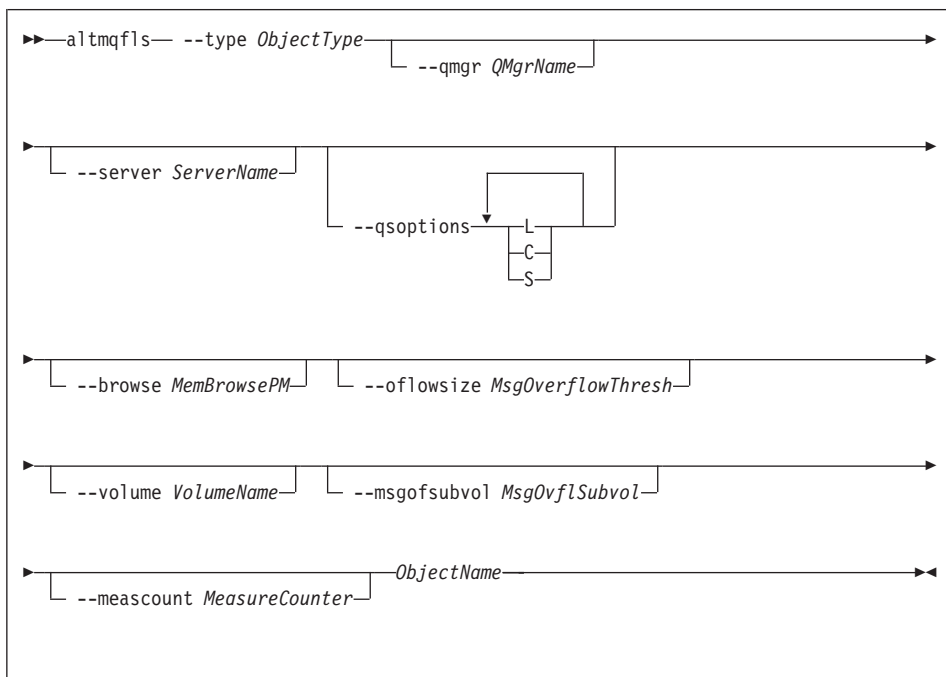
- 事前定義されたローカル・キューに属するメッセージ・ファイルを別のボリュームに移動させて、それらのボリューム間でディスク入出力を分散する。
- ローカル・キューに関連付けられたキュー・ファイルおよびオーバーフロー・ファイルのサイズを変更する。この操作は、キューがオープンしている場合には実行できません。
- ローカル・キューに関連付けられたキュー・サーバー・オプションを変更する。これらのオプションを使用すると、キュー・サーバーがキューに関連したストレージを処理したり、非持続メッセージのチェックポイント機能を制御したりする方法を最適化することができます。さらに、ローカル・キューに関連付けられたキュー・サーバー、あるいは別名、リモート・キュー、モデル・キュー、またはプロセスに関連付けられた状況サーバーを変更したり、測定カウンターをキューに関連付けたりすることができます。

altmqfls は、キューの使用中にそれを再ロードすることを許可していません。

注: **altmqfls** のユーザー・インターフェースおよびコマンド行オプションは、バージョン 2.2.0.1 からかなり変更されています。変更は新しい機能を反映しており、コマンド行オプションにさらに意味のある名前を付けています。また、すべてのパラメーターは、POSIX ルールに準拠するために '...' で始まります。 **altmqfls** を呼び出すスクリプトがこれらの変更を反映するように更新する必要があります。なぜなら、MQSeries バージョン 5.1 はバージョン 2.2.0.1 のコマンド行オプションを拒否するからです。

altmqfls

構文



必須パラメーター

ObjectName

そのメッセージ・ファイルが再配置される持続ローカル・キューの名前です。キューはオープンしていないか、またはそこにコミットされていないメッセージが含まれていなければなりません。

--type *ObjectType*

持続キューを識別します。 *ObjectType* は必ず指定しなければなりません。それは以下のいずれかになります。

ql または **qlocal**

ローカル・キュー

qa または **qalias**

別名キュー

qr または **qremote**

リモート・キュー

qm または **qmodel**

モデル・キュー

proc または process プロセス

オプション・パラメーター

--qmgr *QMgrName*

ローカル・キューが属するキュー・マネージャーの名前です。キュー・マネージャーはすでに開始されている必要があります。キュー・マネージャー名を指定しない場合、デフォルト・キュー・マネージャーが使用されます。

--server *ServerName*

このオブジェクトの状況データを担当する状況サーバーのプロセスまたはキュー・サーバーのプロセスの名前です。このオブジェクトがローカル・キューまたはモデル・キューの場合は、キュー・サーバー名を使用します。オブジェクトがローカル・キューまたはモデル・キュー以外の場合は、状況サーバー名を使用します。

オブジェクトが作成される際に、*ServerName* は DEFAULT に設定され、デフォルト・サーバーがそのキューを担当することを示します。キューの担当が変わるとき、キューは未使用状態になっていなければならず、変更中にすべての非持続メッセージは破棄されます。

--volume *VolumeName*

Compaq NSK ポリウム名です (たとえば、\$DEV)。この値は、altmqfls を使用してメッセージ・ファイルを別のポリウムに移動させる場合に必須です。このパラメーターを指定する際には、タイプとオブジェクト名だけを一緒に使用することができます。他のオプションと組み合わせることはできません。

--qsoptions *LSC*

L、S、および C オプションを使用して (単独でも組み合わせでも可)、キューの検索可能性および信頼性を微調整します。

注: コマンドを発行するたびに、すべての --qsoptions SLC が設定されます。たとえば、--qsoptions S は L と C の設定を解除します。

--qsoptions オプションはコマンド行で 1 回限り指定することができます。オプションは少なくとも 1 つ以上指定しなければなりません。

C メッセージの保管に必要な余分のチェックポイント、余分の IPC メッセージ、および余分のメモリーを処理するために CPU の負荷が求められるにもかかわらず、フォールト・トレランスを提供して、非持続メッセージのチェックポイントがバックアップ・キュー・サーバーに対してとられることを指定します。

このオプションは、非持続メッセージに高度な回復可能性が必要な場合に使用します。キューの作成時にこのパラメーターのデフォルトが設定されます (つまり、チェックポイントがとられます)。

- L** キュー・サーバーが、キューに関連したデータ構造およびチェーンをメモリー内にロックすることを指定します。通常、キューに関連付けられたストレージは、アクセスされなくなったときにキュー・サーバーのアドレスから除去されます。このオプションは、他のキューをできる限り犠牲にして、特定のキューのメモリー・データ構造に高速でアクセスするために使用します。

キューの作成時にこのパラメーターのデフォルトは設定されません (つまり、メモリー内にロックされません)。

- S** キュー・マネージャーに始動時に、キュー・サーバーがローカル・キューをディスクからキャッシュにロードすることを指定します。通常、キューに関するメッセージは、アプリケーションが最初にそれを参照するときにロードされます。このオプションが設定されている場合、キュー・マネージャーが始動すると、キューがロードされます。キューの作成時にこのパラメーターのデフォルトは設定されません (つまり、始動時にロードされません)。

このオプションは、キュー・マネージャーの始動時にキュー・サーバーのアクティビティー (CPU およびディスク入出力) が増えるのと引き換えに、キューが最初にオープンされる際に CPU およびディスク入出力アクティビティーを減らすために使用します。

- なし オプションが設定されないことを指定します。

--browse *MemBrowsePM*

(ディスク上だけでなく) キュー・サーバーのキャッシュに保持する、それぞれの持続メッセージのデータの最大バイト数を指定します。持続メッセージのブラウズ操作中に、キュー・マネージャーは通常、メッセージに関するデータをディスク装置から読み取り、それをアプリケーションに戻します。また、このパラメーターがゼロ以外の値に設定されている場合、指定されたバイト数のデータがメモリーに保持され、ブラウズ操作によりディスクにアクセスしなくてもこのデータがアプリケーションに戻されます。このパラメーターを使用すると、キュー・サーバーによって使用されるメモリー・リソースを増やすことができます。このパラメーターの最小値はゼロ (0) バイトで、最大値は 25,000 バイトです。キューの作成時にこのパラメーターのデフォルトはゼロになります。

--oflowsize *MsgOverflowThresh*

メッセージ・データを保管するためのメッセージ・オーバーフロー・ファ

イル用の最小メッセージ・サイズを指定します。このしきい値より小さい持続メッセージは、キュー・オーバーフロー・ファイルに保管されます。このしきい値と同じか、それより大きい持続メッセージは、専用のメッセージ・オーバーフロー・ファイルに大量のデータを保管します。キューの作成時にこのパラメーターのデフォルトは 200,000 バイトになります。

--msgofsubvol *MsgOvflSubvol*

キュー・サーバーが新規のメッセージ・オーバーフロー・ファイルを作成するときに、その作成場所であるキューが存在するボリューム上のサブボリュームを指定します。デフォルトでは、すべてのキューはキュー・マネージャー・メッセージのサブボリュームを最初に使用します。

--meascount *MeasureCounter*

現行の項目数に初期設定され、その後、メッセージが追加および除去される際にキューを担当するキュー・サーバーによって増減される測定カウンターの名前を指定します (実行中の測定の一部である場合)。

戻りコード

- 0 コマンドは正常に終了しました。
- 10 コマンドは完了しましたが、すべて予期したとおりではありません。
- 20 処理中にエラーが起きました。

例

1. 次の例では、ローカル・キュー `flint.queue` に属しており、キュー・マネージャー `target.queue.mgr` によって所有されるメッセージ・ファイルが、ボリューム `$DATA3` に移動されます。

```
altmqfls --qmgr target.queue.mgr --type ql --volume $DATA3 flint.queue
```

2. 次の例では、キュー・サーバーのプロセス名がオブジェクトに対して変更されます。

```
altmqfls --qmgr target.queue.mgr --type ql --server $TQS2 flint.queue
```

このコマンドによって、`target.queue.mgr` に属するローカル・キュー `flint.queue` が、`$TQS2` というプロセス名でキュー・サーバーに移動されます。

altmqfls

関連コマンド

dspmqfls

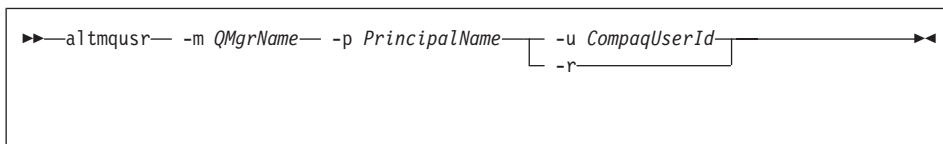
MQSeries ファイルの表示

altmqusr (MQSeries ユーザー情報の更新)

目的

altmqusr コマンドは、MQSeries に対するアクセスを持つ Compaq NSK ユーザー ID に対応するプリンシパルを定義したり、除去したりするために使用します。

構文



説明

このコマンドを使用すると、以下のことを実行できます。

- プリンシパルの作成 (つまり、キュー・マネージャーに対するアクセスを Compaq NSK ユーザー ID に付与する)
- プリンシパルの除去 (つまり、キュー・マネージャーに対するアクセスを Compaq ユーザー ID から取り消す)
- プリンシパル定義の変更

`-u CompaqUserId` が指定されると、**altmqusr** はプリンシパルを作成し、それがまだ存在しない場合は既存の定義を変更します。 `CompaqUserId` は、Compaq 管理ユーザー ID として指定するか、または SAFEGUARD が実行している場合は SAFEGUARD 別名として指定することができます。

`-r` が指定されると、プリンシパルはプリンシパル・データベースから削除されます。

`-u CompaqUserId` か `-r` のどちらかを指定する必要があります。

必須パラメーター

`-m QMgrName`

プリンシパルが属するキュー・マネージャーの名前です。

`-p PrincipalName`

作成、変更、または除去されるプリンシパルの名前です。

altmqusr

-u *CompaqUserId*

Compaq NSK 管理ユーザー ID またはプリンシパル定義に関連付けられる SAFEGUARD 別名です。

-r プリンシパル定義がキュー・マネージャーから除去されることを指定します。

戻りコード

- 0** 正常な操作です。
- 36** 与えられた引き数が無効です。
- 69** ストレージが利用不能です。
- 71** 予期しないエラーです。

例

プリンシパル mquser1 のマッピングを Compaq ユーザー ID mqtest.fred に追加するには、次のようにします。

```
altmqusr -m MT02 -p mquser1 -u mqtest.fred
```

プリンシパル mquser2 のマッピングをグループ group.user01 に追加するには、次のようにします。

```
altmqusr -m MT02 -p mquser2 -u group.user01
```

上記のコマンドの結果を見るには、361ページの『dspmqusr (MQSeries ユーザー情報の表示)』の説明にあるとおりに、**dspmqusr** コマンドを使用します。

プリンシパル mquser1 を除去するには、次のようにします。

```
altmqusr -m MT02 -p mquser1 -r
```

関連コマンド

dspmqusr MQSeries ユーザー情報の表示

cleanrdf (RDF ハウスキーピングの実行)

目的

cleanrdf ユーティリティは、RDF 環境にある 1 次システムのキュー・マネージャーで、日常的なハウスキーピングを実行するために使用します。

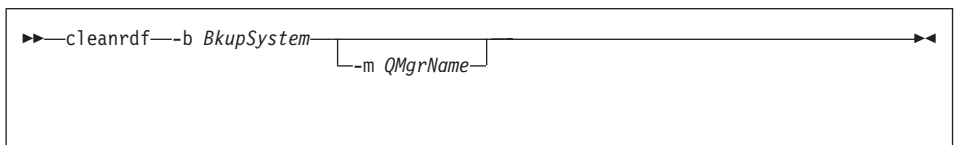
cleanrdf ユーティリティは、1 次システムとバックアップ・システムの両方で論理的に削除されたファイルの除去を完了します。また、一部の監査されていないデータベースをバックアップ・システム上の正しい位置に複写します。

cleanrdf によって呼び出されたユーティリティは、オブジェクト・カタログおよびメッセージ・データベース全体に対して実行するため、このユーティリティの実行時にはパフォーマンスの低下が発生する可能性があります。

ユーティリティによってバックアップ・システムに複写される、監査されないデータベース・ファイルは、次のとおりです。

- MQERRLG1
- MQSINI
- QMINI
- PATHCTL
- SHUTDOWN
- SYNCHIDX
- TRACEOPT
- UMQSINI

構文



必須パラメーター

-b *BkupSystem*

このキュー・マネージャー用の RDF バックアップ・サイトの Compaq NSK システム名です。*BkupSystem* は *¥name* という形式で指定されます (Compaq NSK 環境での標準)。

cleanrdf

オプション・パラメーター

-m *QMgrName*

cleanrdf が実行されるキュー・マネージャーの名前です。キュー・マネージャーの名前を指定しない場合、**cleanrdf** はデフォルト・キュー・マネージャーに対して操作を実行します。

戻りコード

0 コマンドは正常に終了しました。

20 処理中にエラーが起きました。

例

次の例では、**cleanrdf** はキュー・マネージャー `test.queue.mgr` に対して操作を実行します。Compaq NSK ノード ¥HAWK は、このキュー・マネージャーのバックアップ RDF サイトとして構成されました。

```
cleanrdf -b ¥HAWK -m test.queue.mgr
```


cnvclchl (クライアント・チャンネル定義の変換)

目的

cnvclchl コマンドは、MQSC によって CLNTCONN チャンネル用に作成されたクライアント・チャンネル定義を、Compaq 構造化ファイルから、MQSeries クライアントが受け入れ可能な非構造化形式に変換するために使用します。

注:

バージョン 2.2.0.1 の定義ファイルは、レコードの形式が変更されたため、バージョン 5.1 クライアントでは機能しません。MQSeries for Compaq NSK V5.1 をインストールした後、**cnvclchl** コマンドを再実行すると、バージョン 5.1 クライアントで機能する定義ファイルを作成することができます。

構文

```
▶▶—cnvclchl— -m QMgrName —————▶▶  
└── -o OutputFile ─┘
```

必須パラメーター

-m *QMgrName*

変換するチャンネル定義ファイル (CCHDEFS) を所有するキュー・マネージャーを識別します。この値は必須です。

オプション・パラメーター

-o *OutputFile*

変換後の定義を含むファイルを識別します。デフォルトのファイル名は AMQCLCHL です。

例

以下のコマンドは、キュー・マネージャー MT01 用の Compaq 構造化クライアントのチャンネル定義ファイルを、非構造化ファイルに変換します。2 つのクライアント接続チャンネル定義が、出力ファイル

cnvclchl

AMQCLCHL、SYSTEM.DEF.CLNTCONN、および SOLARIS_TO_Cmpaq に含まれています。

```
$DATA01 SZMON 330> cnvclchl -m MV4  
  
MQSeries client channel table being converted  
  
Opening Compaq NSK v5.1 CLNTCONN table  
  
Opening Common v5.1 CLNTCONN table AMQCLCHL for output  
  
Writing Common v5.1 CLNTCONN table entry for SYSTEM.DEF.CLNTCONN  
  
Closing Compaq NSK v5.1 CLNTCONN table  
Closing Common v5.1 CLNTCONN table  
MQSeries client channel table conversion complete.
```

crtmqcvx (データ変換)

目的

crtmqcvx コマンドは、データ・タイプ構造のデータ変換を実行するコード断片を作成するために使用します。このコマンドは、C構造体を変換するために出口で使用できるC関数を生成します。

このコマンドは、変換される構造体を含んでいる入力ファイルを読み取ってから、それらの構造を変換するコード断片を含んでいる出力ファイルを書き込みます。

このコマンドおよび使用方法の詳細については、*MQSeries アプリケーション・プログラミング・ガイド* を参照してください。

構文

```
▶▶—crtmqcvx—SourceFile—TargetFile—◀◀
```

必須パラメーター

SourceFile

変換されるC構造体を含んでいる入力ファイルを指定します。

TargetFile

構造体を変換するために生成されたコード断片を含んでいる出力ファイルを指定します。

戻りコード

- 0 コマンドは正常に終了しました。
- 10 コマンドは終了しましたが、予期しない結果がでました。
- 20 処理中にエラーが起きました。

例

次の例は、ソースC構造体に対してデータ変換コマンドを使用した結果を示しています。次のコマンドが出されました。

```
crtmqcvx source target
```

入力ファイル `source` は、次のようなものです。

```
/* This is a test C structure which can be converted by the */
/* crtmqcvx utility                                         */

struct my_structure
{
    int    code;
    MQLONG value;
};
```

コマンドによって生成される出力ファイル `target` を次に示します。アプリケーションでこれらのコード断片を使用すれば、データ構造体を変換できます。ただし、これを行う場合には、コード断片がサブボリューム `ZMQSLIB` の `MQSeries` ヘッダー・ファイル `MQSVMHTH` に与えられているマクロを使用することを理解しておく必要があります。

```
MQLONG Convertmy_structure(  
    PMQBYTE *in_cursor,  
    PMQBYTE *out_cursor,  
    PMQBYTE in_lastbyte,  
    PMQBYTE out_lastbyte,  
    MQHCONN hConn,  
    MQLONG opts,  
    MQLONG MsgEncoding,  
    MQLONG ReqEncoding,  
    MQLONG MsgCCSID,  
    MQLONG ReqCCSID,  
    MQLONG CompCode,  
    MQLONG Reason)  
{  
    MQLONG ReturnCode = MQRC_NONE;  
  
    ConvertLong(1); /* code */  
  
    AlignLong();  
    ConvertLong(1); /* value */  
  
Fail:  
    return(ReturnCode);  
}
```

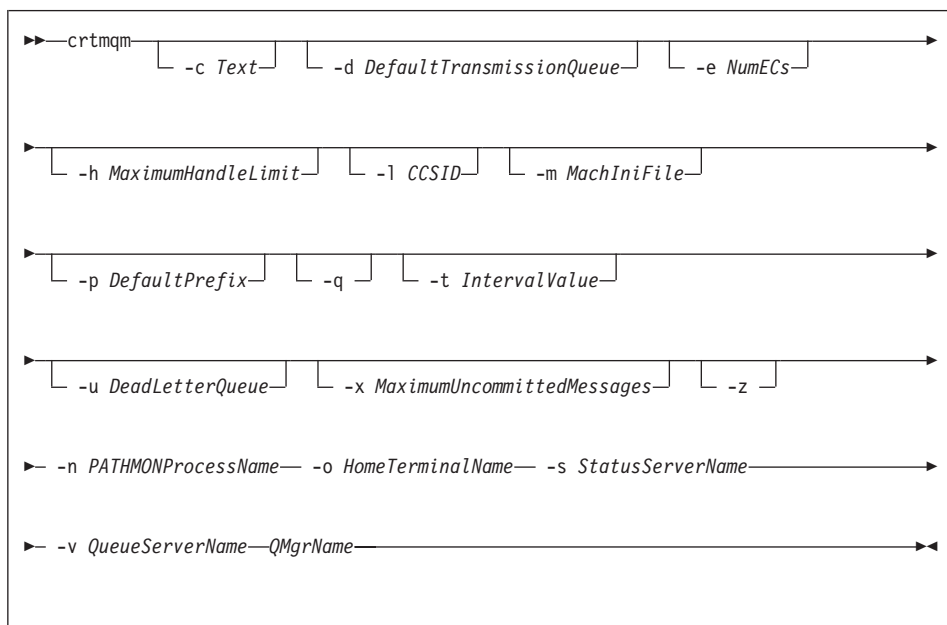
crtmqm (キュー・マネージャーの作成)

目的

crtmqm コマンドは、ローカル・キュー・マネージャーを作成します。キュー・マネージャーが作成された場合は、**strmqm** コマンドを使ってそれを開始します。

キュー・マネージャーを作成すると、関連システムとデフォルトのオブジェクトも自動的に作成されます。

構文



必須パラメーター

-n PATHMONProcessName

キュー・マネージャーの TS/MP PATHMON プロセスのプロセス名。このプロセス名は、システム内で固有でなければなりません。

-o HomeTerminalName

ホーム端末のデバイス名 (\$DDDD.#SS)。たとえば、\$TRM1.#A のようになります。

-s *StatusServerName*

キュー・マネージャーのデフォルトの状況サーバーに指定されるプロセス名。このプロセス名は、システム内で固有でなければなりません。

-v *QueueServerName*

このキュー・マネージャーのデフォルトのキュー・サーバー・プロセスに指定される固有のプロセス名を指定します。

QMgrName

作成するキュー・マネージャーの名前。この名前は、最高 48 文字までです。これは、このコマンドの最後の項目にする必要があります。

オプション・パラメーター**-c** *Text*

そのキュー・マネージャーを説明するテキスト (最大 64 文字)。デフォルトは、すべて空白です。

特殊文字が必要な場合には、記述を二重引用符で囲まなければなりません。

-d *DefaultTransmissionQueue*

伝送キューが宛先について明示的に定義されなかったときにリモート・メッセージが置かれる、ローカル伝送キューの名前。デフォルトはありません。

-e *NumECs*

キュー・マネージャー内の EC プロセスの数。デフォルトは 1 です。

-h *MaximumHandleLimit*

MQSeries for Compaq NSK では、このパラメーターは無視されます。

どのアプリケーションでも同時にオープンできるハンドルの最大数。1 から 999 999 999 までの範囲の値を指定します。デフォルト値は 256 です。

-l *CCSID*

キュー・マネージャーの CCSID。デフォルト値は 819 です。

-m *MachIniFile*

デフォルトの MQSINI ファイルの場所と、環境変数で MQMACHINIFILE で指定されたその場所を指定変更します。

-p *DefaultPrefix*

キュー・マネージャーのボリューム。MQSINI ファイルの QMDefaultVolume 項目を指定変更します。

- q このキュー・マネージャーが、デフォルト・キュー・マネージャーになることを指定します。新しいキュー・マネージャーは、デフォルトとして既存のキュー・マネージャーと置き換わります。

誤ってこのフラグを使用した場合で、既存のキュー・マネージャーをデフォルト・キュー・マネージャーに戻したい場合には、MQSeries 構成ファイルの中の DefaultQueueManager スタンザを編集することができます。

-t IntervalValue

このキュー・マネージャーが制御するすべてのキューについてのトリガー時間間隔 (ミリ秒単位)。この値は、トリガー生成メッセージを受け取った後、トリガー操作が中断される時間を指定します。つまり、あるメッセージがキューに到着してトリガー・メッセージが開始キューに入れられると、指定された時間間隔内に同じキューにメッセージが到着しても、別のトリガー・メッセージは生成されません。

このトリガー時間間隔を使用すれば、アプリケーションは、同じキューの別のメッセージを取り扱うよう警告されるまでに、トリガー条件を取り扱うための十分な時間の余裕が与えられます。生じるすべてのトリガー・イベントを見たい場合があります。その場合には、このフィールドに小さな値かまたはゼロを設定してください。

0 から 999 999 999 までの範囲の値を指定します。デフォルトは、999 999 999 ミリ秒 (11 日以上) です。効果的にデフォルトが使用されるようにすると、トリガー操作は最初のトリガー・メッセージの後、使用不可になることとなります。ただし、そのキューをサービスするアプリケーションは、キュー更新コマンドを使用してトリガー属性をリセットし、トリガー操作を使用可能にすることができます。

-u DeadLetterQueue

送達不能 (未配布メッセージ) キューとして使用されるローカル・キューの名前。メッセージが正しい宛先に送られない場合は、メッセージはこのキューに書き込まれます。

デフォルトでは、送達不能キューはありません。

-x MaximumUncommittedMessages

MQSeries for Compaq NSK では、このパラメーターは無視されます。

同期点においてコミットされないメッセージの最大数を指定します。つまり、これは次のものの合計です。

- キューから取り出すことができるメッセージの数
- キューに書き込むことができるメッセージの数
- この作業単位内で生成されたトリガー・メッセージ

この限界は、同期点制御外で取り出したり書き込まれたりするメッセージには当てはまりません。

1 から 10 000 までの範囲の値を指定します。デフォルト値は、1000 個のコミットされていないメッセージになります。

-z エラー・メッセージを抑制します。

このフラグは、通常、MQSeries の中で不要なエラー・メッセージを抑制するために使用します。このフラグを使用することによって情報が失われる可能性があるため、コマンド行からコマンドを入力する場合には、これは使用しないことをお勧めします。

戻りコード

- 0** キュー・マネージャーが作成されました。
- 8** キュー・マネージャーはすでに存在しています。
- 49** キュー・マネージャーは停止中です。
- 69** ストレージが利用不能です。
- 70** キュー・スペースが利用不能です。
- 71** 予期しないエラーです。
- 72** キュー・マネージャー名のエラーです。
- 111** キュー・マネージャーが作成されました。ただし、プロダクト構成ファイル内のデフォルト・キュー・マネージャー定義の処理中に問題がありました。デフォルト・キュー・マネージャーの指定が誤りである可能性があります。

例

1. 次のコマンドでは、Paint.queue.manager という名前のデフォルト・キュー・マネージャーが作成されます。これには、Paint Shop という説明が与えられています。

```
crtmqm -c "Paint Shop" -n $PANT -o $TRM1.#A -s $PNT1 -v $PQS1 Paint.queue.manager
```

2. この例では、別のキュー・マネージャー travel が作成されます。トリガー間隔は 5000 ミリ秒 (5 秒) に定義され、送達不能レター・キューは、SYSTEM.DEAD.LETTER.QUEUE になります。

```
crtmqm -t 5000 -u SYSTEM.DEAD.LETTER.QUEUE -n $TRAV -o $TRM1.#A -s $TRV1 -v $TQS1 travel
```

crtmqm

トリガー・イベントが 1 度生成されると、追加のトリガー・イベントは 5 秒間使用不可になります。

関連コマンド

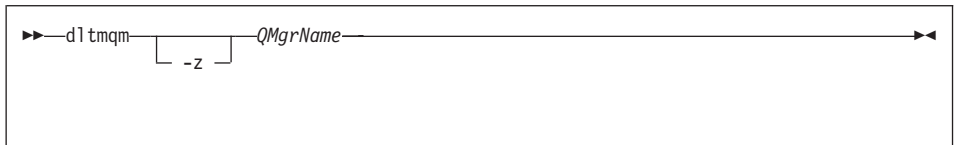
strmqm	キュー・マネージャーを開始します。
endmqm	キュー・マネージャーを終了します。
dltmqm	キュー・マネージャーを削除します。

dltmqm (キュー・マネージャーの削除)

目的

dltmqm コマンドは、指定したキュー・マネージャーを削除するために使用します。削除されるキュー・マネージャーに関連したすべてのオブジェクトも削除されます。キュー・マネージャーを削除するためには、その前に **endmqm** コマンドでそれを終了しておく必要があります。

構文



必須パラメーター

QMGrName

削除するキュー・マネージャーの名前を指定します。

オプション・パラメーター

-z エラー・メッセージを抑制します。

戻りコード

- 0** キュー・マネージャーは削除されました。
- 5** キュー・マネージャーは実行中です。
- 16** キュー・マネージャーは存在していません。
- 69** ストレージが利用不能です。
- 71** 予期しないエラーです。
- 72** キュー・マネージャー名のエラーです。
- 112** キュー・マネージャーは削除されました。ただし、プロダクト構成ファイル内のデフォルト・キュー・マネージャー定義の処理中に問題がありました。デフォルト・キュー・マネージャーの指定が誤りである可能性があります。

例

1. 次のコマンドは、キュー・マネージャー `saturn.queue.manager` を削除します。

dltmqm

```
dltmqm saturn.queue.manager
```

2. 次のコマンドは、キュー・マネージャー `travel` を削除し、コマンドによって引き起こされるメッセージを抑制します。

```
dltmqm -z travel
```

関連コマンド

crtmqm	キュー・マネージャーを作成します。
strmqm	キュー・マネージャーを開始します。
endmqm	キュー・マネージャーを終了します。

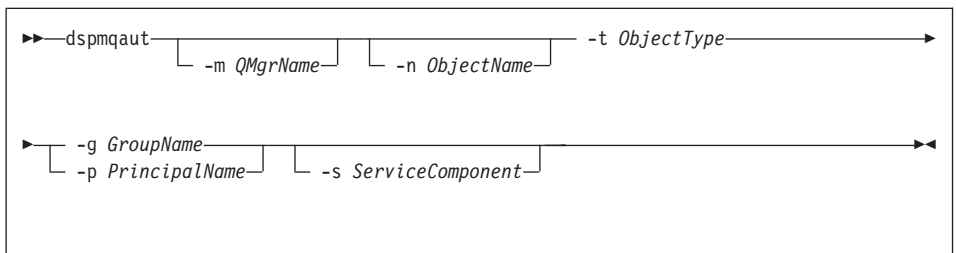
dspmqaout (権限の表示)

目的

dspmqaout コマンドは、指定したオブジェクトに対する現在の許可を表示するために使用します。1つのグループのみを指定できます。

ユーザー ID が複数のグループに属するメンバーである場合には、各グループの許可を調べ、ユーザー ID に適用されるすべての許可を判別します。

構文



必須パラメーター

-g *GroupName*

照会が行われる対象となるユーザー・グループの名前を指定します。指定できるのは **1つ** の名前のみであり、既存のユーザー・グループの名前である必要があります。-g *GroupName* または -p *PrincipalName* のいずれか一方を指定する必要があります。

-p *PrincipalName*

表示しようとしている指定のオブジェクトに対する許可を持っているプリンシパルの名前を指定します。-g *GroupName* または -p *PrincipalName* のいずれか一方を指定する必要があります。

-t *ObjectType*

照会が行われる対象となるオブジェクトのタイプを指定します。可能な値は次のとおりです。

queueまたは**q** オブジェクト・タイプ・パラメーターに一致するキュー (1つまたは複数)

qmgr キュー・マネージャー・オブジェクト

processまたは**prcs**

プロセス

namelist または **nl**
名前リスト

オプション・パラメーター

-m *QMgrName*

照会が行われる対象となるキュー・マネージャーの名前を指定します。

-n *ObjectName*

照会が行われる対象となるオブジェクトの名前を指定します。

これは、それ自体がキュー・マネージャーでない限り、必須パラメーターです。

キュー・マネージャー、キュー、またはプロセス定義の名前を指定しなければなりません。

-s *ServiceComponent*

このパラメーターは、インストール可能な許可サービスを使用している場合にのみ適用されます。それ以外は、無視されます。

インストール可能な許可サービスがサポートされている場合、このパラメーターは、許可が適用される許可サービスの名前を指定します。このパラメーターはオプションです。これを指定しなければ、サービスの最初のインストール可能なコンポーネントに対して許可更新が行われます。

戻されるパラメーター

このコマンドは、許可リストを戻します。その中には、許可パラメーターが何も含まれていないこともあれば、1 つまたはそれ以上の許可パラメーターが含まれていることもあります。戻される各許可パラメーターは、指定のグループの中のユーザー ID が、そのパラメーターで定義された操作を実行する許可を持っていることを意味します。

表9は、種々のオブジェクト・タイプに与えることができる許可を示したものです。

表9. *dspmqaout* コマンドからのセキュリティ権限

権限	キュー	プロセス	キュー・マネージャー	名前リスト
all	✓	✓	✓	✓
alladm	✓	✓	✓	✓
allmqi	✓	✓	✓	✓
altusr			✓	

表9. dspmqaout コマンドからのセキュリティー権限 (続き)

権限	キュー	プロセス	キュー・マネージャー	名前リスト
browse	✓			
chg	✓	✓	✓	✓
chgaut	✓	✓	✓	
clr	✓			
connect			✓	
crt	✓	✓	✓	✓
dlt	✓	✓	✓	✓
dsp	✓	✓	✓	✓
get	✓			
inq	✓	✓	✓	✓
passall	✓			
passid	✓			
put	✓			
set	✓	✓	✓	
setall	✓		✓	
setid	✓		✓	

次のリストは、各パラメーターに関連した許可の定義を示したものです。

- all** オブジェクトに関係のあるすべての操作を使用する。
- alladm** オブジェクトに関係のあるすべての管理操作を実行する。
- allmqi** オブジェクトに関係のあるすべての MQI 呼び出しを使用する。
- altusr** MQI 呼び出しで代替ユーザー ID を指定する。
- browse** BROWSE オプションを指定した MQGET 呼び出しを出して、キューからメッセージを取り出す。
- chg** 指定したオブジェクトの属性を、該当のコマンド・セットを使用して変更する。
- chgaut** オブジェクト上の別のユーザー・グループの許可を、**setmqaut** コマンドを使用して指定する。
- clr** キューをクリアする (PCF コマンド「キューのクリア」のみ)。
- connect** MQCONN 呼び出しを出して、指定のキュー・マネージャーにアプリケーションを接続する。

dspmqaut

crt	指定のタイプのオブジェクトを、該当のコマンド・セットを使用して作成する。
dlt	指定のオブジェクトを、該当のコマンド・セットを使用して削除する。
dsp	指定のオブジェクトの属性を、該当のコマンド・セットを使用して表示する。
get	MQGET 呼び出しを出して、キューからメッセージを取り出す。
inq	MQINQ 呼び出しを出して、特定のキューの照会を行う。
passall	すべてのコンテキストを渡す。
passid	識別コンテキストを渡す。
put	MQPUT 呼び出しを出して、特定のキューにメッセージを書き込む。
set	MQSET 呼び出しを出して、MQI からキューに属性を設定する。
setall	キューにすべてのコンテキストを設定する。
setid	キューの識別コンテキストを設定する。

管理操作の許可は、サポートされている場合には、次のコマンド・セットに適用されます。

- 制御コマンド
- MQSC コマンド
- PCF コマンド

戻りコード

0	正常な操作です。
36	与えられた引き数が無効です。
40	キュー・マネージャーは利用不能です。
49	キュー・マネージャーは停止中です。
69	ストレージが利用不能です。
71	予期しないエラーです。
72	キュー・マネージャー名のエラーです。
133	オブジェクト名が不明です。
145	予期しないオブジェクト名です。
146	オブジェクト名が脱落しています。
147	オブジェクト・タイプが脱落しています。
148	オブジェクト・タイプが無効です。
149	エンティティ名が脱落しています。

例

次の例は、ユーザー・グループ `staff` に関連したキュー・マネージャー `saturn.queue.manager` に関する許可を表示するためのコマンドを示しています。

```
dspmqaout -m saturn.queue.manager -t qmgr -g staff
```

このコマンドの結果を次に示します。

```
Entity staff has the following authorizations for object :
  get
  browse
  put
  inq
  set
  connect
  altusr
  passid
  passall
  setid
```

関連コマンド

setmqaut 権限の設定またはリセット

dspmqcsv (コマンド・サーバーの表示)

目的

dspmqcsv コマンドは、指定したキュー・マネージャーのコマンド・サーバーの状況を表示するために使用します。

状況は次のいずれかです。

- 開始中
- 実行中
- SYSTEM.ADMIN.COMMAND.QUEUE で実行中 (読み取り不可)
- 終了中
- 停止

構文

```
▶▶—dspmqcsv—QMgrName————▶▶
```

必須パラメーター

QMgrName

コマンド・サーバーの状況を要求する対象となるローカル・キュー・マネージャーの名前を指定します。

戻りコード

- 0** コマンドは正常に終了しました。
- 10** コマンドは終了しましたが、予期しない結果がでました。
- 20** 処理中にエラーが起きました。

例

次のコマンドは、`venus.q.mgr:` に関連するコマンド・サーバーの状況を表示します。

```
dspmqcsv venus.q.mgr
```

関連コマンド

strmqcsv	コマンド・サーバーを開始します。
endmqcsv	コマンド・サーバーを終了します。

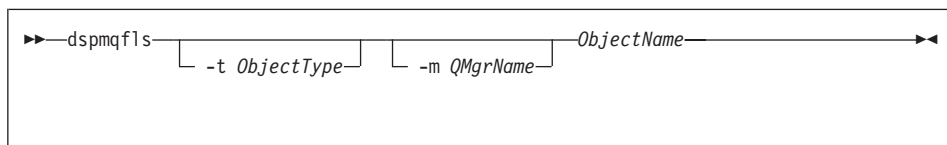
dspmqfls (MQSeries ファイル属性の表示)

目的

dspmqfls コマンドは、指定した基準に一致するすべての MQSeries オブジェクトの実際のファイル・システム名を表示するために使用します。このコマンドを使用すれば、特定の MQSeries オブジェクトに関連したファイルを識別することができます。これは、特定のオブジェクトのバックアップをとるのに役立ちます。名前変換の詳細については、79ページの『ボリューム構造』を参照してください。

さらに、**dspmqfls** コマンドを使用して、キュー・サーバーのプロセス名およびオブジェクトのオプションに関する現行の設定を表示することもできます。

構文



必須パラメーター

ObjectName

MQSeries オブジェクトの名前を指定します。名前は、フラグなしのストリングで、必須パラメーターです。名前を省略すると、エラーが戻されます。

このパラメーターは、ストリングの最後にワイルドカード文字 `*` を指定できます。

オプション・パラメーター

-m *QMgrName*

ファイルを調べる対象となるキュー・マネージャーの名前を指定します。このパラメーターを省略すると、コマンドは、デフォルト・キュー・マネージャーに対して操作を実行します。

-t *ObjType*

MQSeries オブジェクト・タイプを指定します。有効なオブジェクト・タイプを以下に示します。省略名を最初に示し、その後完全な名前を示しています。

- * または **all** 全てのオブジェクト・タイプ。これはデフォルトです。
- q** または **queue** オブジェクト名パラメーターに一致するキュー (1 つまたは複数)
- ql** または **qlocal** ローカル・キュー
- qa** または **qalias** 別名キュー
- qr** または **qremote** リモート・キュー
- qm** または **qmodel** モデル・キュー
- qmgr** キュー・マネージャー・オブジェクト
- prcs** または **process** プロセス
- ctlg** または **catalog** オブジェクト・カタログ
- nl** または **namelist** 名前リスト

注: **dspmqls** コマンドは、キューのすべてのファイルの名前を表示します。

戻りコード

- 0 コマンドは正常に終了しました。
- 10 コマンドは完了しましたが、すべて予期したとおりではありません。
- 20 処理中にエラーが起きました。

例

1. 次のコマンドは、デフォルト・キュー・マネージャーに定義されている、SYSTEM.ADMIN で始まる名前を持つすべてのオブジェクトの詳細を表示します。

```
dspmqls SYSTEM.ADMIN*
```

2. 次のコマンドは、キュー・マネージャー RADIUS に定義された、PROC で始まる名前を持つすべてのプロセスのファイルの詳細を表示します。

dspmqfls

```
dspmqfls -m RADIUS -t prcs PROC*
```

3. 以下のコマンドは、MY.LOCAL.QUEUE のファイル情報を表示します。

```
dspmqfls -m MT02 -t q MY.LOCAL.QUEUE

MQSeries Display MQ Files

CONNECTING.
QLOCAL MY.LOCAL.QUEUE
  $DATA0.MT02M.QMYXLOCA
  $DATA0.MT02M.OMYXLOCA
  $DATA0.MT02M.TMYXLOCA

Queue/Status Server:           DEFAULT
Persistent message browse buffer: 0
Message overflow threshold:    200000
Queue Server Options:          None
Message overflow subvolume:    QMGR0010
Measure Counter:
```

- QMYXLOCA はキュー・ファイル、OMYXLOCA はキューのオーバーフロー・ファイル、そして TMYXLOCA はキュー・マネージャーのデータ・ディレクトリーにあるタッチ・ファイルです。
4. 以下の例は、始動時のロードおよびチェックポイント NPM キュー・サーバーのオプションを設定する、**altmqfls** コマンドを示しています。**dspmqfls** コマンドは、**altmqfls** コマンドの結果を表示します。

```
altmqfls --qmgr MT02 --type ql --qsoptions SC MY.LOCAL.QUEUE  
  
dspmqls -m MT02 -t q MY.LOCAL.QUEUE  
  
MQSeries Display MQ Files  
  
CONNECTING.  
QLOCAL MY.LOCAL.QUEUE  
  $DATA4.MT02M.QMYXLOCA  
  $DATA4.MT02M.OMYXLOCA  
  $DATA4.MT02M.TMYXLOCA  
  
Queue/Status Server:          DEFAULT  
Persistent message browse buffer: 0  
Message overflow threshold:   200000  
Queue Server Options:        SC  
Message overflow subvolume:   QMGR0010  
Measure Counter:
```

関連コマンド

altmqfls キュー・ボリュームを更新します。

dspmqtrc (MQSeries 定様式トレース出力の表示)

目的

dspmqtrc コマンドは、MQSeries の定様式トレース出力を表示する場合に使用します。MQSeries トレースの使用の詳細については、280ページの『MQSeries トレースの使用』を参照してください。

構文

```
▶—dspmqtrc— -t FormatTemplate —InputFileName—▶▶
```

必須パラメーター

InputFileName

不定様式トレースが格納されているファイルの名前を指定します。たとえば、\$DATA.MQTRACE.AMQ12345. とします。

-t *FormatTemplate*

トレースの表示方法の詳細を含んでいるテンプレート・ファイルの名前を指定します。トレース様式のテンプレート・ファイル **AMQTRC** は、サブボリューム **ZMQSSMPL** で提供されます。

関連コマンド

endmqtrc	MQSeries トレースの終了
strmqtrc	MQSeries トレースの開始

dspmqusr (MQSeries ユーザー情報の表示)

目的

dspmqusr コマンドは、指定したプリンシパル、またはキュー・マネージャーのすべてのプリンシパルについての情報を表示するために使用します。

構文

```
▶▶—dspmqusr— -m QMgrName —————▶▶  
└─ -p PrincipalName ─┘
```

説明

このコマンドを使用すると、以下のことを実行できます。

- キュー・マネージャーに対して定義されたすべてのプリンシパル、または特定のプリンシパルを表示します。
- 各プリンシパルと関連した Compaq NSK ユーザー ID と対応する、Compaq NSK 管理グループおよび SAFEGUARD ファイル共有グループを表示します。

必須パラメーター

-m *QMgrName*

プリンシパルの所属先のキュー・マネージャーの名前です。

オプション・パラメーター

-p *PrincipalName*

表示するプリンシパルの名前です。

戻りコード

- 0** 正常な操作です。
- 36** 与えられた引き数が無効です。
- 69** ストレージが利用不能です。
- 71** 予期しないエラーです。

dspmqsqr

例

1. 以下の例は、新しく作成されたキュー・マネージャーに対する **dspmqsqr** コマンドを示しています。

```
dspmqsqr -m MT02
```

Principal	Userid	Username	Alias	GroupName	GroupType
	0.1				
NOBODY	0.0				
mqm	20.255	MQM.MANAGER	n	MQM	a

プリンシパル・データベースには、プリンシパル **mqm** が含まれますが、これはキュー・マネージャーを作成したユーザーの名前にマップします。

2. 以下の例は、**altmqsqsr** を使用してプリンシパルが追加された後の、**dspmqsqr** の出力を示しています。

```
dspmqsqr -m MT02
```

Principal	Userid	Username	Alias	GroupName	GroupType
	0.1				
NOBODY	0.0				
mqm	20.255	MQM.MANAGER	n	MQM	a
mquser1	50.3	MQTEST.FRED	n	MQTEST	a
				MQM	s
mquser2	1.1	GROUP.USER01	n	GROUP	a

Compaq ユーザー ID **MQTEST.FRED** にマップするプリンシパル **mquser1** が追加されました。 **FRED** は、グループ **MQTEST**、および **SAFEGUARD** 別名を使用したグループ **MQM** のメンバーです。

プリンシパル **mquser2** は、Compaq ユーザー ID **GROUP.USER01** にマップします。

関連コマンド

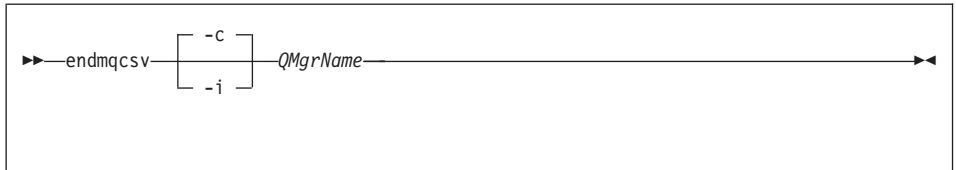
altmqsqsr MQSeries ユーザー情報を更新します。

endmqcsv (コマンド・サーバーの終了)

目的

endmqcsv コマンドは、指定したキュー・マネージャーのコマンド・サーバーを停止するために使用します。

構文



必須パラメーター

QMgrName

コマンド・サーバーを終了させる対象となるキュー・マネージャーの名前を指定します。

オプション・パラメーター

-c コマンド・サーバーが制御された状態で停止することを指定します。コマンド・サーバーは、すでに開始されているコマンド・メッセージの処理があれば、その処理を完了することができます。新しいメッセージが、コマンド・キューから読み取られることはありません。

これはデフォルトです。

-i コマンド・サーバーが即時に停止することを指定します。現在処理されているコマンド・メッセージに関連したアクションは、完了しない可能性があります。

戻りコード

- 0 コマンドは正常に終了しました。
- 10 コマンドは終了しましたが、予期しない結果がでました。
- 20 処理中にエラーが起きました。

endmqcsv

例

1. 次のコマンドは、キュー・マネージャー saturn.queue.manager のコマンド・サーバーを停止します。

```
endmqcsv -c saturn.queue.manager
```

コマンド・サーバーは、停止する前に、すでに開始しているコマンドの処理を完了することができます。新しく受け取ったコマンドは、コマンド・サーバーが再始動されるまで、処理されずにコマンド・キューに残ります。

2. 次のコマンドは、キュー・マネージャー pluto のコマンド・サーバーを即時に停止します。

```
endmqcsv -i pluto
```

関連コマンド

- | | |
|-----------------|---------------------|
| strmqcsv | コマンド・サーバーを開始します。 |
| dspmqcsv | コマンド・サーバーの状況を表示します。 |

endmqm (キュー・マネージャーの終了)

目的

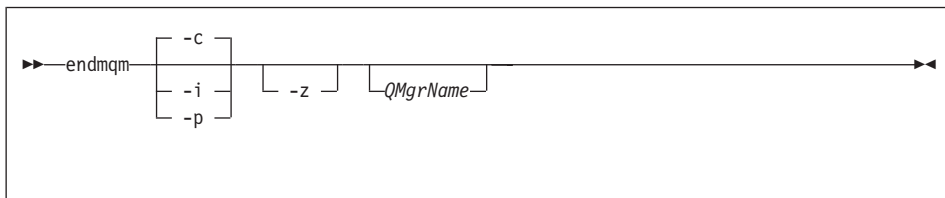
endmqm コマンドは、指定したローカル・キュー・マネージャーを終了 (停止) するために使用します。このコマンドは、次の 3 つのモードのいずれかでキュー・マネージャーを停止します。

- 正常または静止シャットダウン
- 即時シャットダウン
- 優先シャットダウン

キュー・マネージャーの属性および関連したオブジェクトの属性は、影響を受けません。**strmqm** (キュー・マネージャーの開始) コマンドを使用すれば、キュー・マネージャーを再始動できます。

キュー・マネージャーを削除するためには、それを停止し、その後 **dltmqm** (キュー・マネージャーの削除) コマンドを使用する必要があります。

構文



オプション・パラメーター

QMgrName

停止させるメッセージ・キュー・マネージャーの名前です。名前を指定しない場合、デフォルト・キュー・マネージャーが停止します。

- c 制御された (または静止) シャットダウン。キュー・マネージャーは、すべてのアプリケーションが切断された後でのみ停止します。現在処理されている MQI 呼び出しはすべて完了します。これはデフォルトです。
- i 即時シャットダウン。キュー・マネージャーは、現在処理されている MQI 呼び出しをすべて完了してから停止します。このコマンドが出された後に出された MQI 要求はすべて失敗します。完了しなかった作業単位は、キュー・マネージャーが次に開始されるときに、ロールバックされます。
- p 優先シャットダウン

このタイプのシャットダウンは、例外的な状況でのみ使用してください。たとえば、キュー・マネージャーが通常の **endmqm** コマンドで停止しない場合などです。

キュー・マネージャーは、アプリケーションが切断されるのを待たずに、あるいは MQI 呼び出しが完了するのを待たずに停止します。これによって、MQI アプリケーションにとって予測不能な結果が生じる可能性があります。キュー・マネージャー内の停止できなかったすべてのプロセスは、このコマンドが出された後 30 秒してから終了します。

- z コマンドでのエラー・メッセージを抑制します。

戻りコード

- 0 キュー・マネージャーは終了しました。
- 16 キュー・マネージャーは存在していません。
- 36 無効な引き数
- 40 キュー・マネージャーは利用不能です。
- 69 ストレージが利用不能です。
- 71 予期しないエラーです。
- 72 キュー・マネージャー名のエラーです。

例

次の例は、指定されたキュー・マネージャーを終了 (停止) させるコマンドを示しています。

1. 次のコマンドは、デフォルトの名前のキュー・マネージャーを制御された状態で終了させます。現在接続されているすべてのアプリケーションは、切断することが可能です。

```
endmqm
```

2. 次のコマンドは、`saturn.queue.manager` という名前のキュー・マネージャーを即時に終了させます。現行の MQI 呼び出しはすべて完了しますが、新しい呼び出しは完了できません。

```
endmqm -i saturn.queue.manager
```

関連コマンド

crtmqm	キュー・マネージャーを作成します。
strmqm	キュー・マネージャーを開始します。
dltmqm	キュー・マネージャーを削除します。

endmqtrc (MQSeries トレースの終了)

目的

endmqtrc コマンドは、指定したキュー・マネージャーのトレースを終了する場合に使用します。

MQSeries トレースの使用の詳細については、280ページの『MQSeries トレースの使用』を参照してください。

構文



必須パラメーター

-m *QMgrName*

トレースが終了されるキュー・マネージャーの名前です。

キュー・マネージャー名は、**-e** フラグと同じコマンド上に指定できます。

-a このフラグを指定すると、すべてのトレースが終了します。

このフラグは、**必ず** 単独で指定してください。

オプション・パラメーター

-e このフラグを指定すると、指定したキュー・マネージャー上の早期トレースが終了します。

戻りコード

AMQ5611 このメッセージは、コマンドに無効な引き数を指定した場合に出されます。

例

このコマンドは、QM1 というキュー・マネージャーに関するデータのトレースを終了させます。


```
endmqtrc -m QM1
```

関連コマンド

dspmqtrc	定様式トレース出力の表示
strmqtrc	MQSeries トレースの開始

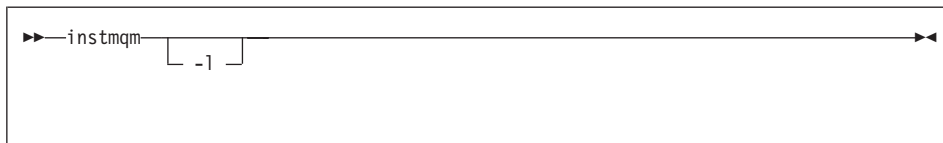
instmqm

instmqm (MQSeries for Compaq NSK のインストール)

目的

instmqm コマンドは、MQSeries for Compaq NSK をインストールするか、またはライセンス情報を更新するために使用します。

構文



オプション・パラメーター

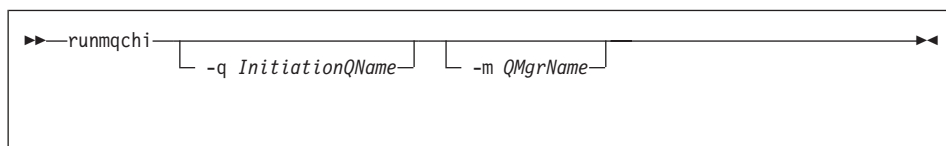
-l instmqm を呼び出してライセンス情報を更新します。

runmqchi (チャンネル・イニシエーターの実行)

目的

runmqchi コマンドは、チャンネル・イニシエーターのプロセスを実行するために使用します。このコマンドの使用の詳細については、*MQSeries 相互通信* を参照してください。

構文



オプション・パラメーター

-q *InitiationQName*

このチャンネル・イニシエーターが処理する開始キューの名前を指定します。値を指定しなければ、SYSTEM.CHANNEL.INTQ が使用されます。

-m *QMgrName*

開始キューが存在しているキュー・マネージャーの名前を指定します。この名前を省略した場合、デフォルト・キュー・マネージャーが使用されず。

戻りコード

- 0** コマンドは正常に終了しました。
- 10** コマンドは終了しましたが、予期しない結果がでました。
- 20** 処理中にエラーが起きました。

戻りコードの 10 か 20 が戻されるようなエラーが起きた場合、チャンネルが関連付けられているキュー・マネージャーのエラー・ログを見て、エラー・メッセージを調べてください。また、システム・エラー・ログも見てください。チャンネルがキュー・マネージャーと関連付けられる前に起きた問題はそこに記録されています。エラー・ログの詳細については、275ページの『エラー・ログ』を参照してください。

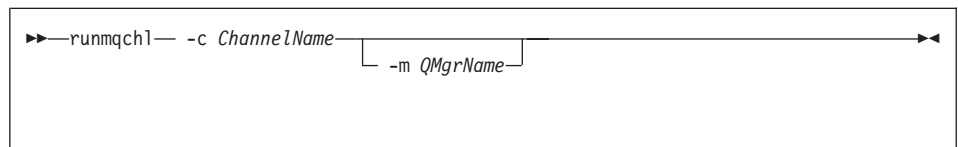
runmqchl (チャネルの実行)

目的

runmqchl コマンドは、送信側 (SDR)、要求側 (RQSTR)、または完全修飾サーバー・チャネルのいずれかを開始する場合に使用します。

チャネルは非同期で実行されます。チャネルを停止するには、MQSC コマンド **STOP CHANNEL** を出します。

構文



必須パラメーター

-c ChannelName

開始するチャネルの名前を指定します。

オプション・パラメーター

-m QMgrName

このチャネルが関連付けられているキュー・マネージャーの名前を指定します。名前を指定しない場合、デフォルト・キュー・マネージャーが使用されます。

戻りコード

- 0** コマンドは正常に終了しました。
- 10** コマンドは終了しましたが、予期しない結果がでました。
- 20** 処理中にエラーが起きました。

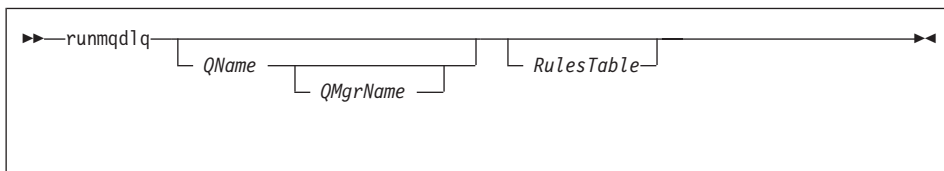
戻りコードの 10 または 20 が生成された場合、関連したキュー・マネージャーのエラー・ログを見て、エラー・メッセージを調べてください。また、MQSeries システム・エラー・ログも見てください (ZMQSSYS にあります)。チャネルがキュー・マネージャーと関連付けられる前に起きた問題はそこに記録されています。

runmqdlq (送達不能キュー・ハンドラーの実行)

目的

runmqdlq コマンドは、送達不能キュー (DLQ) ハンドラーを開始するために使用します。このハンドラーは、送達不能キューのメッセージを処理するユーティリティです。

構文



オプション・パラメーター

QName

処理する送達不能キューの名前です。

QName の値を指定する場合、ルール・テーブルで指定したすべての **INPUTQ** 値を指定変更します。 **runmqdlq** の入力およびルール・テーブルのどちらにも (非ブランクの) 名前が指定されない場合、*QMgrName* パラメーターで指定されたキュー・マネージャーに関連した送達不能キューが処理されます。

QMgrName

処理するキューを所有するキュー・マネージャーの名前です。

QMgrName の値を指定する場合、ルール・テーブルで指定したすべての **INPUTQM** 値を指定変更します。 **runmqdlq** の入力およびルール・テーブルのどちらにも (非ブランクの) 名前が指定されない場合、キューはデフォルトのキュー・マネージャーに属するものと想定されます。

RulesTable

ルール・テーブルを含むファイルの名前です。このルール・テーブルには、少なくとも 1 つのルールが含まれていなければなりません。

デフォルトでは、**runmqdlq** コマンドは、標準 **IN** ファイルからその入力を取ります。コマンドが処理されると、結果と要約がレポートに書き込まれ、標準 **OUT** ファイルに送られます。あるいは、入力をファイルからリダイレクトすることによって、指定したキューにルール・テーブルを適用することもできます。

runmqdlq

runmqdlq の入力にルール・テーブルを指定しない場合、ルールとアクションを対話式に指定しなければなりません。

この場合、DLQ ハンドラーは次のように機能します。

- キーボードから入力を読み取ります。
- `end_of_file` (Ctrl-Y) 文字を受け取るまで、指定されたキューの処理は開始しません。
注釈行と行結合に関する MQSC のルールは、DLQ ハンドラーの入力パラメーターにも適用されます。

ルール・テーブルおよびルール・テーブルの構成方法の詳細については、206ページの『DLQ ハンドラーのルール・テーブル』を参照してください。

runmqlsr (リスナーの実行)

目的

runmqlsr (リスナーの実行) コマンドは、TCP/IP リスナー・プロセスを実行します。

構文

```
runmqlsr [-t tcp] [-p Port] [-m QMgrName]
```

説明

TACL プロンプトから **runmqlsr** を実行すると、リスナーが終了するまで TACL プロンプトに制御を戻しません。つまり、**runmqlsr** は、待ち状態で実行されます。

TACL プロンプトは、障害が発生するか、またはリスナーが停止する場合のみ戻します。**runmqlsr** の前に端末 (TACL) が停止すると、リスナーはそのホーム端末または out ファイルにアクセスできません。**runmqlsr** を呼び出す前に、すべての PARAM (MQEMSEVENTS など) を定義しなければなりません。

以上の点から、リスナーの開始および停止は、キュー・マネージャーの PATHWAY から行うようにお勧めします。そうすることにより、制御の範囲を大幅に広げることができます。

オプション・パラメーター

-p Port

TCP/IP の場合のポート番号。値を指定しないと、QMIMI ファイル中の TCPCConfig スタンザにある TCPListenerPort 項目で指定したポート番号が使用されます。デフォルト値は 1414 です。QMIMI で複数のリスナー・ポートが定義されている場合、次に使用可能なポートが使用されます。

TACL 環境またはプログラムの PATHWAY サーバー・クラス定義に PARAM MPORTNUMBER を指定されている場合、**runmqlsr** は、QMIMI ファイル中のポートの代わりに、指定されたポートを listen します。

runmqlsr

QMINI で指定されたポートの空きがない場合、または **runmqlsr** コマンドで指定したポートが利用不能の場合、**runmqlsr** は失敗します。

-m *QMgrName*

キュー・マネージャーの名前を指定します。名前を指定しない場合、コマンドはデフォルト・キュー・マネージャーに対して操作を実行します。

-t *tcp*

TCP/IP を伝送プロトコルとして識別します。

TACL 環境またはプログラムの PATHWAY サーバー・クラス定義に `DEFINE =TCPIP^PROCESS^NAME` が存在する場合、**runmqlsr** は、デフォルトの代わりに、Guardian TCP/IP サーバー・プロセスを使用します。

PARAM MQPORTNUMBER が指定されるか、またはプログラムの PATHWAY サーバー・クラス定義が指定される場合、**runmqlsr** は QMINI ファイルでリストされているポートの代わりに、指定されたポートを `listen` します。

これは、MQSeries for Compaq NSK でただ 1 つの有効な値 (およびデフォルト) です。

戻りコード

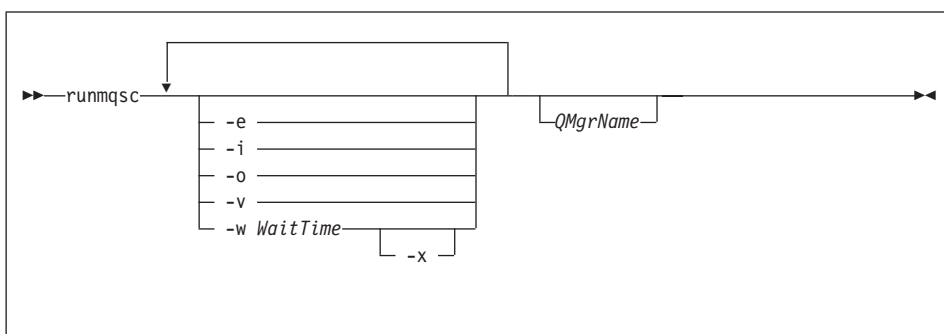
- 0** コマンドは正常に終了しました。
- 10** コマンドは終了しましたが、予期しない結果がでました。
- 20** 処理中にエラーが起きました。

runmqsc (MQSeries コマンドの実行)

目的

runmqsc コマンドは、キュー・マネージャーに対して MQSC コマンドを出すために使用します。MQSC コマンドを使用すると、ローカル・キュー・オブジェクトの定義、更新、または削除などの管理タスクを実行できます。MQSC コマンドとその構文については、*MQSeries MQSC コマンド・リファレンス* で説明しています。

構文



説明

runmqsc コマンドは、次の 3 つのモードで呼び出すことができます。

検証モード

MQSC コマンドは検査されますが、実際には実行されません。各コマンドが正常に実行されるか失敗するかを示す出力レポートが生成されます。このモードは、ローカル・キュー・マネージャーでのみ使用できます。

直接モード

MQSC コマンドは、ローカル・キュー・マネージャーに直接送られます。

間接モード

MQSC コマンドは、リモート・キュー・マネージャーで実行されます。これらのコマンドは、リモート・キュー・マネージャーのコマンド・キューに書き込まれ、キューに入れられた順序で実行されます。コマンドからのレポートは、ローカル・キュー・マネージャーに戻されます。

runmqsc コマンドは、標準 IN ファイルから入力を取ります。コマンドが処理されると、結果と要約がレポートに書き込まれ、そのレポートは標準 OUT ファイルに送られます。

標準 IN ファイルがキーボードから取られる場合は、MQSC コマンドを対話式で入力できます。

ファイルからの入力をリダイレクトすることにより、ファイルに入っている使用頻度の高い一連のコマンドを実行できます。出力レポートをファイルにリダイレクトすることもできます。

注: このコマンドを実行するには、使用するユーザー ID がユーザー・グループ MQM に属していなければなりません。

オプション・パラメーター

- e MQSC コマンドのソース・テキストがレポートにコピーできないようにします。これは、対話式でコマンドを入力する場合に便利です。
- i 入力ファイル名
- o 出力ファイル名
- v 検証モードを指定します。アクションを実行しないで、指定のコマンドを確認します。このモードはローカルのみで使用できます。-w および -x フラグは、同時に指定された場合には無視されます。
- w *WaitTime*
間接モードを指定します。つまり、MQSC コマンドは、別のキュー・マネージャーで実行されることとなります。このためには、必要なチャネルと伝送キューがセットアップされている必要があります。

WaitTime

runmqsc が応答を待つ時間を秒単位で指定します。この時間が経過した後に受け取る応答は破棄されますが、MQSC コマンドはそのまま実行されます。1 から 999 999 秒までの時間を指定します。

各コマンドは、Escape PCF として、宛先キュー・マネージャーのコマンド・キュー (SYSTEM.ADMIN.COMMAND.QUEUE) へ送られます。

応答キューは SYSTEM.MQSC.REPLY.QUEUE に入れられ、結果はレポートに追加されます。これは、ローカル・キューまたはモデル・キューとして定義できます。

間接モード操作は、デフォルト・キュー・マネージャーを介して実行されます。

フラグが指定されている場合には、このフラグは無視されます。

- x 宛先キュー・マネージャーが の下で実行されることを指定します。このフラグは、間接モードでしか適用されません。 -w フラグも指定する必要があります。間接モードでは、MQSC コマンドは、MQSeries (OS/390 版) のコマンド・キューに適した形式で書き込まれます。

QMgrName

MQSC コマンドが実行される宛先キュー・マネージャーの名前を指定します。これを省略すると、MQSC コマンドは、デフォルト・キュー・マネージャーに対して実行されます。

戻りコード

- 00 MQSC コマンド・ファイルは正常に処理されました。
- 10 MQSC コマンド・ファイルは処理されましたが、エラーが起きました。レポートの中にコマンドの失敗の理由が示されています。
- 20 エラー。MQSC コマンドは実行されませんでした。

例

1. 次のコマンドを TACL プロンプトで入力します。

```
runmqsc
```

これで、MQSC コマンドを直接入力することができました。キュー・マネージャー名が指定されなかったので、MQSC コマンドは、デフォルトのキュー・マネージャーで処理されます。

2. 以下の例は、MQSC コマンドの検査のみが行われるように指定する方法を示しています。

```
runmqsc -i $SYSTEM.CONFIG.MQSCIN -v BANK
```

これは、MQSC コマンド・ファイル \$SYSTEM.CONFIG.MQSCIN を検査します。キュー・マネージャー名は BANK です。出力は、現行ウィンドウに表示されます。

runmqsc

3. このコマンドは、BANK というキュー・マネージャーに対して MQSC コマンド・ファイルを実行します。

```
runmqsc -i MQSCFILE -o $TEST.MQ.MQSCOUT BANK
```

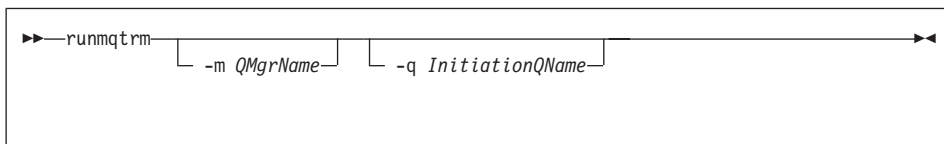
この例では、出力先はファイル \$TEST.MQ.MQSCOUT です。入力ファイルは、現行のサブボリュームにある MQSCFILE です。

runmqtrm (トリガー・モニターの開始)

目的

runmqtrm コマンドは、トリガー・モニターを呼び出すために使用します。トリガー・モニターの詳細については、*MQSeries アプリケーション・プログラミング・ガイド* を参照してください。

構文



オプション・パラメーター

-m *QMgrName*

トリガー・モニターが操作を行う対象となるキュー・マネージャーの名前を指定します。このパラメーターを省略すると、トリガー・モニターは、デフォルト・キュー・マネージャーに対して操作を行います。

-q *InitiationQName*

処理される開始キューの名前を指定します。このパラメーターを省略すると、SYSTEM.DEFAULT.INITIATION.QUEUE が使用されます。

戻りコード

- 10 トリガー・モニターは、エラーによって割り込まれました。
- 20 エラー。トリガー・モニターは実行されませんでした。

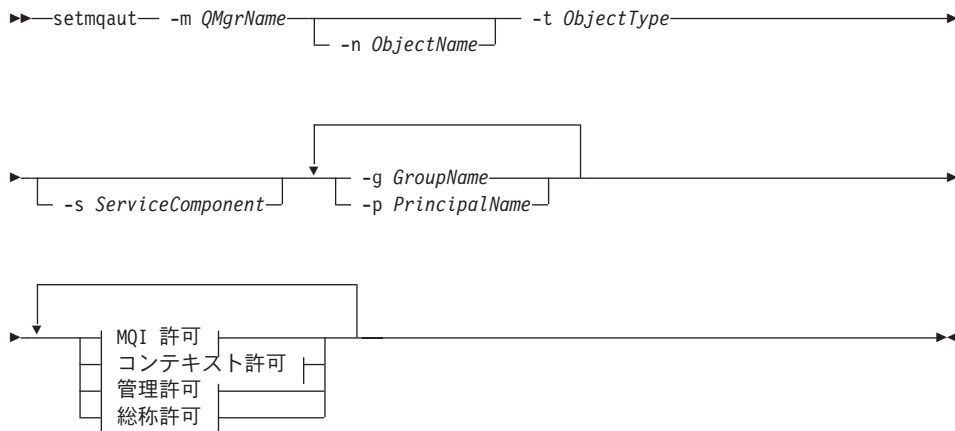
setmqaut

setmqaut (権限の設定 / リセット)

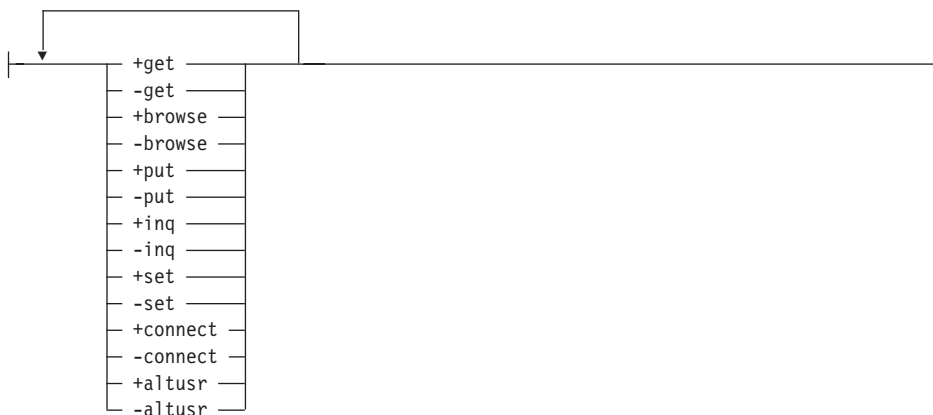
目的

setmqaut コマンドは、オブジェクトまたはオブジェクトのクラスに対する許可を変更するために使用します。どのような数のプリンシパルまたはグループに対してでも、許可を与えたり、許可を取り消したりすることができます。

構文



MQI 許可:



コンテキスト許可:



管理許可:



総称許可:



説明

このコマンドは、許可の設定（つまり、操作の実行許可をユーザー・グループに与えること）および許可のリセット（つまり、操作の実行許可を取り除くこと）の両方を行う場合に使用できます。許可が適用されるユーザー・グループを指定し、同時にキュー・マネージャー、オブジェクト・タイプ、およびオブジェクトのオブジェクト名も指定する必要があります。単一のコマンドで、グループをいくつでも指定できます。

与えることができる許可は、次のように分類されます。

setmqaut

- MQI 呼び出しの発行許可
- MQI コンテキストの許可
- 管理タスク用コマンドの発行許可
- 総称許可

変更するそれぞれの許可は、コマンドの一部として許可リストに指定します。そのリスト内の各項目は、接頭部「+」または「-」が付いたstringです。たとえば、+put を許可リストに含めると、キューに対する MQPUT の発行権限を与えることになります。一方、-put を許可リストに含めると、MQPUT の発行許可を取り除くことになります。

複数の許可を、どのような順序でも指定できますが、それらの許可が対立しないようにする必要があります。たとえば、allmqi と set を同時に指定すると、対立が生じます。

必要な数のグループまたは許可を単一のコマンドに指定できます。

あるユーザー ID が複数のグループに属するメンバーである場合には、そのユーザー ID が属している各グループの許可を合併したものが、適用される許可になります。

必須パラメーター

-g *GroupName*

許可変更の対象となるユーザー・グループの名前を指定します。複数のグループ名を指定できますが、それぞれの名前の前に -g フラグを付ける必要があります。

最低 1 つのプリンシパルまたはグループを指定する必要があります。

-m *QMgrName*

許可変更の対象となるオブジェクトのキュー・マネージャーの名前を指定します。この名前は、最高 48 文字までです。

-p *PrincipalName*

許可変更の対象となるプリンシパルの名前を指定します。複数のプリンシパル名を指定できますが、それぞれの名前の前に -p フラグを付ける必要があります。

最低 1 つのプリンシパルまたはグループを指定する必要があります。

-t *ObjectType*

許可変更の対象となるオブジェクトのタイプを指定します。

可能な値は次のとおりです。

- q または queue
- prcs または process
- qmgr
- nl または namelist

オプション・パラメーター

-n *ObjectName*

許可変更の対象となるオブジェクトの名前を指定します。

これは、それ自身がキュー・マネージャーでない限り、必須パラメーターです。キュー・マネージャー、キュー、またはプロセスの名前を指定する必要がありますが、総称名であってはなりません。

-s *ServiceComponent*

このパラメーターは、インストール可能な許可サービスを使用している場合にのみ適用されます。それ以外は、無視されます。

インストール可能な許可サービスがサポートされている場合、このパラメーターは、許可が適用される許可サービスの名前を指定します。このパラメーターはオプションです。これを指定しなければ、サービスの最初のインストール可能なコンポーネントに対して許可更新が行われます。

Authorizations

付与または除去される許可を指定します。リスト中の各項目には接頭部として、権限が与えられることを示す「+」または許可が除去されることを示す「-」が付きます。たとえば、MQI から MQPUT 呼び出しを発行する権限を与えるには、リストに +put を指定します。MQPUT 呼び出しを発行する許可を除去するには、-put を指定します。

表10は、種々のオブジェクト・タイプに与えることができる許可を示したものです。

表 10. 種々のオブジェクト・タイプについての許可の指定

権限	キュー	プロセス	キュー・マネージャー	名前リスト
all	✓	✓	✓	✓
alladm	✓	✓	✓	✓
allmqi	✓	✓	✓	✓
altusr			✓	
browse	✓			
chg	✓	✓	✓	✓

表 10. 種々のオブジェクト・タイプについての許可の指定 (続き)

権限	キュー	プロセス	キュー・マネージャー	名前リスト
clr	✓			
connect			✓	
crt	✓	✓	✓	✓
dlt	✓	✓	✓	✓
dsp	✓	✓	✓	✓
put	✓			
inq	✓	✓	✓	✓
get	✓			
passall	✓			
passid	✓			
set	✓	✓	✓	
setall	✓		✓	
setid	✓		✓	

MQI 呼び出しについての許可

altusr メッセージで代替ユーザー ID を使用する。

代替ユーザー ID の詳細については、*MQSeries アプリケーション・プログラミング・ガイド* を参照してください。

browse

BROWSE オプションを指定した MQGET 呼び出しを出して、キューからメッセージを取り出す。

connect

MQCONN 呼び出しを出して、指定のキュー・マネージャーにアプリケーションを接続する。

get MQGET 呼び出しを出して、キューからメッセージを取り出す。

inq MQINQ 呼び出しを出して、特定のキューの照会を行う。

put MQPUT 呼び出しを出して、特定のキューにメッセージを書き込む。

set MQSET 呼び出しを出して、MQI からキューに属性を設定する。

注: 複数のオプションを適用するようにキューをオープンする場合は、各オプションについての許可を持っている必要があります。

コンテキストについての許可

passall

すべてのコンテキストを指定のキューに渡す。すべてのコンテキスト・フィールドが元の要求からコピーされます。

passid

アイデンティティ・コンテキストを指定のキューに渡す。識別コンテキストは、要求の識別コンテキストと同じです。

setall

すべてのコンテキストを指定のキューに設定する。これは特別なシステム・ユーティリティによって使用されます。

setid

アイデンティティ・コンテキストを指定のキューに設定する。これは特別なシステム・ユーティリティによって使用されます。

コマンドについての許可

chg 指定のオブジェクトの属性を変更する。

clr 指定のキューをクリアする (PCF の Clear queue コマンドのみ)。

crt 指定のタイプのオブジェクトを作成する。

dlt 指定のオブジェクトを削除する。

dsp 指定のオブジェクトの属性を表示する。

一般操作についての許可

all オブジェクトに適用可能なすべての操作を使用する。

alladm

オブジェクトに適用可能なすべての管理操作を実行する。

allmqi オブジェクトに適用可能なすべての MQI 呼び出しを使用する。

戻りコード

- 0** 正常な操作です。
- 36** 与えられた引き数が無効です。
- 40** キュー・マネージャーは利用不能です。
- 49** キュー・マネージャーは停止中です。
- 69** ストレージが利用不能です。
- 71** 予期しないエラーです。
- 72** キュー・マネージャー名のエラーです。
- 133** オブジェクト名が不明です。
- 145** 予期しないオブジェクト名です。
- 146** オブジェクト名が脱落しています。

setmqaut

- 147 オブジェクト・タイプが脱落しています。
- 148 オブジェクト・タイプが無効です。
- 149 エンティティ名が脱落しています。
- 150 許可の指定が脱落しています。
- 151 許可の指定が無効です。

例

1. 次の例は、許可を与える対象となるオブジェクトが、キュー・マネージャー saturn.queue.manager のキュー orange.queue であることを指定するコマンドを示しています。

```
setmqaut -m saturn.queue.manager -n orange.queue -t queue -g tango +inq +alladm
```

許可は、ユーザー・グループ tango に与えられ、関連した許可リストは、ユーザー・グループ tango について次のことを指定します。

- MQINQ 呼び出しを出すことができる。
 - 該当のオブジェクトに対してすべての管理操作を実行する権限がある。
2. 次の例では、許可リストはユーザー・グループ foxy について次のことを指定します。
 - 指定のキューに対して、MQI から呼び出しを出すことができない。
 - 指定のキューに対してすべての管理操作を実行する権限がある。

```
setmqaut -m saturn.queue.manager -n orange.queue -t queue -g foxy -allmqi +alladm
```

関連コマンド

dspsmqaut 権限の表示

strmqcsv (コマンド・サーバーの開始)

目的

strmqcsv コマンドは、指定したキュー・マネージャーのコマンド・サーバーを開始するために使用します。これにより、MQSeries はコマンド・キューに送られるコマンドを処理できます。

構文

```
▶▶—strmqcsv—QMgrName————▶▶
```

必須パラメーター

QMgrName

コマンド・サーバーを開始させる対象となるキュー・マネージャーの名前を指定します。

戻りコード

- 0 コマンドは正常に終了しました。
- 10 コマンドは終了しましたが、予期しない結果がでました。
- 20 処理中にエラーが起きました。

例

次のコマンドは、キュー・マネージャー `earth` のコマンド・サーバーを開始します。

```
strmqcsv earth
```

関連コマンド

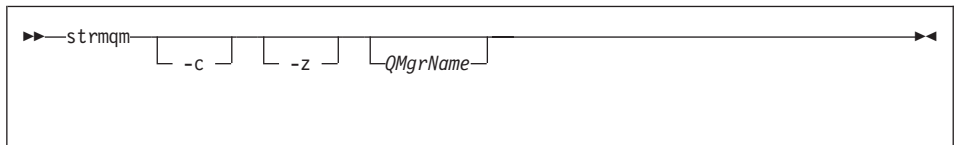
- endmqcsv** コマンド・サーバーを終了します。
- dspmqcsv** コマンド・サーバーの状況を表示します。

strmqm (キュー・マネージャーの開始)

目的

strmqm コマンドは、ローカル・キュー・マネージャーを開始するために使用します。キュー・マネージャーが接続や他の要求の処理に使用可能になった後にだけ、**strmqm** コマンドはコマンド行に戻します。

構文



オプション・パラメーター

- c** キュー・マネージャーを開始し、デフォルトおよびシステム・オブジェクトを再定義してから、キュー・マネージャーを停止します。(キュー・マネージャーのデフォルトおよびシステム・オブジェクトは最初に **crtmqm** コマンドにより作成されます。) キュー・マネージャーに属する既存のシステムおよびデフォルト・オブジェクトは、このフラグを指定すると置き換えられます。

QMGrName

開始させるローカル・キュー・マネージャーの名前を指定します。これを省略すると、デフォルト・キュー・マネージャーが開始します。

- z** エラー・メッセージを抑制します。

このフラグは、不要なエラー・メッセージを抑制するために MQSeries 内で使用します。このフラグを使用することによって情報が失われる可能性があるため、コマンド行からコマンドを入力する場合には、これは使用しないでください。

戻りコード

- 0** キュー・マネージャーは開始しました。
- 3** キュー・マネージャーは作成中です。
- 5** キュー・マネージャーは実行中です。
- 16** キュー・マネージャーは存在していません。
- 49** キュー・マネージャーは停止中です。
- 69** ストレージが利用不能です。

71 予期しないエラーです。

72 キュー・マネージャー名のエラーです。

例

次のコマンドは、キュー・マネージャー account を開始します。

```
strmqm account
```

関連コマンド

crtmqm	キュー・マネージャーを作成します。
dltmqm	キュー・マネージャーを削除します。
endmqm	キュー・マネージャーを終了します。

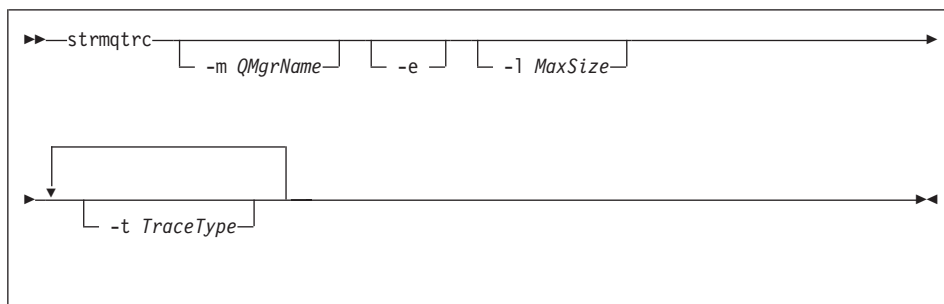
strmqtrc (MQSeries トレースの開始)

目的

strmqtrc コマンドは、トレースを有効にする場合に使用します。このコマンドは、トレースが使用可能であってもなくても使用できます。トレースがすでに有効である場合は、有効なトレース・オプションがこのコマンドの最新の呼び出しで指定したオプションに変更されます。

MQSeries トレースの使用の詳細については、280ページの『MQSeries トレースの使用』を参照してください。

構文



オプション・パラメーター

-m *QMgrName*

トレースされるキュー・マネージャーの名前です。名前を指定しない場合、デフォルト・キュー・マネージャーが使用されます。

指定するキュー・マネージャーは、実行中である必要はなく、存在していることさえ必要ありません。このため、キュー・マネージャーの作成や開始をトレースすることが可能です。

キュー・マネージャー名は、**-e** フラグと同じコマンド上に指定できます。トレース対象のあるエンティティに対して複数のトレース指定が適用される場合、トレースでは、指定されたすべてのオプションが有効になります。

- e** このフラグの指定により早期トレースを要求します。これは、プロセスが所属する MQSeries コンポーネントを認識する前にトレース情報が書き込まれるということです。このフラグを指定すると、すべてのキュー・マネ

ージャの任意のコンポーネントに属するどのプロセスでも早期処理がトレースされます。このフラグを指定しない場合、デフォルトでは早期トレースは実行されません。

-l *MaxSize*

MaxSize の値は、100 万バイト単位でトレース・ファイル (AMQnnnn.TRC) の最大サイズを指定します。たとえば、*MaxSize* を 1 に指定した場合、トレースのサイズは 100 万バイトに制限されます。

トレース・ファイルは、指定した最大値に達すると、その名前を AQnnnn.TRC から AMQnnnn.TRS に変更され、新しい AMQnnnn.TRC ファイルが開始されます。最大値の制限に達すると、すべてのトレース・ファイルが再始動されます。AMQnnnn.TRS ファイルの直前のコピーが存在している場合、そのファイルは削除されます。

-t *TraceType*

処理中にトレースできるポイントを定義します。以下のオプションの 1 つ以上を指定できます。

all	システム内のすべてのトレース・ポイントについてデータを出力します。-t フラグを指定しない場合も、これがデフォルトになります。
api	MQI および主なキュー・マネージャー・コンポーネントに関連するトレース・ポイントについてデータを出力します。
comms	通信ネットワークを介して流れるデータに関連するトレース・ポイントについてデータを出力します。
csflows	共通サービス内の処理フローに関連するトレース・ポイントについてデータを出力します。
lqmfloes	ローカル・キュー・マネージャー内の処理フローに関連するトレース・ポイントについてデータを出力します。
remotefloes	通信コンポーネント内の処理フローに関連するトレース・ポイントについてデータを出力します。
otherfloes	その他のコンポーネント内の処理フローに関連するトレース・ポイントについてデータを出力します。
csdata	共通サービス内の内部データ・バッファに関連するトレース・ポイントについてデータを出力します。

strmqtrc

lqmdata	ローカル・キュー・マネージャー内の内部データ・バッファに関連するトレース・ポイントについてデータを出力します。
remotedata	通信コンポーネント内の内部データ・バッファに関連するトレース・ポイントについてデータを出力します。
otherdata	その他のコンポーネント内の内部データ・バッファに関連するトレース・ポイントについてデータを出力します。
versiondata	実行中の MQSeries のバージョンに関連するトレース・ポイントについてデータを出力します。
commentary	MQSeries コンポーネント内の注釈に関連するトレース・ポイントについてデータを出力します。

このフラグを省略した場合は、すべてのトレース・ポイントが使用可能になり、完全なトレースが生成されます。

注: 複数のトレース・タイプを指定する場合は、それぞれのトレース・タイプに**必ず** 1 つずつ `-t` フラグを付ける必要があります。 `-t` フラグは、それぞれ有効なトレース・タイプが関連付けられていれば、いくつでも指定できます。

複数の `-t` フラグに同じトレース・タイプを指定しても、エラーにはなりません。

戻りコード

AMQ7024

このメッセージは、コマンドに無効な引き数を指定した場合に出されません。

AMQ8304

最大数である 9 個の並行トレースがすでに実行されています。

例

このコマンドは、QM1 というキュー・マネージャーについて、共通サービスおよびローカル・キュー・マネージャーからのデータのトレースを有効にします。

```
strmqtrc -m QM1 -t csdata -t lqmdata
```

関連コマンド

dspmqtrc 定様式トレース出力の表示
endmqtrc MQSeries トレースの終了

upgmqm (V2.2.0.1 キュー・マネージャーのアップグレード)

目的

このコマンドは、バージョン 2.2.0.1 のキュー・マネージャーをアップグレードして、MQSeries for Compaq NSK V5.1 で使用できるようにします。

upgmqm によって起動されるユーティリティは、呼び出し元の端末に進行メッセージを送信します。このユーティリティが完了すると、指定されたキュー・マネージャーは MQSeries for Compaq NSK V5.1 で使用できるようになります。バージョン 5.1 で新規に追加されたキュー・マネージャー属性には、デフォルト値が設定されます。これらの値は通常の方法で変更できます。

新しいバージョン 5.1 の状況サーバーは、バージョン 2.2.0.1 の MQSS サーバーと機能が異なっているため、このアップグレードではすべての既存の MQS-Status nm サーバーのクラスを既存の PATHWAY 構成から削除して、それらを 1 つのデフォルト MQS-Status00 サーバー・クラスおよび 1 つのデフォルト MQS-Queue00 サーバー・クラスに置き換えます。つまり、キュー・マネージャーをアップグレードした後、現在の MQSS サーバー・プロセスに割り当てられたオブジェクトを、新規の状況サーバー・プロセスまたはキュー・サーバー・プロセスに（どちらが適切かに応じて）再度割り当てる必要があります。（バージョン 5.1 では、ローカル・キューだけがキュー・サーバーに割り当てられて、他のすべてのオブジェクトは状況サーバーに割り当てられます。）さらに、PATHWAY 内の状況サーバー・クラスを参照している既存のスクリプトを変更しなければならないこともあります。

アップグレード中にクリーンアップしないことを選択した場合、ユーザーの都合に合わせて以下のファイルを削除できます。

表 11. アップグレード後に削除できるファイルの例

位置	ファイル	例
upgmqm の -p オプションによって指定されたサブボリューム	すべてのファイル	\$VOL.scratch
キュー・マネージャー 'S' サブボリューム	Z*	\$VOL.myv2201S

これらの例は、既存のキュー・マネージャーが \$VOL.myv2201? サブボリューム上に存在すると想定しています。

注: バージョン 2.2.0.1 のキュー・マネージャーが **upgmqm** によってアップグレードされない場合、そのキュー・マネージャーを操作する制御コマン

ドはありません。これには **dltmqm** も含まれており、基本データベースを開こうとすると、によって失敗します。バージョン 2.2.0.1 でのキュー・マネージャーはバージョン 5.1 で不要になるので、コードをアップグレードする前に除去するか、**upgmqm** を使用してアップグレードしてから除去しなければなりません。

構文

```
▶▶—upgmqm—-m QMgrName—-v DefaultQueueServer—-p SubvolumePath—————▶▶
▶—-s DefaultStatusServer—————▶▶
```

必須パラメーター

- m *QMgrName*
upgmqm ユーティリティーを適用する対象となるキュー・マネージャーの名前を指定します。
- v *DefaultQueueServer*
 キュー・マネージャー用のデフォルト・キュー・サーバーに指定された固有のプロセス名。
- p *SubvolumePath*
 ユーティリティーをアップグレードできるサブボリューム・パス (\$VOL.SUBVOL) は、作業ファイルに使用できます。このサブボリュームは、キュー・マネージャーと同じボリューム上にあることが**必要**です。パスのサブボリューム部分だけを指定します。ボリューム名は指定しません。**upgmqm** コマンドは、サブボリュームを有効なものとして受け入れる前に、そのサブボリュームがまだ存在していないことを検査します。
- s *DefaultStatusServer*
 キュー・マネージャー用のデフォルト状況サーバーに指定された固有のプロセス名。

例

この例では、バージョン 2.2.0.1 のキュー・マネージャー Myv2201qm を、デフォルトのキュー・サーバー名を \$MYQS、デフォルトの状況サーバー名を \$MYSS とし、作業ファイルにサブボリューム \$VOL.scratch を使用してアップグレー

upgmqm

ドします (ここで \$VOL は、キュー・マネージャーが存在するボリュームです)。

```
upgmqm -m Myv2201qm -p scratch -s $MYSS -v $MYQS
```

第3部 付録

付録A. MQSeries for Compaq NSK の概要

プログラムと部品番号

- 5724-A39 MQSeries for Compaq NSK、バージョン 5 リリース 1、部品番号 0791003 (日本語版: 29P3591)

ハードウェア要件

最小のハードウェア要件は次のとおりです。

- Guardian D45 またはそれ以降の D4x、G06 またはそれ以降の G0x によってサポートされているマシンの Compaq NSK レンジ。
- ユーザーが選択したネットワーク転送プロトコルをサポートしている特定のハードウェア。

さらに、TMF 監査スペースおよび MQSeries データベース用に指定されたスペース要件を満たす 1 つ以上のミラーリングされたデータ・ディスクを持つことが勧められています。

ソフトウェア要件

最小のソフトウェア要件は次のとおりです。

- TM/MP (TMF)、ENSCRIBE、および EMS を含む、Compaq NSK Guardian D45 またはそれ以降の D4x (K シリーズのハードウェア) および G06 またはそれ以降の G0x (S シリーズのハードウェア) オペレーティング・システム。
- オペレーティング・システムに適合する TS/MP (PATHWAY)。
- TCP および SNA ネットワーク転送を構成、コマンド指示、および制御するための SCF。

SNA 接続性の場合：

- オペレーティング・システムに適合する SNAX/APC および SNAX/XF または SNAX/APN

または

- Insession ICE バージョン 3.2 またはそれ以降

ソフトウェア要件

TCP/IP 接続性の場合：

- オペレーティング・システムに適合する TCP/IP

MQSeries の OSS に基づくパーツ (MQI バインディング、OSS アプリケーション、Java バインディング) を使用するには、オペレーティング・システムと互換性のある OSS プロダクトのバージョンが必要です。

トランザクション・ログは Compaq TM/MP (TMF) プロダクトによって保守されます。

セキュリティ

MQSeries for Compaq NSK は、読み取り、書き込み、実行、および除去操作のために USER および GROUP に対するアクセス制御を提供する、NSK ファイル・システムのセキュリティ機能を使用します。SAFEGUARD は MQSeries for Compaq NSK を使用するために必要ではありませんが、このプロダクトは SAFEGUARD 環境と互換性があります。

すべての MQSeries リソースは、グループ MQM 内の単一のユーザー ID によって所有されています。SCOBOL メニューまたは **runmqsc** を使用して MQSeries を管理するためには、割り当てられたユーザー ID でログインするか、または MQM グループにリンクされていなければなりません。

メンテナンス機能

MQSeries は、次のものを使用して機能します。

- PATHWAY 環境内の SCOBOL リクエスト構成画面を使用するメッセージ・キュー管理 (MQM) 機能。
- **runmqsc** コマンド行インターフェース。
- Compaq ネットワーク・プロトコル・オフリング用に TCP/IP および SNA 環境を保守する、構成、コマンド指示、および制御機能のための SCF ユーティリティ。
- ICE LU 6.2 インターフェースを制御するための、プロダクトと共に供給される ICE ユーティリティ。
- MQSeries Explorer (MQSeries for Compaq NSK には含まれていない)。
- リモート管理に標準の PCF コマンドを使用する他の製品またはユーティリティ。

互換性

保守修正 PTF U473441 を適用した MQI for MQSeries for Compaq NSK V5.1 は、MQSeries for Tandem NonStop Kernel V2.2.0.1 上で実行する既存のアプリケーションと互換性があります。

サポートされるコンパイラー

MQSeries for Compaq NSK V5.1 は、共通ランタイム環境 (CRE) を使用してすべてのオブジェクトをリンクするように構築されています。この方法を使用する場合、バージョン 2.2.0.1 より前の MQI のユーザーは以下の事柄が必要になります。

1. D45 より前の COBOL および C オブジェクト・コードは、D45 (またはそれ以降) で再コンパイルして CRE リンテージを組み込む必要があります。
2. D45 より前のすべての TAL オブジェクト・コードは、D45 (またはそれ以降の) コンパイラーで再コンパイルする必要があり、さらにその TAL プログラムが *Common Run-time Environment Programmer's Guide* で指定されている特別なプログラミング上の考慮事項に適合していることを確認しなければなりません。これらのプログラミング上の考慮事項に関する詳細は、*TAL Programmer's Guide* で説明されています。
3. D45 上のネイティブ・コンパイラーによって生成されたオブジェクト・コードについては、別のバインディングが提供されます。
4. C C プログラムは WIDE メモリー・モデル (32 ビット整数) を使用しなければなりません。
5. COBOL プログラムは CRE の要件に適合しなければなりません。
6. TAL プログラムでは、MQI 関数に渡されるすべての整数が 32 ビットでなければ (または \$INT32() マクロによって 32 ビットにされなければ) なりません。

MQSeries プログラムそのものは、Guardian NSK 用のネイティブ・モード・ツールを使用してコンパイルおよびリンクされます。ネイティブ・モードのアプリケーションは通常はキュー・マネージャー SRL に直接リンクしますがアプリケーションが既に Private SRL を使用している場合は例外です。その場合、アプリケーションが使用できるものは多くても 1 つの Private SRL に制限されるので、アプリケーションは静的 MQI バインディング・ライブラリーにリンクするか、またはアプリケーションの専用 SRL 内に存在するコードを MQSeries SRL と結合して新しい Private SRL にする必要があります。

ライセンス管理

システム・タイプを入力してプログラム権利を定義しなければなりません。このパラメーターはインストール時に入力することも、またはそれ以降でライセンス・アップグレードを購入したときに入力することもできます。始動時にこの値が物理的な Compaq マシン構成と比較されて検査されます。ライセンス登録およびプログラム権利が不十分な場合、警告メッセージが出されます。

言語の選択

提供されるメッセージ・テキスト・ファイルは、Compaq NSK オペレーティング・システムに固有の、7 ビットの文字セットでエンコードされます。

MQSeries for Compaq NSK では、プロダクトのインストール時に各国語を指定することができます。デフォルトのメッセージ言語は U.S. English です。

国際化対応

MQSeries for Compaq NSK によって、キュー・マネージャーの作成時に CCSID を指定できます。(キュー・マネージャーの作成後に CCSID 呼び出しを変更することもできます。) キュー・マネージャー CCSID は 819 をデフォルトとします。MQSeries for Compaq NSK は、キュー・マネージャーの構成済み CCSID への文字セット変換をサポートします。ユーロ文字のサポートを提供するものを含め、MQSeries for Compaq NSK キュー・マネージャーに指定できる CCSID についての詳細は、*MQSeries アプリケーション・プログラミング・リファレンス* を参照してください。

付録B. システム・デフォルト

crtmqm 制御コマンドを使用してキュー・マネージャーを作成する場合、システム・オブジェクトおよびデフォルト・オブジェクトが自動的に作成されます。

- システム・オブジェクトは、キュー・マネージャーまたはチャンネルの操作に必要な MQSeries オブジェクトです。
- デフォルト・オブジェクトは、オブジェクトのすべての属性を定義します。ローカル・キューなどのオブジェクトを作成する場合、明示的に指定していない属性はすべてデフォルト・オブジェクトから継承されます。

表 12. キューのシステムおよびデフォルト・オブジェクト

オブジェクト名	説明
SYSTEM.DEFAULT.ALIAS.QUEUE	デフォルト別名キュー
SYSTEM.DEFAULT.LOCAL.QUEUE	デフォルト・ローカル・キュー
SYSTEM.DEFAULT.MODEL.QUEUE	デフォルト・モデル・キュー
SYSTEM.DEFAULT.REMOTE.QUEUE	デフォルト・リモート・キュー
SYSTEM.DEAD.LETTER.QUEUE	サンプルの送達不能 (未配布メッセージ) キュー
SYSTEM.DEFAULT.INITIATION.QUEUE	デフォルト開始キュー
SYSTEM.CICS.INITIATION.QUEUE	デフォルト CICS [®] 開始キュー
SYSTEM.ADMIN.COMMAND.QUEUE	管理者コマンド・キュー。リモート MQSC コマンドおよび PCF コマンドに使用されます。
SYSTEM.MQSC.REPLY.QUEUE	MQSC 応答先キュー。これは、リモート MQSC コマンドに対する応答の一時動的キューを作成するモデル・キューです。
SYSTEM.ADMIN.QMGR.EVENT	キュー・マネージャー・イベントのイベント・キュー
SYSTEM.ADMIN.PERFM.EVENT	パフォーマンス・イベントのイベント・キュー
SYSTEM.ADMIN.CHANNEL.EVENT	チャンネル・イベントのイベント・キュー
SYSTEM.CHANNEL.INITQ	チャンネル開始キュー

システム・デフォルト

表 12. キューのシステムおよびデフォルト・オブジェクト (続き)

オブジェクト名	説明
SYSTEM.CHANNEL.SYNCQ	チャンネルの同期データを保持するキュー (このオブジェクトは作成されますが、MQSeries for Compaq NSK では使用されません。チャンネル同期情報は TM/MP 保護データベースに保管されます。)
SYSTEM.CLUSTER.COMMAND.QUEUE	リポジトリ・キュー・マネージャーにメッセージを転送するために使用されるキュー
SYSTEM.CLUSTER.REPOSITORY.QUEUE	すべてのリポジトリ情報を保管するために使用されるキュー
SYSTEM.CLUSTER.TRANSMIT.QUEUE	クラスターに対するすべてのメッセージのための伝送キュー

表 13. チャンネルのシステムおよびデフォルト・オブジェクト

オブジェクト名	説明
SYSTEM.DEF.SENDER	デフォルト送信側チャンネル
SYSTEM.DEF.SERVER	デフォルト・サーバー・チャンネル
SYSTEM.DEF.RECEIVER	デフォルト受信側チャンネル
SYSTEM.DEF.REQUESTER	デフォルト要求側チャンネル
SYSTEM.DEF.SVRCONN	デフォルト・サーバー接続チャンネル
SYSTEM.DEF.CLNTCONN	デフォルト・クライアント接続チャンネル
SYSTEM.AUTO.RECEIVER	動的受信側チャンネル
SYSTEM.AUTO.SVRCONN	動的サーバー接続チャンネル
SYSTEM.DEF.CLUSRCVR	クラスターのためのデフォルト受信側チャンネル。クラスター内のキュー・マネージャー上に CLUSRCVR チャンネルを作成するときに指定しなかった属性のデフォルト値を提供するために使用されます。
SYSTEM.DEF.CLUSSDR	クラスターのためのデフォルト送信側チャンネル。クラスター内のキュー・マネージャー上に CLUSSDR チャンネルを作成するときに指定しなかった属性のデフォルト値を提供するために使用されます。

表 14. 名前リストのシステムおよびデフォルト・オブジェクト

オブジェクト名	説明
SYSTEM.DEFAULT.NAMELIST	デフォルト名前リスト

表 15. プロセスのシステムおよびデフォルト・オブジェクト

オブジェクト名	説明
SYSTEM.DEFAULT.PROCESS	デフォルト・プロセス定義

付録C. MQSeries for Compaq NSK 用の TACL 環境の設定

MQSeries は、いくつかの Compaq NSK 環境変数、つまり PARAM を作成して使用します。これらの PARAM を設定するときは、以下の点を考慮してください。

- MQDEFAULTPREFIX PARAM は、すべてのプログラムの環境に存在している必要があります。 **crtmqm** コマンドによって自動的に確立される TS/MP (Pathway) 構成によって、これらの PARAM がキュー・マネージャー・サーバー・プロセス用に正しく設定されることが保証されます。 MQSeries アプリケーションおよび制御コマンドのユーザーは、使用する TACL および TS/MP 構成でも必ずこれらの変数を指定しなければなりません。
- PARAM ステートメントを TACLSTM ファイルに含めて、ログオン時にこれらの PARAM が正しく作成されて、TACL から実行されるプログラムが正しい値を継承するようにします。以下の環境変数も変更して、MQSeries 実行可能プログラムの位置を指定する必要があります。

```
#SET #PMSEARCH $SYSTEM.ZMQSEX [ #PMSEARCH ]
```

MQCONNECTTYPE

この PARAM が存在する場合、それを使用してアプリケーションが FASTPATH 接続を使用する機能を使用不可にすることができます。この PARAM が値 STANDARD に設定されている場合、それらが FASTPATH を要求したとしても、アプリケーションが使用できるのは STANDARD 接続だけとなります。その他の値に設定されている場合は、PARAM が指定されていない場合と同じ処理になります (そのため、アプリケーションは要求すれば FASTPATH 接続を使用できます)。

MQDEFAULTPREFIX

インストール済みのサブボリューム ZMQSSYS を含むボリュームの名前。この PARAM はすべての環境で正しく定義されている必要があります。

たとえば、次のようにします。

```
PARAM MQDEFAULTPREFIX $data00
```

MQEMSEVENTS

この PARAM は、MQSeries EMS イベントを使用可能にします。たとえば、MQSeries 用のすべての EMS イベントをオンに切り替えるためには、以下のように PARAM MQEMSEVENTS を設定します。

```
PARAM MQEMSEVENTS 127
```

MQMACHINIFILE

インストール用の MQSINI ファイルの場所。デフォルト値は、*MQDEFAULTPREFIX.ZMQSSYS.MQSINI* です。この PARAM が必要なのは、デフォルト以外の場所が必要とされる場合だけです。

たとえば、次のようにします。

```
PARAM MQMACHINIFILE $data00.altinst.mqsini
```

MQRDF

この PARAM が ON に設定されている場合、MQSeries は削除操作の動作を変更して、監査済みファイルの RDF と共に機能するようにします。この PARAM が定義されていない場合、または ON 以外の値に設定されている場合、MQSeries は操作関数を通常どおりに削除します。この PARAM を使用する場合、それを管理プログラムを実行するすべてのユーザーの TACL 環境内、およびキュー・マネージャー内に構成されているすべての TS/MP サーバー・クラスの環境内に設定しなければなりません。

MQRDFFUPPROCESSNAME

この PARAM は、**cleanrdf** ユーティリティーによってのみ解釈されます。これを使用して、**cleanrdf** ユーティリティーが作成する FUP サーバー・プロセスに割り当てられる Guardian プロセス名を指定することができます。この PARAM が定義されていない場合、FUP サーバー・プロセス名は、オペレーティング・システムによって割り当てられます。

MQRDFFUPPROGNAME

この PARAM は、**cleanrdf** ユーティリティーによってのみ解釈されま

す。これを使用して、ユーティリティーが使用する FUP 実行可能ファイルの完全修飾名を指定できます。デフォルト値は、<default system name>.\$SYSTEM.SYS00.FUP です。

MQSNOAUT

crtmqm の実行時にこの PARAM が 1 に設定されている場合、OAM が使用不可となっている新規のキュー・マネージャーが作成されます。

たとえば、次のようにします。

```
PARAM MQSNOAUT 1
```

MQLISTENPORTNUM

この PARAM が設定されている場合、TCP/IP リスナー・プロセスはそれを使用して、listen するためのポートを検出します。この PARAM が指定されていない場合、ポートは QMINI ファイルの TCP/IP スタンザから決定されます。

SAVE-ENVIRONMENT ON

アプリケーション・プログラムを実行する際に必要とされ、共通ランタイム環境 (CRE) が PARAM を、環境からアプリケーション・プログラムに確実に渡すようにします。

たとえば、次のようにします。

```
PARAM SAVE-ENVIRONMENT ON
```

この PARAM が設定されていない場合、アプリケーションはキュー・マネージャー名エラーを示す戻りコード 2058 を受け取ります。

この PARAM が必要なのは、非ネイティブのプログラムとして実行している TAL または COBOL アプリケーションだけです。

キュー・サーバーの調整パラメーター

以下の PARAM は、キュー・サーバーの TACL 環境内に定義されている場合、さまざまなハウスキーピング操作用のキュー・サーバーの組み込みデフォルトを指定変更するために使用できます。

MQQSHKEEPINT

この PARAM が設定されている場合、秒数を示す数値を指定して、キ

キュー・サーバーのデフォルトのハウスキーピング・インターバル (60 秒) を変更できます。ハウスキーピング・インターバルは、キュー・サーバーが有効期限切れのメッセージを検出するためにキューを調べて、操作の最適化のためにメモリー使用状況を調べるための頻度を制御します。

MQQSSIGTIMEOUT

この PARAM が設定されている場合、秒数を示す数値を設定して、MQGMO_SET_SIGNAL オプションを指定した MQGET を開始したアプリケーションにシグナル IPC を送達するデフォルトのタイムアウト (60 秒) を指定変更できます。キュー・サーバーが (シグナル生成の条件が満たされた後に) このタイムアウト期限内にシグナルを送達することができない場合、キュー・サーバーはその事実をログに記録してシグナルを取り消します。

MQQSMAXBATCHEXPIRE

この PARAM が設定されている場合、キュー・サーバーがハウスキーピング中に単一トランザクション内で廃棄する有効期限切れ持続メッセージ数のデフォルトの最大数 (100) を、数値を指定して指定変更できます。持続メッセージの有効期限が切れた場合、それらはキュー・データベースから物理的に除去される必要があります、これには内部 TM/MP トランザクションが必要になります。この PARAM を使用することによって、単一の TM/MP トランザクション内で除去されるメッセージの最大数を制御できます。

MQQSMAXMSGSEXPIRE

この PARAM が設定されている場合、キュー・サーバーの単一のハウスキーピング・インスタンス内で検出されて廃棄される有効期限切れメッセージ数のデフォルトの最大数 (300) を、数値を指定して変更できます。

付録D. コマンド・セットの比較

表16 から 415ページの表21 は、異なる管理コマンド・セットから使用することの可能な機能を比較しています。

注: MQSeries for Compaq NSK に当てはまる MQSC コマンドだけが示されています。

表 16. キュー・マネージャーの管理用のコマンド

PCF	MQSC	制御
Change Queue Manager	ALTER QMGR	-
(キュー・マネージャーの作成)*	-	crtmqm
(キュー・マネージャーの削除)*	-	dltmqm
Inquire Queue Manager	DISPLAY QMGR	-
(キュー・マネージャーの停止)*	-	endmqm
Ping Queue Manager	PING QMGR	-
(キュー・マネージャーの開始)*	-	strmqm
注: * PCF コマンドとしては使用できません。		

表 17. コマンド・サーバーの管理用のコマンド

説明	制御
コマンド・サーバーの表示	dspmqcsv
コマンド・サーバーの停止	endmqcsv
コマンド・サーバーの開始	strmqcsv
注: 制御コマンドを使用する代わりに、41ページの『TS/MP (PATHWAY) 管理』で説明しているように、PATHCOM コマンドを使用することもできます。このグループに相当する MQSC コマンドまたは PCF コマンドはありません。	

コマンド・セットの比較

表 18. キューの管理用のコマンド

PCF	MQSC
Change Queue	ALTER QLOCAL ALTER QALIAS ALTER QMODEL ALTER QREMOTE
Clear Queue	CLEAR QLOCAL
Copy Queue	DEFINE QLOCAL(x) LIKE(y) DEFINE QALIAS(x) LIKE(y) DEFINE QMODEL(x) LIKE(y) DEFINE QREMOTE(x) LIKE(y)
Create Queue	DEFINE QLOCAL DEFINE QALIAS DEFINE QMODEL DEFINE QREMOTE
Delete Queue	DELETE QLOCAL DELETE QALIAS DELETE QMODEL DELETE QREMOTE
Inquire Queue	DISPLAY QUEUE
Inquire Queue Names	DISPLAY QUEUE
注: これらの機能に使用できる制御コマンドはありません。	

表 19. プロセスの管理用のコマンド

PCF	MQSC
Change Process	ALTER PROCESS
Copy Process	DEFINE PROCESS(x) LIKE(y)
Create Process	DEFINE PROCESS
Delete Process	DELETE PROCESS
Inquire Process	DISPLAY PROCESS
Inquire Process Names	DISPLAY PROCESS
注: これらの機能に使用できる制御コマンドはありません。	

表 20. チャンネルの管理用のコマンド

PCF	MQSC	制御
Change Channel	ALTER CHANNEL	-
Copy Channel	DEFINE CHANNEL(x) LIKE(y)	-
Create Channel	DEFINE CHANNEL	-
Delete Channel	DELETE CHANNEL	-
Inquire Channel	DISPLAY CHANNEL	-
Inquire Channel Names	DISPLAY CHANNEL	-
Inquire Channel Status	DISPLAY CHSTATUS	-
Ping Channel	PING CHANNEL	-
Reset Channel	RESET CHANNEL	-
Resolve Channel	RESOLVE CHANNEL	-
Start Channel	START CHANNEL	runmqchl
Start Channel Initiator	-	runmqchi
Start Channel Listener	-	runmqlsr
Stop Channel	STOP CHANNEL	-
注: MQSeries for Compaq NSK では、TS/MP または制御コマンド runmqlsr を使用して TCP/IP チャンネル・リスナーを開始します。詳細については、41ページの『TCP/IP リスナーの指定および制御』および 375ページの『runmqlsr (リスナーの実行)』を参照してください。		

表 21. その他の制御コマンド

説明	制御
キュー・ボリューム、キュー・サーバーのメッセージ・ストレージ・オプションの更新	altmqfls
MQSeries プリンシパルの追加、削除、または更新	altmqusr
RDF ハウスキーピング・ユーティリティー	cleanrdf
クライアント・チャンネル定義テーブルの変換	cvclchl
MQSeries 変換出口の作成	crtmqevx
権限の表示	dspmqaout
オブジェクトが使用するファイルの表示。 キュー・サーバー・メッセージ・ストレージ・オプションはオブジェクトについて構成済み。	dspmqfls
MQSeries 定様式トレース出力の表示コマンド	dspmqtrc
MQSeries プリンシパルの表示	dspmqusr

コマンド・セットの比較

表 21. その他の制御コマンド (続き)

説明	制御
MQSeries トレースの終了	endmqtrc
MQSeries for Compaq NSK のインストール	instmqm
送達不能キュー・ハンドラーの実行	runmqdlq
MQSC コマンドの実行	runmqsc
トリガー・モニターの実行	runmqtrm
権限の設定またはリセット	setmqaut
MQSeries トレースの開始	strmqtrc
V2201 キュー・マネージャーのアップグレード	upgmqm
注: 制御コマンド runmqtrm を使用する代わりに、41ページの『TS/MP (PATHWAY) 管理』で説明されているように、PATHCOM コマンドを使用することもできます。このグループに相当する MQSC コマンドまたは PCF コマンドはありません。	

付録E. 手動によるキュー・マネージャーの停止および除去

キュー・マネージャーを停止および除去する通常の方法が失敗した場合、ここで説明する、さらに思い切った方法を実行することができます。

手動によるキュー・マネージャーの停止

endmqm コマンドを使用してキュー・マネージャーを停止する通常の方法は、キュー・マネージャー内部に障害が生じている際にも使用可能です。例外的な状況で、キュー・マネージャーを停止するこの方法が失敗した場合、以下の手順を使用して手動で停止してください。

1. まだ実行中のキュー・マネージャー・プログラムのプロセス ID を見つけます。
2. キュー・マネージャーのデータ・サブボリューム内の TRACEOPT ファイル上にある FUP LISTOPENS は、キュー・マネージャーに属するプロセスの CPU, PIN を提供します。
3. まだ実行中のキュー・マネージャー・プロセスを終了します。 **STOP** コマンドを、前のステップで検出したプロセス ID を指定して使用します。
以下の順序でプロセスを終了します。
 - a. MQECBOSS - EC Boss
 - b. MQEC - ECs
 - c. まだ実行中の他のプロセス

注: キュー・マネージャーを手動で終了すると、FFST が取られて、FD ファイルが生成されることがあります。これはキュー・マネージャーの欠陥と見るべきではありません。

キュー・マネージャーを前に示した方法で終了した場合も、再始動時には通常の方法で行います。

キュー・マネージャーを手動で停止した後でそれを削除したい場合、通常の方法で **dltmqm** コマンドを使用します。何かの理由でこのコマンドによってキュー・マネージャーを削除できない場合、418ページの『手動によるキュー・マネージャーの除去』で詳述されている手動プロセスを使用できます。

手動によるキュー・マネージャーの除去

キュー・マネージャーを手動で除去するには、以下のようにします。

1. 除去したいキュー・マネージャーに対してキュー・マネージャー・プロセスが実行されていないことを確認します。
2. MQSINI ファイルを編集してキュー・マネージャー・スタanzasを除去し、必要であればデフォルトのキュー・マネージャー・スタanzasを変更します。スタanzasを削除する前に、キュー・マネージャー・ファイルの場所を記録します。
3. FUP PURGE コマンドを使用して、キュー・マネージャーのすべてのサブボリュームにあるすべてのファイルを削除します。たとえば、FUP PURGE \$VOL.QMSVOL*.* を実行します。

付録F. MQSeries および Compaq NonStop Server for Java

MQSeries for Compaq NSK は、Compaq NonStop Server for Java バージョン 1.5 および以降と互換性があります。この製品は、MQSeries Java クラスのセット全体をサポートしています。これらの詳細については、*MQSeries Java の使用* で説明されています。

MQSeries は次の 2 つの方法で、NonStop Server for Java と連動します。

- Compaq iTP Webserver に関連して実行するサーブレットとして
- コマンド行から直接実行する Java アプリケーションとして

Compaq NonStop Server for Java 1.5 Reference は、サーブレットの使用について説明しています。

サーブレットまたはアプリケーションを使用して Java から MQSeries にアクセスするためには、以下の MQSeries プロダクト・ライブラリーにリンクする NonStop JVM のカスタム・バージョンを作成する必要があります。

- Product SRL MQSRLLIB、Guardian ZMQSEXE サブボリュームから
- Java バインディング・アーカイブ libMQSESSION、/opt/mqm/lib から

Compaq Java 資料では、JVM を再リンクする方法について説明しています。MQSeries サンプル・ディレクトリー (/opt/mqm/samp) には、以下の 2 つのサンプル make ファイルが含まれています。

MakeJVM.smp

このサンプル make ファイルは、MQSeries へのアクセスを提供する NonStop JVM を再作成する方法を例示しています。この make ファイルを使用するには、MQSeries プロダクト・ライブラリーのインストール場所を反映するように変更します。

MakeJava.smp

Java アプリケーションを作成するためのサンプル make ファイルです。

トランザクションに関する考慮事項

Java 言語は、TM/MP インターフェースへの直接アクセスを提供していません。MQSeries は、JTS の *Current Class* を介して Java 内のトランザクションをサポートしています。以下の表は、*Current* インターフェースについて説明しています。

表 22. Java 言語インターフェース

関数	説明
<code>begin()</code>	新規のトランザクションを開始して、それを呼び出しスレッドに関連付けます。
<code>commit(boolean)</code>	呼び出しスレッドに関連したトランザクションをコミットします。
<code>get_control()</code>	呼び出しスレッドに関連したトランザクションを表す <code>Control</code> オブジェクトを取得します。
<code>get_status()</code>	呼び出しスレッドに関連したトランザクションの状況を取得します。
<code>get_transaction_name()</code>	呼び出しスレッドに関連したトランザクションの記述名を取得します。
<code>resume(ControlRef)</code>	トランザクションと呼び出しスレッドとの関連を設定または再開します。
<code>rollback()</code>	呼び出しスレッドに関連したトランザクションをロールバックします。
<code>suspend()</code>	呼び出しスレッドとトランザクション・コンテキストとの関連を中断します。

JTS の詳細については、Compaq NSK NonStop Java の資料または JDK の資料を参照してください。

付録G. MQSeries for Compaq NSK によってサポートされる MQSC

この付録では、MQSeries for Compaq NSK によってサポートされる MQSeries コマンド (MQSC) をリストします。これらのコマンドの構文の詳細については、*MQSeries MQSC コマンド・リファレンス* を参照してください。

表 23. *MQSeries for Compaq NSK* によってサポートされる *MQSC*

コマンド	説明	<i>MQSeries</i> コマンド・リファレンスの説明と同じか
ALTER CHANNEL	チャンネル属性の変更。	はい、ただし 425ページの『CONNAME』での説明を除く。
ALTER NAMELIST	名前リストを更新します。	はい
ALTER PROCESS	プロセス属性を変更します。	はい
ALTER QALIAS	別名キューの属性を変更します。	はい
ALTER QLOCAL	ローカル・キューの属性を変更します。	はい、ただし 425ページの『HARDENBO と NOHARDENBO』での説明を除く。
ALTER QMGR	キュー・マネージャー属性を変更します。	はい、ただし 425ページの『MAXUMSGS と MAXHANDS』での説明を除く。
ALTER QMODEL	モデル・キューの属性を変更します。	はい、ただし 425ページの『HARDENBO と NOHARDENBO』での説明を除く。
ALTER QREMOTE	リモート・キュー、キュー・マネージャー別名、または返信先キュー別名のローカル定義の属性を変更します。	はい

MQSeries コマンド

表 23. MQSeries for Compaq NSK によってサポートされる MQSC (続き)

コマンド	説明	MQSeries コマンド・リファレンスの説明と同じか
CLEAR QLOCAL	ローカル・キューからメッセージを消去します。	はい
DEFINE CHANNEL	チャンネル定義を作成します。	はい、ただし 425ページの『CONNNAME』での説明を除く。
DEFINE NAMELIST	名前リストを定義します。	はい
DEFINE PROCESS	プロセス定義を作成します。	はい
DEFINE QALIAS	別名キュー定義を作成します。	はい
DEFINE QLOCAL	ローカル・キュー定義を作成します。	はい、ただし 425ページの『HARDENBO と NOHARDENBO』での説明を除く。
DEFINE QMODEL	モデル・キュー定義を作成します。	はい、ただし 425ページの『HARDENBO と NOHARDENBO』での説明を除く。
DEFINE QREMOTE	リモート・キュー、キュー・マネージャー別名、または返信先キュー別名のローカル定義を作成します。	はい
DELETE CHANNEL	チャンネル定義を削除します。	はい
DELETE NAMELIST	名前リストを削除します。	はい
DELETE PROCESS	プロセス定義を削除します。	はい
DELETE QALIAS	別名キュー定義を削除します。	はい
DELETE QLOCAL	ローカル・キュー定義を削除します。	はい
DELETE QMODEL	モデル・キュー定義を削除します。	はい

表 23. MQSeries for Compaq NSK によってサポートされる MQSC (続き)

コマンド	説明	MQSeries コマンド・リファレンスの説明と同じか
DELETE QREMOTE	リモート・キューのローカル定義を削除します。	はい
DISPLAY CHANNEL	チャンネル定義を表示します。	はい、ただし 425ページの『CONNAME』での説明を除く。
DISPLAY CHSTATUS	1 つ以上のチャンネルの状況を表示します。	はい、ただし 424ページの『チャンネル状況情報 (DISPLAY CHSTATUS)』および 425ページの『CONNAME』での説明を除く。
DISPLAY CLUSQMGR	1 つ以上のチャンネルの状況を表示します。	はい、ただし 425ページの『CONNAME』での説明を除く。
DISPLAY NAMELIST	名前の一覧を表示します。	はい
DISPLAY PROCESS	プロセス定義を表示します。	はい
DISPLAY QMGR	キュー・マネージャー属性を表示します。	はい、ただし 425ページの『MAXUMSGS と MAXHANDS』での説明を除く。
DISPLAY QUEUE	キュー属性を表示します。	はい、ただし 425ページの『HARDENBO と NOHARDENBO』での説明を除く。
PING CHANNEL	チャンネルをテストします。	はい
PING QMGR	キュー・マネージャーがコマンドに回答しているかどうかをテストします。	はい
REFRESH CLUSTER	ローカルに保持しているクラスター情報をすべて廃棄して、強制的に再作成します。	はい

MQSeries コマンド

表 23. MQSeries for Compaq NSK によってサポートされる MQSC (続き)

コマンド	説明	MQSeries コマンド・リファレンスの説明と同じか
RESET CHANNEL	チャンネルのメッセージ・シーケンス番号をリセットします。	はい
RESET CLUSTER	クラスターに特別の操作を実行します。	はい
RESOLVE CHANNEL	送信側またはサーバー・チャンネル上の未確定メッセージを解決します。	はい
RESUME QMGR	ローカル・キュー・マネージャーが処理に再び使用可能になり、メッセージを送信できることを、クラスター内の他のキュー・マネージャーに通知します。	はい
START CHANNEL	チャンネルを開始します。	はい
STOP CHANNEL	チャンネルを停止します。	はい
SUSPEND QMGR	ローカル・キュー・マネージャーが処理に使用できないので、メッセージを送信できないことを、クラスター内の他のキュー・マネージャーに通知します。	はい

複数の MQSC コマンドでスクリプトを作成する場合、各行が 72 文字を超えないようにする必要があります。

MQSC の属性

この節では、MQSeries for Compaq NSK に固有の MQSC 属性についての情報を提供します。

チャンネル状況情報 (DISPLAY CHSTATUS)

DISPLAY CHSTATUS コマンドは、MQSeries MQSC コマンド・リファレンスで説明されているようにインプリメントされます。ただし例外として、チャンネル状況が更新されるのはバッチ処理の境界においてだけです。チャンネルのパフォーマンスに影響を与える可能性があるため、チャンネル状況情報は、メッセー

ジの転送ごとに更新されることはありません。つまり、現行のセットと保管済みセットとで共通状況データ値が等しいということです。

MAXUMSGS と MAXHANDS

キュー・マネージャーのオブジェクト属性 MAXUMSGS および MAXHANDS は無視されます。これにより、以下のコマンドが影響を受けます。

```
ALTER QMGR
DISPLAY QMGR
```

HARDENBO と NOHARDENBO

MQSeries for Compaq NSK では、ローカルおよびモデル・キュー属性 HARDENBO および NOHARDENBO は無視されます。メッセージの *Backoutcount* は、持続メッセージに対しては必ず固定され、非持続メッセージでは固定されません。これにより、以下のコマンドが影響を受けます。

```
ALTER QLOCAL
ALTER QMODEL
DEFINE QLOCAL
DEFINE QMODEL
DISPLAY QUEUE
```

CONNAME

TCP チャンネルの CONNAME 属性は、値の先頭で追加のフィールドをオプションで取り、チャンネルに使用する、指定した Guardian TCP/IP Server プロセスの名前を指定します。これにより、以下のコマンドが影響を受けます。

```
ALTER CHANNEL
DEFINE CHANNEL
DISPLAY CHANNEL
DISPLAY CHSTATUS
DISPLAY CLUSQMGR
```

起動されたプログラムの USERDATA

MQSC DEFINE PROCESS または ALTER PROCESS の USERDATA 属性を介してトリガー・モニターに渡されるデータが、スペースを含むストリングである場合、そのストリングを二重引用符で囲む必要があります。たとえば、USERDATA -o \$DISK.VOLUME.PROGRAM -d 1 をトリガー・モニターに渡す場合、次のように二重引用符で囲んで MQSC への入力として指定する必要があります。

HARDENBO と NOHARDENBO

```
'" -o $DISK.VOLUME.PROGRAM -d 1"'
```

MQSC を介してプロセス定義を表示する場合、以下のようになります。

```
"-o $DISK.VOLUME.PROGRAM -d 1"
```

出口名のオブジェクトの属性としての使用

オブジェクトの属性内に出口名が指定される場合は必ず、それらは MQSeries for Compaq NSK に固有の形式になります。

付録H. アプリケーション・プログラミング・リファレンス

続くいくつかの節は、MQSeries for Compaq NSK に新規のものであり、*MQSeries アプリケーション・プログラミング・リファレンス* と併せて使用してください。

構造データ・タイプ

この節では、データ・タイプの変更点について説明します。

構造データ・タイプ	V2.x でサポート	V5.1 でサポート	MQSeries アプリケーション・プログラミング・リファレンス での説明どおり機能するか
MQBO - 開始オプション	いいえ	いいえ	
MQCH - CICS プリッジ・ヘッダー	いいえ	はい	はい
MQCNO - 接続オプション	いいえ	はい	はい、ただし付加的な注あり。詳細は、428ページの『MQCNO - 接続オプション』を参照。
MQDH - 配布ヘッダー	いいえ	はい	はい
MQDLH - 送達不能文字ヘッダー	はい	はい	はい
MQGMO - Get メッセージ・オプション	はい	はい	はい、ただし付加的な注あり。詳細は、429ページの『MQGMO - Get メッセージ・オプション』を参照。
MQIH - IMS プリッジ・ヘッダー	はい	はい	はい
MQMD - メッセージ記述子	はい	はい	はい、ただし付加的な注あり。詳細は、430ページの『MQMD - メッセージ記述子』を参照。

構造データ・タイプ

構造データ・タイプ	V2.x でサポート	V5.1 でサポート	MQSeries アプリケーション・プログラミング・リファレンス での説明どおり機能するか
MQMDE - メッセージ記述子拡張子	いいえ	はい	はい
MQOD - オブジェクト記述子	はい	はい	はい
MQOR - オブジェクト・レコード	いいえ	はい	はい
MQPMO - Put メッセージ・オプション	はい	はい	はい、ただし付加的な注あり。詳細は、431ページの『MQPMO - Put メッセージ・オプション』を参照。
MQPMR - Put メッセージ・レコード	いいえ	はい	はい
MQRMH - メッセージ参照ヘッダー	いいえ	はい	はい
MQRR - 応答レコード	いいえ	はい	はい
MQTM - トリガー・メッセージ	はい	はい	はい
MQTMC2 - トリガー・メッセージ文字フォーマット	はい	はい	はい
MQWIH - ワークロード情報ヘッダー	いいえ	はい	はい
MQXQH - 伝送キュー・ヘッダー	はい	はい	はい

この節では、以下の MQSeries 構造データ・タイプについて説明します。

MQCNO - 接続オプション

MQCNO データ構造は、MQSeries アプリケーション・プログラミング・リファレンス で示されているとおりですが、以下の追加の注記があります。

- 実行の単位はプロセスとして定義されます。
- MQCNO_FASTPATH_BINDING を使用できるのは、キュー・マネージャーに単一接続のあるプロセス内だけです。

- MQCNO_FASTPATH_BINDING では、アプリケーションは、キュー・マネージャーを作成した MQM 管理ユーザー・グループの一部であるユーザー ID の下で実行される必要があります。
- Guardian パラメーター MQCONNECTTYPE は、Options フィールドで指定されたバインド・タイプに関連して使用して、使用するバインディングのタイプを制御できます。このパラメーターが指定されている場合、それには値 **FASTPATH** または **STANDARD** があるはずですが、他の値があっても、それは無視されます。このパラメーターの値は大文字小文字を区別します。
- MQSeries for Compaq NSK は MCNO_VERSION_2 を MQCNO_VERSION_1 と同様にサポートしていますが、ClientConnOffset および ClientConnPtr フィールドは無視されます。

MQGMO - Get メッセージ・オプション

MQGMO 構造は、MGET 呼び出しの入出力パラメーターです。MQSeries for Compaq NSK 内の MQGMO_SET_SIGNAL、MQGMO_WAIT、MQGMO_SYNCPOINT、および MQGMO_NO_SYNCPOINT オプションに関する以下の情報に注意してください。

- メッセージの到着を待機している間にアプリケーションが他の作業に進めるようにしたい場合は、シグナル・オプション MQGMO_SET_SIGNAL を、MQGMO_WAIT の代わりに使用することを検討してください。ただし、このシグナル・オプションは環境固有なので、複数の異なる環境に移植されるアプリケーションには使用しないでください。
- 複数の MQGET 呼び出しが同一のメッセージを待機していて、待ちオプションとシグナル・オプションとが混在している場合、待機している各呼び出しは同等に評価されます。MQGMO_SET_SIGNAL を MQGMO_WAIT と共に指定するとエラーになります。さらに、このオプションを、シグナルが未解決のキュー・ハンドルと共に指定してもエラーになります。
- アプリケーションが MQGET を MQGMO_SET_SIGNAL と共に指定して、WaitInterval が 0 である場合、MQGMO_SET_SIGNAL オプションは無視されて、MQGMO_NO_WAIT が指定された MQGET と同様に処理されます。これは、WaitInterval が 0 になる可能性がある場合、アプリケーションは MQGMO_SET_SIGNAL を指定した MQGET で MQRC_NO_MSG_AVAILABLE を受け取る準備ができていなければならないことを意味します。次の場合には、アプリケーションはシグナル IPC を受け取ります。
 - アプリケーションが MQGET からの MQRC_SIGNAL_REQUEST_ACCEPTED を受け取る場合 (シグナルが posted であることを示している)

- アプリケーションが `file_open` システム・メッセージを処理することが可能で、シグナル送達についてのキュー・サーバーのタイムアウト期限内にシグナル `IPC` を受け取る場合。このタイムアウト値はデフォルトでは 60 秒ですが、キュー・サーバーの環境内に `MQQSIGTIMEOUT PARAM` を指定することによってキュー・サーバーについて変更できます。

タイムアウト期限が満了する前にプロセスを開いて `IPC` を送信できない場合、キュー・マネージャーは、`IPC` メッセージのアプリケーションへの送達の失敗をログに記録します。この時点では、キュー・マネージャーは送達を再試行しません。アプリケーションは柔軟に対応して、無期限にシグナル `IPC` を待機しないようにする必要があります。

- `MQGMO_SYNCPOINT_IF_PERSISTENT` がサポートされるようになりました。
- オプション `MQGMO_SYNCPOINT` または `MQGMO_NO_SYNCPOINT` のいずれも設定されていない場合、MQSeries for Compaq NSK はデフォルトで `MQGMO_SYNCPOINT` になります。
- MQSeries for Compaq NSK は `MsgToken` フィールドをサポートしません。

MQMD - メッセージ記述子

MQMD 構造には、メッセージを説明する制御情報が含まれています。以下の情報に注意してください。

- *MQSeries* アプリケーション・プログラミング・リファレンス で説明されている `BackoutCount` 機能。これはメッセージが作業単位の一部として `MQGET` によって戻された後にバックアウトした回数のカウントです。これは、アプリケーションが、メッセージ内容に基づいて処理エラーを検出するための補助機能として提供されています。バージョン 2.2.0.1 では、`BackoutCount` が計算されていました。
- MQSeries for Compaq NSK では、メッセージの廃棄 (および必要であればレポートの生成) は `MQGET` 呼び出しの際には実行されませんが、キュー・マネージャーの設定値に応じて定期的に行われるキュー・マネージャーによって制御されています。
- `UserIdentifier` フィールドの値は、`MQPUT` または `MQPUT1` の際にキュー・マネージャーによって設定された場合、アプリケーションの有効なユーザー ID に対応するキュー・マネージャーのプリンシパル・データベース内にある *MQSeries* プリンシパル名です。

MQPMO - Put メッセージ・オプション

MQPMO 構造は、MQPUT および MQPUT1 呼び出しの入出力パラメーターです。MQSeries for Compaq NSK 内での MQPMO_NO_SYNCPOINT オプションに関する以下の情報に注意してください。

- オプション MQPMO_SYNCPOINT または MQPMO_NO_SYNCPOINT のいずれも設定されていない場合、MQSeries for Compaq NSK はデフォルトで MQPMO_SYNCPOINT になります。

MQI 呼び出し

この節では、次の MQI 呼び出しの変更点について説明します。

MQI 呼び出しの記述	V2.x でサポート	V5.1 でサポート	MQSeries アプリケーション・プログラミング・リファレンス での説明どおり機能するか
MQBACK - 変更のバックアウト	エラーを戻す ⁴³² ページの1	エラーを戻す ⁴³² ページの1	はい
MQBEGIN - 作業単位の開始	いいえ	エラーを戻す ⁴³² ページの1	はい
MQCLOSE - オブジェクトのクローズ	はい	はい	はい、ただし付加的な注あり。詳細は、432ページの『MQCLOSE - オブジェクトのクローズ』を参照。
MQCMIT - 変更のコミット	いいえ	エラーを戻す ⁴³² ページの1	はい
MQCONN - キュー・マネージャーの接続	はい	はい	はい
MQCONNX - キュー・マネージャーの接続 (拡張)	いいえ	はい	はい
MQDISC - キュー・マネージャーの切断	はい	はい	はい、ただし付加的な注あり。詳細は、433ページの『MQDISC - キュー・マネージャーの切断』を参照。
MQGET - メッセージの取得	はい	はい	はい

MQI 呼び出し

MQI 呼び出しの記述	V2.x でサポート	V5.1 でサポート	MQSeries アプリケーション・プログラミング・リファレンス での説明どおり機能するか
MQINQ - オブジェクト属性に関する問い合わせ	はい	はい	はい、ただし付加的な注あり。詳細は、433ページの『MQINQ - オブジェクト属性に関する問い合わせ』を参照。
MQOPEN - オブジェクトを開く	はい	はい	はい、ただし付加的な注あり。詳細は、433ページの『MQOPEN - オブジェクトを開く』を参照。
MQPUT - メッセージの書き込み	はい	はい	はい
MQPUT1 - 1 つのメッセージの書き込み	はい	はい	はい
MQSET - オブジェクト属性の設定	はい	はい	はい、ただし付加的な注あり。詳細は、434ページの『MQSET- オブジェクト属性の設定』を参照。
MQSYNC - 統計更新の同期化	エラーを戻す ²	エラーを戻す ²	はい

注:

1. MQI 呼び出しはアプリケーションから発行できますが、完了コード MQCC_FAILED および理由コード MQRC_ENVIRONMENT_ERROR を必ず戻します。
2. この呼び出しは、CompCode の MQCC_OK、および理由コードの MQRC_NONE を必ず戻します。

MQCLOSE - オブジェクトのクローズ

MQOPEN 呼び出しの逆である MQCLOSE 呼び出しは、オブジェクトへのアクセスを解放します。

MQSeries for Compaq NSK 上で、MQGMO_SET_SIGNAL オプションが指定された MQGET 要求が、クローズされたキュー・ハンドルに対して未解決となっている場合、その要求は取り消されます。複数の異なるハンドル内に存在する

同一のキューに対するシグナル要求 (*Hobj*) は、影響を受けません (削除されるのが動的キューである場合は例外で、それらも取り消されます)。

動的キューをオープンまたはクローズしている FASTPATH アプリケーションでは、MQSeries は、監査済みデータベースを更新するために、TM/MP トランザクションを開始および終了する場合があります。アプリケーションが TM/MP T ファイルをオープンしている場合 (複数のトランザクションを開始できるので)、ENDTRANSACTION は待機なしの操作となり、アプリケーションは MQSeries によって開始されたトランザクションの完了要求を受け取ります。アプリケーションの設計を検討してこれが当てはまるかどうかを調べ、論理ハンドリング完了が MQSeries によって実行された ENDTRANSACTION 完了に対応できることを検証してください。

MQDISC - キュー・マネージャーの切断

MQCONN の逆である MQDISC 呼び出しは、MQSeries キュー・マネージャーとアプリケーション・プログラムとの間の接続を切断します。

MQSeries アプリケーション・プログラミング・リファレンス にある使用上の注の 3 は誤りです。MQSeries for Compaq NSK 上では、キュー・マネージャーが調整する作業単位が、MQDISC の呼び出し時に進行中である場合、暗黙的な同期点は発生しません。これは、NSK キュー・マネージャーが UOW を調整できないためです。調整は TM/MP サブシステムによって行われます。

MQINQ - オブジェクト属性に関する問い合わせ

MQPUT 呼び出しは、オブジェクトの属性を含む、整数の配列および文字ストリングのセットを戻します。

MQOPEN - オブジェクトを開く

MQOPEN 呼び出しは、オブジェクトへのアクセスを確立します。MQSeries for Compaq NSK 上では、キュー・マネージャーの *MaxHandles* 属性は無視されます。

動的キューをオープンまたはクローズしている FASTPATH アプリケーションでは、MQSeries は、監査済みデータベースを更新するために、TM/MP トランザクションを開始および終了する場合があります。アプリケーションが TM/MP T ファイルをオープンしている場合 (複数のトランザクションを開始できるので)、ENDTRANSACTION は待機なしの操作となり、アプリケーションは MQSeries によって開始されたトランザクションの完了要求を受け取ります。アプリケーションの設計を検討してこれが当てはまるかどうかを調べ、論

MQI 呼び出し

理ハンドリング完了が MQSeries によって実行された ENDTRANSACTION 完了に対応できることを検証してください。

MQSET- オブジェクト属性の設定

MQSET 呼び出しは、ハンドルによって表されるオブジェクトの属性を変更します。そのオブジェクトはキューでなければなりません。MQSeries for Compaq NSK 上では、MQIA_DIST_LISTS セレクターがサポートされます。

MQSET を使用してオブジェクトの属性を変更する FASTPATH アプリケーションでは、MQSeries は監査済みデータベースを更新するために、TM/MP トランザクションを開始および終了します。アプリケーションが TM/MP T ファイルをオープンしている場合 (複数のトランザクションを開始できるので)、ENDTRANSACTION は待機なしの操作となり、アプリケーションは MQSeries によって開始されたトランザクションの完了要求を受け取ります。アプリケーションの設計を検討してこれが当てはまるかどうかを調べ、論理ハンドリング完了が MQSeries によって実行された ENDTRANSACTION 完了に対応できることを検証してください。

MQSeries の属性

MQSeries for Compaq NSK では、すべてのオブジェクトの属性は *MQSeries* アプリケーション・プログラミング・リファレンス で説明されているとおりです。ただし、以下の例外および追加事項があります。

すべてのキューの属性

MQSeries for Compaq NSK では、すべてのキューの属性は *MQSeries* アプリケーション・プログラミング・リファレンス で説明されているとおりです。ただし、以下の例外および追加事項があります。

AlterationDate および AlterationTime 属性が更新されるのは、オブジェクトの属性に管理変更が加えられるときだけです。CurrentQDepth、OpenInputCount、および OpenOutputCount 属性は、動的にのみ変更できます。QDepthHighCount、QDepthLowEvent、QDepthMaxEvent、および QServiceIntervalEvent は動的にも管理的にも変更できますが、管理変更 (MQSC コマンドを使用したり MQSET を介して行う変更) だけが AlterationDate および AlterationTimes 属性を変更します。

ローカルおよびモデル・キューの属性

MQSeries for Compaq NSK では、

- *Archive* 属性は無視されます。

- バックアウト・カウントがディスクに保存されないため、*HardenGetBackout* 属性は無視されます。メッセージをアーカイブする機能はありません。
- 持続メッセージでは、*BackoutCount* 属性は必ず固定になります。非持続メッセージでは、*BackoutCount* 属性は決して固定になりません。しかし、ローカル・キューで、その *-q server C* オプション属性が設定されている場合、*BackoutCount* にはチェックポイントが設定されて、キュー・サーバーをバックアップするようになります。この方法でチェックポイントが設定されるメッセージは、キュー・サーバーの障害に柔軟に対応できます。他の MQSeries プラットフォームとの互換性を保持するために、MQINQ 呼び出しによって MQIA_HARDEN_GET_BACKOUT セレクターを使用して、属性を照会することができます。

キュー・マネージャーの属性

- *MaxMsgLength* は 100 MB です。
- *CommandLevel* は MQCMDL_LEVEL_510 です。
- *SyncPoint* は MQSP_AVAILABLE です。
- *CodedCharSetId* の値は、キュー・マネージャーのインスタンスの作成時に指定されたものです。
- *MaxHandles* 属性は無視されます。MQSeries for Compaq NSK のオープン・ハンドルの最大数を指定することはできません。この最大値は、システム・リソース制約によって決められます。
- *MaxUncommittedMsgs* 属性は無視されます。単一の作業単位で許可されるメッセージの最大数を指定することはできません。この最大値は、リソース制約によって決められます。
- *CCSID* は変更可能です。

データ変換

MQSeries for Compaq NSK V5.1 上のすべての出口をサポートする方法について説明している、459ページの『付録L. ユーザー出口』を参照してください。このメカニズムは前のバージョンから変更されていて、より一貫性があり移植可能な出口のインプリメンテーションがサポートされています。

付録I. アプリケーションの構築および実行

サブボリューム ZMQSSMPL で提供される、サンプル・プログラムおよびサンプル・コンパイルとバインディング・スクリプトは、MQSeries for Compaq NSK における MQI の主な機能を例示し、アプリケーションのコンパイルおよびバインド方法を示します。

アプリケーションの作成

この節では、*MQSeries アプリケーション・プログラミング・ガイド* で説明されている、標準バージョン 5.1 の MQI インターフェースと、MQSeries for Compaq NSK 用のインターフェース間の小さな相違点についての更新情報を記載しています。この節の情報によって、MQSeries for Compaq NSK V5.1 用の *MQSeries アプリケーション・プログラミング・ガイド* は更新されます。

MQGET 待機インターバルとチャネル DISCONT および HBINT の使用

MQGMO_NO_SYNCPOINT を使用した MQGET の実行時に、TM/MP トランザクションは、検索基準を満たす持続メッセージが使用可能な場合のみ、MQSeries によって開始および終了します。同期点でない操作には、WaitIntervalの値に対してアプリケーションによる考慮事項は必要ありません。

MQGMO_SYNCPOINT または MQGMO_SYNCPOINT_IF_PERSISTENT オプションを指定して発行される MQGET の場合、TM/MP トランザクションは、BEGINTRANSACTION を発行するユーザー・アプリケーションの制御下にあります。待機インターバルは TMF 自動打ち切りタイムアウト値を超えてはならず、理想的には TM/MP 監査証跡の有効な量のピンを避けられるように小さい値であるべきです。使用頻度が低いキューまたは夜間アイドル状態のキューに対して、待機を制限なしで指定すると、メッセージがキュー上で使用可能になったときに、自動打ち切りタイムアウトが超過し、MQRC_UOW_CANCELLED (2297) が MQGET に戻されることがあります。待機インターバルの値を高くするか、または制限なしを使用すると、TMF 監査証跡がピンされて、結果的に (訂正されない場合に) TM/MP サブシステムは、システム規模のベースで、トランザクションが使用不可になる場合があります。

アプリケーションの作成

同様に、チャンネルを送信するための DISCONT 値および HBINT 値も、TM/MP トランザクションの長さを制御します。チャンネルは、アイドル時にトランザクションを循環し、TM/MP 監査証跡に有害な影響を与えずに長い切断インターバルを許可します。

パラメーター MQTRANSACTIONLIFE は、チャンネル切断インターバルおよびゼロであるハートビート用の、TM/MP トランザクションのリフレッシュを制御するために使用できます。これは、TM/MP トランザクションの存続期間をさらに長く、または短くする必要がある場合、またはアイドル・チャンネルが生成する TM/MP アクティビティーの量を変更する場合に役立ちます。値を上げると量は減り、低い設定だと量が増えます。

各 MQSeries MQS-ECxx PATHWAY サーバー・クラスに次のように追加します。

```
PARAM MQTRANSACTIONLIFE <number>
```

ここで、<number> は 100 などの数値です。

このパラメーターは、10 秒のデフォルト TM/MP トランザクション・リフレッシュ・インターバルの使用を指定変更します。たとえば、パラメーターが指定されていない、切断インターバルの制限がないチャンネルは、メッセージの着信を待機する間、約 10 秒ごとにリフレッシュします。

作業単位 (トランザクション) 管理

トランザクション管理は、MQSeries そのものではなく、Compaq TM/MP 製品の制御下で実行されます。

これらの相違点の影響は次のとおりです。

- MQPUT および MQGET 呼び出し用の SYNCPOINT オプションは、NO_SYNCPOINT ではなく、SYNCPOINT です。
- MQPUT、MQGET、および MQPUT1 操作にデフォルトの (SYNCPOINT) オプションを使用するには、アプリケーションに、作業単位のコミットを定義するアクティブな TM/MP トランザクションがなければなりません。アプリケーションは、BEGINTRANSACTION() 関数を呼び出すことによって、TM/MP トランザクションを開始します。このトランザクションがアクティブである間にアプリケーションによって実行されるすべての MQPUT、MQPUT1、および MQGET 操作は、同一作業単位 (トランザクション) 内にあります。アプリケーションによって実行される他のデータベース操作もこ

の作業単位内にあります。単一の TM/MP トランザクション内で作成および削除できるメッセージの数とサイズには、システムが課す制限があることに注意してください。アプリケーションが作業単位を完了すると、TM/MP トランザクションは、ENDTRANSACTION() 関数を使用して終了します (作業単位がコミットされる)。何らかのエラーが発生する場合、アプリケーションは、ABORTTRANSACTION() 関数を使用して TM/MP トランザクションを取り消す (作業単位をバックアウトする) ことができます。したがって、標準バージョン 5 の関数 MQCMIT()、MQBACK()、および MQBEGIN() はこの製品ではサポートされません。これらの関数が呼び出されると、エラーが戻されます。

- アプリケーションが MQPUT、MQGET、および MQPUT1 操作に NO_SYNCPOINT オプションを使用する場合、MQSeries は、アプリケーションに戻る前に、TM/MP トランザクションそのものを開始し、キュー操作を実行し、トランザクションをコミットします。したがって、各操作はその固有の作業単位で実行され、一度完了すると、TM/MP を使用してアプリケーションによってバックアウトすることはできません。
- メッセージが持続性のもので、そのために TM/MP 保護のキュー・ファイルを更新する必要がある場合にのみ、MQSeries は、NO_SYNCPOINT 操作用の TM/MP トランザクションそのものを開始することが必要です。
- TM/MP トランザクションは、MQGET、MQPUT、および MQPUT1 以外の MQI 呼び出しの場合には、アクティブである必要はありません。
- MQRC_SYNCPOINT_LIMIT_REACHED 理由コードは、MQSeries for Compaq NSK によって、単一の TM/MP トランザクション内の入出力操作の数についてシステムが課した制限に達したことをアプリケーションに通知するために使用されます。アプリケーションが SYNCPOINT オプションを指定した場合、トランザクションを取り消し (作業単位をバックアウトし)、その作業単位でさらに少ない数の操作で再試行する必要があります。
- MQRC_UOW_CANCELED 理由コードは、作業単位 (TM/MP トランザクション) が、システムそのもの (TM/MP はこの原因となるシステム全体のリソース使用しきい値を課す)、ユーザー処置、またはトランザクションそのものの開始プログラムのいずれかによって、取り消されたことをアプリケーションに通知します。
- MQRC_BACKED_OUT 理由コードは、アプリケーションに、MQSeries が、エラー、または 1 次キュー・サーバーの障害が原因で作業単位の取り消しを強制されたことを通知します。アプリケーションは、ABORTTRANSACTION を呼び出し (操作が同期点であった場合)、再試行する必要があります。
- MQRC_SYNCPOINT_NOT_AVAILABLE 理由コードは、アプリケーションに、MQSeries が、操作を完了するために、必要な TM/MP トランザクシ

アプリケーションの作成

ンを開始または使用できなかったことを通知します。通常これは TM/MP の問題を示し、エラー・ログ内、またはキュー・マネージャーが生成する FFST から追加情報を入手できる場合があります。

一般設計の考慮事項

次の点に注意してください。

- MQI ライブラリー (アプリケーション・プロセスにバインドされる) は、\$RECEIVE、および \$TMP (TM/MP トランザクション疑似ファイル) そのものをオープンしないので、アプリケーションをコーディングして、これらの機能を使用することができます。
- MQI ライブラリーは、キュー・マネージャーを使って、初期通信で SERVERCLASS_SEND_() 呼び出しを使用します。これは接続時に、最低 2 つのプロセス・ファイル・オープン (LINKMON プロセス、およびローカル・キュー・マネージャー・エージェントを使用)、および少数 (10 未満) のディスク・ファイル・オープンを保守します。プロセス・オープンは、MQOPEN 呼び出しの結果として、入力、出力、またはブラウザ用にオープンされるローカル・キューをサポートするキュー・サーバーに対しても行われます。
- MQCONN の呼び出し前には、必ず未解決の待機なし PATHSEND 入出力がないようにする必要があります。MQCONN は、待機なし PATHSEND 入出力を実行し、エラーの原因であるアプリケーションの未解決の入出力の完了を代行受信することができます。
- FASTPATH バインディングのアプリケーションには、マルチスレッド TM/MP リクエスト (MQOPEN、MQCLOSE、および MQSET の前述の説明を参照) でもある場合、特別な考慮事項があります。

XA インターフェース

作業単位 (UOW) 調整用の XA インターフェースは提供されません。すべての UOW 調整は、TM/MP によって実行されます。

MQGMO_LOCK を指定した MQGMO_BROWSE_*

MQGMO_LOCK を指定した MQGMO_BROWSE_* がサポートされるようになりました。MQSeries アプリケーション・プログラミング・リファレンス を参照してください。

起動されるアプリケーション

Compaq NSK 環境で起動される MQSeries アプリケーションは、実行中の TACL プロセスで設定される環境変数を介してユーザー・データを受け取ります。これは、Compaq C プロセスに渡せる引き数リストの長さ制限があるためです。

この情報にアクセスするため、起動されるアプリケーションには、次のようなコードが含まれている必要があります (詳細については `amqsinqa` を参照)。

アプリケーションの作成

```
MQTMC2 *trig;                               /* trigger message structure */
MQTMC2 trigdata;                             /* trigger message structure */
char    *applId;
char    *envData;
char    *usrData;
char    *qmName;

/*****
/*
/* Set the program argument into the trigger message
/*
/*
/*****
trig = (MQTMC2*)argv[1];          /* -> trigger message */

/* get the environment variables and load the rest of the trigger */
memcpy(&trigdata, trig, sizeof(trigdata));

memset(trigdata.ApplId, ' ', sizeof(trigdata.ApplId));
memset(trigdata.EnvData, ' ', sizeof(trigdata.EnvData));
memset(trigdata.UserData, ' ', sizeof(trigdata.UserData));
memset(trigdata.QMgrName, ' ', sizeof(trigdata.QMgrName));

if ( (applId = getenv("TRIGAPPLID")) != 0)
{
    strncpy( trigdata.ApplId ,applId, strlen(applId) );
}

if ( (envData = getenv("TRIGENVDATA")) != 0)
{
    strncpy( trigdata.EnvData , envData, strlen(envData) );
}

if ( (usrData = getenv("TRIGUSERDATA")) != 0)
{
    strncpy( trigdata.UserData, usrData, strlen(usrData) );
}

if ( (qmName = getenv("TRIGQMGRNAME")) != 0)
{
    strncpy( trigdata.QMgrName, qmName, strlen(qmName) );
}

trig = &trigdata;
```

サポートされる言語および環境

MQSeries for Compaq NSK V5.1 は、表24で説明している言語と環境をサポートしています。この表は、アプリケーションが FASTPATH または STANDARD バインディングを使用できるかどうかを示しています。

表 24. サポートされる言語および環境の要約

言語	Guardian での実行 ¹	OSS での実行	STANDARD バインディングの使用	FASTPATH バインディングの使用 ¹
C ネイティブ	可能	可能	可能	可能
C 非ネイティブ	可能	不可	可能	不可
COBOL ネイティブ	可能	可能	可能	可能
COBOL 非ネイティブ	可能	不可	可能	不可
C++ ネイティブ	可能	可能	可能	可能
TAL 非ネイティブ	可能	不可	可能	不可
NonStop Java ²	不可	可能	可能	不可

注:

- Guardian 環境および FASTPATH バインディングの OSS アプリケーションでは、スレッドを使用できません。OSS STANDARD バインディングおよび Java のみがスレッドを使用できます。ご使用のアプリケーションでのスレッドの使用の詳細については、444ページの『スレッドを使用したアプリケーションの作成における考慮事項』を参照してください。
- NonStop Java アプリケーションは、トランザクションに Java トランザクション・サービス (JTS) を使用します。
- FASTPATH バインディングを使用するネイティブ・アプリケーションは、MQSeries for Compaq NSK の共用リソース・ライブラリーを介してのみ MQI を解決します。STANDARD バインディングを使用するネイティブ・アプリケーションは、MQSeries for Compaq NSK の共用リソース・ライブラリーまたは静的なネイティブ MQI ライブラリーのどちらかを使って、MQI を解決できます。FASTPATH および STANDARD バインディングの詳細については、445ページの『FASTPATH バインディングと STANDARD バインディングとの比較』を参照してください。

スレッドを使用したアプリケーションの作成における考慮事項

Guardian アプリケーションはスレッドをサポートしません。それらは独自の連携スレッド化メカニズムをインプリメントできますが、それを使用するプロセスについては、Guardian 環境から MQI を使用するための規則に従う必要があります。

OSS 環境では、POSIX スレッドに基づくスレッド・エミュレーション・パッケージが使用できます。このエミュレーションは、他のスレッド中のコードが実行可能になる前に、スレッドが実行制御をやめる必要のある連携スケジューリング・メカニズムをインプリメントします。アプリケーションはこのスレッド化パッケージを使用してプロセスをスレッドに編成できますが、次の制約事項が適用されます。

- MQI はスレッド間では連携スケジューリングをサポートしません。これは、MQI がアプリケーション・スレッドから呼び出されるときに、どれだけ時間がかかるかに関係なく、他のスレッドは実行制御を取得することはできないという意味です。たとえば、スレッドが「永久に待機」オプションを指定して MQGET を呼び出す場合、MQGET が戻るまで、他のスレッドはそのアプリケーション・プロセスで実行することはできません。
- OSS 環境で実行している FASTPATH バインディングのアプリケーションは、スレッドを使用できません。
- キュー・マネージャーは、マルチスレッドのローカル・キュー・マネージャー・エージェント (LQMA または MQLQMAG プロセス)、またはメッセージ・チャンネル・エージェント (MCA) をサポートしません。

アプリケーションのコンパイルおよびバインド

MQSeries for Compaq NSK MQI は、Compaq ワイド・メモリー・モデル (int データ・タイプは 4 バイト) および共通ランタイム環境 (CRE) を使用してインプリメントされます。アプリケーションが正しく作動するためには、この環境との互換性がなければなりません。互換性を確実にするための各コンパイラーごとの正しいオプションについては、構築ファイルのサンプルを参照してください。

特に、TAL および COBOL アプリケーションは、CRE 関連の Compaq 資料で説明されている、CRE との互換に必要な規則に従う必要があります。

MQGET の MQGMO_SET_SIGNAL 関数を正しく使用するために、COBOL および TAL アプリケーション用のオブジェクト・コードで、HIGHREQUESTERS 属性を ON に設定することが必要です。

インストール時には、ZMQSLIB に入っている MQI ライブラリーの 3 つのバージョンが、MQSeries for Compaq NSK を使用して送達されます。表25で示されるとおり、必ず正しいライブラリーを使用してください。

表 25. MQI ライブラリーの正しいバージョンの使用

プログラム言語	非ネイティブ	ネイティブ / 静的	ネイティブ / 動的
TAL	MQMLIB	適用しない	適用しない
COBOL	MQMLIB	MQMLIBN	MQSRLLIB
C	MQMLIB	MQMLIBN	MQSRLLIB
C++	適用しない	MQMLIBN	MQSRLLIB

FASTPATH バインディングと STANDARD バインディングとの比較

MQSeries for Compaq NSK V5.1 は、FASTPATH バインディングおよび STANDARD バインディングの両方をサポートします。表25 は、バインディングの各タイプをサポートする言語および環境を説明しています。

STANDARD バインディング

アプリケーションで STANDARD バインディングを使用する場合、次の点を考慮してください。

- 非ネイティブ、ネイティブ・アプリケーションは両方とも STANDARD バインディングを使用できます。
- STANDARD バインディングを使用するネイティブ・アプリケーションは、次のいずれかを使用して MQI を解決できます。
 - MQSeries for Compaq NSK の共用リソース・ライブラリー
 - 静的なネイティブ MQI ライブラリー。これは、すでに共用リソース・ライブラリーを使用するアプリケーションをサポートします。

FASTPATH バインディング

アプリケーションで FASTPATH バインディングを使用する場合、次の点を考慮してください。

- FASTPATH バインディングを使用できるのはネイティブ・アプリケーションだけです。
- FASTPATH バインディングを使用するネイティブ・アプリケーションは、MQSeries for Compaq NSK の共用リソース・ライブラリーを介してのみ MQI を解決できます。
- OSS で実行している FASTPATH バインディングのアプリケーションは、スレッドを使用できません。

アプリケーションのコンパイル

- FASTPATH バインディングのアプリケーションは、キュー・マネージャーを作成した Compaq NSK MQM 管理ユーザー・グループで、ユーザー ID の下で実行する必要があります。

アプリケーションの実行

キュー・マネージャーに接続するためには、アプリケーション・プログラムの環境を、以下のように正しく定義する必要があります。

- PARAM MQDEFAULTPREFIX は、すべてのアプリケーションの環境で必須です。
- MQSINI ファイルに代替 (デフォルトでない) 位置を選択した場合、PARAM MQMACHINIFILE が正しく設定されていないと、アプリケーションはキュー・マネージャーに接続できません。
- TAL および COBOL アプリケーションは、それぞれの環境で PARAM SAVE-ENVIRONMENT ON を定義する必要があります。そうしないとキュー・マネージャーには接続できません。

アプリケーションは、低 PIN でも高 PIN でも実行できます。MQSeries 実行可能モジュール自体は、高 PIN として実行するように構成されています。

MQSeries アプリケーションは、Guardian 環境および OSS 環境の両方でサポートされます。

MQSeries アプリケーションは、PATHWAY の下、TACL から、または別のプロセスの子プロセスとして実行できます。アプリケーションは、キュー・マネージャーのシャットダウンで正しく動作すれば、キュー・マネージャー PATHWAY 構成そのものに追加することもできます。

付録J. MQSeries 管理インターフェース (MQSeries Administration Interface (MQAI))

MQSeries for Compaq NSK V5.1 は、MQAI インターフェースをサポートします。

MQAI は、C 言語を使用する MQSeries のプログラミング・インターフェースです。データ・バッグを使用して、MQSeries キュー・マネージャーで管理タスクを実行します。データ・バッグを使用すると、プログラム式コマンド形式 (PCF) を使用するよりも簡単な方法で、オブジェクトのプロパティ（またはパラメーター）を処理することができます。

MQAI では、MQGET および MQPUT 呼び出しを使用するよりも簡単に PCF の操作を行えます。MQAI は、次のことを実行するために使用できます。

- 自己管理アプリケーションおよびアプリケーション・ツールを実装する。
- PCF メッセージの使用を単純化する。MQAI は、MQSeries を管理する簡単な方法です。独自の PCF メッセージを書き込む必要がないので、複雑なデータ構造に関連する問題を回避することができます。
- エラー条件をさらに簡単に処理する。MQSeries コマンド (MQSC) から戻りコードを戻すのは難しいですが、MQAI によってエラー条件の処理が簡単になります。

注: MQSeries for Compaq NSK V5.1 は、C ヘッダー・ファイルのみを提供します。Visual Basic ヘッダー・ファイルは提供しません。

MQAI の詳細については、*MQSeries 管理インターフェースのプログラミングの手引きおよび解説書* を参照してください。

付録K. MQSeries for Compaq NSK サンプル・プログラム

次の C および COBOL のサンプル・プログラムが、MQSeries for Compaq NSK V5.1 で提供されています。

説明	C ソース	C 実行可能モジュール	COBOL85 ソース	COBOL85 実行可能モジュール
キュー上の各メッセージごとに、メッセージ記述子およびコンテキストを読み取って出力します。	amqsbcg0	amqsbcg	サンプルなし	サンプルなし
メッセージを、メッセージ・キューから応答先キューにエコーします。	amqsecha	amqsech	amq0ech0	amq0ech
メッセージをキューから stdout に書き込みます。メッセージはキューに残ります (ブラウザ)。	amqsgbr0	amqsgbr	amq0gbr0	amq0gbr
指定したキューからメッセージを除去し、stdout に書き込みます。	amqsget0	amqsget	amq0get0	amq0get
起動されたキューを読み取り、キュー情報で応答します。	amqsinqa	amqsinq	サンプルなし	サンプルなし
共用入力キューを使用します。	サンプルなし	サンプルなし	amq0inq0	amq0inq
stdin をメッセージにコピーし、指定されたキューにメッセージを書き込みます。	amqsput0	amqsput	amq0put0	amq0put
要求メッセージを指定されたキューに入れ、応答を表示します。	amqsreq0	amqsreq	amq0req0	amq0req
(トリガー機能) 指定したキューへの書き込みを禁止し、応答として結果のステートメントを戻します。	amqsseta	amqsset	amq0set0	amq0set
トリガー・モニター	amqstrg0	amqstrg	サンプルなし	サンプルなし
データ変換出口用のサンプル・スケルトン	amqsvfcn	サンプルなし	サンプルなし	サンプルなし

サンプル・プログラム

説明	C ソース	C 実行可能モジュール	COBOL85 ソース	COBOL85 実行可能モジュール
チャンネル出口用のサンプル・スケルトン	amqsvchn	サンプルなし	サンプルなし	サンプルなし
クラスター・ワークロード出口用のサンプル・スケルトン	amqswlm0	サンプルなし	サンプルなし	サンプルなし
MQLOADEXIT 用のサンプル・スケルトン	amqslxp0	サンプルなし	サンプルなし	サンプルなし

次の TAL サンプル・プログラムが MQSeries for Compaq NSK V5.1 で提供されています。

説明	TAL ソース	TAL 実行可能モジュール
キューから n メッセージを読み取ります。	zmqreadt	zmqread
長さ n の n メッセージをキューに書き込みます。	zmqwritt	zmqwrit

C サンプル・プログラムの作成

非ネイティブ (非ネイティブ静的ライブラリー MQMLIB の使用)

サブボリューム ZMQSSMPL には、非ネイティブのサンプル C アプリケーションを作成するために使用される、以下の TACL マクロ・ファイルが含まれます。

CSAMP

使用法: CSAMP *source-code-file-name*

これは、サブボリューム ZMQSLIB に含まれる組み込みファイルを使用して、C ソース・ファイルをコンパイルするための基本マクロです。たとえば、サンプル AMQSBCG0 をコンパイルするには、CSAMP AMQSBCG0 を使用します。コンパイルが正常に実行されると、マクロは、たとえば、AMQSBCG0 というように、ファイル名の最後の文字を O で置き換えたオブジェクト・ファイルを作成します。

BSAMP

使用法: BSAMP *exe-file-name*

これは、ZMQSLIB 中のユーザー・ライブラリー MQMLIB を使って、オブジェクト・ファイルをバインドするために使用する基本マクロです。たとえば、コンパイル済みのサンプル AMQSBCG0 をバインドするには、BSAMP AMQSBCG を使用します。このマクロは、*exe-file-nameE* という名前の実行可能ファイル、たとえば AMQSBCGE を作成します。

COMPALL

使用法: COMPALL

この TACL マクロは、CSAMP マクロを使用して、各サンプル・ソース・コード・ファイルをコンパイルします。

BINDALL

使用法: BINDALL

この TACL マクロは、BSAMP マクロを使用して、各サンプル・オブジェクト・ファイルを実行可能モジュールにバインドします。

BUILDC

使用法: BUILDC

この TACL マクロは、マクロ COMPALL と BINDALL を使用して、すべての C サンプル・ファイルをバインドします。

ネイティブ (ネイティブ静的ライブラリー MQMLIBN の使用)

ネイティブ・インストールの場合、以下の TACL マクロ・ファイルを使用してサンプル MQI アプリケーションを作成します。

NMCCALL

使用法: NMCCALL

NMCSAMP を使用してすべてのサンプルをネイティブにコンパイルするマクロです。

NMCSAMP

使用法: NMCSAMP *source-code-file-name*

これは、サブボリューム ZMQSLIB に含まれる組み込みファイルを使用して、C ソース・ファイルをコンパイルするための基本マクロです。たとえば、サンプル AMQSBCG0 をコンパイルするには、NMCSAMP AMQSBCG0 を使用します。コンパイルが正常に実行されると、マクロは、たとえば、AMQSBCG0 というように、

C サンプルの作成

ファイル名の最後の文字を O で置き換えたオブジェクト・ファイルを作成します。

NMLDSAMP

使用法: `NMLDSAMP exe-file-name`

この基本マクロは、オブジェクト・ファイルを、ZMQSLIB の静的なネイティブ MQI ライブラリー MQMLIBN とリンクさせます。

NMLDALL

使用法: `NMLDALL`

この TACL マクロは、NMLDSAMP マクロを使用して、各サンプル・オブジェクト・ファイルを実行可能モジュールにバインドします。

NMBUILD C

使用法: `NMBUILD C`

この TACL マクロは、マクロ NMCALL と NMLDALL を使用して、すべてのネイティブ C サンプル・ファイルをバインドします。

ネイティブ (SRL MQSRLLIB の使用)

NMLDSSMP

使用法: `NMLDSSMP exe-file-name`

この基本マクロは、オブジェクト・ファイルを、ZMQSLIB のネイティブ MQ SRL MQSRLLIB とリンクさせます。

NMLDSALL

使用法: `NMLDSALL`

この TACL マクロは、NMLDSSMP マクロを使用して、各サンプル・オブジェクト・ファイルを実行可能モジュールにバインドします。

NMBULDSC

使用法: `NMBULDSC`

この TACL マクロは、マクロ NMCALL と NMLDSALL を使用して、すべてのネイティブ C サンプル・ファイルをバインドします。

NMLDPSRL

使用法: `NMLDPSRL exe-file-name`

この基本マクロは、オブジェクト・ファイルを、ZMQSLIB の MQSeries プライベート SRL とリンクさせます。

NMCPSRL

使用法: `NMCPSRL source-code-file-name`

MQSeries PSRL に含めるためにユーザー・コードをコンパイルするマクロです。

NMLDUSRL

使用法: NMLDUSRL *object-input-file*。ここで、*object-input-file* はリンクされるオブジェクトのリストを含むファイルです。

これは、ユーザー・コードを再リンク可能なライブラリーにリンクするための基本マクロです。

注: 非ネイティブ・アプリケーションは、ネイティブ・キュー・マネージャーに接続でき、その逆も可能です。ネイティブと非ネイティブ操作のすべての組み合わせは有効で、サポートされます。

C++ サンプル・プログラムの作成

ネイティブ (ネイティブ静的ライブラリー MQMLIBN の使用)

NMCPALL

使用法: NMCPALL

NMCCPP を使用してすべてのサンプルをネイティブにコンパイルするマクロです。

NMCCPP

使用法: NMCCPP *source-code-file-name*

これは、サブボリューム ZMQSLIB に含まれる組み込みファイルを使用して、C++ ソース・ファイルをコンパイルするための基本マクロです。たとえば、サンプル IMQSGETP をコンパイルするには、NMCCPP IMQSGETP を使用します。コンパイルが正常に実行されると、マクロは、たとえば、IMQSGETO というように、ファイル名の最後の文字を O で置き換えたオブジェクト・ファイルを作成します。

NMLDCPP

使用法: NMLDCPP *exe-file-name*

この基本マクロは、オブジェクト・ファイルを、ZMQSLIB の静的なネイティブ MQI ライブラリー MQMLIBN とリンクさせます。

NMLDCPPA

使用法: NMLDCPPA

この TAACL マクロは、NMLDSAMP マクロを使用して、各サンプル・オブジェクト・ファイルを実行可能モジュールにバインドします。

C++ サンプルの作成

NMBLDCPP

使用法: NMBUILD

この TACL マクロは、マクロ NMCPPALL と NMLDCPPA を使用して、すべてのネイティブ C++ サンプル・ファイルをバインドします。

ネイティブ (SRL MQSRLLIB の使用)

NMLDCPPS

使用法: NMLDCPPS *exe-file-name*

この基本マクロは、オブジェクト・ファイルを、ZMQSLIB のネイティブ MQ SRL MQSRLLIB とリンクさせます。

NMLDCPSA

使用法: NMLDCPSA

この TACL マクロは、NMLDCPPS マクロを使用して、各サンプル・オブジェクト・ファイルを実行可能モジュールにバインドします。

NMBLDSCP

使用法: NMBLDSCP

この TACL マクロは、マクロ NMCPPALL と NMLDCPSA を使用して、すべてのネイティブ C サンプル・ファイルをバインドします。

COBOL サンプル・プログラムの作成

非ネイティブ (非ネイティブ静的ライブラリー MQMLIB の使用)

サブボリューム ZMQSSMPL には、サンプル COBOL アプリケーションを作成するために使用される、以下のファイルが含まれます。

COBSAMP

使用法: COBSAMP *source-code-file-name*

これは、サブボリューム ZMQSLIB に含まれる定義ファイルを使用して、COBOL ソース・ファイルをコンパイルするための基本マクロです。たとえば、プログラム AMQ0GBR0 をコンパイルするには、COBSAMP AMQ0GBR0 を使用します。コンパイルが正常に実行されると、マクロは、たとえば、AMQ0GBR0 というように、ファイル名の最後の文字を O で置き換えたオブジェクト・ファイルを作成します。

BCOBSAMP

使用法: BCOBSAMP *exe-file-name*

これは、ZMQSLIB 中のユーザー・ライブラリーを使って、オブジェクトをバインドするために使用する基本マクロです。たとえば、コンパイル済みのサンプル AMQ0GBRO をバインドするには、BCOBSAMP AMQ0GBR を使用します。このマクロは、*exe-file-name* AMQ0GBR という名前の実行可能モジュールを作成します。

CCBSMPLS

使用法: CCBSMPLS

この TACL マクロは、各 COBOL サンプル・ソース・コード・ファイルをコンパイルします。

BCBSMPLS

使用法: BIND /IN BCBSMPLS/

このバインド入力ファイルは、各 COBOL サンプル・オブジェクト・ファイルを実行可能モジュールにバインドします。

BUILDCOB

使用法: BUILDCOB

この TACL マクロは、マクロ CCBSMPLS と BCBSMPLS を使用して、すべての COBOL サンプル・ファイルをバインドします。

ネイティブ (ネイティブ静的ライブラリー MQMLIBN の使用)

NMCOBSMP

使用法: NMCOBSMP *source-code-file-name*

これは、ZMQSLIB の MQSeries ネイティブ・ライブラリー MQMLIBN を使用して、ネイティブ・モード COBOL 'NMCOBOL' をコンパイルするためのマクロです。たとえば、プログラム AMQ0GBRO をコンパイルするには、NMCOBSMP AMQ0GBR0 を使用します。コンパイルが正常に実行されると、マクロは、たとえば、AMQ0GBRO というように、ファイル名の最後の文字を 0 で置き換えたオブジェクト・ファイルを作成します。

NMLDCOB

使用法: NMLDCOB *exe-file-name*

このマクロは、オブジェクトを、ZMQSLIB の MQSeries ネイティブ・ライブラリー MQMLIBN とバインドします。たとえば、コ

ンパイル済みのサンプル AMQ0GBRO をバインドするには、NMLDCOB AMQ0GBR を使用します。このマクロは、*exe-file-name* AMQ0GBR という名前の実行可能モジュールを作成します。

NMCOBALL

使用法: NMCOBALL

この TACL マクロは、NMCOBSMP を使用して、各 COBOL サンプル・ソース・コード・ファイルをコンパイルします。

NMLDACOB

使用法: NMLDACOB

このバインド入力ファイルは、各 NMLDCOB サンプル・オブジェクト・ファイルを実行可能モジュールにバインドします。

NMBLDCOB

使用法: NMBLDCOB

この TACL マクロは、マクロ NMCOBALL と NMLDACOB を使用して、すべての COBOL サンプル・ファイルをバインドします。

ネイティブ (SRL MQSRLLIB の使用)

NMCBSSMP

使用法: NMCBSSMP *source-code-file-name*

これは、ZMQSLIB の MQSeries SRL MQSRLLIB を使用して、ネイティブ・モード COBOL 'NMCOBOL' をコンパイルするためのマクロです。たとえば、プログラム AMQ0GBRO をコンパイルするには、NMCBSSMP AMQ0GBR0 を使用します。コンパイルが正常に実行されると、マクロは、たとえば、AMQ0GBRO というように、ファイル名の最後の文字を 0 で置き換えたオブジェクト・ファイルを作成します。

NMLDSCOB

使用法: NMLDSCOB *exe-file-name*

このマクロは、オブジェクトを、ZMQSLIB の MQSeries SRL MQSRLLIB とバインドします。たとえば、コンパイル済みのサンプル AMQ0GBRO をバインドするには、NMLDSCOB

AMQ0GBR を使用します。このマクロは、*exe-file-name* AMQ0GBR という名前の実行可能モジュールを作成します。

NMCBSALL

使用法: NMCBSALL

この TAACL マクロは、NMCBSAMP を使用して、各 COBOL サンプル・ソース・コード・ファイルをコンパイルします。

NMLDSCOB

使用法: NMLDSCOB

このバインド入力ファイルは、各 NMLDSCOB サンプル・オブジェクト・ファイルを実行可能モジュールにバインドします。

NMBLDSCB

使用法: NMBLDSCB

この TAACL マクロは、マクロ NMCBSALL と NMLDSCOB を使用して、すべての COBOL サンプル・ファイルをバインドします。

TAL サンプル・プログラムの作成

非ネイティブ (非ネイティブ静的ライブラリー MQMLIB の使用)

サブボリューム ZMQSSMPL には、サンプル TAL プログラムを作成するために使用される、以下のファイルが含まれます。

TALSAMP

使用法: TALSAMP *source-code-file-name*。これは、サブボリュームに含まれる定義ファイルを使用して、TAL ソース・ファイルをコンパイルするための基本マクロです。たとえば、プログラム ZMQWRITT をコンパイルするには、TALSAMP ZMQWRITT を使用します。コンパイルが正常に実行されると、マクロは、たとえば、ZMQWRITO というように、ファイル名の最後の文字を O で置き換えたオブジェクト・ファイルを作成します。

BTALSAMP

使用法: BTALSAMP *exe-file-name*

これは、ZMQSLIB 中のユーザー・ライブラリーを使って、オブジェクトをバインドするために使用する基本マクロです。たとえば、コンパ

TAL サンプルの作成

イル済みのサンプル ZMQWRITO をバインドするには、BTALSAMP ZMQWRIT を使用します。

CTLSMPLS

使用法: CTLSMPLS

この TACL マクロは、各 TAL サンプル・ソース・コード・ファイルをコンパイルします。

BTLSMPLS

使用法: BIND /IN BTLSMPLS/

このバインド入力ファイルは、各 TAL サンプル・オブジェクト・ファイルを実行可能モジュールにバインドします。

BUILDTAL

使用法: BUILDTAL

この TACL マクロは、マクロ CTLSMPLS と BTLSMPLS を使用して、すべての TAL サンプル・ファイルをバインドします。

OSS でのサンプル・プログラムの作成 (ネイティブ・モード専用)

ディレクトリー - /opt/mqm/samp には、MQSeries 用のサンプル・プログラムと、Make ファイル MQMAKE.SMP が入っています。この Make ファイルには、すべての C、C++、および NMCOBOL サンプルを作成するために必要な、すべてのターゲットが含まれています。

ディレクトリー - /opt/mqm/inc には、OSS でプログラムを作成するために必要な、すべての copylibs およびヘッダー・ファイルが入っています。

注: MQSeries SRL ファイルおよびネイティブ MQI ライブラリー -- MQMLIBN は、Guardian ファイル・システムにのみ存在します。必要であれば、作成スクリプトおよび Make ファイルを、それらのライブラリーを指すように編集する必要があります。たとえば、/G/system/zmqslib/mqsrllib とします。

付録L. ユーザー出口

MQSeries for Compaq NSK V5.1 は、チャンネル出口プログラム、データ変換出口プログラム、およびクラスター・ワークロード管理 (CLWL) 出口プログラムをサポートします。さらに、Compaq NSK に固有のロード・プログラム出口もサポートされます。チャンネル出口の詳細については、*MQSeries 相互通信* を参照してください。データ変換出口の詳細については、*MQSeries アプリケーション・プログラミング・ガイド* および *MQSeries アプリケーション・プログラミング・リファレンス* を参照してください。クラスター・ワークロード管理出口の詳細については、*MQSeries キュー・マネージャー・クラスター* を参照してください。

この付録では、MQSeries for Compaq NSK での出口プログラムの使用に固有の情報を提供します。

サポートされるユーザー出口

表26 は、MQSeries for Compaq NSK でサポートされるユーザー出口の各タイプの特徴をリストします。

表 26. *MQSeries for Compaq NSK* でサポートされるユーザー出口

ユーザー出口のタイプ	出口名の最大長	出口データの最大長	使用可能な対象	チェーン済み
チャンネル MSG 出口	32	32	DEFINE CHANNEL	はい
チャンネル SEND 出口	32	32	DEFINE CHANNEL	はい
チャンネル RECEIVE 出口	32	32	DEFINE CHANNEL	1es
チャンネル SECURITY 出口	32	32	DEFINE CHANNEL	
チャンネル MSGRETRY 出口	32	32	DEFINE CHANNEL	
チャンネル自動定義出口	32	32	ALTER QMGR	
クラスター・ワークロード管理出口	32	32	ALTER QMGR	

サポートされる出口プログラム

表 26. MQSeries for Compaq NSK でサポートされるユーザー出口 (続き)

ユーザー出口のタイプ	出口名の最大長	出口データの最大長	使用可能な対象	チェーン済み
データ変換出口	8	適用しない	不明な FORMAT 名	
MQ_LOAD_ENTRY_POINT_EXIT	固定名	適用しない	上記の出口のいずれかが必要な場合または使用可能な場合に呼び出される	

MQ_LOAD_ENTRY_POINT_EXIT は、Compaq NSK に固有の、唯一のユーザー出口です。

出口名の形式

出口名 (MQ_LOAD_ENTRY_POINT_EXIT を除く) は、最大で 32 文字の長さの、英数字から成るストリングです。関連データ・フィールドをサポートする出口の場合、データは最大で 32 文字の長さのストリングです。

MQ_LOAD_ENTRY_POINT_EXIT - ユーザー出口のロード

すべてのユーザー出口プログラムは、MQSeries プライベート SRL、または静的ライブラリーにリンクしなければなりません。ユーザー出口プログラムには、必要なときに MQSeries によって呼び出せる、外部関数 (記号) が最低 1 つ含まれているべきです。

他の MQSeries ユーザー出口を使用可能にする前に、MQ_LOAD_ENTRY_POINT_EXIT プログラムをインストールして、出口名をエントリー・ポイント・アドレスにマップする必要があります。

MQ_LOAD_ENTRY_POINT_EXIT プログラムは MQSeries SRL または静的ライブラリーにリンクする必要があり、他のユーザー出口のうち 1 つが使用可能になる場合は必ず MQSeries によって呼び出されます。

MQ_LOAD_ENTRY_POINT_EXIT プログラムの名前は固定されています。つまり、その外部関数名は MQ_LOAD_ENTRY_POINT のままでなければなりません。

MQ_LOAD_ENTRY_POINT_EXIT は、自由に出口名をエントリー・ポイントにマップしたり、多数の出口名を同じエントリー・ポイント・アドレスにマップしたりすることができます。

MQSeries は、呼び出されると必ず MQXCC_FAILED (出口名が見つからなかった) を戻す、スタブ MQ_LOAD_ENTRY_POINT_EXIT 関数を提供します。このスタブ出口は、他のユーザー出口を使用可能にする前に、独自の出口で置換する必要があります。

MQ_LOAD_ENTRY_POINT_EXIT (*ExitParms*)

パラメーター:

```
Exitparms (PMQLXP) -- input/output
    LoadExit Parameter Block
```

使用上の注意: MQ_LOAD_ENTRY_POINT_EXIT によって実行できる機能は、出口のプロバイダーによって定義されます。

463ページの図46 には、3 つの出口名をエントリー・ポイント・アドレスにマップする、サンプルの MQ_LOAD_ENTRY_POINT_EXIT があります。

MQLXP - MQ_LOAD_ENTRY_POINT_EXIT パラメーター構造

MQLXP 構造は、ロード出口に渡される情報を説明します。

この構造は、Compaq NSK だけでサポートされます。

フィールド

Struclد (MQCHAR4)

構造 ID。

値は MQLXP_STRUC_ID です。

ロード出口パラメーター構造の ID。

C プログラミング言語の場合、定数 MQLXP_STRUC_ID_ARRAY も定義されます。これには MQLXP_STRUC_ID と同じ値がありますが、ストリングではなく文字の配列です。

これは、出口への入力フィールドです。

Version (MQLONG)

構造バージョン番号。

サポートされる出口プログラム

値は `QLXP_VERSION_1` です。

バージョン -1 ロード出口パラメーター構造。

以下の定数 `QLXP_CURRENT_VERSION` は、現行バージョンのバージョン番号を指定します。

ロード出口パラメーター構造の現行バージョン。

これは、出口への入力フィールドです。

QMgrName (MQCHAR48)

ローカル・キュー・マネージャーの名前。

これは、ロード出口を呼び出したキュー・マネージャーの名前です。名前には、フィールドの長さを満たすまでブランクが埋め込まれます。

これは、出口への入力フィールドです。

EntryPointName (MQCHAR32)

要求されたエントリー・ポイントの名前。

これは、ロード出口が呼び出し可能アドレスに解決する必要があるエントリー・ポイントの名前です。名前には、フィールドの長さを満たすまでブランクが埋め込まれます。

これは、出口への入力フィールドです。

EntryAddress (PMQFUNC)

戻される呼び出し可能なアドレス。

これは要求される `EntryPoint` のアドレスです。

これは出口からの出力フィールドです。

ExitResponse (MQLONG)

出口からの応答。

これは、エントリー名の呼び出し可能アドレスへの解決が正常に行われたかどうかを示すため、出口によって設定されます。以下のうちいずれかでなければなりません。

```
MQXCC_OK  
Success.
```

これは、出口の処理が、`ExitParm` で提供される `EntryPointName` を、呼び出し可能アドレスに正常に解決したことを示します。呼び出し可能アドレスは、`QLXP` 構造で `EntryAddress` フィールドに戻されます。

```
MQXCC_FAILED  
Failed.
```

これは、出口が、ExitParm で提供される EntryPointName を、呼び出し可能アドレスに正常に解決できなかったことを示します。

ExitResponse フィールドで戻される他の値は、MQXCC_FAILED と同じ意味です。

これは出口からの出力フィールドです。

MQ_LOAD_ENTRY_POINT_EXIT の例

図46 は、3 つの出口名 (2 つのチャネル出口と 1 つのデータ変換出口) を、エントリー・ポイント・アドレスにマップする、作業中の

MQ_LOAD_ENTRY_POINT_EXIT プログラムの例です。

MQ_LOAD_ENTRY_POINT_EXIT サンプル・プログラムのソース・コードは、サンプル・サブボリューム (AMQSLXP0) で提供されます。

```

/*****/
/*
/* Program name: AMQSLXP0           (Compaq NSK only)
/*
/* Description: Sample C skeleton of a Load Exit function
/*
/* Statement:      Licensed Materials - Property of IBM
/*
/*                (C) Copyright IBM Corp. 1993, 2001
/*
/*****/
/*
/* Function:
/*
/* AMQSLXP0 is a sample C skeleton of a Load Exit function
/*
/* The function resolves EntryNames to callable addresses
/*
/*
/* Once complete the code should be compiled into a loadable
/* object, the name of the object should be the name of the
/* format to be converted. Instructions on how to do this are
/* contained in the README file in this directory.
/*
/*****/
/*
/* AMQSLXP0 takes the parameters defined for a Load Exit
/* routine in the CMQXC.H header file.
/*
/*****/

```

図 46. サンプル MQLOADEXIT (1/2)

サポートされる出口プログラム

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <limits.h>

#include <cmqc.h>
#include <cmqxc.h>

/*****
/* Load Exit
/*
/*
/*
*****/
void
MQENTRY MQ_LOAD_ENTRY_POINT(
    PMQLXP pExitParms          /* exit Parameter */
)
{
    /* No loadable entry points are defined */

    pExitParms->ExitResponse = MQXCC_FAILED;

    return;
}
/*****
/* End of AMQSLXP0
*****/
```

図 46. サンプル *MQLOADEXIT* (2/2)

上記の *MQ_LOAD_ENTRY_POINT_EXIT* プログラムと、チャンネルおよびデータ変換出口をインストールすると、以下の MQSC コマンドを使用してチャンネルが出口の送受信を行えるようになります。

```
ALTER CHANNEL(CHAN) CHLTYPE(SDR) SENDEXIT(MY_CHANNEL_SEND_EXIT)
ALTER CHANNEL(CHAN) CHLTYPE(SDR) RCVEEXIT(MY_CHANNEL_RCV_EXIT)
```

データ変換出口は、MQGET が変換を使用可能にして実行され (MQGMO_CONVERT)、メッセージ形式名が MY_FORMAT である場合に、MQSeries によって呼び出されます。

ユーザー出口のインストール

作成されるすべてのユーザー出口は、MQSeries プライベート SRL にインストールされる必要があります。さらに、出口タイプによっては、アプリケーション・プログラムが使用する MQSeries 静的ライブラリーにインストールすることが必要になります。

ユーザー出口のタイプ	インストール先
チャンネル MSG 出口	プライベート SRL
チャンネル SEND 出口	プライベート SRL
チャンネル RECEIVE 出口	プライベート SRL
チャンネル SECURITY 出口	プライベート SRL
チャンネル MSGRETRY 出口	プライベート SRL
チャンネル自動定義出口	プライベート SRL
クラスター・ワークロード管理出口	プライベート SRL
データ変換出口	プライベート SRL (アプリケーションが使用する場合、ネイティブ静的ライブラリーおよび非ネイティブ静的ライブラリーの両方またはどちらか一方)
MQ_LOAD_ENTRY_POINT_EXIT	プライベート SRL (アプリケーションが使用する場合、ネイティブ静的ライブラリーおよび非ネイティブ静的ライブラリーの両方またはどちらか一方)

MQSeries プライベート SRL への出口のインストール

MQSeries プライベート SRL に出口プログラムをインストールするには、新しい出口を含む MQSeries プライベート SRL の新しいバージョンを以下のように作成します。

1. 出口関数をコンパイルします。たとえば、次のようにします。

```
NMCPSSL AMQSVFCN
```

2. MQLOAD エントリー・ポイント関数をコンパイルします。たとえば、次のようにします。

```
NMCPSSL AMQSLXP0
```

サポートされる出口プログラム

3. 出口およびエンタリー・ポイント・オブジェクトを、SRL で使用するために再リンク可能なライブラリーにリンクします。たとえば、次のようにします。

```
NMLDUSRL OBJECTS EXITS
```

4. このデータ変換オブジェクトを、MQSeries プライベート SRL の再リンク可能なバージョンにリンクして、NEWMQSRL と呼ばれる MQSeries プライベート SRL の新しいバージョンを作成します。たとえば、次のようにします。

```
MAKEPSRL EXITS $VOL.ZMQSLIB NEWMQSRL
```

5. 現行の MQSeries プライベート SRL にアクセスしている、すべてのキュー・マネージャーおよびアプリケーションを停止します。
6. すべての MQSeries アプリケーションを新しい PSRL に再リンクします。たとえば、次のようにします。

```
NMLDEXES $VOL.ZMQSLIB.NEWMQSRL $VOL.ZMQSEXE
```

7. get アプリケーションをコンパイルします。たとえば、次のようにします。

```
NMCSAMP AMQSGET0
```

8. すべてのユーザー・アプリケーションを新しい PSRL に再リンクします。たとえば、次のようにします。

```
AMQSGET NMLDPSRL AMQSGET
```

9. MQSeries およびすべての MQI アプリケーションを再始動します。

ステップ 466ページの6 と 466ページの8 は、かなり短時間で実行できますが、新しい MQSeries PSRL が ZMQSLIB に入れられて、MQSRLLIB と呼ばれる場合は、省略することも可能です。別の MQSeries PSRL にリンクするために、これらのステップを繰り返すことができます。

MQSeries ネイティブ静的ライブラリーへの出口のインストール

出口は、TACL マクロ NMLDEXIT を使用して、選択したアプリケーションおよび MQI ライブラリーとリンクさせることができます。たとえば、次のようにします。

```
NMLDEXIT Object-File Exit-Object-File
```

MQSeries 非ネイティブ静的ライブラリーへの出口のインストール

出口は、TACL マクロ BEXITE を使用して、選択した実行可能モジュール（またはライブラリー）にバインドできます。

注: このプロシージャは、ターゲット実行可能モジュールを変更します。マクロを使って、ターゲット実行可能モジュールまたはライブラリーのバックアップ・コピーを作成することをお勧めします。

出口関数は、一度コンパイルされると、MQSeries がアクセス可能なターゲット実行可能モジュールまたはライブラリーに、直接バインドされる必要があります。TACL マクロ BEXITE はこの目的で使用されます。たとえば、次のようにします。

```
BEXITE Target-Executable-Or-Library Source-Exit-File-Or-Library
```

たとえば、サンプルのデータ変換出口をサンプル MQSGETA にバインドするには、以下のステップを実行します。

1. 出口関数をコンパイルします。たとえば、次のようにします。

```
CSAMP AMQSVFCN
```

サポートされる出口プログラム

2. MQLOAD エントリー・ポイント関数をコンパイルします。たとえば、次のようにします。

```
CSAMP AMQSLXP0
```

3. get アプリケーションをコンパイルします。たとえば、次のようにします。

```
CSAMP AMQSGET0
```

4. get アプリケーションをバインドします。たとえば、次のようにします。

```
BSAMP AMQSGET
```

5. 出口関数を get アプリケーションにバインドします。たとえば、次のようにします。

```
BEXITE AMQSGET AMQSVFC0
```

6. エントリー・ポイント関数を get アプリケーションにバインドします。たとえば、次のようにします。

```
BEXITE AMQSGET AMQSLXP0
```

別の方法として、すべてのアプリケーションがこのデータ変換出口を持つことになる場合、以下のステップによって、出口がバインドされるユーザー・ライブラリー、およびアプリケーションの両方が作成されます。

1. 出口関数をコンパイルします。たとえば、次のようにします。

```
CSAMP AMQSVFCN
```

2. MQLOAD エントリー・ポイント関数をコンパイルします。たとえば、次のようにします。

```
CSAMP AMQSLXP0
```

3. get アプリケーションをコンパイルします。たとえば、次のようにします。

```
CSAMP AMQSGET0
```

4. 出口関数をユーザー・ライブラリーにバインドします。たとえば、次のようにします。

```
BEXITE ZMQSLIB.MQMLIB AMQSVFC0
```

5. エントリーポイントをユーザー・ライブラリーにバインドします。たとえば、次のようにします。

```
BEXITE ZMQSLIB.MQMLIB AMQSLXP0
```

6. get アプリケーションを変更済みライブラリーにバインドします。たとえば、次のようにします。

```
BSAMP AMQSGET
```

サポートされる出口プログラム

付録M. 通信のセットアップ

この付録では、SNA および TCP/IP 通信プロトコルを使用する MQSeries for Compaq NSK の通信をセットアップする方法を説明します。以下の例を記載しています。

- 480ページの『SNAX 通信の例』
- 487ページの『ICE 通信の例』
- 491ページの『TCP/IP 通信の例』

SNA チャンネル

MQSeries for Compaq NSK V5.1 での SNA チャンネル用の次のチャンネル属性が必要です。

CONNAME

CONNAME の値は、通信プロトコルとして SNAX または ICE のどちらが使用されるかによって異なります。

SNAX が使用される場合:

CONNAME('\$PPPP.LOCALLU.REMOTELU')

送信側、リクエスター、および完全修飾されたサーバー・チャンネルに適用されます。説明を以下に記します。

\$PPPP SNAX/APC プロセスのプロセス名です。

LOCALLU ローカル LU の名前です。

REMOTELU リモート・マシン上のパートナー LU の名前です。

たとえば、次のようにします。

```
CONNAME('$BP01.IYAHT080.IYCNVM03')
```

ICE が使用される場合:

CONNAME('\$PPPP.#OPEN.LOCALLU.REMOTELU')

送信側、リクエスター、および完全修飾されたサーバー・チャンネルに適用されます。説明を以下に記します。

\$PPPP ICE プロセスのプロセス名です。

#OPEN ICE オープン名です。

通信のセットアップ

LOCALLU ローカル LU の名前です。
REMO TELU リモート・マシン上のパートナー LU の名前です。

たとえば、次のようにします。

```
CONNNAME('$ICE.#IYAHT0C.IYAHT0C0.IYCNVM03')+
```

MODENAME

SNA モード名です。たとえば、MODENAME(LU62PS) となります。

TPNAME('LOCALTP[.REMO TELTP]')

トランザクション処理 (TP) 名です。

LOCALTP SNA 通信で使用される PATHWAY でのサーバー・クラス (通常は MQSeries) の名前です。SNAX の場合、サーバー・クラスは、SNAX ディスパッチャーおよび APC プロセスのサーバー・クラスと同じ PATHWAY になければなりません。ICE の場合、これは NOF - ADD TP コマンドで宣言された PATHWAY になければなりません。

REMO TELTP リモート・マシン上の TP の名前です。この値はオプションです。これが指定されず、チャンネルが会話を開始するもの (つまり送信側、リクエスター、または完全修飾されたサーバー・チャンネル) である場合は、LOCALTP 名が使用されます。

LOCALTP および REMO TELTP の値は、最大で 16 文字にすることができます。

LU 6.2 応答側プロセス

MQSeries for Compaq NSK V5.1 では、リモートのキュー・マネージャー・チャンネルからの着信接続要求を listen するために、SNA リスナー・プロセスが必要です。

MQSeries SNA リスナーは、リモート・チャンネルから着信 Attach が到着したときは、SNA 製品によってディスパッチされなければなりません。

SNAX APC Dispatcher を使用することにより、SNAX はパートナー・トランザクション・プログラムからの着信 Attach 要求を受信することができます。これを行うためには、SNAX ディスパッチャーには次のことが必要です。

- SNAX Dispatcher がキュー・マネージャーとは別の PATHWAY で実行されていること。
- APC プロセスが、それと同じ PATHWAY からサーバー・クラスとして実行されていること。

Insession の ICE を使用するには、次のことが必要です。

- トランザクション・プログラム (サーバー・クラス) が PATHWAY で定義されていること。
- トランザクション・プログラムが、そのサーバー・クラスを指す NOF に追加されていること。

その他に別の ICE ディスパッチャーはないため、ICE プロセス自体が着信 Attach 要求を処理します。

SNAX TP ディスパッチング

Compaq SNAX SNA 製品は、APPC Attach がパートナー・トランザクション・プログラムから到着したときの、APPC トランザクション・プログラム (TP) の始動をサポートしています。SNAX Dispatcher は、それらの要求を、ローカル・トランザクション・プログラム・サーバー・クラスを使用して、関連した SNAX \$APC プロセスにディスパッチします。

MQSeries 用の SNAX Dispatcher の要件は、次のとおりです。

- SNAX Dispatcher は、関連した \$APC プロセスと同じ PATHWAY で実行する必要があります。
- 着信 TPNAME ごとに、Dispatcher および \$APC プロセスと同じ PATHWAY のサーバー・クラス (通常は指定された MQSERIES) として定義されなければなりません。
- サーバー・クラス・プログラム名は、MQSeries 実行可能サブボリューム (通常は ZMQSEX) に存在している `runmqslr` プログラムです。
- ディスパッチャー・プロセスはサーバー・クラスを開始して、DISPATCH-TP IPC 要求で、関係のある情報 (\$APC Process、LUName、TPName) をすべてそのサーバー・クラス (TP) に渡します。

ICE TP ディスパッチング

ICE バージョン 3.2 では、その SNA Attach Manager を、その TP が PATHWAY サーバー・クラスである SNAX に同様にインプリメントします。ICE プロセスは Attach 要求を受信し、それ自体がディスパッチャーです。ただし、この ICE プロセスは、TP (サーバー・クラス) と同じ PATHWAY で実行する必要はありません。この場合の環境の要件は次のとおりです。

- アクティブな ICE プロセスが実行されていなければなりません。

通信のセットアップ

- Dispatch の TP が NOF に追加される必要があります。たとえば、次のようにします。

```
ADD TP <tpname>, PROCESS <$process>, SERVERCLASS <Serverclass name>
```

Serverclass name は、通常は MQSeries です。

- 着信 TPNAME ごとに、PATHWAY<\$process> のサーバー・クラス (通常は指定された MQSERIES) として定義されなければなりません。
- サーバー・クラス・プログラム名は、MQSeries 実行可能サブボリューム (通常は ZMQSEXE) に存在している RUNMQLSR プログラムです。
- ICE プロセスはサーバー・クラスを開始して、DISPATCH-TP IPC 要求で、関係のある情報 (\$ICE Process、ApplName、TPName) をすべてそのサーバー・クラス (TP) に渡します。

SNA 環境のセットアップ例

実際の SNA 環境を構成する方法の例を以下に示します。

SNAX APC を使用する: SNAX APC を使用する場合:

- PATHWAY を作成して対象のリスナーが排他的に使用できるようにする必要があります。
- APC プロセスのサーバー・クラスは、その PATHWAY から実行される必要があります。

TACL プロンプトで次のように入力します。

```
TACL> Pathmon /name $PMAP, nowait, out $vhs, cpu 3/4
TACL> Pathcom $PMAP
= 0 LU62SCFG
```

ここで LU62SCFG は、以下を含む編集ファイルです。

```
[ SET PATHMON BACKUPCPU 6
SET PATHWAY MAXTCPS 10
SET PATHWAY MAXTERMS 10
SET PATHWAY MAXPROGRAMS 10
SET PATHWAY MAXSERVERCLASSES 10
SET PATHWAY MAXSERVERPROCESSES 10
SET PATHWAY MAXSTARTUPS 10
SET PATHWAY MAXPATHCOMS 40
SET PATHWAY MAXASSIGNS 32
SET PATHWAY MAXPARAMS 32
START PATHWAY COLD!

SET TCP PROGRAM $ SYSTEM.SYSTEM.PATHTCP2
SET TCP CPUS 3:4
SET TCP MAXTERMS 5
SET TCP MAXSERVERCLASSES 010
SET TCP MAXSERVERPROCESSES 010
SET TCP MAXTERMDATA 08960
SET TCP MAXREPLY 20000 SET TCP NONSTOP 0
SET TCP TCLPROG $system.system.APCP
ADD TCP SNAXAPC-TCP
```

図 47. MQSeries SNAX セットアップ・ファイルのサンプル (1/3)

```
[Configure the SNAX/APC SERVER]
RESET SERVER
SET SERVER PARAM LOGFILE APCLOG
SET SERVER PARAM TRACEFILE APCTRC
SET SERVER PARAM BACKUPCPU -1
SET SERVER PARAM MAXINRUSIZE 4096
SET SERVER PARAM MAXOUTRUSIZE 4096
SET SERVER PARAM MAXAPPLIOSIZE 4096
SET SERVER PARAM DATAPAGES 100
SET SERVER PARAM TRACEPAGES 300
SET SERVER PARAM RMTATTACHDISP QUEUE
SET SERVER PARAM RMTATTACHTIMER -1
SET SERVER PARAM CONFIG APCCFG
SET SERVER PROGRAM $system.system.APCOBJ
SET SERVER OUT $VHS
SET SERVER HOMETERM $VHS
SET SERVER PROCESS $AP02
SET SERVER NUMSTATIC 1
SET SERVER MAXSERVERS 1
SET SERVER CREATEDELAY 0 SECS
SET SERVER DELETEDELAY 1 MINS
SET SERVER CPUS 3:4
ADD SERVER SNAXAPCSVR
```

図 47. MQSeries SNAX セットアップ・ファイルのサンプル (2/3)

通信のセットアップ

```
[Add MQSeries SNAX Listener]
RESET SERVER
SET SERVER PROGRAM $DATA00.ZMQSEXE.RUNMQLSR
SET SERVER PROCESS $lrcv
SET SERVER NUMSTATIC 1
SET SERVER MAXSERVERS 1
SET SERVER CREATEDELAY 0 SECS
SET SERVER DELETEDELAY 1 MINS
SET SERVER STARTUP "-t LU62"
SET SERVER PARAM MQQUEMGRNAME "QMGR"
SET SERVER PARAM MQMACHINIFILE "$DATA03.QMGRD.UMQSINI"
SET SERVER PARAM MQDEFAULTPREFIX "$DATA00"
SET SERVER OUT $VHS
SET SERVER HOMETERM $VHS
SET SERVER CPUS 3:4
[ADD SERVER MQSERIES]
ADD SERVER MQSERIES

START TCP *
[Configure the DISPATCHER]
SET TERM FILE $s.#displg
SET TERM INITIAL SNAXAPC-DISPATCHER
SET TERM TYPE CONVERSATIONAL
SET TERM TCP SNAXAPC-TCP
ADD TERM SNAXAPCSVR01 [First 10 chars are the SNAX/APC server name]

start server MQSERIES
start server SNAXAPCSVR
start term SNAXAPCSVR01
```

図 47. MQSeries SNAX セットアップ・ファイルのサンプル (3/3)

注: Listener サーバー・クラスは、追加の始動パラメーター SET SERVER STARTUP "-t LU62" があることを除けば、キュー・マネージャー固有の PATHWAY の MQS-TCPLIS00 サーバー・クラスと同等です。

Insession ICE を使用する

Insession ICE を使用する場合は、PATHWAY を作成して対象のリスナーが排他的に使用できるようにする必要があります。ICE プロセスはその PATHWAY からは実行されません。

1. 次のように TP を NOF に追加します。

```
ADD TP <tpname>, PROCESS <process>, SERVERCLASS <server> [, <option> ...]
```

説明

process

TP を管理する PATHMON プロセスの名前です。

server

TP が属する PATHMON SERVERCLASS の名前です。

option

次から選択します。

- [ATTACHTIMER n] -- 新しい TP スレッドのディスパッチング後、ICE が ATTACH を待つ合計時間 (100 分の 1 秒単位)。デフォルトは 6000 (60 秒) です。
- [MAXDISPATCHTHREADS n] -- 同時にディスパッチされる TP の最大数。デフォルトは 0 (同時にディスパッチされる TP の数に制限なし)。
- [TIMEOUT n] -- MAXDISPATCHTHREADS が TP に達したときの ICE の ATTACH に対する応答を決めます。TIMEOUT -1 の場合、ATTACH は不定時にキューに書き込まれます。TIMEOUT >0 の場合、ATTACH は n/100 秒でキューに書き込まれます。デフォルトは 0 です (ATTACH は即時にリジェクトされます)。

2. サーバー・クラスを PATHWAY に追加することも必要です。TACL プロンプトで次のように入力します。

```
TACL> Pathmon /name $PMAP, nowait, out $vhs, cpu 3/4
TACL> Pathcom $PMAP
= 0 LU62ICFG
```

ここで LU62ICFG は、以下を含む編集ファイルです。

通信のセットアップ

```
[
SET PATHMON BACKUPCPU 6
SET PATHWAY MAXTCPS 10
SET PATHWAY MAXTERMS 10
SET PATHWAY MAXPROGRAMS 10
SET PATHWAY MAXSERVERCLASSES 10
SET PATHWAY MAXSERVERPROCESSES 10
SET PATHWAY MAXSTARTUPS 10
SET PATHWAY MAXPATHCOMS 40
SET PATHWAY MAXASSIGNS 32
SET PATHWAY MAXPARAMS 32
START PATHWAY COLD!

SET TCP PROGRAM $SYSTEM.SYSTEM.PATHTCP2
SET TCP CPUS 3:4
SET TCP MAXTERMS 5
SET TCP MAXSERVERCLASSES 010
SET TCP MAXSERVERPROCESSES 010
SET TCP MAXTERMDATA 08960SET TCP MAXREPLY 20000
SET TCP NONSTOP 0
SET TCP TCLPROG $system.system.APCP
ADD TCP SNAXAPC-TCP
```

図48. MQSeries SNA セットアップ・ファイル (ICE 用) のサンプル (1/2)

```
[Add MQSeries ICE Listener]
RESET SERVER
SET SERVER PROGRAM $DATA00.ZMQSEXE.RUNMQLSR
SET SERVER PROCESS $lrcv
SET SERVER NUMSTATIC 1
SET SERVER MAXSERVERS 1
SET SERVER CREATEDELAY 0 SECS
SET SERVER DELETEDELAY 1 MINS
SET SERVER STARTUP "-t LU62"
SET SERVER PARAM MQQUEMGRNAME "QMGR"
SET SERVER PARAM MQMACHINIFILE "$DATA03.QMGRD.UMQSINI"
SET SERVER PARAM MQDEFAULTPREFIX "$DATA00"
SET SERVER OUT $VHS
SET SERVER HOMETERM $VHS
SET SERVER CPUS 3:4
[ADD SERVER MQSERIES]
ADD SERVER MQSERIES

START TCP *
start server MQSERIES
```

図48. MQSeries SNA セットアップ・ファイル (ICE 用) のサンプル (2/2)

注: Listener サーバー・クラスは、追加の始動パラメーター SET SERVER STARTUP "-t LU62" があることを除けば、キュー・マネージャー固有の PATHWAY の MQS-TCPLIS00 サーバー・クラスと同等です。

TCP/IP チャンネル

MQSeries for Compaq NSK には、単一の MQSeries キュー・マネージャー環境で複数の TCP/IP プロセスを使用できるオプションがあります。これを使用すると、必要な TCP/IP プロセスを指定したチャンネルに関連付けることによって、キュー・マネージャー内で使用される TCP/IP プロセスを選択できるようになります。アウトバウンド・チャンネル (送信側、サーバー、リクエスター) は、チャンネル定義の CONNAME フィールドで、必要な TCP/IP プロセス名を指定することができます。

runmqsc を使用して次のようにします。

```
alter channel ... conname ('$ZTC1.123.456.789.012(1415)')
alter channel ... conname ('$ZTC1.dnshostname(1415)')
```

MQMC パネルを使用して次のようにします。

```
TCPIP/SNA Process: $ZTC1
```

PCF コマンドを使用して次のようにします。

```
strncpy( pPCFString->String, ('$ZTC1.123.456.789.012(1415)', len );
```

デフォルト以外のリゾルバーのための DNS 解決を再構成するには、すべての PATHWAY ECnn サーバー・クラスに次を追加します。

```
DEFINE =TCPIP^RESOLVER^NAME, FILE filename
```

ここで filename はリゾルバー・ファイルのロケーションです。

ホスト・ファイルを使用する場合は、すべての PATHWAY ECnn サーバー・クラスに次を追加します。

```
DEFINE =TCPIP^HOST^FILE, FILE filename
```

通信のセットアップ

ここで filename はホスト・ファイルのロケーションです。

インバウンド・チャンネルでは、どの TCP/IP プロセスを使用するかを決めるのに環境変数を使います。TCP/IP のリスナーはこのプロセス値を、それぞれの EC に渡し、またそれらのエージェントを介してそれぞれの TCP/IP 応答側プロセスに渡します。インバウンド・チャンネルの TCP/IP プロセスを設定するには次のようにします。

TACL を使用して次のようにします。

```
ADD DEFINE =TCPIP^PROCESS^NAME, FILE processname
```

ここで processname は、TCP/IP プロセスの名前です。

MQS-TCPLIS nn サーバー・クラス用に PATHWAY を使って、次のようにします (nn はリスナー・サーバー・クラス番号)。

```
DEFINE =TCPIP^PROCESS^NAME, FILE ¥HAWK.$ZTC1  
PARAM MQLISTENPORTNUM "1415"
```

TCP/IP での通信での TCP/IP のデフォルト以外のプロセスの使用についての情報は、71ページの『デフォルト以外の TCP/IP プロセスやポートのための MQS-TCPLIS nn サーバー・クラスの再構成』を参照してください。キュー・マネージャーが listen する TCP/IP ポートについての情報は、70ページの『キュー・マネージャーが listen する TCP/IP ポート』を参照してください。

通信の例

この節には、SNA (SNAX および ICE) と TCP/IP の通信セットアップ例があります。

SNAX 通信の例

この節には次を記載しています。

- SNA 伝送路用の SCF 構成ファイルの例
- 伝送路をサポートするための幾つかの SYSGEN パラメーターの例
- SNA プロセス定義用の SCF 構成ファイルの例
- 幾つかの MQSC チャンネル定義の例

SCF SNA 伝送路構成ファイル

SCF 構成ファイルの例を以下に示します。

```

==
== SCF configuration file for defining SNA LINE, PUs and LUs to VTAM®
== Line is called $SNA02 and SYSGEN'd into the Compaq system
==

ALLOW ALL
ASSUME LINE $SNA02

ABORT, SUB LU
ABORT, SUB PU
ABORT

DELETE, SUB LU
DELETE, SUB PU
DELETE

==
== ADD $SNA02 LINE DEFINITION
==

ADD LINE $SNA02, STATION SECONDARY, MAXPUS 5, MAXLUS 1024, RECSIZE 2048, &
    CHARACTERSET ASCII, MAXLOCALLUS 256, &
    PUIDBLK %H05D, PUIDNUM %H312FB

==
== ADD REMOTE PU OBJECT, LOCAL IS IMPLICITLY DEFINED AS #ZNT21
==

ADD PU #PU2, ADDRESS 1, MAXLUS 16, RECSIZE 2046, TYPE (13,21), &
    TRRMTADDR 04400045121088, DYNAMIC ON, &
    ASSOCIATESUBDEV $CHAMB.#p2, &
    TRSSAP %H04, &
    CPNAME IYAQCDRM, SNANETID GBIBMIYA

==
== ADD LOCAL LU OBJECT
==

ADD LU #ZNTLU1, TYPE (14,21), RECSIZE 1024, &
    CHARACTERSET ASCII, PUNAME #ZNT21, SNANAME IYAHT080

==
== ADD PARTNER LU OBJECTS
==

== spinach (HP)

ADD LU #PU2LU1, TYPE(14,21), PUNAME #PU2, SNANAME IYABT0F0

== stingray (AIX)

```

通信のセットアップ

```
ADD LU #PU2LU2, TYPE(14,21), PUNAME #PU2, SNANAME IYA3T995
== coop007 (OS/2)

ADD LU #PU2LU3, TYPE(14,21), PUNAME #PU2, SNANAME IYAFT170
== MVS CICS

ADD LU #PU2LU4, TYPE(14,21), PUNAME #PU2, SNANAME IYCMVM03
== MVS Non-CICS

ADD LU #PU2LU5, TYPE(14,21), PUNAME #PU2, SNANAME IYCNVM03
== finnr100 (NT)

ADD LU #PU2LU6, TYPE(14,21), PUNAME #PU2, SNANAME IYAFT080
== winas18 (AS400)

ADD LU #PU2LU7, TYPE(14,21), PUNAME #PU2, SNANAME IYAFT110
== MQ-Portugese (OS/2)

ADD LU #PU2LU8, TYPE(14,21), PUNAME #PU2, SNANAME IYAHT090
== VSE

ADD LU #PU2LU10, TYPE(14,21), PUNAME #PU2, SNANAME IYZMZSI2

== START UP TOKEN RING ASSOCIATE SUB DEVICE $CHAMB.#P2
== then start the line, pu's and lu's

START LINE $CHAMB, SUB ALL

START
START, SUB PU

STATUS
STATUS, SUB PU
STATUS, SUB LU
```

SYSGEN パラメーター

次の例は、SNA およびトークンリング伝送路をサポートする SYSGEN 用の CONFTEXT ファイルのエントリーです。

```
!*****
!                               LAN MACRO
!*****
! This macro is used for all 361x LAN controllers
! REQUIRES T9375 SOFTWARE PACKAGE

C3613^MLAM           = MLAM
```

```

TYPE 56,          SUBTYPE 0,
PROGRAM          C9376P00,
INTERRUPT        IOP^INTERRUPT^HANDLER,
MAXREQUESTSIZE  32000,
RSIZE            32000,
BURSTSIZE        16,
LINEBUFFERSIZE  32,
STARTDOWN        #;

!*****
!                               SNAX macro for Token ring lines
!*****
TOKEN^RING^SNAX^MACRO = SNATS
TYPE 58,
SUBTYPE 4,
RSIZE 1024,
SUBTYPE 4,
FRAMESIZE 1036 # ;

!*****
!                               SNAX MANAGER
!*****
SSCP^MACRO        = SNASVM
TYPE 13,          SUBTYPE 5,
RSIZE             256 #;

!*****
!                               LAN CONTROLLER
!*****
LAN1      3616    0,1    %130    ;

!***** Service manager
SNAX      6999    0,1    %370    ;

!***** SNAX/Token Ring Pseudocontroller
RING      6997    0,1    %360    ;

!***** Token Ring Line
$CHAMB    LAN1.0, LAN1.1    C3613^MLAM, NAME #LAN1;

!***** Configure the SSCP
$SSCP     SNAX.0, SNAX.1 SSCP^MACRO;

!***** Sna lines for Dummy Controller over Token Ring
$SNA01    RING.0, RING.1 TOKEN^RING^SNAX^MACRO;
$SNA02    RING.2, RING.3 TOKEN^RING^SNAX^MACRO;

```

SNAX/APC プロセス構成

次の定義は、APC プロセス（プロセス名 \$BP01）を SCF によって SNA 伝送路用に構成する例です。

注: APC プロセス \$BP01 は、SNAX APC Dispatcher と同じ PATHWAY で実行されるサーバー・クラスのプロセスとして定義されます。

通信のセットアップ

```
==
== SCF Configuration file for SNAX/APC Lus
==

ALLOW ERRORS

ASSUME PROCESS $BP01

ABORT SESSION *
ABORT TPN *
ABORT PTNR-MODE *
ABORT PTNR-LU *
ABORT LU *

DELETE TPN *
DELETE PTNR-MODE *
DELETE PTNR-LU *
DELETE LU *

==
== ADD LOCAL LU
==
ADD LU IYAHT080, SNANAME GBIBMIYA.IYAHT080, SNAXFILENAME $SNA02.#ZNTLU1, &
    MAXSESSION 256, AUTOSTART YES

== TPnames for MQSeries

ADD TPN IYAHT080.MQSeries

=== Spinach (HP) Partner LU

ADD PTNR-LU IYAHT080.IYABT0F0, SNANAME GBIBMIYA.IYABT0F0, &
    PERIPHERAL-NODE NO, PARALLEL-SESSION-LU YES

ADD PTNR-MODE IYAHT080.IYABT0F0.LU62PS, MODENAME LU62PS, &
    DEFAULTMAXSESSION 8, DEFAULTMINCONWINNER 4, &
    DEFAULTMINCONLOSER 3, MAXAUTOACT 1, RCVWINDOW 4, &
    DEFAULTMAXINRUSIZE 1024, DEFAULTMAXOUTRUSIZE 1024, &
    SENDWINDOW 4

==
== Winas18 (AS400) Partner LU
==

ADD PTNR-LU IYAHT080.IYAFT110, SNANAME GBIBMIYA.IYAFT110, &
    PERIPHERAL-NODE NO, PARALLEL-SESSION-LU YES

ADD PTNR-MODE IYAHT080.IYAFT110.LU62PS, MODENAME LU62PS, &
    DEFAULTMAXSESSION 8, DEFAULTMINCONWINNER 4, &
    DEFAULTMINCONLOSER 3, MAXAUTOACT 1, RCVWINDOW 4, &
    DEFAULTMAXINRUSIZE 1024, DEFAULTMAXOUTRUSIZE 1024, &
    SENDWINDOW 4

==
== Stingray (AIX) Partner LU
==
```

```
ADD PTNR-LU IYAHT080.IYA3T995, SNANAME GBIBMIYA.IYA3T995, &
PERIPHERAL-NODE NO, PARALLEL-SESSION-LU YES

ADD PTNR-MODE IYAHT080.IYA3T995.LU62PS, MODENAME LU62PS, &
DEFAULTMAXSESSION 8, DEFAULTMINCONWINNER 4, &
DEFAULTMINCONLOSER 3, MAXAUTOACT 1, RCVWINDOW 4, &
DEFAULTMAXINRUSIZE 1024, DEFAULTMAXOUTRUSIZE 1024, &
SENDWINDOW 4
==
== coop007 (OS/2) Partner LU
==

ADD PTNR-LU IYAHT080.IYAFT170, SNANAME GBIBMIYA.IYAFT170, &
PERIPHERAL-NODE NO, PARALLEL-SESSION-LU YES

ADD PTNR-MODE IYAHT080.IYAFT170.LU62PS, MODENAME LU62PS, &
DEFAULTMAXSESSION 8, DEFAULTMINCONWINNER 4, &
DEFAULTMINCONLOSER 3, MAXAUTOACT 1, RCVWINDOW 4, &
DEFAULTMAXINRUSIZE 1024, DEFAULTMAXOUTRUSIZE 1024, &
SENDWINDOW 4
==
== MQ-Portugese (OS/2) Partner LU
==

ADD PTNR-LU IYAHT080.IYAHT090, SNANAME GBIBMIYA.IYAHT090, &
PERIPHERAL-NODE NO, PARALLEL-SESSION-LU YES

ADD PTNR-MODE IYAHT080.IYAHT090.LU62PS, MODENAME LU62PS, &
DEFAULTMAXSESSION 8, DEFAULTMINCONWINNER 4, &
DEFAULTMINCONLOSER 3, MAXAUTOACT 1, RCVWINDOW 4, &
DEFAULTMAXINRUSIZE 1024, DEFAULTMAXOUTRUSIZE 1024, &
SENDWINDOW 4
==
== finn100 (NT) Partner LU
==

ADD PTNR-LU IYAHT080.IYAFT080, SNANAME GBIBMIYA.IYAFT080, &
PERIPHERAL-NODE NO, PARALLEL-SESSION-LU YES

ADD PTNR-MODE IYAHT080.IYAFT080.LU62PS, MODENAME LU62PS, &
DEFAULTMAXSESSION 8, DEFAULTMINCONWINNER 4, &
DEFAULTMINCONLOSER 3, MAXAUTOACT 1, RCVWINDOW 4, &
DEFAULTMAXINRUSIZE 1024, DEFAULTMAXOUTRUSIZE 1024, &
SENDWINDOW 4
==
== MVS CICS Partner LU
==

ADD PTNR-LU IYAHT080.IYCMVM03, SNANAME GBIBMIYA.IYCMVM03, &
PERIPHERAL-NODE NO, PARALLEL-SESSION-LU YES

ADD PTNR-MODE IYAHT080.IYCMVM03.LU62PS, MODENAME LU62PS, &
DEFAULTMAXSESSION 8, DEFAULTMINCONWINNER 4, &
DEFAULTMINCONLOSER 3, MAXAUTOACT 1, RCVWINDOW 4, &
DEFAULTMAXINRUSIZE 1024, DEFAULTMAXOUTRUSIZE 1024, &
```

通信のセットアップ

```
SENDWINDOW 4
==
== MVS Non CICS Partner LU
==

ADD PTNR-LU IYAHT080.IYCNVM03, SNANAME GBIBMIYA.IYCNVM03, &
PERIPHERAL-NODE NO, PARALLEL-SESSION-LU YES

ADD PTNR-MODE IYAHT080.IYCNVM03.LU62PS, MODENAME LU62PS, &
DEFAULTMAXSESSION 8, DEFAULTMINCONWINNER 4, &
DEFAULTMINCONLOSER 3, MAXAUTOACT 1, RCVWINDOW 4, &
DEFAULTMAXINRUSIZE 1024, DEFAULTMAXOUTRUSIZE 1024, &
SENDWINDOW 4

==
== VSE Partner LU
==

ADD PTNR-LU IYAHT080.IYZMZSI2, SNANAME GBIBMIYA.IYZMZSI2, &
PERIPHERAL-NODE NO, PARALLEL-SESSION-LU YES

ADD PTNR-MODE IYAHT080.IYZMZSI2.LU62PS, MODENAME LU62PS, &
DEFAULTMAXSESSION 8, DEFAULTMINCONWINNER 4, &
DEFAULTMINCONLOSER 3, MAXAUTOACT 1, RCVWINDOW 4, &
DEFAULTMAXINRUSIZE 1024, DEFAULTMAXOUTRUSIZE 1024, &
SENDWINDOW 4

==
== Start the LUs
==

START LU IYAHT080, SUB ALL
START TPN *
```

MQSeries アプリケーションでは、MQSeries と通信サーバー・プロセスの間のプロセス間データ転送の最大サイズを制御する Maxapplio 値を、デフォルトよりも大きい 32000 に設定する必要があります。

Channel 定義

以下に、SNAX 構成をサポートする MQSeries チャンネル定義の例を幾つか示します。

- CICS を運用していない MVS/ESA 上の MQSeries への送信側チャンネル:

```
DEFINE CHANNEL(MT01.VM03.SDRC.0002) CHLTYPE(SDR) +
  TRPTYPE(LU62) +
  SEQWRAP(9999999) MAXMSGL(2048) +
  XMITQ('VM03NCM.TQ.SDRC.0001') +
  CONNAME('$BP01.IYAHT080.IYCNVM03') +
  MODENAME('LU62PS') TPNAME(MQSERIES)
```

- MVS/ESA 上の MQSeries からの受信側チャネル:

```
DEFINE CHANNEL(VM03.MT01.SDRC.0002) CHLTYPE(RCVR) +
  TRPTYPE(LU62) REPLACE DESCR('Receiver channel from VM03NCM') +
  SEQWRAP(9999999) +
  MAXMSGL(2048)
```

- 会話の開始が可能な、またはリモート・リクエスター・チャネルによって開始できる、MVS/ESA 上の MQSeries へのサーバー・チャネル:

```
DEFINE CHANNEL(MT01.VM03.RQSV.0002) CHLTYPE(SVR) +
  TRPTYPE(LU62) +
  SEQWRAP(9999999) MAXMSGL(2048) +
  XMITQ('VM03NCM.TQ.RQSV.0001') +
  CONNAME('$BP01.IYAHT080.IYCNVM03') +
  MODENAME('LU62PS') TPNAME(MQSERIES)
```

ここで、MQSeries は、MVS™ キュー・マネージャーが listen している TPNAME です。

ICE 通信の例

MQSeries 用の ICE を構成するには 2 段階あります。

1. ICE プロセス自体を構成する必要があります。
2. 伝送路 (以下の例では \$ICE01) と SNA の情報が ICE プロセスに入力される必要があります。

ICE プロセスを構成する

以下に ICE プロセス構成の例を示します。この構成は、デフォルトでは GOICE というファイルに置かれます。

```
?tacl macro
clear all
param backupcpu 1
param cinittimer 120
param collector $0
param config icectl
param idblk 05d
param idnum 312FF
param cpname IYHR00C
param datapages 64
param dynamicrlu yes
param genesis $gen
param maxrcv 32000
param loglevel info
param netname GBIBMIYA
param password xxxxxxxxxxxxxxxxxxxx
param retrys1 5
param secuserid super.super
param startup %1%
param timer1 20
param timer2 300
param usstable default
run $system.ice.ice/name $ICE,nowait,cpu 0,pri 180,highpin off/
```

注:

1. パスワード PARAM は、xxxxxxxxxxxxxxxxxxxxx で置き換えられています。
2. MQSeries アプリケーションでは、MQSeries と通信サーバー・プロセスの間のプロセス間データ転送の最大サイズを制御する maxrcv PARAM を、デフォルトよりも大きい 32000 に設定する必要があります。

伝送路および APC の情報を定義する

ICE プロセスがこの構成で始動すると、Node Operator Facility (NOF**) を使って、次の情報が ICE プロセスに入力されます。この例は、トークンリング・ポート \$CHAMB.#ICE 上で動作する、\$ICE01 という伝送路を定義しています。

```
==
== ICE definitions for PU IYHR00C.
== Local LU for this PU is IYAHT0C0.
==

ALLOW ERRORS

OPEN $ICE

ABORT LINE $ICE01, SUB ALL

DELETE LINE $ICE01, SUB ALL
```



```
==
== ADD TOKEN RING LINE
==

ADD LINE $ICE01, TNDM $CHAMB.#ICE, &
      IDBLK %H05D, &
      PROTOCOL TOKENRING, WRITEBUFFERSIZE 8192

==
== ADD PU OBJECT
==

ADD PU IYHR00C, LINE $ICE01, MULTIRROUTE YES, &
      DMAC 400045121088, DSAP %H04, &
      NETNAME GBIBMIYA, IDNUM %H312FF, IDBLK %H05D, &
      RCPNAME GBIBMIYA.IYAQCDRM, SSAP %H08

==
== Add Local APPL Object
==

DELETE APPL IYAHT0C0
ADD APPL IYAHT0C0, ALIAS IYAHT0C0, PROTOCOL CPIC, &
      OPENNAME #IYAHT0C

==
== Add Mode LU62PS
==

DELETE MODE LU62PS
ADD MODE LU62PS, MAXSESS 8, MINCONWIN 4, MINCONLOS 3

==
== Add Partner LU Objects
==

== spinach (HP)

ABORT RLU IYABT0F0
DELETE RLU IYABT0F0
ADD RLU IYABT0F0, MODE LU62PS, PARSESS YES

== stingray (AIX)

ABORT RLU IYA3T995
DELETE RLU IYA3T995
ADD RLU IYA3T995, MODE LU62PS, PARSESS YES

== coop007 (OS/2)

ABORT RLU IYAFT170
DELETE RLU IYAFT170
ADD RLU IYAFT170, MODE LU62PS, PARSESS YES
```

通信のセットアップ

```
== MVS CICS

ABORT RLU IYCMVM03
DELETE RLU IYCMVM03
ADD RLU IYCMVM03, MODE LU62PS, PARSESS YES

== MVS Non-CICS

ABORT RLU IYCNVM03
DELETE RLU IYCNVM03
ADD RLU IYCNVM03, MODE LU62PS, PARSESS YES

== finnr100 (NT)

ABORT RLU IYAFT080
DELETE RLU IYAFT080
ADD RLU IYAFT080, MODE LU62PS, PARSESS YES

== winas18 (AS400)

ABORT RLU IYAFT110
DELETE RLU IYAFT110
ADD RLU IYAFT110, MODE LU62PS, PARSESS YES

ABORT RLU IYAHT080
DELETE RLU IYAHT080
ADD RLU IYAHT080, MODE LU62PS, PARSESS YES

==
== START UP ICE LINE $ICE01 AND SUB DEVICE
==

START LINE $ICE01, SUB ALL
```

注: この構成が機能するには、ポート #ICE がトークンリング伝送路として定義されていない必要があります。

たとえば、SCF に次のコマンドを入力します。

```
add port $chamb.#ice, type tr8025, address %H08
start port $chamb.#ice
```

ここで \$chamb はトークンリング・コントローラーで、ポートの SAP は %08 です。

ICE 用のチャネル定義

この ICE 構成をサポートする MQSeries チャネル定義を幾つか示します。

- CICS を運用していない MVS/ESA 上の MQSeries への送信側チャンネル:

```
DEFINE CHANNEL(MT01.VM03.SDRC.ICE) CHLTYPE(SDR) +
  TRPTYPE(LU62) +
  SEQWRAP(9999999) MAXMSGL(2048) +
  XMITQ('VM03NCM.TQ.SDRC.ICE') +
  CONNAME('$ICE.#IYAHT0C.IYAHT0C0.IYCNVM03')+
  MODENAME('LU62PS') TPNAME(MQSERIES)
```

- MVS/ESA 上の MQSeries からの受信側チャンネル:

```
DEFINE CHANNEL(VM03.MT01.SDRC.ICE) CHLTYPE(RCVR) +
  TRPTYPE(LU62) REPLACE DESCR('Receiver channel from VM03NCM') +
  SEQWRAP(9999999) +
  MAXMSGL(2048) +
  TPNAME(VM03NCMSDRCRCVR)
```

- 会話の開始が可能な、またはリモート・リクエスター・チャンネルによって開始できる、MVS/ESA 上の MQSeries へのサーバー・チャンネル:

```
DEFINE CHANNEL(MT01.VM03.RQSV.ICE) CHLTYPE(SVR) +
  TRPTYPE(LU62) +
  SEQWRAP(9999999) MAXMSGL(2048) +
  XMITQ('VM03NCM.TQ.RQSV.ICE') +
  CONNAME('$ICE.#IYAHT0C.IYAHT0C0.IYCNVM03')+
  MODENAME('LU62PS') TPNAME(MQSERIES) +
```

ここで、MQSeries は、MVS キュー・マネージャーが listen している TPNAME です。

TCP/IP 通信の例

ここでは、TCP/IP を介してリモートの MQSeries システムとの通信を確立する方法の例を示します。

QMINI での TCPConfig スタンザ

QMINI ファイルには、適切な TCPConfig スタンザが含まれていなければなりません。たとえば、次のようにします。

通信のセットアップ

```
TCPCConfig:
  TCPPort=1414
  TCPNumListenerPorts=1
  TCPLListenerPort=1996
  TCPKeepAlive=1
```

TCPPort 値は、CONNNAME フィールドにポート値がないチャンネル用のデフォルトのアウトバウンド・ポートです。TCPLListenerPort は、コマンド行で **runmqtsr** を使うときに -p オプションが指定されない場合に使用されるデフォルト・ポートを示します。

TCP/IP 送信側チャンネルを定義する

TCP/IP 送信側チャンネルが定義されなければなりません。この例では、キュー・マネージャーは、SPINACH というホスト上の MH01 です。

```
DEFINE CHANNEL(MT01_MH01_SDR0001) CHLTYPE(SDR) +
  TRPTYPE(TCP) +
  SEQWRAP(9999999) MAXMSGL(4194304) +
  XMITQ('MH01_TQ_SDR0001') +
  CONNAME('SPINACH.HURSLEY.IBM.COM(2000)')
```

このチャンネルは、ホスト SPINACH 上の TCP/IP ポート番号 2000 に接続を試みます。

次に示すのは、デフォルト のアウトバウンド TCP/IP ポートを使った、ホスト SPINACH 上のキュー・マネージャー MH01 用の TCP/IP 送信側チャンネル定義の例です。

```
DEFINE CHANNEL(MT01_MH01_SDR0001) CHLTYPE(SDR) +
  TRPTYPE(TCP) +
  SEQWRAP(9999999) MAXMSGL(4194304) +
  XMITQ('MH01_TQ_SDR0001') +
  CONNAME('SPINACH.HURSLEY.IBM.COM')
```

CONNNAME ではポート番号は指定されていません。そのため、QMINI ファイル (1414) の TCPPort エントリーで指定された値が使用されます。

TCP/IP 受信側チャンネルを定義する

次は TCP/IP 受信側チャンネルの例です。

```
DEFINE CHANNEL(MH01_MT01_SDR001) CHLTYPE(RCVR) +
TRPTYPE(TCP)
```

TCP/IP 受信側チャンネルには CONNAME 値は必要ありませんが、TCP/IP リスナーが実行されていなければなりません。TCP/IP リスナーを始動するには、2とおりの方法があります。次のいずれかを行います。

1. PATHCOM を使ってキュー・マネージャーの PATHWAY にアクセスし、次のように入力します。

```
start server mqs-tcplis00
```

または

TACL プロンプトから次のように入力します。

```
runmqtsr -m QMgrName
```

QMINI ファイル (この例では 1996) で定義されたポートを listen する TCP/IP リスナーが始動します。

注: このポート番号は、runmqtsr の -p ポート フラグでオーバーライドされます。

リモート・システム上の TCP/IP 送信側チャンネルを定義する

ここでの受信側チャンネルに接続するためのリモート・システムに対する送信側チャンネル定義は、次のようなものになります。

```
DEFINE CHANNEL(MH01_MT01_SDR001) CHLTYPE(SDR) +
TRPTYPE(TCP) +
XMITQ('MT01_TQ_SDR001') +
CONNAME('Compaq.ISC.UK.IBM.COM(1996)')
```

通信のセットアップ

複数の TCP/IP リスナーをサポートする QMINI を構成する

キュー・マネージャーが複数の TCP/IP リスナーをサポートできるようにするには、MQS-TCPLIS00 に基づいて、追加の listener ごとに新しい PATHWAY のサーバー・クラスを作成する必要があります。

また、各 TCP/IP listener はそれ自体のリスナー・ポート・エントリーを、QMINI ファイルの TCPConfig スタンザに持っていなければなりません。

たとえば、次のようにします。

```
TCPConfig:  
TCPPort=1414  
TCPNumListenerPorts=3  
TCPListenerPort=1996  
TCPListenerPort=1997  
TCPListenerPort=1998  
TCPKeepAlive=1
```

TCPNumListenerPorts は、TCPListenerPort エントリー の数 (この例では 3 つ) と一致する必要があります。QMINI ファイルは 3 つの TCP/IP リスナーをサポートして、ポート 1996、1997 および 1998 を listen することができます。これらの 3 つのポートをサポートするサーバー・クラスは通常、MQS-TCPLIS00、MQS-TCPLIS01、および MQS-TCPLIS02 と呼ばれます。

サーバー・クラスの追加について詳しくは、41ページの『TS/MP (PATHWAY) 管理』を参照してください。

付録N. MQSeries クライアント

MQSeries クライアントは、キュー・マネージャーを含まない MQSeries システムです。MQSeries クライアント・コードは、クライアント・システムで実行中のアプリケーションから、接続先である MQSeries サーバー・システム上のキュー・マネージャーに MQI 呼び出しを送信します。

この付録では、MQSeries for Compaq NSK V5.1 に固有の MQSeries クライアントについての情報を記載しています。MQSeries クライアント と併せて使用してください。

クライアント・サポート

MQSeries for Compaq NSK は、TCP/IP または SNA LU 6.2 プロトコルを使用してサーバーに接続できる、すべての MQSeries クライアントへの MQSeries サーバー・システムとして機能します。ただし、MQSeries for Compaq NSK クライアントはありません。

MQSeries クライアントが MQSeries for Compaq NSK 上のキュー・マネージャーに接続する場合、次のようになります。

- MQ*_SYNCPOINT オプションが指定された MQGET、MQPUT、または MQPUT1 は、接続ハンドルと関連付けられていない場合は、Compaq トランザクションを開始します。
- MQ*_SYNCPOINT オプションまたは MQ*_NO_SYNCPOINT オプションが指定されていない MQGET、MQPUT、または MQPUT1 は、接続ハンドルと関連付けられていない場合は、Compaq トランザクションを開始します。
- MQCMIT 呼び出しは、接続ハンドルと関連付けられている場合は、Compaq トランザクションをコミットします。MQBACK 呼び出しは、接続ハンドルと関連付けられている場合は、Compaq トランザクションを取り消します。

どの場合でも、Compaq BEGINTRANSACTION が失敗すると、MQCC_FAILED の *CompCode*、および MQRC_SYNCPOINT_NOT_AVAILABLE の *Reason* が呼び出し側に戻されます。

セキュリティに関する考慮事項

MQSeries for Compaq NSK は、クライアントの妥当性検査のため、次のようなチャネル・セキュリティ出口の使用をサポートします。

MQSeries クライアント

- MQSeries クライアントとサーバーとの間で接続が確立された後、クライアントは、MQCONN 呼び出しから戻る前にサーバー上でセキュリティー出口を呼び出します。
- サーバー・セキュリティー出口は、クライアント・セキュリティー出口に情報を戻すことができます。

たとえば、このダイアログによって、サーバーとクライアントとの間の機密データの通信が許可されます。クライアントがセキュリティー出口を定義していない場合、ローカル環境変数 `MQ_USER_ID` および `MQ_PASSWORD` の値が、チャンネル属性を介してサーバーに渡されます。これらの属性は、サーバー・セキュリティー出口が妥当性検査用に使用できます。

付録O. プログラム式システム管理

MQSeries for Compaq NSK では、MQSeries の次のシステム管理機能をサポートしています。

- インストールメンテーション・イベント
- プログラム式コマンド形式 (PCF)
- インストール可能なサービス

この付録では MQSeries for Compaq NSK のこれらの機能の要約を述べます。詳細記述については、資料 *MQSeries プログラム式システム管理* をご覧ください。

インストールメンテーション・イベント

MQSeries for Compaq NSK では、イベント・キューにイベント・メッセージを生成する MQSeries の標準のインストールメンテーション・イベントをサポートしています。

キューおよびキュー・マネージャーの属性に適切な値を指定することにより、イベントを使用可能にしたり使用不可にしたりすることができます。以下を使用します

- MQSC (資料 *MQSeries MQSC コマンド・リファレンス* に説明があります)
- PCF コマンド (資料 *MQSeries プログラム式システム管理* に説明があります)
- メッセージ・キュー管理 (MQM) (57ページの『第4章 キュー・マネージャーの管理』に説明があります)

MQSeries for Compaq NSK がサポートするイベント・タイプ

MQSeries for Compaq NSK は、次のイベント・タイプをサポートしています。

表 27. *MQSeries for Compaq NSK* がサポートするイベント・タイプ

イベント・タイプ	イベント名
権限イベント	不許可 (タイプ 1) (Not Authorized (type 1))

イベント

表 27. MQSeries for Compaq NSK がサポートするイベント・タイプ (続き)

イベント・タイプ	イベント名
チャンネル・イベント	チャンネル活動化 (Channel Activated) チャンネル変換エラー (Channel Conversion Error) チャンネル非活動化 (Channel Not Activated) チャンネル開始 (Channel Started) チャンネル停止 (Channel Stopped)
禁止イベント	使用禁止を取得 (Get Inhibited) 使用禁止をプット (Put Inhibited)
ローカル・イベント	別名基本キュー・タイプ・エラー (Alias Base Queue Type Error) キュー・タイプ・エラー (Queue Type Error) 別名基本キュー不明 (Unknown Alias Base Queue) オブジェクト名不明 (Unknown Object Name)
パフォーマンス・イベント	キュー項目数大 (Queue Depth High) キュー項目数小 (Queue Depth Low) キュー・フル (Queue Full) キュー・サービス・インターバル大 (Queue Service Interval High) キュー・サービス・インターバル OK (Queue Service Interval OK)
リモート・イベント	デフォルト伝送キュー・タイプ・エラー (Default Transmission Queue Type Error) デフォルト伝送キュー使用エラー (Default Transmission Queue Usage Error) キュー・タイプ・エラー (Queue Type Error) リモート・キュー名エラー (Remote Queue Name Error) 伝送キュー使用エラー (Transmission Queue Usage Error) デフォルト伝送キュー不明 (Unknown Default Transmission Queue) リモート・キュー・マネージャー不明 (Unknown Remote Queue Manager) 伝送キュー不明 (Unknown Transmission Queue)
開始と停止イベント	キュー・マネージャー・アクティブ (Queue Manager Active) キュー・マネージャー非アクティブ (Queue Manager Not Active)

イベント・メッセージの形式

MQSeries for Compaq NSK は、MQSeries の標準のイベント・メッセージの形式をサポートしています。つまり、イベント・メッセージには、メッセージ記

述子 (MQMD) とメッセージ・データ の 2 つの部分があります。このメッセージ・データは、イベントのタイプに固有のイベント・ヘッダーと何らかのデータを含みます。

イベント・メッセージの MQMD 構造については、430ページの『MQMD - メッセージ記述子』に要約があります。イベント・ヘッダー構造 (MQCFH) については、500ページの表29 に要約があります。

表 28. イベント・メッセージの MQMD 構造

パラメーター	タイプ	値
<i>StrucId</i>	MQCHAR4	MQMD_STRUC_ID
<i>Version</i>	MLONG	MQMD_VERSION_1
<i>Report</i>	MLONG	MQRO_NONE
<i>MsgType</i>	MLONG	MQMT_DATAGRAM
<i>Expiry</i>	MLONG	MQEI_UNLIMITED
<i>Feedback</i>	MLONG	MQFB_NONE
<i>Encoding</i>	MLONG	イベントを生成するキュー・マネージャーのエンコード方式。
<i>CodedCharSetId</i>	MLONG	イベントを生成するキュー・マネージャーのコード化文字セット ID (CCSID)。
<i>Format</i>	MQCHAR8	MQFMT_EVENT
<i>Priority</i>	MLONG	イベント・キューがローカル・キューの場合のそのデフォルトの優先順位、またはイベントを生成するキュー・マネージャーでのそのローカル定義。
<i>Persistence</i>	MLONG	イベント・キューがローカル・キューの場合のそのデフォルトのパーシスタンス (永続性)、またはイベントを生成するキュー・マネージャーでのそのローカル定義。
<i>MsgId</i>	MQBYTE24	この値はキュー・マネージャーによって一意的に生成されます。
<i>CorrelId</i>	MQBYTE24	MQCI_NONE
<i>BackoutCount</i>	MLONG	この値は常に 0 です。
<i>ReplyToQ</i>	MQCHAR48	常にブランクです。
<i>ReplyToQMgr</i>	MQCHAR48	発生システムでのキュー・マネージャー名。
<i>UserIdentifier</i>	MQCHAR12	常にブランクです。

イベント

表 28. イベント・メッセージの MQMD 構造 (続き)

パラメーター	タイプ	値
<i>AccountingToken</i>	MQBYTE32	MQACT_NONE
<i>ApplIdentityData</i>	MQCHAR32	常に空白です。
<i>PutApplType</i>	MQLONG	メッセージを書き込んだアプリケーションのタイプ。
<i>PutApplName</i>	MQCHAR28	メッセージを書き込んだアプリケーションの名前。
<i>PutDate</i>	MQCHAR8	メッセージが書き込まれたときの日付で、キュー・マネージャーによって生成されます。
<i>PutTime</i>	MQCHAR8	メッセージが書き込まれたときの時刻で、キュー・マネージャーによって生成されます。
<i>ApplOriginData</i>	MQCHAR4	常に空白です。

表 29. イベント・ヘッダー構造 (MQCFH)

パラメーター	タイプ	値
<i>Type</i>	MQLONG	MQCFT_EVENT
<i>StrucLength</i>	MQLONG	MQCFH_STRUC_LENGTH
<i>Version</i>	MQLONG	MQCFH_VERSION_1
<i>Command</i>	MQLONG	MQCMD_Q_MGR_EVENT MQCMD_PERFM_EVENT MQCMD_CHANNEL_EVENT
<i>MsgSeqNumber</i>	MQLONG	常に 1 です。
<i>Control</i>	MQLONG	MQCFC_LAST
<i>CompCode</i>	MQLONG	MQCC_OK MQCC_WARNING
<i>Reason</i>	MQLONG	イベントを識別する理由コード。
<i>ParameterCount</i>	MQLONG	MQCFH 構造の次にくるパラメーター構造の数。

プログラム式コマンド形式 (PCF)

MQSeries for Compaq NSK は、標準のプログラム式コマンド形式 (PCF) の機能 (資料 *MQSeries* プログラム式システム管理 に説明があります) をサポートしています。PCF メッセージは、メッセージ記述子 (MQMD) とメッセージ・データの 2 つの部分で構成されます。メッセージ・データは、

MQCFIN、MQCFIL、MQCFST、および MQCFSL 構造で定義される、PCF ヘッダー (MQCFH) と何らかの PCF パラメーターを含みます。

PCF メッセージ記述子 (MQMD) の要約が、表30 にあります。PCF ヘッダー構造 (MQCFH) の要約が、502ページの表31 にあります。PCF パラメーター構造の要約が、502ページの表32 から 503ページの表35 にあります。

PCF メッセージ記述子

MQSeries for Compaq NSK では、標準の PCF メッセージ記述子を適用しています。つまり、メッセージ記述子には以下のフィールドが含まれます。

表 30. PCF メッセージ記述子

フィールド	値
<i>Report</i>	任意の有効値
<i>MsgType</i>	MQMT_REQUEST
<i>Expiry</i>	任意の有効値
<i>Feedback</i>	MQFB_NONE
<i>Encoding</i>	メッセージ・データに使用されるエンコード方式。必要な場合は変換が行われる。
<i>CodedCharSetId</i>	メッセージ・データに使用される CCSID。必要な場合は変換が行われる。
<i>Format</i>	MQFMT_ADMIN MQFMT_PCF (ユーザー・データ用)
<i>Priority</i>	任意の有効値
<i>Persistence</i>	任意の有効値
<i>MsgId</i>	MQML_NONE を含む任意の有効値
<i>CorrelId</i>	MQML_NONE を含む任意の有効値
<i>ReplyToQ</i>	キュー名
<i>ReplyToQMgr</i>	キュー・マネージャー名
Message context fields	MQPMO_DEFAULT_CONTEXT を含む任意の有効値

PCF ヘッダー (MQCFH)

MQSeries for Compaq NSK では、標準の PCF ヘッダーを適用しています。つまり、PCF ヘッダー構造には以下のフィールドを含みます。

表 31. PCF ヘッダー

フィールド	タイプ	値
<i>Type</i>	MQLONG	MQCFT_COMMAND MQCFT_RESPONSE MQCFT_EVENT
<i>StrucLength</i>	MQLONG	MQCFH_STRUC_LENGTH
<i>Version</i>	MQLONG	MQCFH_VERSION_1
<i>Command</i>	MQLONG	有効なコマンド ID。
<i>MsgSeqNumber</i>	MQLONG	メッセージのシーケンス番号。
<i>Control</i>	MQLONG	MQCFC_LAST MQCFC_NOT_LAST
<i>CompCode</i>	MQLONG	MQCC_OK MQCC_WARNING MQCC_FAILED MQCC_UNKNOWN
<i>Reason</i>	MQLONG	完了コードを修飾する理由コード。
<i>ParameterCount</i>	MQLONG	パラメーター構造のカウント。

PCF スtring・パラメーター (MQCFST)

MQSeries for Compaq NSK では、標準の PCF String・パラメーター構造 (MQCFST) を適用しています。つまり、PCF String・パラメーター構造には以下のフィールドを含みます。

表 32. PCF String・パラメーター

フィールド	タイプ	値
<i>Type</i>	MQLONG	MQCFT_STRING
<i>StrucLength</i>	MQLONG	MQCFST 構造の長さ (バイト単位)。
<i>Parameter</i>	MQLONG	パラメーター ID
<i>CodedCharSetId</i>	MQLONG	コード化文字セット ID (CCSID)。
<i>StringLength</i>	MQLONG	<i>String</i> フィールドでのデータの長さ (バイト単位)。
<i>String</i>	MQCHAR × <i>StringLength</i>	String 値。

PCF 整数リスト・パラメーター (MQCFIL)

MQSeries for Compaq NSK では、標準の PCF 整数リスト・パラメーター構造 (MQCFIL) を適用しています。つまり、PCF 整数リスト・パラメーター構造に

は以下のフィールドを含みます。

表 33. PCF 整数リスト

フィールド	タイプ	値
<i>Type</i>	MQLONG	MQCFT_INTEGER_LIST
<i>StrucLength</i>	MQLONG	MQCFIL 構造の長さ (バイト単位)。
<i>Parameter</i>	MQLONG	パラメーター ID
<i>Count</i>	MQLONG	<i>Values</i> 配列の中のエレメントの数。
<i>Values</i>	MQLONG × <i>Count</i>	パラメーター値。

PCF 整数 (MQCFIN)

MQSeries for Compaq NSK では、標準の PCF 整数構造 (MQCFIN) を適用しています。つまり、PCF 整数構造には以下のフィールドを含みます。

表 34. PCF 整数

フィールド	タイプ	値
<i>Type</i>	MQLONG	MQCFT_INTEGER
<i>StrucLength</i>	MQLONG	MQCFIN_STRUC_LENGTH
<i>Parameter</i>	MQLONG	パラメーター ID
<i>Value</i>	MQLONG	パラメーター値

PCF スtring・リスト (MQCFSL)

MQSeries for Compaq NSK では、標準の PCF スtring・リスト構造 (MQCFSL) を適用しています。つまり、PCF スtring・リスト構造には以下のフィールドを含みます。

表 35. PCF スtring・リスト

フィールド	タイプ	値
<i>Type</i>	MQLONG	MQCFT_STRING_LIST
<i>StrucLength</i>	MQLONG	MQCFSL 構造の長さ (バイト単位)
<i>Parameter</i>	MQLONG	パラメーター ID
<i>CodedCharSetId</i>	MQLONG	<i>Strings</i> フィールドのデータの CCSID。
<i>Count</i>	MQLONG	<i>Strings</i> フィールドの String の数。

表 35. PCF スtring・リスト (続き)

フィールド	タイプ	値
<i>StringLength</i>	MQLONG	<i>Strings</i> フィールドの各Stringの長さ (バイト単位)。
<i>Strings</i>	MQCHAR × <i>StringLength</i> × <i>Count</i>	<i>Parameter</i> フィールドで示されるパラメーターのString値のセット。

MQSeries for Compaq NSK がサポートする PCF コマンド

以下の MQSeries PCF コマンドが MQSeries for Compaq NSK でサポートされています。これらのコマンドの詳細な説明は、資料 *MQSeries プログラム式システム管理* をご覧ください。

表 36. MQSeries for Compaq NSK がサポートする PCF コマンド

コマンド	コマンド ID
Change Channel	MQCMD_CHANGE_CHANNEL
Change Namelist	MQCMD_CHANGE_NAMELIST
Change Process	MQCMD_CHANGE_PROCESS
Change Queue	MQCMD_CHANGE_Q
Change Queue Manager	MQCMD_CHANGE_Q_MGR
Clear Queue	MQCMD_CLEAR_Q
Copy Channel	MQCMD_COPY_CHANNEL
Copy Namelist	MQCMD_COPY_NAMELIST
Copy Process	MQCQ_COPY_PROCESS
Copy Queue	MQCMD_COPY_Q
Create Channel	MQCMD_CREATE_CHANNEL
Create Namelist	MQCMD_CREATE_NAMELIST
Create Process	MQCMD_CREATE_PROCESS
Create Queue	MQCMD_CREATE_Q
Delete Channel	MQCMD_DELETE_CHANNEL
Delete Namelist	MQCMD_DELETE_NAMELIST
Delete Process	MQCMD_DELETE_PROCESS
Delete Queue	MQCMD_DELETE_Q
Escape	MQCMD_ESCAPE
Inquire Channel	MQCMD_INQUIRE_CHANNEL
Inquire Channel Names	MQCMD_INQUIRE_CHANNEL_NAMES

表 36. MQSeries for Compaq NSK がサポートする PCF コマンド (続き)

コマンド	コマンド ID
Inquire Channel Status	MQCMD_INQUIRE_CHANNEL_STATUS
Inquire Cluster Queue Manager	MQCMD_INQUIRE_CLUSTER_Q_MGR
Inquire Namelist	MQCMD_INQUIRE_NAMELIST
Inquire Process	MQCMD_INQUIRE_PROCESS
Inquire Process Names	MQCMD_INQUIRE_PROCESS_NAMES
Inquire Queue	MQCMD_INQUIRE_Q
Inquire Queue Manager	MQCMD_INQUIRE_Q_MGR
Inquire Queue Names	MQCMD_INQUIRE_Q_NAMES
Ping Channel	MQCMD_PING_CHANNEL
Ping Queue Manager	MQCMD_PING_Q_MGR
Refresh Cluster	MQCMD_REFRESH_CLUSTER
Reset Channel	MQCMD_RESET_CHANNEL
Reset Queue Statistics	MQCMD_RESET_Q_STATS
Resolve Channel	MQCMD_RESOLVE_CHANNEL
Resume Queue Manager Cluster	MQCMD_RESUME_Q_MGR_CLUSTER
Start Channel	MQCMD_START_CHANNEL
Stop Channel	MQCMD_STOP_CHANNEL
Suspend Queue Manager Cluster	MQCMD_SUSPEND_Q_MGR_CLUSTER

注: MQSeries for Compaq NSK では、Start Channel Initiator および Start Channel Listener コマンドはサポートしていません。

PCF コマンド応答

MQSeries for Compaq NSK では、コマンド・サーバーが各コマンドに対して標準の応答メッセージを生成します。応答には次の 3 つのタイプがあります。

- OK 応答
- エラー応答
- データ応答

詳しくは、資料 *MQSeries プログラム式システム管理* を参照してください。

インストール可能なサービス

MQSeries for Compaq NSK は、許可サービスと名前サービスをサポートしています。

許可サービス・インターフェース

許可サービスによって、キュー・マネージャーは許可機能呼び出すことができます。たとえば、キュー・マネージャーは、特定のユーザー ID がキューを開く許可を与えられていることを、許可サービスを使って検査することができます。

許可サービス・コンポーネントは MQSeries for Compaq NSK で提供されます。このコンポーネントはオブジェクト権限マネージャー (OAM) と呼ばれます。デフォルトでは、この OAM は、制御コマンド **dspmqaout** (表示権限) および **setmqaut** (設定権限) でアクティブになり機能します。

この OAM をユーザー独自の許可サービス・コンポーネントで増強したり置き換えることができます。このことについては、資料 *MQSeries プログラム式システム管理* で述べられています。

名前サービス・インターフェース

名前サービスは、キュー・マネージャーが、キューを所有するキュー・マネージャーの名前を解決する際の支援を行います。

標準の名前サービス・インターフェース (資料 *MQSeries プログラム式システム管理* に説明があります) が、MQSeries for Compaq NSK によってサポートされています。

付録P. MQSeries for Compaq NSK で使用する EMS イベント・テンプレート

EMS テンプレート・ファイル (SMQSTMPL) には、MQSeries の EMS イベントの定義用のソース・コードが含まれています。これらの定義によって、EMS イベント・メッセージにある情報の表示の仕方を制御し、また各 EMS イベント・メッセージに含まれるデータのタイプと意味も表示します。

イベントの次のタイプが生成されます。

ZMQS-VAL-EVT-ERROR

FFST (システム・リソースの障害、ソフトウェア障害、またはハードウェア障害)。

ZMQS-VAL-EVT-ERR

MQSeries に関するエラーで、FFST イベントとディスク上のログ・データを参照します。

ZMQS-VAL-EVT-MSG

キュー・マネージャーやチャネルの始動などについての、MQSeries メッセージ。これらのイベントはすべて MQSeries の AMQxxxxx というログ・メッセージに対応しており、それらと同じ情報とテキストを含んでいます。各メッセージの変数データは、イベント・メッセージ内の個々のトークンに含まれています。AMQxxxxx メッセージの詳細については、MQSeries メッセージを参照してください。

ZMQS-VAL-EVT-QMGR

権限、禁止、ローカル、リモート、開始、および停止の各イベントについてのキュー・マネージャーのイベント。これらの EMS イベントの情報内容は、実際は、それらと対応する PCF イベント・メッセージの情報内容と同じです。これについては、MQSeries プログラム式システム管理 で説明されています。イベント・メッセージでの個々のトークンには、各イベント・メッセージでの変数データが含まれます。

ZMQS-VAL-EVT-PERF

パフォーマンス・イベントで、MQSeries の標準のパフォーマンス・イベントと対応しています。これらのイベントは、キュー・マネージャー内のキューについての統計データを報告します。パフォーマンス・イベントでの変数データは、イベント・メッセージ内の個々のトークンに含まれています。

EMS イベント・テンプレート

ZMQS-VAL-EVT-CHNL

チャンネル・イベントで、MQSeries の標準のチャンネル・イベントと対応しています。チャンネル・イベントは、チャンネルの状況の変化や、キュー・マネージャー間の通信での問題を報告します。他のタイプのイベント・メッセージと同様に、チャンネル・イベントでの変数データは、イベント・メッセージ内の個々のトークンに含まれています。

以下は、EMS テンプレートの定義からの抜粋です。

```
VERSION: "IBM.MQS - 10JAN97"
```

```
SSID: ZMQS-VAL-SSID
```

```
SSNAME: "MQSeries", "MQS"
```

```
==
```

```
== This is an EMS FFST message
```

```
==
```

```
MSG: ZEMS-TKN-EVENTNUMBER, ZMQS-VAL-EVT-ERROR
```

```
  OVERRIDE ZEMS-TKN-EMPHASIS ZSPI-VAL-TRUE
```

```
  "MQSeries FFST from component COMP_<1>           "
```

```
  "<*>CR> Error Code : <2>                           "
```

```
  "<*>CR> Severity : <3>                               "
```

```
  "<*>CR> Module Name : <4>                           "
```

```
  "<*>CR> Probe ID : <5>                               "
```

```
  "<*>CR> Error Text :                               "
```

```
  "<*>CR> <6>"
```

```
  1: ZMQS-TKN-COMPONENT
```

```
  2: ZMQS-TKN-ERROR-CODE
```

```
  3: ZMQS-TKN-SEVERITY
```

```
  4: ZMQS-TKN-MODULE-NAME
```

```
  5: ZMQS-TKN-PROBE-ID
```

```
  6: ZMQS-TKN-ERROR-TEXT
```

```
==
```

```
== This is an EMS Display Message Event
```

```
==
```

```
MSG: ZEMS-TKN-EVENTNUMBER, ZMQS-VAL-EVT-MSG
```

```
  "MQSeries message: <1>                               "
```

```
  "
```

```
  "<*>CR> EXPLANATION :                               "
```

```
    "<*>CR> <2>                                           "
```

```
  "
```

```
  "<*>CR> ACTION :                                     "
```

```
    "<*>CR> <3>"
```

```
  1: ZMQS-TKN-ERROR-TEXT
```

```
  2: ZMQS-TKN-ERROR-TEXT-2
```

```
  3: ZMQS-TKN-ERROR-TEXT-3
```

```
==
```

```
== This is an EMS Report Error Event
```

```
==
```

```

MSG: ZEMS-TKN-EVENTNUMBER, ZMQS-VAL-EVT-ERR
      OVERRIDE ZEMS-TKN-EMPHASIS ZSPI-VAL-TRUE
      "MQSeries Error"
      "<*CR> Error Code : <1>"
      "<*CR> Function : <2>"
      "<*CR> Probe ID : <3>"
      "<*CR> FFST File : <4>"
      1: ZMQS-TKN-ERROR-CODE
      2: ZMQS-TKN-MODULE-NAME
      3: ZMQS-TKN-PROBE-ID
      4: ZMQS-TKN-FILE-NAME

==
== This is an EMS copy of PCF Queue Manager event message
== for authority, inhibit, local, remote, start_and_stop events
==
MSG: ZEMS-TKN-EVENTNUMBER, ZMQS-VAL-EVT-QMGR
      "MQSeries QMgr Event from <1>"
      "<*CR> Reason : <2>"
      "<*IF 3><*CR> Reason Qualifier : <4>"
      "<*IF 5><*CR> User ID : <6>"
      "<*IF 13><*CR> Object QMgr : <14>"
      "<*IF 9><*CR> Options : <10>"
      "<*IF 11><*CR> Command : <12>"
      "<*IF 15><*CR> Queue Name : <16>"
      "<*IF 17><*CR> Queue Type : <18>"
      "<*IF 19><*CR> Base Queue Name : <20>"
      "<*IF 21><*CR> XMit Queue Name : <22>"
      "<*IF 30><*CR> Application Type : <31>"
      "<*IF 32><*CR> Application Name : <33>"
      1: ZMQS-TKN-QMGR
      2: ZMQS-TKN-REASON
      3: TOKENPRESENT (ZMQS-TKN-REASON-QUALIFIER)
      4: ZMQS-TKN-REASON-QUALIFIER
      5: TOKENPRESENT (ZMQS-TKN-USER-ID)
      6: ZMQS-TKN-USER-ID
      9: TOKENPRESENT (ZMQS-TKN-OPTIONS)
      10: ZMQS-TKN-OPTIONS
      11: TOKENPRESENT (ZMQS-TKN-COMMAND)
      12: ZMQS-TKN-COMMAND
      13: TOKENPRESENT (ZMQS-TKN-OBJ-QMGR)
      14: ZMQS-TKN-OBJ-QMGR
      15: TOKENPRESENT (ZMQS-TKN-Q-NAME)
      16: ZMQS-TKN-Q-NAME
      17: TOKENPRESENT (ZMQS-TKN-Q-TYPE)
      18: ZMQS-TKN-Q-TYPE
      19: TOKENPRESENT (ZMQS-TKN-BASE-Q-NAME)
      20: ZMQS-TKN-BASE-Q-NAME
      21: TOKENPRESENT (ZMQS-TKN-XMIT-Q-NAME)
      22: ZMQS-TKN-XMIT-Q-NAME
      30: TOKENPRESENT (ZMQS-TKN-APPL-TYPE)
      31: ZMQS-TKN-APPL-TYPE
      32: TOKENPRESENT (ZMQS-TKN-APPL-NAME)
      33: ZMQS-TKN-APPL-NAME

```

EMS イベント・テンプレート

```
==
== This is an EMS copy of PCF Performance event message
==
MSG: ZEMS-TKN-EVENTNUMBER, ZMQS-VAL-EVT-PERF
"MQSeries Performance Event from <1>           "
"<*>CR> Reason : <2>                           "
"<*>CR> Queue Name : <3>                         "
"<*>CR> Time Since Last Reset : <4>              "
"<*>CR> Highest Queue Depth : <5>                "
"<*>CR> # Of Messages Enqueued : <6>             "
"<*>CR> # Of Messages Dequeued : <7>            "
1: ZMQS-TKN-QMGR
2: ZMQS-TKN-REASON
3: ZMQS-TKN-Q-NAME
4: ZMQS-TKN-TIME-SINCE-RESET
5: ZMQS-TKN-HIGH-Q-DEPTH
6: ZMQS-TKN-MSG-ENQ-COUNT
7: ZMQS-TKN-MSG-DEQ-COUNT
```

```
==
== This is an EMS copy of PCF Channel event message
==
MSG: ZEMS-TKN-EVENTNUMBER, ZMQS-VAL-EVT-CHNL
"MQSeries Channel Event from <1>                 "
"<*>CR> Reason : <2>                             "
"<*>CR> Channel Name : <3>                       "
"<*>CR> XMit Queue Name : <5>                    "
"<*>CR> Connection Name : <7>                   "
"<*>CR> Reason Qualifier : <9>                  "
"<*>CR> Format : <11>                            "
"<*>CR> Return Code : <13>                      "
"<*>CR> Auxiliary rc 1 : <15>                   "
"<*>CR> Auxiliary rc 2 : <17>                   "
"<*>CR> CCSID 1 : <19>                           "
"<*>CR> Auxiliary string 1 : <21>                "
"<*>CR> CCSID 2 : <23>                           "
"<*>CR> Auxiliary string 2 : <25>                "
"<*>CR> CCSID 3 : <27>                           "
"<*>CR> Auxiliary string 3 : <29>                "
1: ZMQS-TKN-QMGR
2: ZMQS-TKN-REASON
3: ZMQS-TKN-CHANNEL-NAME
5: ZMQS-TKN-XMIT-Q-NAME
7: ZMQS-TKN-CONN-NAME
9: ZMQS-TKN-REASON-QUALIFIER
11: ZMQS-TKN-FORMAT
13: ZMQS-TKN-RETURN-CODE
15: ZMQS-TKN-RETURN-CODE-2
17: ZMQS-TKN-RETURN-CODE-3
19: ZMQS-TKN-CCSID
21: ZMQS-TKN-ERROR-TEXT
```

23: ZMQS-TKN-CCSID-2
25: ZMQS-TKN-ERROR-TEXT-2
27: ZMQS-TKN-CCSID-3
29: ZMQS-TKN-ERROR-TEXT-3

付録Q. 特記事項

本書はアメリカ合衆国で提供されている製品およびサービス用に作成されたものであり、本書に記載の製品、サービス、またはフィーチャーが日本においては提供されていない場合があります。日本で利用可能な製品、サービス、およびフィーチャーについては、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等な製品、プログラム、またはサービスを使用することができます。ただし、IBM 製以外の製品と組み合わせた場合、その操作の評価と検証については、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権の許諾については、下記の宛先に、書面にてご照会ください。

〒106-0032 東京都港区六本木 3 丁目 2-31

AP 事業所

IBM World Trade Asia Corporation

Intellectual Property Law & Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。 IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

本書は定期的に見直され、必要な変更 (たとえば、技術的に不適確な表現や誤植など) は、本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するもので

特記事項

はありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM United Kingdom Laboratories,
Mail Point 151,
Hursley Park,
Winchester,
Hampshire,
England
SO21 2JN.

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。また、IBM 以外の製品に関するパフォーマンスの正確性、互換性、またはその他の要求は確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、IBM のアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あ

らゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほめかしたり、保証することはできません。お客様は、IBM のアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。

商標

以下は、IBM Corporation の米国およびその他の国における商標です。

AIX	IBM
MQSeries	AS/400
MVS/ESA	FFST
CICS	OS/2
First Failure Support Technology	VSE/ESA
OS/390	BookManager
IBMLink	MVS
SupportPac	VTAM

Compaq and NonStop are trademarks of Compaq Computer Corporation.

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

Microsoft、Windows、および Windows NT は、Microsoft Corporation の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, 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。

UNIX は、The Open Group がライセンスしている米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標または登録商標です。

参照文献

この節では、現在の MQSeries 全プロダクトに関する資料について説明します。

MQSeries 複数のプラットフォームに共通する資料

これらの資料のほとんど (MQSeries 「ファミリー」資料と呼ばれる場合もある) は、すべての MQSeries プロダクトに適用されます。最新の MQSeries プロダクトは次のとおりです。

- MQSeries for AIX V5.2
- MQSeries for AS/400 V5.2
- MQSeries for AT&T GIS UNIX V2.2
- MQSeries for Compaq OpenVMS Alpha V5.1
- MQSeries (Compaq Tru64 UNIX 版) V5.1
- MQSeries for HP-UX V5.2
- MQSeries for Linux V5.2
- MQSeries (OS/2 Warp 版) V5.1
- MQSeries for OS/390 V5.2
- MQSeries for SINIX and DC/OSx V2.2
- MQSeries for Sun Solaris V5.2
- MQSeries for Sun Solaris, Intel Platform Edition, V5.1
- MQSeries for Compaq NSK V5.1
- MQSeries for VSE/ESA V2.1.1
- MQSeries for Windows V2.0
- MQSeries for Windows V2.1
- MQSeries for Windows NT and Windows 2000 V5.2

複数のプラットフォームを対象とする MQSeries 資料は、次のとおりです。

- *MQSeries Brochure* (G511-1908)

- *An Introduction to Messaging and Queuing* (GC33-0805)
- *MQSeries 相互通信* (SC88-7775)
- *MQSeries キュー・マネージャー・クラスター* (SD88-7165)
- *MQSeries クライアント* (GC88-7495)
- *MQSeries システム管理の手引き* (SC88-7776)
- *MQSeries MQSC コマンド・リファレンス* (SC88-7315)
- *MQSeries イベント・モニター* (SC88-8705)
- *MQSeries プログラム式システム管理* (SC88-7562)
- *MQSeries 管理インターフェースのプログラミングの手引きおよび解説書* (SD88-7145)
- *MQSeries メッセージ* (GC88-7777)
- *MQSeries アプリケーション・プログラミング・ガイド* (SC88-7253)
- *MQSeries アプリケーション・プログラミング・リファレンス* (SC88-7354)
- *MQSeries アプリケーション・プログラミング解説書 要約* (SX88-7026)
- *MQSeries C++ の使用* (SC88-7778)
- *MQSeries Java の使用* (SD88-7163)
- *MQSeries アプリケーション・メッセージング・インターフェース* (SC88-8704)

MQSeries プラットフォーム固有の資料

各 MQSeries プロダクトには、MQSeries ファミリーの資料以外に、プラットフォーム固有の資料が少なくとも 1 冊あります。

MQSeries for AIX V5.2

MQSeries for AIX インストール・ガイド (GC88-7770)

MQSeries for AS/400 V5.2

MQSeries for AS/400[®] インストール・ガイド (GD88-7309)

MQSeries (AS/400 版) システム管理の手引き (SD88-7310)

MQSeries (AS/400 版) アプリケーション・プログラミング解説書 (ILE RPG) (SD88-7311)

MQSeries for AT&T GIS UNIX V2.2

MQSeries (AT&T GIS UNIX[®] 版) システム管理の手引き (SC88-7646)

MQSeries for Compaq OpenVMS Alpha, V5.1

MQSeries for Compaq OpenVMS Alpha スタートアップ・ガイド (GC88-8943)

MQSeries for Compaq OpenVMS Alpha システム管理ガイド (SC88-8942)

MQSeries for Compaq NSK, V5.1

MQSeries for Compaq NSK スタートアップ・ガイド (GC88-8945)

MQSeries for Compaq NSK システム管理ガイド (SC88-8944)

MQSeries (Compaq Tru64 UNIX 版) V5.1

MQSeries (Compaq Tru64 UNIX 版) インストールの手引き (GC88-8625)

MQSeries for HP-UX V5.2

MQSeries for HP-UX インストール・ガイド (GC88-7772)

MQSeries for Linux V5.2

MQSeries for Linux インストール・ガイド (GD88-7354)

MQSeries (OS/2 Warp 版) V5.1

MQSeries (OS/2 Warp 版) インストールの手引き (GC88-7771)

MQSeries for OS/390 V5.2

MQSeries for OS/390[®] 概説および計画ガイド (GC88-8615)

MQSeries for OS/390 システム・セットアップ・ガイド (SC88-8616)

MQSeries for OS/390 システム管理ガイド (SC88-8617)

MQSeries for OS/390 問題判別ガイド (GC88-8846)

MQSeries for OS/390 メッセージおよびコード (GC88-8845)

MQSeries for OS/390 Licensed Program Specifications (GC34-5893)

MQSeries for OS/390 Program Directory

MQSeries link for R/3, Version 1.2

MQSeries link for R/3 使用者の手引き (GD88-7041)

MQSeries for SINIX and DC/OSx V2.2

*MQSeries for SINIX and DC/OSx
System Management Guide*
(GC33-1768)

MQSeries for Sun Solaris V5.2

MQSeries for Sun Solaris インストール・ガイド (GC88-7773)

MQSeries for Sun Solaris, Intel Platform Edition, V5.1

MQSeries for Sun Solaris, Intel Platform Edition インストール・ガイド (GC88-8749)

MQSeries for VSE/ESA V2.1.1

MQSeries for VSE/ESA™ Licensed Program Specifications
(GC34-5365)

MQSeries for VSE/ESA System Management Guide (GC34-5364)

MQSeries for Windows V2.0

MQSeries (Windows® 拡張版) 使用者の手引き (GC88-7649)

MQSeries for Windows V2.1

MQSeries (Windows 拡張版) 使用者の手引き (GC88-7208)

MQSeries for Windows NT and Windows 2000 V5.2

MQSeries for Windows NT and Windows 2000 インストール・ガイド (GD88-7162)

MQSeries (Windows NT® 版) コンポーネント・オブジェクト・モデル・インターフェースの使用
(SD88-7161)

MQSeries LotusScript Extension
(SD88-7146)

ソフトコピー資料

MQSeries に関する大部分の資料は、ハードコピー形式とソフトコピー形式の両方で提供されています。

HTML 形式

以下の MQSeries プロダクトには、HTML 形式の MQSeries 関連資料が付属しています。

- MQSeries for AIX V5.2
- MQSeries for AS/400 V5.2
- MQSeries for Compaq OpenVMS Alpha V5.1
- MQSeries (Compaq Tru64 UNIX 版) V5.1
- MQSeries for HP-UX V5.2
- MQSeries for Linux V5.2
- MQSeries (OS/2 Warp 版) V5.1
- MQSeries for OS/390 V5.2
- MQSeries for Sun Solaris V5.2
- MQSeries for Sun Solaris, Intel Platform Edition, V5.1
- MQSeries for Windows NT and Windows 2000 V5.2 (コンパイルされた HTML)
- MQSeries link for R/3 V1.2

HTML 形式の MQSeries 資料は、次の URL アドレスの MQSeries ソフトウェア・サーバーのホーム・ページにもあります。

<http://www.ibm.com/software/mqseries/>

PDF

Adobe Acrobat Reader を使用することによって、PDF ファイルの表示や印刷を行えます。

Adobe Acrobat Reader を入手したい場合、Acrobat Reader の使用が可能なプラットフォーム

参考文献

ームについての最新情報が必要な場合は、次の Adobe Systems Inc. Web サイトにアクセスしてください。

<http://www.adobe.com/>

以下の MQSeries プロダクトには、PDF バージョンの MQSeries 関連資料が付属しています。

- MQSeries for AIX V5.2
- MQSeries for AS/400 V5.2
- MQSeries for Compaq OpenVMS Alpha V5.1
- MQSeries (Compaq Tru64 UNIX 版) V5.1
- MQSeries for HP-UX V5.2
- MQSeries for Linux V5.2
- MQSeries (OS/2 Warp 版) V5.1
- MQSeries for OS/390 V5.2
- MQSeries for Sun Solaris V5.2
- MQSeries for Sun Solaris, Intel Platform Edition, V5.1
- MQSeries for Windows NT and Windows 2000 V5.2
- MQSeries link for R/3 V1.2

現在の MQSeries 全資料の PDF バージョンは、次の MQSeries プロダクト・ファミリー Web サイトからも入手できます。

<http://www.ibm.com/software/mqseries/>

インターネットで利用できる MQSeries 情報

MQSeries プロダクト・ファミリーのホーム・ページの URL は以下のとおりです。

<http://www.ibm.com/software/mqseries/>

この URL からのリンクを利用することにより、次のようなサービスを利用できます。

- MQSeries プロダクト・ファミリーの最新情報へのアクセス

- HTML 形式の MQSeries 資料の入手
- MQSeries SupportPac™ のダウンロード

参考資料

- *SNAX/APC Planning and Configuration Manual* (Compaq 部品番号 098289)

SNAX/APC は、SNA の Compaq インプリメンテーション用に LU 6.2 をサポートしています。このガイドは、SNAX/APC のインストールおよび構成方法を説明しています。

- *SCF Reference Manual for SNAX/APC* (Compaq 部品番号 064525)

SNAX/APC は、SNA の Compaq インプリメンテーション用に LU 6.2 をサポートしています。このガイドは、オペレーターおよびネットワーク・マネージャーによる SNAX/APC の構成および制御を可能にする、サブシステム制御機能 (SCF) 対話式インターフェースについて説明しています。

- *Pathway System Management Guide* (Compaq 部品番号 096881)

このガイドは、Pathway トランザクション処理システムを構成および制御するためのガイドラインを示しています。

- *Introduction to NonStop Transaction Manager/MP (TM/MP)* (Compaq 部品番号 085812)

このガイドは、TMF サブシステムを使用して、ビジネス・トランザクションおよびデータベースの保全性を保護する方法について説明しています。

- *Introduction to Compaq Networking and Data Communications* (Compaq 部品番号 093148)

このガイドは、Compaq ネットワーキング およびデータ通信の概念、タスク、製品、およびマニュアルについての概要を提供します。

- *Intersystem Communications Environment (ICE) Installation Guide* (V3.2 またはそれ以降の版)

このガイドは、ICE をインストールして、ICE 始動パラメーターを構成する方法について説明しています。(ICE は、SNA の Insession のインプリメンテーション用に LU 6.2 をサポートしています。)

- *Intersystem Communications Environment (ICE) Administrator's Guide* (V3.2 またはそれ以降の版)

このガイドは、ICE、そのインターフェース、およびそのユーティリティーを構成して操作するための方法について説明しています。

用語集および略語集

この用語集では、本書で使用する MQSeries の用語および省略語を定義しています。特定の用語が見つからない場合は、索引または *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994 を参照してください。

この用語集には、*American National Dictionary for Information Systems*, ANSI X3.172-1990 (copyright 1990 by the American National Standards Institute (ANSI)) から引用した用語および定義が含まれています。この資料は American National Standards Institute, 11 West 42 Street, New York, New York 10036 から購入することができます。この資料からの定義には、定義の最後に (A) という記号を付けて区別してあります。

[ア行]

アプリケーション・キュー (application queue). アプリケーションが使用するキュー。

アラート (alert). ネットワーク内の管理サービス中心拠点へ送られるメッセージで、問題または緊急問題を識別するためのもの。

一時動的キュー (temporary dynamic queue). クローズ時に削除される動的キュー。一時動的キューは、キュー・マネージャーで障害が発生した場合に回復されないの、非持続メッセージしか入れることができない。「永続動的キュー (permanent dynamic queue)」と対比。

イベント (event). 「チャンネル・イベント (channel event)」、「インスツルメンテーション・イベント (instrumentation event)」、「パフォーマンス・イベント (performance event)」、および

「キュー・マネージャー・イベント (queue manager event)」を参照。

イベント・キュー (event queue). キュー・マネージャーが、イベントを検出した後、イベント・メッセージを書き込むキュー。イベントの各カテゴリー (キュー・マネージャー、パフォーマンス、またはチャンネルのイベント) には、それぞれ独自のイベント・キューがある。

イベント・データ (event data). イベント・メッセージでは、イベントについての情報 (キュー・マネージャー名、イベントを引き起こしたアプリケーションなど) を含んでいるメッセージ・データ部分。「イベント・ヘッダー (event header)」も参照。

イベント・ヘッダー (event header). イベント・メッセージでは、イベントについての理由コードのイベント・タイプを識別するメッセージ・データ部分。

イベント・メッセージ (event message). MQSeries システムのネットワーク内でのインスツルメンテーション・イベントの起点に関連した情報 (イベントのカテゴリー、イベントを引き起こしたアプリケーションの名前、キュー・マネージャー統計など) を内容とする。

インスツルメンテーション・イベント (instrumentation event). MQSeries システムのネットワーク内でのキュー・マネージャーの操作を監視するために使用できる機能。MQSeries は、キュー・マネージャーのリソース定義、パフォーマンス条件、およびチャンネル条件をモニターするために、インスツルメンテーション・イベントを提供する。インスツルメンテーション・イベントは、イベントをシステム・オペレーターに表示する管理アプリケーション内の、ユーザー作成

報告メカニズムで使用することができる。さらに、インストゥルメンテーション・イベントを使用すれば、アプリケーションは、報告書をモニターし、適切なアラートを作成するために、他の管理ネットワークのエージェントとして動作することができる。

インストール可能サービス (installable services). MQSeries (UNIX システム用)、MQSeries for Compaq、MQSeries for OS/2 Warp、および MQSeries for Windows NT and Windows 2000 において、独立したコンポーネントとして提供される追加機能。各コンポーネントをインストールするかどうかはオプションであり、お客様作成のコンポーネント、または第三者のコンポーネントを代わりに使用できる。「許可サービス (*authorization service*)」、「名前サービス (*name service*)」、および「ユーザー ID サービス (*user identifier service*)」も参照。

インターネット・プロトコル (Internet Protocol (IP)). インターネット環境において、データの送信元から宛先までの経路を定めるために使用されるプロトコル。TCP や UDP といった他のプロトコル層がこの上に構成される基盤層。

永続動的キュー (permanent dynamic queue). 削除が明示的に要求された場合に限り、クローズ時に削除される動的なキュー。永続動的キューはキュー・マネージャーが失敗したときに回復されるので、持続メッセージを入れることができる。「一時動的キュー (*temporary dynamic queue*)」と対比。

応答側 (responder). 分散キューイングにおいて、他のシステムからのネットワーク接続要求に応答するプログラム。

応答先キュー (reply-to queue). MQPUT 呼び出しを出したプログラムが応答メッセージまたはレポート・メッセージの送信を望む、キューの名前。

応答メッセージ (reply message). 要求メッセージへの応答に使用されるメッセージのタイプ。

オブジェクト (object). MQSeries では、オブジェクトとは、キュー・マネージャー、キュー、プロセス定義、名前リスト、またはチャンネルを指す。

オブジェクト記述子 (object descriptor). 特定の MQSeries オブジェクトを識別するデータ構造のこと (MQOD)。オブジェクト記述子には、オブジェクトの名前およびオブジェクト・タイプが含まれる。

オブジェクト権限マネージャー (Object authority manager (OAM)). MQSeries (UNIX システム用)、MQSeries for Compaq、および MQSeries for Windows NT and Windows 2000 では、コマンドおよびオブジェクトの管理のためのデフォルトの許可サービス。OAM は、顧客提供のセキュリティー・サービスで置き換えたり、顧客提供のセキュリティー・サービスと組み合わせで実行したりすることができる。

オブジェクト・ハンドル (object handle). プログラムがその稼動対象である MQSeries オブジェクトにアクセスする際に用いる、ID またはトークン。

[力行]

解決パス (resolution path). アプリケーションが、MQOPEN 呼び出しに対する入力において別名またはリモート・キューを指定する時にオープンされる一組のキュー。

開始キュー (initiation queue). キュー・マネージャーがトリガー・メッセージを入れるローカル・キュー。

拡張プログラム間通信 (APPC). LU 6.2 アーキテクチャーと、製品へのその多様なインプリメンテーションを特徴とする汎用機能。

管理コマンド (administration command).

MQSeries オブジェクト (たとえば、キュー、プロセス、名前リスト) を管理するために使用する MQSeries のコマンド。

管理バッグ (administration bag). MQAI では

MQSeries を管理するために作成されるデータ・バッグの 1 つのタイプで、これによりメッセージ内のデータ項目の順序の変更、リストの作成、およびセレクターの検査を行うことができる。

完了コード (completion code). MQI 呼び出しがどのように終了したかを示す戻りコード。

基本マッピング・サポート (basic mapping support (BMS)). さまざまな端末に使用される制御文字に関係なく、入出力表示データをフォーマットし、複数ページの出力メッセージを発送する、CICS とアプリケーション・プログラム間のインターフェース。

キュー (queue). MQSeries オブジェクト。メッセージ・キューイング・アプリケーションは、キューにメッセージを置き、またキューからメッセージを得ることができる。キューは、キュー・マネージャーによって所有され、保守される。ローカル・キューには、処理を待っているメッセージのリストを含めることができる。その他のタイプのキューには、メッセージを含めることができない。それらはその他のキューを指す。または動的キューのモデルとして使用できる。

キューイング (queuing). 「メッセージ・キューイング (message queuing)」を参照。

キュー・サーバー (queue server). ローカル・キューについてのすべてのメッセージング操作をサポートする、NonStop プロセスのペア。

キュー・マネージャー (queue manager). アプリケーションにキュー・サービスを提供するシステム・プログラム。このプログラムは、キュー・マネージャーが所有するキュー上のメッセージにプログラムがアクセスできるようにする、アプリケーション・プログラミング・インターフェース

を提供する。「ローカル・キュー・マネージャー (local queue manager)」および「リモート・キュー・マネージャー (remote queue manager)」も参照。特定のキュー・マネージャーの属性を定義する MQSeries オブジェクト。

キュー・マネージャー・イベント (queue manager event).

次のことを示すイベント。

- キュー・マネージャーが使用するリソースに関連してエラー条件が発生した。たとえば、使用できないキューが原因で起きたエラー条件。
- キュー・マネージャーで重大な変更が行われた。たとえば、キュー・マネージャーが停止または開始した。

共通ランタイム環境 (Common Run-Time Environment (CRE)). システム・プログラマーとアプリケーション・プログラマーが、混合言語のプログラムを書くことができるようにしたサービスのセット。これらの共用のランタイム・サービスは、C、COBOL85、FORTRAN、Pascal、および TAL の各プログラムで使用できる。

許可検査 (authorization checks). たとえば、キューをオープンしたりキュー・マネージャーに接続するためにユーザーがオブジェクトに対して管理コマンドを発行しようとするときに実行されるセキュリティ検査。

許可検査 (authorization checks). MQSeries オブジェクトをオープンしようとするとき実行されるセキュリティ検査。

許可ファイル (authorization file). あるオブジェクト、あるオブジェクト・クラス、またはすべてのオブジェクト・クラスに関するセキュリティ定義を提供するファイル。

クライアント (client). ローカル・ユーザー・アプリケーションに、サーバーのキューイング・サービスへのアクセスを提供する実行時コンポーネント。アプリケーションが使用するキューは、サーバーに置かれている。「MQI クライアント (MQI client)」も参照。

クライアント接続チャンネル・タイプ (client connection channel type). MQI クライアントに関連付けられた MQI チャンネル定義のタイプ。

「サーバー接続チャンネル・タイプ (server connection channel type)」も参照。

クライアント・アプリケーション (client application). ワークステーションで実行されるアプリケーションであって、サーバーのキューイング・サービスにアクセスできるようにクライアントにリンクされているもの。

クラスター (cluster). 何らかの方法で論理的に関連付けられているキュー・マネージャーのネットワーク。

コード化文字セット ID (coded character set identifier (CCSID)). 各文字とそれに割り当てられたコード・ポイントの集合、すなわちコード化セットの名前。

コールバック (call back). 要求側のメッセージ・チャンネルが、まず送信側を呼び出し、次にクローズしてコールバックを待つことにより、送信側チャンネルからの転送を開始する。

構成ファイル (configuration file) (ini ファイルとも呼ばれる). ログ、通信、インストール可能サービスなどに関する構成情報が格納されるファイル。「スタンザ (stanza)」も参照。

顧客情報管理システム (Customer Information Control System (CICS)). ユーザー作成のアプリケーション・プログラムによる、データ・ファイルへの並行オンライン・アクセスが可能な IBM トランザクション管理システム。CICS には、データベースの構築、使用、および保守のための機能も含まれる。

コマンド (command). MQSeries では、キュー・マネージャーが実行できる命令。

コマンド処理プログラム (command processor). コマンドを処理する MQSeries のコンポーネント。

コマンド・サーバー (command server).

MQSeries のコンポーネントの 1 つで、システム・コマンド入力キューからコマンドを読み取り、そのコマンドを検査して、コマンド・プロセッサに有効なコマンドを渡すもの。

コマンド・バッグ (command bag). MQAI では MQSeries オブジェクトを管理するために作成されるバッグの 1 つのタイプだが、これによるメッセージ内のデータ項目の順序の変更やリストの作成は行えない。

コミット (commit). トランザクションを完了するときの動作で、これによりデータベースに対する変更内容が記録されて継続的なものになる。トランザクションがコミットされた後に、保護リソースが解放される。

コンテキスト (context). メッセージの起点についての情報。

コンテキスト・セキュリティ (context security). メッセージがメッセージの起点の詳細をメッセージ記述子内に持つように強制して、セキュリティを処理できるようにした方式。

[サ行]

サーバー (server). (1) MQSeries では、リモート・ワークステーションで実行されるクライアント・アプリケーションにキュー・サービスを提供するキュー・マネージャー。(2) クライアント/サーバーという 2 つのプログラム間の独特な情報フロー・モデルにおいて、情報に関する要求に回答するプログラム。「クライアント (client)」も参照。

サーバー接続チャンネル・タイプ (server connection channel type). キュー・マネージャーを実行するサーバーに関連した MQI チャンネル定義のタイプ。「クライアント接続チャンネル・タイプ (client connection channel type)」も参照。

サーバー・チャンネル (server channel). メッセージ・キューイングにおいて、要求側チャンネルに応答し、伝送キューからメッセージを除去したり、通信リンクを介してメッセージを要求側チャンネルに移動したりするチャンネル。

サービス間隔 (service interval). サービス間隔イベントの条件が満たされたかどうかを判別するときに、キュー・マネージャーが書き込みまたは読み取りと、それに続く読み取りとの間の経過時間を比較する対象となる時間間隔。キューのサービス間隔はキュー属性によって指定される。

サービス間隔イベント (service interval event). サービス間隔に関連したイベント。

再同期 (resynch). MQSeries において、メッセージ転送は再始動せずに、チャンネルを開始し、未確定の状況メッセージがあれば解決するように指示するオプション。

先入れ先出し法 (first-in-first-out (FIFO)). キューに最も長い間入っていた項目が、次に検索される項目となるような仕方で行われる、キューイングの技法。

作業単位 (unit of work). 2 つの整合点の間でアプリケーションにより実行される操作の回復可能な一連の操作。作業単位はトランザクションが開始する時、またはユーザーが要求した同期点の後で始まる。また、ユーザー要求の同期点、またはトランザクションの終了時に、作業単位は終わる。「リカバリー単位 (unit of recovery)」と対比。

作業論理単位 (logical unit of work (LUW)). 「作業単位 (unit of work)」を参照。

時間非依存メッセージング (time-independent messaging). 「非同期メッセージング (asynchronous messaging)」を参照。

システム制御コマンド (system control command). バッファー・プール、保管クラス、ページ・セットなどの、プラットフォームに特定のエンティティの操作に使用されるコマンド。

システム・コマンド入力キュー (system-command input queue). アプリケーション・プログラムが MQSeries コマンドを入れることができるローカル・キュー。コマンドはコマンド・サーバーによってキューから検索される。コマンド・サーバーはコマンドの妥当性検査を行い、実行のため、それらをコマンド・プロセッサに渡す。

システム・バッグ (system bag). データ・バッグの 1 つのタイプで、MQAI によって作成される。

持続メッセージ (persistent message). キュー・マネージャーの再始動の後も残るメッセージ。「非持続メッセージ (nonpersistent message)」と対比。

シャットダウン (shutdown). 「即時シャットダウン (immediate shutdown)」、「優先シャットダウン (preemptive shutdown)」、および「静止シャットダウン (quiesced shutdown)」を参照。

受信側チャンネル (receiver channel). メッセージ・キューイングにおいて、送信側チャンネルに回答するチャンネルで、通信リンクからメッセージを受け取り、それらをローカル・キューに入れる。

出力パラメーター (output parameter). MQI 呼び出しのパラメーターで、呼び出しが完了または失敗した場合にキュー・マネージャーがこのパラメーターに情報を戻す。

順次配布 (sequential delivery). MQSeries では、受信チャンネルがメッセージを保管する際、メッセージの順序を再設定できるように、順序番号を付けてメッセージを伝送する方式。これは、メッセージを一度だけ、正しい順序で送達しなければならない場合に必要となる方式である。

順序番号折り返し値 (sequential number wrap value). MQSeries では、通信リンクの両端が確実に、その現行メッセージの順序番号を同時にリセットするための方式。順序番号を付けてメッセージを送るならば、受信チャンネルは、メッセージを保管する際、メッセージの順序を確実に再設定することができる。

状況サーバー (Status Server). ローカル・キュー以外のすべてのオブジェクトについての状況情報をサポートする。デフォルトの状況サーバーでは、キュー・マネージャーについてのチャンネル状況情報も扱う。

症状ストリング (symptom string). IBM ソフトウェア・サポート・データベースの検索用に設計された、構造化形式で表示される診断情報。

スタンザ (stanza). 構成ファイルの中の 1 グループの行であって、キュー・マネージャー、クライアント、またはチャンネルの動作を修正するパラメーターに値を割り当てるもの。システム上の MQSeries では、構成 (ini) ファイルには多くのスタンザが含まれる。

スレッド (thread). MQSeries では、1 つのオペレーティング・システム・プラットフォームで利用できる並行実行の最低レベル。

制御されたシャットダウン (controlled shutdown). 「静的シャットダウン (*quiesced shutdown*)」を参照。

静止 (quiescing). MQSeries において、停止される前のキュー・マネージャーの状態。この状態では、プログラムは処理を終了することはできるが、新規プログラムを開始することはできない。

静的シャットダウン (quiesced shutdown). MQSeries において、接続しているすべてのアプリケーションが切断できるようにするキュー・マネージャーのシャットダウン。「即時シャットダウン (*immediate shutdown*)」および「優先シャットダウン (*preemptive shutdown*)」と対比。

セキュリティ有効化インターフェース (security enabling interface (SEI)). 許可の検査、ユーザー ID の提供、または認証を行う顧客作成またはベンダー作成のプログラムが従う必要がある MQSeries インターフェース。MQSeries フレームワークの一部。

接続 (connect). キュー・マネージャーの接続ハンドルを提供することであって、その接続ハンドルはそれ以降の MQI 呼び出しでアプリケーションによって使用される。接続は、MQCONN 呼び出しによって行われる、または MQOPEN 呼び出しによって自動的に行われる。

接続ハンドル (connection handle). プログラムが、接続先のキュー・マネージャーにアクセスするための ID またはトークン。

送信側チャンネル (sender channel). メッセージ・キューイングにおいて、転送を開始し、伝送キューからメッセージを除去したり、通信リンクを介してメッセージを受信側または要求側チャンネルに移動したりするチャンネル。

送達不能キュー (dead-letter queue (DLQ)). キュー・マネージャーまたはアプリケーションがそれらの正しい宛先に送達できないようなメッセージの送信先となるキュー。

送達不能キュー・ハンドラー (dead-letter queue handler). MQSeries 提供のユーティリティーであって、ユーザー作成のルール・テーブルに従って送達不能キュー (DLQ) を監視し、キュー上のメッセージを処理する。

即時シャットダウン (immediate shutdown). MQSeries において、アプリケーションが切断するまで待つことのないキュー・マネージャーのシャットダウン。現行の MQI 呼び出しは処理を完了できるが、即時シャットダウンが要求された後の新規の MQI 呼び出しは失敗する。「静止シャットダウン (*quiesced shutdown*)」および「優先シャットダウン (*preemptive shutdown*)」と対比。

属性 (attribute). MQSeries オブジェクトの特性を定義する、一組の特性の 1 つ。

[タ行]

第 1 障害サポート・テクノロジー (First Failure Support Technology (FFST)). ソフトウェア問題を検出して報告するために、MQSeries (UNIX システム用)、MQSeries for OS/2 Warp、MQSeries for Windows NT and Windows 2000、および MQSeries (AS/400 版) で使用される。

代替ユーザー・セキュリティー (alternate user security). MQSeries オブジェクトをオープンするなどの、1 つのユーザー ID の権限を別のユーザー ID が使用することを可能にするセキュリティー機能。

単一フェーズ・コミット (single-phase commit). 別のリソース・マネージャーによって制御されているリソースに対して行った更新との間での調整を行わなくても、プログラムがキューに対して更新をコミットできる方式。「2 フェーズ・コミット (*two-phase commit*)」と対比。

単一フェーズ・バックアウト(single-phase back out). まだ進行中のアクションが終了しないようにし、そのアクションの一部をなすすべての変更が全部取り消されるようにする方式。

チェックポイント (checkpoint). 重要な情報がログに書き込まれる時点。「同期点 (*syncpoint*)」と対比。MQSeries (UNIX システム用) では、ログに記述されているデータ・レコードがキューのデータ・レコードと同じになる時点。チェックポイントは自動的に生成され、システム再始動処理時に使用される。

蓄積交換 (store and forward). パケット、メッセージ、またはフレームが、その宛先に再送される前に、データ・ネットワーク内に一時的に保管されること。

チャンネル (channel). 「メッセージ・チャンネル (*message channel*)」を参照。

チャンネル制御機能 (channel control function (CCF)). 伝送キューから通信リンクへ、および通信リンクからローカル・キューへメッセージを移動するプログラムで、オペレーター・パネル・インターフェースと連携してチャンネルのセットアップならびに制御を可能にするもの。

チャンネル定義ファイル (channel definition file (CDF)). MQSeries では、伝送キューと通信リンクを関連付ける通信チャンネル定義を含んでいるファイル。

チャンネル・イベント (channel event). チャンネル・インスタンスが利用可能になったか利用不可になったかを示すイベント。チャンネル・イベントは、チャンネルの両端のキュー・マネージャーで生成される。

データグラム (datagram). MQSeries がサポートする最も単純なメッセージ。このタイプのメッセージは応答を必要としない。

データ項目 (data item). MQAI では、データ・バッグ内に含まれる項目。整数項目または文字ストリング項目、およびユーザー項目またはシステム項目。

データ変換インターフェース (data conversion interface (DCI)). 異なる機械エンコードと CCSID の間でのアプリケーション・データの変換を行う顧客作成またはベンダー作成のプログラムが従う必要がある MQSeries インターフェース。MQSeries フレームワークの一部。

データ・バッグ (data bag). MQAI では、オブジェクトのプロパティー (またはパラメーター) を処理することができるバッグ。

デフォルト・オブジェクト (default object). すべての属性が定義されているオブジェクト (たとえば、キュー) の定義。ユーザーがオブジェクトを定義したときにそのオブジェクトに指定可能な

属性を全部は指定しなかった場合は、キュー・マネージャーが、指定されていない属性の代わりにデフォルト属性を使用する。

伝送キュー (transmission queue). リモート・キュー・マネージャーを宛先とする準備済みメッセージが一時的に保管されるローカル・キュー。

伝送制御プロトコル (Transmission Control Protocol (TCP)). TCP/IP プロトコル組の一部。パケット交換通信ネットワークにおけるホスト間のホスト対ホストのプロトコル。TCP は、コネクション型データ・ストリーム送達を備えている。送達は信頼性が高く、順序が維持される。

伝送制御プロトコル / インターネット・プロトコル (Transmission Control Protocol/Internet Protocol (TCP/IP)). ローカルおよび広域ネットワークの両方で、ピアツーピア接続機能をサポートする通信プロトコルの組。

伝送プログラム (transmission program). 「メッセージ・チャネル・エージェント (*message channel agent*)」を参照。

同期点 (syncpoint). トランザクションの保護リソースが整合される、トランザクションの処理中の中間点また終点。同期点では、リソースへの変更内容を安全にコミットしたり、または直前の同期点にそれらをバックアウトしたりできる。

同期メッセージング (synchronous messaging). プログラムがメッセージをメッセージ・キューに書き込む、プログラム間の通信の方式。同期メッセージングでは、送信側プログラムは、そのメッセージに対する応答を待ってから、それ自身の処理を再開する。「非同期メッセージング (*asynchronous messaging*)」と対比。

動的キュー (dynamic queue). プログラムがモデル・キュー・オブジェクトをオープンするとき作成されるローカル・キュー。「永続動的キュー (*permanent dynamic queue*)」および「一時動的キュー (*temporary dynamic queue*)」も参照。

トリガー処理 (triggering). MQSeries では、キュー上の事前決定条件が満たされると、キュー・マネージャーが自動的にアプリケーションを開始できるようにする機能。

トリガー・イベント (trigger event). キュー・マネージャーに、開始キュー上にトリガー・メッセージを作成させるイベント (たとえば、キューに届いているメッセージなど)。

トリガー・メッセージ (trigger message). トリガー・モニターが始動するプログラムに関する情報が入られるメッセージ。

トリガー・モニター (trigger monitor). 連続して実行されるアプリケーションで、1 つまたは複数の開始キューのためにサービスを行うもの。トリガー・メッセージが開始キューに到達すると、トリガー・モニターがそのメッセージを検索する。トリガー・モニターは、トリガー・イベントが発生したキューをサービスするプロセスを開始するために、トリガー・メッセージの中の情報を使用する。

トリガー・モニター・インターフェース (trigger monitor interface (TMI)). 顧客作成またはベンダー作成のトリガー・モニター・プログラムが従う必要がある MQSeries インターフェース。MQSeries フレームワークの一部。

[ナ行]

名前サービス・インターフェース (name service interface (NSI)). キュー名の所有権を解決する顧客作成またはベンダー作成のプログラムが従う必要がある MQSeries インターフェース。MQSeries フレームワークの一部。

名前リスト (namelist). 名前、たとえば、キュー名のリストが入っている MQSeries オブジェクト。

入出力パラメーター (input/output parameter). MQI 呼び出しのパラメーターで、呼び出しを行う際ユーザーはこのパラメーターに情報を提供し、MQI 呼び出しが完了または失敗すると、キュー・マネージャーがこのパラメーター内の情報を変更する。

入力パラメーター (input parameter). MQI 呼び出しのパラメーターで、呼び出しを行う際ユーザーはこのパラメーターに情報を提供する。

ヌル文字 (null character). X'00' で表される文字。

[八行]

バッグ (bag). 「データ・バッグ (*data bag*)」を参照。

バックアウト (back out). 現行のリカバリー単位または作業単位中に行われたすべての変更内容を取り消す操作。この操作の完了後、新規のリカバリー単位または作業単位が開始される。

パフォーマンス・イベント (performance event). 限界条件が起きたことを示すイベントのカテゴリー。

パフォーマンス・トレース (performance trace). トレース・データがパフォーマンス分析およびチューニングに使用される、MQSeries トレース・オプション。

ハンドル (handle). 「接続ハンドル (*connection handle*)」および「オブジェクト・ハンドル (*object handle*)」を参照。

非持続メッセージ (nonpersistent message). キュー・マネージャーを再始動すると消えるメッセージ。「持続メッセージ (*persistent message*)」と対比。

非同期メッセージング (asynchronous messaging). プログラムがメッセージをメッセージ・キューに書き込む、プログラム間の通信の

方式。非同期メッセージングでは、送信側プログラムは、そのメッセージへの応答を待つことなく、送信側の処理を進行させる。「同期メッセージング (*synchronous messaging*)」と対比。

ブラウズ (browse). メッセージ・キューイングにおいて、MQGET 呼び出しを使用してキューからメッセージを除去せずにメッセージをコピーすること。「読み取り (*get*)」も参照。

ブラウズ・カーソル (browse cursor). メッセージ・キューイングにおいて、次の順序のメッセージを識別するためにキューをブラウズする際に使用される標識。

プラットフォーム (platform). MQSeries では、キュー・マネージャーがそのもとで実行されているオペレーティング・システムのこと。

フレームワーク (Framework). MQSeries では、お客様またはベンダーが、MQSeries プロダクトの提供する一部の機能を拡張したり置き換えたりするプログラムを作成できるようにするプログラミング・インターフェースの集合。インターフェースは次のとおり。

- MQSeries データ変換インターフェース (DCI)
- MQSeries メッセージ・チャンネル・インターフェース (MCI)
- MQSeries 名前サービス・インターフェース (NSI)
- MQSeries セキュリティ有効化インターフェース (SEI)
- MQSeries トリガー・モニター・インターフェース (TMI)

プログラム一時修正 (program temporary fix (PTF)). プログラムの現行版の変更が加えられていないリリースの欠陥の結果であると、IBM 技術員が診断した問題の解決またはバイパス。

プログラム式コマンド形式 (programmable command format (PCF)). 次の場合に使用される MQSeries メッセージのタイプ。

- ユーザー管理アプリケーションが、指定されたキュー・マネージャーのシステム・コマンド入力キューに PCF コマンドを書き込む。
- ユーザー管理アプリケーションが、指定されたキュー・マネージャーから PCF コマンドの結果を読み取る。
- キュー・マネージャーが、イベント発生の通知として使用する。

「MQSC」と対比。

プログラム診断依頼書 (authorized program analysis report (APAR)). 未変更の現行リリースのプログラムに生じた障害のために引き起こされた問題のレポート。

プロセス定義オブジェクト (process definition object). MQSeries アプリケーションの定義を含む MQSeries オブジェクト。たとえば、キュー・マネージャーはトリガー・メッセージを用いた作業を行うときにこの定義を使用する。

分散アプリケーション (distributed application). メッセージ・キューイングにおいて、それぞれが異なるキュー・マネージャーに接続できる一方で、全体として単一のアプリケーションを構成するアプリケーション・プログラムの集合。

分散キュー管理 (distributed queue management). メッセージ・キューイングでは、他のシステムのキュー・マネージャーへのメッセージ・チャンネルの設定と制御。

分散コンピューター環境 (Distributed Computing Environment (DCE)). 基本サービスを提供し、分散アプリケーションの開発を容易にするミドルウェア。DCE は Open Software Foundation (OSF) によって定義されている。

別名キュー・オブジェクト (alias queue object). MQSeries オブジェクトの 1 つで、ローカル・キュー・マネージャーに対して定義されている基本キューの別名を名前として持つもの。アプリケーションまたはキュー・マネージャーが

別名キューを使用するときには、別名が解決され、要求した操作が関連する基本キューに対して実行される。

保留イベント (pending event). CICS アダプターからの接続要求の結果として発生した未スケジュールのイベント。

[マ行]

未配布メッセージ・キュー (undelivered-message queue). 「送達不能キュー (dead-letter queue)」を参照。

メッセージ (message). メッセージ・キューイング・アプリケーションにおいて、プログラム間で送信される連絡。「持続メッセージ (persistent message)」および「非持続メッセージ (nonpersistent message)」も参照。システム・プログラミングにおいて、端末オペレーターまたはシステム管理者に宛てられた情報。

メッセージ記述子 (message descriptor). メッセージの形式および表現を記述する制御情報で、MQSeries メッセージの一部として伝送されるもの。メッセージ記述子の形式は、MQMD 構造によって定義される。

メッセージ順序番号付け (message sequence numbering). 通信リンク上での伝送中に、メッセージに固有の番号を与えるプログラミング技法。これにより、すべてのメッセージが受信されたかどうかの検査、メッセージを元の順序でキュー内に配置すること、および重複メッセージの廃棄のための受信処理が可能になる。

メッセージ優先順位 (message priority). MQSeries において、キュー上のメッセージが検索される順序およびトリガー・イベントが生成されるかどうかに影響を与え得るメッセージの属性。

メッセージング (messaging). 「同期メッセージング (*synchronous messaging*)」および「非同期メッセージング (*asynchronous messaging*)」を参照。

メッセージ・キュー (message queue). キュー (*queue*) の同義語。

メッセージ・キューイング (message queuing). アプリケーション内の各プログラムが、他のプログラムとの通信を、キューにメッセージを入れることを行うプログラミング技法。

メッセージ・キュー管理 (message queue management). MQSeries for Compaq NSK V2.2 のメッセージ・キュー管理 (Message Queue Management (MQM)) 機能では、PCF コマンド形式および制御コマンドを使用する。MQM は、端末管理プロセス (Terminal Control Process (TCP)) の制御下で PATHWAY SCOBOL リクエスターとして実行され、C 言語 API を呼び出して PCF コマンドを実行する MQM SERVERCLASS サーバーを使用する。各キュー・マネージャーはそれぞれ独自の PATHWAY 構成のもとで制御されるため、システム上に構成されたそれぞれのキュー・マネージャーごとに別々の MQM のインスタンスがある。したがって、MQM は、それが属するキュー・マネージャーの管理に限定される。

メッセージ・キュー・インターフェース (message queue interface (MQI)). MQSeries キュー・マネージャーにより提供されるプログラミング・インターフェース。このプログラミング・インターフェースによって、アプリケーション・プログラムはメッセージ・キュー・サービスにアクセスすることができる。

メッセージ・チャンネル (message channel). 分散メッセージ・キューイングにおいて、メッセージをあるキュー・マネージャーから別のキュー・マネージャーに移動するメカニズム。メッセージ・チャンネルは、2 つのメッセージ・チャンネル・エージェント (一方は送信側で他方は受信側) お

よび 1 つの通信リンクで構成される。「MQI チャンネル (*MQI channel*) 」と対比。

メッセージ・チャンネル・インターフェース (message channel interface (MCI)). MQSeries キュー・マネージャーと他のメッセージング・システムとの間でメッセージを送信する顧客作成またはベンダー作成のプログラムが従う必要がある MQSeries インターフェース。MQSeries フレームワークの一部。

メッセージ・チャンネル・エージェント (message channel agent (MCA)). 準備済みメッセージを、伝送キューから通信リンクに、あるいは通信リンクから宛先キューに伝送するプログラム。「メッセージ・キュー・インターフェース (*message queue interface*)」も参照。

モデル・キュー・オブジェクト (model queue object). プログラムが動的キューを作成するときにテンプレートとして動作するキュー属性のセット。

戻りコード (return codes). 完了コードおよび理由コードの集合的な名前。

[ヤ行]

ユーザー・バッグ (user bag). MQAI ではデータ・バッグの 1 つのタイプで、ユーザーによって作成される。

ユーティリティ (utility). MQSeries では、システム・オペレーターまたはシステム管理者に、MQSeries コマンドによって提供される機能に加えて、さらに機能を提供する一連のプログラム。複数の機能を呼び出すユーティリティもある。

優先シャットダウン (preemptive shutdown). MQSeries において、接続しているアプリケーションが切断するまで待つことも、現行 MQI 呼び出しが完了するまで待つこともないキュー・マネージャーのシャットダウン。「即時シャットダウン

ン (*immediate shutdown*)」および「静止シャットダウン (*quiesced shutdown*)」と対比。

要求側チャネル (requester channel). メッセージ・キューイングにおいて、送信側チャネルがリモートから開始できるチャネル。要求側チャネルは、通信リンク全体の送信側チャネルからメッセージを受け入れ、そのメッセージをメッセージ内の指定されたローカル・キューに入れる。「サーバー・チャネル (*server channel*)」も参照。

要求メッセージ (request message). 別のプログラムからの応答を要求するために使用されるメッセージのタイプ。

[ラ行]

リカバリー単位 (unit of recovery). 単一のリソース・マネージャー内のリカバリー可能な一連の操作。「作業単位 (*unit of work*)」と対比。

リスナー (listener). MQSeries の分散キューイングにおいて、着信側ネットワーク接続についての情報をモニターするプログラム。

リニア・ロギング (linear logging). MQSeries (UNIX システム用)、MQSeries for OS/2 Warp、および MQSeries for Windows NT and Windows 2000において、再始動データを一連のファイルに保持するプロセス。新規ファイルが必要に応じて一連のファイルに追加される。データが書き込まれるスペースは、キュー・マネージャーが再始動するまで利用されない。「循環ロギング (*circular logging*)」と対比。

リポジトリ (repository). クラスターのメンバーであるキュー・マネージャーについての情報のコレクション。この情報には、キュー・マネージャー名、それらのロケーション、それらのチャネル、それらがホストとなるキューなどが含まれる。

リモート・キュー (remote queue). リモート・キュー・マネージャーに属するキュー。プログラ

ムはリモート・キューにメッセージを入れることはできるが、リモート・キューからメッセージを得ることはできない。「ローカル・キュー (*local queue*)」と対比。

リモート・キューイング (remote queuing). メッセージ・キューイングにおいて、アプリケーションが、他のキュー・マネージャーに属しているキューにメッセージを入れることができるようにするサービスの提供。

リモート・キューのローカル定義 (local definition of a remote queue). ローカル・キュー・マネージャーに属する MQSeries オブジェクト。このオブジェクトは、別のキュー・マネージャーが所有するキューの属性を定義する。また、キュー・マネージャーの別名付けおよび応答先キューの別名付けに使用される。

リモート・キュー・オブジェクト (remote queue object). 「リモート・キューのローカル定義 (*local definition of a remote queue*)」を参照。

リモート・キュー・マネージャー (remote queue manager). あるキュー・マネージャーが、あるプログラムが接続されるキュー・マネージャーではない場合、そのプログラムにとってそのキュー・マネージャーはリモートになる。

理由コード (reason code). MQI 呼び出しの失敗または部分的な成功の理由を記述する戻りコード。

ルール・テーブル (rules table). 送達不能キュー・ハンドラーが DLQ 上のメッセージに適用する 1 つまたは複数の規則が入った制御ファイル。

レポート・メッセージ (report message). 別のメッセージに関する情報を与えるメッセージのタイプ。メッセージが送達された、その宛先に到着した、その期限が切れた、何らかの理由でメッセージを処理できなかったなどということを、レポート・メッセージは示すことができる。

ローカル定義 (local definition). ローカル・キュー・マネージャーに属する MQSeries オブジェクト。

ローカル・キュー (local queue). ローカル・キュー・マネージャーに属するキュー。ローカル・キューには、処理を待っているメッセージのリストが含まれている。「リモート・キュー (*remote queue*)」と対比。

ローカル・キュー・マネージャー (local queue manager). プログラムが接続されるキュー・マネージャーであって、メッセージ・キューイング・サービスをプログラムに提供する。プログラムが接続されていないキュー・マネージャーは、それらがそのプログラムと同じシステム上で実行されているとしても、リモート・キュー・マネージャーと呼ばれる。

ロールバック (rollback). 「バックアウト (*back out*)」と同義。

ログ (log). MQSeries では、キュー・マネージャーがメッセージを受信、伝送、および送達するときに行う作業を記録する。これにより、障害時にキュー・マネージャーは回復できる。

[数字]

2 フェーズ・コミット (two-phase commit). 単一のトランザクションによって複数のリソース・マネージャーが使用されるときに、リカバリー可能リソースへの変更内容を調整するためのプロトコル。「単一フェーズ・コミット (*single-phase commit*)」と対比。

A

APAR. プログラム診断依頼書 (Authorized Program Analysis Report)。

APPC. 拡張プログラム間通信 (Advanced Program to Program Communication)。

B

BMS. 基本マッピング・サポート (Basic Mapping Support)。

C

CCF. チャネル制御機能 (channel control function)。

CCSID. コード化文字セット ID (Coded character set identifier)。

CDF. チャネル定義ファイル (Channel definition file)。

CICS. 顧客情報管理システム (Customer Information Control System)。

CRE. 共通ランタイム環境 (Common Run-Time Environment)。

D

DCE. 分散コンピューター環境 (Distributed Computing Environment)。

DCI. データ変換インターフェース (Data conversion interface)。

DLQ (送達不能キュー (dead-letter queue)). キュー・マネージャーまたはアプリケーションがそれらの正しい宛先に送達できないようなメッセージの送信先となるキュー。

E

EC. EC はキュー・マネージャーにおける補助的な制御プロセスで、エージェントの設定の機能を果たす。

EC Boss. Execution Controller Boss のことで、キュー・マネージャーにおけるメインの制御プロセス。

EMS. イベント・モニター・システム (Event Monitoring System)。

F

FFST. 第 1 障害サポート・テクノロジー (First Failure Support Technology)。

FIFO (先入れ先出し法 (first-in-first-out)). キューに最も長い間入っていた項目が、次に検索される項目となるような仕方で検索が行われる、キューイングの技法。

G

get. メッセージ・キューイングにおいて、MQGET 呼び出しを使用してキューからメッセージを除去すること。「ブラウズ (*browse*)」も参照。

I

ICE. Intersystem Communications Environment のことで、Compaq ベースのソフトウェア・プロダクトのファミリー。これにより Compaq コンピューター上のさまざまなアプリケーションにアクセスすることができる。

ini ファイル (ini file). 「構成ファイル (*configuration file*)」を参照。

IP. インターネット・プロトコル (Internet Protocol)

M

MCA (メッセージ・チャンネル・エージェント (message channel agent)). 準備済みメッセージを、伝送キューから通信リンクに、あるいは通信リンクから宛先キューに伝送するプログラム。

MCI (メッセージ・チャンネル・インターフェース (message channel interface)). MQSeries キュー・マネージャーと他のメッセージング・システムとの間でメッセージを送信する顧客作成またはベンダー作成のプログラムが従う必要がある MQSeries インターフェース。MQSeries フレームワークの一部。

MQAI. MQSeries 管理インターフェース (MQSeries Administration Interface)。

MQI (message queuing interface). MQSeries キュー・マネージャーにより提供されるプログラミング・インターフェース。このプログラミング・インターフェースによって、アプリケーション・プログラムはメッセージ・キュー・サービスにアクセスすることができる。

MQI クライアント (MQI client). キュー・マネージャー全体をインストールしなくてもシステムにインストールできる MQSeries プロダクトの一部。MQI クライアントは、アプリケーションから MQI 呼び出しを受け入れて、サーバー・システム上のキュー・マネージャーと通信する。

MQI サーバー (MQI server). MQI サーバーとは、キューイング・サービスを 1 つまたは複数のクライアントに提供するキュー・マネージャーのこと。たとえばキューなどの MQSeries オブジェクトはすべて、キュー・マネージャー・システム上、つまり MQI サーバー・マシン上にのみ存在する。サーバーは、通常のローカル MQI アプリケーションもサポートする。

MQI チャンネル (MQI channel). MQI クライアントをサーバー・システム上のキュー・マネージャーに接続して、MQI 呼び出しと応答のみを両方向に転送するもの。「メッセージ・チャンネル (*message channel*)」と対比。

MQM. メッセージ・キュー管理 (Message Queue Management)。

MQSC. MQSeries コマンド (MQSeries command)。

MQSeries. メッセージ・キュー・サービスを提供する、IBM ライセンス・プログラムのファミリー。

MQSeries 管理インターフェース (MQSeries Administration Interface (MQAI)). MQSeries に対するプログラミング・インターフェース。

MQSeries コマンド (MQSeries commands (MQSC)). すべてのプラットフォームで同じ形を持ち、MQSeries オブジェクトの操作に使用される、人が読めるコマンド。「プログラム式コマンド形式 (*programmable command format (PCF)*)」と対比。

N

NetBIOS. ネットワーク BIOS (Network Basic Input/Output System)。IBM トークンリング・ネットワークに接続される IBM パーソナル・コンピュータ上で使用されるアプリケーション・プログラム用のオペレーティング・システム・インターフェース。

NSI. 名前サービス・インターフェース (Name service interface)。

P

PCF. プログラム式コマンド形式 (Programmable command format)。

PCF コマンド (PCF command). 「プログラム式コマンド形式 (*programmable command format*)」を参照。

PING (ping). 分散キューイングにおいて、テスト・メッセージの交換を使用することにより、メッセージ・チャンネルが機能しているかどうかを確認する診断援助機能。

PTF. プログラム一時修正 (program temporary fix)。

R

RBA. 相対バイト・アドレス (relative byte address)。

S

SEI. セキュリティ有効化インターフェース (Security enabling interface)。

SIT. システム初期設定テーブル (system initialization table)。

SNA. システム・ネットワーク体系 (System Network Architecture)。

T

TACL. タンデム拡張コマンド言語。

TCP. 伝送制御プロトコル (Transmission Control Protocol)。

TCP/IP. 伝送制御プロトコル / インターネット・プロトコル (TCP/IP)

TMF. トランザクション管理機能 (Transaction Management Facility)。

TMI. トリガー・モニター・インターフェース (Trigger monitor interface)。

TM/MP. NonStop Transaction Manager/MP。

transid. 「トランザクション ID (*transaction identifier*)」を参照。

U

UDP. User Datagram Protocol。

User Datagram Protocol (UDP). TCP/IP プロトコル組の一部。インターネット・プロトコル (IP) 層の上に直接構築されるパケット・レベルのプロトコル。UDP はコネクションレスで、TCP

と比べると信頼性は劣る。TCP/IP ホスト・システム間のアプリケーション対アプリケーションのプログラムで使用される。

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

アイドル状態のエージェント・プロセス 67
アクション・キーワード 211
アプリケーション
 起動される 441
 キュー内の持続メッセージのブラウザの影響 291
 コンパイルおよびバインド 444
 作成 437
 時間に依存しない 8
 実行 446
 設計の考慮事項 269
 チューニング 283
 データ 8
 トラステッド 295
 プログラミング・エラーの例 263
 並行トランザクションの管理 308
FASTPATH 結合
 障害からのリカバリー 310
FASTPATH の使用の制限 297
FASTPATH バインディング 445
guardian 444
MQI 管理サポート 117
OpenTMF の使用 307
STANDARD 結合
 障害からのリカバリー 310
STANDARD バインディング 445
アプリケーションのコンパイルおよびバインド 444
一時キュー 10

イベント
 インストゥルメンテーション
 解説 221
 使用する理由 222
 説明 221
 タイプ 222
 メッセージ 224
 有効化 223
 キュー 223
 タイプ 222
 チャンネル 222
 トリガー 222
 EMS 507
 MQSeries for Compaq NSK でのサポート 497
イベントのタイプ 222
イベント・キュー 18
イベント・ドリブン処理 8
イベント・メッセージの形式 498
インストゥルメンテーション・イベント
 説明 221
 タイプ 222
 メッセージ 224
 目的 222
 有効化 223
 MQSeries for Compaq NSK によるサポート 497
インストール可能サービス
 オブジェクト権限マネージャー (OAM) 179
 使用不可 183
 許可サービス 506
 名前サービス 506
エージェント・プロセス 34, 67
永続キュー 10
エスケープ PCF 40, 152
エラー・メッセージ 121
エラー・ログ 275
 確立前に生じるエラー 276
 サブボリューム 81

エラー・ログ 275 (続き)
 例 277
応答先キュー 18
応答先キュー別名 173
大文字小文字の区別
 制御コマンド 38
 MQSC コマンド 120
オブジェクト
 アクセス 177
 キュー 15
 キュー・マネージャー
 MQI 呼び出し 14
システム
 デフォルト 20, 405
 復元 86
 タイプ 11
 デフォルト
 属性 133
 復元 86
 トリガー操作のための 146
 名前 119
 プロセス定義 18
 命名規則 324
 リモート管理 157
オブジェクト名の変換 83
オブジェクト・カタログ 309
オブジェクト・タッチ・ファイル 309
オペレーター・コマンド、応答なし 264
オペレーター・メッセージ 277

[カ行]

解決、チャンネルの 114
開始
 キュー・マネージャー 86
 チャンネル 112, 164
 トレース 95
開始キュー
 説明 16

- 開始キュー (続き)
 - チャンネル・イニシエーターの変更 42
 - 定義 148
- 開始と停止イベント 497
- 確認、MQSC コマンドの 128
- 可用性
 - 構成 318
 - 定義 300
- 可用性に関する構成 318
- 環境変数 409
- 監査証跡のサイズ、TM/MP 235
- 幹線図、見方 325
- 管理
 - 許可 197
 - コマンド・セット 37
 - 制御コマンド 37
 - プログラム式コマンド形式 (PCF) コマンド 39
 - MQSeries コマンド (MQSC) 38
 - リモート 160
 - オブジェクト 157
 - チャンネル 161
 - 伝送キュー 161
 - ローカル 117
- 起動されたアプリケーション
 - USERDATA を渡す 425
- 起動されるアプリケーション 441
- 基本データベース 309
- キュー
 - アプリケーション、トリガー操作のための定義 146
 - 一時 10
 - イベント 18, 223
 - 応答先 18, 172
 - オブジェクト
 - 別名 15
 - モデル 15
 - リモート 15
 - ローカル 15
 - 開始
 - 定義 148
 - トリガー・メッセージ 16
 - 許可 189
 - クラスター伝送キュー 289
- キュー (続き)
 - 項目数の測定 293
 - コマンド 18
 - サイズ 132
 - 事前定義 10
 - 説明 10
 - 送達不能 17, 58
 - 属性 14, 434
 - 属性、変更 135
 - 定義 14
 - 伝送 17
 - 作成 171
 - 定義 162
 - デフォルト 58, 171
 - リモート管理 161
 - 動的 10
 - 取り扱い 131
 - 物理サイズ 132
 - ブラウズ 138
 - 分散、不正出力 273
 - 別名 15
 - 別名、取り扱い 142
 - 未配布メッセージ 17, 58
 - モデル 15
 - 定義 145
 - 取り扱い 145
 - リモート 13, 15
 - キュー・マネージャー別名 172
 - 作成 168
 - 取り扱い 172
 - ローカル 13, 15
 - クリア 138
 - コピー 134
 - 削除 138
 - 定義 131
 - MQI アプリケーションのための 117
 - キュー項目数 134
 - キューの属性の変更 135
 - キュー・オーバーフロー・ファイル 287, 309
 - 再配置 289
 - キュー・サーバー
 - キューの測定カウンターの保守 293
- キュー・サーバー (続き)
 - キューの分散 288
 - キュー・サーバー始動時のデータ構造の構築 292
 - 障害からのリカバリー 310
 - 除去 79
 - 説明 31
 - チューニング 287
 - データ構造およびチェーンをメモリー内にロックする 292
 - デフォルトではない場合の追加と除去 77
 - デフォルトの名前 76
 - 非持続データの管理 305
 - 非持続メッセージのチェックポイント 293
 - プロセス
 - 追加 45
 - メッセージをメモリーに保管する 291
 - リカバリーおよび再始動 239
 - CPU 分散 288
 - PARAMS 411
 - キュー・ファイル 287, 309
 - 区画 290
 - 再配置 289
 - キュー・マネージャー
 - イベント 222
 - エラー・ログ・サブボリューム 81
 - オブジェクト
 - MQI 呼び出し 14
 - オブジェクト権限マネージャー
 - 使用不可 183
 - 説明 179
 - 開始 86
 - 数 58
 - 可用性 301
 - 許可 189
 - 構成可能なプロパティ 61
 - 構成ファイル
 - 内容 245
 - バックアップ 58
 - コマンド・サーバー 154
 - 固有の名前 58
 - コンポーネント 32

- キュー・マネージャー (続き)
 - 再始動 88
 - 削除 89, 347
 - 作成
 - ガイドライン 58
 - デフォルト 83
 - crtmqm コマンド 342
 - シャットダウン
 - 制御された 87
 - 静止 87
 - 障害からのリカバリー 310
 - 除去、手動 418
 - 説明 13
 - 即時シャットダウン 87
 - 属性 435
 - チャンネル同期サブボリューム 82
 - データ・ファイル・サブボリューム 80
 - 停止 87
 - 手動 417
 - ディレクトリー 189
 - デフォルト 58
 - 誤った削除 344
 - 誤った変更 89
 - 変更 89
 - プロセス 27
 - 別名、リモート・キューの 172
 - ホーム・ボリューム 62
 - メッセージ・キュー・サブボリューム 81
 - モニター 222
 - 優先シャットダウン 87, 88
 - リモート管理 157
 - ローカル管理 117
 - endmqm コマンド 365
 - FFST サブボリューム 80
 - MVS/ESA での 166
 - runmqsc に指定 123
 - キュー・マネージャーの起動コマンド 390
 - キュー・マネージャー・サーバー
 - 障害からのリカバリー 310
 - キュー・マネージャー・プロセスの優先順位 68
 - キュー・モード、runmqsc の 165
- 強制停止、チャンネルの (即時停止) 112
 - 行の最大長、MQSC コマンドの 126
 - 共用メモリー・セグメント 32
 - 共用リソース・ライブラリー 30
 - 許可
 - 管理 197
 - ファイル 201
 - ユーザー・グループ 182
 - リスト 184
 - dspmqaout コマンド 187
 - MQI 193
 - setmqaut コマンド 187
 - 許可サービス 24
 - インターフェース 506
 - 禁止イベント 497
 - 区画化
 - キュー・ファイル 290
 - クライアント 22, 495
 - クライアント・チャンネル定義 309
 - クラスター
 - キュー・マネージャーワークロード出口 23
 - キュー・マネージャーの 11
 - 説明 159
 - ネットワーク可用性の向上 318
 - リモート・キューイング 157
 - ワークロード管理出口 459
 - クラスター伝送キュー 289
 - クリア、ローカル・キューの 138
 - グローバル作業単位
 - 説明 25
 - 権限
 - イベント 497
 - インストール可能サービス 187
 - コマンド 187
 - コンテキスト 190
 - 設定 / リセット・コマンド 382
 - 代替ユーザー 189
 - 言語
 - サポートされる 443
 - 現行キュー項目数 134
 - 更新、キュー・マネージャーの属性 124
 - 構成ファイル
 - 概要 243
- 構成ファイル (続き)
 - キュー・マネージャー (QMINI)
 - 使用不可、オブジェクト権限 マネージャーの 183
 - スタンザ 246
 - 内容 245
 - バックアップ 58
 - 編集する 254
 - MQSeries (MQSINI) 244
 - 概要 244
 - 内容 244
 - パス 129
 - バックアップ 58
 - 構造データ・タイプ 427
 - 構文エラー、MQSC コマンドの 121
 - 構文図、見方 325
 - コピー、キューの 100
 - コピー、チャンネルの 115
 - コマンド
 - キュー・ファイル属性の更新 (altmqfls) 327
 - キュー・マネージャーの開始 (strmqm) 390
 - キュー・マネージャーの削除 (dltmqm) 347
 - キュー・マネージャーの作成 (crtmqm) 342
 - キュー・マネージャーの終了 (endmqm) 365
 - クライアント・チャンネル定義の交換 (cnvclchl) 337
 - 権限の設定 / リセット (setmqaut) 186, 382
 - 権限の表示 (dspmqaout) 349
 - コマンド・サーバーの開始 (strmqcsv) 389
 - コマンド・サーバーの終了 (endmqcsv) 363
 - コマンド・サーバーの表示 (dspmqcsv) 354
 - 指定したプリンシパルについての情報の表示 (dspmquqr) 361
 - 制御 37
 - altmqfls 327
 - altmqusr 333
 - cleanrdf 335

コマンド (続き)

cnvclchl 337
crtmqcvx 339
crtmqm 342
dltmqm 347
dspmqaut 349
dspmqcsv 354
dspmqfls 356
dspmqtrc 360
dspmqusr 361
endmqcsv 363
endmqm 365
endmqtrc 368
instmqm 370
runmqchi 371
runmqchl 372
runmqdlq 373
runmqlsr 375
runmqsc 377
runmqtrm 381
setmqaut 382
strmqcsv 389
strmqm 390
strmqtrc 392
upgmqm 396

セキュリティ・コマンド

dspmqaut 188
setmqaut 184

セットの比較 413

送達不能キュー・ハンドラーの実行 373

チャンネルの実行 (runmqchl) 372

チャンネル・イニシエーターの実行 (runmqchi) 371

トリガー・モニターの開始 (runmqtrm) 381

プログラム式コマンド形式 (PCF) 39

リスナーの実行 (runmqlsr) 375

Compaq NSK ユーザー ID に対応するプリンシパルの定義 (altmqusr) 333

DLQ ハンドラーの実行 (runmqdlq) 205

MQSC

確認 128

コマンド (続き)

MQSC (続き)

コマンド・ファイル 126

使用 39

ALTER QLOCAL 135

ALTER QREMOTE 170

DEFINE CHANNEL 162

DEFINE QALIAS 143

DEFINE QLOCAL 134

DEFINE QLOCAL LIKE 134

DEFINE QLOCAL

REPLACE 135

DEFINE QMODEL 145

DEFINE QREMOTE 168

DELETE QLOCAL 138

DISPLAY QREMOTE 170

MQSeries for Compaq NSK のインストール (instmqm) 370

MQSeries 定様式トレース出力の表示 (dspmqtrc) 360

MQSeries トレースの開始 (strmqtrc) 392

MQSeries トレースの終了 (endmqtrc) 368

RDF ハウスキーピングの実行 (cleanrdf) 335

runmqsc 120

V2.2.0.1 キュー・マネージャーのアップグレード (upgmqm) 396

コマンド・エラー 263

コマンド・キュー 18

コマンド・サーバー

開始コマンド 389

コマンド・サーバーの開始 154

コマンド・サーバーを停止する 155

終了コマンド 363

障害からのリカバリー 310

状況の表示 154

表示コマンド 354

リモート管理 154

PATHCOM コマンドを使用した構成 41

コマンド・セット

管理 37

比較 413

コマンド・ファイル 126

コミットとバックアウト 231

コンテキスト権限 190

[サ行]

サーバー接続チャンネルの自動定義 165

サーバー・クラス

命名規則

キュー・サーバー 79

状況サーバー 76

ユーザー定義 55

MQS-TCPLISnn 71

サービス・コンポーネント 23

再始動、キュー・マネージャーの 88

最大

キューのサイズ 10

メッセージ数 10

作業単位

説明 25

作業単位管理 438

削除

キュー・マネージャー 89, 347

チャンネル 111

ローカル・キュー 102, 138

作成

キュー 97

キュー・マネージャー 58, 83

チャンネル定義 107

伝送キュー 171

プリンシパル 85

プロセス定義 148

サブボリューム 34

参考資料 520

サンプル・プログラム

C バージョンの作成 450

COBOL バージョンの作成 454

C++ バージョンの作成 453

MQSeries for Compaq NSK で提供される 449

OSS でのサンプルの作成 458

TAL バージョンの作成 457

時間に依存しないアプリケーション 8

シグナル・オプション 429

- システム・オブジェクト
 - 復元 86
- システム・デフォルト 405
- システム・デフォルト・オブジェクト 20
- 事前定義キュー 10
- 持続データ 302
- 持続メッセージ
 - 可用性 302
 - ストレージ 287
 - チューニング 285
 - デフォルトの最大数の指定変更 412
 - 非持続メッセージとの違い 9
- シャットダウン、キュー・マネージャーの 87
- 修正コマンド 39
- 終了、キュー・マネージャーの 87
- 終了、対話式 MQSC コマンドの 122
- 受信側のチャンネルの自動定義 165
- 手動によるキュー・マネージャーの除去 418
- 障害からのリカバリー 310
- 使用可能化
 - セキュリティ 183
- 状況サーバー
 - サーバー・クラス
 - MQS-STATUS00 44
 - 障害からのリカバリー 310
 - 除去 76
 - デフォルトではない場合の追加と除去 74
 - デフォルトの名前 73
 - プロセス
 - 追加 45
 - リカバリーおよび再始動 239
- 使用不可、オブジェクト権限マネージャー (OAM) の 183
- 除去、キュー・サーバーの 79
- 除去、状況サーバーの 76
- 除去、デフォルト以外のキュー・サーバーの 77
- 除去、デフォルト以外の状況サーバーの 74
- 所定稼働環境 401
- 処理、イベント・ドリブ 8
- スーパーユーザー (MQSeries)
 - MQM 178
- スケーラビリティ 283
- スタンザ
 - MQSINI 244
 - QMINI 246
- スレッド
 - アプリケーションの 444
- スワップ・スペースの割り振り 72
- 制御コマンド 37
 - 大文字小文字の区別 38
 - altmqfls 327
 - altmqusr 333
 - cleanrdf 335
 - cnvelchl 337
 - crtmqcvx 339
 - crtmqm 342
 - dltmqm 347
 - dspmqaout 349
 - dspmqcsv 354
 - dspmqlfs 356
 - dspmqtrc 360
 - dspmqusr 361
 - endmqcvs 363
 - endmqm 365
 - endmqtrc 368
 - instmqm 370
 - runmqchi 371
 - runmqchl 372
 - runmqdlq 373
 - runmqqls 375
 - runmqsc 120, 377
 - runmqtrm 381
 - setmqaut 382
 - strmqcvs 389
 - strmqm 390
 - strmqtrc 392
 - upgmqm 396
- 制御されたシャットダウン 87
- 制限
 - オブジェクト名 324
 - MQM オブジェクトへのアクセス 177
- 静止シャットダウン 87
- セキュリティ 177 (続き)
 - コマンドの使用 184, 188
 - 使用可能化 183
 - プリンシパルの作成 85
 - リモート 191
 - OAM (オブジェクト権限マネージャー) 24
 - セグメント ID 69
 - 送達不能キュー
 - 指定 58
 - 説明 17
 - ハンドラー 373
 - 送達不能ヘッダー、MQDLH 205
 - 測定カウンター 293
 - ソフトコピー資料 519

[夕行]

- 代替ユーザー権限 189
- タイプ、オブジェクトの 11
- タイムアウト、MQSC コマンドの応答の 166
- 対話式 MQSC
 - からのフィードバック 121
 - 終了 122
 - 使用 120
- タッチ・ファイル 287, 309
 - 再配置 289
- 単一障害ポイント 300
- 単一フェーズ・コミット 231
- チェックポイント
 - キュー・サーバーによる非持続メッセージの 293
- チャンネル
 - イベント 222, 497
 - エスケープ・コマンドの許可 198
 - 開始 164
 - 開始プログラムの実行コマンド 371
 - 数の制限 69
 - キュー・マネージャー間での定義 15
 - 構成 248
 - コマンド 192
 - 実行コマンド 372
 - 自動定義 165

- チャンネル (続き)
 - セキュリティー 192
 - セキュリティー要件 192
 - 説明 19, 157
 - 定義 162
 - 出口 459
 - メッセージ・チャンネル 286
 - リモート管理 161
 - リモート・キューイング 157
 - ICE 490
 - TCP/IP 受信側チャンネルを定義する 492
 - TCP/IP 送信側チャンネルを定義する 492
- チャンネル同期サブボリューム 82
- チャンネル同期ファイル 309
- チャンネルの自動定義 165
- チャンネルの動的定義 165
- チャンネル・イニシエーター
 - 障害からのリカバリー 310
 - PATHCOM コマンドを使用した構成 42
- 追加、デフォルト以外のキュー・サーバーの 77
- 追加、デフォルト以外の状況サーバーの 74
- 通信のセットアップ 471
- 通信の例
 - ICE 487
 - SNAX 481
 - TCP/IP 491
- 通知、イベントの 223
- データベース
 - 一貫性 305
 - 外部的な一貫性 306
 - 監査データベース・ファイル 309
 - TM/MP によって保護される 306
- データ変換 174, 435
 - デフォルトのデータ変換 174
 - ユーザー定義のメッセージ形式 175
 - crmqcvx コマンド 339
 - EBCDIC 65
- データ変換出口 459
- データ保全性
 - 構成 319
 - 定義 301
- データ保全性に関する構成 319
- データ・タイプ
 - 構造データ・タイプ 427
- データ・ファイル・サブボリューム、キュー・マネージャー 80
- 定義、キューの 97
- 停止
 - キュー・マネージャー 365, 417
 - チャンネル 112
 - トレース 95
- ディスク・ボリューム
 - キュー・ファイルの区画化 290
- ディレクトリー、キュー・マネージャー 189
- 出口
 - インストール 465
 - クラスター・ワークロード出口 23
 - チャンネル出口 23
 - 名前の形式 460
 - ユーザー出口 23, 459
 - MQ_LOAD_ENTRY_POINT_EXIT 460
- デバッグ
 - 共通プログラミング・エラー 263
 - 予備的な検査 257
 - 2次検査 263, 268
- デフォルト
 - オブジェクト 20, 86
 - キュー・サーバー、名前 76
 - キュー・マネージャー 58
 - 誤った削除 344
 - 誤った変更 89
 - 処理されるコマンド 121
 - 変更 89, 123
 - システム・オブジェクト 405
 - 状況サーバー、名前 73
 - 伝送キュー 58, 171
 - TCP/IP ポート 70
- デフォルト以外の TCP/IP プロセス 71
- デフォルト以外のキュー・サーバー、追加と除去 77
- デフォルト以外の状況サーバー、追加と除去 74
- デフォルトのデータ変換 174
- 伝送キュー
 - キュー・マネージャー間での定義 15
 - 作成 171
 - 説明 17
 - 定義 162
 - デフォルト 58, 171
 - リモート管理 161
- テンプレート、EMS イベント 224
- 同期点 306
 - 制限 232
- 同期点、パフォーマンスの考慮事項 270
- 動的キュー 10
 - 許可 189
 - 説明 10
- 動的バインディング 30
- トラブルシューティング 235
- トランザクション
 - 複数管理 308
- トランザクション・サポート 231
- トランザクション・マネージャー (NonStop TM/MP) 231
- トリガー
 - イベント 222
 - 開始キュー上のメッセージ 16
 - モニター
 - 開始コマンド 381
 - 説明 17
 - PATHCOM コマンドを使用した構成 42
- トリガー操作
 - アプリケーション・キューの定義 146
 - 定義 8
 - のためのオブジェクトの管理 146
 - トリガー操作のためのオブジェクトの管理 146
- 取り出しアルゴリズム、メッセージの 11
- トレース
 - データ・サンプル 280

トレース (続き)
パフォーマンスの考慮事項 280
トレース、MQSeries オブジェクトの
93
トレース・データのサンプル 280

[ナ行]

内容
MQSINI 244
QMINI 246
名前
オブジェクト 12
オブジェクトで有効な 324
名前、デフォルトのキュー・サーバ
ーの 76
名前、デフォルトの状況サーバの
73
名前サービス 24
名前サービス・インターフェース
506
名前の変換、オブジェクトの 83
名前リスト定義 309
入出力のリダイレクト、MQSC コマ
ンドでの 124
ネイティブ・バインディング 30

[ハ行]

ハイパーテキスト・マークアップ言
語 (HTML) 519
バインディング 30
パターン・マッチング・キーワー
ド、ルール・テーブル 209
発行、MQSeries コマンドの 118
バッファ
ブラウザ中のメッセージ 291
パネル
MQM 90
パフォーマンス 283
アプリケーション設計の考慮事項
269
トレースを使用する場合の考慮事
項 280
パフォーマンス・イベント 222, 497
パラメーター
altmqfls コマンド 328

パラメーター (続き)
altmqusr コマンド 333
cleanrdf コマンド 335
cnvclchl コマンド 337
crtmqcvx コマンド 339
crtmqm コマンド 342
dlmqm コマンド 347
dspmqaut コマンド 349
dspmqcsv コマンド 354
dspmqfls コマンド 356
dspmqtrc コマンド 360
dspmqusr コマンド 361
endmqcsv コマンド 363
endmqm コマンド 365
endmqtrc コマンド 368
instmqm コマンド 370
runmqchi コマンド 371
runmqchl コマンド 372
runmqdlq コマンド 373
runmqslr コマンド 375
runmqsc コマンド 378
runmqtrm コマンド 381
setmqaut コマンド 384
strmqcsv コマンド 389
strmqm コマンド 390
strmqtrc コマンド 392
upgmqm コマンド 397
非クライアント・チャンネル定義 309
非持続データ 302
非持続メッセージ
可用性 305
持続メッセージとの違い 9
ストレージ 288
チェックポイント 293
チューニング 285
データ保全性に関する構成 319
同期化ロギング 286
非ネイティブ・バインディング 30
表示
キュー・マネージャーの属性
122
権限コマンド 349
コマンド・サーバの状況 154
コマンド・サーバ・コマンド
354
チャンネル状況 111

表示 (続き)
プロセス定義 149
MQSeries 定様式トレース出力コ
マンド 360
MQSeries ファイル・コマンド
356
ファイル
オブジェクト・カタログ 309
オブジェクト・タッチ・ファイル
309
監査データベース・ファイル
309
基本データベース 309
キュー・オーバーフロー・ファイ
ル 287, 309
位置 289
キュー・ファイル 287, 309
位置 289
許可 201
構成
キュー・マネージャー
(QMINI) 245
問題判別での 279
MQSeries (MQSINI) 244
初期設定 32
タッチ
別名およびリモート 36
タッチ・ファイル 287, 309
位置 289
名前リスト 36
名前リスト定義 309
非クライアント・チャンネル定義
309
メッセージ・オーバーフロー
287
位置 289
しきい値の変更 290
EC 制御ファイル 309
ENSCRIBE 34
OAM データベース 309
フィードバック、MQSC コマンドか
らの 121
復元
システム・オブジェクト 86
デフォルト・オブジェクト 86
複数障害ポイント 300

- 不正出力 270
 - ブラウザ、キューの 138
 - プリンシパル
 - 作成 85, 185
 - データベース、PRIDB 201
 - 命名 180
 - 目的 24
 - プログラマブル式コマンド形式 (PCF) 39, 500
 - プログラミング・エラーの例 263
 - プログラム、提供される例 449
 - プロセス間通信コンポーネント (IPCC) 34
 - プロセス定義
 - 作成 148
 - 説明 18
 - 表示 149
 - 分散キューイング
 - 送達不能キュー 17
 - 不正出力 273
 - 未配布メッセージ・キュー 17
 - 別名キュー
 - 許可 189
 - 説明 15
 - 別名付け
 - 応答先キュー 173
 - キュー・マネージャー 172
 - 変更
 - キュー 101
 - チャンネル 105
 - CCSID 65
 - ホーム・ボリューム、キュー・マネージャーの 62
 - 保護されるリソース 181
 - ボリューム
 - 構造 79
 - ボリューム、変更 136
- [マ行]**
- 無許可アクセスからの保護 177
 - 命名規則
 - 各国語サポート 324
 - キュー・サーバー・クラス 79
 - 状況サーバー・クラス 75
 - メッセージ
 - インストールメンテーション・イベントの 224
 - オペレーター 277
 - 可変長 269
 - 記述子 8
 - キューイング 7
 - キューに現れない 270
 - グループ化 9
 - サイズ 287
 - 持続 285
 - セグメント 9
 - 説明 8
 - 特定の検索 269
 - 取り出しアルゴリズム 11
 - 長さ 9, 269
 - パフォーマンスの考慮事項 269
 - 非持続 285
 - 未配布 279
 - 予期しない情報を含む 272
 - メッセージ長、短くする 135
 - メッセージ・オーバーフロー・ファイル 287
 - 再配置 289
 - しきい値の変更 290
 - データ保全性に関する構成 319
 - メッセージ・キュー管理 (MQM) 90
 - メッセージ・キュー・インターフェース 7
 - メッセージ・キュー・サブボリューム 81
 - メッセージ・チャンネル 286
 - メッセージ・チャンネル・エージェンツ (MCA)
 - 機能 30
 - 障害からのリカバリー 310
 - メッセージ・ドリブン処理 7
 - メモリー・バッファー 288
 - モデル・キュー
 - 説明 15
 - 属性 434
 - 定義 145
 - 取り扱い 145
 - 戻りコード 258
 - altmqfls コマンド 331
 - cleanrdf コマンド 336
 - 戻りコード 258 (続き)
 - crtmqcvx コマンド 339
 - crtmqm コマンド 345
 - dltmqm コマンド 347
 - dspmquaut コマンド 352
 - dspmqcsv コマンド 354
 - dspmqlfs コマンド 357
 - dspmqustr コマンド 361
 - endmqcsvg コマンド 363
 - endmqm コマンド 366
 - endmqtrc コマンド 368
 - runmqchi コマンド 371
 - runmqchl コマンド 372
 - runmqslr コマンド 376
 - runmqsc コマンド 379
 - runmqtrm コマンド 381
 - setmqaut コマンド 387
 - strmqcsvg コマンド 389
 - strmqm コマンド 390
 - strmqtrc コマンド 394
 - モニター
 - TMF 状況 234
 - モニター、キューの 102
 - モニター、キュー・マネージャーの 222
 - モニター、チャンネルの 110
 - 問題判別
 - 共通プログラミング・エラー 263
 - 構成ファイル 279
 - コマンドから応答がない 264
 - 最初に検査すること 257
 - さらに他の検査 263, 268
 - 不正出力 273
- [ヤ行]**
- ユーザー
 - グループ 180
 - 属する、複数のユーザー・グループに 180
 - ユーザー ID
 - 許可 188
 - 権限 178
 - Compaq NSK ログイン・ユーザー 188

ユーザー定義サーバー・クラス 55
ユーザー定義のメッセージ形式の変換 175
ユーザー出口 459
 クラスター・ワークロード 23
 説明 23
 チャンネル出口 23
 データ変換出口 23
ユーザー・グループ
 許可 182
 MQM 178
ユーロ通貨記号サポート 404
有効化
 インストルメンテーション・イベント 223
優先キュー・マネージャー・シャットダウン 88
要件
 ソフトウェア 401
 ハードウェア 401
用語、本書で使用する 523
用語集 523

[ラ行]

ライセンス管理 404
ライブラリー
 MQI 444
リスナーの実行 (runmqtsr コマンド) 375
リセット、メッセージ・シーケンス番号 (MSN) 113
リソースの保護 177
リポジトリ・キャッシュ・マネージャー
 障害からのリカバリー 310
リポジトリ・マネージャー
 障害からのリカバリー 310
 プロセス 29
リモート
 イベント 497
 オブジェクト管理 157
 キュー
 応答先キュー別名として 173
 キュー・マネージャー別名としての 172
 許可 189

リモート (続き)
 キューイング
 勧告 167
 キュー定義の作成 168
 キュー・オブジェクトの取り扱い 172
 セキュリティに関する考慮事項 191
 MQSC コマンドの発行 165
リモート管理
 コマンド・サーバー 154
 初歩的な問題 167
リモート・キュー 157
 説明 13, 15
理由コード 13
ルール・テーブル、DLQ ハンドラー 206
 構文 213
 処理 215
 制御データ・エントリー
 INPUTQ キーワード 207
 INPUTQM キーワード 207
 RETRYINT キーワード 208
 WAIT キーワード 208
パターンと処理 (ルール)
 ACTION キーワード 211
 APPLIDAT キーワード 209
 APPLNAME キーワード 209
 APPLTYPE キーワード 210
 DESTQ キーワード 210
 DESTQM キーワード 210
 FEEDBACK キーワード 210
 FORMAT キーワード 210
 FWDQ キーワード 211
 FWDQM キーワード 212
 HEADER キーワード 212
 MSGTYPE キーワード 210
 PERSIST キーワード 210
 PUTAUT キーワード 212
 REASON キーワード 210
 REPLYQ キーワード 211
 REPLYQM キーワード 211
 RETRY キーワード 213
 USERID キーワード 211
例 217

例
 エラー・ログ 277
 共通プログラミング・エラー 263
 通信のセットアップ 480
 altmqfls コマンド 331
 altmqusr コマンド 334
 cleanrdf コマンド 336
 cnvclchl コマンド 337
 crtmqcvx コマンド 339
 crtmqm コマンド 345
 dltnmqm コマンド 347
 dspmqaut コマンド 353
 dspmqcsv コマンド 354
 dspmqfls コマンド 357
 dspmqusr コマンド 362
 endmqcsv コマンド 364
 endmqm コマンド 366
 endmqtrc コマンド 368
 runmqsc コマンド 379
 setmqaut コマンド 388
 strmqcsv コマンド 389
 strmqm コマンド 391
 strmqtrc コマンド 394
 upgmqm コマンド 397
ローカル管理 117
ローカル作業単位
 説明 25
ローカル・イベント 497
ローカル・キュー
 開始 16
 クリア 138
 コマンド 18
 削除 138
 説明 13, 15
 送達不能 17
 属性 434
 定義 131
 定義のコピー 134
 伝送 17
 未配布メッセージ 17
ローカル・キュー・マネージャー・エージェント (LQMA) 295
 機能 28
 障害からのリカバリー 310
ロード・バランシング 63

ロールバック 231
ログ、エラー 275, 277
ロック、キャッシュ内に 292

[ワ行]

ワークロード・バランシング 283
 クラスターを使用した 11

[数字]

2 フェーズ・コミット 231

A

ADD SERVER、PATHCOM コマンド 41
AllQueueManagers スタンザ、MQSINI ファイル 244
ALTER SERVER
 PATHCOM コマンド 43
altmqfls コマンド 136, 327
 関連コマンド 332
 キュー項目数の測定 293
 キュー・サーバー始動時のデータ構造の構築 292
 キュー・サーバーによるメモリー内へのロック 292
 キュー・ファイルの再配置 289
 非持続メッセージのチェックポイント 293
 ブラウザしきい値の設定 291
altmqusr コマンド 333
 関連コマンド 334
APC PATHWAY 定義、例 483
APPLIDAT キーワード、ルール・テーブル 209
APPLNAME キーワード、ルール・テーブル 209
APPLTYPE キーワード、ルール・テーブル 210
Authority スタンザ、QMINI ファイル 247

C

C ネイティブ
 サポートされる言語 443

C 非ネイティブ
 サポートされる言語 443
CCSID
 変換メソッド 66
 MQSeries for Compaq NSK によるサポート 404
CCSID パラメーター 65
CCSID ファイル 174
ChannelInitiator スタンザ、QMINI ファイル 247
Channels スタンザ、QMINI ファイル 248
 「Channels」メニュー、MQM 103
cleanrdf コマンド 335
cnvclchl コマンド 337
COBOL ネイティブ
 サポートされる言語 443
COBOL 非ネイティブ
 サポートされる言語 443
Compaq NonStop Server for Java 419
Compaq NSK ログイン・ユーザー ID 188
Configuration スタンザ、QMINI ファイル 246
CONNAME
 プロセス属性 425
CorrelId
 使用時のパフォーマンスの考慮事項 269
CPU
 可用性に関する構成 318
 キュー・サーバー 288
 高使用量 294
 チューニング 283
 ワークロード・バランス 43
crtmqcvx コマンド 339
crtmqm コマンド 342
CURDEPTH 293
C++ ネイティブ
 サポートされる言語 443

D

DefaultPrefix パラメーター、crtmqm コマンド 62

DefaultProcess スタンザ、QMINI ファイル 246
DefaultQueueManager スタンザ、MQSINI ファイル 244
DESTQ キーワード、ルール・テーブル 210
DESTQM キーワード、ルール・テーブル 210
DISPLAY CHSTATUS コマンド 424
DLQ ハンドラー
 呼び出し 205
 ルール・テーブル 206
dltmqm コマンド 347
dspmqaut コマンド 349
 使用 184, 187
dspmqcsv コマンド 354
dspmqfls コマンド 356
dspmqtrc コマンド 360
dspmqusr コマンド 361

E

EBCDIC
 データ変換 65
EC
 機能 28
 障害からのリカバリー 310
 制御ファイル 309
EC Boss
 障害からのリカバリー 310
EC Boss の役割 63
EC スタンザ、QMINI ファイル 246
EC プロセスの数 62
ECBoss
 機能 28
ECBoss スタンザ、QMINI ファイル 246
EMS イベント 224, 279
 処理のためのプログラムの作成 229
 代替コレクター、指定 229
 デフォルト・コレクター 229
 MQEMSEVENTS PARAM の設定 228
EMS イベント・テンプレート、MQSeries 507

EMSCollectorName 67
endmqcsv コマンド 363
endmqm コマンド 87, 365
endmqtrc コマンド 368
ENSCRIBE ファイル 25, 34
Event Management Service (EMS) イベント 224
ExpectedNumECs 62
ExtPoolSize エントリー
 QMINI ファイル 72

F

FASTPATH バインディング 295, 445
 過負荷の削減 296
 使用可能化 296
 制限 297
FEEDBACK キーワード、ルール・テーブル 210
FFST
 サブボリューム 80
 調査 281
FORMAT キーワード、ルール・テーブル 210
FREEZE SERVER、PATHCOM コマンド 41
FWDQ キーワード、ルール・テーブル 211
FWDQM キーワード、ルール・テーブル 212

G

Guardian セグメント ID 69
Guardian ユーザー ID 309

H

HARDENBO
 プロセス属性 425
HEADER キーワード、ルール・テーブル 212
HomeTerminalName パラメーター 63
HTML (ハイパーテキスト・マークアップ言語) 519

I

ICE TP 473
ICE 通信の例 487
INPUTQ キーワード、ルール・テーブル 207
INPUTQM キーワード、ルール・テーブル 207
instmqm コマンド 370

J

Java
 互換性 419
 サポートされる言語 443
Java トランザクション・サービス 443

L

LIKE 属性、DEFINE QLOCAL 134
LQMAgent スタンザ、QMINI ファイル 247
LQMAgentPriority 68
LU6.2 応答側プロセス 472
LU6.2 リスナー 32

M

make ファイル
 Java 419
MaxActiveChannels 69
Maxapplio の値 486
MaxChannels 68
MAXHANDS
 プロセス属性 425
MaxIdleAgentReuse 68
MaxIdleAgents 68
maxrev PARAM 488
MAXUMSGS
 プロセス属性 425
MCAAgentPriority 68
MCACaller スタンザ、QMINI ファイル 247
MCALU62Responder スタンザ、QMINI ファイル 247

MCATCPResponder スタンザ、QMINI ファイル 247
MinIdleLQMAgents 67
MinIdleMCACallers 67
MinIdleMCALU62Responders 67
MinIdleMCATCPResponders 67
MQAI
 説明 153, 447
MQCFH、PCF ヘッダー 501
MQCFIL、PCF 整数リスト・パラメーター 502
MQCFIN、PCF 整数 503
MQCFSL、PCF スtring・リスト 503
MQCFST、PCF スtring・パラメーター 502
MQCNO 428
MQCNO_FASTPATH_BINDING 296
MQCONN 295
MQCONNECTTYPE 296
MQCONNX 296
MQDATACONVEXIT 435
MQDEFAULTPREFIX 409
MQDISC 433
MQDLH、送達不能ヘッダー 205
MQEMSEVENTS 409
MQEMSEVENTS、環境変数 228
MQGMO 429
MQGMO_BROWSE_* 440
MQGMO_SET_SIGNAL オプション 429
MQI
 キュー・マネージャー呼び出し 14
 許可 193
 説明 7
 呼び出し 431
 ライブラリー 444
 ローカル管理のサポート 117
TAL プログラム言語
 MQCLOSE 432
 MQDISC 433
 MQINQ 433
 MQSET 434
MQINQ 433

MQIServer スタンザ、QMINI ファイル 247

MQLISTENPORTNUM 409

MQM
 ユーザー ID 178, 188
 ユーザー・グループ 178

MQM (メッセージ・キュー管理) インターフェース 90

MQMACHINIFILE 409

MQMACHINIFILE、環境変数 244

MQMC 機能 90

MQMD 430

MQMREFRESHINT
 PATHWAY パラメーター 45

MQOPEN 許可 193

MQPMO 431

MQPUT および MQPUT1、パフォーマンスの考慮事項 270

MQPUT 許可 193

MQQSHKEEPINT 411

MQQSMAXBATCHEXP 412

MQQSMAXMSGSEXPIRE 412

MQQSSIGTIMEOUT 412

MQRC_Q_SPACE_NOT_AVAILABLE
 戻りコード 132

MQRDF 409

MQRDFUPPROCESSNAME 409

MQRDFUPPROGNAME 409

MQSC 40
 行の最大長 126
 コマンド応答のタイムアウト 166
 コマンドの確認 128
 コマンドの使用 124
 コマンドの発行方法 118
 コマンド・ファイル
 実行 128
 出力レポート 126
 対話式でのコマンドの発行 120
 対話式での入力終了 122
 チャンネルでのセキュリティ要件 192
 入出力のリダイレクト 124
 問題
 リモート 167
 ローカル 129

MQSC 40 (続き)
 リモート側での発行 165

MQSeries for Compaq NSK によるサポート 421

MQSC コマンド
 大文字小文字の区別 120
 行の最大長 126
 使用 39
 ALTER QLOCAL 135
 ALTER QREMOTE 170
 DEFINE CHANNEL 162
 DEFINE QALIAS 143
 DEFINE QLOCAL 134
 DEFINE QLOCAL LIKE 134
 DEFINE QLOCAL
 REPLACE 135
 DEFINE QMODEL 145
 DEFINE QREMOTE 168
 DELETE QLOCAL 138
 DISPLAY QREMOTE 170

MQSeries for Compaq NSK
 アーキテクチャー 27
 アプリケーションの構築 437
 アプリケーションの実行 437
 クライアント・サポート 495
 コンポーネント 33
 サブボリューム 34
 サポートされる MQSC 421
 スーパーユーザー、MQM 178
 スケーラビリティ 283
 で使用される EMS イベント・テンプレート 507
 パフォーマンス 283
 ファイル 34
 ボリューム構造 79
 ユーザー出口 459
 要約 401
 TM/MP (TMF) サポート 231

MQSeries for Compaq NSK の概要 401

MQSeries for Compaq NSK のチューニング 283

MQSeries オブジェクトの属性
 キュー 14
 キュー・マネージャー
 更新 124

MQSeries オブジェクトの属性 (続き)
 キュー・マネージャー (続き)
 表示 122
 デフォルト 133
 ローカル・キューの変更 135
 ALL 属性 133
 MQSC と PCF の比較 40

MQSeries 参考資料 517

MQSeries トレースの起動コマンド 392

MQSeries トレースの終了 368

MQSET 434

MQSINI、構成ファイル
 概要 244
 へのパス 129
 編集する 254
 例 244

MQSNOAUT 409

MQSNOAUT、環境変数 183

MQSS サーバー 32

MQS-CHANINIT00
 TS/MP サーバー・クラス 42

MQS-CMDSERV00
 TS/MP サーバー・クラス 41

MQS-EC00
 TS/MP サーバー・クラス 43

MQS-STATUS00 サーバー・クラス 44

MQS-TCPLIS00
 TS/MP サーバー・クラス 41

MQS-TCPLISnn
 再構成 71

MQS-TRIGMON00
 TS/MP サーバー・クラス 42

MQTRANSACTIONLIFE
 TM/MP トランザクションのリフレッシュ 438

MQZAO 定数および権限 194

MQ_LOAD_ENTRY_POINT_EXIT 460

MsgId
 使用時のパフォーマンスの考慮事項 269

MSGTYPE キーワード、ルール・テーブル 210

MVS/ESA キュー・マネージャー 166

N

NOHARDENBO

プロセス属性 425

NonStop Java

サポートされる言語 443

NonStop TM/MP (トランザクション・マネージャー)

231

NonStop Tuxedo

TM/MP とトランザクション環境
308

NonStop キュー・サーバー

286

NonStop プロセス・ペア

31

NPMSPEED チャンネル属性

286

NSKSegidRange

69

NumECs パラメーター、crtmqm コマンド

62

O

OAM (オブジェクト権限マネージャー)

置き換え 506

概要 24

機能 180

グループ 180

重要操作 188

使用不可 183

dspmqaout コマンド 188

setmqaut コマンド 184, 186

OAM データベース

309

OAM プリンシパル

309

OpenTMF

307

P

PARAM (環境変数)

409

PATHCOM コマンド

ADD SERVER 41

ALTER SERVER 43

FREEZE SERVER 41

START SERVER 41

STATUS SERVER 41

STOP SERVER 41

THAW SERVER 41

PathmonProcName

64

PATHWAY

エラー 262

PATHWAY (続き)

可用性に関する構成 318

キュー・サーバーのサーバー・クラス 288

構成の例 46

サーバー・クラス 54

PCF コマンド

エスケープ PCF 152

MQAI、の使い方を単純化するための使用 153

MQSC および PCF での属性
152

PCF を使用した管理用タスクを自動化する 151

PCF コマンド応答

505

PCF (プログラマブル式コマンド形式)

MQSeries for Compaq NSK によるサポート 504

PCF (プログラム式コマンド形式)

500

紹介 39

ストリング・パラメーター
(MQCFST) 502

ストリング・リスト
(MQCFSL) 503

整数 (MQCFIN) 503

整数リスト・パラメーター
(MQCFIL) 502

ヘッダー (MQCFH) 501

メッセージ記述子 501

MQCFH (ヘッダー) 501

MQCFIL (整数リスト・パラメーター) 502

MQCFIN (整数) 503

MQCFSL (ストリング・リスト)
503

MQCFST (ストリング・パラメーター) 502

PDF (Portable Document

Format) 519

PERSIST キーワード、ルール・テーブル

210

PMSEARCH

409

Portable Document Format

(PDF) 519

PRIDB、プリンシパル・データベース 201

PUTAUT キーワード、ルール・テーブル 212

Q

QMDefaultVolume 62

QMINI、構成ファイル

概要 245

複数の TCP/IP リスナーをサポートする構成 494

編集する 254

例 249

「Queue Manager

Menu」、MQM 92

QueueManager スタンザ、MQSINI

ファイル 244

「Queues」メニュー、MQM 95

R

RDF (リモート・データベース複写機能)

を使用した災害時回復 240

cleanrdf コマンド 85, 335

REASON キーワード、ルール・テーブル

210

REPLACE 属性、DEFINE コマンド

126

REPLYQ キーワード、ルール・テーブル

211

REPLYQM キーワード、ルール・テーブル

211

RETRY キーワード、ルール・テーブル

213

RETRYINT キーワード、ルール・テーブル

208

return codes

altmqusr コマンド 334

runmqchi コマンド 371

runmqchl コマンド 372

runmqdlq コマンド 205, 373

runmqqlsr コマンド 375

runmqsc

確認 128

runmqsc (続き)
キュー・マネージャーの指定
123
キュー・モード 165
コマンド 377
終了 122
使用 124
対話式での使用 120
入出力のリダイレクト 124
フィードバック 121
問題 129
MQSC コマンドの発行 118
runmqstrm コマンド 381

S

SAFEGUARD 25, 182, 402
SAVE-ENVIRONMENT ON 409
SCF 構成ファイル、例 481
Service スタンザ、QMINI ファイル
247
ServiceComponent スタンザ、QMINI
ファイル 248
setmqaut コマンド 382
インストール可能サービス 187
関連コマンド 388
使用 184, 186
SNA
CPU の消費 295
SNA リスナー 274
SNA プロトコル 471
SNAX 通信の例 481
STANDARD バインディング 445
Start Channel Initiator 505
Start Channel Listener 505
START SERVER, PATHCOM コマ
ンド 41
STATUS SERVER, PATHCOM コマ
ンド 41
stdin, runmqsc での 124
stdout, runmqsc での 124
STOP SERVER, PATHCOM コマン
ド 41
strmqcsvg コマンド 389
関連コマンド 389
strmqm コマンド 390
関連コマンド 391

strmqtrc コマンド 392
関連コマンド 395
SupportPac 520
system.cluster.transmit.queue 289

T

TACL 環境変数 409
TAL 非ネイティブ
サポートされる言語 443
TCPConfig スタンザ
TCP/IP の構成 491
TCPConfig スタンザ、QMINI ファイ
ル 248
TCPListener スタンザ、QMINI ファ
イル 247
TCPListenerPort 70
TCPNumListenerPorts 70
TCPPort 70
TCP/IP
チャンネル 479
リモート管理 21
TCP/IP 通信の例 491
TCP/IP プロセス 71
TCP/IP プロトコル 471
TCP/IP リスナー
開始 41
構成 41
障害からのリカバリー 310
停止 41
THAW SERVER, PATHCOM コマン
ド 41
TM/MP 25
外部データベースとの調整 306
監査ファイル 285
データベース・ファイルの監査
306
TM/MP (TMF) サポート 231, 438
TS/MP 管理 41
TS/MP サーバー・クラス
MQS-CHANINIT00 42
MQS-CMDSERV00 41
MQS-EC00 43
MQS-TCPLIS00 41
MQS-TRIGMON00 42
TuningParameters スタンザ、QMINI
ファイル 248

Tuxedo、NonStop
TM/MP とトランザクション環境
308

U

upgmqm コマンド 396
USERDATA
プロセス属性 425
USERID キーワード、ルール・テー
ブル 211

W

WAIT キーワード、ルール・テーブ
ル 208

X

XA インターフェース 440
XA 準拠データベース 25

Z

ZMQSTEMPL、EMS イベント・テン
プレート・ファイル 224



部品番号: CT8Z0JA

Printed in Japan

SC88-8944-00



日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木3-2-12

(1P) P/N: CT8Z0JA



Spine information:



MQSeries® for Compaq
NonStop™ Kernel

MQSeries for Compaq NSK V5.1 シス
テム管理ガイド

バージョン 5.1