

Integration Designer
バージョン 7.5

**Hello World Part 2: サービス・コンポーネントおよび
Web インターフェース**

IBM

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、31 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。

このエディションは、IBM Integration Designer のバージョン 7.5 に適用されます。

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： Integration Designer
Version 7.5
Version 7 Release 5
Hello World Part 2: Service Components and Web Interfaces

発行： 日本アイ・ビー・エム株式会社

担当： トランスレーション・オペレーション・センター

第1刷 2011.6

© Copyright IBM Corporation 2005, 2011.

目次

第 1 章 概要	1	第 3 章 サンプルの実行	25
概要	1	ソリューションの探索	25
第 2 章 ユーザーによるビルド	7	モジュールのデプロイ	27
「Hello World パート 1」サンプルのインポート	7	サーバーからのモジュールの除去	27
統合ソリューションの作成	8	第 4 章 インポート	29
新しいインターフェースの作成	10	特記事項	31
モジュールの作成	12	ご利用条件	35
モジュールへのデータの取り込み	12		
ヒューマン・タスク実装の作成	14		
ビジネス・プロセス実装の作成	17		

第 1 章 概要

このサンプルでは、IBM® Integration Designer を使用して、既存の Web サービスに対するメディエーション呼び出しを作成し、デプロイし、実行する方法を示します。

このサンプルでは、以下のアクティビティの実行方法を説明します。

- ワークベンチをナビゲートします。
- 共有成果物用のライブラリー・プロジェクトを作成します。
- 統合ロジック用のメディエーション・モジュール・プロジェクトを作成します。
- Web サービスを呼び出します。
- メディエーション・フローでの Web サービス呼び出しのメディエーションにより、そのインターフェースを別のインターフェースにマップします。
- 異なるインターフェースを持つ新しいサービスとして、メディエーション・フローを公開します。
- メディエーション・モジュールをデプロイします。
- メディエーション・モジュールを実行およびテストします。

このサンプルは、WebSphere® Enterprise Service Bus または IBM Process Server で実行されます。

概要

このチュートリアルを使用して作業するときは、フラット・ファイルやアダプターのほか、IBM Integration Designer で作成されたアプリケーションとの間でアダプターを使用して行うデータ処理に関連して、いくつかの概念を理解しておく必要があります。また、作成するアプリケーションについての高度な理解も必要です。

このセクションでは、以下の概念について説明します。

- フラット・ファイル
- WebSphere Adapter for Flat Files
- インバウンド処理
- ファイル分割
- レコード区切り文字
- サイズによる分割
- カスタム・データ・バインディング
- アウトバウンド処理
- このチュートリアルでのアプリケーションの概要

フラット・ファイルとは

「フラット・ファイル」という用語は、ローカル・ファイル・システムに保管されているすべてのファイルを指し、構造化データベース内のファイルなどのより複雑なファイル・セットと対立する用語です。フラット・ファイル内のデータ・レコードは順次に保管されるのが一般的で、データベース・ストレージに見られるような索引、キー、リレーションシップなどのメタデータを伴っていません。Adapter for Flat Files を使

用すると、すべてのタイプのファイルを読み書きできます。2 つの一般的なデータ・フォーマットは、XML レコードとコンマ区切り値 (CSV) レコードです。次の例は、各フォーマットのレコードを示しています。

XML フォーマット

```
<?xml version="1.0" encoding="UTF-8"?>
<customer>
  <title>Mr</title>
  <name>Smith</name>
  <city>Ottawa</city>
  <state>ON</state>
</customer>
<customer>
  <title>Mrs</title>
  <name>Jones</name>
  <city>Winnipeg</city>
  <state>MB</state>
</customer>
```

CSV フォーマット

```
Mr,Smith,Ottawa,ON
Mrs,Jones,Winnipeg,MB
```

WebSphere Adapter for Flat Files とは

あるディレクトリーにファイル・セットを出力する外部システムがあるとします。例えば、注文情報を格納したテキスト・ファイルを生成する注文処理システムなどが考えられます。それらの情報はオフピーク時間にバッチ・モードで処理されます。この場合は、WebSphere Adapter for Flat Files を IBM Integration Designer と一緒に使用して、それらのファイルの作成と管理に役立てます。

ご使用のアプリケーションは、Adapter for Flat Files を使用してファイルの作成と管理を行うか、ディレクトリーをモニターし、そのディレクトリーからファイルを読み取ることができます。Adapter for Flat Files に関連したサービスを作成するときは、次の 2 つのオプションがあり、どちらのオプションも製品のインフォメーション・センターに文書化されています。

- シンプル外部サービス・ウィザード: このウィザードを使用して、ユーザーは WebSphere Adapter for Flat Files が読み書きを行うディレクトリーとそれらのディレクトリーのデータ・フォーマットを指定し、サービスを作成します。その後、このウィザードによって、そのサービスが生成されます。多くのユーザーにとって、サービスを作成するには、この迅速で簡単な方法で十分です。このチュートリアルでは、この方法を使用します。
- 拡張サービス・ウィザード: このウィザードを使用して、ユーザーはより長時間の包括的なプロセスによるサービスを作成します。ユーザーによっては、このさらに詳細な方法の方が適している場合があります。最初に、必要となるアダプターのタイプを指定し、次に WebSphere Adapter for Flat Files が読み書きを行うディレクトリーと、それらのディレクトリーのデータ・フォーマットを指定します。その後、拡張セクションのフィールドがある後続の各ページを使用して、生成されるサービスをさらに細かく制御します。

インバウンドおよびアウトバウンド処理とは

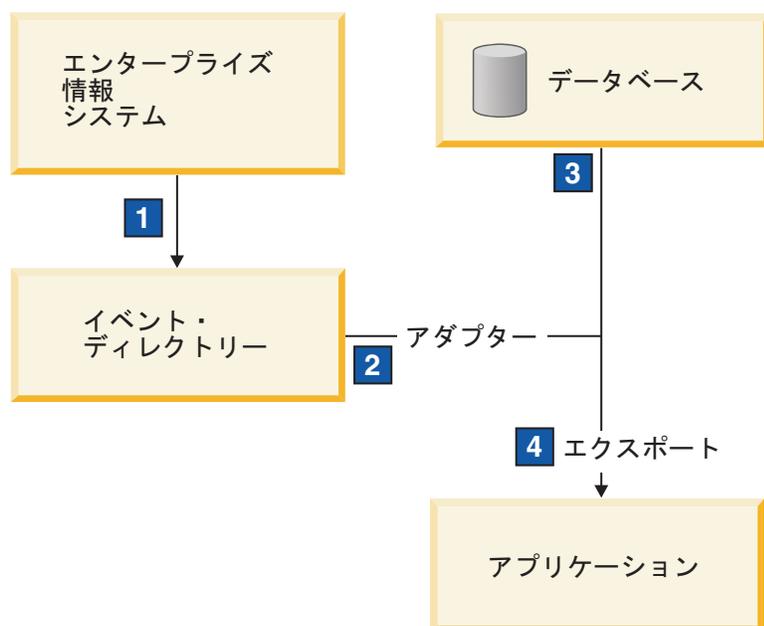
インバウンド処理 は WebSphere Adapter for Flat Files がファイル・システムをモニターし、新規ファイルを読み取り、アプリケーション内の操作ヘデータを送信する操作モードです。

アウトバウンド処理は、WebSphere Adapter for Flat Files が (アプリケーションのコンポーネントから) 要求を受信し、ファイル操作を実行し、該当する場合は結果を呼び出し元に戻す操作モードです。操作の例としては、ファイルの作成、ファイルへの書き込み、特定のファイルが存在するかどうかの検査が含まれます。

インバウンド処理

インバウンド処理のとき、Adapter for Flat Files はイベント・ディレクトリーによって生成されるイベント (例えば、イベント・ディレクトリーにファイルが置かれます) を listen します。イベントとは、イベント・ディレクトリー内でどのような変更が発生したかを記録したレコードのことです。Adapter for Flat Files が新規ファイルの有無をモニターするディレクトリーは、イベント・ディレクトリーと呼ばれます。

次の図は、インバウンド処理のときに発生する 4 つのステップを示しています。



1. 外部システムが、ファイルをイベント・ディレクトリーへ出力します。
2. Adapter for Flat Files がイベント・ディレクトリーにファイルがあるかどうかポーリングを行い、それらのファイルからのデータをイベントに変換します。特定のタイプのファイルの有無、または特定の期間に作成されたファイルの有無についてポーリングを行うよう、アダプターを構成できます。
3. イベントは、一時的にイベント・ストアに置かれます。これはデータベースであるか、イベント・テーブルのメモリー内表現です。データベースを使用した場合、仮にイベントが作成された後にサーバーがダウンしても、イベントがアプリケーションによる処理の前に失われることはありません。これは、保証付き配信と呼ばれます。メモリー内テーブルを使用するとイベントの処理が速くなりますが、イベント・リカバリー機能は失われます。
4. Adapter for Flat Files はイベント・ストアからイベントを取得し、外部サービス・ウィザードによって作成されてエクスポートされたインバウンド・インターフェースを介して、各イベントをビジネス・オブジェクトの形で受け渡します。

このデータ取得方式 (つまり、データをビジネス・オブジェクトの形で受け渡す方式) は、非パススルー、または *Data Transformation Framework (DTF)* と呼ばれ、構造化データに対して機能します。

アプリケーションがデータ・ファイルのフォーマットを認識しない場合は、Adapter for Flat Files をパススルー・モードで稼働するよう構成し、非構造化データに対して機能させることができます。その場合、イベントはビジネス・オブジェクトに変換されません。

ファイル分割

ファイル分割機能は、取得したいファイルが大きい場合、またはそれぞれのファイルに複数のレコードが入っている場合に使用します。区切り文字または固定サイズ値に基づいて、ファイルをより小さなチャンクに分割でき、ファイルの各パーツを並列で処理できます。各チャンクは別々のイベントと見なされ、エクスポートされたインバウンド・インターフェースを通じてアプリケーション内の操作へ個別に送信されます。

レコード区切り文字

一般に、入力はファイルごとに単一のレコードとして保管されます。しかし、入力ファイルに複数のレコードが含まれているときは、多くの場合、区切り文字によってファイル内のレコードが分離されます。区切り文字は任意のテキスト・ストリングで、通常は文字の組み合わせの後に `¥r¥n` (プラットフォームに依存する改行文字) を付けます。アダプターは、区切り文字を含んでいるファイルの読み取りと書き込みの両方ができます。

前述した XML フォーマットおよび CSV フォーマットの例を使用すると、次の例では、#### の後に改行を付けたストリングを XML ファイル内で区切り文字として使用しています。CSV ファイルでは、改行を区切り文字としています。

を区切り文字とした XML フォーマット

```
<?xml version="1.0" encoding="UTF-8"?>
<customer>
  <title>Mr</title>
  <name>Smith</name>
  <city>Ottawa</city>
  <state>ON</state>
</customer>
####
<customer>
  <title>Mrs</title>
  <name>Jones</name>
  <city>Winnipeg</city>
  <state>MB</state>
</customer>
####
```

改行を区切り文字とした CSV フォーマット

```
Mr,Smith,Ottawa,ON
Mrs,Jones,Winnipeg,MB
```

サイズによる分割

サイズによる分割フィーチャーは、ファイルをより小さなチャンクに分割し、アプリケーション内の操作へ 1 つずつ転送するために使用される点で、区切り文字による分割によく似ています。このフィーチャーはパススルー・シナリオで非構造化データに使用します。この非構造化データはビジネス・オブジェクトに変換されないからです。アダプターでの分割条件は、バイト数として指定します。アダプターはファイルをそのバイト・サイズのチャンク (イベント) として読み取ります。各チャンクは分割条件で定義されたサイズになりますが、最後のチャンクだけは例外で、それより小さくなる場合があります。

カスタム・データ・バインディング

フラット・ファイル内のデータは、前述した XML フォーマットや CSV フォーマットなど、さまざまなフォーマットになる可能性があります。それ以外のフォーマットとしては、名前と値のペア、タブ区切り、固定幅などのフォーマットがあります。データ・ハンドラー は、データ・ファイル内のフォーマットからビジネス・オブジェクトの属性へのマッピングを行います。

Adapter for Flat Files のデフォルトのデータ・バインディングを使用すると、ファイルを XML フォーマットとの間で変換することができます。それ以外のフォーマットでは、ファイル・フォーマットとビジネス・オブジェクトの間のマッピングを定義したカスタム・データ・ハンドラーを使用する必要があります。カスタム・データ・ハンドラーを実装するには、特定のフォーマットのデータを変換する Java クラスを作成し、アダプターを構成するときに、そのクラスを外部サービス・ウィザードに入力します。

カスタム・データ・ハンドラーについては、IBM Integration Designer の資料を参照してください。

アウトバウンド処理

ファイルの書き込みまたは変更を行うために、アプリケーションはアウトバウンド・インターフェースで定義された操作を使用します。アウトバウンド・インターフェースは、外部サービス・ウィザードを使用して作成します。以下のさまざまな操作タイプを使用して、非構造化データ (パススルー・モード) と構造化データ (非パススルー・モード) のどちらでも作成できます。

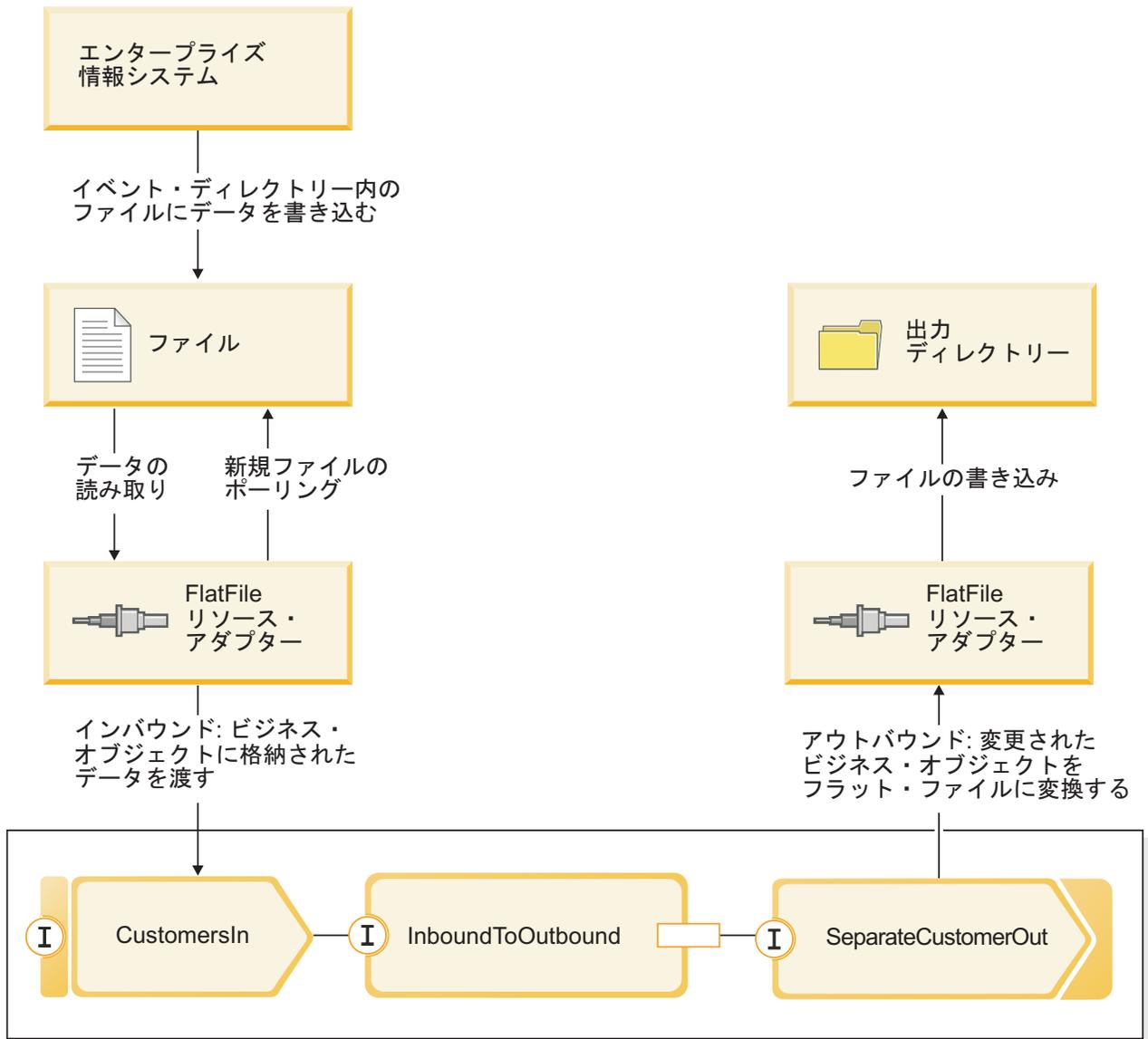
- 作成 - データを新規ファイルに保管するか、空のファイルを作成します。
- 追加 - 既存のファイルにデータを追加します。まだファイルが存在していなかった場合は、新規ファイルが作成されます。
- 上書き - 既存のファイルを新規データで上書きします。
- 削除 - 既存のファイルを削除します。
- 存在 - 特定のファイルが存在するかどうかを検査します。
- リスト - ディレクトリー内のファイルをリストします。
- 取得 - ファイルから内容を読み取ります。

シンプル外部サービス・ウィザードでは、作成操作と追加操作のみが使用されます。

このチュートリアルでのアプリケーションの概要

顧客レコード情報を収集するために、ファイル・システム内の特定のディレクトリーをモニターするアプリケーションを作成する場合を考えてみます。ファイルが作成されると、アダプターは指定されたファイル・スプリッター区切り文字を使用して、ファイルの内容をビジネス・オブジェクトに分割します。それぞれのビジネス・オブジェクトについてエクスポートが呼び出されます。エクスポートはメディエーション・フローへワイヤリングされており、そのフローはアダプターを呼び出して、レコードを出力ディレクトリー内の独自のファイルに書き込みます。

次の図は、このシナリオを示しています。



顧客レコードを含んでいるファイルがインバウンド・イベント・ディレクトリーに入っている場合、リソース・アダプターはそれらを取得し、顧客ビジネス・オブジェクトを作成し、モジュール内のエクスポートを呼び出します。このエクスポートはメディエーション・フローを開始します。メディエーション・フローはファイル・サービスを呼び出し、そのファイル・サービスは、既存のファイルの数に基づき、その名前を使用して、新しいファイル名を生成します。

第 2 章 ユーザーによるビルド

ユーザーは、このサンプルを自分自身でビルドすることができます。

学習目標

IBM Integration Designer でウィザードとエディターを使用してこの実地サンプルをビルドする際は、以下のことを行います。

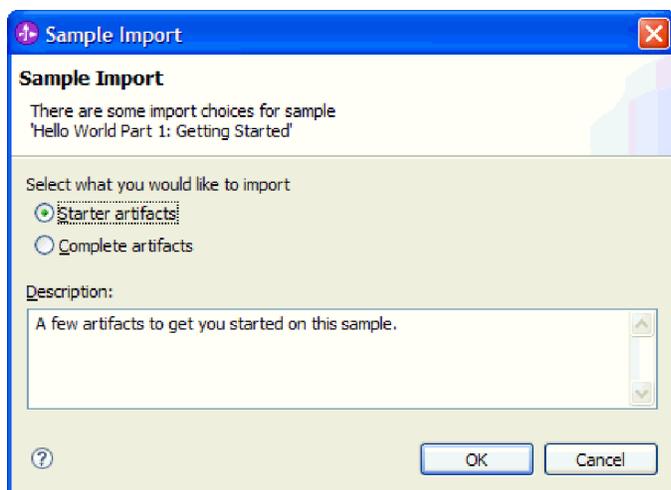
- ディレクトリーを作成する
- 開発するサービスを格納するためのモジュールを作成する
- 開発時間を節約するために、ビジネス・オブジェクトをインポートする
- アダプター・パターン・ウィザードを使用して、インバウンド・サービスを作成する
- 同じウィザードを使用してアウトバウンド・サービスを作成する
- メディエーション・フローを作成します。
- モジュールのデプロイ
- モジュールのテスト

「Hello World パート 1」サンプルのインポート

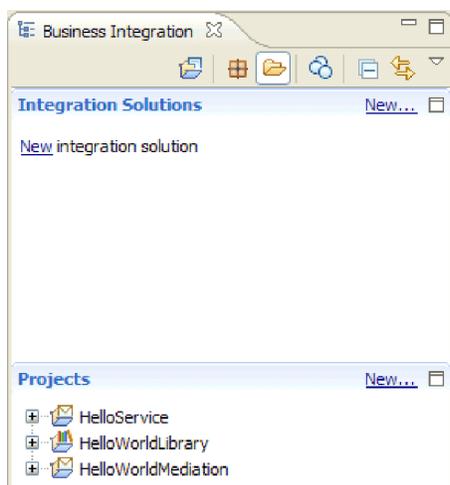
「Hello World パート 2」サンプル・アプリケーションをビルドするには、ワークスペース内で「Hello World パート 1」サンプル・アプリケーションが必要になります。パート 1 のサンプルをユーザー自身でビルドしなかった場合、しかもパート 1 のサンプルの既製バージョンをインポートしなかった場合は、この時点で、使用するワークスペースにインポートしてください。

既製の「Hello World パート 1」サンプルをワークスペースにインポートするには、以下の手順を実行します。

1. IBM Integration Designer で、「ヘルプ」>「サンプルおよびチュートリアル」>「**IBM Integration Designer 7.5**」を選択します。「サンプルおよびチュートリアル」ページが開きます。
2. 「**Hello World パート 2: サービス・コンポーネントおよび Web インターフェース**」セクションの下で、「インポート」リンクをクリックします。
3. 以下に示すように、インポート用の 2 つのオプションが表示されます。



4. 「開始者の成果物」を選択し、「OK」をクリックします。これで、「Hello World パート 2」サンプルをビルドするためにワークスペース内で必要となる「Hello World パート 1」サンプルがインポートされます。ここで、以下に示すように、「ビジネス・インテグレーション」ビューに 2 つのモジュール・プロジェクトと 1 つのライブラリー・プロジェクトが表示されます。

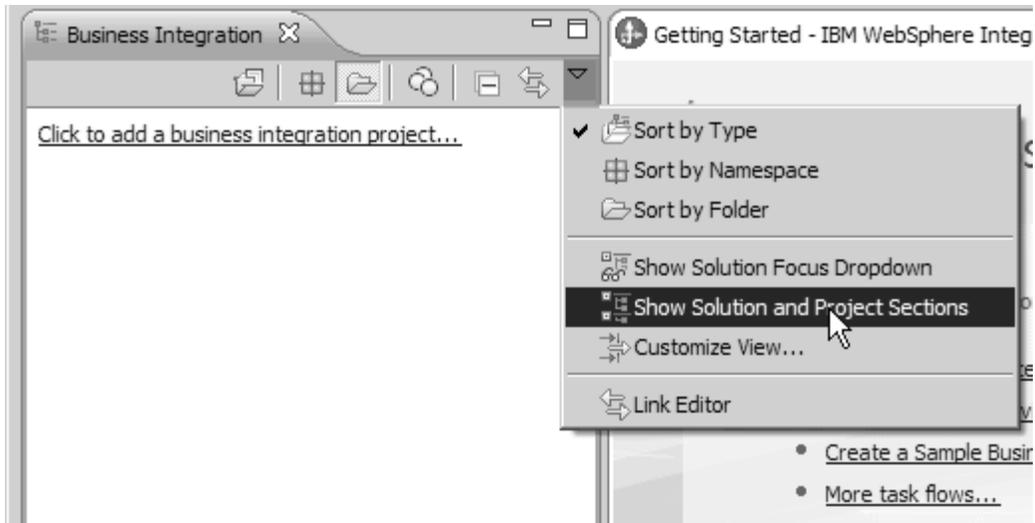


統合ソリューションの作成

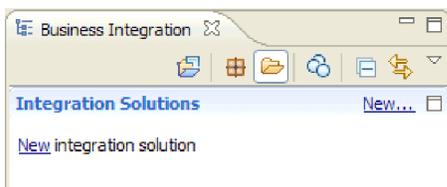
多くのプロジェクトがある場合は、統合ソリューションを使用して、関連したプロジェクトをグループ化すると便利です。統合ソリューションは、他のプロジェクトを参照するために使用される、単なる「スーパープロジェクト」です。統合ソリューションを使用すると、グループ化した複数の関連プロジェクトを、より簡単にテストしたり、チーム環境で処理したりすることができるようになります。IBM Integration Designer には、統合ソリューション内のプロジェクト間の関係を視覚化するために役立つ、統合ソリューション・エディターが用意されています。

統合ソリューションを作成するには、以下の手順を実行します。

1. 下矢印をクリックして「ビジネス・インテグレーション」ビューの「統合ソリューション」セクションを開き、ビューのメニューを表示して、「ソリューションおよびプロジェクト・セクションの表示」を選択します。

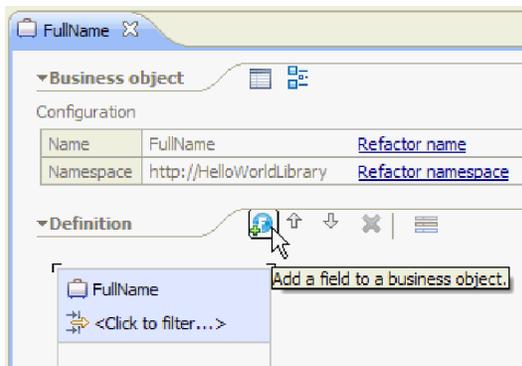


2. 「ビジネス・インテグレーション」ビューの「統合ソリューション」セクションで、以下の図に示されている「新規」リンクをクリックします。



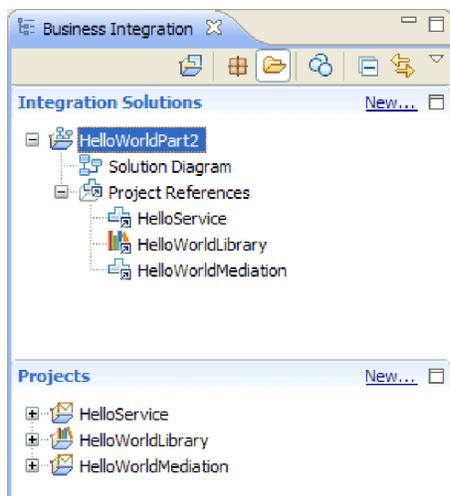
「新規統合ソリューション」ウィザードが開きます。

3. 「統合ソリューション名」フィールドに HelloWorldPart2 と入力して、「次へ」をクリックします。
4. 以下の図に示すように、「すべて選択」ボタンをクリックして、モジュール・プロジェクトとライブラリー・プロジェクト (**HelloService**、**HelloWorldLibrary**、および **HelloWorldMediation**) を選択します。



リストに他のプロジェクトがあって、そのプロジェクトも選択されている場合は、名前の横にあるチェック・ボックスをクリアしてください。

5. 「終了」をクリックすると、新規統合ソリューションがワークスペースに追加されます。「ビジネス・インテグレーション」ビューの「統合ソリューション」セクションで、**HelloWorldPart2** 統合ソリューションを展開して、「プロジェクト参照」を展開します。以下の図に示すように、**HelloService**、**HelloWorldLibrary**、および **HelloWorldMediation** のすべてのプロジェクトは、統合ソリューションによって参照されています。



注: 必要に応じて、「ビジネス・インテグレーション」ビューの「統合ソリューション」セクションと「プロジェクト」セクションを分けているサッシをグラブして、上下にドラッグすることができます。

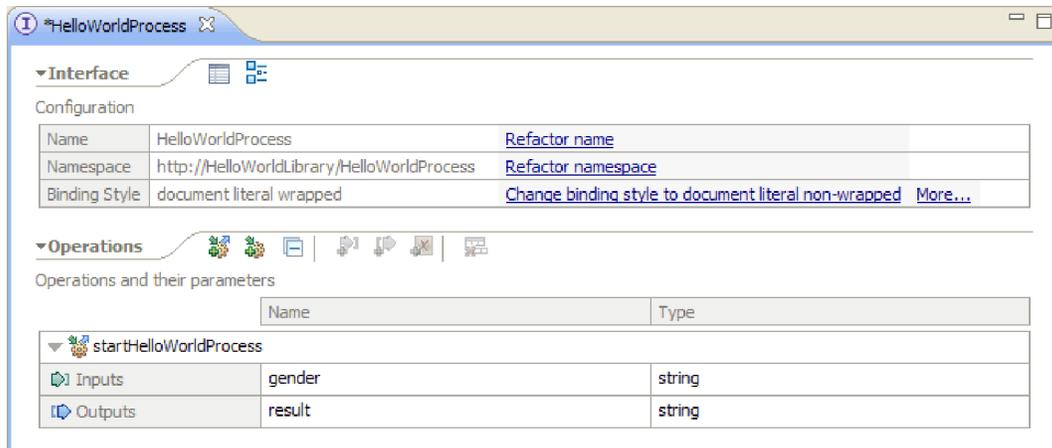
新しいインターフェースの作成

ビジネス・プロセスには、呼び出し元の性別のSTRINGを入力として取り、連結された最終結果STRINGを返す独自のインターフェースが必要です。

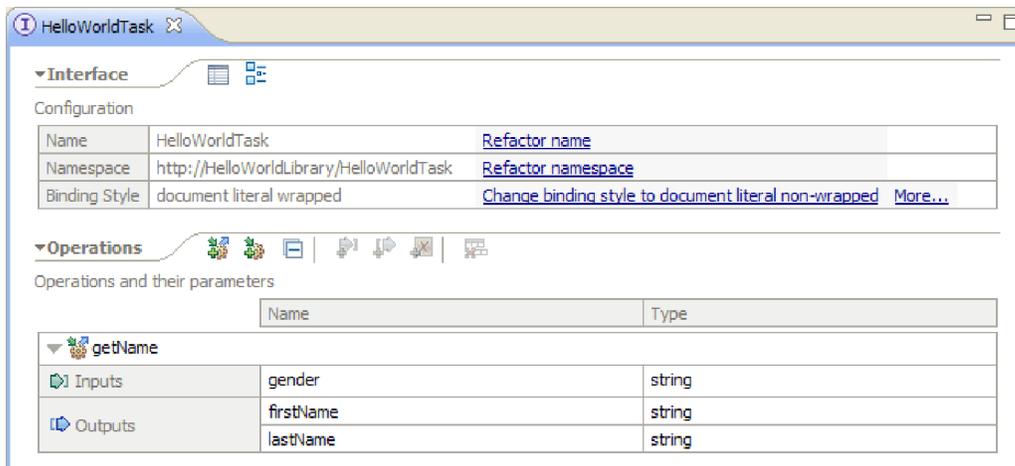
ヒューマン・タスクにも、独自のインターフェースが必要です。STRINGを入力として取り、要求されたタスクのユーザー・インターフェース・フォームにそのSTRINGを表示します。そこで、ユーザーに名と姓の入力を求めるプロンプトを出し、ヒューマン・タスクが完了すると、この名と姓を出力として返します。

既存の HelloWorldLibrary ライブラリー・プロジェクトで新しいインターフェースを 2 つ作成するには、以下の手順を実行します。

1. 以下の手順を実行して、ビジネス・プロセスのインターフェースを作成します。
 - a. 「ビジネス・インテグレーション」ビューの「プロジェクト」セクションで、「HelloWorldLibrary」を展開し、次に、「インターフェース」カテゴリを右クリックして、「新規」>「インターフェース」を選択します。「新規インターフェース」ウィザードが開きます。
 - b. 「名前」フィールドに HelloWorldProcess と入力して、「終了」をクリックします。インターフェース・エディターが開きます。
 - c. 「操作」セクションで「要求/応答操作の追加」アイコンをクリックして要求/応答操作を追加し、デフォルト名の **operation1** を **startHelloWorldProcess** に名前変更します。
 - d. デフォルトの入力名を **input1** から **gender** に変更し、デフォルトの出力名を **output1** から **result** に変更します。この時点で、インターフェースは次の図のようになります。



- e. **Ctrl+S** を押してインターフェースを保存し、インターフェース・エディターを閉じます。
2. 以下の手順を実行して、ビジネス・プロセスのインターフェースを作成します。
 - a. 「ビジネス・インテグレーション」ビューの「プロジェクト」セクションで、「HelloWorldLibrary」を展開し、次に、「インターフェース」カテゴリを右クリックして、「新規」>「インターフェース」を選択します。「新規インターフェース」ウィザードが開きます。
 - b. 「名前」フィールドに HelloWorldTask と入力して、「終了」をクリックします。インターフェース・エディターが開きます。
 - c. 「操作」セクションで「要求/応答操作の追加」アイコンをクリックして要求/応答操作を追加し、デフォルトの操作名 **operation1** を **getName** に名前変更します。
 - d. 「出力」行の任意の場所を右クリックして、「出力の追加」を選択します。新規出力が **output2** という名前で追加されます。
 - e. デフォルトの入力名を **input1** から **gender** に変更します。
 - f. デフォルトの出力名を **output1** から **firstName** に変更し、デフォルトの出力名を **output2** から **lastName** に変更します。この時点で、インターフェース・エディターは次の図のようになります。



- g. **Ctrl+S** を押してインターフェースを保存し、インターフェース・エディターを閉じます。

モジュールの作成

モジュールは、プロジェクト・フォルダーによく似ています。モジュールには、開発するサービスが格納されます。

ビジネス・インテグレーション・パースペクティブでモジュールを作成するには、以下の手順に従います。

1. メニューから、「ファイル」>「新規」>「モジュール」を選択します。「新規モジュール」ウィンドウが開きます。
2. 「モジュール名」フィールドに `SeparateCustomers` と入力します。「完了」をクリックします。
3. 「ビジネス・インテグレーション」ビューにモジュールが作成されます。

モジュール名 `SeparateCustomers` は、前のセクションで定義された入力ファイルに指定されている名前空間と、同じ値でなければならないことに注意してください。

モジュールへのデータの取り込み

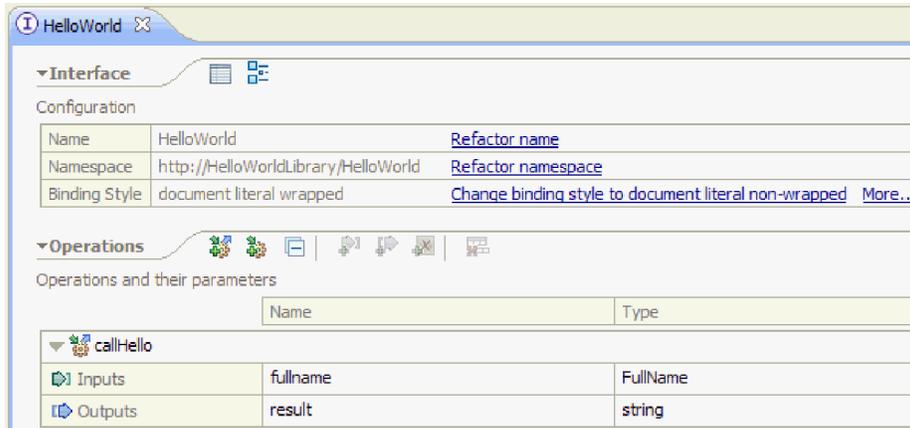
ここで、新しい `HelloWorldProcess` モジュール用のアセンブリ・エディターを使用して、ビジネス・プロセス・コンポーネント、ヒューマン・タスク・コンポーネント、およびインポート・コンポーネントを追加する必要があります。

モジュールを取り込むには、以下の手順を実行します。

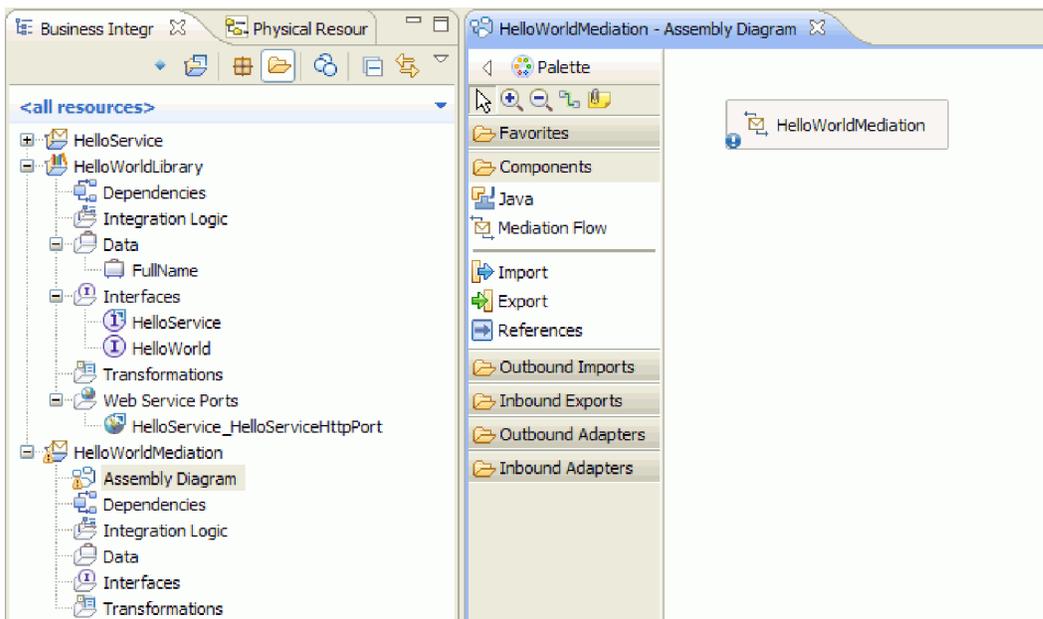
1. アセンブリ・エディターが `HelloWorldProcess` モジュール用に開いていない場合は、「ビジネス・インテグレーション」ビューの「プロジェクト」セクションで「**HelloWorldProcess**」モジュールを展開し、「アセンブリ・ダイアグラム」をダブルクリックして、アセンブリ・エディターを開きます。
2. アセンブリ・エディターのパレットで、「プロセス」コンポーネントをクリックしてエディターのキャンバスまでドラッグします。
3. キャンバスで、新規プロセス・コンポーネントをクリックし、名前を `HelloWorldProcess` に変更して、**Enter** キーを押します。

注: 誤ってコンポーネントをダブルクリックすると「開く」ダイアログ・ボックスが開き、すぐにコンポーネントを実装するかどうか尋ねられます。「開く」ダイアログが表示された場合は、「いいえ」をクリックしてください (または **Esc** キーを押します)。

4. キャンバスで、新しい「**HelloWorldProcess**」コンポーネントをクリックしてコンポーネントの上に吹き出しバーを表示し、円で囲まれた「I」アイコンをクリックして「インターフェースの追加」ダイアログ・ボックスを開きます。
5. 「一致するインターフェース」リストで、「**HelloWorldProcess**」を選択して「**OK**」をクリックします。下図に示すように、新しい「**HelloWorldProcess**」インターフェースが「**HelloWorldProcess**」コンポーネントに追加されます。



6. パレットからキャンバスに「ヒューマン・タスク」をドラッグ・アンド・ドロップします。
7. 新しいヒューマン・タスク・コンポーネントを HelloWorldTask に名前変更します。
8. HelloWorldProcess コンポーネントにインターフェースを追加した場合と基本的に同じ手順で、インターフェース **HelloWorldTask** を新規ヒューマン・タスク・コンポーネントに追加します。
9. **HelloWorldProcess** プロセス・コンポーネントを **HelloWorldTask** ヒューマン・タスク・コンポーネントにワイヤリングして、「OK」をクリックします。「ワイヤーの追加」ダイアログ・ボックスが開いたら、「OK」をクリックします。以下に、ここまでの作業の結果を示します。

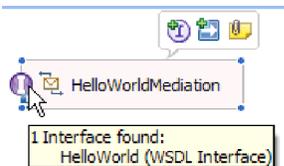


10. **Ctrl+S** を押して作業を保管します。
11. 「ビジネス・インテグレーション」ビューの「プロジェクト」セクションで、「HelloWorldMediation」プロジェクトを展開し、次に「アセンブリ・ダイアグラム」カテゴリーを展開して、「HelloWorldMediationExport」を選択し、HelloWorldProcess モジュール用のアセンブリ・エディターのキャンバスにドラッグします。「コンポーネントの作成」ダイアログ・ボックスが開きます。
12. 「SCA バインディング付きインポート」を選択し、「OK」をクリックします。SCA インポート・コンポーネントが生成され、「Hello World パート 1」サンプルからのモジュールの呼び出しに使用できます。

- 新しいインポートを **HelloWorldImport** に名前変更します。
- HelloWorldProcess** プロセス・コンポーネントを **HelloWorldImport** インポートにワイヤリングします。「ワイヤーの追加」ダイアログ・ボックスが開いたら、「OK」をクリックします。これで、アセンブリー・ダイアグラムは、以下のようになります。



- デフォルトで、アセンブリー・エディターのキャンパスは、自動レイアウト・モードになっているため、各コンポーネントは自動的に配置されます。ただし、コンポーネントの位置を手動で調整すると、自動レイアウト機能はオフになります。自動レイアウトのオン/オフを確認するには、ワークベンチ下部の状況表示行を見ます。状況が「オフ」の場合、以下の図に示すように、アセンブリー・エディターのキャンパス内で右クリックし、「自動レイアウト」を選択することによって、自動レイアウトをもう一度オンにすることができます。



あるいは、自動レイアウトをオフにしたまま、「コンテンツのレイアウト」を選択することによって、一回限りのレイアウトを実行することもできます。

- アセンブリー・エディターの内容を保存します。

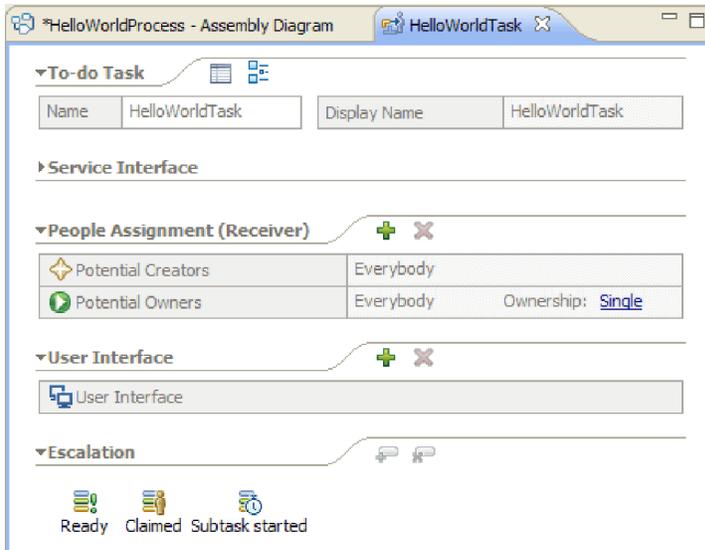
ヒューマン・タスク実装の作成

ここでは、ヒューマン・タスク・コンポーネントの実装を作成します。

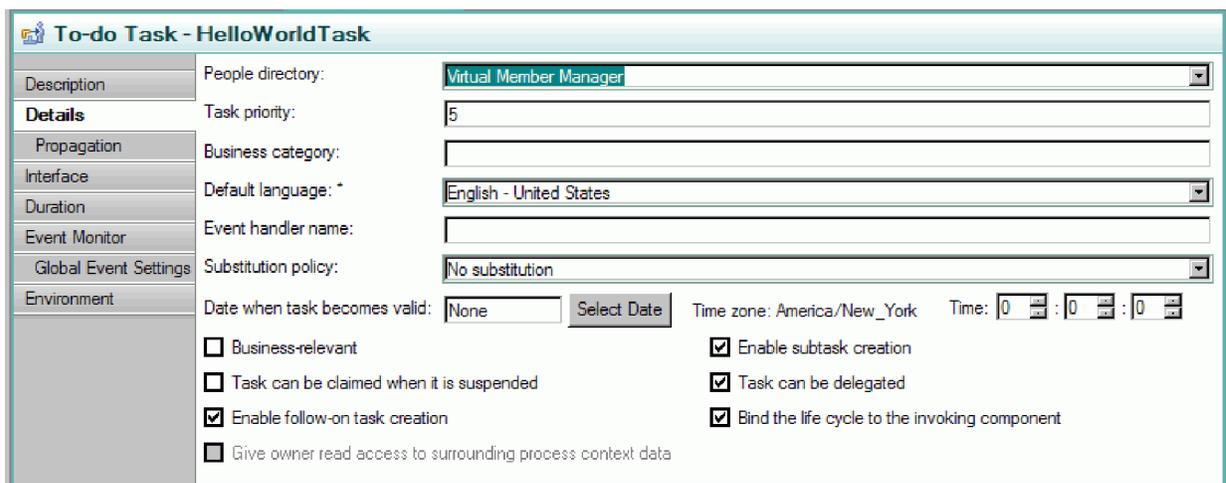
ヒューマン・タスク とは、個人によって実行される作業単位のことです。ヒューマン・タスクには、予定タスクなどのいくつかの種類があります。予定タスクでは、システムが個人にタスクを割り当てると、その個人が必要な情報を提供してタスクを完了し、その結果、プロセスに情報が返されます。

ヒューマン・タスク実装を作成するには、以下の手順を実行します。

- アセンブリー・エディターで、「**HelloWorldTask**」コンポーネントをダブルクリックします。「開く」ダイアログで「はい」をクリックして、実装を作成することを示します。「実装の生成」ダイアログで「OK」をクリックして、プロジェクトのルート・フォルダーに実装ファイルを作成することを示します。以下に示すように、ヒューマン・タスク・エディターが開きます。

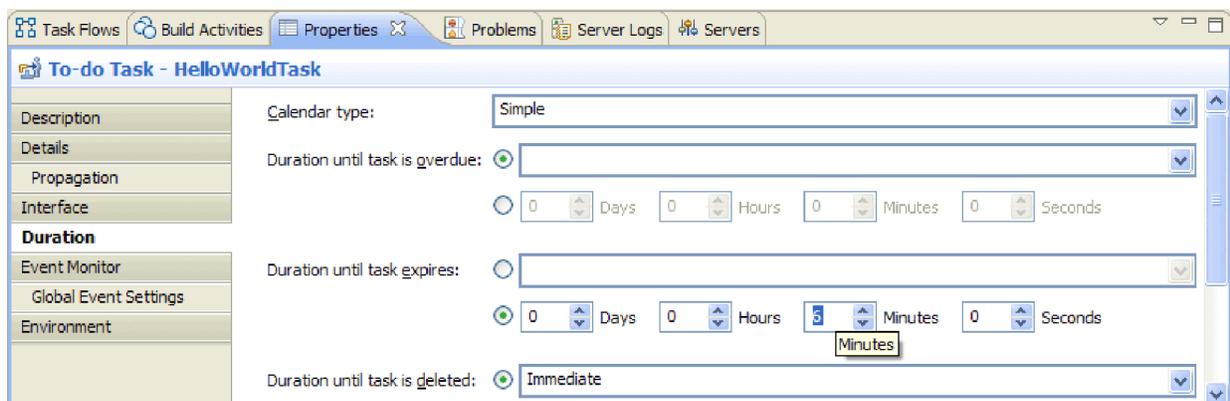


2. 厳密に言えば、ここでこれ以上の作業は必要ありませんが、学習を行います。したがって、ヒューマン・タスク・エディターのキャンバスの下部近くで、空白部分をクリックしてください。エディターの下方で「プロパティー」ビューを選択し、次に「詳細」タブをクリックします。
3. 以下に示すように、「プロパティー」ビューの右下で、「ライフ・サイクルを呼び出し側コンポーネントにバインドする」チェック・ボックスを選択します。



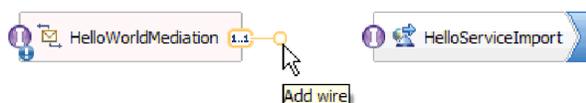
このアクションにより、タスクを呼び出したプロセスがクリーンアップされると、このタスクの未処理のインスタンス (予定) が確実にクリーンアップされます。

4. 「プロパティー」ビューで、「期間」タブを選択します。以下に示すように、「タスクの有効期限が切れるまでの期間」を 6 分に設定します。

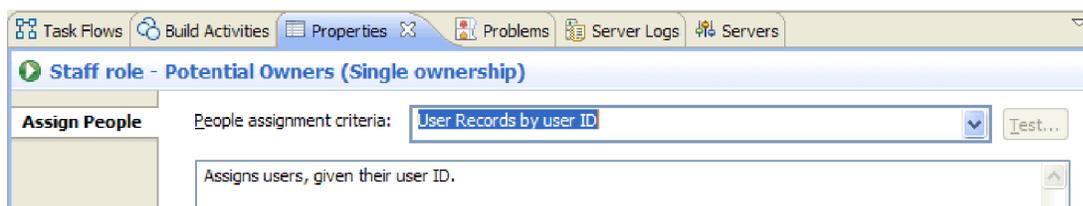


これにより、妥当な時間内にタスクを要求して完了するのを忘れた場合、タスク自身でそのクリーンアップを行います。

5. エディター・キャンバスで、下図の「担当者割り当て (受信側)」セクションに注目してください。



デフォルトでは、このヒューマン・タスクのインスタンスは誰でも作成でき (言い換えれば、予定を作成でき)、また誰でもそのインスタンスを要求し、作業を行うことができます。ただし、これは制限できます。「潜在的所有者」行で「全員 (Everybody)」セルを選択して、「プロパティ」ビューにフォーカスを移動します。そこには、「担当者の割り当て」タブのみが存在します。以下に示すように、「担当者割り当て基準」フィールドで、「User Records by user ID」を選択します。



「担当者の割り当て」ページで、表までスクロールダウンし、「ユーザー ID」の値を、インストール時にサーバーに指定したユーザー ID に設定します。以下の図に示すように、ユーザー ID を変更しない場合、引き続きデフォルトのユーザー ID admin を指定します。

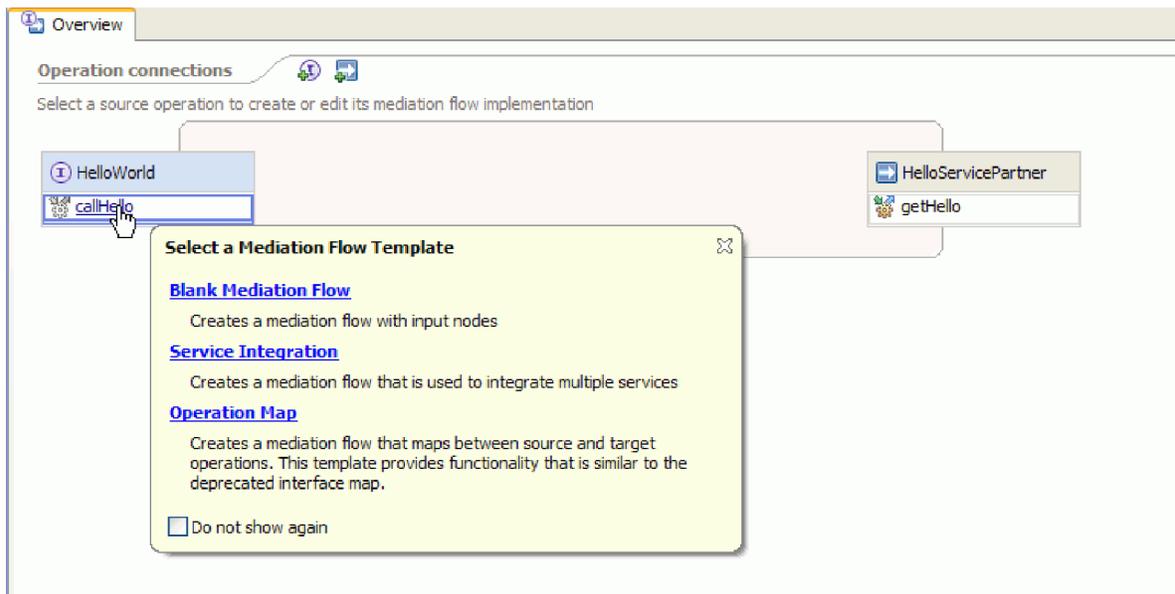
If only one person qualifies, claim the task automatically.

Name	Value
UserID *	admin
AlternativeID1	
AlternativeID2	

エディターの上部近くの「担当者割り当て」セクションで、以下の図に示すようにテーブルが変更されているのを確認できます。

People Assignment (Receiver)	
Potential Creators	Everybody
Potential Owners	User Records by user ID Ownership: Single
	UserID * admin

6. オプション: 以下に示すように、「エスカレーション」セクションに注意してください。



有効期限が切れるまでにユーザーがタスクを処理する期間を指定するほか、タスクが作成された後（「準備完了」ステート）、一定時間内に要求されなかった場合、またはタスクが要求された後（「要求済み」ステート）、一定時間内に完了しなかった場合のために、一連のエスカレーション・アクションを指定することもできます。

アクションには、誰かのために新しい予定タスクを作成すること、または E メール通知を送信することが含まれます。ここで、エスカレーションを設定し、ステート・アイコンの 1 つが選択された場合に、緑色の「プラス」ボタンを使用します。

期間とエスカレーションの両方に関して、経過時間を絶対時、分、または日で指定するだけでなく、ビジネス・カレンダーで指定することもできます。ビジネス・カレンダーを作成し、指定することにより、不連続な時間を識別できます。例えば、2 営業日の経過後にのみエスカレーションが発生するように指定することができます。

7. ヒューマン・タスク・エディターを保存して閉じてから、アセンブリー・エディターでの作業内容を保存します。

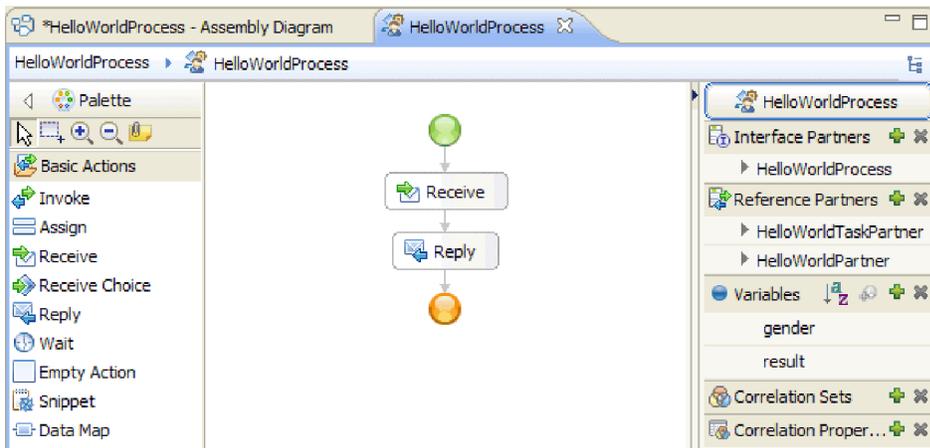
ビジネス・プロセス実装の作成

ここでは、ビジネス・プロセス・コンポーネントの実装を作成します。これは、ヒューマン・タスクの実装よりも複雑な手順です。

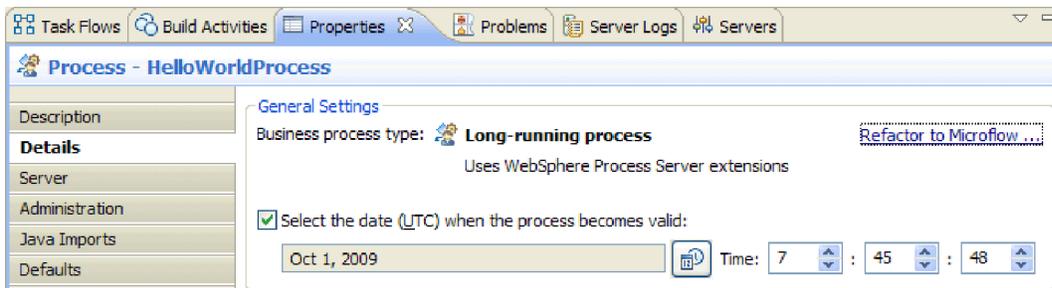
ビジネス・プロセス実装を作成するには、以下の手順を実行します。

1. アセンブリー・エディターで、「**HelloWorldProcess**」プロセス・コンポーネントをダブルクリックして作業を開始します。「開く」ダイアログで「はい」をクリックし、「実装の生成」ダイアログで

「OK」をクリックします。以下に示すように、ビジネス・プロセス・エディターが開きます。

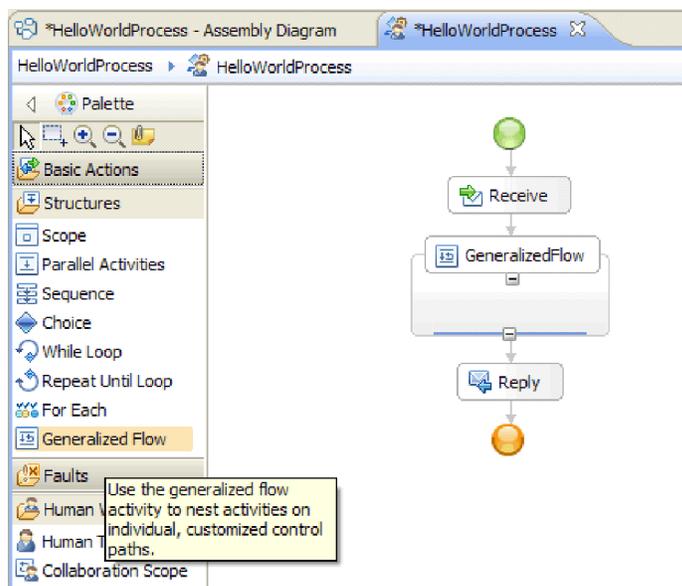


2. ユーザー・プロセスはヒューマン・タスクを呼び出す予定ですが、ヒューマン・タスクは応答に長い時間がかかる可能性があります。結果として、プロセスを「長期実行」として定義する必要があります。エディター内の任意の空白部分をクリックし、「プロパティ」ビューに移動して「詳細」タブを選択します。以下に示すように、このプロセスは「長期実行プロセス」として識別されていることに注意してください。



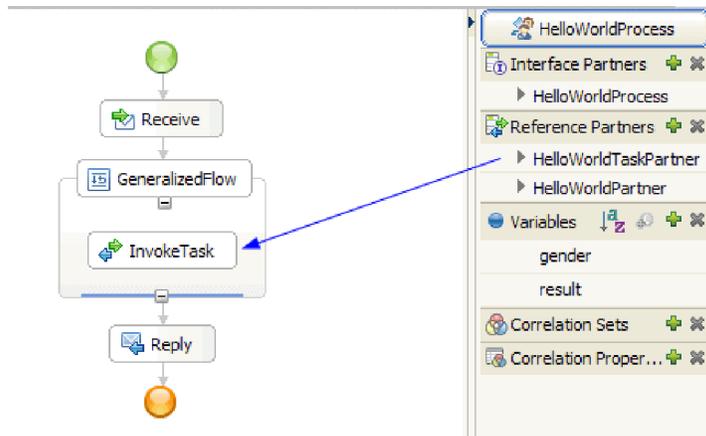
このプロセスを長期実行プロセスから microflow に変更するには、「マイクロフローへのリファクタリング」リンクをクリックします。これにより、プロセス定義だけでなく、アセンブリ・エディターのプロセス・コンポーネントも変更されます。これは、プロセスが変更され、この変更の影響を受けるすべてのダウンストリーム成果物が更新されることを意味します。

3. ビジネス・プロセスは、アクティビティ すなわち個々のステップで構成されています。最初にプロセスを、要求/応答操作を持つインターフェース用に作成した場合、その操作を通してプロセスを開始するための **Receive** アクティビティと、呼び出し元に応答を返すための **Reply** アクティビティの 2 つのアクティビティが事前に提供されています。この 2 つのアクティビティの間に、独自のアクティビティを挿入します。以下に示すように、パレット上の「構造」カテゴリを展開して、「GeneralizedFlow」をドラッグし、**Receive** と **Reply** の間にドロップします。



GeneralizedFlow など、いくつかのアクティビティは構造化されていますが、これは、他のアクティビティを含む目的を意味しています。ブランチするフローを生成するには、Parallel Activities、GeneralizedFlow、または Collaboration Scope を使用する必要があります。

- 以下に示すように、キャンバスの右側のトレイで「参照パートナー」カテゴリを展開し、「HelloWorldTaskPartner」をドラッグして「GeneralizedFlow」構造にドロップし、その名前を **InvokeTask** に設定します。



これで、この参照を呼び出す構成済みの Invoke アクティビティが作成され、アセンブリー・エディターに戻ってヒューマン・タスク・コンポーネントにワイヤリングされます。これは、実行時に、このアクティビティによってヒューマン・タスクのインスタンスが作成されることを意味します。つまり、「予定」タスクが作成され、その潜在的な所有者のいずれかから要求があるまで待機します。ただし、まだ完全には構成されていません。入力パラメーター・データを取得する場所と、出力結果を保管する場所を定義する必要があります。

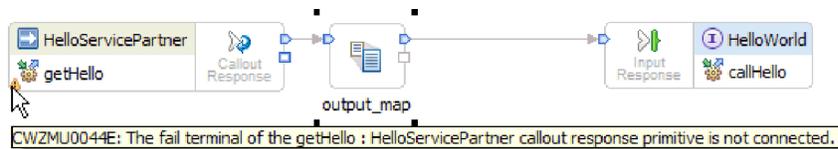
- キャンバス上で、「**InvokeTask**」アクティビティを選択し、エディターの下側の「プロパティ」ビューを開きます。「詳細」タブを選択します。テーブルの「入力」行で、「gender」行のセル内および「変数からの読み取り」という名前の最後の列内で「なし」をクリックします。ドロップダウン・リストに、このプロセスで現在定義されているすべての変数が表示されます。これらの変数は一致する型を持ち、この時点では、この型はプロセス・インターフェースの入力パラメーターと出力パラメーターの

みで構成されています。このプロセスの入力パラメーターである「gender」変数を選択します。この選択は、この変数のテキスト・データを、ヒューマン・タスクへの入力として渡すことを意味しています。

- 同様に、「出力」行で、「firstName」行の「変数への格納」という名前の最後の列のセルで「なし」をクリックしますが、今回は「新規」を選択して、**firstname** という名前の新しい変数を作成します。「lastName」行でも同様にして、**lastname** という名前の新しい変数を作成します。変数は、この参照パートナー・パラメーターに該当する型で作成されることに注意してください。テーブルは、以下のようになります。

	Name	Type	Read from Variable
Inputs	gender	string	gender
	Name	Type	Store into Variable
Outputs	firstName	string	firstname
	lastName	string	lastname

- 実行時に、**InvokeTask** が戻った後、ユーザーの名と姓の含まれる変数を取得します。呼び出す必要がある「Hello World パート 1」サンプル・サービスにも「Mister」などの「title」が必要です。つまり、値を保持する変数を定義する必要があります。下図に示すように、トレイの「変数」カテゴリーの横にあるプラス (+) アイコンをクリックします。



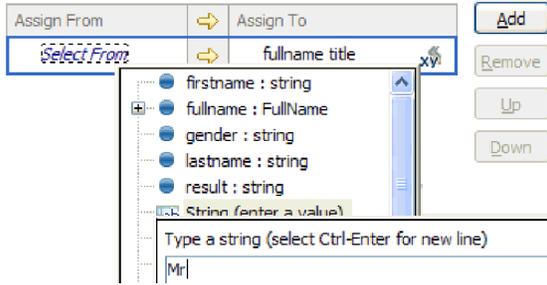
「変数の追加」ダイアログ・ボックスが開きます。

- 「名前」フィールドで、変数名 **fullname** を入力し、「Hello World パート 1」サンプルで作成したビジネス・オブジェクトである「**FullName**」型を選択します。「OK」をクリックします。これは、最初のサンプルのサービスが入力として必要とする変数の一致するデータ型です。ここで、「変数」リストは以下のようになります。

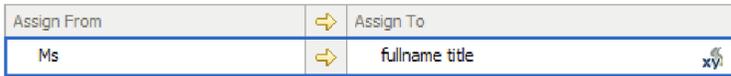
Variables
gender
result
firstname
lastname
fullname

fullname 変数には、すべて **string** 型の 3 つのフィールド、**title**、**firstName**、および **lastName** が定義されています。以下の手順で説明するように、サービスを呼び出す前に、この 3 つのフィールドすべてに値を設定する必要があります。

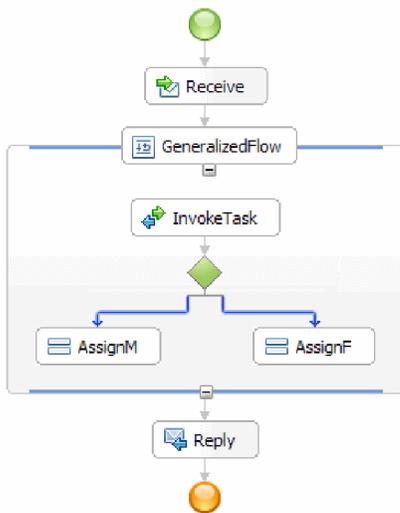
- パレットの「基本アクション」カテゴリーを展開して、「Assign」アクティビティをキャンパスの「GeneralizedFlow」構造の内部にドラッグ・アンド・ドロップします。名前を「AssignM」に変更します。
- 「AssignM」を選択することも、「プロパティ」ビューに移動し、「詳細」タブを選択します。テーブルの「割り当て先」列で、「選択先」をクリックします。ドロップダウン・リストで、「**fullname**」を展開し、「title」を選択します。
- 同じテーブル内の「割り当て元」列で、「選択元」をクリックします。以下に示すように、ドロップダウン・リストから「**ストリング (値を入力)**」を選択し、Mr と入力します。



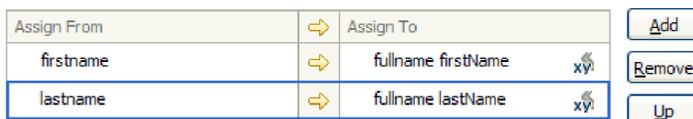
12. キャンバスで、「**InvokeTask**」アクティビティを「**AssignM**」アクティビティにワイヤリングして、ポップアップ・ダイアログ・ボックスから「**リンクの追加**」を選択します。
13. もう一度、パレットから「**Assign**」アクティビティをドラッグして「**GeneralizedFlow**」構造内にドロップし、「**AssignF**」という名前を付けます。以下に示すように構成して、「**title**」フィールドに「**Ms**」を割り当てます。



14. **InvokeTask** アクティビティを **AssignF** アクティビティにワイヤリングします。結果として表示されるポップアップで、「**リンクの追加**」を選択します。フローは、以下のようになります。

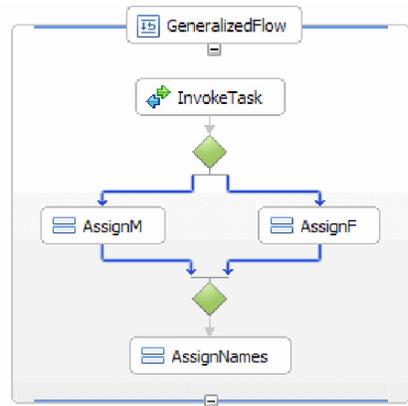


15. ここで、fullname の残りの 2 つのフィールド、すなわち「**firstName**」と「**lastName**」に値を割り当てます。これらのフィールドには、ヒューマン・タスクから返された値を設定します。パレットから再度「**Assign**」を「**GeneralizedFlow**」構造の内部にドラッグ・アンド・ドロップして、名前を「**AssignNames**」に変更します。「**追加**」ボタンを使用して別の行を「**Assign**」テーブルに追加し、下図に示すように構成します。

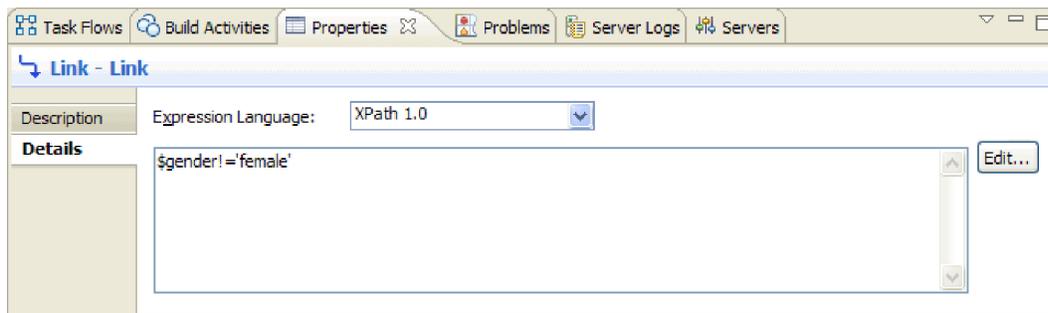


16. 「**AssignF**」アクティビティと「**AssignM**」アクティビティの両方を、この新しい **AssignNames** アクティビティにワイヤリングして、プロンプトが表示されたら「**リンクの追加**」を選択します。

GeneralizedFlow は、以下のようになります。

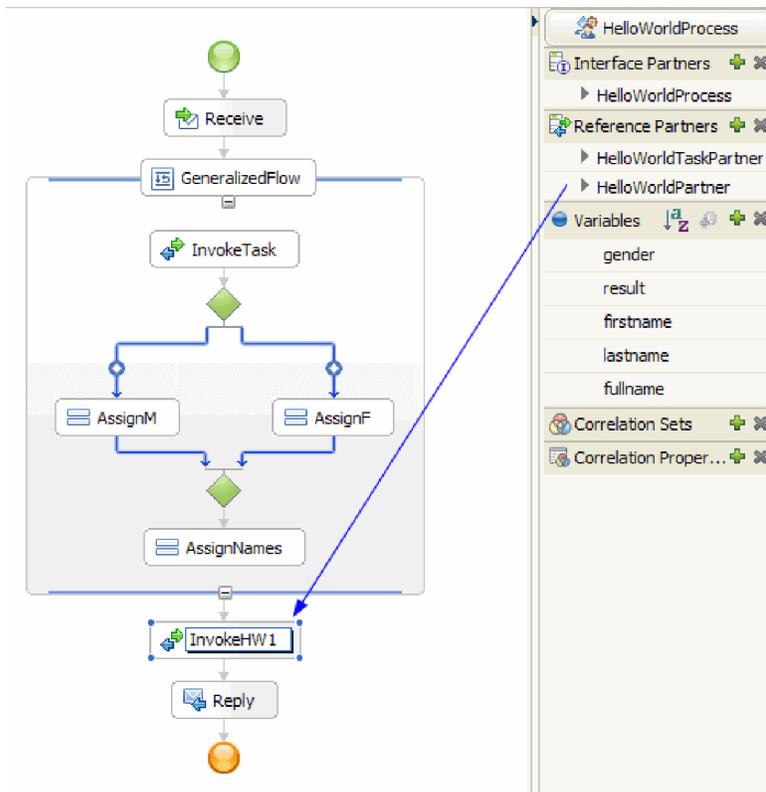


17. **InvokeTask** からの 2 つのリンクに条件を設定し、各リンクに従うタイミングに関する基準を作成する必要があります。 **AssignF** に入る青色のリンクを選択し、「プロパティ」ビューの「詳細」タブに移動します。「式言語」で「XPath 1.0」を選択し、「編集」ボタンをクリックします。XPath Expression Builder が開きます。
18. 以下のいずれかの方法で、性別の条件を「female」に設定します。
 - `$gender='female'` という式を直接入力して、「OK」をクリックします。
 - 「Simple XPath の挿入 (Insert Simple XPath)」をクリックしてデータ型のリストで「\$gender」を選択し、「オプションの条件の追加 (Add an optional condition)」領域で「演算子」として等号 (=) を選択し、対応する「値」フィールドに `female` と入力します。「OK」を 2 回クリックします。
19. 「AssignM」に入るリンクを選択し、以下に示すように、そのプロパティとして「式言語」を「XPath 1.0」、「条件」を `$gender!='female'` に設定します。



オプション: 「プロパティ」ビューの「説明」タブの「表示名」を設定することにより、リンクにラベルを付けることができます。次いで、そのリンクを右クリックし、「リンクのラベルを表示する (Show Labels on Links)」を選択すると、ラベルを表示できます。

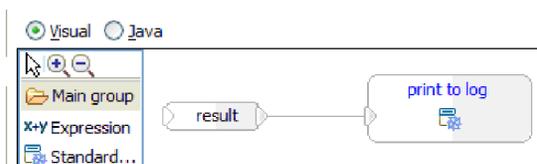
20. 以下に示すように、トレイに「HelloWorldPartner」をドラッグして「Reply」アクティビティの真上にドロップし、**InvokeHW1** という名前を付けます。



21. 以下に示すように、**InvokeHW1** アクティビティの「プロパティ」ビューの「詳細」タブで、入力パラメーターと出力パラメーターを **fullname** および **result** の変数にバインドします。

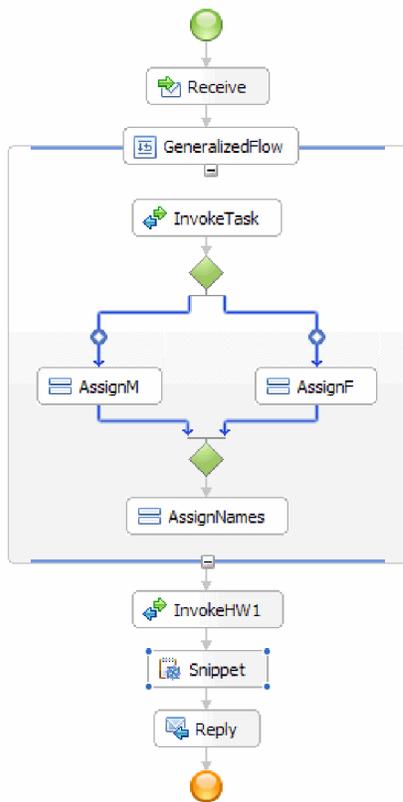
	Name	Type	Read from Variable
Inputs	fullname	FullName	fullname →
Outputs	result	string	← result

22. パレット内の「基本アクション」カテゴリから、「Snippet」を「Reply」の真上にドラッグします。「プロパティ」ビューの「詳細」タブに、Visual Java Snippet エディターが表示されます。「プロパティ」タブをダブルクリックして、フルスクリーンにします。パレットで、「標準」をクリックします。「標準 Visual Snippet の追加」ダイアログが表示されます。「utility」を展開し、「ログの出力」を選択して「OK」をクリックします。Visual Snippet エディターのキャンバスをクリックします。「print to log」ノードが表示されます。
23. 右側のトレイから Visual Snippet エディター・キャンバスに **result** 変数をドラッグします。「result」ノードから「print to log」ノードにワイヤリングします。すると、Snippet は以下のようになります。

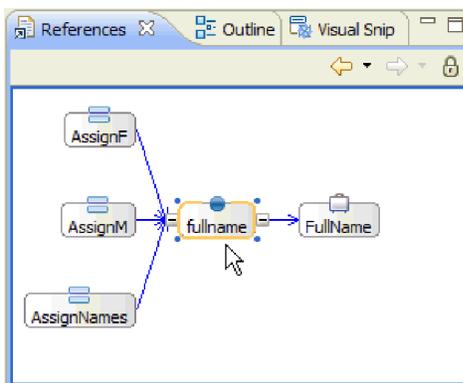


「プロパティ」ビューを、通常サイズに復元します。これで、「result」変数の内容を SystemOut に出力するよう、Java コードを視覚的に作成しました。技術的には、1 つのノードを別のノードにワイヤリングすることにより、入力パラメーターをメソッドに提供します。

24. これで、プロセスのオーサリングが完了しました。プロセスを保管します。以下ようになります。



25. アセンブリ・エディターに切り替えて変更内容を保存し、ビジネス・プロセス・エディターに戻ります。
26. オプション: プロセス内で、特定の変数を使用するアクティビティーを知りたい場合があります。それを知る方法があります。トレイ内で、「fullname」変数を選択します。「ウィンドウ」メニューから、「ビューの表示」>「参照」を選択します。パースペクティブの左下に「参照」ビューが開きます。このビューに表示されるグラフには、下図のように、変数を設定するすべてのアクティビティーが、その変数に対する入力として示されます。



おめでとうございます。これで、オーサリング手順が完了しました。今度はテストを行います。

第 3 章 サンプルの実行

サンプルのビルドまたはインポートが完了したら、サンプルを実行できます。

既製のサンプルをインポートした場合は、ディレクトリーおよび入力ファイルの作成に示すように、ディレクトリーを作成し、入力ファイルを追加する必要があります。また、アダプターのインポートに示すように、WebSphere Adapter for Flat Files もインポートする必要があります。

サンプルを実行するには、以下の手順に従います。

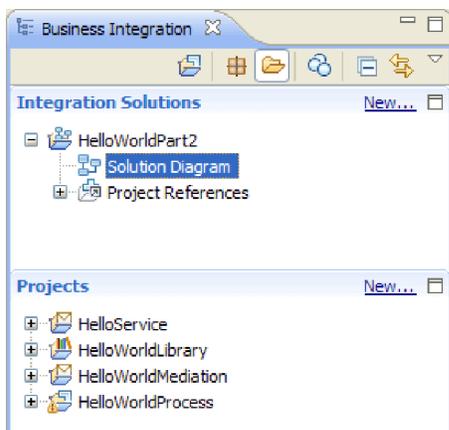
- 必要であれば、このサンプルの PDF 版を印刷します。既にサンプルをインポートしてあり、それを実行したい場合もあります。PDF 版を見ると、サンプルがどのようにして作成されたかが分かります。
- サンプルを自分自身でビルドするのではなく、インポートした場合は、WebSphere Adapter for Flat Files をインポートします。
- モジュールをサーバーにデプロイします。
- モジュールを実行し、テストします。

ソリューションの探索

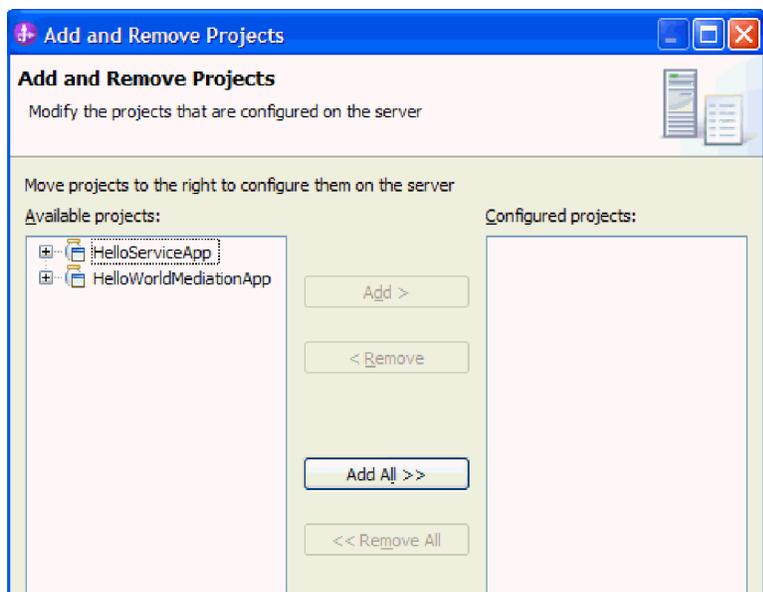
アプリケーションのさまざまな要素のすべてがどのように組み合わせられているのかを確認するために、その全体像を知ることが必要な場合があります。この作業には、統合ソリューション・エディターが役立ちます。

ソリューションを探索するには、以下の手順を実行します。

1. 以下に示すように、「ビジネス・インテグレーション」ビューで統合ソリューション **HelloWorldPart2** を展開します。

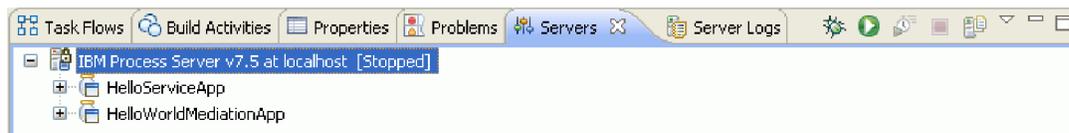


2. 「ソリューション・ダイアグラム」をダブルクリックします。次の図に示すように、統合ソリューション・エディターでソリューション・ダイアグラムが開きます。

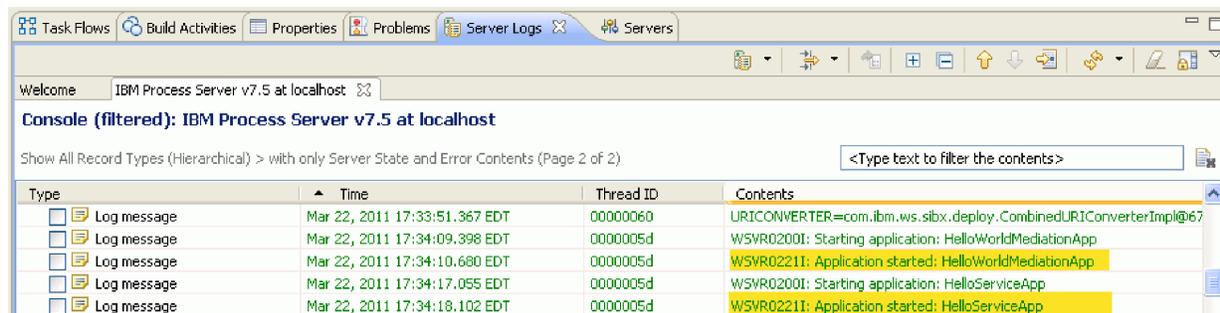


ソリューション・エディターでは、統合ソリューション内の 3 つのモジュールがすべて表示されます。各モジュールの左端にはエクスポートが表示され、各モジュールの右端にはインポートが表示されます (ただし、HelloService モジュールは除く)。非 SCA インポートおよびエクスポートは、統合ソリューションの端に接続されています。SCA インポートおよびエクスポートは、モジュール間の直接通信専用に設計されているため、統合ソリューションの端に接続されていません。モジュールが SCA バインディングによって別のモジュールを呼び出す場合、この 2 つのモジュール間に接続が表示されます。

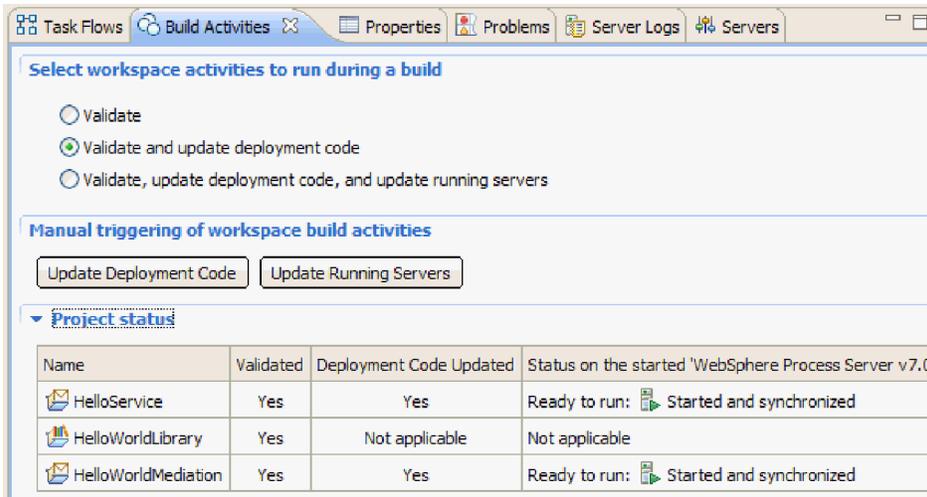
3. 下図に示すように、ソリューション・エディターで「すべてのプロジェクトを表示」アイコンをクリックし、**HelloWorldLibrary** ライブラリーを表示します。



これで、下図に示すように、統合ソリューション・ダイアグラムに **HelloWorldLibrary** ライブラリーと、このライブラリーを参照するモジュールへの接続が表示されます。



モジュールの内容を表示する場合は、モジュールの右上隅にある「モジュールの展開 (Expand module)」アイコンをクリックします。例えば、HelloWorldMediation モジュールの「モジュールの展開 (Expand module)」アイコンをクリックすると、以下の図に示すような内容が表示されます。



コンテキスト・メニュー項目に加えて、ツールバー上の他のアイコンを調べてみることもできます。例えば、アイコンやメニュー項目を使用すると、アセンブリー・エディターを開いたり、コメント用の付せんを追加したり、モジュールの背景色を設定したり、他の多くのタスクを実行したりすることができます。

4. ソリューション・ダイアグラムを閉じます。設定の保管を求めるプロンプトが出された場合には、「いいえ」をクリックします。

これで、統合ソリューションについてある程度理解したので、次にテストを行います。

モジュールのデプロイ

ランタイムでアプリケーションをテストできるよう、モジュールをサーバーにデプロイします。

アプリケーションがインバウンド処理モードで暗黙的に開始されたために、テストはモジュールがデプロイされるとすぐに始まります。

モジュールをサーバーにデプロイするには、以下の手順を実行します。

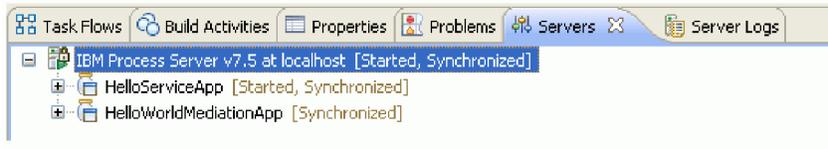
1. 「サーバー」をクリックして「サーバー」ビューを開きます。
2. 「IBM Process Server」を右クリックし、コンテキスト・メニューから「開始」を選択します。
3. サーバーが「開始済み」ステートに入った後、「IBM Process Server」を右クリックします。コンテキスト・メニューから、「追加と削除」を選択します。「使用可能プロジェクト」から、「SeparateCustomersApp」を選択します。「追加」をクリックします。「完了」をクリックします。
4. 前述のように、アプリケーションがインバウンド処理モードであるために、実行時のモジュールのテストはこの時点で開始されます。予期される出力については、モジュールのテストを参照してください。

サーバーからのモジュールの除去

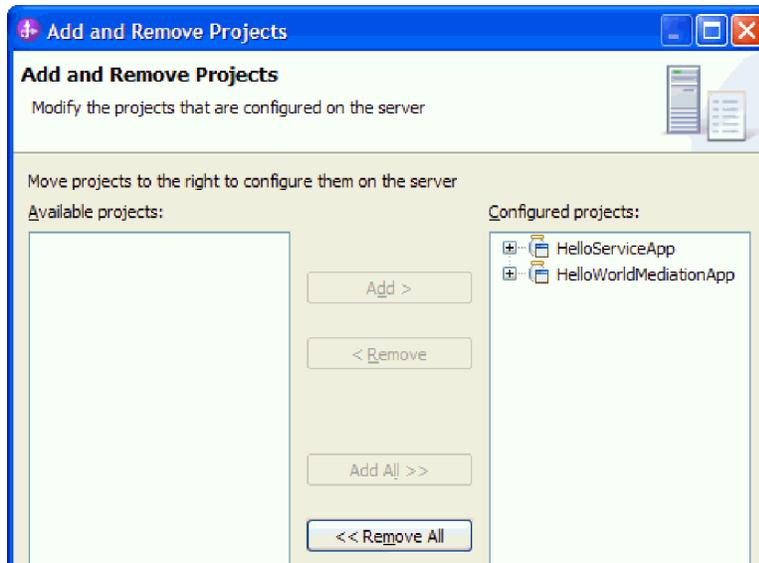
一般に、モジュールのテストが済んだら、サーバーからモジュールを除去します。これにより、サーバーにデプロイしたモジュールのみを対象としてテストの準備をすることになるため、サーバーの負荷が減り、パフォーマンスが向上します。

サーバーからモジュールを除去するには、以下の手順を実行します。

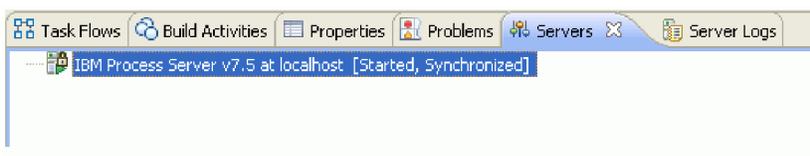
1. 「サーバー」タブをクリックします。以下の図に示すように、「サーバー」ビューが開きます。



2. 「サーバー」ビューで、「**IBM Process Server**」を右クリックして、「**プロジェクトの追加および除去**」を選択します。次の図に示すように、「プロジェクトの追加および除去」ダイアログ・ボックスが開きます。



3. 「すべて除去」をクリックします。「構成済みプロジェクト」リストから、アプリケーションが除去されます。
4. 「完了」をクリックします。サーバーからプロジェクトが除去されることを通知するダイアログ・ボックスが開いた場合、「**OK**」をクリックします。以下に示すように、「サーバー」ビュー内のサーバーの下にアプリケーションが表示されなくなります。



おめでとうございます。「Hello World パート 1: 入門」サンプルが完了しました。

第 4 章 インポート

このサンプルの完全な既製バージョンをインポートすることができます。

既製のサンプルをインポートするには、以下を実行します。

1. IBM Integration Designer で、「ヘルプ」>「サンプルおよびチュートリアル」>「**IBM Integration Designer 7.5**」を選択します。「サンプルおよびチュートリアル」ページが開きます。
2. 「ファイルの読み書き」セクションの下で「インポート」リンクをクリックします。「サンプルのインポート」ページが開きます。このページを使用して、単にそのサンプルだけをインポートすることも (ワークスペース内にフラット・ファイル・コネクタ・プロジェクト CWYFF_FlatFile が既に存在する場合)、フラット・ファイル・アダプター付きのサンプルをインポートすることもできます。該当するオプションを選択し、「**OK**」を選択します。
3. 既製のサンプルがワークベンチにインポートされます。

サンプルを実行する手順は、トピック『サンプルの実行』に書かれています。その説明には、インポートしたアプリケーションにアダプターを追加するための別のインポートが含まれています。アダプターのインポートを参照してください。

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

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。

〒242-8502

神奈川県大和市下鶴間1623番14号

日本アイ・ビー・エム株式会社

法務・知的財産

知的財産権ライセンス渉外

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。

Intellectual Property Dept. for WebSphere Software

IBM Corporation

3600 Steeles Ave. East

Markham, Ontario

Canada L3R 9Z7

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

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いません。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。

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。

表示されている IBM の価格は IBM が小売り価格として提示しているもので、現行価格であり、通知なしに変更されるものです。卸価格は、異なる場合があります。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。

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。

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めかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. _年を入れる_. All rights reserved.

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は表示されない場合があります。

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 (提供されている場合)、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

IBM、IBM ロゴおよび ibm.com は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、www.ibm.com/legal/copytrade.shtml をご覧ください。

Adobe、Adobe ロゴ、PostScript、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。

Java およびすべての Java 関連の商標およびロゴは Oracle やその関連会社の米国およびその他の国における商標または登録商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

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。

Intel、Intel ロゴ、Intel Inside、Intel Inside ロゴ、Intel Centrino、Intel Centrino ロゴ、Celeron、Intel Xeon、Intel SpeedStep、Itanium、および Pentium は Intel Corporation または子会社の米国およびその他の国における商標または登録商標です。

UNIX は The Open Group の米国およびその他の国における登録商標です。

ご利用条件

これらの資料は、以下の条件に同意していただける場合に限りご使用いただけます。

個人使用: これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布（頒布、送信を含む）または表示（上映を含む）することはできません。

商業的使用: これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。

お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入関連法規を含む、すべての関連法規を遵守するものとします。

IBM は、これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態を提供され、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なしで提供されます。

© Copyright IBM Corporation 2005, 2011. All Rights Reserved.