

# Microsoft Message Queue Server

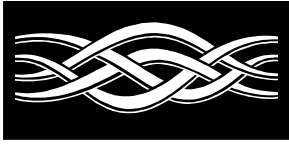
(MSMQ)



**Microsoft**

Microsoft®  
**Windows NT Server**

ネットワーク オペレーティング システム



技術資料

1997年8月

## Microsoft Message Queue Server 概要

---

**Microsoft**

## 免責と商標

この文書に含まれる情報は、製品の発行日における Microsoft Corporation の現在の見解を示しているものです。Microsoft は変化を続けるマーケット状況に対応する必要があるため、本書を Microsoft の側のコミットメントとして見なしてはならず、発行日以降における情報の正確さを保証するものではありません。

この文書中のほかの会社の製品の説明は、読者への便宜上使用されているに過ぎません。Microsoft はこれらの説明の正確さを保証するものではありません。またこれらの製品は時と共に変化する可能性があります。説明では、理解を助けるために、網羅的な説明は避け、要点を簡潔に示すに止めています。これらの製品の正式な説明については、各製品のメーカーにお問い合わせください。

この文書は情報提供のみを目的としています。Microsoft は明示または黙示を問わず、この文書においていかなる保証もしません。

©1995 Microsoft Corporation. All rights reserved. Microsoft、Visual Basic、Windows は米国 Microsoft Corporation の登録商標です。Windows NT は同社の登録商標です。

DECnet は米国 Digital Equipment Corporation の商標です。

MVS は米国 International Business Machines の商標です。

UNIX は米国およびその他の国における Novell, Inc. の登録商標で、X/Open Company Ltd. から独占使用権を得ています。

**Microsoft**

# Microsoft Message Queue Server (MSMQ)

ホワイト ペーパー

*Business Systems Technology Series*

## **Microsoft Business Systems Technology Series について**

Microsoft Business Systems Technology Series は、Windows NT Server と Microsoft BackOffice 製品ファミリを対象に情報テクノロジー(IT)のプロフェッショナルを教育することを目的に用意された一連のホワイト ペーパーからなります。Microsoft 製品に使われている最新テクノロジーがしばしば題材として取り上げられことは確かですが、このホワイト ペーパーの本来の目的は、主要なテクノロジーが現在どのような進展を見せているか、Microsoft におけるこれらのテクノロジーの使われ方、およびこの情報がテクノロジー プランナーに与える影響について1つの考え方を読者に示すことです。

エグゼクティブ サマリ .....	2
1. メッセージのキューイングとは .....	3
<b>1. メッセージ キューイングとは</b>	
2. MSMQ の概要とコンセプト .....	6
MSMQ ではメッセージの蓄積交換方式を使用 .....	6
キューの動的ロケーション .....	7
同期リモート プロシージャ コール、Windows ソケット、および Messaging API (MAPI)とは異なる MSMQ .....	7
異種ネットワークの異なるプラットフォームにまたがるエンタープライズレベルの通信 .....	8
簡単にできるメッセージング アプリケーションの開発 .....	10
タイプの異なるメッセージの送信と応答の受信 .....	10
通信のセキュリティの確保 .....	11
モバイル ユーザーおよびモバイル システムと通信 .....	11
複合ネットワーク向けにスケーラブルなアプリケーションを開発 .....	12
優先順位に基づいて決められるアプリケーションの応答時間 .....	12
ネットワーク障害に対する強い耐性と高いサービス可用性 .....	12
簡単にできる MSMQ のインストールとセットアップ .....	13
Message Queue Explorer を使ってネットワークを管理 .....	14
効率のよいメッセージのルーティング .....	14
ビジネス ニーズの変化に容易に対応 .....	15
ソフトウェア開発とメンテナンスの時間を短縮 .....	15
回線コストの節約 .....	15

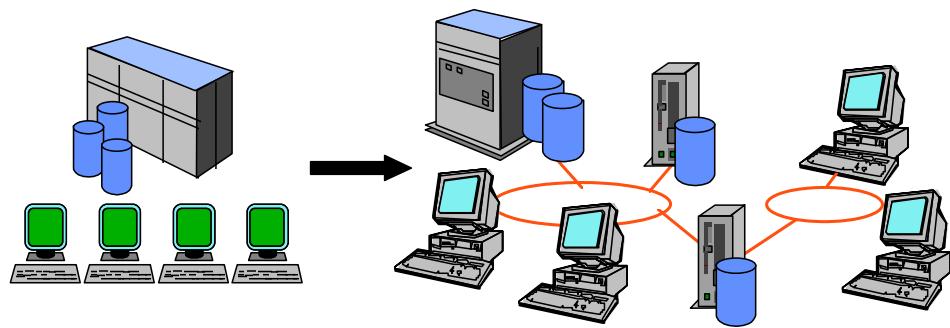
Microsoft® Message Queue Server (この文書では"MSMQ"とも言います)を使用すれば、一時的にオフラインになる場合がある異種ネットワークやシステム間で、実行時間がそれぞれ異なるアプリケーションが相互に通信できるようになります。アプリケーションが MSMQ を使ってメッセージを送ると、MSMQ は中間の記憶領域を使ってそのメッセージが確実に目的のキューに届くようにします。MSMQ を使うことで、メッセージの確実な配信が保証され、効率のよいルーティングやセキュリティ、および優先順位に基づくメッセージングが可能になります。

MSMQ を使用することで、エンドユーザーは、通信し合うアプリケーションやシステムの現在の状態に関係なく、ネットワークにまたがって一時的にオフラインのシステムと通信することができます。MSMQ が安全で確実なネットワーク通信を提供するので、開発者はネットワークの問題に煩わされずにビジネスプログラミングに専念できます。システム管理者は Message Queue Explorer を利用して効率的にネットワークを管理できます。MIS の意思決定者にとっては、通信の信頼性やネットワーク資源の利用効率の向上、ソフトウェア開発とメンテナンス時間の短縮、エンドユーザー、開発者、およびシステム管理者の生産性の向上といったメリットがあります。

### メッセージ キューイングの必要性

企業内での分散コンピューティングが普及するに伴い、アプリケーション間での柔軟で高信頼な通信の確立がますます重要視されるようになってきています。実際のビジネスでは、別のシステム上で実行されている独立のアプリケーションが相互に通信する必要があったり、場合によっては同時に実行されていないアプリケーション間でメッセージを交換しなければならない場面すらしばしばあります。

ビジネスでネットワークに負荷のかかるアプリケーションの使用が増えてきているため、ネットワークはますます障害を起こしやすい状況になっています。ネットワークが一時的にオフラインになってもアプリケーション間の通信を保証する通信テクノロジーが一層重要になります。また、接続を常時期待できないようなリモートシステムとの通信に依存する度もますます高まっています。リモートシステムとの通信でも、ローカルアプリケーションが送ったメッセージは最終的にリモートアプリケーションに到達しなければなりません。ネットワークの普及が進み、無線システムも登場しつつある今、ユーザーにとって低速な通信回線からの応答待ちの時間をなくすることは重要です。通信は、「低速なネットワーク」であっても、ネットワーク要求を処理しながらユーザーが仕事を継続できるようなメカニズムになっていなければなりません。最後に通信テクノロジーでは、ネットワーク資源の有効利用も重要です。



**図1: 分散コンピューティングに移行するには柔軟で高信頼な通信メカニズムが必要**

アプリケーションの開発者自身がネットワーキングソフトウェアを書けば、このような要求を実現することも可能です。しかし、これでは個々のアプリケーションが複雑になるだけでなく、プログラマはネットワークの問題に時間を取られ、肝心のビジネスプログラミングに集中できなくなります。つまり、このような通信テクノロジーはアプリケーションとは独立に提供し、アプリケーションの作成者には、容易にこのテクノロジーを利用できるようにすることが重要です。

アプリケーションは、同時に実行されていない場合でも通信し合わなければならないため、送信側アプリケーションからメッセージを受け取り、受信側アプリケーションが起動するまでこのメッセージを格納する作業は通信テクノロジーが受け持たなければなりません。同様に、通信テクノロジーは、ネットワークがオフラインになったときにも、メッセージを格納し、ネットワークの再起動と同時にメッセージを転送できるようにしなければなりません。つまり、通信テクノロジーは、送信側から受け取ったメッセージをいったん中間の記憶領域に格納し、送信が可能になった時点で格納されたメッセージを受信側のキューに転送します。この送信先のキューに届くまでのメッセージの格納と送り出しの方式をメッセージキューイングと呼びます。Microsoft Message Queue Server (MSMQ)はこのテクノロジーに基づいた分散コンピューティングサービスです。

**Microsoft Message Queue Server (MSMQ)が高信頼で柔軟な通信を提供**

MSMQ は、実行される時間帯が異なる複数のアプリケーションに、ネットワークやシステムの現在の状態に依存しない通信手段を提供します。MSMQ では次のことが可能です。

- 開発者がネットワーク プログラミングではなく、ビジネス プログラミングに専念できる。
- ユーザーは、通信相手のシステムやネットワーク、および送信先アプリケーションが一時的にオフラインになっても、ネットワーク間の通信を保証される。
- システム管理者は効率的なネットワーク管理ができ、オフラインまたはアクセス不能になっている可能性が高いリモート システムも安全に管理できる。
- MIS の意思決定者にとって、通信の信頼性やネットワーク資源の利用効率の向上、ソフトウェア開発とメンテナンス時間の短縮、エンドユーザー、開発者およびシステム管理者の生産性の向上を期待できる。

MSMQ を使用することで、カスタマには次のメリットが得られます。

- 非同期メッセージング: MSMQ 非同期メッセージングを使うと、アプリケーションはほかのアプリケーションにメッセージを送信でき、独立して並列動作し続けることができます。
- 保証されるメッセージの配信: MSMQ を介してメッセージを送ると、送信先のアプリケーションが実行されていなかったり、ネットワークやシステムがオフラインになっても、メッセージを確実に相手に送ることができます。
- ローカルおよびエンド to エンド トランザクション: トランザクショナル モードでメッセージを送受信すると、ローカル データベースの更新やメッセージの送受信が確実にアトミック単位で処理されることを保証します。したがって、すべてがうまくいくか、まったくなにも行われなかったかのどちらかになります。送信側のアプリケーションは、受信側でトランザクションがコミットされたことを確認することもできます。トランザクショナル モードで処理されたメッセージは、確実に順番どおりに送信先に送られ、同じメッセージが 2 回以上送られることは間違ってもありません。
- ルーティングおよび動的構成: MSMQ は異種ネットワークにまたがる場合でも柔軟なルーティングを提供します。ネットワークの構成も、システムやネットワーク自体に大きな変更を加えることなく動的に変更できます。
- コネクションレスメッセージング: MSMQ を使用するアプリケーションではターゲットとなるアプリケーションとの直接のセッションを確立する必要がありません。MSMQ を使用するアプリケーションは、メッセージを送信する時点でオフラインになっている場合もあります。
- セキュリティ: MSMQ は、Windows NT™のセキュリティ機能と暗号化やデジタル署名のための Crypto API に基づく安全な通信を提供します。
- 優先順位を付けたメッセージング: MSMQ ではネットワーク間でのメッセージの転送を優先順位に従って制御できるため、業務上重要なアプリケーションは、重要度の低いアプリケーションよりも優先的に通信を行うことができます。

MSMQ は次の要素から構成されています。

- MSMQ を必要とするどのシステムにも存在するランタイム
- MSMQ ActiveX コンポーネントなど、Visual Basic、Internet Information Server

## 2. MSMQ の概要とコンセプト

Active Server Pages、または ActiveX コンテナから呼び出しできる MSMQ を使用するアプリケーションを開発するための Software Development Kit (SDK)

- ルーティング サービスを提供し、MQIS(構成データを保持する動的な情報ストア)をホストする MSMQ Server
- MSMQ 通信を管理するための Message Queue Explorer

この節では、MSMQ の主な機能とコンセプトの一部を取り上げ、説明します。この節の目的は MSMQ の機能を順序だてて説明することではなく、この製品の目的とこれらの目的を実現するうえで MSMQ が行っていることをよく理解していただくことです。

以下の各項目の優先順位は特に考慮されていません。ここで取り上げたトピックの一覧は目次で参照してください。

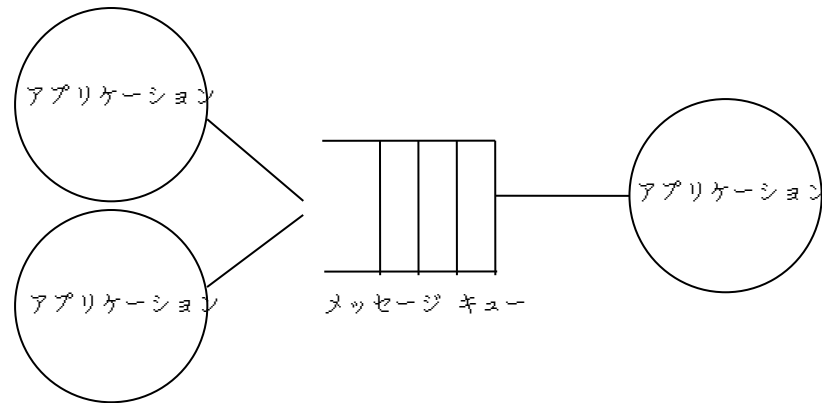
### MSMQ ではメッセージの蓄積転送方式を使用

MSMQ はメッセージを中間の記憶領域に格納する方法で、確実なメッセージの確実な配信を保証しています。アプリケーションは、送信先のキュー上にある受信側の別のアプリケーションにメッセージを送る際に、最初に MSMQ にメッセージを渡します。MSMQ はこのメッセージを、送信が可能になるまで中間の記憶領域に保持します。送信側のアプリケーションはそのまま動作を続けることができ、メッセージが受信側に届くのを待つ必要はありません。その後 MSMQ はそのメッセージを受け取り側のアプリケーションのキューに送ります。必要に応じてネットワーク上でのルーティングも行います。各段階で、MSMQ は、メッセージが送り先の中間記憶領域または目的のキューに届いたことの確認を得てはじめて送り元の中間記憶領域からそのメッセージを削除します。メッセージが最終目的地に達すると、受け取り側のメッセージ キューは、要求があれば肯定応答を送り側に返します。肯定応答の送り先は送信アプリケーションによって指定されたメッセージ キューになります。

アプリケーションは、メッセージ キューを調べることにより、メッセージと肯定応答を受け取ることが出来ます。あるいは、メッセージまたは肯定応答がメッセージ キューに届いたら自動的に通知してくれるように要求することもできます。後者の場合は、アプリケーションは自動的に通知してくれるように要求することで、メッセージに煩わされずに動作を継続できます。メッセージが着信すると、アプリケーションには MSMQ が自動的に通知してくれます。



各メッセージキューはオーナーアプリケーションによって作成され、オーナーは必要に応じて、ほかのアプリケーションに対してメッセージをキューから削除することを許可できます。適切なアクセス権を持ったアプリケーションは、キューからメッセージを削除せずにメッセージキュー内のメッセージを読むことができます。メッセージは、ラベル、優先順位、復元性、およびセキュリティなどの関連付けられたプロパティを持つことができます。



**図2: MSMQ はメッセージ キューを使って通信を提供する**

## キューの動的ロケーション

MSMQ がスケラビリティに富み、管理が容易な理由は、動的にレプリケートされる情報リポジトリと MSMQ が統合されているからです。この情報リポジトリには、企業内の MSMQ マシンとキューの情報がパブリッシュされます。MSMQ の最初のリリースでは自身の情報リポジトリがインプリメントされることとなりますが、後続のリリースでは Windows NT Directory Service と統合される予定です。

特定のキューの特性はすべて、キュー オブジェクトのプロパティとしてこの情報リポジトリに格納されます。たとえば、そのキューにアクセスできるユーザーやアクセスの種類を制御するアクセス制御リスト (ACL) もこうしたプロパティの 1 つです。

メッセージをキューに入れるほかの製品では、アプリケーションは、送り先となるキューを指定するのにそのキューが存在するマシンのネットワーク アドレスを使いますが、MSMQ キューの場合は、キューの作成時にそのキューに対して Globally Unique Identifier (GUID) が割り当てられ、この GUID を使えば、このキューがどのマシン上にあっても、任意のアプリケーションからキューにアクセスすることができます。

アプリケーションはキュー属性のどれかに基づいて動的にキューを探すことができるため、実行時に利用可能な資源やサービスを容易に見つけ出すことができます。

MSMQ キューを分散情報サービス内のオブジェクトとして捉えるこの考え方は、企業内のどこからでも MSMQ を管理できる MSMQ Explorer の基盤にもなっています。

## 同期リモート プロシージャ コール、Windows ソケット、および Messaging API (MAPI)とは異なる MSMQ

MSMQ は通信中のアプリケーションがセッションを維持する必要がないコネクションレス メッセージ サービスですが、同期 RPC では通信中のアプリケーションがセッションを維持する必要があります。同期 RPC は、同じプロトコルを話すアプリケーション間のオンライン通信に適しています。MSMQ は、通信に関係するアプリケーションがオンラインでもオフラインでも、使用するプロトコルが同じでも異なっても、条件に関係なく利用できます。同期 RPC のインプリメンテーションは MSMQ から利用できません。

Windows ソケットはネットワーク アプリケーションの記述用に低レベルの API を提供します。Windows ソケットでは、実行時間の異なる複数のアプリケーションが相互に通信することはできません。また、一時的にオフラインになるネットワークを介してメッセージを送った場合、確実な配信を保証できません。

Messaging API (MAPI)は、高レベルな API のセットで、アプリケーション間のメッセージの通信もできなくはありませんが、電子メール指向で、汎用的なメッセージ キューイング メカニズムではありません。

## 異種ネットワークの異なるプラットフォームにまたがるエンタープライズレベルの通信

MSMQ は TCP/IP と IPX/SPX ネットワークといった異種ネットワーク間でもメッセージを送信することができます。MSMQ を使用するアプリケーションなら、メッセージが通るネットワークのタイプを意識する必要はありません。

MSMQ を Windows®ファミリのオペレーティング システム上で提供する以外に、Microsoft はサードパーティと連携してほかのプラットフォームとの相互運用性も提供しています。たとえば、Level8 Systems はゲートウェイ製品の "MSMQ Gateway"を開発しています。以下は Level8 MSMQ Gateway 機能の説明です

Level8 Systems MSMQ Gateway は Windows ファミリ上の MSMQ の世界とほかのプラットフォームとの橋渡しを異なる 3つの方法でサポートしています。

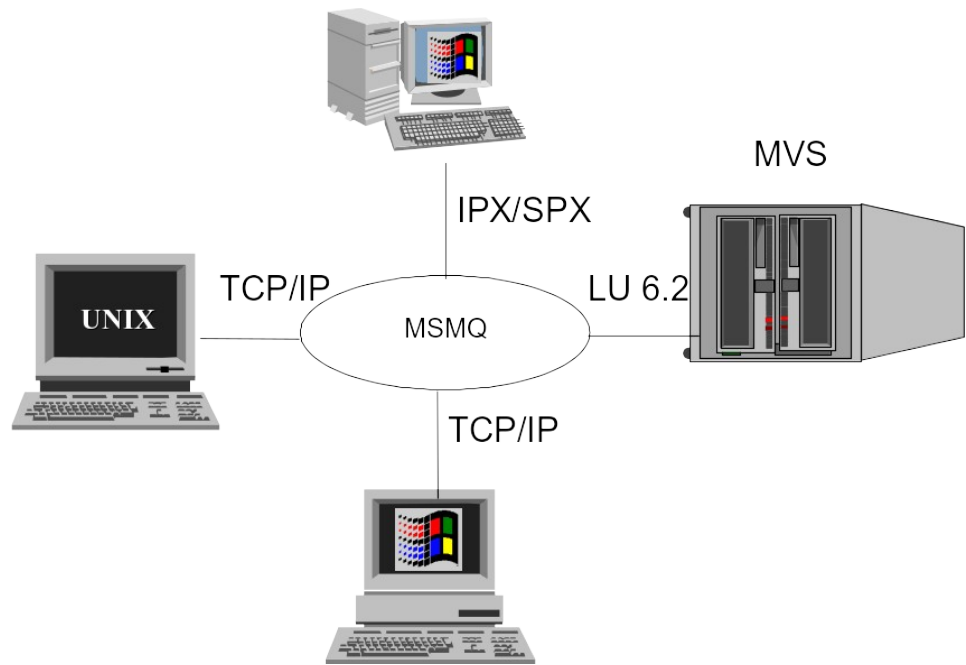
**レガシー プラットフォーム上の MSMQ API:** Level8 MSMQ Gateway は、レガシープラットフォームと MSMQ Gateway マシンを RPC ベースで直接接続する方法で、レガシープラットフォーム上で MSMQ アプリケーションプログラミング インターフェイス (API)ライブラリをサポートしています。MSMQ Gateway マシンでは、実際に MSMQ 呼び出しが呼び出し側のアプリケーションのために実行されています。

Level8 MSMQ Gateway では、IBM のメインフレーム (MVS/CICS)、Unix のプラットフォーム (Sun-Solaris、HP-Unix および AIX)、OS/2、VMS、および AS/400 をサポートすることを計画しています。

**MSMQ から IBM MQSeries Queue を介して Queue Gateway へ:** Level8 MSMQ Gateway は、IBM MQSeries のネイティブな API (MQI)呼び出しと MSMQ API とのマッピングをサポートしており、IBM MQSeries アプリケーションと MSMQ Network 間のシームレスなアクセスを実現しています。

MQSeries との接続をサポートしているプラットフォームには、MVS/CICS、AS/400、AIX、AT&T GIS、VMS、OS/2 など MQSeries 2.0 プラットフォーム全部が含まれます。

**MSMQ から CICS Transient Data Queue Gateway へ:** ネイティブな CICS Transient Data アプリケーション API と MSMQ API とのマッピングをサポートしており、ネイティブな CICS Transient Data アプリケーションから MSMQ Network へのアクセスと、MSMQ アプリケーションから CICS Transient Data 資源へのアクセスを実現しています。



**図 3 : MSMQ は異種のプラットフォームとネットワークを統合**

Level8 MSMQ Gateway のようなゲートウェイ製品なら、MSMQ を使って異種システムを統合できます。MSMQ API はさまざまなプラットフォームで利用できるため、単一の API を書くだけで多くのプラットフォームに対応できます。

カスタマはレガシー システムに対するこれまでの投資を保護しながら、革新的な PC ベースのテクノロジーを取り込んで進化していくことができます。ビジネスロジックをメインフレーム システムから Windows プラットフォームへと、段階的にダウンサイジングしていくことが可能です。Windows 上で動作する新しい MSMQ アプリケーションでは、ゲートウェイを介して別のプラットフォームのレガシー アプリケーションとそのまま通信できます。Windows 以外のプラットフォームで実行されるアプリケーションについては、Windows 以外のプラットフォームでも利用できる MSMQ API を使用するようにアプリケーションを移行してしまうか、あるいはアプリケーションは変更しないで、ネイティブな API、たとえば MQSeries API を呼び出して、ゲートウェイのマッピング機能を利用してこれらの API を MSMQ に変換する方法のいずれかを選択できます。

## 簡単にできるメッセージング アプリケーションの開発

MSMQ API を使用すれば、開発者は非同期または同期メッセージング機能を持つアプリケーションを書くことができます。このメッセージング機能は、第 1 節「メッセージのキューイング」で説明した蓄積転送方式を使ってインプリメントされています。MSMQ には、C/C++ と ActiveX コンテナ (たとえば Microsoft Visual Basic®) のための SDK のほか、テストとデバッグ用ツールも付属しています。

MSMQ が提供する簡単な API を使って、開発者は次のことができます。

- キューを作成し、レプリケートされた動的なディレクトリ サービスに登録する
- キューに関する情報を取得し、設定する
- キューを探し出す
- キューを閉じ、削除する
- メッセージをキューに入れ、キューから取り出す
- メッセージとデータベースの更新をトランザクションとしてカプセル化する
- キュー中にあるメッセージを覗き見る

MSMQ ActiveX コンポーネントには、MSMQ メッセージ キューを開始する非常に便利な方法があります。Internet Information Server Active Server Pages (ほかの VB Script ホストでも同様です) に埋め込まれた VB Scripts から MSMQ ActiveX コンポーネントを呼び出すことができるのです。

MSMQ は MAPI インターフェイスも提供しています。MSMQ MAPI プロバイダまたは MSMQ Exchange Connector を使うと、MAPI ベースのアプリケーションが MSMQ ターゲット キューを相手にメール メッセージやフォームを送受信できます。

## タイプの異なるメッセージの送信と応答の受信

MSMQ では、開発者は次のタイプのメッセージを指定できます。

- **Guaranteed delivery (配信保証)**: このタイプを指定されたメッセージは、マシンがクラッシュした場合でも、目的地に至る次の立ち寄り先まで確実に届くことが保証されます。
- **Express delivery (速達)**: インメモリの蓄積転送。マシンがクラッシュした場合を除き、このメッセージも、次の立ち寄り先まで届くことが保証されます。
- **Transaction delivery (トランザクション配信)**: このタイプのメッセージは目的のキューに確実に 1 回だけ届くことが保証されます。このタイプのメッセージは順番どおりに着信し、通常ほかのデータベース更新操作といっしょにカプセル化されアトミックな更新として送受信されます。

開発者は、送信されるメッセージごとに MSMQ から受け取りたい応答の種類を指定できます。次のタイプの応答を送信されるメッセージごとに指定できます。

- **Full acknowledgment (完全肯定応答)**: MSMQ は、送信元のアプリケーションで定義されている応答キューに肯定応答を送ります。
- **Negative acknowledgment (否定応答)**: メッセージが相手先によって正常に受け取られた場合は、MSMQ は肯定応答を送りません。失敗した場合にのみ、送信元アプリケーションのキューにエラー メッセージを送ります。
- **No notification (通知なし)**: このタイプを指定した場合は、送信元アプリケーションに肯定応答もエラー メッセージも送られません。

## 通信のセキュリティの確保

MSMQ は、Windows NT セキュリティを利用して、キューやメッセージなどのすべての MSMQ エンティティに対して認証とアクセス制御を行います。これは、アプリケーションが MSMQ オブジェクトに対して何か操作を実行しようとした場合、現在のユーザーがその操作を許可されているかどうか、そのオブジェクトに付属しているアクセス制御リスト (ACL) に照らし合わせて、必ず確認されることを意味しています。たとえば、あるキューのメッセージを読みたいユーザーは、そのキューへの Read アクセス権を持っている必要があります。

MSMQ は、X509 証明書を使用してデジタル署名に基づくメッセージの認証も行うことができます。MSMQ は、そのメッセージがメッセージに添付されている証明書のオーナーによって間違いなく署名されたものであることを確認します。受け取り側のアプリケーションはこの証明書を調べることで送信者の身元を確認できます。

MSMQ がセキュアな通信をサポートしていることを示すもう 1 つの側面は、要求に応じてメッセージ本体を暗号化できることです。メッセージ本体は、受取人となるユーザーのパブリック キーを使って暗号化されるので、当のユーザー以外のだれもこのメッセージ本体を解読できません。

MSMQ におけるデジタル署名と暗号化のサポートは、Windows NT 4.0 と Internet Explorer 3.0 for Windows 95 で提供されている Cryptographic API を使用してインプリメントされています。

## モバイル ユーザーおよびモバイル システムと通信

MSMQ を介して送られたメッセージは送信先のシステムがオンラインになった時点で確実に相手に届くことが保証されます。このことはモバイルユーザー用のアプリケーションを開発する際に威力を発揮します。モバイルユーザー用のアプリケーションを書く場合でも、開発者は、モバイルユーザーのアプリケーションに送られたメッセージが、アプリケーションがオンラインに復帰した後に確実に相手先に届くことを保証しなくて済みます。MSMQ に書き込むだけで、開発者は、メッセージがモバイルユーザーに確実に配信されるという保証が得られます。モバイルユーザーは、オンラインになっている場合でも、低速なモデムによる通信を行うのが普通です。開発者は、こうしたユーザーでも、ネットワーク要求を発行した後そのまま作業を続けられ、低速な通信リンクからの応答を待たなくて済むアプリケーションを簡単に作成できます。

モバイルユーザーはオフラインの状態でもメッセージを送信できます。これらのメッセージは MSMQ によってローカルのキューに入れられ、ユーザーがネットワークに接続したときに、相手先に向けて送信されます。

## 複合ネットワーク向けにスケーラブルなアプリケーションを開発

複合ネットワークの場合、従来のコネクション指向のネットワークング モデルでは、しばしば失敗したり、パフォーマンスの問題を起こします。これはアプリケーションが必要もないのにネットワーク パフォーマンスと緊密に結びついて

いるためです。MSMQ では、アプリケーションはメッセージの配信に使われるメカニズムから切り離されます。このことは、コネクション指向の通信がボトルネックとなってパフォーマンスに大きな影響を及ぼす大規模ネットワークや複合ネットワークで特に重要です。

企業内に分散コンピューティングが浸透するにつれ、ネットワーク上の分散したサイトで独立に実行されるアプリケーションの数はますます増えつつあります。こうしたアプリケーションは、柔軟で信頼性のある通信サービスが利用でき、簡単に通信できる環境を必要としています。ローカルとリモートにクライアントを持ち、オフラインになっている場合もあり得るネットワークやシステムを介して、時間を異にして実行される複数のアプリケーション間で効率的な通信を保証する通信メカニズムが、MIS の意思決定者にとっても同様に必要です。MSMQ はまさにそのような通信サービスであり、エンドユーザー、開発者、およびシステム管理者の生産性向上をもたらします。

## 優先順位に基づいて決められるアプリケーションの応答時間

MSMQ はメッセージを優先順位に基づいて送受信できます。この機能を利用してエンドユーザーは緊急度の低いアプリケーションの処理を後回しにして、業務上重要なアプリケーションの応答時間を短縮することができます。たとえば、重要な POS アプリケーションとより緊急度の低いニュース アプリケーションを扱っているユーザーなら、ニュース アプリケーションの応答時間を低く抑えることで、POS アプリケーションの応答時間を確実に短縮できます。

## ネットワーク障害に対する強い耐性と高いサービス可用性

MSMQ は、ネットワーク障害があっても、ネットワークが復帰後のメッセージ配信保証します。このことは、障害が起こりやすい複合ネットワークや使用率の高いネットワークでは特に重要です。ネットワークの障害があってもユーザーがメッセージを再送する必要がないということは、それだけネットワークのトラブルに強いということです。

MSMQ によって軽くなったもう 1 つの負担は、送信側のアプリケーションがメッセージの送信を始めたときに、メッセージの表示などの処理を行うために、受信側のアプリケーションも起動したり、使用可能にしなければならなかったことです。MSMQ では、受信側アプリケーションは、後の適当な時間にメッセージを目的のキューから取り出して読むことができ、応答もそのときにすればよくなります。

MSMQ を使用することで、システム管理者はネットワーク管理の効率を高めることができます。アプリケーションにとって、メッセージを MSMQ に渡すだけで、メッセージの確実な配信が保証されることになります。これは、メッセージの送信先が現在オフライン中のリモート ユーザーであっても同様です。システム管理者にとって、ローカルのオンライン システムを扱う手軽さでオフラインのリモート システムを管理することができ、発行したコマンドもリモート ユーザーがネットワークに接続した時点で確実にリモート システムに送られます。

## 簡単にできる MSMQ のインストールとセットアップ

**MSMQ サーバー:** MSMQ を使用する場合、システム管理者は、MSMQ 資源をパブリッシュするための動的なリポジトリとなる MSMQ Information Service をインストールするために、企業内の 2、3 のサーバーに MSMQ Server をインストールする必要があります。MSMQ Routing Server 機能はサイトのトポロジーに合わせてインストールします。なお、これらの MSMQ 機能をセットアップするには、既存の Windows NT Server システムが利用できます。専用サーバーは必要ありません。

**MSMQ クライアント:** MSMQ ランタイムは MSMQ ベースの通信に参加するすべてのノードにインストールする必要があります。インストールは、MSMQ の無人セットアップ オプションを使って自動的に行なえ、セットアップ時に選択を要求される項目はあらかじめサイト管理者のによって設定されています。

**管理:** MSMQ には Message Queue Explorer と呼ばれる管理ツールが提供されています。この管理ツールをネットワーク上の 2、3 のノードにインストールします。

**Software Development Kit:** SDK は MSMQ サービスを必要とするアプリケーションの開発に使われるシステムにインストールします。

## Message Queue Explorer を使ってネットワークを管理

Message Queue Explorer では、システム管理者は次のことが行なえます。

- ルーティングの構成を動的に変更する
- パフォーマンスを分析する
- ネットワーク トポロジーを調べる
- メッセージの緊急度に基づいてネットワーク トラフィックを制御する
- 請求、課金、およびトラブルシューティングのためにログを調べる

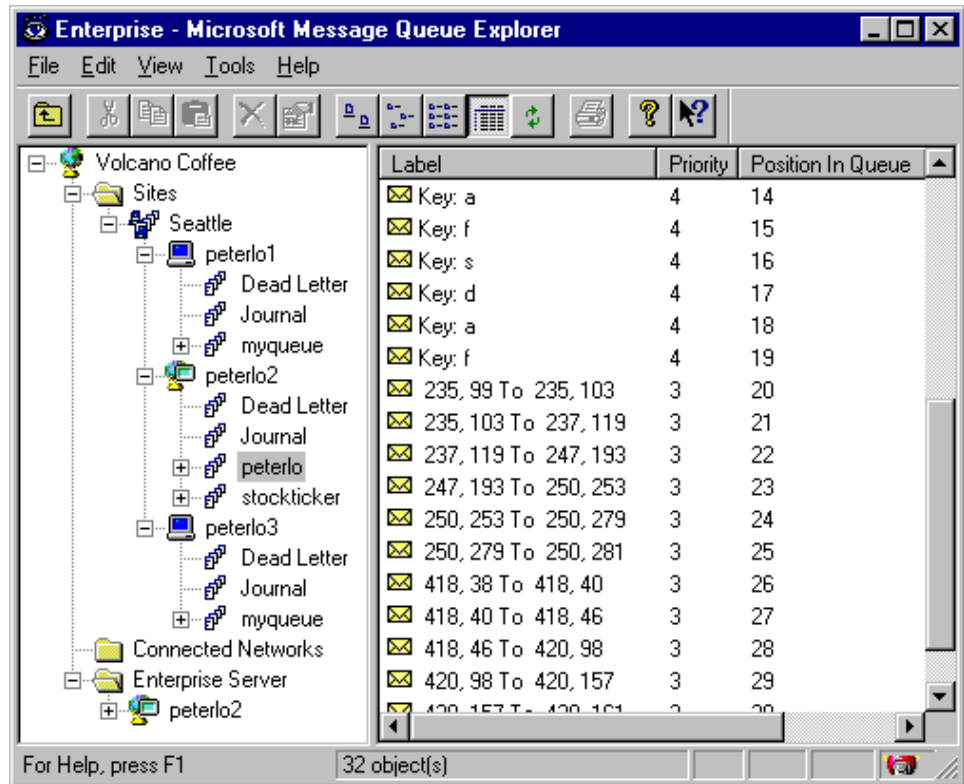


図4 : MSMQ Explorer

メッセージを効率よくルーティング

MSMQは、使用されるネットワークプロトコルにかかわらずメッセージを送信先にルーティングできます。MSMQは非同期メッセージングをサポートしているため、送信先がオフラインになっていると、メッセージは送信先に最も近接した場所に回付されます。MSMQは基盤となるプロトコルのルーティング機能を拡張し、より高度なルーティングを提供します。

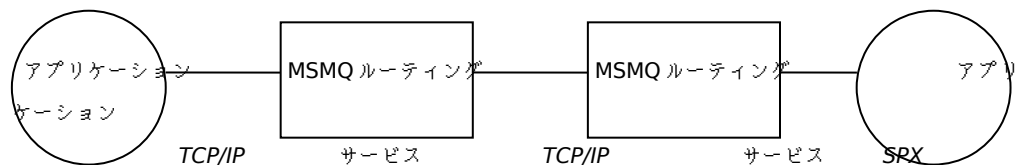


図5 : MSMQ は異種プロトコル間のルーティングを提供



## ビジネス ニーズの変化に容易に対応

MSMQ を使用するアプリケーションでは、ほかのアプリケーションと通信するためのインターフェイスや通信モード(メッセージ)が汎用性を持っています。したがって、ビジネスのニーズが変化した場合でも、アプリケーションは同じ通信メカニズムを引き続き使用できます。アプリケーションにとっては、ビジネスに  
関係する部分のみ変更すればよく、プログラムをビジネス ニーズの変化に容易に対応させることができます。

## ソフトウェア開発とメンテナンスの時間を短縮

開発者はビジネス向けのプログラミングに専念でき、ネットワーク周りのプログラミングに時間を取られなくなります。MSMQがあれば、ネットワーク アプリケーションは簡単に書いてしまうからです。また、MSMQ なら、オフラインのネットワークやシステムにまたがって通信する場合でも、実行時間を異にするアプリケーションにメッセージを確実に配信できます。このように配信が保証されるメッセージングシステムを必要とする場合でも、開発者は作成するアプリケーションにこうした機能を組み込む必要がありません。MSMQ がその機能を受け持ってくれます。MSMQ は複数のプラットフォームで利用できるため、開発者はソースコードベースを1つだけ作成し、保守すればよく、結果的に開発とメンテナンスの時間を短縮できます。

## 回線コストの節約

MSMQ はメッセージを非同期に伝送します。MSMQ では、通信コストが高くつくときは使用する回線数を抑さえ、伝送コストが安くなるまでメッセージを保持することができます。結果的に、アプリケーションは少ない回線を上手に使用することになり、コスト削減がもたらされます。