

Sun Java™ System Message Queue

リリースノート

バージョン 3.5 SP1

Part No. 817-7201

このリリースノートには、Sun Java™ System Message Queue (旧 Sun™ ONE Message Queue) バージョン 3.5 SP1 のリリース時点で得られる重要な情報が含まれています。さらにこのドキュメントには、3.5 以前のバージョンからアップグレードするユーザー向けの、Message Queue 3.5 リリースノートの内容も含まれています。Message Queue 3.5 バージョンに関する、新しい機能や拡張機能、既知の制限と問題、技術的な注意点、およびその他の情報が記載されています。

このリリースノートの最新版は、次の Sun Java System ドキュメント Web サイトから入手できます <http://docs.sun.com/db/prod/entsys?l=ja>。ソフトウェアのインストールおよび設定前だけでなく、以後も定期的にこの Web サイトをチェックして、最新版のリリースノートやマニュアルをご覧ください。

このリリースノートには次の節があります。

- [改訂履歴](#)
- [Message Queue 3.5 SP1 について](#)
- [修正されたバグ](#)
- [重要な情報](#)
- [既知の問題および制約](#)
- [再配布可能ファイル](#)
- [問題の報告方法とフィードバックの提供方法](#)
- [Sun の追加情報](#)

改訂履歴

表 1 改訂履歴

日付	変更内容の説明
2004 年 3 月 12 日	バグ情報の更新。「既知の問題および制約」の節の更新 「再配布可能ファイル」の節の追加。「ドキュメントの更新」の節の更新。「互換性の問題」の節の更新。「Sun Java System の情報」の節の更新
2004 年 1 月 9 日	PointBase 4.8 バージョンサポートの更新情報および C-API 機能の更新情報

Message Queue 3.5 SP1 について

Message Queue 3.5 SP1 は、Message Queue 3.5 を更新したもので、Message Queue 3.5 の新機能がすべて含まれています。さらに、Message Queue 3.5 SP1 ではバグが修正され、ブランド名も新しくなりました。現在、この製品は Sun Java™ System 製品ファミリに属しています。

Message Queue 3.5 SP1 は、JMS 1.1 互換性テスト群 (CTS) に合格し、Java™ Message Service (JMS) 1.1 仕様への準拠が認定されています。

この節では、Message Queue 3.5 SP1 での変更点、および以前のリリース、Message Queue 3.5 での変更点を説明しています。

Message Queue 3.5 SP1

Message Queue 3.5 SP1 では、バグが修正され、製品とドキュメントのブランド名が新しくなりました。

Message Queue 3.5

Message Queue 3.5 には、多くの新機能が含まれています。

- 3 ページの「C クライアントサポート (Enterprise Edition)」
- 4 ページの「Java クライアント接続のフェイルオーバー (Enterprise Edition)」
- 5 ページの「ブローカメッセージのフロー制御の強化」
- 6 ページの「Java クライアントのフロー制御の強化」
- 6 ページの「新しい送信先メトリックス」
- 7 ページの「リモート監視 API (Enterprise Edition)」
- 7 ページの「Message Queue の JMS 用リソースアダプタ (J2EE アプリケーションサーバーサポート)」
- 8 ページの「カスタムメッセージの通知」
- 8 ページの「キュー配信ポリシーの強化 (Enterprise Edition)」
- 9 ページの「クラスタパフォーマンスの強化 (Enterprise Edition)」
- 10 ページの「ローカル送信先 (Enterprise Edition)」
- 10 ページの「安全なブローカクラスタ (Enterprise Edition)」
- 10 ページの「持続ストアパフォーマンスの強化」
- 11 ページの「インスタンス固有の認証と承認」
- 12 ページの「RPM ベースの Linux のインストール」
- 12 ページの「Solaris オペレーティングシステム、x86 版のサポート」

以降の節で、これらの機能について説明しています。

C クライアントサポート (Enterprise Edition)

Message Queue 3.5 には、C API および C ランタイムサポート (これ以後 C クライアント機能と呼ぶ) が含まれています。C クライアント機能は、以前のシステムを Message Queue メッセージングシステムに統合する際に使用できます。JMS 仕様がほとんどすべて実装されています。次の内容を除くすべての JMS 機能をサポートします。特定の本体のタイプ (マップ、ストリーム、オブジェクトなど)、キューブラウザの機能、およびアプリケーションサーバーの機能 (分散トランザクションや ConnectionConsumer オブジェクトなど)。

C クライアント機能のサポートは、別個にインストールされるライブラリのセットによって行われ、Enterprise Edition のライセンスでのみ使用可能です。したがって、Platform Edition から Enterprise Edition へアップグレードする場合、Enterprise Edition のライセンスファイルと C ライブラリの両方をインストールする必要があります。

有効な 90 日間の Enterprise Edition トライアルライセンスを所持する Platform Edition のユーザーは、imq-feedback@sun.com の電子メールアドレスを使用して Sun に連絡し、C-API SDK をリクエストすると、C クライアント機能を使用できます。技術担当者がこれらのリクエストに対応し、匿名 FTP サイトから C-API SDK を入手できるようにします。90 日間の Enterprise Edition ライセンスの期限が切れた後、ユーザーは引き続き C クライアントを構築できますが、C クライアントを Platform Edition ブローカーに接続することはできなくなります。

C クライアント機能では、それぞれのオペレーティングシステムプラットフォームに対応した特定のコンパイラバージョンが必要になり、新しいシステム要件が Enterprise Edition に追加されることになります (詳細は、『Message Queue インストールガイド』を参照)。さらに C クライアント機能は、Netscape Portable Runtime (NSPR) および Network Security Service (NSS) の各ライブラリと依存関係にあります。Message Queue 3.5 の C クライアント機能は、Linux Red Hat Advanced Server 2.1 での正常動作がテストされています。テストされた NSPR と NSS ライブラリのバージョンは、Linux 版では認定されていません。

今回、C-API は、基本認証タイプをサポートしていません。この認証タイプを使用するようにブローカーを設定すると、MQCreateConnection 関数への呼び出しは失敗して、MQ_UNSUPPORTED_AUTH_TYPE という結果になります。

C クライアント関数のマニュアルには、参照マニュアルとプログラミングマニュアル、および C-API クライアントの例の両方が含まれています。詳細は、『Message Queue C Client Developer's Guide』を参照してください。

Java クライアント接続のフェイルオーバー (Enterprise Edition)

Message Queue 3.5 では、障害が発生した接続を、元のブローカー上だけではなく、異なるブローカー上でも復元できる、強化された自動再接続機能をサポートしています (クライアント接続のフェイルオーバー)。再接続は、特定のブローカーインスタンスに対するものではなく、メッセージサービスに対するものです。この動作を実装するには、接続ファクトリ管理対象オブジェクトを設定して (Message Queue 3.5 には新しいメッセージサービスのアドレス指定スキームがある)、ブローカーアドレスのセットを指定します (`imqAddressList`)。クライアントランタイムがメッセージサービスへの接続の確立または再確立を必要とする場合、使用可能なブローカーを見つけるまでまたは見つけることに失敗するまで、リストのブローカーに優先順で接続を試みます。それぞれのブローカーに、接続を試みる回数 (`imqAddressListIterations`) と、接続を試みる間隔 (`imqAddressListInterval`) を指定できます。

自動再接続が元のものとは異なるブローカーインスタンスに対しての場合、障害が発生した (または接続が解除された) ブローカーによって保持される持続メッセージや他の状態情報は失われます。これは、クラスタの種々のブローカーインスタンスが共有の、高可用性の持続ストアを使用していないためです。しかし、自動的に異なるブローカーインスタンスに再接続するクライアントランタイムの機能によって、バックアップブローカーまたはブローカークラスタを (完璧ではないが) フェイルオーバー保護に使用できるようにする、復元シナリオを作成できます。

さらに自動再接続が有効で、関連する接続が失敗する場合、Message Queue 3.5 は一時送信先を持続します。これは、クライアントが一時送信先に再接続および再アクセスする可能性があるためです。一時送信先は他の物理的な送信先のように扱われるので、未使用のあらゆる一時送信先のブローカを定期的にページすることが必要な場合もあります。

詳細は、『Message Queue Java Client Developer's Guide』を参照してください。

Message Queue は以前のリリースでも自動再接続機能をサポートしており、接続が失敗した場合にクライアントランタイムは自動的にブローカに再接続することはできました。しかし、再接続でクライアント側の状態をブローカ上で完全に復元できない場合、たとえば、接続が確立されている間だけ存在する、処理済みセッションや一時送信先を使用する場合は、再接続できませんでした。

ブローカメッセージのフロー制御の強化

送信先へのメッセージのフローの制御を改善し、メッセージの生成がその消費よりずっと速いという状態を防止するために、ブローカの機能が強化されています。さらに他の新しい Message Queue 3.5 の機能も、送信先からのメッセージのフローのボトルネックを除去するのに役立つ場合があります。[6 ページの「Java クライアントのフロー制御の強化」](#)および[8 ページの「キュー配信ポリシーの強化 \(Enterprise Edition\)」](#)を参照してください。

ブローカのメッセージフローの強化には次の点が含まれます。

- 送信先に関連付けられたプロデューサの数に制限値を設定します。送信先は、新しい `maxNumProducers` 属性を持ち、この制限値に達する場合は、新しいプロデューサを作成できません。
- 送信先が `maxTotalMsgBytes` および `maxNumMsgs` の制限値に達する場合の、新しい、設定可能な制限値および制限時の動作を有効にします。特に、次の変更が実装されています。
 - `maxTotalMsgBytes` および `maxNumMsgs` の送信先属性がトピック送信先にも拡張された (以前はキュー送信先だけに適用)
 - 自動作成の送信先に対して、`maxTotalMsgBytes` および `maxNumMsgs` の設定が有効
 - (Enterprise Edition) 管理者が、上記の制限値のいずれかに達した場合の応答動作を選択可能。次の動作を指定できる。プロデューサの減速化 (FLOW_CONTROL)、最も古いメッセージの破棄 (REMOVE_OLDEST)、メッセージの有効期限に従って優先順位が最低のメッセージの破棄 (REMOVE_LOW_PRIORITY)、または最新のメッセージの破棄 (REJECT_NEWEST) か、あるいはこれらすべての動作
 - 接続ベースではなく、プロデューサごとに、プロデューサフロー制御の制限動作 (FLOW_CONTROL) を実装。(以前の実装では、1 つの送信先が接続を介して、過大な数のメッセージを受信した場合、すべてのプロデューサがシャットダウンされた)。プロデューサごとのフロー制御によって、過大な数のメッセージの送信先に関連付けられた接続のプロデューサだけがシャットダウンされ、接続された他のプロデューサは他の送信先にメッセージを送信し続けることができる

- 管理者は、特定の送信先を停止および再開できます。プロデューサから送信先へのまたは送信先からコンシューマへの、またはその両方のメッセージの配信を停止できます。これは、新しい次の 2 つの `imqcmd` サブコマンド、`pause` および `resume` で行われます。
 - `imqcmd pause dst -n myQueue -t q -pst PRODUCERS`
 - `imqcmd resume dst -n myQueue -t q`

詳細は、『Message Queue 管理ガイド』を参照してください。

Java クライアントのフロー制御の強化

Message Queue 3.5 クライアントランタイムは、接続ベースごとのほか、コンシューマごとでもメッセージのフローを管理します。コンシューマごとにバッファされるメッセージの数に制限値を設けることができ、それによって、いずれのコンシューマも他のコンシューマによって処理が困難になる事態を防止できます。さらにこの機能によって、複数のコンシューマに対してキューが配信される場合、複数のコンシューマ間でメッセージの配信バランスを改善できます。Message Queue クライアントランタイムのメモリーリソースの管理にも役立ちます。

新しい接続ファクトリ属性、`imqConsumerFlowLimit` は、同じ接続を共有するすべてのコンシューマで、コンシューマごとにバッファされるメッセージの数を制限します。コンシューマバッファのメッセージの数が `imqConsumerFlowLimit` のしきい値の割合 (`imqConsumerFlowThreshold`) 未満の場合、ブローカーはそのコンシューマが消費する別のメッセージのまとまりをクライアントランタイムに配信できます。1 つの接続ですべてのコンシューマ用にバッファされるメッセージの総数が、`imqConnectionFlowLimit` を超過する場合、その接続を介してのメッセージの配信は、総数が接続制限数未満になるまで中止されます。

クライアントランタイムフロー制御の以前の実装では、クライアントランタイムでバッファされるメッセージの数に制限数を設け、消費されるのを待機していました (`imqConnectionFlowLimit`)。この機能の目的は、ゆっくりと消費するクライアントがメモリーの不足でクラッシュしないように、メッセージのバッファに使用されるクライアントメモリーの量を制限するためでした。この機能は接続レベルで実装され、1 つの接続で多数のコンシューマをサポートする場合、1 つのコンシューマにメッセージが殺到すると、他のコンシューマによるメッセージの受信を妨害する可能性があります。

詳細は、『Message Queue Java Client Developer's Guide』を参照してください。

新しい送信先メトリックス

Message Queue 3.5 では、送信先ごとにメッセージとコンシューマの追跡が強化され、メモリーと使用量の監視と制御が改善されました。

新しいメトリックスは、新しい `imqcmd metrics dst` サブコマンドの出力として表示されます。このコマンドによって、累積合計 (サンプリングの開始から)、現在値、平均値 (取得されたサンプルから計算)、メッセージおよびコンシューマメトリックスの両方の最大値 (サンプリングの開始から) が表示されます。

たとえば、`imqcmd metrics dst -m ttl` コマンドは次の情報を返します。

- メッセージフロー
 - メッセージフロー入力: 累積合計、レート
 - メッセージフロー出力: 累積合計、レート
- ブローカで格納されるメッセージ (メモリーや持続ストア内の通知されていないメッセージ)
 - メッセージ数: 現在、最大、平均
 - メッセージのバイト数: 現在、最大、平均
- これまでの最大メッセージ、バイト数

`imqcmd metrics dst -m con` コマンドは次の情報を返します。

- アクティブコンシューマの数: 現在、最大、平均 (8 ページの「[キュー配信ポリシーの強化 \(Enterprise Edition\)](#)」を参照)
- バックアップコンシューマの数: 現在、最大、平均 (8 ページの「[キュー配信ポリシーの強化 \(Enterprise Edition\)](#)」を参照)

詳細は、『[Message Queue 管理ガイド](#)』を参照してください。

リモート監視 API (Enterprise Edition)

Message Queue 3.5 には、リモート (またはローカル) JMS クライアントが簡単にブローカメトリックスの監視や分析を行える、メッセージベースの API が装備されています。この API は、ブローカ自体、Java VM、および個々の送信先についてのメトリックス情報を含む、メッセージを生成できるブローカに基づいています (6 ページの「[新しい送信先メトリックス](#)」を参照)。1 つまたは複数のコンシューマがこれらの送信先に登録されると、監視されているエンティティに応じて、これらのメッセージは特定のトピック送信先に送信されます。そのあと消費するクライアントは、メッセージの取得、ヘッダプロパティ (type) を使用したメッセージのフィルタ化、そして含まれるメトリックス情報の抽出を行うことができます。

詳細は、『[Message Queue 管理ガイド](#)』と『[Message Queue Java Client Developer's Guide](#)』の両方を参照してください。

Message Queue は以前、Message Queue 管理ユーティリティを使用した、ブローカメトリックスデータのローカルログ作成とメトリックス情報のリモートクエリだけをサポートしていました。これらの機能は、重要なメトリックスデータを提供していましたが、このデータを簡単に分析するには便利ではありませんでした。

Message Queue の JMS 用リソースアダプタ (J2EE アプリケーションサーバーサポート)

Message Queue 3.5 には、Message Queue JMS メッセージサービスを互換性のある任意の J2EE アプリケーションサーバーにプラグインするための JMS リソースアダプタが含まれています。

リソースアダプタは、J2EE Connector Architecture Specification (JCA 1.5) に準拠し、付加的な機能を J2EE アプリケーションサーバーにプラグインする (EIS、メッセージシステムなどに接続することによって) ための標準的な方法です。このアーキテクチャによって、J2EE アプリケーションサーバーは、たとえば、JCA 1.5 を実装する任意の JMS プロバイダに接続することで、JMS メッセージングをサポートできます。つまり、アプリケーションサーバー環境で、配備および実行される J2EE コンポーネントは、プラグイン JMS プロバイダ (クライアントランタイムとサーバー) を使用して、JMS メッセージを交換できます。

詳細は、『Message Queue 管理ガイド』を参照してください。

カスタムメッセージの通知

Message Queue は現在、JMS `CLIENT_ACKNOWLEDGE` クライアント通知モードをサポートしており、JMS クライアントは、この通知モードでメッセージの消費を明示的に通知します。`CLIENT_ACKNOWLEDGE` モードでは、クライアントがメッセージオブジェクトの `acknowledge()` メソッドを呼び出し、そのセッションが、メソッドの以前の呼び出しからセッションが消費してきたすべてのメッセージを通知するようにします。

Message Queue 3.5 では、ユーザーが個々のメッセージを通知することで、この動作を強化しています。つまり、ユーザーは、そのときまでに消費されたすべてのメッセージをバッチ処理として通知するのではなく、特定のメッセージだけを通知できます。この処理は、コード上で新しい `acknowledge()` メソッドを呼び出す、特別な Message Queue メッセージタイプにメッセージオブジェクトをキャストすることによって実行されます。これにより、JMS の標準規格から外れた、特別なアプリケーションの要求に対応できます。

詳細は、『Message Queue Java Client Developer's Guide』を参照してください。

キュー配信ポリシーの強化 (Enterprise Edition)

複数のコンシューマにキューの配信を実装する場合、以前は、3 つの別個のキュー配信ポリシー (シングル、フェイルオーバー、ラウンドロビン) として実装されていましたが、変更されました。Message Queue 3.5 では、配信の際に設定可能な数のアクティブおよびバックアップコンシューマの間でロードバランスが行われる、より一般化された方法を使用します。Message Queue 3.5 の実装は、新しい次の送信先属性に基づいて行われます。

- `maxNumActiveConsumers`: ロードバランスされたキューの配信で 1 つまたは多数のアクティブなコンシューマの数を指定する
- `maxNumBackupConsumers`: いずれかのアクティブコンシューマに障害が発生した場合に、アクティブコンシューマの代わりが可能な、ゼロまたは多数のバックアップコンシューマの数を指定する

コンシューマの数がこれら 2 つの属性の合計を超過する場合には、新しいコンシューマは拒否されます。

Message Queue Platform Edition は、ロードバランスされたキュー配信を最大 2 つまでのコンシューマに対してサポートし、Enterprise Edition では、サポートされるコンシューマの数に制限がありません。

この新しいロードバランスメカニズムには、異なるコンシューマのメッセージ消費速度が考慮に入れています。これは次のように動作します。

- 送信先の最初のキューメッセージは、設定可能なサイズ (送信先の `consumerFlowLimit` 属性) のバッチ処理として、使用可能なアクティブコンシューマに (送信先に登録された順序で) ルーティングされます。いったんこれらのメッセージが配信されると、送信先に着信した追加のメッセージも、コンシューマが使用可能になる (つまり、コンシューマが以前に配信したすべてのメッセージを通知する) にしたがって、1 つずつコンシューマにルーティングされます。アクティブコンシューマに障害が発生した場合には、最初のバックアップコンシューマがアクティブになり、障害が発生したコンシューマの動作を引き継ぐことになります。
- ブローカクラスタ環境の配信メカニズムの場合、ローカルコンシューマを優先するように設定できます。新しい送信先属性、`localDeliveryPreferred` では、送信先が生成されたローカルブローカにコンシューマがない場合に限り、ユーザーはメッセージをリモートコンシューマに配信するよう指定できます。これにより、(それぞれのホームブローカ経由での) リモートクライアントへのルーティングがスループット低下の原因となりかねない状態で、パフォーマンスを向上させることができます。この属性では送信先の範囲がローカルのみでの配信に制限されない必要があります。9 ページの「[クラスタパフォーマンスの強化 \(Enterprise Edition\)](#)」を参照してください。

詳細は、『[Message Queue 管理ガイド](#)』を参照してください。

クラスタパフォーマンスの強化 (Enterprise Edition)

ブローカクラスタ環境では、送信先がすべてのブローカで複製され、それらの送信先に配信されたすべてのメッセージは、その送信先用として登録されたコンシューマを持つすべてのブローカに転送され、メッセージのほんのわずかな割合しか、所定のいずれかのコンシューマに配信されない場合でさえこの転送が発生します (たとえば、選択基準を使用した永続サブスクリバやロードバランスされたキュー配信に含まれるキューレシーバの場合など)。このブローカ間のトラフィックにより、特に新しいコンシューマがアクティブになる際、メッセージが殺到する原因となる場合があります。クラスタのブローカ間の過大なトラフィックを削減するため、Message Queue 3.5 では次の強化機能を導入しています。

- コンシューマ接続へのメッセージの配信を調整する新しいフロー制御メカニズムの採用しています。つまり、コンシューマは、送信先からクライアントランタイムへのメッセージの配信を調整し、ブローカ間の不必要なメッセージの受け渡しを防止します。これらのメカニズムはクライアントランタイム側での停滞防止にも役立ちます。6 ページの「[Java クライアントのフロー制御の強化](#)」を参照してください。
- ブローカ間の不必要なメッセージの受け渡しを削減するために、実装をキュー配信から複数のコンシューマへと変更します (8 ページの「[キュー配信ポリシーの強化 \(Enterprise Edition\)](#)」を参照)。この実装には新しいキュー送信先属性、`localDeliveryPreferred` も含まれ、これによって、複数のコンシューマへのキュー配信でリモートコンシューマよりもローカルコンシューマを優先するように指定できます (8 ページの「[キュー配信ポリシーの強化 \(Enterprise Edition\)](#)」を参照)。

詳細は、『[Message Queue 管理ガイド](#)』を参照してください。

ローカル送信先 (Enterprise Edition)

新しい送信先属性、`isLocalOnly` によって、クラスタの他のブローカに接続されたコンシューマとは対照的に、ローカルコンシューマ (送信先が作成されたブローカに接続されたコンシューマ) だけにメッセージの配信を限定するよう、送信先を指定できます。同様に、ローカルプロデューサから送信されたメッセージだけを受信することもできます。このプロパティを使用して、クラスタ内の異なるブローカに同じ名前の独立した、非対話の送信先を作成し、それらの 1 つに障害が発生する場合に備えて、メッセージを 2 つの送信先に送信することにより、フェイルオーバーに類似したシナリオを設定できます。

詳細は、『Message Queue 管理ガイド』を参照してください。

安全なブローカクラスタ (Enterprise Edition)

クラスタとメッセージサーバー間で、安全な暗号化メッセージの配信が必要な状況に備えて、Message Queue 3.5 からは、クラスタ内のブローカ間メッセージの安全な配信をサポートします。クラスタ内で、安全な暗号化メッセージの配信を確保するには、クラスタ内部の接続を、SSL ベースのトランスポートプロトコルを使用するよう設定する必要があります。

詳細は、『Message Queue 管理ガイド』を参照してください。

持続ストアパフォーマンスの強化

Message Queue 単層型ファイルデータストアと JDBC 互換データストアの両方の実装が、パフォーマンスの向上のために Message Queue 3.5 では変更されています。これらの機能の強化については後続の 2 つの節で説明しています。詳細は、『Message Queue 管理ガイド』を参照してください。

組み込み持続性 (単層ファイルデータストア)

Message Queue 単層ファイルデータストアのパフォーマンスの強化には、Message Queue 3.01 (またはそれ以前のバージョン) から Message Queue 3.5 へアップグレードのさいに表面化する、データフォーマット内部の変更が関係しています。

Message Queue 3.5 ブローカインスタンスが最初に起動し、ファイルストアの以前のバージョンを参照すると、ファイルストアの移行が自動的に処理されます。以前のファイルストアのコピーはインスタンスディレクトリに保存され、移行が完了すると、手動でそのコピーを削除する必要があります。ストアの 2 つのコピー用に十分なディスク容量がない場合、以前のファイルストアを自動的に削除するには、次のコマンドで示すように、オプションを追加してバージョン 3.5 のブローカを起動します。

```
imqbrokerd -upgrade-store-nobackup
```

(`upgrade-store-nobackup` には空白なし)

新しい単層型ファイルデータストアのルートが次の、

```
.../instances/instanceName/filestore/
```

から次へと変更されます。

```
.../instances/instanceName/fs350/.
```

さらに、Message Queue Command ユーティリティ (imqcmd) は、ファイルストアメトリックスを提供するため、次のように強化されています。

```
imqcmd metrics dst -n destName -t type -m dsk
```

そして imqcmd には、次の新しいコンパクトコマンドが含まれています。

```
imqcmd compact dst -n destName -t type
```

プラグイン持続性 (JDBC™ 互換データストア)

Message Queue 3.5 では、Message Queue JDBC 互換データストアが変更され、ブローカメモリー管理の機能の強化 (5 ページの「ブローカメッセージのフロー制御の強化」および 6 ページの「新しい送信先メトリックス」を参照) とより広範囲のデータベースベンダーをサポートしています。メモリー管理の機能強化 (送信先によるメッセージの分類) をサポートするための修正は、透過的に処理されるスキーマの変更ですが、データベースベンダーの追加をサポートするには、プラグイン持続性の設定方法の変更が必要になります。

Message Queue 3.5 ブローカインスタンスが、最初に持続ストアの以前のバージョンで起動されると、JDBC 互換ストアの移行が自動的に処理されます。以前のテーブルはそのままの状態に保たれますが、移行が完了したなら、新しい imqdbmgr delete oldtbl コマンドを使用して手動で削除する必要があります。ストアの 2 つのコピー用に十分な容量がない場合、以前のテーブルを自動的に削除するには、ブローカを次の新しいオプションで起動します。imqbrokerd -upgrade-store-nobackup

付加的なデータベースサポートの場合、以前は、プラグインする JDBC 互換データベースへの SQL ステートメントをカスタマイズできませんでした (SQL ステートメントは内部で生成された)。Message Queue 3.5 では、Message Queue データベーススキーマを作成する SQL コードのカスタマイズを可能にする、新しいインスタンス設定プロパティが含まれています。各データベーステーブルには設定可能なプロパティがあり、そのプロパティはテーブルを作成する SQL コードです。このプロパティは、プラグインデータベースで使用するデータ型を適切に指定するために必要です。以前の Cloudscape データベースではなく、PointBase の組み込みデータベースに基づく例が提供されています。

インスタンス固有の認証と承認

デフォルトで、Message Queue 3.5 は各インスタンスが専用のファイルベースのユーザーリポジトリと専用のアクセス制御ファイルを持つようにし、両方とも次の標準の場所に配置されます。

```
.../instances/instanceName/etc/
```

この 2 つのファイルは、ブローカインスタンスが最初に起動する際に作成されます。ブローカがこれらのファイルを以前の場所で発見した場合 (通常、以前のバージョンからのアップグレードの場合)、そのファイルをインスタンス固有の場所にコピーします。ブローカがこれらのファイルを以前の場所で発見しない場合 (通常、新規のインストール場合)、そのファイルのデフォルトバージョンをインスタンス固有の場所に配置します。

インスタンス固有のユーザーリポジトリをサポートするため、それぞれの imqusermgr コマンドの適用対象となるインスタンス固有のユーザーリポジトリを指定する、-i instanceName オプションがユーザーマネージャユーティリティ (imqusermgr) に追加されました。

詳細は、『Message Queue 管理ガイド』を参照してください。

以前、デフォルトでは、1 台のコンピュータのブローカの全インスタンスが、同じファイルベースのユーザーリポジトリ (したがって、同じクライアントログインパスワード) と同じアクセス制御ファイルを共有していました。しかし、インスタンス設定ファイルに指定されている、特定の LDAP ユーザーリポジトリの場所または特定のアクセス制御ファイルを使うように、各ブローカを設定することはできました。

RPM ベースの Linux のインストール

Linux での Message Queue 3.5 のインストールは、ソフトウェアパッケージ (RPM) のインストール、アンインストール、検証、クエリ、および更新の各機能を備えた、コマンドライン駆動型のパッケージ管理システムである Red Hat Package Manager (RPM) を使用して行われます。

さらに、Linux にインストールされる Message Queue のディレクトリ構造は、Linux アンバンドル製品で使用されている標準の場所に対応するよう変更されます。(Solaris™ と Linux の両プラットフォームには、それぞれの製品がオペレーティングシステムにバンドルされるか、バンドルされないかに応じた標準設定がある) 特に Solaris での状態と同様、Linux でもルート Message Queue インストールディレクトリはなくなりました。

詳細は、『Message Queue インストールガイド』を参照してください。

Solaris オペレーティングシステム、x86 版のサポート

Solaris 9 の場合、Message Queue 3.5 は SPARC プロセッサに加えて x86 プロセッサもサポートします。

ハードウェアおよびソフトウェア要件

このリソースに必要なハードウェアとソフトウェア、およびサポートされる製品やプラットフォームの詳細については、『Message Queue インストールガイド』を参照してください。

修正されたバグ

この節では、修正されたバグを次のとおり簡潔に説明しています。

- 13 ページの表 2 では、Message Queue 3.5 SP1 で修正されたバグを説明しています。
- 14 ページの表 3 では、Message Queue 3.5 で修正されたバグを説明しています。

それ以前に修正されたバグのリストについては次を参照してください。

- Message Queue 3.0.1 Service Pack 2 については、次のサイトで入手可能な『Message Queue 3.0.1 SP 2 リリースノート』を参照してください。
<http://docs.sun.com/db/doc/817-3827-10?l=ja&q=3.0.1>
- Message Queue 3.0.1 については、次のサイトで入手可能な『Message Queue 3.0.1 Release Notes』を参照してください。
http://docs.sun.com/coll/S1_MessageQueue_301
- Message Queue 3.0 については、次のサイトで入手可能な『Message Queue 3.0 Release Notes』を参照してください。
http://docs.sun.com/coll/S1_MessageQueue_30

修正されたバグの詳細については、次の Java Developer Connection サイトで報告全体を見ることができます。

<http://developer.java.sun.com/developer/bugParade>

Message Queue 3.5 SP1 で修正されたバグ

表 2 は、Message Queue 3.5 SP1 で修正されたバグの一覧と説明です (14 ページの表 3 は Message Queue 3.5 で修正されたバグの一覧と説明)。

表 2 Message Queue 3.5 SP1 で修正されたバグ

バグの番号	説明
4942723	共有スレッドプールオプションで大きなメッセージを送信するとブローカがメモリー不足になることがある
4944894	共有スレッドプールを使用すると、場合によってはブローカが <code>CancelledKeyException</code> を生成することがある
4947239	プロデューサを作成して閉じることを繰り返すとクライアントメモリーが少量増大する
4947993	アクティブな永続サブスクライバのある送信先または永続サブスクリプション破棄できない

表 2 Message Queue 3.5 SP1 で修正されたバグ (続き)

バグの番号	説明
4948525	メッセージの入力バイトとメッセージの出力バイトのメトリックス出力に負の数字が表示されることがある。このことは、2143510810 バイト以上が送信される場合に発生する
4948563	パケット変換 : 3.5 ブローカに送信される各 2.0 SP1 メッセージに、INFO メッセージが表示される。2.0 SP1 クライアントがメッセージを 3.5 ブローカに送信するたびに、次の INFO レベルメッセージが表示される。 [04/Nov/2003:10:34:16 PST] Internal Error: Unknown ProducerUID 0
4949781	ブローカの起動時に、クラスタブロードキャストエラーを使用できない
4952332	プライマリコンシューマが同じ接続のバックアップコンシューマでフェイルオーバーした場合、メッセージが順不同で配信されることがある
4956748	Oracle データベースでマスターブローカを使用できない
4964703	C-API: MQGetMessageHeaders() 関数で返される MESSAGE_ID ヘッダには、「ID:」のプレフィックスが付いていない
4964712	C-API: MQGetMessageHeaders() 関数で設定される MESSAGE_ID ヘッダは、メッセージの送信の際に無視されない
4969583	C-API: 同じメッセージの処理で 1 度以上 MQAcknowledgeMessages() を呼び出してしまうことがある
4983150	ブローカが再起動しメッセージを再配信する際、JMSRedelivered フラグが設定されない
4983699	メッセージの持続に失敗すると、ブローカはストアがスローした例外を失う

Message Queue 3.5 で修正されたバグ

表 3 は Message Queue 3.5 で修正されたバグの一覧と説明です。

表 3 Message Queue 3.5 で修正されたバグ

バグの番号	説明
4449354	きわめてまれな場合として、Session.recover および Session.rollback の各メソッドの呼び出しと同時に、Connection.stop、Connection.start、および Connection.close の各メソッドを別々のスレッドで呼び出すと、メッセージ再配信の順序が予期しない結果になることがある
4630183	送信先を廃棄してもブローカに永続サブスクリプションが残る
4753010	サーバー VM で Java プロセスのネイティブヒープセグメントが、限りなく増大する
4761626	大量のコンシューマを自動作成キューで作成 / 破棄すると、メッセージが失われる
4855307	デフォルト設定では以前のプロパティ名 (bindDN) が使用されているので、ブローカは LDAP リポジトリを認証できない
4883126	自動再接続機能が正しく動作しない
4888270	最初にトランザクションで送信されたメッセージを再送すると、ブローカエラーになる

表 3 Message Queue 3.5 で修正されたバグ (続き)

バグの番号	説明
4431924	<p>imqadmin: モーダルダイアログが、デッドロック状態になることがある</p> <p>管理コンソール (imqadmin) では、アプリケーションモーダルのダイアログを使用しています。これらのダイアログのほとんどは、たとえば「ブローカを追加」メニュー項目を選択するなど、グラフィカルユーザーインターフェースとの対話によって明示的に表示されるものです。ただし、ブローカ接続が切断された場合も、ダイアログは表示されます。同時に複数のダイアログが開いている場合、管理コンソールはロックされます。「閉じる」ボタンを使ってどちらかのモーダルダイアログを終了することはできません。</p>
4703406	<p>connection.start() を最初に呼び出さなくても QueueBrowser が機能することがある</p> <p>QueueBrowser でキューが検索できるように接続で Connection.start() を呼び出す必要があります。Connection.start() の呼び出しに失敗した場合、QueueBrowser 列挙は nextElement() でブロックし、結果として java.util.NoSuchElementException がスローされます。</p>
4866814	<p>Solaris では、ブローカが 64 ビット jvm で起動 (ブローカを「-vmargs -d64」で起動) している場合、syslog を使用してエラーメッセージや警告メッセージをログに記録できない。これは、Message Queue のベータリリースには、64 ビットバージョンのライブラリ libimqutil.so.1 が含まれていないためである</p>
4872121	<p>ブローカは 127.0.0.1 以外の IP アドレスを持たない非ネットワークシステムで起動しない</p>
4879902	<p>ブローカでメモリーの増加の速度が遅い</p>
4881968	<p>imq.autocreate.topic が「false」に設定される場合、新しい監視クライアントが作成できない</p>
4884827	<p>CTS1.3 MDB/EJB CMT テストが Message Queue 3.5 & AppServer 7.0 で失敗する</p>
4885654	<p>システムによる送信先の取得中に、自動作成された送信先に新しいメッセージがパブリッシュされると、プロデューサーに障害が発生することがある</p>
4887506	<p>あるプライマリコンシューマからバックアップコンシューマへのフェイルオーバー中に、メッセージが順不同で配信されることがある</p>
4888939	<p>送信先の最大サイズ (maxNumMsgs) が非常に小さい (5 つのメッセージ未満の) 場合、送信先で FLOW_CONTROL の動作を行う C クライアントおよび Java クライアントはメッセージの受信を中止することがある</p>
4889002	<p>プロパティ imq.transaction.autorollback が 3.5 ベータでサポートされない</p>
4891874	<p>コンシューマベースのフロー制御でメッセージのコンシューマへの配信が中止することがある。この問題は 4896133: ConnectionConsumers と Sun Java System Application Server 7 のメッセージ駆動 Beans とともに発生する可能性が高い。</p>
4895262	<p>HTTPS クライアントが HTTPSTunnelServlet 経由のブローカとの接続に失敗する</p>
4897500	<p>クラスタで、クライアントが unsubscribe() を呼び出し永続サブスクリプションを削除する際、そのクライアントに接続しているブローカの永続サブスクリプションだけが削除される。これは、他のブローカにプロデュースされたメッセージが引き続きそのサブスクライバ用に格納されていることを意味している</p>
4898020	<p>クラスタで Message Queue 3.0.* ブローカと Message Queue 3.5 ブローカをともに使用できない。両方が混在したクラスタを起動すると 3.0.1 ブローカで次のエラーが生成される。</p> <p>Configuration mismatch: Aborting connection with broker [...] because following configuration properties do not match - null imq.queue.deliverypolicy</p>
4888983	<p>imqcmd list dur が、同じ永続名を持つ永続サブスクリプションを表示しない</p>

重要な情報

この節には主な製品のドキュメントに含まれていない最近の情報が含まれています。この章で説明する項目は次のとおりです。

- [インストールの注意点](#)
- [互換性の問題](#)
- [ドキュメントの更新](#)

インストールの注意点

Message Queue を Solaris、Linux、および Windows の各プラットフォームにインストールする際の、システム要件、サポートされるソフトウェアプラットフォームや製品、インストール前の手順、アップグレードの手順、および他の関連情報については、『Message Queue インストールガイド』を参照してください。

互換性の問題

この節では、Message Queue 3.5 SP1 と Message Queue 3.5 の互換性の問題を説明しています。

Message Queue の次回のメジャーリリースに関する問題

Message Queue の次回のメジャーリリースでは、次の点で互換性がなくなる予定です。この情報は、このような変更にご留意いただく目的で今回提供しています。

- J2SE 1.3 のすべてのリリースに対する Message Queue クライアントのサポートは中止される。J2SE 1.4 は引き続きサポートされる
- Message Queue のすべてのコマンドラインインタフェースは、パスワードコマンドラインの引数として提供するオプションを削除するよう変更される。たとえば、次のような場合

```
imqbrokerd -ldappassword <passwd> imqcmd -p <passwd>
```


パスワードを指定する代替のメカニズムが用意される
- ブローカログファイルのフォーマットが変更される。現在のフォーマットに依存するアプリケーションは動作しなくなる可能性がある
- Message Queue の一部としてインストールされる個々のファイルの場所が変更される可能性がある。これによって、特定の Message Queue ファイルの現在の場所に依存する既存のアプリケーションの動作が中断する可能性がある
- imqkeytool プログラムが製品から削除される可能性がある。代わりに、J2SE keytool がサポートされる

- 次回のメジャーバージョン以前の Message Queue のバージョンを使用する Message Queue クライアントは、製品のそのバージョンで提供される新機能にはアクセスできない可能性がある
- MQAcknowledgeMessages() 関数が、通知済みのメッセージの C クライアントによって (MQ_CLIENT_ACKNOWLEDGE を使用して) 呼び出される際にはエラーが生成されない。この動作は変更予定

Message Queue 3.5 の問題

Message Queue 3.5 は、通常 Message Queue 3.0 (さらに後継のバージョン 3.0.1、3.0.1 Service Pack 1、および 3.0.1 Service Pack 2) と互換性があります。しかし、ブローカプロパティ、管理対象オブジェクト、持続スキーマ、ファイルの場所、および管理ツールに変更が加えられており、それは、Message Queue 3.0 バージョンから Message Queue 3.5 バージョンへのアップグレードの際にも影響する可能性があります。

Message Queue 3.5 のインストールでは、Message Queue 3.0 IMQ_VARHOME ディレクトリを削除したり、上書きしたりしません。このディレクトリには、設定およびセキュリティ関連のファイルが収められています。このデータのほとんどは Message Queue 3.5 と互換性があり、『Message Queue インストールガイド』の手順に従って保存できます。

Message Queue 3.0 から Message Queue 3.5 へアップグレードする際に、対処が必要な可能性のある問題は次のとおりです。

- [ブローカの互換性](#)
- [プロパティおよび属性の変更](#)
- [公開 .jar ファイルの場所](#)

管理対象オブジェクトの互換性、クライアントの互換性、および管理ツールの互換性については、『Message Queue インストールガイド』を参照してください。

ブローカの互換性

Message Queue 3.5 ブローカは Message Queue 3.0 ブローカと相互に運用可能ですが、ブローカプロパティと持続格納スキーマには変更が加えられています。Message Queue 3.0 のデータによっては、引き続き Message Queue 3.5 と互換性があるものがあります。詳細は、『Message Queue インストールガイド』を参照してください。

プロパティおよび属性の変更

この節には、Message Queue 3.5 で加えられたブローカのプロパティ、送信先属性、および接続ファクトリ属性の変更の要約が含まれています。

ブローカのプロパティ 次の表では、Message Queue 3.5 の新しいプロパティ、推奨されていないプロパティ、およびプロパティ名の変更の詳細を示しています。詳細については、『Message Queue 管理ガイド』の第 2 章を参照してください。

表 4 新しい Message Queue 3.5 のブローカのプロパティ

プロパティ名	機能の参照先
imq.persist.file.message.max_record.size	10 ページの「持続ストアパフォーマンスの強化」
imq.persist.file.destination.message.filepool.limit	10 ページの「持続ストアパフォーマンスの強化」
imq.metrics.topic.enabled	7 ページの「リモート監視 API (Enterprise Edition)」
imq.metrics.topic.interval	7 ページの「リモート監視 API (Enterprise Edition)」
imq.metrics.topic.persist	7 ページの「リモート監視 API (Enterprise Edition)」
imq.metrics.topic.timetolive	7 ページの「リモート監視 API (Enterprise Edition)」
imq.autocreate.destination.maxNumMsgs	5 ページの「ブローカメッセージのフロー制御の強化」
imq.autocreate.destination.maxTotalMsgBytes	5 ページの「ブローカメッセージのフロー制御の強化」
imq.autocreate.destination.maxBytesPerMsg	5 ページの「ブローカメッセージのフロー制御の強化」
imq.autocreate.destination.maxNumProducers	5 ページの「ブローカメッセージのフロー制御の強化」
imq.autocreate.queue.maxNumActiveConsumers	8 ページの「キュー配信ポリシーの強化 (Enterprise Edition)」
imq.autocreate.queue.maxNumBackupConsumers	8 ページの「キュー配信ポリシーの強化 (Enterprise Edition)」
imq.autocreate.queue.consumerFlowLimit	6 ページの「Java クライアントのフロー制御の強化」 および 8 ページの「キュー配信ポリシーの強化 (Enterprise Edition)」
imq.autocreate.topic.consumerFlowLimit	6 ページの「Java クライアントのフロー制御の強化」
imq.autocreate.queue.localDeliveryPreferred	9 ページの「クラスタパフォーマンスの強化 (Enterprise Edition)」
imq.autocreate.destination.isLocalOnly	10 ページの「ローカル送信先 (Enterprise Edition)」

表 5 推奨されていない Message Queue 3.5 のブローカのプロパティ

プロパティ名
imq.persist.file.message.fdpool.limit
imq.persist.file.message.filepool.limit
imq.redelivered.optimization
imq.queue.deliverypolicy

次の表では、Message Queue 3.5 の新しい送信先属性と推奨されていない送信先属性の詳細を示しています。詳細については、『Message Queue 管理ガイド』の第 6 章を参照してください。

表 6 新しい Message Queue 3.5 の送信先属性

送信先タイプ	属性名	機能の参照先
Queue と Topic	maxNumMsgs	5 ページの「ブローカメッセージのフロー制御の強化」
Queue と Topic	maxTotalMsgBytes	5 ページの「ブローカメッセージのフロー制御の強化」
Queue と Topic	limitBehavior	5 ページの「ブローカメッセージのフロー制御の強化」
Queue と Topic	maxBytesPerMsg	5 ページの「ブローカメッセージのフロー制御の強化」
Queue と Topic	maxNumProducers	5 ページの「ブローカメッセージのフロー制御の強化」
Queue のみ	maxNumActiveConsumers	8 ページの「キュー配信ポリシーの強化 (Enterprise Edition)」
Queue のみ	maxNumBackupConsumers	8 ページの「キュー配信ポリシーの強化 (Enterprise Edition)」
Queue と Topic	consumerFlowLimit	6 ページの「Java クライアントのフロー制御の強化」 および 8 ページの「キュー配信ポリシーの強化 (Enterprise Edition)」
Queue のみ	localDeliveryPreferred	9 ページの「クラスタパフォーマンスの強化 (Enterprise Edition)」
Queue と Topic	isLocalOnly	10 ページの「ローカル送信先 (Enterprise Edition)」

表 7 推奨されていない Message Queue 3.5 の送信先属性

送信先タイプ	属性名
Queue	QueueDeliveryPolicy

接続ファクトリ属性 次の表では、Message Queue 3.5 の新しい接続ファクトリ属性の詳細を示しています。詳細については、『Message Queue Java Client Developer's Guide』の第 4 章を参照してください。Message Queue 3.0 バージョンの接続ファクトリ属性は、Message Queue 3.5 でもサポートされ、Message Queue 製品の次のメジャーリリースまでは引き続きサポートされます。

表 8 新しい Message Queue 3.5 の接続ファクトリ属性

属性名	機能の参照先
imqAddressList	4 ページの「Java クライアント接続のフェイルオーバー (Enterprise Edition)」
imqAddressListBehavior	4 ページの「Java クライアント接続のフェイルオーバー (Enterprise Edition)」
imqAddressListIterations	4 ページの「Java クライアント接続のフェイルオーバー (Enterprise Edition)」

表 8 新しい Message Queue 3.5 の接続ファクトリ属性 (続き)

属性名	機能の参照先
imqReconnectEnabled	4 ページの「Java クライアント接続のフェイルオーバー (Enterprise Edition)」
imqReconnectAttempts	4 ページの「Java クライアント接続のフェイルオーバー (Enterprise Edition)」
imqReconnectInterval	4 ページの「Java クライアント接続のフェイルオーバー (Enterprise Edition)」
imqConsumerFlowLimit	6 ページの「Java クライアントのフロー制御の強化」
imqConsumerFlowThreshold	6 ページの「Java クライアントのフロー制御の強化」

表 9 名前が変更された接続ファクトリ属性

以前の名前	Message Queue 3.5 での名前
imqFlowControlCount	imqConnectionFlowCount
imqFlowControlIsLimited	imqConnectionFlowLimitEnabled
imqFlowControlLimit	imqConnectionFlowLimit

公開 .jar ファイルの場所

Solaris プラットフォームの場合、公開 .jar の場所が Message Queue 3.0.1 では、Message Queue 3.0 の場所 (/usr/share/lib/imq/) から /usr/share/lib/ に移動しました。この移動した jar ファイルに対して Message Queue 3.0.1 で /usr/share/lib/imq/ に作成されていたシンボリックリンクは、削除されました。

該当する .jar ファイルは次のとおりです。

- jms.jar
- imq.jar
- imqxm.jar
- activation.jar
- saaj-api.jar
- saaj-impl.jar
- mail.jar
- commons-logging.jar
- jaxm-api.jar
- fscontext.jar

ドキュメントの更新

この節では、バージョン 3.5 SP1 およびバージョン 3.5 の Message Queue マニュアルに加えられた変更について説明しています。

バージョン 3.5 SP1 での変更

次の Message Queue 3.5 SP1 マニュアルは、この製品のバージョン 3.5 から更新されました。

インストールガイド

Message Queue インストールガイドは更新され、ブランド名の変更やプラットフォームサポート情報が反映されています。

管理ガイド

『Message Queue 管理ガイド』は、『Message Queue 管理者ガイド』から名前が変更され、ブランド名の変更が反映されるように更新されました。このドキュメントのパフォーマンス監視を説明する章も内容が詳しくなり更新されています。

Java Client Developer's Guide

『Message Queue Java Client Developer's Guide』は更新され、ブランド名の変更が反映されています。

C Client Developer's Guide

『Message Queue C Client Developer's Guide』は更新され、ブランド名の変更が反映されています。

バージョン 3.5 での変更

次の Message Queue 3.5 マニュアルは、この製品のバージョン 3.0.1 から更新されました。これらの更新済みマニュアルは、次の Message Queue 3.5 マニュアル Web サイトで入手できます。

http://docs.sun.com/coll/S1_MessageQueue_35

インストールガイド

Message Queue 3.5 製品には、更新済みの Message Queue インストールガイドが含まれています。これには、新しいソフトウェア要件、Solaris インストール手順の変更、Linux の新しい Red Hat Package Manager (RPM) を使用するインストール手順とインストールディレクトリ構造、および Windows インストールに関するわずかな変更が含まれています。

訂正：プラグイン持続性サポートの場合、表 1-2 では、PointBase バージョン 4.5 が Message Queue でサポートされていると明記されています。しかし実際は、サポートされているバージョンは PointBase バージョン 4.8 です。

管理者ガイド

『Message Queue 管理ガイド』は更新されており、Message Queue 3.5 の変更が含まれています。[2 ページの「この節では、Message Queue 3.5 SP1 での変更点、および以前のリリース、Message Queue 3.5 での変更点を説明しています。」](#)を参照してください。

Java Client Developer's Guide

『Message Queue Java Client Developer's Guide』は、以前の『Message Queue Developer's Guide』の内容のほとんどが含まれ、さらに更新されており、Message Queue 3.5 の変更も含まれています。2 ページの「この節では、Message Queue 3.5 SP1 での変更点、および以前のリリース、Message Queue 3.5 での変更点を説明しています。」を参照してください。

C Client Developer's Guide

『Message Queue C Client Developer's Guide』は、Message Queue マニュアルセットに新たに追加され、Message Queue C クライアントアプリケーションの作成方法を説明しています。

既知の問題および制約

この節では Message Queue 3.5 SP1 と Message Queue 3.5 の既知の問題、制約、およびバグを説明しています。バージョン 3.5 SP1 は、単に Message Queue 3.5 のブランド名を変更したバージョンに過ぎないので、この節で提供される情報は両方のバージョンに当てはまります。

現時点のバグ、その状態、および回避方法の一覧については、Java Developer Connection™ メンバーは、Java Developer Connection Web サイトの「Bug Parade」ページを参照してください。新しいバグを報告する前に、このページをチェックしてください。すべての Message Queue バグがリストされているわけではありませんが、このページはある問題が報告済みかどうかを知りたい場合に活用できます。

関連ページは次のとおりです。

<http://developer.java.sun.com/developer/bugParade>

注	Java Developer Connection のメンバーになるのは無料ですが、登録が必要です。Java Developer Connection のメンバーになる方法についての詳細は、Sun の「For Developers」Web ページを参照してください。
----------	---

新しいバグの報告や機能に関する要求を行うには、imq-feedback@sun.com 宛てにメールを送信してください。

既知の問題

この節では Message Queue 3.5 SP1 の既知の問題を説明しています。中には Message Queue 3.5 バージョンで発生した問題もあります。この章では、Message Queue 3.5 製品の Enterprise Edition および Platform Edition の両方に該当するものと、Enterprise Edition だけに該当するものとの問題を分類しています。

Enterprise Edition および Platform Edition

- 製品のブランド名変更に伴い、以前、文字列を返す API の
「Sun ONE Message Queue, Sun Microsystems, Inc.」
は、今回、次のような文字列を返します。
「Sun Java(tm) System Message Queue」
- 製品のブランド名の変更に伴い、MQGetMetaData() から MQ_NAME_PROPERTY に関連付けられた値の正確な比較を行う、MQ 3.5 FCS でコンパイルされた C クライアントプログラムは、ランタイムに 3.5 SP1 mqcrct 共有ライブラリを使用すると失敗します。
- Windows プラットフォームには、バックログサイズの最大値に応じて、1 つのブローカに対して TCP/IP 経由で同時に起動できる接続数に制限が設けられています。バックログとは、TCP スタック内の接続用バッファです。TCP 同時起動接続の数が、バックログサイズを超えることはできません。たとえば、Windows 2000 Professional ではバックログを 5 に、Windows 2000 Server では 200 に制限しています。
- Windows XP を使用する場合、インバウンド接続の数に制限が設けられています。Windows XP Professional の場合、ネットワーク経由で同時に接続可能な他のコンピュータの最大数は 10 台です。この制限には、プロトコルを共有するすべてのトランスポートおよびリソースが含まれます。Windows XP Home Edition の場合、ネットワークを介して同時に接続可能な他のコンピュータの最大数は 5 台です。この制限は Windows XP を実行するブローカに接続できるクライアントの数に影響します。

まったく動作していないファイル、印刷、名前付きパイプ、またはメールスロットのセッションは、AutoDisconnect 時間の期限が切れると自動的に切断されます。デフォルトの AutoDisconnect 時間は 15 分です。セッションが切断されると、10 個の接続の 1 つが使用可能になるので、別のユーザーが Windows XP システムに接続できるようになります。したがって、サーバー用途などにあまり多用しないシステムの場合、AutoDisconnect 時間を短く設定するなら、10 箇所の接続制限または 5 箇所の接続制限の問題を軽減できることもあります。詳細は次を参照してください。

<http://support.microsoft.com/default.aspx?scid=kb;EN-US;314882>

- ブローカインスタンスを少なくとも 1 回起動しなければ、ブローカのインスタンス設定ファイルを編集することはできません。そのブローカインスタンスがはじめて起動されるまでは、config.properties ファイルが存在しないからです。プラグイン可能な持続性を使用するか、ほかの設定プロパティを設定してブローカを設定するには、ブローカを 1 度実行して config.properties ファイルを作成します。

プラットフォーム	場所
Solaris	<code>/var/imq/instances/<i>instanceName</i>/props/config.properties</code>
Linux	<code>/var/opt/imq/instances/<i>instanceName</i>/props/config.properties</code>
Windows	<code>IMQ_VARHOME\instances\instanceName\props\config.properties</code>

`config.properties` ファイルが作成されたら、このファイルを編集して設定プロパティ値を追加してから、ブローカを再起動します。

Enterprise Edition のみ

- このリリースでは、フル接続のブローカクラスタのみサポートされています。つまり、クラスタ内のすべてのブローカは、そのクラスタ内のほかのブローカと相互に直接やり取りする必要があります。 `imqbrokerd -cluster` コマンド行引数を使用してブローカに接続する場合は、そのクラスタ内のすべてのブローカが含まれていることを確認してください。
- ブローカクラスタでマスターブローカを使用しない場合、ブローカにより保存され、そのクラスタに追加される持続性の情報は、クラスタ内のほかのブローカには伝播されません。
- SSL を使用した接続サービスでは現在、自己署名サーバー証明書、つまりホスト信頼モードしかサポートしていません。
- HTTP トランスポートを使用している JMS クライアントが、`Ctrl-C` の使用などにより突然終了した場合、ブローカがクライアント接続や関連するすべてのリソースを解放するまでに、およそ 1 分かかります。

この 1 分の間にクライアントのほかのインスタンスが開始し、同じ **ClientID** を使おうとした場合、永続サブスクリプション、つまりキューは、「リソースの衝突」の例外を受け取ります。このことは実際の問題ではなく、上記の終了処理の結果にすぎません。およそ 1 分経過後にクライアントが起動すると、すべて問題なく動作します。

既知のバグ

表 10 は Message Queue 3.5 SP1 の重要なバグの一覧です。

表 10 Message Queue 3.5 の既知のバグ

バグの番号	詳細
4683029	<p>すべての Solaris および Windows スクリプトで、<code>-javahome</code> オプションの値に空白文字が含まれると動作しない</p> <p><code>-javahome</code> オプションは Message Queue コマンドおよびユーティリティで使用し、使用する代替の Java 2 互換のランタイムを指定します。ただし、代替の Java 2 互換のランタイムへのパスには、空白文字を含めることはできません。</p> <p>空白文字を含むパスの例は次のとおりです。</p> <p>Windows の場合：</p> <p style="padding-left: 40px;"><code>C:\¥jdk 1.4</code> (Windows の場合、パス全体が引用符で囲まれているなら空白文字を含むことも可能。 例：<code>"C:\¥jdk 1.4"</code>)</p> <p>Solaris の場合：</p> <p style="padding-left: 40px;"><code>/work/java 1.4</code></p> <p>回避方法：Java ランタイムを、空白文字が含まれない場所またはパスにインストールします。</p>
4939923	<p>共有スレッドプールの使用時、ブローカの JVM がメモリーで実行している場合、ブローカが <code>NullPointerException</code> を生成することがある</p> <p>回避方法：なし。このバグは J2SE 1.4.2_03 で修正されている</p>
4941058	<p>フロー制御中の送信先が最大制限数にならないことがある。送信先が、設定した最大制限数になる前に、プロデューサが送信先へのメッセージの送信を中止する状態が発生することがある</p> <p>回避方法：なし</p>
4941066	<p>送信先は割り当てられたバイト制限値を少し超過することがある</p> <p>回避方法：なし</p>
4941127	<p>メッセージが個々のメッセージサイズの制限値を超過する場合、送信先は必ずしも完全には読み込まない。送信先で許容されるメッセージのサイズの制限値が大きなメッセージの格納後に変更される場合、送信先はきちんと読み込まない</p> <p>回避方法：大きなメッセージが消費されるまでメッセージサイズの制限値を増大させ、次いでメッセージの制限値を減少させる。他の大きなメッセージの受け取りを防止するため、この間、送信先へのプロデュースを停止することもできる</p>
4946531	<p>メッセージのプロデュース中に、特に害はない <code>NullPointerException</code> がまれに発生することがある</p> <p>回避方法：なし。Null ポインタの例外は問題なく無視できる</p>
4949398	<p>対応する送信先が読み込み中の場合、<code>imqcmd query dst</code> は Number of Messages および Total Message Bytes の不正確な値を報告する。送信先が読み込み前または後の場合、報告される値は正確である</p> <p>回避方法：問題は送信先が読み込み中の場合にだけ発生する。いったん送信先の読み込みが終了したなら、返される値は正確になる</p>

表 10 Message Queue 3.5 の既知のバグ (続き)

バグの番号	詳細
4950166	<p>jdk1.4.2_02 と x86 のシステムで実行するとブローカにランダムなエラーが発生する。詳細は、「J2SE bug 4947404」を参照</p> <p>回避方法: -XX:UseSSE=0 でブローカを起動する。たとえば次のとおり。</p> <pre>imqbrokerd -tty -vmargs -XX:UseSSE=0</pre>
4950601	<p>JDBC 持続ストアの使用時に、imqcmd metrics dst がブローカの内部エラーをトリガーしてそれが出力される</p> <p>ディスクの使用率メトリックス情報はファイルストアにだけ当てはまる。しかしメトリックス情報の取得時に、ブローカはディスクの使用率情報の取得をストアタイプに関係なく試行する。ファイルストアの代わりにデータベースを使用している場合、ブローカは次のエラーメッセージを出力する。</p> <pre>06/Nov/2003:22:57:36 PST] ERROR [B3100]: Unexpected Broker Internal Error : [unable to disk usage for destinationT:topic1] : com.sun.messaging.jmq.jmsserver.util.BrokerException: The operation does not apply to plugged-in persistent store.</pre> <p>回避方法: なし</p>
4951010	<p>ブローカクラスタで、開始しない可能性のあるリモート接続へのメッセージをブローカがキューに入れる</p> <p>回避方法: いったんその接続が開始すると、メッセージはコンシューマによって受信される。コンシューマの接続が閉じている場合、メッセージは別のコンシューマへ再配信される</p>
4953348	<p>Windows 2000 で HTTPS createQueueConnection が例外をスローする場合がある</p> <p>回避方法: 接続を再試行する</p>
4953354	<p>持続ストアがあまりにも多くの送信先を開く場合、ブローカがアクセス不可能になる</p> <p>回避方法: この状態はブローカがシステムのオープンファイル記述子の制限に達したことが原因である。Solaris や Linux では、ulimit コマンドを使って、ファイル記述子の制限を増やす</p>
4954974	<p>Windows XP の場合、CD 媒体を使用すると、インストールが自動的に開始しない</p> <p>回避方法: Windows のエクスプローラで、CD の windows のフォルダをダブルクリックし、次に imq3_5-ent-win.exe ファイルをダブルクリックするとインストーラが起動する</p>
4983525	<p>Linux Red Hat Advanced Server 3.0 では、自動作成された送信先に対するメッセージプロデューサの作成が失敗することがある</p> <p>回避方法: プロデューサの再作成を試みる。2 回目には正常に作成される可能性が高い。あるいは、管理者権限で作成された送信先を使用する</p>
4986318	<p>クライアントが、突然、次のような ACKNOWLEDGE_REPLY メッセージを生成することがある。</p> <pre>***** Packet: ACKNOWLEDGE_REPLY(25):26-192.18.86.227-42976-1075458056557 Magic/Version: 469754818/301Size: 97 Type: ACKNOWLEDGE_REPLY(25) Expiration: 0 Timestamp: 1075458056557 Source IP: 192.18.86.227 Source Port: 42976Sequence: 26</pre> <p>回避方法: なし。ブローカでは、クライアントがこのエラーメッセージを生成するタイミングとなる状態がまれにある。このエラーは無視できる。失われるメッセージはない</p>

表 10 Message Queue 3.5 の既知のバグ (続き)

バグの番号	詳細
4991257	<p>持続ストアが JDBC ベースであるブローカクラスタの永続サブスクリバへ大きな持続メッセージを送信すると、ブローカのハングアップまたはブローカによるエラーメッセージの生成、あるいはその両方が発生する</p> <p>回避方法 : 次のブローカプロパティを使用して、ブローカのロックプロトコルのタイムアウトを長くする</p> <pre>imq.cluster.timeout=<timeout-in-seconds></pre> <p>デフォルト設定は 60 である。持続する大きなメッセージに時間がかかる場合、持続ストアまたはスイッチを異なる持続ストアに調整することが必要な場合もある</p>
5006686	<p>imqbrokerd.conf の ARGS の例が正しくない</p> <p>回避方法 : 例のように引用符で値を囲む必要はない</p> <pre>ARGS="-name newbroker -port 8888"</pre> <p>値は次のようにする必要がある。</p> <pre>ARGS=-name newbroker -port 8888</pre>

再配布可能ファイル

Sun Java System Message Queue 3.5 SP1 には、バイナリ形式で使用および自由に配布が可能なファイルの次のセットが含まれています。

jms.jar
imq.jar
imqxm.jar
fscontext.jar
providerutil.jar
jndi.jar
ldap.jar
ldapbpjar
jaas.jar
jsse.jar
jnet.jar
jcert.jar

さらに、LICENSE ファイルおよび COPYRIGHT ファイルも再配布できます。

問題の報告方法とフィードバックの提供方法

問題を報告するには、imq-feedback@sun.com 宛てにメールを送信してください。

Message Queue に関する問題があり、サポート契約をしている場合は、次の方法のいずれかを使ってカスタマサポートにご連絡ください。

- Sun Software Support オンラインサービスの Web サイトをご利用ください。
<http://www.sun.com/service/sunone/software>
このサイトには、メンテナンスプログラムおよびサポート連絡先番号だけでなく、Knowledge Base、オンラインサポートセンター、および ProductTracker へのリンクがあります。
- 保守契約を結んでいるお客様の場合は、専用ダイヤルをご利用ください。

テクニカルサポートスタッフが問題解決のお手伝いをいたします。

- カスタマサポートをご利用の際は、次の情報をご用意ください。問題が発生した箇所や動作への影響など、問題の具体的な説明
- マシン機種、OS バージョン、および、問題に影響すると思われるパッチやそのほかのソフトウェアなどの製品バージョン
- 問題を再現するために行なった操作に関する詳しい説明
- エラーログやコアダンプ

コメントの送信先

Sun ではマニュアルの改善に関心を払っており、お客様のコメントおよびご提案を必要としています。Sun にフィードバックする場合、次の Web ベースのフォームをご使用ください。

<http://www.sun.com/hwdocs/feedback/>

マニュアルの完全なタイトルと Part No. を該当するフィールドにお書きください。Part No. は書籍のタイトルのページまたはマニュアルの最上部に記されており、通常、7 桁または 9 桁の数字です。たとえば、この『Message Queue 3.5 SP1 リリースノート』の Part No. は 817-7201 です。

Sun の追加情報

Message Queue のマニュアル以外に、次に示す追加情報を利用できます。

ディスカッションフォーラム

Sun Java System Software Forum

次のサイトでは、Sun Java System Message Queue フォーラムが利用できます。

<http://softwareforum.sun.com/NASApp/jive/forum.jsp?forum=24>

ご参加を歓迎いたします。

Java Technology Forum

Java Technology Forums には、関連する JMS のフォーラムがあります。

<http://forum.java.sun.com>

SunSolve Knowledge Base

Sun Java System Message Queue に関する情報は、次の SunSolve Knowledge Base からオンラインで利用できます。

<http://sunsolve.Sun.COM/pub-cgi/search.pl?mode=advanced>

「All Free Collections」を選択して、「Message Queue」で検索してください。

Sun Java System の情報

次のサイトにも、Sun Java System に関する有益な情報が掲載されています。

- Message Queue 製品情報
http://www.sun.com/software/products/message_queue/index.html
- Message Queue に関するマニュアル
<http://docs.sun.com/db/prod/entsys?l=ja>
- Sun マニュアル
<http://docs.sun.com/>
- Sun Java System ソフトウェア製品とサービス
<http://www.sun.com/software>
- Sun ソフトウェアサポートサービス
<http://www.sun.com/service/sunone/software>
- Sun サポートと Knowledge Base
<http://sunsolve.sun.com>
- Sun サポートとトレーニングサービス
<http://www.sun.com/supporttraining>
- Sun 開発者向け情報
<http://developers.sun.com/>
- Sun 開発者サポートサービス
<http://www.sun.com/developers/support>
- Sun ソフトウェアデータシート
<http://www.sun.com/software>

Copyright © 2004 Sun Microsystems, Inc. All rights reserved.

U.S. Government Rights - Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements. Use is subject to license terms. この配布には、第三者が開発したソフトウェアが含まれている可能性があります。

Sun、Sun Microsystems、Sun のロゴ、Java、Solaris、および Sun™ ONE は米国およびその他の国における米国 Sun Microsystems, Inc. の商標もしくは登録商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。

UNIX は、X/Open Company Limited が独占的にライセンスしている米国および他の国における登録商標です。