

**IBM WebSphere Business Integration
Adapters**



Adapter for Exchange ユーザーズ・ガイド

Adapter バージョン 2.0.x

お願い

本書および本書で紹介する製品をご使用になる前に、85 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM WebSphere Business Integration Adapter for Exchange (5724-H21) バージョン 2.0.0 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)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： IBM WebSphere Business Integration Adapters
Adapter for Exchange User Guide
Adapter Version 2.0.x

発 行： 日本アイ・ピー・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 2004.7

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 2003, 2004. All rights reserved.

© Copyright IBM Japan 2004

目次

本書について	v
対象読者	v
関連文書	v
表記上の規則	vi
改訂の要約	vii
Adapter for Exchange 2.0.0 の新機能	vii
第 1 章 アダプターの概要	1
アダプターのアーキテクチャー	1
データ・ストレージおよびアクセス	3
コネクタの動作方法	4
アダプター環境	9
第 2 章 アダプターのインストール	13
コネクタのインストール	13
Event Listener のインストール	14
第 3 章 タスクの構成	17
Event Listener に必要な構成タスク	17
コネクタに必要な構成タスク	22
イベントの登録	25
アダプターの始動	28
アダプターの停止	31
1 台のサーバー上に複数のアダプター・インスタンスを作成	32
第 4 章 ビジネス・オブジェクトの理解	35
メタデータの定義	35
ビジネス・オブジェクト構造の概要	36
Exchange Server のビジネス・オブジェクト	36
第 5 章 トラブルシューティングとエラー処理	41
コネクタのエラー処理	41
Event Listener のエラー処理	42
トレース・メッセージ	43
付録 A. コネクタの標準構成プロパティ	47
新規プロパティと削除されたプロパティ	47
標準コネクタ・プロパティの構成	47
標準プロパティの要約	49
標準構成プロパティ	53
付録 B. Connector Configurator	67
Connector Configurator の概要	67
Connector Configurator の始動	68
System Manager からの Configurator の実行	69
コネクタ固有のプロパティ・テンプレートの作成	69
新規構成ファイルの作成	72
既存ファイルの使用	73
構成ファイルの完成	75

構成ファイル・プロパティの設定	75
構成ファイルの保管	82
構成ファイルの変更	83
構成の完了	83
グローバル化環境における Connector Configurator の使用	84
特記事項	85
プログラミング・インターフェース情報	87
商標	87
索引	89

本書について

IBM^(R) WebSphere^(R) Business Integration Adapters ポートフォリオは、主要な e-business テクノロジー、エンタープライズ・アプリケーション、レガシー・システムおよびメインフレーム・システムにコネクティビティーを提供します。製品セットには、ビジネス・プロセスの統合に向けてコンポーネントをカスタマイズ、作成、および管理するためのツールとテンプレートが含まれています。

本書では、IBM WebSphere Business Integration Adapter for Exchange のインストール、構成、トラブルシューティング、およびビジネス・オブジェクト開発について説明します。

対象読者

本書は、お客様のサイトでアダプターを使用するコンサルタント、開発者、およびシステム管理者を対象としています。

関連文書

この製品に付属する資料の完全セットで、すべての WebSphere Business Integration Adapter のインストールに共通な機能とコンポーネントについて説明します。また、特定のコンポーネントに関する参考資料も含まれています。

以下のサイトから資料をインストールするか、オンラインで直接閲覧することができます。

- WebSphere MQ Integrator Broker または WebSphere Application Server を統合ブローカーとして使用している場合:
www.ibm.com/software/websphere/wbiadapters/infocenter
- InterChange Server を統合ブローカーとして使用している場合:
www.ibm.com/websphere/integration/wicsserver/infocenter

上記のサイトには資料のダウンロード、インストール、および表示に関する簡単な説明が記載されています。

資料のセットは主に Portable Document Format (PDF) ファイルで構成されていますが、HTML 形式のファイルもあります。それらを読むには、Netscape Navigator や Internet Explorer などの HTML ブラウザーや、Adobe Acrobat Reader 4.0.5 以上のバージョンが必要です。ご使用のプラットフォーム用に Adobe Acrobat Reader の最新バージョンを取得するには、Adobe Web サイトの www.adobe.com にアクセスしてください。

注: 本書の発行後に公開されたテクニカル・サポートの技術情報や速報に、本書の対象製品に関する重要な情報が記載されている場合があります。これらの情報は、WebSphere Business Integration Support Web サイト

(<http://www.ibm.com/software/integration/websphere/support/>) にあります。関心のあるコンポーネント・エリアを選択し、「Technotes」セクションと「Flashes」セクションを参照してください。

表記上の規則

本書では、以下のような規則を使用しています。

Courier フォント	コマンド名、ファイル名、入力情報、システムが画面に出力した情報など、記述されたとおりの値を示します。
イタリック 青のアウトライン	初出語、変数名、または相互参照を示します。 オンラインで表示したときのみ見られる青のアウトラインは、相互参照用のハイパーリンクです。アウトラインの内側をクリックすると、参照先オブジェクトにジャンプします。
{ }	構文の記述行の場合、中括弧 {} で囲まれた部分は、選択対象のオプションです。1 つのオプションのみを選択する必要があります。
	構文の記述行の場合、パイプで区切られた部分は、選択対象のオプションです。1 つのオプションだけを選択する必要があります。
[]	構文の記述行の場合、大括弧 [] で囲まれた部分は、オプションのパラメーターです。
...	構文の記述行の場合、省略符号 ... は直前のパラメーターが繰り返されることを示します。例えば、option[,...] は、複数のオプションをコンマで区切って指定できることを意味します。
< >	1 つの名前の個々のエレメントを互いに区別するために、不等号括弧によって個々のエレメントが囲まれます。例えば、<server_name><connector_name>tmp.log のように使用します。
/、¥	本書では、ディレクトリー・パスの規則として円記号 (¥) を使用します。UNIX システムの場合には、円記号をスラッシュ (/) に置き換えてください。すべての製品パス名は、マシン上でコネクタがインストールされているディレクトリーを基準とした相対パス名です。
<i>ProductDir</i>	IBM WebSphere Business Integration Adapters 製品のインストール先ディレクトリーを表します。デフォルトのディレクトリーは WebSphereAdapters です。

改訂の要約

本章では、「*Adapter for Exchange ユーザーズ・ガイド*」に対する本リリースでの変更点について説明します。

Adapter for Exchange 2.0.0 の新機能

本書は、以下に示す変更点を含むように更新されました。

- CDO に代わって WebDAV がアクセス・モードとして使用され、アクセスは URI により行われます。
- コネクタはすべてのサポート・プラットフォームで稼動します (Windows 2000、AIX、HP、および Solaris 8.0)。
- コネクタは Windows サービスとして実行できます。
- コネクタは、単一スレッド・モードに限定されなくなりました。
- ビジネス・オブジェクト定義ファイルは .in から .xsd に変更されました。

第 1 章 アダプターの概要

本章では、IBM WebSphere Business Integration Adapter for Exchange および関連するシステム体系について説明します。アダプターは、構成および開発ツール、メッセージ・ファイル、およびコネクター、つまり統合ブローカーとの間でデータの転送を実行するランタイム・コンポーネントのセットから構成されます。

コネクターは、アプリケーション固有のコンポーネントおよびコネクター・フレームワークで構成されます。アプリケーション固有のコンポーネントには、特定のアプリケーション用 (この場合は Exchange) のコードが含まれています。コネクター・フレームワークは統合ブローカーとアプリケーション固有のコンポーネントの間の仲介役として機能し、そのコードはどのコネクターにも共通です。コネクター・フレームワークは、統合ブローカーとアプリケーション固有のコンポーネントとの間で以下のようなサービスを提供します。

- ビジネス・オブジェクトの受信と送信
- 始動メッセージや管理メッセージの交換の管理

本書には、アプリケーション固有のコンポーネントおよびコネクター・フレームワークの両方に関する情報が含まれています。

アダプターに対する統合ブローカーの関係についての詳細は、「*IBM WebSphere Business Integration システム管理ガイド*」(WebSphere InterChange Server の場合)、「*WebSphere Message Brokers 使用アダプター・インプリメンテーション・ガイド*」、または「*アダプター実装ガイド (WebSphere Application Server)*」を参照してください。

Adapter for Exchange には、Event Listener と呼ばれるコンポーネントも含まれています。このコンポーネントは、Microsoft Exchange Server と同じマシンで稼働する COM+ イベント・シンク・アプリケーションです。

本章の内容は、次のとおりです。

- 『アダプターのアーキテクチャー』
- 3 ページの『データ・ストレージおよびアクセス』
- 4 ページの『コネクターの動作方法』
- 9 ページの『アダプター環境』

アダプターのアーキテクチャー

Adapter for Microsoft Exchange Server を使用すると、統合ブローカーと Exchange Server 間でビジネス・オブジェクトを交換できます。

Exchange Server は、各種の E メール・プロトコルをサポートする、強力な電子メッセージング・バック・エンドです。Exchange Server を使用すると、個々のユーザーだけでなく組織全体が、パブリック・フォルダーやプライベート・フォルダーを介して予定表、連絡先リスト、およびその他の各種ファイルをホスティングできま

す。Adapter for Exchange を使用することにより、Exchange Server と他のアプリケーション間でデータを交換できるようになります。

図 1 に、コネクターの一般的なアーキテクチャーを示します。

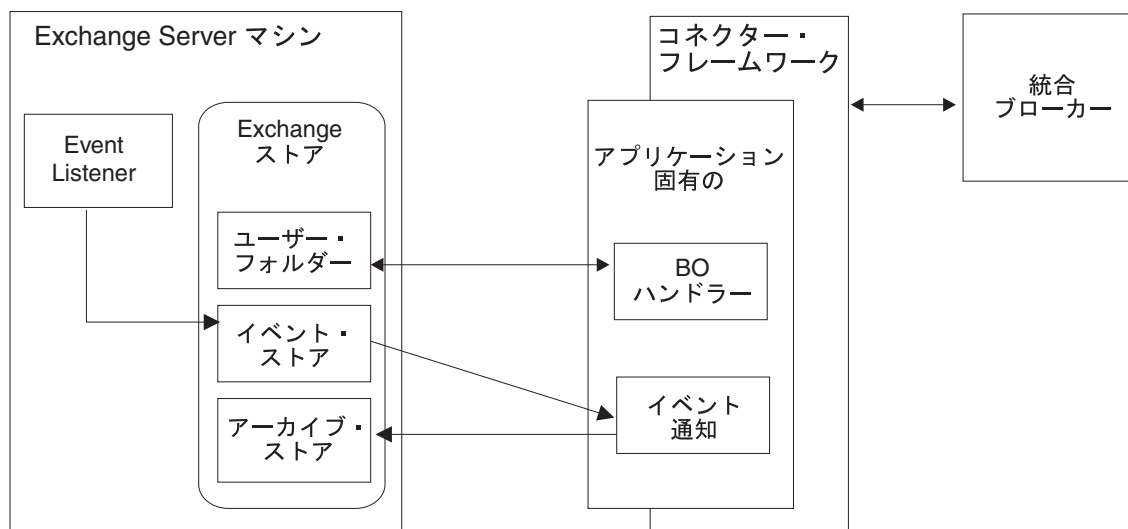


図 1. Exchange Server コネクタ・アーキテクチャー

Exchange Server コネクターのアプリケーション固有のコンポーネントには、イベント通知機能と 6 種類のサポート Exchange ビジネス・エンティティ (AppointmentItems、ContactItems、MailItems、TaskItems、Recipients、および RecurrencePatterns) をすべて認識できる単一ビジネス・オブジェクト・ハンドラーが含まれています。

アプリケーション固有のコンポーネントは、以下のタスクを実行します。

- イベント (例えば、ContactItem の作成) の検出およびイベントのアーカイブなどの、Exchange イベントの処理
- 統合ブローカーにパブリッシュするために、検出されたイベントに基づく該当するアプリケーション固有のビジネス・オブジェクトの作成
- 統合ブローカーサービス要求呼び出しの処理
- エラー・メッセージおよびトレース・メッセージの記録

Event Listener

Adapter for Exchange には、Event Listener と呼ばれるコンポーネントも含まれています。コネクタ自体とは異なり、Event Listener コンポーネントは、Exchange Server と同じマシンにインストールされます。そして、Exchange Server 内のサブスクライブされたイベント (例えば、項目の作成、変更、または削除) をすべて listen し、これをイベント・ストアに記録します。コネクタは、このイベント・ストアからイベントを取り出します。

このコンポーネントは、COM+ イベント・シンク・アプリケーションとしてインストールする必要があります。イベント・シンクは、定義されたトリガー (この場合はサブスクライブされたイベントの受信) によって起動されるアプリケーションです。

イベントの作成先となるフォルダーごとにイベント登録ファイルを作成し、変更に対してサブスクリプションする必要があります。イベント登録ファイルは、イベント・シンクにバインドされる隠しファイルです。Microsoft インフォメーション・ストア・サービスを開始すると、これらの隠しファイルが検索され、検索された隠しファイルはフォルダーのリストに追加されてモニターされます。モニター対象フォルダー内の項目を保管または削除すると、Information Store サービスによって、イベント登録に関連するイベント・シンクが通知されます。

Event Listener は、サブスクリプションされたイベントを受け取るたびに、以下のタスクを実行します。

- 構成プロパティの検索
- イベントからの情報の抽出
- イベントに対する固有 ID の生成
- Event Listener の構成時に指定した CWEvent ユーザー (Exchange_User と呼ばれる) にイベント情報を E メールで送信
- Windows イベント・ログにイベント・メッセージを記録

データ・ストレージおよびアクセス

Exchange Server は、パブリックまたはプライベートいずれかのストア内にデータを編成します。このようなインフォメーション・ストアには、さらにデータを編成するためのフォルダーを作成できます。プライベート・インフォメーション・ストア内のフォルダーには、受信トレイ、送信トレイ、連絡先、タスクなど、個々のユーザーのフォルダーが含まれます。パブリック・インフォメーション・ストア内のフォルダーには、組織のメンバー全員が使用可能なフォルダー (グローバル・アドレス一覧フォルダーなど) が含まれます。

Connector for Exchange Server では、カスタム・イベント・ストアおよびアーカイブ・ストアのフォルダーだけでなく、Exchange Server ユーザーのフォルダーも使用します。イベント・ストアには、Event Listener が受け取ったすべてのイベントが保持されます。また、アーカイブ・ストアには、アーカイブされたすべてのイベントが保持されます。これらのフォルダーにアクセスするために、コネクタは WebDAV API を使用します。WebDAV は HTTP 1.1 プロトコルの拡張で、アプリケーションが Exchange ストア内のフォルダー、項目、および項目プロパティにアクセスできるようになります。

各フォルダーには、項目が含まれます。項目は、Exchange Server における基本的なビジネス・エンティティであり、MailItem、ContactItem、AppointmentItem、および TaskItem などのオブジェクトを含みます。

AppointmentItems に Recipient オブジェクトが含まれる場合があります。WebDAV では Recipient オブジェクトに直接アクセスすることはできません。親 AppointmentItem を介して Recipient オブジェクトにアクセスする必要があります。

TaskItems および AppointmentItems は繰り返し可能で、同じタスクまたは予定が複数回発生することを指定します。Adapter for Exchange バージョン 1.0.x は、CDO を使用して Exchange ストア情報にアクセスしており、TaskItem の繰り返しを使用できましたが、今回の Adapter for Exchange バージョン 2.0.x では使用できませ

ん。タスクが発生するたびに別の TaskItems を作成する必要があります。
AppointmentItem の繰り返しは正常に処理されます。

各項目には、メッセージの内容を定義する複数のフィールドが含まれています。例えば、ContactItem には、FirstName フィールド、LastName フィールド、および MailingAddress フィールドがあります。WebDAV は、名前、タイプ、およびプロパティを表す URI タグで構成されるアプリケーション固有情報 (ASI) 値のプロパティ情報を使用してこれらの属性フィールドにアクセスします。例えば、ContactItem には、「Birthday」というフィールドがあります。WebDAV では、このフィールドは次のように識別されます。

- Name = Birthday
- Type = Date
- AppSpecificInfo = PropertyName=urn:schemas:contacts:bday;Access=Get/Put

さらに、Outlook を使用すると 4 つのカスタム・フィールドをその ContactItem に追加できます。これらのフィールドにアクセスするには、「UserField1」や「UserField2」などの名前を使用します。

コネクターの動作方法

コネクターは、要求およびイベントの 2 つのモードで動作します。要求モードの場合、要求された操作を表すビジネス・オブジェクトをコネクターに送信して、統合ブローカーが Exchange Server に要求を発行したときにプロセスが開始されます。コネクターは、アクティブな動詞に基づいてそのオブジェクトを処理し、その後 Exchange Server に該当する要求を送信します。要求の性質によっては、コネクターは情報を処理して Exchange Server から統合ブローカーに戻すこともあります。

イベント・モードの場合、プロセスは Exchange Server で開始されます。コネクターはサーバー上のイベント・ストアをポーリングして、サブスクライブされた Exchange イベントのレコードを探します。このようなイベントが検出されるたびに、コネクターは該当するビジネス・オブジェクトを作成し、WebDAV 呼び出しでユーザーのフォルダーから検索されたデータを取り込みます。その後、コネクターはビジネス・オブジェクトを統合ブローカーにパブリッシュします。

この処理については、以下のセクションで詳しく説明します。

要求

統合ブローカーからの要求は、Retrieve、Create、Update、または Delete の 4 つの操作のいずれかの形式となります。Retrieve は、Exchange データベースから情報を検索することをコネクターに要求します。Create、Update、または Delete は、初期要求に含まれる情報に基づいて、それ自体に対して該当するアクションを実行することをデータベースに指示するようにコネクターに要求します。

Retrieve

統合ブローカーが、コネクターに Exchange から情報を検索することを要求する場合、基本キー値と呼ばれる値を設定します。この値を使用して、コネクターは Exchange データベースから該当する情報を取り出すことができます。要求される情

報は多くの場合複雑なので、情報は通常、階層で保管されています。コネクタは基本キー値を使用して、データベースの項目階層を再帰的に下降します。

コネクタが情報をブローカーに戻すために作成するビジネス・オブジェクトもまた階層で、階層ビジネス・オブジェクトと呼ばれます。階層ビジネス・オブジェクトは、1つのトップレベルのビジネス・オブジェクトと1つ以上の子ビジネス・オブジェクトから構成されます。コネクタは、データベース内の項目の構造に正確に一致するビジネス・オブジェクトを構成しようとします。つまり、統合ブローカーに戻される個別のビジネス・オブジェクトのすべての単純属性が、Exchange データベース内の対応するフィールドの値と一致し、戻される階層ビジネス・オブジェクトに含まれる子ビジネス・オブジェクトの数がデータベース内の子の数および順序と一致します。

注: 階層ビジネス・オブジェクト という用語は、格納されているすべての子ビジネス・オブジェクトを含めた、ビジネス・オブジェクト全体を表します。個別ビジネス・オブジェクト という用語は、そのビジネス・オブジェクトが格納している (あるいは格納されている) 子ビジネス・オブジェクトから独立した、単一のビジネス・オブジェクトを表します。トップレベルのビジネス・オブジェクト という用語は階層のトップレベルにある個別ビジネス・オブジェクトを表します。このオブジェクトは、親ビジネス・オブジェクトを持ちません。

Create

統合ブローカーが、ブローカーがコネクタに渡すビジネス・オブジェクトに基づいて、データベース内に新しい Exchange Server 項目を作成するようにコネクタに要求すると、コネクタは以下のステップを実行します。

1. 所有権を持って含まれている、それぞれの単一カーディナリティーの子ビジネス・オブジェクトを再帰的に処理します。
2. 所有権を持たずに含まれている、それぞれの単一カーディナリティーの子ビジネス・オブジェクトを処理します。
3. トップレベルのビジネス・オブジェクトを処理します。
4. 親/子関係を子エンティティーに格納する単一カーディナリティーの子ビジネス・オブジェクトをそれぞれ処理します。
5. それぞれの複数カーディナリティーの子ビジネス・オブジェクトを処理します。

Update

統合ブローカーが、データベース内の Exchange Server 項目を更新するようにコネクタに要求すると、コネクタは以下のステップを実行します。

1. ソース・ビジネス・オブジェクトの基本キー値を使用して、対応するエンティティーをデータベースから検索します。
2. ソース・ビジネス・オブジェクトの情報を使用して、トップレベルの項目の単一カーディナリティーの子をすべて、再帰的に更新します。
3. ソース・ビジネス・オブジェクト内の対応する属性に値 CxIgnore が含まれている単純属性を除いて、検索した項目のすべての単純属性を更新します。
4. ソース・ビジネス・オブジェクトの情報を使用して、検索した項目のすべての配列を処理します。

Delete

統合ブローカーが、データベースから階層ビジネス・オブジェクトを削除するようにコネクターに要求すると、コネクターはトップレベルの項目を物理的に削除します。

イベント

イベントとは、任意の Exchange Server 項目の作成、変更、または削除です。Exchange インフォメーション・ストアは、イベントが発生したことを Event Listener に通知します。イベントはイベント・ストアに配置されて、イベント ID が指定されます。コネクターはその pollForEvents () メソッドを呼び出し、イベントで表される情報は、イベント通知で検出された ItemURI を使用して検索されて処理されます。

Create

コネクターは、イベント・ストアで Create イベントを検出すると、イベントで指定する型のビジネス・オブジェクトと構造が一致するビジネス・オブジェクトを新規に作成し、ItemURI フィールドにイベントで検出された値を設定します。さらにコネクターは、データベースから関連する項目を検索し、それを使用して新規ビジネス・オブジェクトを設定します。ビジネス・オブジェクトの動詞は Create に設定され、コネクターはこれを統合ブローカーにパブリッシュします。

Update

コネクターは、イベント・ストアで Update イベントを検出すると、イベントで指定する型のビジネス・オブジェクトと構造が一致するビジネス・オブジェクトを新規に作成し、ItemURI フィールドにイベントで検出された値を設定します。さらにコネクターは、データベースから関連する項目を検索し、それを使用して新規ビジネス・オブジェクトを設定します。ビジネス・オブジェクトの動詞は Update に設定され、コネクターはこれを統合ブローカーにパブリッシュします。

Delete

コネクターは、イベント・ストアで Delete イベントを検出すると、イベントで指定する型のビジネス・オブジェクトと構造が一致するビジネス・オブジェクトを新規に作成し、ItemURI フィールドにイベントで検出された値を設定します。ビジネス・オブジェクトの動詞は Delete に設定され、コネクターはこれを統合ブローカーにパブリッシュします。

コネクターは、Exchange Server によって起動される物理削除操作のみをサポートしています。

イベント処理

Exchange Server オブジェクトが作成、更新、または削除されるたびに、コネクターの Event Listener はイベント通知またはイベントをイベント・ストアに保管します。コネクターが pollForEvents() メソッドを呼び出すと、イベント・ストア内の通知が検出されて、その後処理されます。

PollQuantity コネクター・プロパティーは、コネクターがイベント・ストアから検索するイベントの数を指定します。コネクターが検索するイベントごとに、以下のタスクが実行されます。

- コネクターが、イベント状況を IN_PROGRESS に更新します。
- コネクターが、ビジネス・オブジェクトに対して生成されたイベントに統合ブローカーがサブスクライブされているかどうかを判別し、以下のいずれかを実行します。
 - サブスクリプションがある場合、コネクターが必要なアクション (Create、Update、または Delete) を実行します。
 - サブスクリプションがない場合、コネクターがイベント状況を UNSUBSCRIBED に更新し、ArchiveProcessed コネクター・プロパティの設定に応じて、イベントをアーカイブします。

表 1 に、イベントが使用する状況値を示します。

表 1. イベント状況

イベント状況	状況の数値	説明
READY_FOR_POLL	0	イベントは次のポーリング呼び出しで選出されるための準備が整っています。
IN_PROGRESS	1	コネクターがイベントを選出し、処理しています。
UNSUBSCRIBED	2	このイベントにサブスクリプションはありません。
SUCCESS	3	コネクターはイベントを正常に処理しました。
ERROR_PROCESSING_EVENT	-1	イベントの処理中にコネクターでエラーが発生しました。
ERROR_POSTING_EVENT	-2	統合ブローカーへのビジネス・オブジェクトのパブリッシング中にコネクターでエラーが発生しました。
ERROR_OBJECT_NOT_FOUND	-3	イベント作成先のビジネス・オブジェクトが見つかりません。

イベント通知

イベント通知を使用するには、2 つの Exchange Server ユーザー・フォルダー (イベントの保管用と、イベントのアーカイブ用) を作成する必要があります。この 2 つのフォルダーの作成および構成に関する説明については、13 ページの『コネクターのインストール』を参照してください。

イベントのアーカイブ

ArchiveProcessed コネクター・プロパティを使用して、イベント状況の更新後にイベントがアーカイブされているかどうかを判別します。8 ページの表 2 に、ArchiveProcessed の設定を示します。

表 2. ArchiveProcessed プロパティの使用法

ArchiveProcessed 値	イベント状況	コネクタの振る舞い
true	イベントは正常に処理されました	コネクタによって、状況が SUCCESS のイベントがアーカイブされます。イベントはアーカイブされると、イベント・ストアから削除されません。
	イベントの処理に失敗しました	コネクタによって、状況が ERROR_POSTING_EVENT または ERROR_OBJECT_NOT_FOUND のイベントがアーカイブされます。イベントはアーカイブされると、イベント・ストアから削除されます。
	ビジネス・オブジェクトに対するサブスクリプションがないため、イベントが処理されません	コネクタによって、状況が UNSUBSCRIBED のイベントがアーカイブされます。イベントはアーカイブされると、イベント・ストアから削除されます。
false	イベントは正常に処理されました	イベント・ストアにイベントが残ります。
	イベントの処理に失敗しました	イベント・ストアに状況 ERROR_POSTING_EVENT、ERROR_OBJECT_NOT_FOUND、または ERROR_PROCESSING_EVENT のイベントが残ります。
	ビジネス・オブジェクトに対するサブスクリプションがないため、イベントが処理されません	イベント・ストアに状況 UNSUBSCRIBED のイベントが残ります。

イベントのリカバリー

コネクタを使用して、システム障害後にイベントをリカバリーします。始動時に、コネクタは `recoverInProgressEvents()` メソッドを呼び出し、前回コネクタ・セッションを終了したときに状況が `IN_PROGRESS` であったイベントを検索します。

`InDoubtEvents` コネクタ・プロパティを使用して、イベント・リカバリーのタイプ (ある場合) を決定します。このプロパティの設定については、23 ページの『アプリケーション固有の構成プロパティ』を参照してください。

保証付きイベント・デリバリー

コネクタは、コネクタ・フレームワークの重複イベント除去 (DEE) 機能および `setDEEID()` メソッドを利用して、保証付きイベント・デリバリー (GED) をサポートしています。そのため、イベントを統合ブローカーにデリバリーした後、イベント・ストア内のイベントの状況を更新する前にコネクタが終了した場合でも、イベントは統合ブローカーへ 1 度のみ送信されます。

GED を利用するためには、表 3 に示すように、3 つのコネクター構成プロパティを設定する必要があります。

表 3. GED を使用可能にするコネクター・プロパティの設定

プロパティ名	説明	GED に必要な値
DuplicateEventElimination	標準構成プロパティ。True に設定すると、イベント ID がモニター・キューに格納され、重複イベントのデリバリーを防ぎます。	True
MonitorQueue	標準構成プロパティ。コネクターが処理済みイベント ID を格納する JMS キューを指定します。	適切なキューに設定する必要があります (例えば、MONITORQUEUE)
InDoubtEvents	コネクター固有の構成プロパティ。コネクター終了時に状況が IN_PROGRESS であったイベントを処理する方法を指定します。	Reprocess

コネクター構成プロパティの設定の詳細については、47 ページの『付録 A. コネクターの標準構成プロパティ』を参照してください。

トランザクション・サポート

Exchange Server Extensible Storage Engine (ESE) によって、トランザクションのサポートが提供されています。ESE は、トランザクション・ロギング・システムであり、システム障害の後もデータ保全性を保証します。イベント・ストアおよびアーカイブ・ストアが Exchange Server メールボックスに配置されているため、ESE はイベント処理トランザクションを打ち切るかまたはロールバックする機能をサポートしています。

Event Listener がイベントを処理できない場合 (使用不可になっているため、またはシステム・エラーのため)、イベント・サービスはイベントを保持し、Event Listener が復元したときにそのイベントを Event Listener にデリバリーします。

アダプター環境

アダプターをインストール、構成、および使用する前に、以下のセクションで説明するような環境の要件を把握しておく必要があります。

- 『ブローカーの互換性』
- 10 ページの『インストールの前提条件』
- 11 ページの『ロケール依存データの処理』

ブローカーの互換性

アダプターが使用するアダプター・フレームワークは、アダプターと通信する統合ブローカーのバージョンとの互換性を備えている必要があります。Adapter for Exchange のバージョン 2.0.x は、以下のアダプター・フレームワークおよび以下の統合ブローカーでサポートされています。

- **アダプター・フレームワーク:** WebSphere Business Integration Adapter Framework
バージョン 2.1.0、2.2.0、2.3.0、2.3.1、2.4.0、および 2.4.1
- **統合ブローカー:**
 - WebSphere InterChange Server、バージョン 4.2.x
 - WebSphere MQ Integrator、バージョン 2.1.0
 - WebSphere MQ Integrator Broker、バージョン 2.1.0
 - WebSphere Business Integration Message Broker 5.0 (CSD02 を適用済み)
 - WebSphere Application Server Enterprise、バージョン 5.0.2 (WebSphere Studio Application Developer Integration Edition バージョン 5.0.1 と併用)

例外については、「リリース情報」を参照してください。

注: 統合ブローカーのインストールおよびその前提条件に関する説明については、以下の資料を参照してください。WebSphere InterChange Server (ICS) については、「システム・インストール・ガイド (UNIX 版)」または「システム・インストール・ガイド (Windows 版)」を参照してください。Message Brokers (WebSphere MQ Integrator Broker、WebSphere MQ Integrator、および WebSphere Business Integration Message Broker) の場合は、「*WebSphere Message Brokers 使用アダプター・インプリメンテーション・ガイド*」およびそれぞれの Message Brokers のインストールに関する資料を参照してください。一部の資料は次の Web サイトにあります。

<http://www.ibm.com/software/integration/mqfamily/library/manualsa/>。

WebSphere Application Server については、「*アダプター実装ガイド (WebSphere Application Server)*」および次の資料を参照してください。

<http://www.ibm.com/software/webservers/appserv/library.html>

インストールの前提条件

コネクターのインストール先となるマシンに以下のソフトウェアがインストールされ、かつ構成が完了している必要があります。

- Adapter Application Development Kit (ADK)
- 9 ページの『ブローカーの互換性』に示されている統合ブローカーのいずれか

注: バージョン 2.0.x では、Exchange コネクターのインストールは Windows 2000 に限定されていません。AIX、HP、および Solaris 8.0 もサポートします。

本書では、アダプターのインストールについてのみ説明します。このセクションに示した前提条件となるソフトウェアをインストールする際には、適切な製品資料を参照してください。

アダプターおよび関連ファイルのインストール

WebSphere Business Integration Adapter 製品のインストールについては、「*WebSphere Business Integration Adapters インストール・ガイド*」を参照してください。この資料は、次の Web サイトの WebSphere Business Integration Adapters Infocenter にあります。

<http://www.ibm.com/websphere/integration/wbiadapters/infocenter>

ロケール依存データの処理

コネクタは国際化されています。2 バイト文字セット (DBCS) を、同じく 2 バイト文字セットをサポートするインターフェースに送信することができ、また指定された言語のメッセージ・テキストの送信をサポートします。ある文字コードを使用する場所から別の文字コード・セットを使用する場所へデータを転送する場合、コネクタは、そのデータの意味が伝わるように文字変換を実行します。

Java 仮想マシン (JVM) 内部の Java ランタイム環境では、Unicode 文字コード・セットでデータを表現します。Unicode には、最もよく知られた文字コード・セット (単一バイトとマルチバイトの両方) の文字エンコードが含まれています。

WebSphere Business Integration システム内のほとんどのコンポーネントは、Java で書かれています。そのため、統合コンポーネント間でデータを転送する際に、ほとんどの場合文字変換は必要ありません。

第 2 章 アダプターのインストール

本章では、IBM WebSphere Business Integration Adapter for Exchange のインストール方法を説明します。本章の内容は、次のとおりです。

- 『コネクターのインストール』
- 14 ページの『Event Listener のインストール』

コネクターのインストール

このセクションでは、コネクターおよび関連ビジネス・オブジェクトをインストールするために必要な操作を説明します。ソフトウェアの前提条件および互換性については、9 ページの『アダプター環境』を参照してください。

マシン上で、前提条件となるソフトウェアすべてのインストールを完了すると、コネクターおよびビジネス・オブジェクトをインストールできるようになります。

アダプター (コネクターおよびビジネス・オブジェクトを含む) のインストールについては、次の Web サイトで WebSphere Business Integration Adapters Infocenter にある「WebSphere Business Integration Adapters インストール・ガイド」を参照してください。

<http://www.ibm.com/websphere/integration/wbiadapters/infocenter>

以下の表に、コネクターとともにインストールされるファイルを示します。表中の *ProductDir* は、IBM WebSphere Business Integration Adapter ソフトウェアをインストールしたディレクトリーを指します。

インストールされる Windows ファイル構造

以下の表に、Windows マシンにインストールする場合にコネクターで使用されるファイル構造を示します。

表 4. コネクターとともにインストールされるファイル - Windows

ディレクトリー	インストールされるファイル
<i>ProductDir</i> \connectors \Exchange	start_Exchange.bat: コネクター始動ファイル BIA_Exchange.jar: コネクター・コード
<i>ProductDir</i> \connectors \Exchange\dependencies	webdav.jar: WebDAV ライブラリー
<i>ProductDir</i> \bin\DATA\App	BIA_ExchangeConnectorTemplate: コネクターの構成ファイル・テンプレート
<i>ProductDir</i> \connectors \messages	BIA_ExchangeConnector.txt: コネクター・メッセージ・ファイル
<i>ProductDir</i> \repository \Exchange\2000	AppointmentItem.xsd、ContactItem.xsd、MailItem.xsd、Recipient.xsd、RecurrencePattern.xsd、TaskItem.xsd : サンプル・ビジネス・オブジェクト

インストールされる UNIX ファイル構造

以下の表に、UNIX マシンにインストールする場合にコネクタで使われるファイル構造を示します。

表 5. コネクタとともにインストールされるファイル - UNIX

ディレクトリー	インストールされるファイル
<i>ProductDir</i> /connectors /Exchange	start_Exchange.sh: コネクタ始動ファイル BIA_Exchange.jar: コネクタ・コード
<i>ProductDir</i> /connectors /Exchange/dependencies	webdav.jar: WebDAV ライブラリー
<i>ProductDir</i> /connectors /messages	BIA_ExchangeConnector.txt: コネクタ・メ ッセージ・ファイル
<i>ProductDir</i> /repository /Exchange/2000	AppointmentItem.xsd、ContactItem.xsd、 MailItem.xsd、Recipient.xsd、 RecurrencePattern.xsd、TaskItem.xsd : サンプル・ビジネス・オブジェクト

ビジネス・オブジェクトのインストール

アダプターには、標準の Exchange 項目を対象とするサンプル・ビジネス・オブジェクトが多数用意されています。独自のビジネス・オブジェクトを作成するとき、これらを使用することもできます。サンプル・ビジネス・オブジェクトにアクセスするには、以下の手順に従います。

1. 上記の表にリストされているご使用のオペレーティング・システムの該当する .xsd ディレクトリーにナビゲートします。
2. Connector Configurator を使用して .xsd ファイルを開き、そのファイルをユーザーのプロローカー・リポジトリーに保存します。
3. コネクタ構成プロパティ RepositoryDirectory を使用して、サンプル・ビジネス・オブジェクトを配置するディレクトリーが指定されていることを確認します。

Event Listener のインストール

このセクションでは、Event Listener をインストールするために必要な操作を説明します。また、Event Listener をインストールする前に、前提条件としてマシン上に備えておく必要のあるソフトウェアを示します。

Event Listener のインストール前提条件

Event Listener のインストール先となるマシンに、以下のソフトウェアがインストールされ、かつ構成が完了している必要があります。

- Microsoft Windows 2000
- Microsoft Exchange Server 2000
- Outlook 2000 (ユーザー・アカウントのセットアップ用)

- Microsoft Collaboration Data Objects (CDO) ライブラリー・バージョン 1.21 (exoledb.dll が実行可能であること)

CDO ライブラリーは、Exchange Server 2000 および Outlook 2000 に付属されています。

- Microsoft Exchange Information Store サービスおよび Directory サービス

これらのサービスは、Microsoft Exchange Server 2000 に付属されています。

- 以下のイベント登録用ツールのいずれか

- Exchange Explorer: この GUI は、Microsoft の Web サイトからダウンロードする必要があります。

注: Exchange Explorer を使用するには、Microsoft Visual Basic 6.0 以降がマシンにインストールされていなければなりません。

- regevent.vbs: このユーティリティーは、Microsoft Visual Studio に付属しています。Microsoft の Web サイトからダウンロードすることもできます。

Event Listener のインストール

以下のようにして、Exchange Server が稼働しているマシン上に Event Listener の実行可能ファイル (BIA_Exchange.dll) および構成ファイル (BIA_Exchange.ini) をインストールする必要があります。

1. WebSphere Business Integration Adapter の CD-ROM を挿入し、CD の %connectors%\Exchange\dependencies\EventListener ディレクトリーに移動します。
2. BIA_Exchange.dll ファイルを CD-ROM から Exchange Server マシンにコピーします。ファイルのインストール先としてお勧めするロケーションは、C:\Program Files\Exchsrvr\bin ですが、ほかのロケーションにファイルをインストールすることもできます。他のユーザーが誤ってファイルを削除することがないように、BIA_Exchange.dll のインストール先となるディレクトリーに適切なセキュリティが設定されており、他のユーザーや管理者がシステムが使用している DLL であると認識できる情報が十分に提供されていることを確認してください。
3. BIA_Exchange.ini ファイルを、Adapter の CD-ROM (%connectors%\Exchange\dependencies\EventListener) から Exchange Server マシンの C:\WINNT ディレクトリーにコピーしてください。

第 3 章 タスクの構成

本章では、アダプターが Exchange Server と通信できるようにするために必要な構成タスクについて説明します。本章の内容は、次のとおりです。

- 『Event Listener に必要な構成タスク』
- 22 ページの『コネクターに必要な構成タスク』
- 25 ページの『イベントの登録』
- 28 ページの『アダプターの始動』
- 31 ページの『アダプターの停止』
- 32 ページの『1 台のサーバー上に複数のアダプター・インスタンスを作成』

Event Listener に必要な構成タスク

Event Listener コンポーネントは、COM+ コンポーネントとして Exchange Server にインストールされるイベント・シンクです。Exchange Server が稼働するマシンで、以下の構成タスクを実行する必要があります。

1. イベント・シンクの実行に必要なユーザー・アカウントを作成し、作成したアカウントに適切な権限を割り当てます。『イベント・シンクのユーザー・アカウントおよび権限の設定』を参照してください。通常的环境下では、コネクターのユーザー・アカウントも作成します。
2. Event Listener 実行可能ファイル (BIA_Exchange.dll) を Exchange Server マシンに登録します。18 ページの『Event Listener の登録』を参照してください。
3. Exchange Server 用の COM+ アプリケーションを作成し、作成したアプリケーションに Event Listener コンポーネントを追加します。18 ページの『COM+ アプリケーションの作成』を参照してください。
4. Event Listener 構成ファイルを変更します。20 ページの『構成ファイルの変更』を参照してください。
5. Windows イベント・ビューアーで、Event Listener のメッセージを表示させるようにします。21 ページの『Event Listener で使用する Windows イベント・ビューアーの構成』を参照してください。

イベント・シンクのユーザー・アカウントおよび権限の設定

Event Listener などの Exchange 2000 イベント・シンクはプロセス不足になるため、実行するには何らかの形式での認証が必要です。したがって、イベント・シンクを実行する権限を持つユーザー・アカウントを作成する必要があります。本書では、このアカウントに一般名 *Exchange_User* を使用します。ほとんどの場合、このアカウントはコネクターのユーザー・アカウントとしても機能します。

1. Windows システム管理ツールを使用してユーザー・アカウントを作成します。ユーザーがドメインの一部になる場合は (通常)、ユーザー・アカウントは Microsoft Active Directory を使用して追加する必要があります。その他の場合は、ユーザー・アカウントはコンピューターの「ローカル ユーザーとグループ」プログラムを使用して作成する必要があります。

2. Exchange System Manager を開きます。これは、新規アカウントの権限を設定するために使用します。
3. 使用するサーバーに達するまでサーバー・ツリーをナビゲートします。例えば、管理グループとサーバー・グループをクリックして行く必要がある場合があります。
4. サーバーを右クリックします。ポップアップ・メニューが表示されます。
5. 「プロパティ」をクリックします。
6. 「セキュリティ」タブをクリックします。
7. 「名前」ダイアログ・ボックスで「追加」をクリックして、表示されたリストからユーザーを選択します。
8. 権限レベルについて「フル コントロール」を選択します。
9. 「OK」をクリックします。

詳細については、Microsoft Knowledge Base の記事 262054 (*XADM: How to Get Service Account Access to All Mailboxes in Exchange 2000*) を参照してください。

注: 変更が有効になるまで数時間かかることがあります。変更を即時に有効にするには、Exchange Server をリブートします。

Event Listener の登録

以下の手順に従って、Regsvr32 ユーティリティを用いて Event Listener コンポーネントを Exchange Server マシンに登録する必要があります。

1. 「スタート」→「ファイル名を指定して実行」を選択します。「ファイル名を指定して実行」ダイアログ・ボックスが表示されます。
2. 「名前」フィールドに次のように入力します。

```
regsvr32 installation_directory¥BIA_Exchange.dll
```

この場合の *installation_directory* は、BIA_Exchange.dll ファイルをインストールしたロケーション (例えば、C:¥Program Files¥Exchsrvr¥bin) を指します。

Event Listener ファイルが正常に登録されたことを確認するためのダイアログ・ボックスが表示されます。

COM+ アプリケーションの作成

以下のステップを実行して、Exchange Server マシンに COM+ アプリケーションを作成します。この COM+ アプリケーションは、Event Listener のイベント・シンクを実行するために必要です。

1. 「スタート」→「プログラム」→「管理ツール」→「コンポーネント サービス」を選択します。「コンポーネント サービス」ダイアログ・ボックスが表示されます。
2. 「コンポーネント サービス」ツリーの横にある正符号 (+) をクリックして、ツリーを拡張します。
3. 「コンポーネント サービス」の下にある「COM+ アプリケーション」フォルダーを右マウス・ボタンでクリックし、「新規作成」→「アプリケーション」を選択します。「COM アプリケーション インストール ウィザードへようこそ」が起動します。

4. 「ようこそ」ダイアログ・ボックスの「次へ」をクリックします。「新しいアプリケーションのインストールまたは作成」ダイアログ・ボックスが表示されます。
5. 「空のアプリケーションを作成する」をクリックします。「空のアプリケーションを作成」ダイアログ・ボックスが表示されます。
6. 「新しいアプリケーション名の入力」フィールドに「BIA_Exchange」と入力します。
7. 「アクティブ化の種類」ボックスで「サーバー アプリケーション」を選択して、「次へ」をクリックします。「アプリケーション ID の設定」ダイアログ・ボックスが表示されます。
8. 「アカウント」ボックスで「このユーザー」を選択して、「参照」をクリックします。「ユーザーまたはグループの選択」ダイアログ・ボックスが表示されます。
9. 17 ページの『イベント・シンクのユーザー・アカウントおよび権限の設定』で設定した Exchange ユーザー名を選択して、「OK」をクリックします。選択したユーザー名が「ユーザー」フィールドに挿入されます。
10. 「パスワード」フィールドに Exchange_User のパスワードを入力します。
11. 「パスワードの確認入力」フィールドにパスワードを再入力して、「次へ」をクリックします。
12. 「完了」をクリックすると、アプリケーションが作成され、ウィザードが終了します。これにより、「COM+ アプリケーション」フォルダーの下に、対応する「コンポーネント」および「ロール」サブディレクトリーが含まれた BIA_Exchange アプリケーション・フォルダーが追加されます。

COM+ アプリケーションへの Event Listener の追加

新しい COM+ アプリケーションの作成が完了したら、以下の手順に従って、作成したアプリケーションに Event Listener コンポーネントを追加する必要があります。

1. 「コンポーネント サービス」ダイアログ・ボックスの BIA_Exchange アプリケーション・フォルダーにナビゲートします。
2. 「コンポーネント」フォルダーを右マウス・ボタンでクリックして、「新規作成」を選択します。「COM アプリケーション インストール ウィザードへようこそ」が起動します。
3. 「ようこそ」ダイアログ・ボックスの「次へ」をクリックします。「新しいアプリケーションのインストールまたは作成」ダイアログ・ボックスが表示されます。
4. 「新しいコンポーネントをインストールする」を選択して、「次へ」をクリックします。「新しいコンポーネントのインストール」ダイアログ・ボックスが開きます。
5. 「インストールするファイル」ウィンドウの「追加」をクリックします。
6. マシンにインストール済みの BIA_Exchange.dll ファイルを参照して、「OK」をクリックします。ウィザードによってファイルが分析され、「見つかったコンポーネント」ウィンドウに Events コンポーネントがリストされます。

7. 「次へ」をクリックしてから、「完了」をクリックしてウィザードを終了します。BIA_Exchange.Events.1 コンポーネントが、BIA_Exchange アプリケーションの「コンポーネント」フォルダーに配置されます。

構成ファイルの変更

Event Listener には、BIA_Exchange.ini と呼ばれる構成ファイルが必要です。このファイルには、Event Listener で使用する構成プロパティが含まれています。表 6 に、編集または設定する必要のあるプロパティを示します。ここにユーザー名が指定されているので、Event Listener がそれ自体の原因で発生したイベントをフィルターで排除し、「ピンポン」現象を避けることができます。

表 6. Event Listener 構成プロパティ

プロパティ	説明	デフォルト値
CwEventID	前回使用したイベント ID。	ゼロ (0)
CwEventUser	Exchange_User アカウントについてのユーザーのログオン名を指定します。Event Listener はこのアカウントのメールボックスに通知を送信します。	CwEvent
CwAgentUsername	コネクターのアカウントのフルネームを指定します。ほとんどの場合は Exchange_User と同じですが、別個のアカウントを 2 つ持つことができます。ただし、両アカウントの権限は、すべてのメールボックスにアクセスできるように設定されている必要があります。	CwAgent
CwAgentUserGUID	コネクターのアカウントの GUID を指定します。ほとんどの場合、このアカウントは Exchange_User と同じです。	N/A
CwEventUsername	Exchange_User アカウントのフルネームを指定します。	CwEvent
EventsTraceLevel	Event Listener のトレース・メッセージのレベルを指定します。	0 (トレースなし)

ファイルを以下のように変更してください。

- CwEventUser のエントリを変更して、17 ページの『イベント・シンクのユーザー・アカウントおよび権限の設定』で設定した Exchange_User アカウントの名前を指定します。Event Listener はイベントの通知を受け取ると、このユーザー・アカウントにイベント情報を記載した E メール・メッセージを送信します。
- CwAgentUserGUID のエントリに、コネクターに関連したユーザー・アカウントの GUID を指定します。

注: この値を取得するには、Event Listener を起動して、そのログ項目を調べると便利です。例えば、次のようにします。

1. EventsTraceLevel を 5 に設定します。
 2. イベント登録を Tasks フォルダーに作成します (25 ページの『イベントの登録』を参照してください)。
 3. Outlook を使用して新規タスクを作成します。これにより、Event Listener が起動されます。
 4. イベント・ビューアーを使用して、このトリガーに対応するアプリケーション・ログのエントリーを調べ、CwAgentUserGUID の値を探します。
- さらにフィルタリングする場合は、CwEventUsername エントリーおよび CwAgentUserName エントリーを変更して、フィルタリングする Exchange ユーザー・アカウントの表示名を指定してください。

Event Listener で使用する Windows イベント・ビューアーの構成

Event Listener は、エラー・メッセージおよびトレース・メッセージを Windows イベント・ビューアーに記録します。Windows イベント・ビューアーが Event Listener のメッセージを検出して正しく表示できるようにするためには、Windows レジストリーを手動で編集する必要があります。

注意

Windows レジストリーを誤って編集すると、ご使用のマシンに問題が生じることがあります。レジストリーの編集は慎重に行ってください。レジストリーに精通していない場合は、Windows システム管理者に援助を求めてください。

Windows レジストリーを更新するには、以下のステップを実行します。

1. Windows イベント・ビューアーが開いている場合、閉じておきます。
2. 「スタート」→「ファイル名を指定して実行」をクリックし、「ファイル名を指定して実行」ダイアログ・ボックスに regedit と入力します。Windows レジストリーが開きます。
3. キー HKEY_LOCAL_MACHINE¥SYSTEM¥CurrentControlSet¥Services¥Eventlog ¥Application に移動し、そのキーを強調表示します。
4. 右マウス・ボタンでクリックしてコンテキスト・メニューを表示させてから、「新規」→「キー」をクリックします。新規のキーがレジストリーの Application キーの下に追加されます。
5. キーに BIA_Exchange という名前を付けます。
6. Windows レジストリーの右ペインを右マウス・ボタンでクリックし、コンテキスト・メニューを表示させます。「新規」→「文字列値」をクリックします。新規の記入項目が追加されます。
7. 新規記入項目に EventMessageFile という名前を付けます。
8. EventMessageFile 記入項目をダブルクリックし、「文字列の編集」ダイアログ・ボックスを開きます。

9. 「値のデータ」フィールドで、BIA_Exchange.dll ファイルの絶対パス名 (例えば、`ProductDir¥connectors¥Exchange¥dependencies¥eventListener¥BIA_Exchange.dll`) を指定してから、「OK」をクリックします。
10. Windows レジストリーの右ペインを右マウス・ボタンでクリックし、コンテキスト・メニューを表示させます。「新規」→「DWORD 値」をクリックします。新規の記入項目がキーに追加されます。
11. 新規記入項目に TypesSupported という名前を付けます。
12. TypesSupported 記入項目をダブルクリックし、「DWORD 値の編集」ダイアログ・ボックスを開きます。
13. 「値のデータ」フィールドで 7 と入力し、デフォルト基数 (16 進数) が選択された状態のままにしておきます。「OK」をクリックします。
14. 「レジストリ」→「レジストリ エディタの終了」をクリックして、Windows レジストリーを閉じます。

コネクタに必要な構成タスク

コネクタ用に、以下の構成タスクを実行する必要があります。

- ご使用のインストール・システムで Exchange_User アカウントとは異なるアカウントが必要な場合は、コネクタのユーザー・アカウントを作成します。通常的环境下では、コネクタは Exchange_User アカウントを使用します。『コネクタのユーザー・アカウントの作成』を参照してください。

注: 次のステップでは、Outlook 2000 が Windows マシンにインストールされている必要があります。

- アーカイブ・ストアを作成します。『アーカイブ・ストアの作成』を参照してください。
- コネクタ構成プロパティを設定します。23 ページの『コネクタ構成プロパティの設定』を参照してください。

コネクタのユーザー・アカウントの作成

ほとんどの場合、17 ページの『イベント・シンクのユーザー・アカウントおよび権限の設定』で作成した Exchange_User アカウントはコネクタが使用するアカウントと同じです。ご使用のインストール・システムで、何らかの理由によりコネクタのアカウントが Exchange_User アカウントと異なる必要がある場合は、Windows システム管理ツールを使用してアカウントを作成します。

アーカイブ・ストアの作成

最初にコネクタのユーザー・アカウントで Outlook にログインすると、Outlook により、アダプターのイベント・ストアとして機能する受信トレイが自動的に作成されます。アーカイブ・ストアとして機能するそのアカウントのメール・フォルダーも追加する必要があります。このためには、次の手順で行います。

1. Outlook クライアントを開き、「Outlook Today」— 「メールボックス - Exchange_user_name」を選択します。
2. 以下の値を用いて新規のフォルダーを作成します。新規フォルダーの作成の詳細については、Outlook の資料を参照してください。

表 7. アーカイブ・ストア・フォルダーのフィールド値

フィールド名	入力する値
名前	Archive 注: この値をコネクターの ArchiveFolder 構成プロパティに使用する必要があります。
フォルダに保存するアイテム	Mail Items
フォルダを作成する場所	Mailbox-user_name。ここで、user_name は、アーカイブ・ストアに使用するアカウントの名前です。通常は Exchange_User アカウントですが、コネクターがアクセスできるアカウントであればどれでも構いません。

コネクター構成プロパティの設定

アダプターのコネクター・コンポーネントには、標準構成プロパティとアプリケーション固有の構成プロパティの、2 つのタイプの構成プロパティがあります。コネクターを実行する前に、これらのプロパティの値を設定する必要があります。

標準構成プロパティ

標準のコネクター・プロパティを構成するには、Connector Configurator ツールを使用します。詳細については、67 ページの『付録 B. Connector Configurator』に記載されています。このツールはコネクター構成用のグラフィカル・ユーザー・インターフェースを提供します。「標準構成プロパティ (Standard Config Properties)」タブをクリックして、構成プロパティを追加または変更します。

コネクター構成プロパティの値を指定すると、Connector Configurator は指定した値をアダプター・リポジトリに保管 (ICS の場合)、または構成ファイルを生成してアダプターのローカル・リポジトリに配置 (WebSphere MQ Integrator Broker の場合) します。

コネクターは、始動時に構成値を取得します。実行時のセッション中に、1 つ以上のコネクター・プロパティの値を変更できます。

- AgentTraceLevel など、一部のコネクター構成プロパティへの変更は動的であり、即時に有効になります。
- それ以外のコネクター・プロパティへの変更は静的であり、変更後にコンポーネントまたはシステムの再始動が必要です。

プロパティが動的であるか静的であるかを判別するには、Connector Configurator の「更新メソッド」欄を参照してください。

アプリケーション固有の構成プロパティ

アプリケーション固有のコネクター構成プロパティは、コネクターが実行時に必要とする情報を提供します。また、コネクター内の静的情報やロジックを、再コーディングや再ビルドを行わずに変更する方法も提供します。

これらのプロパティを構成するには、Connector Configurator を使用します。「アプリケーション構成プロパティ」タブをクリックして、構成プロパティの追加または変更を行います。詳細については、67 ページの『付録 B. Connector Configurator』を参照してください。

表 8 に、コレクターのアプリケーション固有の構成プロパティと、プロパティの説明および可能な値を示します。

表 8. Exchange コネクターのアプリケーション固有の構成プロパティ

プロパティ	説明	指定可能な値	デフォルト値	必須?
ArchiveFolderURI	イベントがアーカイブされるディレクトリーの完全修飾名	イベント・アーカイブ・ディレクトリーの絶対パス名		はい
ArchiveProcessed	イベントをアーカイブする方法	true、false アーカイブの振る舞いについて詳しくは、7 ページの『イベントのアーカイブ』を参照してください。	true	はい
DateFormat	ビジネス・オブジェクトの Date 属性のフォーマットを指定します。フォーマットは java.text.SimpleDateFormat クラスに基づいています。	日付記号のすべての有効な組み合わせ (例えば、MM/dd/yyyy HH:mm:ss)。一般的な記号には以下のようなものがあります。 <ul style="list-style-type: none"> • y—年 • M—月 • d—日 • h—時 (午前または午後、12 時間形式) • H—時 (24 時間形式) • m—分 • s—秒 • S—ミリ秒 • a—午前または午後の指定 	MM/dd/yyyy HH:mm:ss	はい
EventFolderURI	イベントが保管されるディレクトリーの完全修飾名	イベント・ディレクトリーの絶対パス名		はい

表 8. Exchange コネクタのアプリケーション固有の構成プロパティ (続き)

プロパティ	説明	指定可能な値	デフォルト値	必須?
InDoubtEvents	コネクタの終了時に状況が IN_PROGRESS であったイベントを処理する方法	<ul style="list-style-type: none"> • FailOnStartup: 致命的エラーの記録および E メール通知の送信を実行し、FAIL を戻します。 • Reprocess: 次のポーリング時に選出されるように、イベント状況を READY_FOR_POLL に変更します。 • LogError: エラーをログに記録します。 • Ignore: イベントを無視します。 	Reprocess	はい
WindowsName	コネクタのユーザー名 (通常は Exchange_User と同じ)。Exchange Server へのログオンに使用されます。	任意の有効なログオン名		はい
WindowsPassword	WindowsName アカウントのパスワード	任意の有効なパスワード		はい
RecipientBusinessObj	Recipient 子ビジネス・オブジェクトの名前を指定します。	任意のテキスト・ストリング	Recipient	いいえ
RecurrencePatternBusinessObj	RecurrencePattern の子ビジネス・オブジェクトの名前を指定します。	任意のテキスト・ストリング	RecurrencePattern	いいえ

イベントの登録

イベントが生成される Exchange フォルダーごとに、イベント登録ファイルを作成する必要があります。イベント登録は、Event Listener にバインドされた隠しファイルに格納されます。Microsoft インフォメーション・ストア・サービスは、始動時にイベント登録ファイルを検索し、モニター対象フォルダーのリストに、指定されたフォルダーを追加します。

単一のフォルダー用のイベント登録 (浅い登録)、親フォルダーとそのすべてのサブフォルダー用のイベント登録 (深い登録)、または組織内のすべてのフォルダー用のイベント登録 (ストア全体の登録) を作成することができます。

イベント登録は、Exchange Explorer の GUI を用いて容易に作成することができます。RegEvent.vbs ユーティリティを使用して作成することもできます。以下のセクションでは、これらのツールを用いたイベント登録の作成方法を説明します。

Exchange Explorer によるイベント登録の作成

Exchange Explorer は、イベント登録作成用の GUI です。このツールは Microsoft の Web サイトからダウンロードすることができます。

ヒント

Microsoft のダウンロード・ページ (www.microsoft.com/downloads) で以下の検索基準を指定することにより、Exchange Explorer を探し出すことができます。

- Exchange 製品の検索を指定し、キーワード SDK を指定します。
- 検索結果が戻されたら、Exchange SDK Tools のカテゴリを選択します。
- ファイル `exchangesdktools.exe` を探し出してダウンロードします。Microsoft の Web サイトが提供する指示に従って、ダウンロードおよびインストールを実行します。

「スタート」 → 「プログラム」 → 「Exchange SDK」 → 「Exchange SDK Development Tools」 → 「Exchange Explorer」 をクリックして、Exchange Explorer を始動します。「認証 (Authorization)」ダイアログ・ボックスで、ログオンのプロンプトが出されたら、以下の値を入力します。

- ユーザー名: イベントを登録するユーザー・アカウントの名前
- パスワード: そのユーザー・アカウントのパスワード
- Exchange ストア URL: `http://machine_name/exchange/user_name/`。ここで、`machine_name` は Exchange Server を実行するマシンの名前を指し、`user_name` は「ユーザー名」フィールドで入力した値を指します。

浅いイベント登録の作成

フォルダーに対して浅いイベント登録を作成するには以下のステップを実行します。

1. Exchange Explorer を始動し、ログオンします。
2. 「Exchange ストア階層 (Exchange Store Hierarchy)」ペインで、登録するフォルダーを選択します。
3. 「ファイル」 → 「イベント登録を追加 (Add Event Registration)」をクリックします。Exchange ストア・イベント・シンク登録ウィザードが開き、イベント登録の名前を選択するプロンプトが表示されます。
4. 「このイベント シンク登録の名前を入力 (Enter a name for this event sink registration)」フィールドに使用する名前を入力し、「次へ」をクリックします。
5. 登録したいイベントの種類を指定するように求められます。「同期」をクリックし、「次へ」をクリックします。
6. 登録したい同期イベントの種類を指定するように求められます。**OnSyncSave** および **OnSyncDelete** の両方をクリックし、「次へ」をクリックします。
7. イベント登録の範囲を指定するように求められます。「浅い (Shallow)」をクリックし、「次へ」をクリックします。
8. 登録を行うイベント・シンクの種類を指定するように求められます。「COM イベント シンク (COM Event Sink)」をクリックし、「次へ」をクリックします。

9. イベント・シンク・クラスを指定するように求められます。「イベント シンク クラスの ProglID (What is the ProglID of your event sink class?)」フィールドで、BIA_Exchange.Events.1 と入力し、「次へ」をクリックします。
10. これまでに選択した項目を確認するように求めるプロンプトが出されます。イベント登録に問題がなければ、「完了」をクリックしてウィザードを終了します。

これでイベント登録が作成されます。登録されたフォルダー内で項目が作成、更新、または削除されるたびに、Event Listener ヘイイベントが送信されます。

注: 登録項目作成時にエラーが発生した場合は、41 ページの『第 5 章 トラブルシューティングとエラー処理』を参照してください。

深いイベント登録の作成

深いイベント登録 (親フォルダーとそのすべてのサブフォルダーが登録されます) を作成するには以下のステップを実行します。

1. Exchange Explorer を始動し、ログオンします。
2. 「Exchange ストア階層 (Exchange Store Hierarchy)」ペインで、登録するフォルダーを選択します。
3. 「ファイル」→「イベント登録を追加 (Add Event Registration)」をクリックします。Exchange ストア・イベント・シンク登録ウィザードが開き、イベント登録の名前を選択するプロンプトが表示されます。
4. 「このイベント シンク登録の名前を入力 (Enter a name for this event sink registration)」フィールドに使用する名前を入力し、「次へ」をクリックします。
5. 登録したいイベントの種類を指定するように求められます。「同期」をクリックし、「次へ」をクリックします。
6. 登録したい同期イベントの種類を指定するように求められます。**OnSyncSave** および **OnSyncDelete** の両方をクリックし、「次へ」をクリックします。
7. イベント登録の範囲を指定するように求められます。「深い (Deep)」をクリックし、「次へ」をクリックします。
8. 登録を行うイベント・シンクの種類を指定するように求められます。「COM イベント シンク (COM Event Sink)」をクリックし、「次へ」をクリックします。
9. イベント・シンク・クラスを指定するように求められます。「イベント シンク クラスの ProglID (What is the ProglID of your event sink class?)」フィールドで、BIA_Exchange.Events.1 と入力し、「次へ」をクリックします。
10. これまでに選択した項目を確認するように求めるプロンプトが出されます。イベント登録に問題がなければ、「完了」をクリックしてウィザードを終了します。

これでイベント登録が作成されます。登録されたフォルダーまたはそのサブフォルダー内で項目が作成、更新、または削除されるたびに、Event Listener ヘイイベントが送信されます。

注: 登録項目作成時にエラーが発生した場合は、41 ページの『第 5 章 トラブルシューティングとエラー処理』を参照してください。

ストア全体のイベント登録の作成

プライベート・メールボックス・ストア向けのストア全体の (グローバルな) イベント登録を行うことができるのは、次のフォルダーのみです。ここで、*domain* はご使用の E メール・ドメインを指し、*GUID* は SystemMailbox のグローバルな一意の ID を指します。file:///backofficestorage/*domain*/mbx/SystemMailbox{*GUID*}/StoreEvents/GlobalEvents

さらに、ストア全体のイベント登録を作成するには、適切なアクセス許可が必要です。グローバル・イベントを登録できるのは、Administrator グループまたは Domain Administrators グループのメンバー、あるいは Exchange Administrators 役割のユーザーのみです。

適切なストア全体のイベント登録を作成するには、26 ページの『浅いイベント登録の作成』または 27 ページの『深いイベント登録の作成』で説明されているステップを実行してください。

RegEvent.vbs によるイベント登録の作成

RegEvent.vbs ユーティリティによって、浅いイベント登録、深いイベント登録、およびストア全体のイベント登録を作成することができます。RegEvent.vbs ユーティリティを用いて浅いイベント登録と深いイベント登録を作成する方法の詳細については、Microsoft Developer Network (MSDN) Library および Knowledge Base (www.msdn.microsoft.com) を参照し、キーワード RegEvent を検索してください。

RegEvent.vbs によるストア全体のイベント登録の作成

プライベート・メールボックス・ストア向けのストア全体の (グローバルな) イベント登録を行うことができるのは、次のフォルダーのみです。ここで、*domain* はご使用の E メール・ドメインを指し、*GUID* は SystemMailbox のグローバルな一意の ID を指します。file:///backofficestorage/*domain*/mbx/SystemMailbox{*GUID*}/StoreEvents/GlobalEvents

さらに、ストア全体のイベント登録を作成するには、適切なアクセス許可が必要です。グローバル・イベントを登録できるのは、Administrator グループまたは Domain Administrators グループのメンバー、あるいは Exchange Administrators 役割のユーザーのみです。

次のコマンドを入力して、ストア全体のイベント登録を作成します。

```
cscript RegEvent.vbs Add "OnSyncSave;OnSyncDelete" ¥  
BIA_Exchange.Events.1 "file:///backofficestorage/domain/mbx ¥  
/SystemMailbox{GUID}StoreEvents/GlobalEvents/globalSynchReg.eml" ¥  
-m ANY
```

アダプターの始動

Adapter for Exchange Server を始動するには、以下の手順に従います。

1. コネクターを始動します。
2. Event Listener を始動します。

コネクターの始動

始動スクリプトおよびメソッド

コネクターは、**コネクター始動スクリプト**を使用して明示的に始動できます。始動スクリプトは、次に示すようなコネクターのランタイム・ディレクトリーに存在していなければなりません。

Windows の場合:

```
ProductDir¥connectors¥connName
```

または

UNIX の場合:

```
ProductDir/connectors/connName
```

ここで *connName* は、コネクターを示し、*ProductDir* は、製品をインストールしたディレクトリーを指します。Windows で実行するコネクターの場合、このファイルは *ProductDir¥connectors¥Exchange¥start_Exchange.bat* です。

UNIX で実行するコネクターの場合、このファイルは

```
ProductDir/connectors/Exchange/start_Exchange.sh
```

 です。

コネクター始動スクリプトは、以下に示すいずれかの方法で起動することができます。

- Windows の場合、「スタート」メニューから

「プログラム」>「IBM WebSphere Business Integration Adapters」>「アダプター」>「コネクター」を選択します。デフォルトでは、プログラム名は「IBM WebSphere Business Integration Adapters」となっています。ただし、これはカスタマイズすることができます。また、ご使用のコネクターへのデスクトップ・ショートカットを作成することもできます。

- Windows の場合、コマンド行から該当するディレクトリーにアクセスします。

```
start_connName connName brokerName [-cconfigFile ]
```

ここで、*connName* はコネクターの名前で、*brokerName* は以下のようにご使用の統合ブローカーを表します。

- WebSphere InterChange Server の場合は、*brokerName* に ICS インスタンスの名前を指定します。
- WebSphere Message Brokers (WebSphere MQ Integrator、WebSphere MQ Integrator Broker、または WebSphere Business Integration Message Broker) または WebSphere Application Server の場合は、*brokerName* にブローカーを示す文字列を指定します。

注: Windows システム上の WebSphere Message Broker または WebSphere Application Server の場合は、*-c* オプションに続いてコネクター構成ファイルの名前を指定しなければなりません。ICS の場合は、*-c* はオプションです。

- UNIX の場合、コマンド行から該当するディレクトリーにアクセスします。

```
connector_manager -start Exchange [-cconfigFile]
```

- Adapter Monitor から (WebSphere Business Integration Adapters 製品のみ)。Adapter Monitor は System Manager 始動時に起動されます。

このツールを使用して、コネクターのロード、アクティブ化、非アクティブ化、休止、シャットダウン、または削除を行うことができます。

- System Monitor から (WebSphere InterChange Server 製品のみ)。

このツールを使用して、コネクターのロード、アクティブ化、非アクティブ化、休止、シャットダウン、または削除を行うことができます。

- Windows の場合のみ、コネクターは Windows サービスとして実行できます。詳細については、「*WebSphere Business Integration Adapters インストール・ガイド*」を参照してください。

コマンド行の始動オプションなどのコネクターの始動方法の詳細については、以下の資料のいずれかを参照してください。

- WebSphere InterChange Server については、「システム管理ガイド」を参照してください。
- WebSphere Message Brokers については、「*WebSphere Message Brokers 使用アダプター・インプリメンテーション・ガイド*」を参照してください。
- WebSphere Application Server については、「アダプター実装ガイド (WebSphere Application Server)」を参照してください。

コネクターの始動時に実行されるタスク

コネクターを始動すると、以下のタスクが実行されます。

- 構成情報の検索
- 検索された構成情報を持つ Exchange Server へのログオン
- サポートされるビジネス・オブジェクト定義の検索
- コネクターがサポートするイベント・サブスクリプションのリストの検索
- コネクター・バージョンのリターン
- ビジネス・オブジェクト・ハンドラーへのポインターのリターン
- 始動時に検索された、サブスクライブされたイベントのビジネス・オブジェクトの作成

Event Listener の始動

デフォルトでは、Event Listener は、トリガー (登録されたイベントが到着) が発生したときに Information Service によって自動的に起動されます。ただし、Event Listener を手動で始動する必要がある場合は、次のようにして Windows コンポーネント・サービス管理ツールを使用します。

1. 「スタート」 → 「プログラム」 → 「管理ツール」 → 「コンポーネント サービス」を選択します。「コンポーネント サービス」ダイアログ・ボックスが開きます。
2. 「コンポーネント サービス」フォルダーおよび「COM+ アプリケーション」フォルダーを展開します。

3. 「COM+ アプリケーション」フォルダー内の BIA_Exchange アプリケーションを右マウス・ボタンでクリックしてコンテキスト・メニューを表示させ、「開始」をクリックします。

アダプターの停止

Adapter for Exchange Server を停止するには、以下の手順に従います。

1. コネクタを停止します。
2. Event Listener を停止します。

コネクタの停止

コネクタを停止する方法は、以下に示すように、コネクタが始動された方法によって異なります。

- コマンド行からコネクタを始動した場合は、コネクタ始動スクリプトを用いて、以下の操作を実行します。
 - 始動スクリプトを起動すると、そのコネクタ用の個別の「コンソール」ウィンドウが作成されます。このウィンドウで、「Q」と入力して Enter キーを押すと、コネクタが停止します。
- Adapter Monitor から (WebSphere Business Integration Adapters 製品のみ)。Adapter Monitor は System Manager 始動時に起動されます。

このツールを使用して、コネクタのロード、アクティブ化、非アクティブ化、休止、シャットダウン、または削除を行うことができます。

- System Monitor から (WebSphere InterChange Server 製品のみ)。

このツールを使用して、コネクタのロード、アクティブ化、非アクティブ化、休止、シャットダウン、または削除を行うことができます。

Event Listener の停止

デフォルトでは、Event Listener は、アイドル状態のまま指定された時間 (デフォルト値は 3 分) が経過すると自動的にシャットダウンされます。このタイムアウト値は変更可能です。手動でシャットダウンされるまで Event Listener を稼働させるように指定することもできます。シャットダウン・オプションを指定するには、以下の手順に従います。

1. Windows コンポーネント・サービス管理ツールを開きます (「スタート」→「プログラム」→「管理ツール」→「コンポーネント サービス」をクリックします)。
2. 「コンポーネント サービス」フォルダーおよび「COM+ アプリケーション」フォルダーを展開します。
3. 「COM+ アプリケーション」フォルダー内の BIA_Exchange アプリケーションを右マウス・ボタンでクリックしてコンテキスト・メニューを表示させ、「プロパティ」をクリックします。
4. 「プロパティ」ダイアログ・ボックスの「詳細設定 (Advanced)」タブに移動します。
5. Event Listener が自動的にシャットダウンされないようにしたい場合は、「アイドル状態の時も実行を続ける」をクリックします。

6. デフォルトと異なるタイムアウト値を指定する場合は、「一定時間アイドルした後でシャットダウンする」フィールドに値 (分単位) を入力します。
7. 「OK」をクリックします。

手動で Event Listener をシャットダウンする場合は、以下の手順に従います。

1. Windows コンポーネント・サービス管理ツールを開きます (「スタート」→「プログラム」→「管理ツール」→「コンポーネント サービス」をクリックします)。
2. 「コンポーネント サービス」フォルダーおよび「COM+ アプリケーション」フォルダーを展開します。
3. 「COM+ アプリケーション」フォルダー内の BIA_Exchange アプリケーションを右マウス・ボタンでクリックしてコンテキスト・メニューを表示させ、「シャットダウン」をクリックします。

1 台のサーバー上に複数のアダプター・インスタンスを作成

コネクターの複数のインスタンスを作成する作業は、いろいろな意味で、カスタム・コネクターの作成と同じです。以下に示すステップを実行することによって、コネクターの複数のインスタンスを作成して実行するように、ご使用のシステムを設定することができます。次のようにする必要があります。

- コネクター・インスタンス用に新規ディレクトリーを作成します。
- 必要なビジネス・オブジェクト定義が設定されていることを確認します。
- 新規コネクター定義ファイルを作成します。
- 新規始動スクリプトを作成します。

新規ディレクトリーの作成

それぞれのコネクター・インスタンスごとにコネクター・ディレクトリーを作成する必要があります。このコネクター・ディレクトリーには、次の名前を付けなければなりません。

Windows の場合:

```
ProductDir¥connectors¥connectorInstance
```

UNIX の場合:

```
ProductDir/connectors/connectorInstance
```

ここで connectorInstance は、コネクター・インスタンスを一意的に示します。コネクターに、コネクター固有のメタオブジェクトがある場合は、コネクター・インスタンスごとにそれらを作成する必要があります。メタオブジェクトをファイルとして保管する場合は、ファイルを connectorInstance ディレクトリーに格納します。

ビジネス・オブジェクト定義の作成

各コネクター・インスタンスのビジネス・オブジェクト定義がプロジェクト内にまだ存在しない場合は、それらを作成する必要があります。

1. 初期コネクターに関連付けられているビジネス・オブジェクト定義を変更する必要がある場合は、適切なファイルをコピーし、**Business Object Designer** を使用してそれらのファイルをインポートします。初期コネクターの任意のファイルをコピーできます。変更を加えた場合は、名前を変更してください。
2. 初期コネクターのファイルは、次のディレクトリーに入っていないとなりません。

Windows の場合:

```
ProductDir¥Exchange¥initialConnectorInstance
```

UNIX の場合:

```
ProductDir/Exchange/initialConnectorInstance
```

作成した追加ファイルは、*ProductDir¥repository* または *ProductDir/repository* の適切な *connectorInstance* サブディレクトリー内に存在している必要があります。

コネクター定義の作成

Connector Configurator 内で、コネクター・インスタンスのコネクター構成ファイル (CFG) を作成します。これを行うには、以下のステップを実行します。

1. 初期コネクターの構成ファイルをコピーし、名前変更します。
2. 各コネクター・インスタンスが、サポートされるビジネス・オブジェクト (および関連メタオブジェクト) を正しくリストしていることを確認します。
3. 必要に応じて、コネクター・プロパティをカスタマイズします。

始動スクリプトの作成

始動スクリプトは以下のように作成します。

1. 初期コネクターの始動スクリプトをコピーし、コネクター・ディレクトリーの名前を含む名前を付けます。

```
dirname
```

2. この始動スクリプトを、32 ページの『新規ディレクトリーの作成』で作成したコネクター・ディレクトリーに格納します。
3. 始動スクリプトのショートカットを作成します (Windows のみ)。
4. 初期コネクターのショートカット・テキストをコピーし、新規コネクター・インスタンスの名前に一致するように (コマンド行で) 初期コネクターの名前を変更します。

これで、ご使用の統合サーバー上でコネクターの両方のインスタンスを同時に実行することができます。

カスタム・コネクター作成の詳細については、「コネクター開発ガイド (C++ 用)」または「コネクター開発ガイド (Java 用)」を参照してください。

第 4 章 ビジネス・オブジェクトの理解

本章では、ビジネス・オブジェクトの構造、アダプターによるビジネス・オブジェクトの処理方法、およびアダプターがそれらを行うにあたっての前提条件について説明します。

本章の内容は、次のとおりです。

- 『メタデータの定義』
- 36 ページの『ビジネス・オブジェクト構造の概要』
- 36 ページの『Exchange Server のビジネス・オブジェクト』

メタデータの定義

Adapter for Microsoft Exchange Server は、メタデータ主導型です。WebSphere Business Integration システムでは、メタデータは、サポート・アプリケーション (この場合は Exchange Server) によりエクスポートされたデータ構造を記述するアプリケーション固有情報として定義されています。メタデータを使用してビジネス・オブジェクト定義が作成されます。このビジネス・オブジェクト定義は、実行時にコネクタがビジネス・オブジェクトを作成するときに使用されます。

メタデータ主導型アダプターは、サポートする各ビジネス・オブジェクトを、ビジネス・オブジェクト定義にエンコードされているメタデータに基づいて処理します。これによりアダプターは、基本となるコネクタ・コードの変更を要求することなく、新規または変更されたビジネス・オブジェクト定義を処理できます。

ビジネス・オブジェクト定義に記述されたアプリケーション固有のメタデータには、ビジネス・オブジェクトの構造とその属性プロパティの設定についての情報が含まれています。実行時には、インスタンス化されたビジネス・オブジェクトが実際のデータ値を持ちます。

アダプターには、サポートするビジネス・オブジェクトの構造、親と子のビジネス・オブジェクト間の関係、およびデータのフォーマットに関する前提事項があります。そのため、ビジネス・オブジェクトの構造が、Exchange Server 内の対応する項目用に定義された構造と正確に一致することが重要となります。構造が一致しないと、アダプターはビジネス・オブジェクトを正しく処理できません。

注: 項目を変更する場合 (例えば、フィールドを追加)、必ず対応するビジネス・オブジェクトを変更して、変更内容を反映する必要があります。このためには、Business Object Designer を使用します。この種の変更を行うための詳細については、「ビジネス・オブジェクト開発ガイド」を参照してください。

ビジネス・オブジェクト構造の概要

WebSphere Business Integration システムでは、ビジネス・オブジェクト定義は以下のものから構成されます。

- タイプ名
- サポートされる動詞
- 属性

実行時に、ビジネス・オブジェクト定義は特定のビジネス・オブジェクトとしてインスタンス化されます。このインスタンスはアプリケーションのビジネス・エンティティの特定の内容 (例えば、タスク・エントリー) を表します。

属性にはデータ自体が含まれることがあります (これらは単純属性と呼ばれます)、または子ビジネス・オブジェクトまたは関連データを含む子ビジネス・オブジェクトの配列を指すことがあります。単純属性のみを含むビジネス・オブジェクトは、フラット・ビジネス・オブジェクトと呼ばれます。子ビジネス・オブジェクトまたは子ビジネス・オブジェクトの配列を指す属性と単純属性を含むビジネス・オブジェクトは、階層ビジネス・オブジェクトと呼ばれます。階層ビジネス・オブジェクトでは、キー値で親ビジネス・オブジェクトとすべての子ビジネス・オブジェクトと接続します。

カーディナリティー 1 コンテナ・オブジェクト、または単一カーディナリティー関係は、親ビジネス・オブジェクトの属性が単一の子ビジネス・オブジェクトを含む場合に生じます。この場合、子ビジネス・オブジェクトは単一のレコードのみを含むことができるコレクションを表します。属性タイプは子ビジネス・オブジェクトのものと同一です。

カーディナリティー n コンテナ・オブジェクト、または複数カーディナリティー関係は、親ビジネス・オブジェクトの属性に子ビジネス・オブジェクトの配列が含まれる場合に生じます。この場合、子ビジネス・オブジェクトは複数のレコードを含むことができるコレクションを表します。属性タイプは、子ビジネス・オブジェクトの配列の属性タイプと同じになります。

Exchange Server のビジネス・オブジェクト

Adapter for Exchange Server は、Exchange Outlook Message オブジェクト・モデルに基づいてビジネス・オブジェクトをサポートします。サポートされるビジネス・オブジェクトには、以下の高水準オブジェクトが含まれます。

- **AppointmentItem**: 一回限りまたは繰り返し発生する「予定表」フォルダー内の予定。
- **ContactItem**: 「連絡先」フォルダー内のユーザー。
- **MailItem**: メールボックス・フォルダー (「受信トレイ」など) 内のメール・メッセージ。MailItem オブジェクトは、デフォルトのビジネス・オブジェクトです。
- **TaskItem**: 「タスク」フォルダー内のタスク。タスクは、割り当て、委任、または自主規制が可能であり、特定の時間フレーム内で実行する必要があります。

さらに、共通属性データを表す次の 2 つのオブジェクトがあります。

- **Recipient**: メッセージの宛先あるいは送信先となる単一アドレス。

- **RecurrencePattern: AppointmentItem** ビジネス・オブジェクトまたは **TaskItem** ビジネス・オブジェクトの繰り返しパターン。

注: ビジネス・オブジェクト定義は、新しいリリースで更新されている可能性があります。本書の記述時点では、**TaskItem RecurrenceItems** はコネクターにアクセスできません。ご使用のリリースに該当するかどうかまたはその他の変更があるかどうかの確認については、「リリース情報」を参照してください。このアダプターの以前のバージョンを実行している場合は、更新されたビジネス・オブジェクト定義をすべて最新にする必要があります。そのためには、該当するファイルを **BO Designer** にロードして (ディレクトリー・パスについては、13 ページの『コネクターのインストール』を参照)、ユーザーのプロローカー・リポジトリーに保存します。

このアダプターの現行リリースでは、**Attachment** ビジネス・オブジェクトをサポートしていません。

Exchange 項目に子がない場合、単一アダプターのビジネス・オブジェクトが **Exchange** ビジネス・エンティティー全体を表します。例えば、繰り返しパターンを持たない予定は、**AppointmentItem** ビジネス・オブジェクトで表します。

Exchange 項目が階層構造である場合、アダプターのビジネス・オブジェクトは親 **ビジネス・エンティティー** を表し、その属性は子情報を持つ子オブジェクトを参照します。例えば、繰り返し発生する予定は、**RecurrencePattern** 子 **ビジネス・オブジェクト** を指す属性を持つ親 **AppointmentItem** ビジネス・オブジェクトで表します。

属性プロパティー

ビジネス・オブジェクト属性を定義するプロセスには、さまざまなプロパティーの設定を含みます。表 9 に、単純属性のプロパティーを示します。

表 9. 単純属性のプロパティー

属性	説明
Name	属性の名前を指定します。
Type	属性のデータ型 (Boolean 、 Date 、 Integer 、または String) を指定します。
MaxLength	N/A
IsKey	属性がキー・フィールドであるかどうかを指定します。
IsForeignKey	N/A
IsRequired	属性がビジネス・オブジェクトの必要フィールドであるかどうかを指定します。
AppSpecificInfo	この属性のアクセスに使用する URI を指定します。
DefaultValue	指定されていると、インバウンド・ビジネス・オブジェクトに属性が設定されていない場合、およびコネクターの UseDefaults プロパティーの設定が true の場合にコネクターがこの値を使用します。

表 10 に、子オブジェクトを参照する親ビジネス・オブジェクト属性のプロパティを示します。

表 10. 子ビジネス・オブジェクトを参照する属性のプロパティ

属性	説明
Name	子ビジネス・オブジェクトの名前を指定します。
Type	子ビジネス・オブジェクトのデータ型を指定します。有効なデータ型は以下のとおりです。 <ul style="list-style-type: none"> • Recipient • RecurrencePattern
ContainedObjectVersion	子ビジネス・オブジェクトのバージョンを指定します。
Relationship	参照オブジェクトは親オブジェクトに含まれていることを指定します。
IsKey	N/A
IsForeignKey	N/A
IsRequired	属性がビジネス・オブジェクトに必要であるかどうかを指定します。
AppSpecificInfo	N/A
Cardinality	特定の属性から参照可能な子オブジェクトの数を指定します。値は 1 (1 つのオブジェクトのみ参照可能な場合) または N (複数のオブジェクトが参照可能な場合) です。

アプリケーション固有の情報

アプリケーション固有の情報により、ビジネス・オブジェクトの操作および処理に関する、アプリケーション依存の情報がコネクタに提供されます。ビジネス・オブジェクト定義を拡張または変更する場合には、定義内のアプリケーション固有の情報と、コネクタが予期する構文とを必ず一致させる必要があります。

アプリケーション固有の情報はビジネス・オブジェクトに指定でき、また各ビジネス・オブジェクトの属性にも指定できます。

ビジネス・オブジェクト用のアプリケーション固有テキスト

ビジネス・オブジェクトのアプリケーション固有の情報では、アダプター用の 2 つのキー・プロパティ (OutlookMessageClass および ContentClass) を扱います。OutlookMessageClass は、現行の Exchange Adapter ビジネス・オブジェクトに対応する Exchange 項目の名前を指定します。ContentClass は、現行の Exchange Adapter ビジネス・オブジェクトに対応する WebDAV ネーム・スペースを指定します。以下の表は、該当する相関関係を示します。

表 11. ビジネス・オブジェクトのアプリケーション固有のテキスト

アダプター・ ビジネス・ オブジェクト	OutlookMessageClass	ContentClass
AppointmentItem	IPM:Appointment	urn:content-classes:appointment
ContactItem	IPM:Contact	urn:content-classes:person
MailItem	IPM:Note	urn:content-classes:message
TaskItem	IPM:Task	urn:content-classes:task

単純属性用のアプリケーション固有テキスト

表 12 に示すように、コネクタは単純属性用のアプリケーション固有テキストを使用します。

表 12. 単純属性用のアプリケーション固有テキスト

パラメーター	説明
PropertyName	この属性に対応する URI を指定します。
Access	現在の属性が読み取り/書き込み (Get/Put) であるか、読み取り専用 (Get) であるかを指定します。

動詞に関するアプリケーション固有テキスト

表 13 に示すように、コネクタは動詞用のアプリケーション固有テキストを使用します。

表 13. 動詞に関するアプリケーション固有テキスト

パラメーター	説明
SetToNullOnCxIgnore	サービス呼び出し要求の処理中に CxIgnore や CxBlank が見つかった場合の動作を定義します。この値は、true または false です。true の場合に CxIgnore または CxBlank が見つかり、値は null にセットされます。false の場合に CxIgnore が見つかり、値は変更されません。false の場合に CxBlank が見つかり、値は null にセットされます。

第 5 章 トラブルシューティングとエラー処理

本章では、アダプターによるエラー処理について説明します。本章の内容は、次のとおりです。

- 『コネクターのエラー処理』
- 42 ページの『Event Listener のエラー処理』
- 43 ページの『トレース・メッセージ』

コネクターのエラー処理

コネクターは、トレース・レベルに関係なく、処理中に発生した異常状態をすべてログに記録します。コネクターは、エラー・テキストをコネクター・ログ・ファイルに書き込みます。このファイルの名前およびロケーションは、LogFileName コネクター構成プロパティを使用して設定します。

メッセージには、状態および結果の詳細記述が含まれます。また、ビジネス・オブジェクト・ダンプやスタック・トレースなどのデバッグ時に補助となる追加情報が含まれる場合もあります (特例)。

エラー・メッセージの完全なリストについては、*ProductDir¥connectors¥messages* ディレクトリーにインストールされている *BIA_ExchangeConnector.txt* ファイルを参照してください。

表 14 に、よく見られるエラーのいくつかを示し、コネクターがこれらのエラーを処理する方法を説明します。

表 14. コネクターのエラー

エラーの説明	エラー・タイプ	エラー処理
HTTP 認証中に LOGON_FAILED が戻されました。ユーザー名またはユーザー・パスワード (あるいは両方) が無効であるか、サーバーが停止しています。	致命的エラー	初期化時にコネクターがエラーを検出します。致命的エラーをログに記録し、終了します。
コネクターは、ビジネス・オブジェクト・サービス呼び出し要求の処理時に、接続エラーを検出します。	致命的エラー	コネクターは、致命的エラーをログに記録し、戻りコード APPRESPONSE_TIMEOUT を送信して E メール通知を起動してから終了します。
コネクターは、アンサブスクライブされたイベントの検索を試みています。	警告	コネクターは、警告をログに記録し、状況を UNSUBSCRIBED としてアーカイブ・ストアにイベントを保管します。

表 14. コネクターのエラー (続き)

エラーの説明	エラー・タイプ	エラー処理
コネクターが統合ブローカーへビジネス・オブジェクトのパブリッシュを試みたときに、gotApplEvent() メソッドで問題が発生しています。	エラー	コネクターはエラーをログに記録します。状況を ERROR_POSTING_EVENT としてアーカイブ・ストアにイベントが保管されます。
サービス呼び出し要求時に、メッセージに対する Create、Update、または Delete 操作が失敗しています。	エラー	コネクターはエラーをログに記録し、呼び出し側コラボレーションまたはフローに CWConnectorConstants.FAIL を戻します。
フォルダー、項目、またはフィールドが存在しないため、イベント処理時に Exchange Server エラー ITEM_NOT_FOUND が戻されています。	エラー	コネクターはエラーをログに記録します。状況を ERROR_OBJECT_NOT_FOUND としてアーカイブ・ストアにイベントが保管されます。
コネクターがアクセス許可を持たないオブジェクトを表示または更新しようとしたときに、Exchange Server エラー E_ACCESSDENIED が戻されています。	エラー	コネクターはエラーをログに記録します。状況を ERROR_PROCESSING_EVENT としてアーカイブ・ストアにイベントが保管されます。
メールボックス割り当て量を超過したため、項目を更新できないときに、Exchange Server エラー E_FAIL が戻されます。	エラー	コネクターはエラーをログに記録します。状況を ERROR_PROCESSING_EVENT としてアーカイブ・ストアにイベントが保管されます。

Event Listener のエラー処理

Event Listener は、トレース・レベルにかかわらず、処理中に発生した異常条件をすべてログに記録します。Windows イベント・ログに Error イベントとしてエラー・テキストを書き込みます。

表 15 に、Event Listener で最も多く発生するエラーを示します。

表 15. Event Listener エラー

エラーの説明	エラー・タイプ	エラー処理
Event Listener は、構成ファイル (BIA_Exchange.ini) を開くことができません。	警告	Event Listener は、構成ファイルで指定された値ではなく、デフォルト値を使用します。
CwEvent ユーザーが存在しないか、またはメールボックスに空きがないため、Eメールを CwEvent ユーザーに送信できません。	エラー	すべての既知の情報とともにエラーがログに記録されます。

イベント登録エラー

Exchange Explorer を使用してイベント登録を作成する際に、以下のエラーを受け取る可能性があります。

An error has occurred creating the registration item.
Event sinks must be registered as a COM+ Application prior to creating the registration event item, or there may be issues connecting to the network or server. Also check the data for accuracy. Choose Yes to retry your request, No to abort, or Cancel to check your data and try again.

イベント登録のデータが正確であり、ネットワークとサーバーが機能していることを確認してください。それでもこのエラーが発生する場合は、19 ページの『COM+ アプリケーションへの Event Listener の追加』に示すように、イベント・シンクを COM+ アプリケーションとして登録するようにしてください。

トレース・メッセージ

トレースは、コネクタまたは Event Listener の振る舞いを詳細にトレースするためにオンにできるオプションのデバッグ機能です。トレース・メッセージは構成可能であり、動的変更が可能です。要求の詳細に応じてさまざまなレベルを設定します。以下のセクションで、Exchange アダプターの各コンポーネントのトレースについて説明します。

推奨: トレースは、パフォーマンスを向上させ、ファイル・サイズを小さくするために、実動システム上ではオフにするか、できるだけ低レベルに設定しておいてください。

コネクタでのトレースの使用

トレース・メッセージは、デフォルトでは「STDOUT」(画面) に書き込まれます。また、トレースをファイルに書き込むように構成することもできます。

表 16 に、各トレース・レベルでコネクタが出力する各種のトレース・メッセージを示します。すべてのトレース・メッセージがコネクタ・プロパティ `TraceFileName` によって指定されたファイルに表示されます。これらのメッセージは、IBM WebSphere Business Integration Adapter アーキテクチャーによって出力されるトレース・メッセージに追加されます。

表 16. コネクタのトレース・メッセージ

トレース・レベル	トレース・メッセージ
レベル 0	コネクタのバージョンを識別するメッセージ。このレベルでは他のトレースは行われません。これはデフォルトのトレース・レベルです。
レベル 1	トレース・メッセージには以下のようなものがあります。 <ul style="list-style-type: none">• 状況メッセージ。• 処理される各ビジネス・オブジェクトの識別 (キー) 情報を提供するメッセージ。
レベル 2	ビジネス・オブジェクト処理時にコネクタが検出または検索した配列および子ビジネス・オブジェクトに関する情報を含む、ビジネス・オブジェクト・ハンドラー・メッセージ。

表 16. コネクターのトレース・メッセージ (続き)

トレース・レベル	トレース・メッセージ
レベル 3	ビジネス・オブジェクト処理に関する情報を提供するメッセージ。例えば、コネクターがビジネス・オブジェクト間で一致するものを検出した場合、またはビジネス・オブジェクトの配列中で特定のビジネス・オブジェクトを検出した場合に、このタイプのメッセージがデリバリーされます。
レベル 4	トレース・メッセージには以下のようなものがあります。 <ul style="list-style-type: none"> アプリケーション固有のテキスト・メッセージ (例えば、ビジネス・オブジェクトのアプリケーション固有のテキスト・フィールドを構文解析する関数によって戻される値を示すメッセージ)。 コネクターが、コネクターのプロセス・フローのトレースに役立つ関数を開始または終了したかどうかを識別するメッセージ。
レベル 5	トレース・メッセージには以下のようなものがあります。 <ul style="list-style-type: none"> コネクターの初期化を示すメッセージ (例えば、検索された各構成プロパティの値を示すメッセージ)。 コネクターに処理される前のビジネス・オブジェクトの表現 (すなわち、コネクターがコラボレーションから受け取ったときのビジネス・オブジェクトの状態を表示します) およびコネクターによる処理が完了した後のビジネス・オブジェクトの表現 (すなわち、コネクターがコラボレーションに戻すときのビジネス・オブジェクトの状態を表示します) を構成するメッセージ。 処理時のビジネス・オブジェクト・ダンプを構成するメッセージ。

Event Listener でのトレースの使用

デフォルトでは、トレース・メッセージは Information イベントとして Windows イベント・ログに書き込まれます。表 17 では、Event Listener が各トレース・レベルで出力するトレース・メッセージのタイプを説明しています。

表 17. Event Listener のトレース・メッセージ

トレース・レベル	トレース・メッセージ
レベル 0	Event Listener 識別メッセージ (例えば、2002/07/10 15:01:46.812: This is version 1.0 of the Exchange adapter)。このメッセージは、最初にイベント・シンクが作成されたときに必ず表示されます
レベル 1	トレース・メッセージには以下のようなものがあります。 <ul style="list-style-type: none"> 状況メッセージ。 処理される各イベントの識別 (キー) 情報を提供するメッセージ。 onSave()、onDelete()、onSyncSave()、onSyncDelete()、onTimer()、onMDBStartUp()、および onMDBShutdown() メソッドを実行するたびに発行されるメッセージ。
レベル 2	E メール・メッセージが CwEvent ユーザーに送信されるたびに発行されるメッセージ。
レベル 3	イベント固有のフラグ (例えば、EVT_NEW_ITEM、EVT_COPY、EVT_MOVE など) を表示するメッセージ。

表 17. *Event Listener* のトレース・メッセージ (続き)

トレース・レベル	トレース・メッセージ
レベル 4	イベント・ストアのフィールド値を示すメッセージ。フィールドには、EventID、Event type、FolderURI、MessageURI、およびPriority があります。
レベル 5	<p>トレース・メッセージには以下のようなものがあります。</p> <ul style="list-style-type: none"> • 初期化メッセージ (例えば、検索された各構成プロパティの値を示すメッセージ)。 • 個々のメソッドの入り口点と出口点をトレースするメッセージ。

付録 A. コネクターの標準構成プロパティ

この付録では、WebSphere Business Integration Adapter のコネクタ・コンポーネントの標準構成プロパティについて説明します。この付録の内容は、以下の統合ブローカーで実行されるコネクタを対象としています。

- WebSphere InterChange Server (ICS)
- WebSphere MQ Integrator、WebSphere MQ Integrator Broker、および WebSphere Business Integration Message Broker (WebSphere Message Brokers (WMQI) と総称)
- WebSphere Application Server (WAS)

コネクタによっては、一部の標準プロパティが使用されないことがあります。Connector Configurator から統合ブローカーを選択するときには、そのブローカーで実行されるアダプターについて構成する必要のある標準プロパティのリストが表示されます。

コネクタ固有のプロパティの詳細については、該当するアダプターのユーザーズ・ガイドを参照してください。

注: 本書では、ディレクトリー・パスの規則として円記号 (¥) を使用します。UNIX システムを使用している場合は、円記号をスラッシュ (/) に置き換えてください。また、各オペレーティング・システムの規則に従ってください。

新規プロパティと削除されたプロパティ

以下の標準プロパティは、本リリースで追加されました。

新規プロパティ

- XMLNamespaceFormat

削除されたプロパティ

- RestartCount

標準コネクタ・プロパティの構成

アダプター・コネクタには 2 つのタイプの構成プロパティがあります。

- 標準構成プロパティ
- コネクタ固有のプロパティ

このセクションでは、標準構成プロパティについて説明します。コネクタ固有の構成プロパティについては、該当するアダプターのユーザーズ・ガイドを参照してください。

Connector Configurator の使用

Connector Configurator からコネクタ・プロパティを構成します。Connector Configurator には、System Manager からアクセスします。Connector Configurator の使用法の詳細については、付録『Connector Configurator』を参照してください。

注: Connector Configurator と System Manager は、Windows システム上でのみ動作します。コネクタを UNIX システム上で稼働している場合でも、これらのツールがインストールされた Windows マシンが必要です。UNIX 上で動作するコネクタのコネクタ・プロパティを設定する場合は、Windows マシン上で System Manager を起動し、UNIX の統合ブローカーに接続してから、コネクタ一用の Connector Configurator を開く必要があります。

プロパティ値の設定と更新

プロパティ・フィールドのデフォルトの長さは 255 文字です。

コネクタは、以下の順序に従ってプロパティの値を決定します (最も番号の大きい項目が他の項目よりも優先されます)。

1. デフォルト
2. リポジトリ (WebSphere InterChange Server が統合ブローカーである場合のみ)
3. ローカル構成ファイル
4. コマンド行

コネクタは、始動時に構成値を取得します。実行時セッション中に 1 つ以上のコネクタ・プロパティの値を変更する場合は、プロパティの**更新メソッド**によって、変更を有効にする方法が決定されます。標準コネクタ・プロパティには、以下の 4 種類の更新メソッドがあります。

• 動的

変更を System Manager に保管すると、変更が即時に有効になります。コネクタが System Manager から独立してスタンドアロン・モードで稼働している場合 (例えば、いずれかの WebSphere Message Brokers と連携している場合) は、構成ファイルでのみプロパティを変更できます。この場合、動的更新は実行できません。

• エージェント再始動 (ICS のみ)

アプリケーション固有のコンポーネントを停止して再始動しなければ、変更が有効になりません。

• コンポーネント再始動

System Manager でコネクタを停止してから再始動しなければ、変更が有効になりません。アプリケーション固有コンポーネントまたは統合ブローカーを停止、再始動する必要はありません。

• サーバー再始動

アプリケーション固有のコンポーネントおよび統合ブローカーを停止して再始動しなければ、変更が有効になりません。

特定のプロパティの更新方法を確認するには、「Connector Configurator」ウィンドウ内の「更新メソッド」列を参照するか、次に示す 49 ページの表 18 の「更新メソッド」列を参照してください。

標準プロパティの要約

表 18 は、標準コネクタ構成プロパティの早見表です。標準プロパティの依存関係は RepositoryDirectory に基づいているため、コネクタによっては使用されないプロパティがあり、使用する統合ブローカーによってプロパティの設定が異なる可能性があります。

コネクタを実行する前に、これらのプロパティの一部の値を設定する必要があります。各プロパティの詳細については、次のセクションを参照してください。

注: 表 18 の「注」列にある「Repository Directory は REMOTE」という句は、ブローカーが InterChange Server であることを示します。ブローカーが WMQI または WAS の場合には、リポジトリ・ディレクトリは LOCAL に設定されます。

表 18. 標準構成プロパティの要約

プロパティ名	指定可能な値	デフォルト値	更新メソッド	注
AdminInQueue	有効な JMS キュー名	CONNECTORNAME /ADMININQUEUE	コンポーネント再始動	Delivery Transport は JMS
AdminOutQueue	有効な JMS キュー名	CONNECTORNAME/ADMINOUTQUEUE	コンポーネント再始動	Delivery Transport は JMS
AgentConnections	1 から 4	1	コンポーネント再始動	Delivery Transport は MQ および IDL: Repository Directory は <REMOTE> (ブローカーは ICS)
AgentTraceLevel	0 から 5	0	動的	
ApplicationName	アプリケーション名	コネクタ・アプリケーション名として指定された値	コンポーネント再始動	
BrokerType	ICS、WMQI、WAS		コンポーネント再始動	
CharacterEncoding	ascii7、ascii8、SJIS、Cp949、GBK、Big5、Cp297、Cp273、Cp280、Cp284、Cp037、Cp437 注: これは、サポートされる値の一部です。	ascii7	コンポーネント再始動	
ConcurrentEventTriggeredFlows	1 から 32,767	1	コンポーネント再始動	Repository Directory は <REMOTE> (ブローカーは ICS)
ContainerManagedEvents	値なしまたは JMS	値なし	コンポーネント再始動	Delivery Transport は JMS

表 18. 標準構成プロパティの要約 (続き)

プロパティ名	指定可能な値	デフォルト値	更新 メソッド	注
ControllerStoreAndForwardMode	true または false	true	動的	Repository Directory は <REMOTE> (ブローカーは ICS)
ControllerTraceLevel	0 から 5	0	動的	Repository Directory は <REMOTE> (ブローカーは ICS)
DeliveryQueue		CONNECTORNAME/DELIVERYQUEUE	コンポーネン ト再始動	JMS トランスポー トのみ
DeliveryTransport	MQ、IDL、または JMS	JMS	コンポーネン ト再始動	Repository Directory が ローカルの 場合は、値は JMS のみ
DuplicateEventElimination	true または false	false	コンポーネン ト再始動	JMS トランスポー トのみ: Container Managed Events は <NONE> でなけれ ばならない
FaultQueue		CONNECTORNAME/FAULTQUEUE	コンポーネン ト再始動	JMS トランスポー トのみ
jms.FactoryClassName	CxCommon.Messaging.jms .IBMMQSeriesFactory ま たは CxCommon.Messaging .jms.SonicMQFactory または任意の Java クラス 名	CxCommon.Messaging. jms.IBMMQSeriesFactory	コンポーネン ト再始動	JMS トランスポー トのみ
jms.MessageBrokerName	FactoryClassName が IBM の場合は crossworlds.queue. manager を使用。 FactoryClassName が Sonic の場合 localhost:2506 を使用。	crossworlds.queue.manager	コンポーネン ト再始動	JMS トランスポー トのみ
jms.NumConcurrentRequests	正整数	10	コンポーネン ト再始動	JMS トランスポー トのみ
jms.Password	任意の有効なパスワード		コンポーネン ト再始動	JMS トランスポー トのみ
jms.UserName	任意の有効な名前		コンポーネン ト再始動	JMS トランスポー トのみ
JvmMaxHeapSize	ヒープ・サイズ (メガバイ ト単位)	128m	コンポーネン ト再始動	Repository Directory は <REMOTE> (ブローカーは ICS)

表 18. 標準構成プロパティの要約 (続き)

プロパティ名	指定可能な値	デフォルト値	更新 メソッド	注
JvmMaxNativeStackSize	スタックのサイズ (キロバイト単位)	128k	コンポーネント再始動	Repository Directory は <REMOTE> (ブローカーは ICS)
JvmMinHeapSize	ヒープ・サイズ (メガバイト単位)	1m	コンポーネント再始動	Repository Directory は <REMOTE> (ブローカーは ICS)
ListenerConcurrency	1 から 100	1	コンポーネント再始動	Delivery Transport は MQ でなければならぬ
Locale	en_US、ja_JP、ko_KR、zh_CN、zh_TW、fr_FR、de_DE、it_IT、es_ES、pt_BR 注: これは、サポートされるロケールの一部です。	en_US	コンポーネント再始動	
LogAtInterchangeEnd	true または false	false	コンポーネント再始動	Repository Directory は <REMOTE> でなければならぬ (ブローカーは ICS)
MaxEventCapacity	1 から 2147483647	2147483647	動的	Repository Directory は <REMOTE> でなければならぬ (ブローカーは ICS)
MessageFileName	パスまたはファイル名	CONNECTORNAMEConnector.txt	コンポーネント再始動	
MonitorQueue	任意の有効なキュー名	CONNECTORNAME/MONITORQUEUE	コンポーネント再始動	JMS トランスポートのみ: DuplicateEvent Elimination は true でなければならぬ
OADAutoRestartAgent	true または false	false	動的	Repository Directory は <REMOTE> でなければならぬ (ブローカーは ICS)

表 18. 標準構成プロパティの要約 (続き)

プロパティ名	指定可能な値	デフォルト値	更新メソッド	注
OADMaxNumRetry	正数	1000	動的	Repository Directory は <REMOTE> でなければなりません (ブローカーは ICS)
OADRetryTimeInterval	正数 (単位: 分)	10	動的	Repository Directory は <REMOTE> でなければなりません (ブローカーは ICS)
PollEndTime	HH:MM	HH:MM	コンポーネント再始動	
PollFrequency	正整数 (単位: ミリ秒) no (ポーリングを使用不可にする) key (コネクタのコマンド・プロンプト・ウィンドウで文字 p が入力された場合にのみポーリングする)	10000	動的	
PollQuantity	1 から 500	1	エージェント再始動	JMS トランスポートのみ: Container Managed Events を指定
PollStartTime	HH:MM(HH は 0 から 23、MM は 0 から 59)	HH:MM	コンポーネント再始動	
RepositoryDirectory	メタデータ・リポジトリの場所		エージェント再始動	ICS の場合は <REMOTE> に設定する。 WebSphere MQ Message Brokers および WAS の場合: C:\crossworlds¥repository に設定する
RequestQueue	有効な JMS キュー名	CONNECTORNAME/REQUESTQUEUE	コンポーネント再始動	Delivery Transport は JMS

表 18. 標準構成プロパティの要約 (続き)

プロパティ名	指定可能な値	デフォルト値	更新 メソッド	注
ResponseQueue	有効な JMS キュー名	CONNECTORNAME/RESPONSEQUEUE	コンポーネント再始動	Delivery Transport が JMS の場合: Repository Directory が <REMOTE> の場合のみ必要
RestartRetryCount	0 から 99	3	動的	
RestartRetryInterval	適切な正数 (単位: 分): 1 から 2147483547	1	動的	
RHF2MessageDomain	mrm、xml	mrm	コンポーネント再始動	Delivery Transport が JMS であり、かつ WireFormat が CwXML である
SourceQueue	有効な WebSphere MQ 名	CONNECTORNAME/SOURCEQUEUE	エージェント再始動	Delivery Transport が JMS であり、かつ Container Managed Events が指定されている場合のみ
SynchronousRequestQueue		CONNECTORNAME/ SYNCHRONOUSREQUESTQUEUE	コンポーネント再始動	Delivery Transport は JMS
SynchronousRequestTimeout	0 以上の任意の数値 (ミリ秒)	0	コンポーネント再始動	Delivery Transport は JMS
SynchronousResponseQueue		CONNECTORNAME/ SYNCHRONOUSRESPONSEQUEUE	コンポーネント再始動	Delivery Transport は JMS
WireFormat	CwXML、CwBO	CwXML	エージェント再始動	Repository Directory が <REMOTE> でない場合は CwXML。Repository Directory が <REMOTE> であれば CwBO
WsifSynchronousRequestTimeout	0 以上の任意の数値 (ミリ秒)	0	コンポーネント再始動	WAS のみ
XMLNameSpaceFormat	short、long	short	エージェント再始動	WebSphere MQ Message Brokers および WAS のみ

標準構成プロパティ

このセクションでは、各標準コネクタ構成プロパティの定義を示します。

AdminInQueue

統合ブローカーからコネクタへ管理メッセージが送信される時に使用されるキューです。

デフォルト値は `CONNECTORNAME/ADMININQUEUE` です。

AdminOutQueue

コネクタから統合ブローカーへ管理メッセージが送信される時に使用されるキューです。

デフォルト値は `CONNECTORNAME/ADMINOUTQUEUE` です。

AgentConnections

`RepositoryDirectory` が `<REMOTE>` の場合のみ適用可能です。

`AgentConnections` プロパティは、`orb.init[]` により開かれる ORB (オブジェクト・リクエスト・ブローカー) 接続の数を制御します。

このプロパティのデフォルト値は 1 に設定されます。必要に応じてこの値を変更できます。

AgentTraceLevel

アプリケーション固有のコンポーネントのトレース・メッセージのレベルです。デフォルト値は 0 です。コネクタは、設定されたトレース・レベル以下の該当するトレース・メッセージをすべてデリバリーします。

ApplicationName

コネクタのアプリケーションを一意的に特定する名前です。この名前は、システム管理者が WebSphere Business Integration システム環境をモニターするために使用されます。コネクタを実行する前に、このプロパティに値を指定する必要があります。

BrokerType

使用する統合ブローカー・タイプを指定します。オプションは ICS、WebSphere Message Brokers (WMQI、WMQIB または WBIMB) または WAS です。

CharacterEncoding

文字 (アルファベットの文字、数値表現、句読記号など) から数値へのマッピングに使用する文字コード・セットを指定します。

注: Java ベースのコネクタでは、このプロパティは使用しません。C++ ベースのコネクタでは、現在、このプロパティに `ascii7` という値が使用されています。

デフォルトでは、ドロップダウン・リストには、サポートされる文字エンコードの一部のみが表示されます。ドロップダウン・リストに、サポートされる他の値を追

加するには、製品ディレクトリーにある `¥Data¥Std¥stdConnProps.xml` ファイルを手動で変更する必要があります。詳細については、Connector Configurator に関する章を参照してください。

ConcurrentEventTriggeredFlows

`RepositoryDirectory` が `<REMOTE>` の場合のみ適用可能です。

コネクターがイベントのデリバリー時に並行処理できるビジネス・オブジェクトの数を決定します。この属性の値を、並行してマップおよびデリバリーできるビジネス・オブジェクトの数に設定します。例えば、この属性の値を 5 に設定すると、5 個のビジネス・オブジェクトが並行して処理されます。デフォルト値は 1 です。

このプロパティーを 1 よりも大きい値に設定すると、ソース・アプリケーションのコネクターが、複数のイベント・ビジネス・オブジェクトを同時にマップして、複数のコラボレーション・インスタンスにそれらのビジネス・オブジェクトを同時にデリバリーすることができます。これにより、統合ブローカーへのビジネス・オブジェクトのデリバリーにかかる時間、特にビジネス・オブジェクトが複雑なマップを使用している場合のデリバリー時間が短縮されます。ビジネス・オブジェクトのコラボレーションに到達する速度を増大させると、システム全体のパフォーマンスを向上させることができます。

ソース・アプリケーションから宛先アプリケーションまでのフロー全体に並行処理を実装するには、次のようにする必要があります。

- `Maximum number of concurrent events` プロパティーの値を増加して、コラボレーションが複数のスレッドを使用できるように構成します。
- 宛先アプリケーションのアプリケーション固有コンポーネントが複数の要求を並行して実行できることを確認します。つまり、このコンポーネントがマルチスレッド化されているか、またはコネクター・エージェント並列処理を使用でき、複数プロセスに対応するよう構成されている必要があります。`Parallel Process Degree` 構成プロパティーに、1 より大きい値を設定します。

`ConcurrentEventTriggeredFlows` プロパティーは、順次に行われる単一スレッド処理であるコネクターのポーリングでは無効です。

ContainerManagedEvents

このプロパティーにより、JMS イベント・ストアを使用する JMS 対応コネクターが、保証付きイベント・デリバリーを提供できるようになります。保証付きイベント・デリバリーでは、イベントはソース・キューから除去され、単一 JMS トランザクションとして宛先キューに配置されます。

デフォルト値はありません。

`ContainerManagedEvents` を JMS に設定した場合には、保証付きイベント・デリバリーを使用できるように次のプロパティーも構成する必要があります。

- `PollQuantity` = 1 から 500
- `SourceQueue` = /SOURCEQUEUE

また、MimeType、DHClass (データ・ハンドラー・クラス)、および DataHandlerConfigMOName (オプションのメタオブジェクト名) プロパティーを設定したデータ・ハンドラーも構成する必要があります。これらのプロパティーの値を設定するには、Connector Configurator の「データ・ハンドラー」タブを使用します。

これらのプロパティーはアダプター固有ですが、例の値は次のようになります。

- MimeType = text/xml
- DHClass = com.crossworlds.DataHandlers.text.xml
- DataHandlerConfigMOName = M0_DataHandler_Default

「データ・ハンドラー」タブのこれらの値のフィールドは、ContainerManagedEvents を JMS に設定した場合にのみ表示されます。

注: ContainerManagedEvents を JMS に設定した場合、コネクターはその pollForEvents() メソッドを呼び出さなくなるため、そのメソッドの機能は使用できなくなります。

このプロパティーは、DeliveryTransport プロパティーが値 JMS に設定されている場合にのみ表示されます。

ControllerStoreAndForwardMode

RepositoryDirectory が <REMOTE> の場合のみ適用可能です。

宛先側のアプリケーション固有のコンポーネントが使用不可であることをコネクター・コントローラーが検出した場合に、コネクター・コントローラーが実行する動作を設定します。

このプロパティーを true に設定した場合、イベントが ICS に到達したときに宛先側のアプリケーション固有のコンポーネントが使用不可であれば、コネクター・コントローラーはそのアプリケーション固有のコンポーネントへの要求をブロックします。アプリケーション固有のコンポーネントが作動可能になると、コネクター・コントローラーはアプリケーション固有のコンポーネントにその要求を転送します。

ただし、コネクター・コントローラーが宛先側のアプリケーション固有のコンポーネントにサービス呼び出し要求を転送した後でこのコンポーネントが使用不可になった場合、コネクター・コントローラーはその要求を失敗させます。

このプロパティーを false に設定した場合、コネクター・コントローラーは、宛先側のアプリケーション固有のコンポーネントが使用不可であることを検出すると、ただちにすべてのサービス呼び出し要求を失敗させます。

デフォルト値は true です。

ControllerTraceLevel

RepositoryDirectory が <REMOTE> の場合のみ適用可能です。

コネクター・コントローラーのトレース・メッセージのレベルです。デフォルト値は 0 です。

DeliveryQueue

DeliveryTransport が JMS の場合のみ適用されます。

コネクタから統合ブローカーへビジネス・オブジェクトが送信されるときに使用されるキューです。

デフォルト値は CONNECTORNAME/DELIVERYQUEUE です。

DeliveryTransport

イベントのデリバリーのためのトランスポート機構を指定します。指定可能な値は、WebSphere MQ の MQ、CORBA IIOP の IDL、または Java Messaging Service の JMS です。

- RepositoryDirectory がリモートの場合は、DeliveryTransport プロパティの指定可能な値は MQ、IDL、または JMS であり、デフォルトは IDL になります。
- RepositoryDirectory がローカル・ディレクトリーの場合は、指定可能な値は JMS のみです。

DeliveryTransport プロパティに指定されている値が、MQ または IDL である場合、コネクタは、CORBA IIOP を使用してサービス呼び出し要求と管理メッセージを送信します。

WebSphere MQ および IDL

イベントのデリバリー・トランスポートには、IDL ではなく WebSphere MQ を使用してください (1 種類の製品だけを使用する必要がある場合を除きます)。

WebSphere MQ が IDL よりも優れている点は以下のとおりです。

- 非同期 (ASYNC) 通信:
WebSphere MQ を使用すると、アプリケーション固有のコンポーネントは、サーバーが利用不能である場合でも、イベントをポーリングして永続的に格納することができます。
- サーバー・サイド・パフォーマンス:
WebSphere MQ を使用すると、サーバー・サイドのパフォーマンスが向上します。最適化モードでは、WebSphere MQ はイベントへのポインターのみをリポジトリ・データベースに格納するので、実際のイベントは WebSphere MQ キュー内に残ります。これにより、サイズが大きい可能性のあるイベントをリポジトリ・データベースに書き込む必要がありません。
- エージェント・サイド・パフォーマンス:
WebSphere MQ を使用すると、アプリケーション固有のコンポーネント側のパフォーマンスが向上します。WebSphere MQ を使用すると、コネクタのポーリング・スレッドは、イベントを選出した後、コネクタのキューにそのイベントを入れ、次のイベントを選出します。この方法は IDL よりも高速です。IDL の場合、コネクタのポーリング・スレッドは、イベントを選出した後、ネットワーク経由でサーバー・プロセスにアクセスしてそのイベントをリポジトリ・データベースに永続的に格納してから、次のイベントを選出する必要があります。

JMS

Java Messaging Service (JMS) を使用しての、コネクタとクライアント・コネクタ・フレームワークとの間の通信を可能にします。

JMS をデリバリー・トランスポートとして選択した場合は、`jms.MessageBrokerName`、`jms.FactoryClassName`、`jms.Password`、`jms.UserName` などの追加の JMS プロパティが Connector Configurator 内に表示されます。このうち最初の 2 つは、このトランスポートの必須プロパティです。

要確認: 以下の環境では、コネクタに JMS トランスポート機構を使用すると、メモリー制限が発生することもあります。

- AIX 5.0
- WebSphere MQ 5.3.0.1
- ICS が統合ブローカーの場合

この環境では、WebSphere MQ クライアント内でメモリーが使用されるため、(サーバー側の) コネクタ・コントローラーと (クライアント側の) コネクタの両方を始動するのは困難な場合があります。ご使用のシステムのプロセス・ヒープ・サイズが 768M 未満である場合には、次のように設定することをお勧めします。

- `CWSharedEnv.sh` スクリプト内で `LDR_CNTRL` 環境変数を設定する。

このスクリプトは、製品ディレクトリー配下の `%bin` ディレクトリーにあります。テキスト・エディターを使用して、`CWSharedEnv.sh` スクリプトの最初の行として次の行を追加します。

```
export LDR_CNTRL=MAXDATA=0x30000000
```

この行は、ヒープ・メモリーの使用量を最大 768 MB (3 セグメント * 256 MB) に制限します。プロセス・メモリーがこの制限値を超えると、ページ・スワッピングが発生し、システムのパフォーマンスに悪影響を与える場合があります。

- `IPCCBaseAddress` プロパティの値を 11 または 12 に設定する。このプロパティの詳細については、「システム・インストール・ガイド (UNIX 版)」を参照してください。

DuplicateEventElimination

このプロパティを `true` に設定すると、JMS 対応コネクタによるデリバリー・キューへの重複イベントのデリバリーが防止されます。この機能を使用するには、コネクタに対し、アプリケーション固有のコード内でビジネス・オブジェクトの `ObjectEventId` 属性として一意のイベント ID が設定されている必要があります。これはコネクタ開発時に設定されます。

このプロパティは、`false` に設定することもできます。

注: `DuplicateEventElimination` を `true` に設定する際は、`MonitorQueue` プロパティを構成して保証付きイベント・デリバリーを使用可能にする必要があります。

FaultQueue

コネクタでメッセージを処理中にエラーが発生すると、コネクタは、そのメッセージを状況表示および問題説明とともにこのプロパティに指定されているキューに移動します。

デフォルト値は `CONNECTORNAME/FAULTQUEUE` です。

JvmMaxHeapSize

エージェントの最大ヒープ・サイズ (メガバイト単位)。このプロパティは、`RepositoryDirectory` の値が `<REMOTE>` の場合にのみ適用されます。

デフォルト値は `128M` です。

JvmMaxNativeStackSize

エージェントの最大ネイティブ・スタック・サイズ (キロバイト単位)。このプロパティは、`RepositoryDirectory` の値が `<REMOTE>` の場合にのみ適用されます。

デフォルト値は `128K` です。

JvmMinHeapSize

エージェントの最小ヒープ・サイズ (メガバイト単位)。このプロパティは、`RepositoryDirectory` の値が `<REMOTE>` の場合にのみ適用されます。

デフォルト値は `1M` です。

jms.FactoryClassName

JMS プロバイダーのためにインスタンスを生成するクラス名を指定します。JMS をデリバリー・トランスポート機構 (`DeliveryTransport`) として選択する際は、このコネクタ・プロパティを必ず設定してください。

デフォルト値は `CxCommon.Messaging.jms.IBMMQSeriesFactory` です。

jms.MessageBrokerName

JMS プロバイダーのために使用するブローカー名を指定します。JMS をデリバリー・トランスポート機構 (`DeliveryTransport`) として選択する際は、このコネクタ・プロパティを必ず設定してください。

デフォルト値は `crossworlds.queue.manager` です。ローカル・メッセージ・ブローカーに接続する場合は、デフォルト値を使用します。

リモート・メッセージ・ブローカーに接続すると、このプロパティは次の (必須) 値をとります。

`QueueMgrName:<Channel>:<HostName>:<PortNumber>`

各変数の意味は以下のとおりです。

`QueueMgrName`: キュー・マネージャー名です。

`Channel`: クライアントが使用するチャンネルです。

`HostName`: キュー・マネージャーの配置先のマシン名です。

`PortNumber`: キュー・マネージャーが `listen` に使用するポートの番号です。

例えば、次のようにします。

```
jms.MessageBrokerName = WBIMB.Queue.Manager:CHANNEL1:RemoteMachine:1456
```

jms.NumConcurrentRequests

コネクターに対して同時に送信することができる並行サービス呼び出し要求の数(最大値)を指定します。この最大値に達した場合、新規のサービス呼び出し要求はブロックされ、既存のいずれかの要求が完了した後で処理されます。

デフォルト値は 10 です。

jms.Password

JMS プロバイダーのためのパスワードを指定します。このプロパティーの値はオプションです。

デフォルトはありません。

jms.UserName

JMS プロバイダーのためのユーザー名を指定します。このプロパティーの値はオプションです。

デフォルトはありません。

ListenerConcurrency

このプロパティーは、統合ブローカーとして ICS を使用する場合の MQ Listener でのマルチスレッド化をサポートしています。このプロパティーにより、データベースへの複数イベントの書き込み操作をバッチ処理できるので、システム・パフォーマンスが向上します。デフォルト値は 1 です。

このプロパティーは、MQ トランスポートを使用するコネクターにのみ適用されません。DeliveryTransport プロパティーには MQ を設定してください。

Locale

言語コード、国または地域、および、希望する場合には、関連した文字コード・セットを指定します。このプロパティーの値は、データの照合やソート順、日付と時刻の形式、通貨記号などの国/地域別情報を決定します。

ロケール名は、次のフォーマットで指定します。

```
ll_TT.codeset
```

ここで、以下のように説明されます。

ll	2 文字の言語コード (普通は小文字)
TT	2 文字の国または地域コード (普通は大文字)
codeset	関連文字コード・セットの名前。名前のこの部分は、通常、オプションです。

デフォルトでは、ドロップダウン・リストには、サポートされるロケールの一部のみが表示されます。ドロップダウン・リストに、サポートされる他の値を追加する

には、製品ディレクトリーにある `¥Data¥Std¥stdConnProps.xml` ファイルを手動で変更する必要があります。詳細については、Connector Configurator に関する章を参照してください。

デフォルト値は `en_US` です。コネクタがグローバル化に対応していない場合、このプロパティの有効な値は `en_US` のみです。特定のコネクタがグローバル化に対応しているかどうかを判別するには、以下の Web サイトにあるコネクタのバージョン・リストを参照してください。

<http://www.ibm.com/software/websphere/wbiadapters/infocenter>、または
<http://www.ibm.com/websphere/integration/wicsserver/infocenter>

LogAtInterchangeEnd

RepositoryDirectory が <REMOTE> の場合のみ適用可能です。

統合ブローカーのログ宛先にエラーを記録するかどうかを指定します。ブローカーのログ宛先にログを記録すると、E メール通知もオンになります。これにより、エラーまたは致命的エラーが発生すると、InterchangeSystem.cfg ファイルに指定された MESSAGE_RECIPIENT に対する E メール・メッセージが生成されます。

例えば、LogAtInterChangeEnd を true に設定した場合にコネクタからアプリケーションへの接続が失われると、指定されたメッセージ宛先に、E メール・メッセージが送信されます。デフォルト値は false です。

MaxEventCapacity

コントローラー・バッファ内のイベントの最大数。このプロパティはフロー制御が使用し、RepositoryDirectory プロパティの値が <REMOTE> の場合のみ適用されます。

値は 1 から 2147483647 の間の正整数です。デフォルト値は 2147483647 です。

MessageFileName

コネクタ・メッセージ・ファイルの名前です。メッセージ・ファイルの標準位置は、製品ディレクトリーの `¥connectors¥messages` です。メッセージ・ファイルが標準位置に格納されていない場合は、メッセージ・ファイル名を絶対パスで指定します。

コネクタ・メッセージ・ファイルが存在しない場合は、コネクタは InterchangeSystem.txt をメッセージ・ファイルとして使用します。このファイルは、製品ディレクトリーに格納されています。

注: 特定のコネクタについて、コネクタ独自のメッセージ・ファイルがあるかどうかを判別するには、該当するアダプターのユーザズ・ガイドを参照してください。

MonitorQueue

コネクタが重複イベントをモニターするために使用する論理キューです。このプロパティは、DeliveryTransport プロパティ値が JMS であり、かつ DuplicateEventElimination が TRUE に設定されている場合のみ使用されます。

デフォルト値は CONNECTORNAME/MONITORQUEUE です。

OADAutoRestartAgent

RepositoryDirectory が <REMOTE> の場合のみ有効です。

コネクタが自動再始動およびリモート再始動機能を使用するかどうかを指定します。この機能では、MQ により起動される Object Activation Daemon (OAD) を使用して、異常シャットダウン後にコネクタを再始動したり、System Monitor からリモート・コネクタを始動したりします。

自動再始動機能およびリモート再始動機能を使用可能にするには、このプロパティを true に設定する必要があります。MQ によりトリガーされる OAD 機能の構成方法については、「システム・インストール・ガイド (Windows 版)」または「システム・インストール・ガイド (UNIX 版)」を参照してください。

デフォルト値は false です。

OADMaxNumRetry

RepositoryDirectory が <REMOTE> の場合のみ有効です。

異常シャットダウンの後で MQ によりトリガーされる OAD がコネクタの再始動を自動的に試行する回数の最大数を指定します。このプロパティを有効にするためには、OADAutoRestartAgent プロパティを true に設定する必要があります。

デフォルト値は 1000 です。

OADRetryTimeInterval

RepositoryDirectory が <REMOTE> の場合のみ有効です。

MQ によりトリガーされる OAD の再試行時間間隔の分数を指定します。コネクタ・エージェントがこの再試行時間間隔内に再始動しない場合は、コネクタ・コントローラーはコネクタ・エージェントを再び再始動するように OAD に要求します。OAD はこの再試行プロセスを OADMaxNumRetry プロパティで指定された回数だけ繰り返します。このプロパティを有効にするためには、OADAutoRestartAgent プロパティを true に設定する必要があります。

デフォルト値は 10 です。

PollEndTime

イベント・キューのポーリングを停止する時刻です。形式は HH:MM です。ここで、HH は 0 から 23 時を表し、MM は 0 から 59 分を表します。

このプロパティには必ず有効な値を指定してください。デフォルト値は HH:MM ですが、この値は必ず変更する必要があります。

PollFrequency

これは、前回のポーリングの終了から次のポーリングの開始までの間の間隔です。PollFrequency は、あるポーリング・アクションの終了から次のポーリング・アク

ションの開始までの時間をミリ秒単位で指定します。これはポーリング・アクション間の間隔ではありません。この論理を次に説明します。

- ポーリングし、PollQuantity の値により指定される数のオブジェクトを取得します。
- これらのオブジェクトを処理します。一部のアダプターでは、これは個別のスレッドで部分的に実行されます。これにより、次のポーリング・アクションまで処理が非同期に実行されます。
- PollFrequency で指定された間隔にわたって遅延します。
- このサイクルを繰り返します。

PollFrequency は以下の値のいずれかに設定します。

- ポーリング・アクション間のミリ秒数 (整数)。
- ワード key。コネクターは、コネクターのコマンド・プロンプト・ウィンドウで文字 p が入力されたときにのみポーリングを実行します。このワードは小文字で入力します。
- ワード no。コネクターはポーリングを実行しません。このワードは小文字で入力します。

デフォルト値は 10000 です。

要確認: 一部のコネクターでは、このプロパティの使用が制限されています。このようなコネクターが存在する場合には、アダプターのインストールと構成に関する章で制約事項が説明されています。

PollQuantity

コネクターがアプリケーションからポーリングする項目の数を指定します。アダプターにコネクター固有のポーリング数設定プロパティがある場合、標準プロパティの値は、このコネクター固有のプロパティの設定値によりオーバーライドされます。

電子メール・メッセージもイベントと見なされます。コネクターは、電子メールに関するポーリングを受けたときには次のように動作します。

コネクターは、1 回目のポーリングを受けると、メッセージの本文を選出します。これは、本文が添付とも見なされるからです。本文の MIME タイプにはデータ・ハンドラーが指定されていないので、コネクターは本文を無視します。

コネクターは PO の最初の添付を処理します。この添付の MIME タイプには対応する DH があるので、コネクターはビジネス・オブジェクトを Visual Test Connector に送信します。

2 回目のポーリングを受けると、コネクターは PO の 2 番目の添付を処理します。この添付の MIME タイプには対応する DH があるので、コネクターはビジネス・オブジェクトを Visual Test Connector に送信します。

これが受け入れられると、PO の 3 番目の添付が届きます。

PollStartTime

イベント・キューのポーリングを開始する時刻です。形式は HH:MM です。ここで、HH は 0 から 23 時を表し、MM は 0 から 59 分を表します。

このプロパティーには必ず有効な値を指定してください。デフォルト値は HH:MM ですが、この値は必ず変更する必要があります。

RequestQueue

統合ブローカーが、ビジネス・オブジェクトをコネクターに送信するときに使用されるキューです。

デフォルト値は CONNECTOR/REQUESTQUEUE です。

RepositoryDirectory

コネクターが XML スキーマ文書を読み取るリポジトリの場所です。この XML スキーマ文書には、ビジネス・オブジェクト定義のメタデータが含まれています。

統合ブローカーが ICS の場合はこの値を <REMOTE> に設定する必要があります。これは、コネクターが InterChange Server リポジトリからこの情報を取得するためです。

統合ブローカーが WebSphere Message Broker または WAS の場合は、この値を <local directory> に設定する必要があります。

ResponseQueue

DeliveryTransport が JMS の場合のみ適用可能で、RepositoryDirectory が <REMOTE> の場合のみ必須です。

JMS 応答キューを指定します。JMS 応答キューは、応答メッセージをコネクター・フレームワークから統合ブローカーへデリバリーします。統合ブローカーが ICS の場合、サーバーは要求を送信し、JMS 応答キューの応答メッセージを待ちます。

RestartRetryCount

コネクターによるコネクター自体の再始動の試行回数を指定します。このプロパティーを並列コネクターに対して使用する場合、コネクターのマスター側のアプリケーション固有のコンポーネントがスレーブ側のアプリケーション固有のコンポーネントの再始動を試行する回数が指定されます。

デフォルト値は 3 です。

RestartRetryInterval

コネクターによるコネクター自体の再始動の試行間隔を分単位で指定します。このプロパティーを並列コネクターに対して使用する場合、コネクターのマスター側のアプリケーション固有のコンポーネントがスレーブ側のアプリケーション固有のコンポーネントの再始動を試行する間隔が指定されます。指定可能な値の範囲は 1 から 2147483647 です。

デフォルト値は 1 です。

RHF2MessageDomain

WebSphere Message Brokers および WAS でのみ使用されます。

このプロパティにより、JMS ヘッダーのドメイン名フィールドの値を構成できます。JMS トランスポートを介してデータを WMQI に送信するときに、アダプター・フレームワークにより JMS ヘッダー情報、ドメイン名、および固定値 `mrm` が書き込まれます。この構成可能なドメイン名により、ユーザーは WMQI ブローカーによるメッセージ・データの処理方法を追跡できます。

サンプル・ヘッダーを以下に示します。

```
<mcd><Msd>mrm</Msd><Set>3</Set><Type>
Retek_POPhyDesc</Type><Fmt>CwXML</Fmt></mcd>
```

デフォルト値は `mrm` ですが、このプロパティには `xml` も設定できます。このプロパティは、`DeliveryTransport` が JMS に設定されており、かつ `WireFormat` が `CwXML` に設定されている場合にのみ表示されます。

SourceQueue

`DeliveryTransport` が JMS で、`ContainerManagedEvents` が指定されている場合のみ適用されます。

JMS イベント・ストアを使用する JMS 対応コネクタでの保証付きイベント・デリバリーをサポートするコネクタ・フレームワークに、JMS ソース・キューを指定します。詳細については、55 ページの『`ContainerManagedEvents`』を参照してください。

デフォルト値は `CONNECTOR/SOURCEQUEUE` です。

SynchronousRequestQueue

`DeliveryTransport` が JMS の場合のみ適用されます。

同期応答を要求する要求メッセージを、コネクタ・フレームワークからブローカーにデリバリーします。このキューは、コネクタが同期実行を使用する場合にのみ必要です。同期実行の場合、コネクタ・フレームワークは、`SynchronousRequestQueue` にメッセージを送信し、`SynchronousResponseQueue` でブローカーから戻される応答を待機します。コネクタに送信される応答メッセージには、元のメッセージの ID を指定する 相関 ID が含まれています。

デフォルトは `CONNECTORNAME/SYNCHRONOUSREQUESTQUEUE` です。

SynchronousResponseQueue

`DeliveryTransport` が JMS の場合のみ適用されます。

同期要求に対する応答として送信される応答メッセージを、ブローカーからコネクタ・フレームワークにデリバリーします。このキューは、コネクタが同期実行を使用する場合にのみ必要です。

デフォルトは `CONNECTORNAME/SYNCHRONOUSRESPONSEQUEUE` です。

SynchronousRequestTimeout

`DeliveryTransport` が JMS の場合のみ適用されます。

コネクタが同期要求への応答を待機する時間を分単位で指定します。コネクタは、指定された時間内に応答を受信できなかった場合、元の同期要求メッセージをエラー・メッセージとともに障害キューに移動します。

デフォルト値は 0 です。

WireFormat

トランスポートのメッセージ・フォーマットです。

- RepositoryDirectory がローカル・ディレクトリーの場合は、設定は CwXML になります。
- RepositoryDirectory の値が <REMOTE> の場合には、設定値は CwBO です。

WsifSynchronousRequestTimeout

WAS 統合ブローカーでのみ使用されます。

コネクタが同期要求への応答を待機する時間を分単位で指定します。コネクタは、指定された時間内に応答を受信できなかった場合、元の同期要求メッセージをエラー・メッセージとともに障害キューに移動します。

デフォルト値は 0 です。

XMLNamespaceFormat

WebSphere Message Brokers および WAS 統合ブローカーでのみ使用されます。

ビジネス・オブジェクト定義の XML 形式でネーム・スペースを short と long のどちらにするかをユーザーが指定できるようにするための、強力なプロパティです。

デフォルト値は short です。

付録 B. Connector Configurator

本章では、Connector Configurator を使用してアダプターの構成プロパティー値を設定する方法について説明します。

Connector Configurator を使用して次の作業を行います。

- コネクタを構成するためのコネクタ固有のプロパティー・テンプレートを作成する
- 構成ファイルを作成する
- 構成ファイル内のプロパティーを設定する

注:

本書では、ディレクトリー・パスの規則として円記号 (¥) を使用します。UNIX システムを使用している場合は、円記号をスラッシュ (/) に置き換えてください。また、各オペレーティング・システムの規則に従ってください。

本章では、次のトピックについて説明します。

- 『Connector Configurator の概要』
- 68 ページの『Connector Configurator の始動』
- 69 ページの『コネクタ固有のプロパティー・テンプレートの作成』
- 72 ページの『新規構成ファイルの作成』
- 75 ページの『構成ファイル・プロパティーの設定』
- 84 ページの『グローバル化環境における Connector Configurator の使用』

Connector Configurator の概要

Connector Configurator では、次の統合ブローカーで使用するアダプターのコネクタ・コンポーネントを構成できます。

- WebSphere InterChange Server (ICS)
- WebSphere MQ Integrator、WebSphere MQ Integrator Broker、および WebSphere Business Integration Message Broker (WebSphere Message Brokers (WMQI) と総称)
- WebSphere Application Server (WAS)

Connector Configurator を使用して次の作業を行います。

- コネクタを構成するためのコネクタ固有のプロパティー・テンプレートを作成する。
- **コネクタ構成ファイル**を作成します。インストールするコネクタごとに構成ファイルを 1 つ作成する必要があります。
- 構成ファイル内のプロパティーを設定する。
場合によっては、コネクタ・テンプレートでプロパティーに対して設定されているデフォルト値を変更する必要があります。また、サポートされるビジネス・オブジェクト定義と、ICS の場合はコラボレーションとともに使用するマップを

指定し、必要に応じてメッセージング、ロギング、トレース、およびデータ・ハンドラー・パラメーターを指定する必要があります。

Connector Configurator の実行モードと使用する構成ファイルのタイプは、実行する統合ブローカーによって異なります。例えば、使用している統合ブローカーが WMQI の場合、Connector Configurator を System Manager から実行するのではなく、直接実行します (『スタンドアロン・モードでの Configurator の実行』を参照)。

コネクタ構成プロパティには、標準の構成プロパティ (すべてのコネクタが持つプロパティ) と、コネクタ固有のプロパティ (特定のアプリケーションまたはテクノロジーのためにコネクタで必要なプロパティ) とが含まれます。

標準プロパティはすべてのコネクタにより使用されるので、標準プロパティを新規に定義する必要はありません。ファイルを作成すると、Connector Configurator により標準プロパティがこの構成ファイルに挿入されます。ただし、Connector Configurator で各標準プロパティの値を設定する必要があります。

標準プロパティの範囲は、ブローカーと構成によって異なる可能性があります。特定のプロパティに特定の値が設定されている場合にのみ使用できるプロパティがあります。Connector Configurator の「標準のプロパティ」ウィンドウには、特定の構成で設定可能なプロパティが表示されます。

ただし**コネクタ固有プロパティ**の場合は、最初にプロパティを定義し、その値を設定する必要があります。このため、特定のアダプターのコネクタ固有プロパティのテンプレートを作成します。システム内で既にテンプレートが作成されている場合には、作成されているテンプレートを使用します。システム内でまだテンプレートが作成されていない場合には、70 ページの『新規テンプレートの作成』のステップに従い、テンプレートを新規に作成します。

注: Connector Configurator は、Windows 環境内でのみ実行されます。UNIX 環境でコネクタを実行する場合には、Windows で Connector Configurator を使用して構成ファイルを変更し、このファイルを UNIX 環境へコピーします。

Connector Configurator の始動

以下の 2 種類のモードで Connector Configurator を開始および実行できます。

- スタンドアロン・モードで個別に実行
- System Manager から

スタンドアロン・モードでの Configurator の実行

どのブローカーを実行している場合にも、Connector Configurator を個別に実行し、コネクタ構成ファイルを編集できます。

これを行うには、以下のステップを実行します。

- 「スタート」>「プログラム」から、「IBM WebSphere InterChange Server」>「IBM WebSphere Business Integration Tools」>「Connector Configurator」をクリックします。
- 「ファイル」>「新規」>「コネクタ構成」を選択します。

- 「システム接続: Integration Broker」の隣のプルダウン・メニューをクリックします。使用しているブローカーに応じて、ICS、WebSphere Message Brokers、または WAS を選択します。

Connector Configurator を個別に実行して構成ファイルを生成してから、System Manager に接続してこの構成ファイルを System Manager プロジェクトに保存することもできます (75 ページの『構成ファイルの完成』を参照)。

System Manager からの Configurator の実行

System Manager から Connector Configurator を実行できます。

Connector Configurator を実行するには、以下のステップを実行します。

1. System Manager を開きます。
2. 「System Manager」ウィンドウで、「統合コンポーネント・ライブラリー」アイコンを展開し、「コネクタ」を強調表示します。
3. System Manager メニュー・バーから、「ツール」>「**Connector Configurator**」をクリックします。「Connector Configurator」ウィンドウが開き、「新規コネクタ」ダイアログ・ボックスが表示されます。
4. 「システム接続: Integration Broker」の隣のプルダウン・メニューをクリックします。使用しているブローカーに応じて、ICS、WebSphere Message Brokers、または WAS を選択します。

既存の構成ファイルを編集するには、以下のステップを実行します。

- 「System Manager」ウィンドウの「コネクタ」フォルダーでいずれかの構成ファイルを選択し、右クリックします。Connector Configurator が開き、この構成ファイルの統合ブローカー・タイプおよびファイル名が上部に表示されます。
- Connector Configurator で「ファイル」>「開く」を選択します。プロジェクトまたはプロジェクトが保管されているディレクトリーからコネクタ構成ファイルを選択します。
- 「標準のプロパティ」タブをクリックし、この構成ファイルに含まれているプロパティを確認します。

コネクタ固有のプロパティ・テンプレートの作成

コネクタの構成ファイルを作成するには、コネクタ固有プロパティのテンプレートとシステム提供の標準プロパティが必要です。

コネクタ固有プロパティのテンプレートを新規に作成するか、または既存のコネクタ定義をテンプレートとして使用します。

- テンプレートの新規作成については、70 ページの『新規テンプレートの作成』を参照してください。
- 既存のファイルを使用する場合には、既存のテンプレートを変更し、新しい名前でのこのテンプレートを保管します。既存のテンプレートは `¥WebSphereAdapters¥bin¥Data¥App` ディレクトリーにあります。

新規テンプレートの作成

このセクションでは、テンプレートでプロパティを作成し、プロパティの一般特性および値を定義し、プロパティ間の依存関係を指定する方法について説明します。次にそのテンプレートを保管し、新規コネクタ構成ファイルを作成するためのベースとして使用します。

Connector Configurator でテンプレートを作成するには、以下のステップを実行します。

1. 「ファイル」>「新規」>「コネクタ固有プロパティ・テンプレート」をクリックします。
2. 「コネクタ固有プロパティ・テンプレート」ダイアログ・ボックスが表示されます。
 - 「新規テンプレート名を入力してください」の下の「名前」フィールドに、新規テンプレートの名前を入力します。テンプレートから新規構成ファイルを作成するためのダイアログ・ボックスを開くと、この名前が再度表示されます。
 - テンプレートに含まれているコネクタ固有のプロパティ定義を調べるには、「テンプレート名」表示でそのテンプレートの名前を選択します。そのテンプレートに含まれているプロパティ定義のリストが「テンプレートのプレビュー」表示に表示されます。
3. テンプレートを作成するときには、ご使用のコネクタに必要なプロパティ定義に類似したプロパティ定義が含まれている既存のテンプレートを使用できます。ご使用のコネクタで使用するコネクタ固有のプロパティが表示されるテンプレートが見つからない場合は、自分で作成する必要があります。
 - 既存のテンプレートを変更する場合には、「変更する既存のテンプレートを選択してください: 検索テンプレート」の下の「テンプレート名」テーブルのリストから、テンプレート名を選択します。
 - このテーブルには、現在使用可能なすべてのテンプレートの名前が表示されます。テンプレートを検索することもできます。

一般特性の指定

「次へ」をクリックしてテンプレートを選択すると、「プロパティ: コネクタ固有プロパティ・テンプレート」ダイアログ・ボックスが表示されます。このダイアログ・ボックスには、定義済みプロパティの「一般」特性のタブと「値」の制限のタブがあります。「一般」表示には以下のフィールドがあります。

- **一般:**
 - プロパティ・タイプ
 - 更新されたメソッド
 - 説明
- **フラグ**
 - 標準フラグ
- **カスタム・フラグ**
 - フラグ

プロパティの一般特性の選択を終えたら、「値」タブをクリックします。

値の指定

「値」タブを使用すると、プロパティの最大長、最大複数値、デフォルト値、または値の範囲を設定できます。編集可能な値も許可されます。これを行うには、以下のステップを実行します。

1. 「値」タブをクリックします。「一般」のパネルに代わって「値」の表示パネルが表示されます。
2. 「プロパティを編集」表示でプロパティの名前を選択します。
3. 「最大長」および「最大複数値」のフィールドに値を入力します。

新規プロパティ値を作成するには、以下のステップを実行します。

1. 「プロパティを編集」リストでプロパティを選択し、右マウス・ボタンでクリックします。
2. ダイアログ・ボックスから「追加」を選択します。
3. 新規プロパティ値の名前を入力し、「OK」をクリックします。右側の「値」パネルに値が表示されます。

「値」パネルには、3つの列からなるテーブルが表示されます。

「値」の列には、「プロパティ値」ダイアログ・ボックスで入力した値と、以前に作成した値が表示されます。

「デフォルト値」の列では、値のいずれかをデフォルトとして指定することができます。

「値の範囲」の列には、「プロパティ値」ダイアログ・ボックスで入力した範囲が表示されます。

値が作成されて、グリッドに表示されると、そのテーブルの表示内から編集できるようになります。

テーブルにある既存の値の変更を行うには、その行の行番号をクリックして行全体を選択します。次に「値」フィールドを右マウス・ボタンでクリックし、「値の編集 (Edit Value)」をクリックします。

依存関係の設定

「一般」タブと「値」タブで変更を行ったら、「次へ」をクリックします。「依存関係: コネクター固有プロパティ・テンプレート」ダイアログ・ボックスが表示されます。

依存プロパティは、別のプロパティの値が特定の条件に合致する場合にのみ、テンプレートに組み込まれて、構成ファイルで使用されるプロパティです。例えば、テンプレートに `PollQuantity` が表示されるのは、トランスポート機構が `JMS` であり、`DuplicateEventElimination` が `True` に設定されている場合のみです。プロパティを依存プロパティとして指定し、依存する条件を設定するには、以下のステップを実行します。

1. 「使用可能なプロパティ」表示で、依存プロパティとして指定するプロパティを選択します。

2. 「プロパティを選択」フィールドで、ドロップダウン・メニューを使用して、条件値を持たせるプロパティを選択します。
3. 「条件演算子」フィールドで以下のいずれかを選択します。

== (等しい)

!= (等しくない)

> (より大)

< (より小)

>= (より大か等しい)

<= (より小か等しい)

4. 「条件値」フィールドで、依存プロパティをテンプレートに組み込むために必要な値を入力します。
5. 「使用可能なプロパティ」表示で依存プロパティを強調表示させて矢印をクリックし、「依存プロパティ」表示に移動させます。
6. 「完了」をクリックします。Connector Configurator により、XML 文書として入力した情報が、Connector Configurator がインストールされている %bin ディレクトリーの %data¥app の下に保管されます。

新規構成ファイルの作成

構成ファイルを新規に作成するには、構成ファイルの名前を指定し、統合ブローカーを選択する必要があります。

- 「System Manager」ウィンドウで「コネクタ」フォルダーを右クリックし、「新規コネクタの作成」を選択します。Connector Configurator が開き、「新規コネクタ」ダイアログ・ボックスが表示されます。
- スタンドアロン・モードの場合は、Connector Configurator で「ファイル」>「新規」>「コネクタ構成」を選択します。「新規コネクタ」ウィンドウで、新規コネクタの名前を入力します。

また、統合ブローカーも選択する必要があります。選択したブローカーによって、構成ファイルに記述されるプロパティが決まります。ブローカーを選択するには、以下のステップを実行します。

- 「Integration Broker」フィールドで、ICS 接続、WebSphere Message Brokers 接続、WAS 接続のいずれかを選択します。
- この章で後述する説明に従って「新規コネクタ」ウィンドウの残りのフィールドに入力します。

コネクタ固有のテンプレートからの構成ファイルの作成

コネクタ固有のテンプレートを作成すると、テンプレートを使用して構成ファイルを作成できます。

1. 「ファイル」>「新規」>「コネクタ構成」をクリックします。
2. 以下のフィールドを含む「新規コネクタ」ダイアログ・ボックス表示されません。

- **名前**

コネクタの名前を入力します。名前では大文字と小文字が区別されます。入力する名前は、システムにインストールされているコネクタのファイル名に対応した一意の名前でなければなりません。

要確認: Connector Configurator では、入力された名前のスペルはチェックされません。名前が正しいことを確認してください。

- **システム接続**

ICS 接続、WebSphere Message Brokers 接続、WAS のいずれかをクリックします。

- **コネクタ固有プロパティ・テンプレートを選択 (Select Connector-Specific Property Template)**

ご使用のコネクタ用に設計したテンプレートの名前を入力します。「**テンプレート名**」表示に、使用可能なテンプレートが表示されます。「**テンプレート名**」表示で名前を選択すると、「**プロパティ・テンプレートのプレビュー**」表示に、そのテンプレートで定義されているコネクタ固有のプロパティが表示されます。

使用するテンプレートを選択し、「**OK**」をクリックします。

3. 構成しているコネクタの構成画面が表示されます。タイトル・バーに統合ブローカーとコネクタの名前が表示されます。ここですべてのフィールドに値を入力して定義を完了するか、ファイルを保管して後でフィールドに値を入力するかを選択できます。
4. ファイルを保管するには、「**ファイル**」>「**保管**」>「**ファイルに**」をクリックするか、「**ファイル**」>「**保管**」>「**プロジェクトに**」をクリックします。プロジェクトに保管するには、System Manager が実行中でなければなりません。ファイルとして保管する場合は、「**ファイル・コネクタを保管**」ダイアログ・ボックスが表示されます。***.cfg** をファイル・タイプとして選択し、「**ファイル名**」フィールド内に名前が正しいスペル (大文字と小文字の区別を含む) で表示されていることを確認してから、ファイルを保管するディレクトリーにナビゲートし、「**保管**」をクリックします。Connector Configurator のメッセージ・パネルの状況表示に、構成ファイルが正常に作成されたことが示されます。

要確認: ここで設定するディレクトリー・パスおよび名前は、コネクタの始動ファイルで指定するコネクタ構成ファイルのパスおよび名前に一致している必要があります。

5. 本章で後述する手順に従って、「Connector Configurator」ウィンドウの各タブにあるフィールドに値を入力し、コネクタ定義を完了します。

既存ファイルの使用

使用可能な既存ファイルは、以下の 1 つまたは複数の形式になります。

- **コネクタ定義ファイル。**
コネクタ定義ファイルは、特定のコネクタのプロパティと、適用可能なデフォルト値がリストされたテキスト・ファイルです。コネクタの配布パッケージ

ジの `¥repository` ディレクトリー内には、このようなファイルが格納されていることがあります (通常、このファイルの拡張子は `.txt` です。例えば、XML コネクタの場合は `CN_XML.txt` です)。

- ICS リポジトリ・ファイル。
コネクタの以前の ICS 実装で使用した定義は、そのコネクタの構成で使用されたリポジトリ・ファイルで使用可能になります。そのようなファイルの拡張子は、通常 `.in` または `.out` です。
- コネクタの以前の構成ファイル。
これらのファイルの拡張子は、通常 `*.cfg` です。

これらのいずれのファイル・ソースにも、コネクタのコネクタ固有プロパティのほとんど、あるいはすべてが含まれますが、本章で後述するように、コネクタ構成ファイルは、ファイルを開いて、プロパティを設定しない限り完成しません。

既存ファイルを使用してコネクタを構成するには、Connector Configurator でそのファイルを開き、構成を修正し、そのファイルを再度保管する必要があります。

以下のステップを実行して、ディレクトリーから `*.txt`、`*.cfg`、または `*.in` ファイルを開きます。

1. Connector Configurator 内で、「ファイル」>「開く」>「ファイルから」をクリックします。
2. 「ファイル・コネクタを開く」ダイアログ・ボックス内で、以下のいずれかのファイル・タイプを選択して、使用可能なファイルを調べます。
 - 構成 (`*.cfg`)
 - ICS リポジトリ (`*.in`、`*.out`)

ICS 環境でのコネクタの構成にリポジトリ・ファイルが使用された場合には、このオプションを選択します。リポジトリ・ファイルに複数のコネクタ定義が含まれている場合は、ファイルを開くとすべての定義が表示されません。

- すべてのファイル (`*.*`)

コネクタのアダプター・パッケージに `*.txt` ファイルが付属していた場合、または別の拡張子で定義ファイルが使用可能である場合は、このオプションを選択します。

3. ディレクトリー表示内で、適切なコネクタ定義ファイルへ移動し、ファイルを選択し、「開く」をクリックします。

System Manager プロジェクトからコネクタ構成を開くには、以下のステップを実行します。

1. System Manager を始動します。System Manager が開始されている場合にのみ、構成を System Manager から開いたり、System Manager に保管したりできます。
2. Connector Configurator を始動します。
3. 「ファイル」>「開く」>「プロジェクトから」をクリックします。

構成ファイルの完成

構成ファイルを開くか、プロジェクトからコネクターを開くと、「Connector Configurator」ウィンドウに構成画面が表示されます。この画面には、現在の属性と値が表示されます。

構成画面のタイトルには、ファイル内で指定された統合ブローカーとコネクターの名前が表示されます。正しいブローカーが設定されていることを確認してください。正しいブローカーが設定されていない場合、コネクターを構成する前にブローカー値を変更してください。これを行うには、以下のステップを実行します。

1. 「標準のプロパティ」タブで、BrokerType プロパティの値フィールドを選択します。ドロップダウン・メニューで、値 ICS、WMQI、または WAS を選択します。
2. 選択したブローカーに関連付けられているプロパティが「標準のプロパティ」タブに表示されます。ここでファイルを保管するか、または 78 ページの『サポートされるビジネス・オブジェクト定義の指定』の説明に従い残りの構成フィールドに値を入力することができます。
3. 構成が完了したら、「ファイル」>「保管」>「プロジェクトに」を選択するか、または「ファイル」>「保管」>「ファイルに」を選択します。

ファイルに保管する場合は、*.cfg を拡張子として選択し、ファイルの正しい格納場所を選択して、「保管」をクリックします。

複数のコネクター構成を開いている場合、構成をすべてファイルに保管するには「すべてファイルに保管」を選択し、コネクター構成をすべて System Manager プロジェクトに保管するには「すべてプロジェクトに保管」をクリックします。

Connector Configurator では、ファイルを保管する前に、必須の標準プロパティすべてに値が設定されているかどうかを確認されます。必須の標準プロパティに値が設定されていない場合、Connector Configurator は、検証が失敗したというメッセージを表示します。構成ファイルを保管するには、そのプロパティの値を指定する必要があります。

構成ファイル・プロパティの設定

新規のコネクター構成ファイルを作成して名前を付けるとき、または既存のコネクター構成ファイルを開くときには、Connector Configurator によって構成画面が表示されます。構成画面には、必要な構成値のカテゴリーに対応する複数のタブがあります。

Connector Configurator では、すべてのブローカーで実行されているコネクターで、以下のカテゴリーのプロパティに値が設定されている必要があります。

- 標準のプロパティ
- コネクター固有のプロパティ
- サポートされるビジネス・オブジェクト
- トレース/ログ・ファイルの値
- データ・ハンドラー (保証付きイベント・デリバリーで JMS メッセージングを使用するコネクターの場合に該当する)

注: JMS メッセージングを使用するコネクタの場合は、データをビジネス・オブジェクトに変換するデータ・ハンドラーの構成に関して追加のカテゴリが表示される場合があります。

ICS で実行されているコネクタの場合、以下のプロパティの値も設定されている必要があります。

- 関連付けられたマップ
- リソース
- メッセージング (該当する場合)

要確認: Connector Configurator では、英語文字セットまたは英語以外の文字セットのいずれのプロパティ値も設定可能です。ただし、標準のプロパティおよびコネクタ固有プロパティ、およびサポートされるビジネス・オブジェクトの名前では、英語文字セットのみを使用する必要があります。

標準プロパティとコネクタ固有プロパティの違いは、以下のとおりです。

- コネクタの標準プロパティは、コネクタのアプリケーション固有のコンポーネントとブローカー・コンポーネントの両方によって共有されます。すべてのコネクタが同じ標準プロパティのセットを使用します。これらのプロパティの説明は、各アダプター・ガイドの付録 A にあります。変更できるのはこれらの値の一部のみです。
- アプリケーション固有のプロパティは、コネクタのアプリケーション固有コンポーネント (アプリケーションと直接対話するコンポーネント) のみに適用されます。各コネクタには、そのコネクタのアプリケーションだけで使用されるアプリケーション固有のプロパティがあります。これらのプロパティには、デフォルト値が用意されているものもあれば、そうでないものもあります。また、一部のデフォルト値は変更することができます。各アダプター・ガイドのインストールおよび構成の章に、アプリケーション固有のプロパティおよび推奨値が記述されています。

「標準プロパティ」と「コネクタ固有プロパティ」のフィールドは、どのフィールドが構成可能であるかを示すために色分けされています。

- 背景がグレーのフィールドは、標準のプロパティを表します。値を変更することはできますが、名前の変更およびプロパティの除去はできません。
- 背景が白のフィールドは、アプリケーション固有のプロパティを表します。これらのプロパティは、アプリケーションまたはコネクタの特定のニーズによって異なります。値の変更も、これらのプロパティの除去も可能です。
- 「値」フィールドは構成できます。
- プロパティごとに「更新メソッド」フィールドが表示されます。これは、変更された値をアクティブにするためにコンポーネントまたはエージェントの再始動が必要かどうかを示します。この設定を構成することはできません。

標準コネクタ・プロパティの設定

標準のプロパティの値を変更するには、以下の手順を実行します。

1. 値を設定するフィールド内でクリックします。

2. 値を入力するか、ドロップダウン・メニューが表示された場合にはメニューから値を選択します。
3. 標準のプロパティの値をすべて入力後、以下のいずれかを実行することができます。
 - 変更内容を破棄し、元の値を保持したままで Connector Configurator を終了するには、「ファイル」>「終了」をクリックし (またはウィンドウを閉じ)、変更内容を保管するかどうかを確認するプロンプトが出されたら「いいえ」をクリックします。
 - Connector Configurator 内の他のカテゴリーの値を入力するには、そのカテゴリーのタブを選択します。「標準のプロパティ」(またはその他のカテゴリー) で入力した値は、次のカテゴリーに移動しても保持されます。ウィンドウを閉じると、すべてのカテゴリーで入力した値を一括して保管するかまたは破棄するかを確認するプロンプトが出されます。
 - 修正した値を保管するには、「ファイル」>「終了」をクリックし (またはウィンドウを閉じ)、変更内容を保管するかどうかを確認するプロンプトが出されたら「はい」をクリックします。「ファイル」メニューまたはツールバーから「保管」>「ファイルに」をクリックする方法もあります。

アプリケーション固有の構成プロパティの設定

アプリケーション固有の構成プロパティの場合、プロパティ名の追加または変更、値の構成、プロパティの削除、およびプロパティの暗号化が可能です。プロパティのデフォルトの長さは 255 文字です。

1. グリッドの左上端の部分で右マウス・ボタンをクリックします。ポップアップ・メニュー・バーが表示されます。プロパティを追加するときは「追加」をクリックします。子プロパティを追加するには、親の行番号で右マウス・ボタンをクリックし、「子を追加」をクリックします。
2. プロパティまたは子プロパティの値を入力します。
3. プロパティを暗号化するには、「暗号化」ボックスを選択します。
4. 76 ページの『標準コネクタ・プロパティの設定』の説明に従い、変更内容を保管するかまたは破棄するかを選択します。

各プロパティごとに表示される「更新メソッド」は、変更された値をアクティブにするためにコンポーネントまたはエージェントの再始動が必要かどうかを示します。

要確認: 事前設定のアプリケーション固有のコネクタ・プロパティ名を変更すると、コネクタに障害が発生する可能性があります。コネクタをアプリケーションに接続したり正常に実行したりするために、特定のプロパティ名が必要である場合があります。

コネクタ・プロパティの暗号化

「コネクタ固有プロパティ」ウィンドウの「暗号化」チェック・ボックスにチェックマークを付けると、アプリケーション固有のプロパティを暗号化することができます。値の暗号化を解除するには、「暗号化」チェック・ボックスをクリックしてチェックマークを外し、「検証」ダイアログ・ボックスに正しい値を入力し、「OK」をクリックします。入力された値が正しい場合は、暗号化解除された値が表示されます。

各プロパティとそのデフォルト値のリストおよび説明は、各コネクターのアダプター・ユーザーズ・ガイドにあります。

プロパティに複数の値がある場合には、プロパティの最初の値に「暗号化」チェック・ボックスが表示されます。「暗号化」を選択すると、そのプロパティのすべての値が暗号化されます。プロパティの複数の値を暗号化解除するには、そのプロパティの最初の値の「暗号化」チェック・ボックスをクリックしてチェックマークを外してから、「検証」ダイアログ・ボックスで新規の値を入力します。入力値が一致すれば、すべての複数值が暗号化解除されます。

更新メソッド

付録『コネクターの標準構成プロパティ』の 48 ページの『プロパティ値の設定と更新』にある更新メソッドの説明を参照してください。

サポートされるビジネス・オブジェクト定義の指定

コネクターで使用するビジネス・オブジェクトを指定するには、Connector Configurator の「サポートされているビジネス・オブジェクト」タブを使用します。汎用ビジネス・オブジェクトと、アプリケーション固有のビジネス・オブジェクトの両方を指定する必要があり、またそれらのビジネス・オブジェクト間のマップの関連を指定することが必要です。

注: コネクターによっては、アプリケーションでイベント通知や (メタオブジェクトを使用した) 追加の構成を実行するために、特定のビジネス・オブジェクトをサポートされているものとして指定することが必要な場合もあります。詳細は、「コネクター開発ガイド (C++ 用)」または「コネクター開発ガイド (Java 用)」を参照してください。

ご使用のブローカーが ICS の場合

ビジネス・オブジェクト定義がコネクターでサポートされることを指定する場合や、既存のビジネス・オブジェクト定義のサポート設定を変更する場合は、「サポートされているビジネス・オブジェクト」タブをクリックし、以下のフィールドを使用してください。

ビジネス・オブジェクト名: ビジネス・オブジェクト定義がコネクターによってサポートされることを指定するには、System Manager を実行し、以下の手順を実行します。

1. 「ビジネス・オブジェクト名」リストで空のフィールドをクリックします。
System Manager プロジェクトに存在するすべてのビジネス・オブジェクト定義を示すドロップダウン・リストが表示されます。
2. 追加するビジネス・オブジェクトをクリックします。
3. ビジネス・オブジェクトの「エージェント・サポート」(以下で説明) を設定します。
4. 「Connector Configurator」ウィンドウの「ファイル」メニューで、「プロジェクトに保管」をクリックします。追加したビジネス・オブジェクト定義に指定されたサポートを含む、変更されたコネクター定義が、System Manager の ICL (Integration Component Library) プロジェクトに保管されます。

サポートされるリストからビジネス・オブジェクトを削除する場合は、以下の手順を実行します。

1. ビジネス・オブジェクト・フィールドを選択するため、そのビジネス・オブジェクトの左側の番号をクリックします。
2. 「Connector Configurator」ウィンドウの「編集」メニューから、「行を削除」をクリックします。リスト表示からビジネス・オブジェクトが除去されます。
3. 「ファイル」メニューから、「プロジェクトの保管」をクリックします。

サポートされるリストからビジネス・オブジェクトを削除すると、コネクタ定義が変更され、削除されたビジネス・オブジェクトはコネクタのこの実装で使用不可になります。コネクタのコードに影響したり、そのビジネス・オブジェクト定義そのものが System Manager から削除されることはありません。

エージェント・サポート: ビジネス・オブジェクトがエージェント・サポートを備えている場合、システムは、コネクタ・エージェントを介してアプリケーションにデータを配布する際にそのビジネス・オブジェクトの使用を試みます。

一般に、コネクタのアプリケーション固有ビジネス・オブジェクトは、そのコネクタのエージェントによってサポートされますが、汎用ビジネス・オブジェクトはサポートされません。

ビジネス・オブジェクトがコネクタ・エージェントによってサポートされるよう指定するには、「エージェント・サポート」ボックスにチェックマークを付けます。「Connector Configurator」ウィンドウでは「エージェント・サポート」の選択の妥当性は検査されません。

最大トランザクション・レベル: コネクタの最大トランザクション・レベルは、そのコネクタがサポートする最大のトランザクション・レベルです。

ほとんどのコネクタの場合、選択可能な項目は「最大限の努力」のみです。

トランザクション・レベルの変更を有効にするには、サーバーを再始動する必要があります。

ご使用のブローカーが WebSphere Message Broker の場合

スタンドアロン・モードで作業している (System Manager に接続していない) 場合、手動でビジネス・オブジェクト名を入力する必要があります。

System Manager を実行している場合、「サポートされているビジネス・オブジェクト」タブの「ビジネス・オブジェクト名」列の下にある空のボックスを選択できます。コンボ・ボックスが表示され、コネクタが属する統合コンポーネント・ライブラリー・プロジェクトから選択可能なビジネス・オブジェクトのリストが示されます。リストから必要なビジネス・オブジェクトを選択します。

「メッセージ・セット ID」は、WebSphere Business Integration Message Broker 5.0 のオプションのフィールドです。この ID が提供される場合、一意である必要はありません。ただし、WebSphere MQ Integrator および Integrator Broker 2.1 の場合は、一意の ID を提供する必要があります。

ご使用のブローカーが WAS の場合

使用するブローカー・タイプとして WebSphere Application Server を選択した場合、Connector Configurator にメッセージ・セット ID は必要ありません。「サポートされているビジネス・オブジェクト」タブには、サポートされるビジネス・オブジェクトの「ビジネス・オブジェクト名」列のみが表示されます。

スタンドアロン・モードで作業している (System Manager に接続していない) 場合、手動でビジネス・オブジェクト名を入力する必要があります。

System Manager を実行している場合、「サポートされているビジネス・オブジェクト」タブの「ビジネス・オブジェクト名」列の下にある空のボックスを選択できます。コンボ・ボックスが表示され、コネクターが属する統合コンポーネント・ライブラリー・プロジェクトから選択可能なビジネス・オブジェクトのリストが示されます。このリストから必要なビジネス・オブジェクトを選択します。

関係付けられたマップ (ICS のみ)

各コネクターは、現在 WebSphere InterChange Server でアクティブなビジネス・オブジェクト定義、およびそれらの関連付けられたマップのリストをサポートします。このリストは、「関連付けられたマップ」タブを選択すると表示されます。

ビジネス・オブジェクトのリストには、エージェントでサポートされるアプリケーション固有のビジネス・オブジェクトと、コントローラーがサブスクリプション・コラボレーションに送信する、対応する汎用オブジェクトが含まれます。マップの関連によって、アプリケーション固有のビジネス・オブジェクトを汎用ビジネス・オブジェクトに変換したり、汎用ビジネス・オブジェクトをアプリケーション固有のビジネス・オブジェクトに変換したりするとき、どのマップを使用するかが決定されます。

特定のソースおよび宛先ビジネス・オブジェクトについて一意的に定義されたマップを使用する場合、表示を開くと、マップは常にそれらの該当するビジネス・オブジェクトに関連付けられます。ユーザーがそれらを変更する必要はありません (変更できません)。

サポートされるビジネス・オブジェクトで使用可能なマップが複数ある場合は、そのビジネス・オブジェクトを、使用する必要のあるマップに明示的にバインドすることが必要になります。

「関連付けられたマップ」タブには以下のフィールドが表示されます。

- **ビジネス・オブジェクト名**

これらは、「サポートされているビジネス・オブジェクト」タブで指定した、このコネクターでサポートされるビジネス・オブジェクトです。「サポートされているビジネス・オブジェクト」タブでビジネス・オブジェクトを追加指定した場合、その内容は、「Connector Configurator」ウィンドウの「ファイル」メニューから「プロジェクトに保管」を選択して、変更を保管した後に、このリストに反映されます。

- **関連付けられたマップ**

この表示には、コネクターの、サポートされるビジネス・オブジェクトでの使用のためにシステムにインストールされたすべてのマップが示されます。各マップのソース・ビジネス・オブジェクトは、「**ビジネス・オブジェクト名**」表示でマップ名の左側に表示されます。

- **明示的**

場合によっては、関連付けられたマップを明示的にバインドすることが必要になります。

明示的バインディングが必要なのは、特定のサポートされるビジネス・オブジェクトに複数のマップが存在する場合のみです。ICS は、ブート時、各コネクターでサポートされるそれぞれのビジネス・オブジェクトにマップを自動的にバインドしようとします。複数のマップでその入力データとして同一のビジネス・オブジェクトが使用されている場合、サーバーは、他のマップのスーパーセットである 1 つのマップを見つけて、バインドしようとします。

他のマップのスーパーセットであるマップがないと、サーバーは、ビジネス・オブジェクトを単一のマップにバインドすることができないため、バインディングを明示的に設定することが必要になります。

以下の手順を実行して、マップを明示的にバインドします。

1. 「**明示的 (Explicit)**」列で、バインドするマップのチェック・ボックスにチェックマークを付けます。
2. ビジネス・オブジェクトに関連付けるマップを選択します。
3. 「Connector Configurator」ウィンドウの「**ファイル**」メニューで、「**プロジェクトに保管**」をクリックします。
4. プロジェクトを ICS に配置します。
5. 変更を有効にするため、サーバーをリブートします。

リソース (ICS)

「リソース」タブでは、コネクター・エージェントが、コネクター・エージェント並列処理を使用して同時に複数のプロセスを処理するかどうか、またどの程度処理するかを決定する値を設定できます。

すべてのコネクターがこの機能をサポートしているわけではありません。複数のプロセスを使用するよりも複数のスレッドを使用する方が通常は効率的であるため、Java でマルチスレッドとして設計されたコネクター・エージェントを実行している場合、この機能を使用することはお勧めできません。

メッセージング (ICS)

メッセージング・プロパティは、DeliveryTransport 標準プロパティの値として MQ を設定し、ブローカー・タイプとして ICS を設定した場合にのみ、使用可能です。これらのプロパティは、コネクターによるキューの使用方法に影響します。

トレース/ログ・ファイル値の設定

コネクタ構成ファイルまたはコネクタ定義ファイルを開くと、Connector Configurator は、そのファイルのログおよびトレースの値をデフォルト値として使用します。Connector Configurator 内でこれらの値を変更できます。

ログとトレースの値を変更するには、以下の手順を実行します。

1. 「トレース/ログ・ファイル」タブをクリックします。
2. ログとトレースのどちらでも、以下のいずれかまたは両方へのメッセージの書き込みを選択できます。

- コンソールに (STDOUT):
ログ・メッセージまたはトレース・メッセージを STDOUT 表示に書き込みます。

注: STDOUT オプションは、Windows プラットフォームで実行しているコネクタの「トレース/ログ・ファイル」タブでのみ使用できます。

- ファイルに:
ログ・メッセージまたはトレース・メッセージを指定されたファイルに書き込みます。ファイルを指定するには、ディレクトリー・ボタン (省略符号) をクリックし、指定する格納場所へ移動し、ファイル名を指定し、「保管」をクリックします。ログ・メッセージまたはトレース・メッセージは、指定した場所の指定したファイルに書き込まれます。

注: ログ・ファイルとトレース・ファイルはどちらも単純なテキスト・ファイルです。任意のファイル拡張子を使用してこれらのファイル名を設定できます。ただし、トレース・ファイルの場合、拡張子として .trc ではなく .trace を使用することをお勧めします。これは、システム内に存在する可能性がある他のファイルとの混同を避けるためです。ログ・ファイルの場合、通常使用されるファイル拡張子は .log および .txt です。

データ・ハンドラー

データ・ハンドラー・セクションの構成が使用可能となるのは、DeliveryTransport の値に JMS を、また ContainerManagedEvents の値に JMS を指定した場合のみです。すべてのアダプターでデータ・ハンドラーを使用できるわけではありません。

これらのプロパティに使用する値については、付録 A の『コネクタの標準構成プロパティ』の ContainerManagedEvents の下の説明を参照してください。その他の詳細は、「コネクタ開発ガイド (C++ 用)」または「コネクタ開発ガイド (Java 用)」を参照してください。

構成ファイルの保管

コネクタの構成が完了したら、コネクタ構成ファイルを保管します。Connector Configurator では、構成中に選択したブローカー・モードでファイルを保管します。Connector Configurator のタイトル・バーには現在のブローカー・モード (ICS、WMQI、または WAS) が常に表示されます。

ファイルは XML 文書として保管されます。XML 文書は次の 3 通りの方法で保管できます。

- System Manager から、統合コンポーネント・ライブラリーに *.con 拡張子付きファイルとして保管します。
- 指定したディレクトリーに保管します。
- スタンドアロン・モードで、ディレクトリー・フォルダーに *.cfg 拡張子付きファイルとして保管します。デフォルトでは、このファイルは %WebSphereAdapters%bin%Data%App に保管されます。
- WebSphere Application Server プロジェクトをセットアップしている場合には、このファイルを WebSphere Application Server プロジェクトに保管することもできます。

System Manager でのプロジェクトの使用法、および配置の詳細については、以下のインプリメンテーション・ガイドおよび実装ガイドを参照してください。

- ICS: 「*WebSphere InterChange Server* インプリメンテーション・ガイド」
- WebSphere Message Brokers: 「*WebSphere Message Brokers* 使用アダプター・インプリメンテーション・ガイド」
- WAS: 「*アダプター実装ガイド (WebSphere Application Server)*」

構成ファイルの変更

既存の構成ファイルの統合ブローカー設定を変更できます。これにより、他のブローカーで使用する構成ファイルを新規に作成するときに、このファイルをテンプレートとして使用できます。

注: 統合ブローカーを切り替える場合には、ブローカー・モード・プロパティーと同様に他の構成プロパティーも変更する必要があります。

既存の構成ファイルでのブローカーの選択を変更するには、以下の手順を実行します (オプション)。

- Connector Configurator で既存の構成ファイルを開きます。
- 「標準のプロパティー」タブを選択します。
- 「標準のプロパティー」タブの「**BrokerType**」フィールドで、ご使用のブローカーに合った値を選択します。
 現行値を変更すると、プロパティー画面の利用可能なタブおよびフィールド選択がただちに變更され、選択した新規ブローカーに適したタブとフィールドのみが表示されます。

構成の完了

コネクターの構成ファイルを作成し、そのファイルを変更した後で、コネクターの始動時にコネクターが構成ファイルの位置を特定できるかどうかを確認してください。

これを行うには、コネクターが使用する始動ファイルを開き、コネクター構成ファイルに使用されている格納場所とファイル名が、ファイルに対して指定した名前およびファイルを格納したディレクトリーまたはパスと正確に一致しているかどうかを検証します。

グローバル化環境における Connector Configurator の使用

Connector Configurator はグローバル化され、構成ファイルと統合ブローカー間の文字変換を処理できます。Connector Configurator では、ネイティブなエンコード方式を使用しています。構成ファイルに書き込む場合は UTF-8 エンコード方式を使用します。

Connector Configurator は、以下の場所で英語以外の文字をサポートします。

- すべての値のフィールド
- ログ・ファイルおよびトレース・ファイル・パス (「トレース/ログ・ファイル」タブで指定)

CharacterEncoding および Locale 標準構成プロパティのドロップ・リストに表示されるのは、サポートされる値の一部のみです。ドロップ・リストに、サポートされる他の値を追加するには、製品ディレクトリーの %Data%Std%stdConnProps.xml ファイルを手動で変更する必要があります。

例えば、Locale プロパティの値のリストにロケール en_GB を追加するには、stdConnProps.xml ファイルを開き、以下に太文字で示した行を追加してください。

```
<Property name="Locale"
isRequired="true"
updateMethod="component restart">
  <ValidType>String</ValidType>
  <ValidValues>
    <Value>ja_JP</Value>
    <Value>ko_KR</Value>
    <Value>zh_CN</Value>
    <Value>zh_TW</Value>
    <Value>fr_FR</Value>
    <Value>de_DE</Value>
    <Value>it_IT</Value>
    <Value>es_ES</Value>
    <Value>pt_BR</Value>
    <Value>en_US</Value>
    <Value>en_GB</Value>
    <DefaultValue>en_US</DefaultValue>
  </ValidValues>
</Property>
```

特記事項

本書に記載の製品、サービス、または機能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、および機能については、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、機能的に同等の製品、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 以外の製品とプログラムの操作またはサービスの評価および検証は、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。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-0032
東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
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。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更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 Burlingame Laboratory Director
IBM Burlingame Laboratory
577 Airport Blvd., Suite 800
Burlingame, CA 94010
U.S.A

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せんが、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。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

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。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。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。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

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、サンプル・プログラムが書かれているオペレーティング・プラットフォーム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のめかしたり、保証することはできません。

プログラミング・インターフェース情報

プログラミング・インターフェース情報は、プログラムを使用してアプリケーション・ソフトウェアを作成する際に役立ちます。

一般使用プログラミング・インターフェースにより、お客様はこのプログラム・ツール・サービスを含むアプリケーション・ソフトウェアを書くことができます。

ただし、この情報には、診断、修正、および調整情報が含まれている場合があります。診断、修正、調整情報は、お客様のアプリケーション・ソフトウェアのデバッグ支援のために提供されています。

警告: 診断、修正、調整情報は、変更される場合がありますので、プログラミング・インターフェースとしては使用しないでください。

商標

以下は、IBM Corporation の商標です。

IBM
IBM ロゴ
AIX
CrossWorlds
DB2
DB2 Universal Database
Lotus
Lotus Domino
Lotus Notes
MQIntegrator
MQSeries
Tivoli
WebSphere

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

MMX、Pentium および ProShare は、Intel 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。

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です。



IBM WebSphere Business Integration Adapter FrameWork V2.4.0

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]

アーカイブ、イベントの 7
アーカイブ・ストア、作成 22
アーカイブ・フォルダー 3
アーキテクチャー、アダプターの 1
浅いイベント登録 26
値、イベント状況 7
値、Archive Store フォルダのフィールド 22
アダプター、停止 31
アダプター、複数インスタンス 32
アダプター環境 9
アダプターのアーキテクチャー 1
アダプターのインストール 10, 13
アダプターの環境 9
アダプターの停止 31
アプリケーション、COM+ 18
アプリケーション固有の構成プロパティ 23
アプリケーション固有の情報、ビジネス・オブジェクト用 38
アプリケーション固有のテキスト、単純属性用 39
アプリケーション固有のテキスト、動詞用 39
アプリケーション固有のテキスト、ビジネス・オブジェクト用 38
イベント、数 6
イベント、登録 25
イベント状況 7
イベント処理 6
イベント通知 7
イベント定義 6
イベント登録、浅い 26
イベント登録、エラー 43
イベント登録、ストア全体 28
イベント登録、深い 27
イベント登録、Exchange Explorer による 25
イベント登録、RegEvent.vbs による 28
イベントのアーカイブ 7
イベントのリカバリー 8
イベント・シンク 17
イベント・デリバリー、保証 8

イベント・フォルダー 3
イベント・モード処理 6
イベント・ログ、Windows 44
インストール、Event Listener 14
インストール、アダプター 10, 13
インストール、コネクタ 13
エラー 41
エラー、イベント登録 43
エラー処理 41
エラー処理、Event Listener 42

[カ行]

概要 1
関連文書 v
規則、表記上の vi
構成 17
構成、Event Listener 17
構成、Windows イベント・ビューアー 21
構成タスク 17
構成タスク、コネクタ用 22
構成ファイル、変更 20
構成プロパティ、コネクタ 23
構成プロパティ、コネクタ標準 47
互換性、統合ブローカーとの 9
コネクタ、処理 4
コネクタ構成プロパティ 23
コネクタのインストール 13
コネクタの停止 31
コネクタ・プロパティ 23
コネクタ・プロパティ構成 67
コネクタ・プロパティの構成 67

[サ行]

作成、アーカイブ・ストア 22
サポート、トランザクション 9
サポートされるソフトウェア 9
サポートされる統合ブローカー 9
サンプル・ビジネス・オブジェクト 14
システム障害およびイベントのリカバリー 8
ストア 3
ストア全体の登録の作成 28
属性プロパティ 37
ソフトウェア、必要なソフトウェア
サポートされる 9

[タ行]

単純属性 39
通知、イベントの 7
データ、ロケール依存 11
データ・ストア 3
データ・ストレージおよびアクセス 3
停止、Event Listener 31
動詞 39
登録、イベント 25
登録、ストア全体のイベント 28
登録、深いイベント 27
トラブルシューティング 41
トランザクション・サポート 9
トランザクション・ロギング 9
トレース、Event Listener による 44
トレース・メッセージ 43

[ハ行]

ビジネス・オブジェクト 35
ビジネス・オブジェクト、アプリケーション固有テキスト 38
ビジネス・オブジェクト、サポートされる 36
ビジネス・オブジェクト、サンプル 14
ビジネス・オブジェクト、定義 5
ビジネス・オブジェクト構造 36
ビジネス・オブジェクト定義、定義 36
必要なソフトウェア 9
表記上の規則 vi
標準構成プロパティ 23, 47
ファイル構造 - UNIX 14
ファイル構造 - Windows 13
フィルター操作 21
フォルダー、イベント・ストア内 3
深いイベント登録 27
複数アダプターの始動 32
複数インスタンス、アダプター 32
プロパティ、ビジネス・オブジェクト属性 37
変更、構成ファイル 20
保証付きイベント・デリバリー 8

[マ行]

メタデータ、定義 35
メッセージ、トレース 43
メッセージ・クラス、Outlook 38

[ヤ行]

ユーザー・アカウント、ユーザー 17
要求モード処理 4

[ラ行]

リカバリー、システム障害後のイベント・
リカバリー 8
ロケール依存データ 11

[数字]

2 バイト文字セット、サポート 11

A

AppointmentItem 3, 36
AppointmentItem オブジェクト 3
ArchiveProcessed コネクタ・プロパティ
ー 7

C

COM+ 17
COM+ アプリケーション 18
Connector Configurator、使用 67
ContactItem 3, 36
ContactItem オブジェクト 3
ContentClass 38
CwAgentUserGUID 20
CwAgentUsername 20
CwEventID 20
CwEventUsername 20

E

Event Listener 2
Event Listener の始動 30
Event Listener の登録 18
Event Listener、インストール 14
Event Listener、エラー 42
Event Listener、概要 1
Event Listener、構成 17
Event Listener、始動 30
Event Listener、停止 31
Event Listener、登録 18
Event Listener、トレースを使用 44
EventsTraceLevel 20
Exchange Explorer、イベント登録の作成
25

G

GED (保証付きイベント・デリバリー) 8

M

MailItem 36
MailItem オブジェクト 3
MailItem メッセージ 3

O

Outlook メッセージ・クラス 38

P

PollQuantity 6

R

Recipient 36
RecurrencePattern 37
RegEvent.vbs、イベント登録の作成用 28
Regsvr32 ユーティリティー 18

T

TaskItem 3, 36

U

URL v

W

Web サイト v
WebDAV 3
Windows イベント・ビューアー、構成
21
Windows イベント・ログ 44
Windows ユーザー・アカウント、イベン
ト・シンク用 17