

Magic xpi 逆引き辞典

目次

1. Magic xpi 逆引き辞典	1
Magic xpi 逆引き辞典	1
概要	2
概要	2
保証された配達通知をセットアップするには	2
長いトランザクションで動作するには	3
ポーリング・メカニズムをセットアップするには	5
印刷のために Magic xpi プロジェクトからファイルを送信するには	6
データをコンテキスト間で共有するには	6
値の順序/配列を格納するには	6
一連の値で繰返すには	9
2つのコンテキストが同時に同じファイルを更新することを防止するには	10
インストール	10
インストール	10
Magic xpa3.x と Magic xpi4.x の両方を同じ PC にインストールする	10
コンポーネント	11
コンポーネント	11
Magic xpa から Magic xpi プロジェクトを呼び出すには	11
Magic xpi で Magic xpa のユーザ定義関数を使用するには	14
Magic xpi を使用して、Sugar にクオートを作成するには	17
Apache の ActiveMQ を使用した JMS コンポーネントを設定するには	17
HTTP トリガを作成するには	18
Exchange コネクタを使用する時に、任意のタイムゾーンでミーティングを設定するには	20
MSMQ を使用するには	21
データマップ	27
データマップ	27
XML 要素をチェックするには	28
データマップのデータベースウィザードを使用しないで、SQL ステートメントを作成するには	28
データマップを使用してデータベース処理を実行するには	30
DB2/400 データベースに対する言語サポートを作成するには	34
ダイナミックなテキストファイルを作成するには	35
テーブル内の他のカラムの値でテーブルカラムを更新するには	37
フラットファイル送り元から送り先に負数を設定するために、データマップを使用するには	38
Multi-Occurrence 複合物から特定のデータを取得するには	38
兄弟要素のデータに基づく Multi-Occurrence 複合物から特定のデータを取得するには	40
特定のインスタンスにマッピングするには	42
特定のインスタンスからマッピングするには	44
独立した送り元と順序/行構成にリンクするには	46
変換テーブルを使用するには	48
データ要素を統一するには	51
データ要素を分割するには	52
ヘッダと行レコードの両方を含むフラットな構造を階層的な XML にマッピングするには	55
送り元上で発生した繰返しをカウントするには	62
Multi-Occurrence 複合物から単一データ要素を取り出すには	64
データマップの特定の要素を検索するには	67
反復可能な複合物の中に、特定の数の項目を作成するには	68
データベース	69
データベース	69
動的な Where 句を使用するには	70

ODBC を使用するには	72
Magic xpi の DateTime データベース項目を使用するには	73
Magic xpi の内部データベースを変更するには	75
Magic xpi の内部データベースを手動で作成するには	77
SQL Server Express を使用するとき、パフォーマンスの問題を解決するには	78
SQL トランザクション用にデフォルトの BLOB サイズを変更するには	79
データベース・トランザクション上でロールバックを開始するには	80
マッピングのために MSSQL の DateTime タイプの送り元を Oracle の DateTime タイプに変換するには	80
Magic xpi に Oracle のシーケンスを読み込むには	80
Magic xpi でストアードプロシージャを呼び出すには	80
日付と時刻	83
日付と時刻	83
Magic xpi で DateTime の日付タイプを使用するには	83
Unix 時間を計算するには	83
現在の日付を取得するには	84
現在の時刻を取得するには	84
日付値を加算するには	85
時効値を加算するには	85
指定された日付に対応する月の最初の日付を取得するには	86
指定された日付に対応する年の最初の日付を取得するには	86
指定された日付に対応する月の最後の日付を取得するには	86
指定された日付に対応する年の最後の日付を取得するには	87
指定された日付に対応する曜日名を取得するには	87
指定された日付に対応する月名を取得するには	88
指定された日付の年/月/日の各部分を取得するには	88
指定された日付の週単位の日数を取得するには	89
日付値を文字列に変換するには	89
文字列で格納された日付を日付値に変換するには	90
XML DateTime フォーマットを計算するには	91
ユリウス歴の日付を計算するには	92
配備	93
配備	93
自動的に開始するようにプロジェクトを設定するには	93
配備する PC に、どのファイルをコピーする必要があるか	94
プロジェクトを配備環境に移動させる時にどの構成を修正する必要があるか	94
配備のために使用する実行可能ファイルを作成するには	94
プロジェクトのディレクトリスキャナコンポーネントがネットワークドライブからファイルにアクセスすることが できることを確認するには	95
Web サーバのエイリアスを定義する時に、クライアントのユーザとパスワードを取得するには	96
配備されたプロジェクトで HTTP リクエストのボディを受け取るには	97
HTTP リクエストの HTTP Verb を取得するには	97
運用環境でコンポーネントのロギングを保持する必要があるか	98
Web サービスのトリガを配備するには	99
ユーザ定義のコンポーネントを配備環境に移動するには	101
開発時の検討事項	102
開発時の検討事項	102
配備をより簡単にするために開発ガイドラインとは	102
Web サービスコンシューマを配備 PC に移行するには	102
ディレクトリスキャナコンポーネント	103
ディレクトリスキャナコンポーネント	103
ディレクトリスキャナのトリガによって処理されたファイルの数を制御するには	103
Dynamics CRM コネクタ	104

Magic xpi 逆引き辞典

Dynamics CRM コネクタ	104
非アクティブな状態にあるオブジェクトを更新するには	104
Dynamics CRM を使用して PhoneCall(アクティブ)を作成するには	106
エンコード	114
エンコード	114
ファイルが指定されたエンコードで定義されることを確認するには	114
指定されたエンコードでファイルを作成するには	117
Unicode か ANSI、UTF-8 のファイルをメモリに読み込ませるには	121
データマップを使用する場合、Unicode か ANSI、UTF-8 のフラットファイルでレコードを処理するには	122
Unicode か UTF-8 BOM を BLOB ファイルに追加するには	123
UTF-8 の内容を含むバイナリ BLOB を Unicode BLOB に変換するには	126
UTF-8 の内容を含む Unicode BLOB をバイナリ BLOB に変換するには	131
エラー処理	135
エラー処理	135
クラッシュ後にワークプロセスを再起動するには	136
事前に定義された後にワークプロセスの実行を停止するには	137
エラーが発生したときに、フローを中断するには	139
エラーを受け取った時に、E メールを受け取るようにするには	139
エラーフローでエラーについての追加情報を取得するには	140
エラーが発生したら、ステップのリトライ回数をコントロールするには	141
特定のエラーが発生した後に、フローを異なる場所に移動するには	142
どのフローがエラーを呼び出したかを識別するには	143
どのステップがエラーを呼び出したかを識別するには	143
どのトリガがエラーを呼び出したかを識別するには	144
どの最終ステップ ID がエラーを呼び出したかを識別するには	144
どの最終フローID がエラーを呼び出したかを識別するには	145
エラー処理フローからのフローにエラーを返すには	145
データベースエラーが発生した場合に、データマップを終了するには	146
例外が発生したデータベースのレコードフィールドを取り出すには	147
エラーを無視し、次のレコードに継続するには	148
クラッシュから回復する時にリソースを消去することができるか	149
フローにタイムアウトを設定するには	150
式	151
式	151
式ウィンドウで式の体制を整えるには	151
自動的にタイプ入力された関数の名前を完了するには	152
関数のヘルプページを表示するには	154
式エディタから関数リストを開くには	155
式の中で文字列値を定義するには	155
JD Edwards Enterprise One コネクタ	156
JD Edwards Enterprise One コネクタ	156
Magic xpi で JD Edwards Enterprise One の呼び出しを暗号化するには	156
JD Edwards Enterprise One で新しい顧客レコードを作成するために Magic xpi を使用するには	156
ライセンス	158
ライセンス	158
プロジェクトのためにスレッドを確保するには	158
プロジェクトの潜在的な並列実行機能を制限するには	159
プロジェクトが運用ライセンスを使用していることを確認するには	159
プロジェクトのライセンスを変更するには	160
ライセンスをリフレッシュするには	161
複数のサーバを使用するためにライセンスを設定するには	161
ナビゲーションとワークスペース	162

ナビゲーションとワークスペース	162
Magic xpi スタジオのデフォルトのオープニング表示を定義するには	162
プロジェクトのオブジェクト内のテキストを検索するには	163
プロジェクトのエラーをチェックするには	164
ツールティップをより長く表示させるには	168
プロジェクトのデバッグ	169
プロジェクトのデバッグ	169
デバッガを使用して、プロジェクトをデバッグするには	169
ブレイクポイントをプロジェクトに設定するには	171
データベースゲートウェイによって記録された情報を制御するには	172
動的なゲートウェイログを作成するには	173
SAP R/3(JCO)コンポーネントのためにログファイルを作成するには	173
Magic xpi をデバッグするためにデバッグツールを使用するには	174
Magic xpi をデバッグするために User-Mode Dump Heap (UMDH) を使用するには	186
Magic xpi で .NET ユーティリティコンポーネントをデバッグするには	187
プロジェクト	187
プロジェクト	187
新しいプロジェクトを作成するには	188
既存のプロジェクトを開くには	189
プロジェクトをバックアップするには	191
Magic xpi プロジェクトを削除するには	192
プロジェクトをリネームするには	193
Salesforce コネクタ	194
Salesforce コネクタ	194
Magic xpi の Salesforce アプリをインストールするには	194
最初の Salesforce プロジェクトを作成するには	197
setRefbyExternalId 関数を使用して Salesforce における Update と Upsert を行うには	200
Salesforce を Web サービスと呼ぶために XMLSpy を使用するには	202
SAPB1 コネクタ	206
SAPB1 コネクタ	206
SAPB1 コネクタで複数オブジェクトを照会するには	206
SAPB1 コネクタで複数のオブジェクトを追加するには	211
SAPB1 コネクタをアップグレードするには	212
SAPB1 フィールド内の値をクリアにするには	212
SAPB1 HANA を使用するには	212
SAP R/3 コネクタ	213
SAP R/3 コネクタ	213
Magic xpi を起動させるために SAP R/3 システムを設定するには	213
マスタデータ分布のために SAP R/3 システムを設定するには	218
サーバ	220
サーバ	220
Magic xpi のサーバメッセージを管理するには	220
Magic xpi サーバ用のゲートウェイログを作成するには	222
ソース管理	226
ソース管理	226
ソース管理を使用して作業する環境を設定するには	227
ソース管理によって管理されるプロジェクトを作成するには	230
既存のプロジェクトをソース管理データベースに追加するには	233
プロジェクトをソース管理から削除するには	236
ソース管理プロバイダの定義、変更を行うには	238
オブジェクトを以前のコピーにロールバックするには	239
変更内容をトラッキングするには	240

Magic xpi 逆引き辞典

Space	242
Space	242
Magic xpi 4.x のために Unicast Discovery の設定を行うには	242
1 台の PC 上のサーバの複数インスタンスを実行するには	244
モニタから新規サーバインスタンスを追加するには	245
モニタからサーバインスタンスをシャットダウンするには	246
Salesforce を使用してプロジェクトをモニタするには	247
Web サービス	249
Web サービス	249
Web サービスのトリガを使用するには	249
Web サービスのコンシューマとして Magic xpi を設定するには	257
統合 Windows 認証によって保護されている Web サービスを呼び出すには	261
Magic xpi の中の Web サービスに基本認証を提供するには	262
WS-Security で動作している Web サービスプロバイダとして、Magic xpi を設定するには	264
HTTP コンポーネントを使用して Web サービス呼び出しを行うには	268
手動で Systinet Server for Java をインストールするには	270
Magic xpi で Web サービス Over SSL のコンシューマとして使用するには	279
IIS7 で証明書を処理して、Systinet に SOAP リクエストを送るには	288
2. 索引	295

Magic xpi 逆引き辞典

Magic xpi 逆引き辞典

このセクションの中のトピックは、Magic xpi を使用して異なる統合シナリオを統合する方法についての一般的な概要について提供します。

ここでは各トピックのそれぞれ適切な箇所で、以下の情報を提供を提供しています。

- 入門と概要
- ソリューションを実装するために必要な異なるステージ
- 追加／補足情報

関連トピック：

[概要](#)

[インストール](#)

[コンポーネント](#)

[データマップ](#)

[データベース](#)

[日付と時刻](#)

[配備](#)

[開発時の検討事項](#)

[Dynamics CRM コネクタ](#)

[エンコード](#)

[エラー処理](#)

[式](#)

[JD Edwards Enterprise One コネクタ](#)

[ライセンス](#)

[ナビゲーションとワークスペース](#)

[プロジェクトのデバッグ](#)

[プロジェクト](#)

[Salesforce コネクタ](#)

[SAPB1 コネクタ](#)

[SAP R/3 コネクタ](#)

[サーバ](#)

[ソース管理](#)

[Space](#)

[Web サービス](#)

概要

概要

以下のトピックは、Magic xpi での一般的なシナリオについて説明しています。

[保証された配達通知をセットアップするには](#)

[長いトランザクションで動作するには](#)

[ポーリング・メカニズムをセットアップするには](#)

[印刷のために Magic xpi プロジェクトからファイルを送信するには](#)

[データをコンテキスト間で共有するには](#)

[値の順序/配列を格納するには](#)

[一連の値で繰返すには](#)

[2つのコンテキストが同時に同じファイルを更新することを防止するには](#)

保証された配達通知をセットアップするには

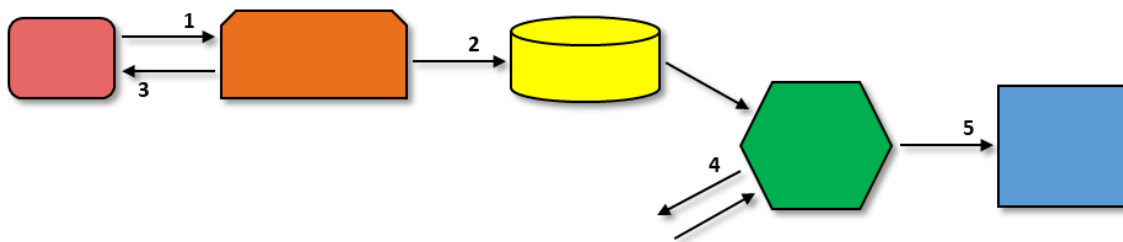
このトピックでは、メッセージが配信された時とその結果プロセスが完了した時、どのように Magic xpi に通知するかを設定する方法について説明しています。

HTTPトリガや Web サービスなど、即時の応答が求められているリクエストを EAI エンジンが受け取ったが、即座の処理や応答が不可能なケースがあります。

このような状況では、以下のことを行う必要があります。

- リクエストにリクエストが届いたことを通知するメカニズムを作成します。
- 到着したリクエストを処理のため確実に保存します。

以下のダイアグラムが処理を表しています。



以下のステップがダイアグラムに示されています。

1. リクエストが HTTP から送られます。
2. リクエストが確実な保存場所に保存されます。
3. リクエストの配信を通知するメッセージが返されます。
4. 通知とは独立してリクエストが処理されます。
5. 通知が(例えば、E メールで)送られます。

ステージ(2)では、リクエスト情報はデータベースやキュー、もしくはファイルに格納することも可能です。

Magic xpi 逆引き辞典

実装ステージ

上記の機能を実装するには、以下のステップを Magic xpi のフローに作成する必要があります。

- リクエストを処理し、永続的なストレージにそれらを格納し、確認(ACK)を送る処理。
- 保存エリアからリクエストを取得し、リクエストを処理し、それからプロセスの完了通知を送る処理。

この機能は、以下のものを使用することで Magic xpi で実現することができます。

1. リクエストを受け取る HTTP トリガは、キューにリクエスト情報を格納し、ID を持つ確認をクライアントに返します。
2. 保存エリアとして機能する MSMQ コンポーネント
3. リクエストが MSMQ コンポーネントに到達するのを待つ追加処理。

リクエストが到着すると、処理が起動されます。次に、リクエストが処理され、成功か失敗かの通知が(例えば、Eメールによって)送られます。

長いトランザクションで動作するには

長いトランザクションは、ワークフロー実装シナリオの分野です。通常、ワークフローの実装シナリオには、ワークフローのプロセスを一旦保留し、ユーザの入力を待ってから継続するユーザ介入ポイントがあります。

Magic xpi を使用した長いトランザクションのソリューションは、Magic xpi フローを実行することにより起動されたステップを含むワークフロープロセスを参照します。長いトランザクションが実行されると、ステップ間のコンテキストを保持する必要があります。これはトランザクションのコンテキストを ODS に保存し、次のステップに入る時にコンテキストを読み戻します。

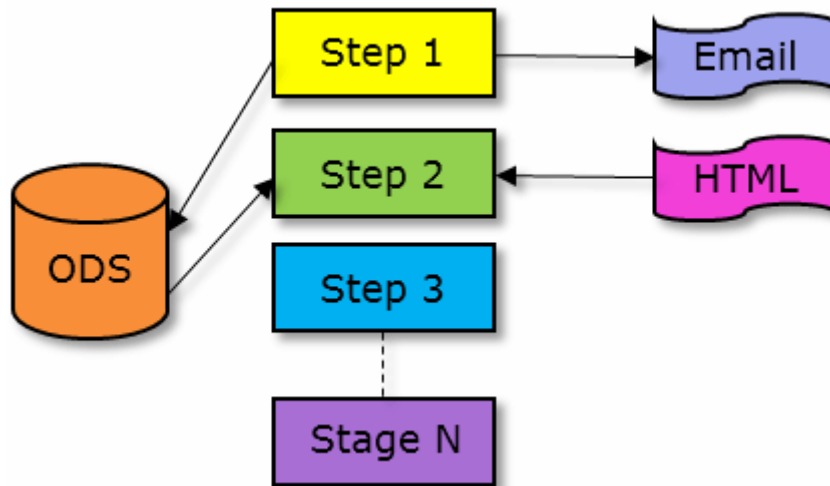
簡単な長いトランザクションの例としては、顧客が特定金額以上の注文を出した場合です。ビジネスプロセスでは、その額を超えた注文においては、会計課で手動で承認を取ることを要求しています。注文プロセスは、会計課が注文依頼を承認もしくは却下するまで中断されることになります。

以下は長いトランザクションで、ひとつのユーザ介入ポイントを達成することが要求されている実施段階の説明です。このプロセスは eMail メッセージと応答フォームのある HTML ページを使用します。Web サービスやファイル、COM を使用する等、要求機能を実行する他の方法もあります。また、Magic xpi トリガコンポーネントを使用して長いトランザクションから次のステップを起動することもできます。

このプロセスは、以下の 3 つの大きなステップを含んでいます。

- 注文の承認を手動で受け付ける為の HTML フォームを添付した E メールを送信します。
- 現在のコンテキストデータを ODS に保存します。
- 応答フロー (HTTP トリガ) を起動し、格納されたコンテキストデータを読み出し、フローの残りを実行します。

以下の図が三つのステップを説明しています。



実装ステージ

A. ユーザ介入を準備する

以下のステップは、通常ユーザ介入を必要とするフローの最後に実行されるべきです。

- 応答するために以下の内容を含む [送信] ボタン付きの HTML 文書を用意します。
 - Magic xpi サーバ HTTP リクエスト
 - アプリケーションとしての Magic xpi サーバ
 - 呼び出されるプログラムは、HTTP でなければなりません
 - 次のフローで使用される HTTP トリガ名
 - 次のステップでのコンテキストデータ読み出しの為にトランザクションコンテキスト ID を含め、実行に必要な他のパラメータ
- 7. フローデータユーティリティを使用して、トランザクションコンテキストデータを ODS に保存します。データは、変数やユーザ情報等あらゆる格納タイプを使用することができます。選択されたコンテキスト ID は、ユニークでなければならず、データはそれをキーとして使用します。
- 8. 作成された HTML 応答フォームを添付した E メールメッセージを用意し送信します。

B. 長いトランザクションで次のステップを起動する

長いトランザクションの次のステップは、長いトランザクションを継続するのに必要な情報を持っている Magic xpi サーバへの HTTP リクエストによって起動されます。

- 新しいフローを HTTP トリガコンポーネントで作成します。フロー名は応答 HTML 文書の [送信] ボタンで使用されているものと同じ名前を使用します。
- フローデータユーティリティを使用して、トランザクションコンテキストデータをコンテキスト ID (HTTP リクエストパラメータとして渡された) を使用して ODS から読み出します。
- 長いトランザクションをユーザ介入に応じて継続します。

ポーリング・メカニズムをセットアップするには

ポーリングは、EAI/BPM 産業においてとても重要なコンセプトです。このトピックでは、Magic xpi がインテグレーションプロジェクトでどのように簡単にポーリング機能を設定できるか、いくつかの方法を紹介しています。

ポーリングとは、他のプログラムやデバイスを一つのプログラム又はデバイスから連続的にチェックし、プログラムやデバイスの状態を監視する機能です。EAI/BPM の世界では、これは特定の FTP アドレスに届くファイルであったり、特定のキューに入力されたメッセージであったり、特定のアドレスに届いた eMail メッセージであったりします。ポーリングデバイスがデバイス（からの信号）が到着したと判断した場合、例えばファイルが特定の場所に作成された場合、デバイスはインテグレーションシナリオで定義した通りに実行される処理を起動します。

このトピックでは、以下のフローについて説明しています。

- ポーリングを指定されたデバイスで行います。
- デバイス（からの信号）が到着すると実行を継続する他のフローもしくはステップを起動します。

Magic xpi でポーリングフローを定義する方法がいくつかあります。以下のものを使用することができます。

- トリガコンポーネント
- 自動起動、自動繰り返しフロー

実施ステージ

オプション 1: トリガ

特定ファイルの到着を常に監視するディレクトリスキャナコンポーネントをトリガとして使用することができます。ファイルが到着すると、トリガがフローもしくはステップを起動し、そのまま次のファイルの到着を監視し続けます。

オプション 2: 自動起動と自動繰り返しフロー

このオプションでは、[自動起動]と[自動繰り返し]のプロパティを **Yes** に設定してフローを使用します。[自動繰り返し]プロパティでは、フローは最後のフローコンポーネントの実行が終了すると自動的にフローの再実行を行います。言い換えると、このプロパティを使用することでループを作成することができるということです。

ポーリングを有効にするにはフローを以下のように作成します。

1. フローの最初のステップは特定のデバイス（からの信号）を探します。
2. デバイス（からの信号）が到着するとフローが実行されます。
3. デバイス（からの信号）が到着しないとフローは停止します。
4. [自動起動] プロパティを **Yes** に設定すると、フローを起動し直し、最初のステップでデバイスからの最新の信号を探します。

追加情報

このトピックでは、ディレクトリスキャナコンポーネントと関連していますが、ポーリングをメッセージング等、他のコンポーネントで実装することもできます。メッセージングコンポーネントをキューのポーリングを行うトリガとして使用することができます。これにより、キューに届くそれぞれのメッセージがイベントを起動し、フローの実行を継続させることができます。

印刷のために Magic xpi プロジェクトからファイルを送信するには

Magic xpi プロジェクトから印刷するためにファイルを送るには、DOS コマンドを使用する必要があります。

印刷コマンドを設定するには、ファイル管理コンポーネントの **Command Line** メソッドを使用してください。

注 意: ほとんどの印刷コマンドでは、ファイルを参照するための構文は、ローカルファイルへのパスです。プロジェクトから印刷することができる前に、ファイルを Magic xpi のフローのローカルファイルとして(ファイル管理コンポーネントを使用して)保存する必要があることを意味します。

データをコンテキスト間で共有するには

フロー間でのデータ共有はデータの保全や集中管理を可能にします。

子リニアフローは、親フローのコンテキストあるいはメモリーセグメントの範囲内で実行されます。フロー変数とコンテキスト変数は同じコンテキスト内の全てのフローで共有されます。

パラレルフローは、異なるコンテキストで実行され、独自のフロー変数とコンテキスト変数を持っています。フローが停止するとコンテキストは閉じられ、このフローで使用されていた全てのフロー変数とコンテキスト変数は消去され無くなります。

実行されている他の全てのフローあるいは特定のフローとデータを共有する必要がある場合があります。

Magic xpi において、異なるコンテキスト上で動作しているフロー間でデータを共有したい場合、通常はグローバル変数やグローバル ODS、または外部記装置を使用する必要があります。

実施ステージ

フロー間のデータ共有の実際の例は、異なる倉庫の在庫の確認とそれに続く結果の確認です。この処理実装するには以下のようにします。

- 在庫残高を確認する元のリクエストを処理するフローを作成します。フローは **3** つの並列分岐(各倉庫に **1** つ)をもとに作成してください。
- 実行中、フローは **3** つの並列分岐を発生させ、それらがリニアの分岐上の「完了を待つ」プロパティを使用して、完成するのを待ちます。これらのフローは在庫残高を調べ、**ODS** システムを現在のストック残高で更新し、現在の在庫残高とその倉庫が注文要求に答えられるかを示します。
- すべての並列分岐が終了したら、フローのリニア部分は継続し、**ODS** からすべての **3** 棟の倉庫の在庫を読み込み、倉庫の在庫量が注文要件を満たすことができるかどうかを決定します。

値の順序/配列を格納するには

Operational Data Storage (ODS) システムは、複数のフローコンポーネントまたはフローによって共有することができるように、データを保存します。本質的には、ODS は、少容量のデータで多くのクエリが利用できるようにするクイックアクセスのデータストレージです。これはデータウェアハウスとは異なります。これは、大量の情報が格納され、クエリが大容量のデータ上で動作します。

Magic xpi の ODS システムは、ODS に保存される各項目のためのエントリを格納するデータテーブルを持っています。ODS に保存されたすべての項目は、フローのシーケンス ID と UserKey によって識別されます。

Magic xpi 逆引き辞典

データの配列を ODS の UserKey に追加することができます。配列は、同じ UserKey で複数の値のストレージを利用することができます。配列における値は、インデックスと呼ばれます。例えば、ODS1_1 は、ODS1 の UserKey における最初のインデックスです。

データが多くのソースから到着し、後で順番に処理されなければならない場所で、ODS の配列は並行処理のアーキテクチャで使用されます。従って、マルチスレッド環境においては、データを格納するためにユニークな名前を ODS エントリに割り当てることは必要ではありません。

ODS データは文字、数値、日付、時間、論理そして BLOB の各データタイプをサポートし、配列内の各値は異なるデータタイプかもしれません。例えば、配列におけるインデックス 1 は文字で、インデックス 2 は数値であるかもしれません。

Magic xpi ODS システムは、これらのデータストレージのモードをサポートします。

- **ローカルフローストレージ(ODS ローカル)** これは同じフロー内の並行ステップで使用するために特定のフローのデータを ODS データベースに保存します。サーバが開始されると、データは ODS から消去されます。
- **グローバルストレージ(グローバル ODS)** フロー間のデータを保存するためにこのモードを使用することができます。すべてのフローは、データを取り出すことができます。

実装

ODS リポジトリ内に手動で ODS データを入力することができます。ODS リポジトリには、すべての固定の ODS エントリ名と説明が入っています。フローデータユーティリティで ODS を使用する時に 使用する ODS エントリを選択することができます。

ODS エントリをリポジトリに追加する

ODS エントリをリポジトリに追加するには、以下の手順に従ってください。

1. **ODS リポジトリ**を開くには、ソリューションエクスプローラの [リポジトリ] セクションで [ODS] をダブルクリックしてください。または、**Shift+F6** を押下することで ODS リポジトリを開くこともできます。
2. [ODS リポジトリ] ダイアログボックスで、[追加] をクリックすると新しいエントリを作成することができます。
3. 以下の情報をリポジトリのカラムに入力してください。
 - **名前** ODS データの名前を入力してください。これは **UserKey** です。
 - **説明** データの目的の確認に役立つ説明に入力してください。これはオプションです。
 - **範囲** グローバル ODS、またはローカル ODS のどちらかを選択してください。
4. 情報を入力し終わったら、リポジトリを閉じて、データを保存するために [OK] をクリックしてください。

ODS エントリでデータを管理する

ODS エントリの挿入、更新、削除、消去を行うためにフローデータユーティリティを使用することができます。これはフローの一部として自動的に ODS データを処理します。

1. **フローデータユーティリティ**を、フローにドラッグしてください。
2. フローデータユーティリティをダブルクリックするか、右クリックしてコンテキストメニューから [構成] を選択し、[フローデータ構成] ダイアログボックスを開きます。

3. [フローデータ構成] ダイアログボックスで、[追加] をクリックして、新規エントリを作成します。サービスを設定するために、以下のガイドラインを使用してください。
- **アクション** …… 以下のどれかの動作を選択することができます。
 - **Update** …… データを置き換えるか、変数に追加します。例えば、ファイルの内容を **UserBLOB** または **UserString** に追加することができます。フロー変数の値を変更することもできます。
 - **Insert** …… データを **ODS** キーインデックスに追加します。インデックスは、**ODS** 配列の一部です。これは **1** つの **ODS** キーでより多くの複数データを入力することを可能にします。配列のための個々のデータエントリはインデックスの中に含まれています。**Insert** を選択すると、[インデックス] フィールドは無効になります。**ODS** レコードが作成される時、次のインデックスとして自動的に作成されます。
 - **Delete** …… **ODS** エントリを消去します。これは **UserKey** レベルでデータを消去し、**UserKey** は存在なくなります。
注意：Delete を選択すると、**ODS** におけるすべてのデータが削除されます。
 - **Clear** …… 内部の値を指定された **ODS UserKey** から削除します。しかし、**UserKey** は残っていて、いつでもデータを追加したり、変更することができます。
 - **Reset** …… フロー変数の値を作成時に入力されたオリジナルのデフォルト値に変更します。フロー変数を作成する時にデフォルト値を入力した場合のみ、このオプションが使用されます。
 - **タイプ** …… **ODS ローカル/ODS グローバル**な(上記を参照)
 - **名前** …… 変数リストから選択して **ODS** データの名前を入力してください。これは以前に作成された変数のリストです。
 - **データタイプ** …… データに基づいたデータタイプを設定してください。
 - **インデックス** …… 例えば、3 つの注文が到着した後に、同じ **ODS** キーを持つ 3 つのレコードがあります。これらのレコードの **ODS** インデックスは **1**、**2**、および **3** になります。**Insert** や **Update**、または **Clear** を行いたい **ODS** インデックスを入力してください。**Update** または **Clear** を行う場合に **ODS** 配列内のどのインデックスかを示すためにインデックスを入力したり、新規データを配列に挿入する時に、追加されるインデックス番号を指定するために、インデックスを入力します。このアクションは **ODS グローバル**と **ODS ローカル**のデータタイプをサポートします。
注意：Clear または **Update** を選択し、インデックスを入力しない場合、最初のインデックスエントリは更新されるか、消去されます。
 - **更新式** …… **ODS** への更新/挿入で使いたいデータを入力してください。
 - **条件** …… これはフリーテキストのフィールドです。更新される条件を示す論理式に入力してください。**True** は、更新が行われることを示します。**False** は、更新されないことを示します。

ODS 関数

Magic xpi は、以下の ODS 関数を提供します。

Magic xpi 逆引き辞典

- **ODSArrayCnt** この関数は指定された ODS 配列内の要素をカウントします。
- **ODSExists** この関数は、指定された ODS インデックスが ODS データベースに存在しているかどうかをチェックします。
- **ODSGet** この関数は、指定された ODS エントリの値を取り出します。

一連の値で繰返すには

Magic xpi で一連の値を繰返すには、3 つの異なる方法があります。

- 自動繰り返し
- GoTo
- フロー呼び出し

自動繰り返し

[フロープロパティ]ダイアログボックスでこのオプションを定義します。自動繰り返しは、フロー内で最後のフローコンポーネントが完了すると、サーバが再びフローを呼び出すかどうかを指定します。ループ効果を作成するためにこの機能を使用することができます。

GoTo

フロー内で、1 つの分岐の実行が完了した時、どのコンポーネントを実行すべきかを指定することができます。Magic xpi の GoTo コマンドを使用してこれを行うことができます。GoTo コマンドを使用するには以下のようにします。

1. 移動したいコンポーネントを選択してください。
2. 右クリックして、[GoTo] を選択してください。
3. 移動したいコンポーネントを選択してください。
4. Magic xpi は接続をします。

これらのステップが完了したら、Magic xpi は、次に実行されるコンポーネントを示す点線が表示されます。

注 以下の場合にのみ、GoTo コマンドを使用することができます。

意:

- 共通の親コンポーネントを共有するコンポーネントを使用している場合。
- リニアのコンポーネント、またはリニア分岐内のコンポーネントを指定している場合。

フロー呼び出し

これは、送り先の定義の一部としてデータマッピングユーティリティで利用可能なオプションです。データソースからのデータによって呼び出されるフローの回数に応じて、複数回のループを作成するためにフロー呼び出しを使用することができます。

例

この例は、XML フォーマットで顧客からの注文を受け取ることを仮定しています。XML ファイルには、必要なすべての項目と数量が含まれています。各注文は、異なる数の項目を含んでいるかもしれず、注文に対する全体の在庫量をチェックするための仕組みを設定する必要があります。

1. 2つの新しいフローを作成してください：**FlowA** と **FlowB**。
2. 2つの新しいフロー変数を **FlowB** に追加してください：**F_Items** と **F_Amount**。
3. データマップユーティリティを **FlowA** にドラッグしてください。
4. データマップをダブルクリックするか、右クリックして「設定」をクリックしてください。「データマップ」ウィンドウが開きます。
5. ツールボックスの「マップスキーマ」セクションから、XML タイプを「データマップ」ウィンドウの送り元エリアにドラッグしてください。
6. XML 送り元の「プロパティ」ペインで、の XML 送り元を設定してください。
7. ツールボックスの「マップスキーマ」セクションから、フロー呼び出しタイプを「データマップ」ウィンドウの送り先エリアにドラッグしてください。
8. フロー呼び出しの送り先の「プロパティ」ペインで、**FlowB** を選択してください、次に「保存」をクリックしてください。
9. 「データマップ」ウィンドウで、値を XML ファイルから 2 つのフロー変数にマッピングしてください。
10. 「データマップ」ウィンドウで、送り元の複合要素を送り先カラムのフロー名にマッピングしてください。

これによって、注文の中に含まれている項目数と一致するまで、2 番目のフローが何度も呼び出されることになります。

2つのコンテキストが同時に同じファイルを更新することを防止するには

一方のコンテキストがテキストファイルを更新し、それを保存すると、他のコンテキストが保存したファイルの変更が失われるかもしれません。

この問題を解決するには、Magic xpi のロックリソースサービスを使用することができます。ロックリソースサービスは、フローロジックの重要なセクションでロックすることを可能にします。このセクションにユニークな名前を指定します。これはロックシステムに追加されます。ロックを開放するには、アンロックリソースサービスを使用します。

プロセスがこの重要なセクションに達すると、ロックされたとしてフラグを設定します。このセクションに達している他のプロセスは、ロックされたセクションでリソースを修正することができません。

ロックが開放されるまで待つことになります。無制限にロックされないようにするために、タイムアウトを定義することもできます。

関連トピック：

ロックユーティリティ

インストール

インストール

以下のテクニカルノートは、Magic xpi の一般的な問題について扱っています。

[Magic xpa3.x と Magic xpi4.x の両方を同じ PC にインストールする](#)

Magic xpa3.x と Magic xpi4.x の両方を同じ PC にインストールする

Magic xpa3.x と Magic xpi4.x を同じ PC にインストールするには、一方のインストール(GigaSpaces と Magic xpa の両方に対して)のデフォルトのディスカバリポートを変更する必要があります。

Magic xpi 逆引き辞典

例えば、Magic xpi を変更すると決めた場合、以下を行ってください。

1. **magicxpi-setenv.bat** ファイルで、次の行を有効にしてください: **set DISCOVERY_PORT=<my new port value>**

これによって、新しいディスカバリポートが GigaSpaces インフラストラクチャーに設定されます。

2. **Magic.ini**、**ife.ini**、および **ifm.ini** ファイルで、新しい特性を **JVM_ARGS** セクションに追加してください: **-Dcom.gs.multicast.discoveryPort=<my new port value>**

これは、デフォルトのディスカバリポートをサーバ、スタジオ、およびモニタに設定します。運用環境だけで使用する場合は、**Magic.ini** ファイルのこの値を変更するだけですみます。

注 意:	Magic xpi4.5 から、 ife.ini と ifm.ini ファイルはインストール処理の対象でなくなりました。
---------	--

また、Web リクエストまたはコマンドラインリクエスト用にポートを設定する必要があります。

- **mgreq.ini** ファイル(コマンドライン用はルートフォルダ、Web リクエスト用は **Scripts\config** フォルダ内)の **locator** の設定行に移動し、以下の構文を使用してポートを追加してください: **<locator>:<port>**

コンポーネント

コンポーネント

以下のトピックは、Magic xpi のコンポーネントについて扱っています。

[Magic xpa から Magic xpi プロジェクトを呼び出すには](#)

[Magic xpi で Magic xpa のユーザ定義関数を使用するには](#)

[Magic xpi を使用して、Sugar にクオートを作成するには](#)

[Apache の ActiveMQ を使用した JMS コンポーネントを設定するには](#)

[HTTP トリガを作成するには](#)

[Magic xpi で JD Edwards Enterprise One の呼び出しを暗号化するには](#)

[JD Edwards Enterprise One で新しい顧客レコードを作成するために Magic xpi を使用するには](#)

[Exchange コネクタを使用する時に、任意のタイムゾーンでミーティングを設定するには](#)

[MSMQ を使用するには](#)

Magic xpa から Magic xpi プロジェクトを呼び出すには

Magic xpa の[コールリモート]処理コマンドを使用して Magic xpi のプロジェクトを呼び出すことができます。

このために必要なセットアップは、Magic xpa をどこから実行しているかに依存します。2 つの異なる方法でこれを実行することができます。

- Magic xpi をインストールする時に、インストールされる Magic xpa から
- スタンドアロンの Magic xpa から

Magic xpa と Magic xpi を同じ PC にインストールしたい場合、次を参照してください:[Magic xpa3.x と Magic xpi4.x の両方を同じ PC にインストールする](#)

Jar ファイルを定義する

両方のシナリオで、<Magic xpi インストール>¥Runtime¥java¥lib フォルダから **Magic.ini** ファイルに定義された Magic xpa の **CLASSPATH** エントリに 以下の jar ファイルをコピーする必要があります。

- commons-lang3-3.1.jar
- jaxp-xercesImpl.jar
- log4j-1.2.16.jar
- log-commons-logging.jar
- mgxpi-commons.jar
- mgxpi-gs.jar
- uniRequester.jar
- xpicommons-gs.jar

スタンドアロンの Magic xpa を使用している場合、<Magic xpi インストール>¥Runtime¥GigaSpaces¥lib¥ から全ての jar ファイルを追加する必要があります。

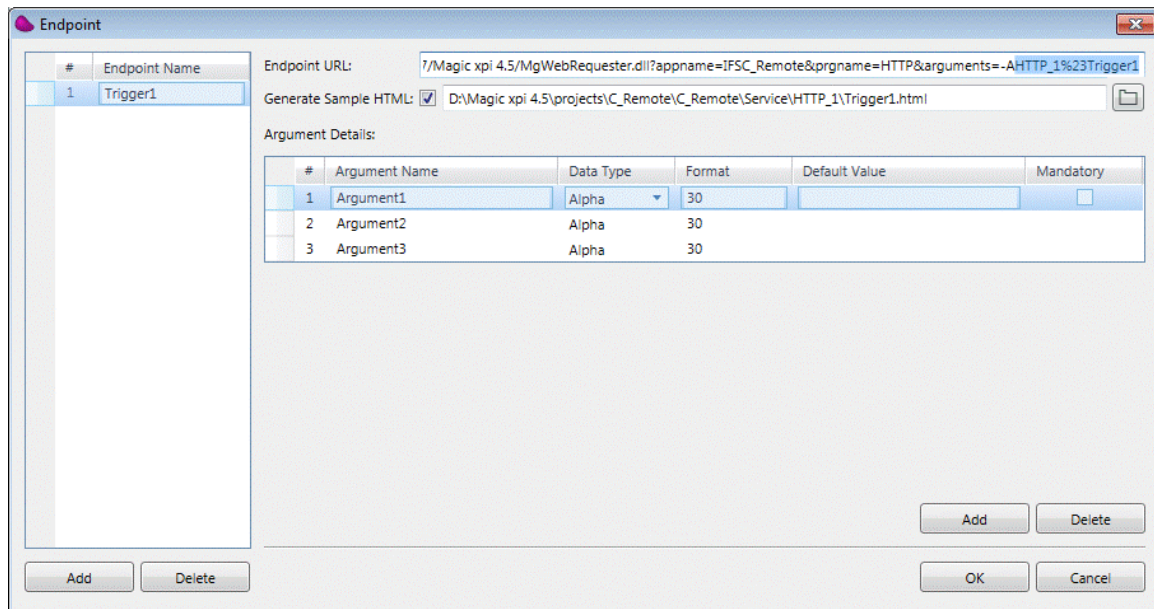
注	Magic xpa と Magic xpi で使用される Java Runtime のバージョンが同じであることを確認してください。
----------	--

Magic xpi の実装ステップ

コンポーネント SDK で作成されたトリガを使用している場合、Magic xpi では何もする必要がありません。
コンポーネントビルダで作成されたトリガを使用している場合、以下を行う必要があります。

1. Magic xpi を開いてください。
2. [プロジェクト] メニューを選択し、[設定] を選択してください。
3. HTTP サービスを追加してください。
4. Web サーバとエイリアスを入力してください。
5. [エンドポイント] ボタンをクリックしてください。
6. ダイアログボックスの左下の [追加] をクリックしてください。
7. 引数を追加するために、ダイアログボックスの右下の [追加] をクリックしてください。
8. [サンプル HTML の作成] チェックボックスを選択してください。
9. [エンドポイント URL] を記録してください。Magic xpa でサーバを定義している時に、これが必要になります。

Magic xpi 逆引き辞典

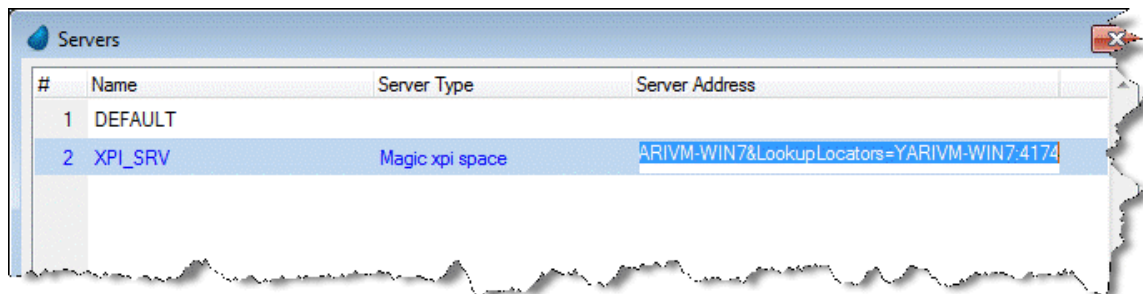


Magic xpa の実装ステップ

Magic xpa で、サーバとサービスを作成する必要があります。

Magic xpi サーバを作成する

1. Magic xpa を開いてください。
2. [オプション] メニューから、[設定]、次に [サーバ] を選択してください。
3. 新しいサーバを定義し、[サーバタイプ] に **Magic xpi Space** を設定してください。



4. [サーバアドレス] カラムで、以下の値(Space との接続)を定義してください。
SpaceName=MAGIC_SPACE&LookupGroupName=<name>&LookupLocators=<host:[port]>

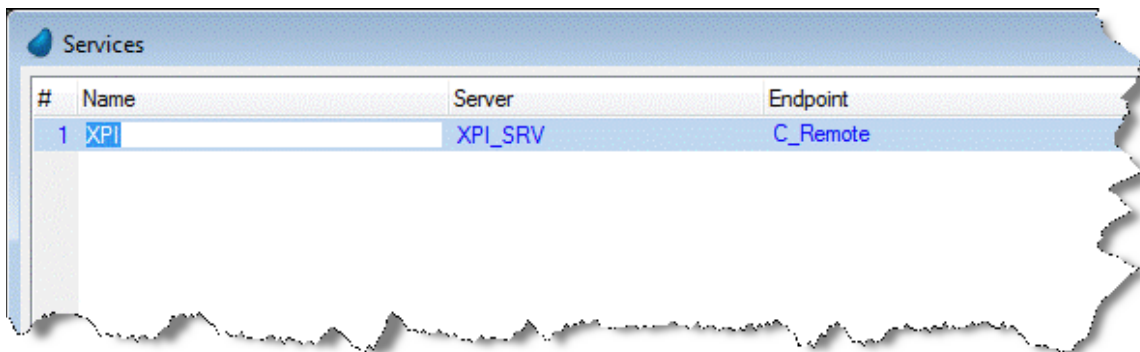
例えば： SpaceName=MAGIC_SPACE&LookupGroupName=Magic_xpi_4.5_YARIVM-WIN7&LookupLocators=YARIVM-WIN7:4174

注 意： LookupGroupName/LookupLocators の値は、Magic xpi の **Magic.ini** ファイルから削除してください。LookupLocators プロパティに値が設定されていない場合、エントリは等号(=)までの状態で残しておいてください。

Magic xpi サービスを作成する

1. Magic xpa の [オプション] メニューから、[設定]、次に [サービス] を選択してください。

2. 新しいサービスを定義し、前のセクションで作成したサーバを選択してください。
3. [エンドポイント] カラムで、Magic xpi のプロジェクト名(例えば C_Remote)を入力してください。



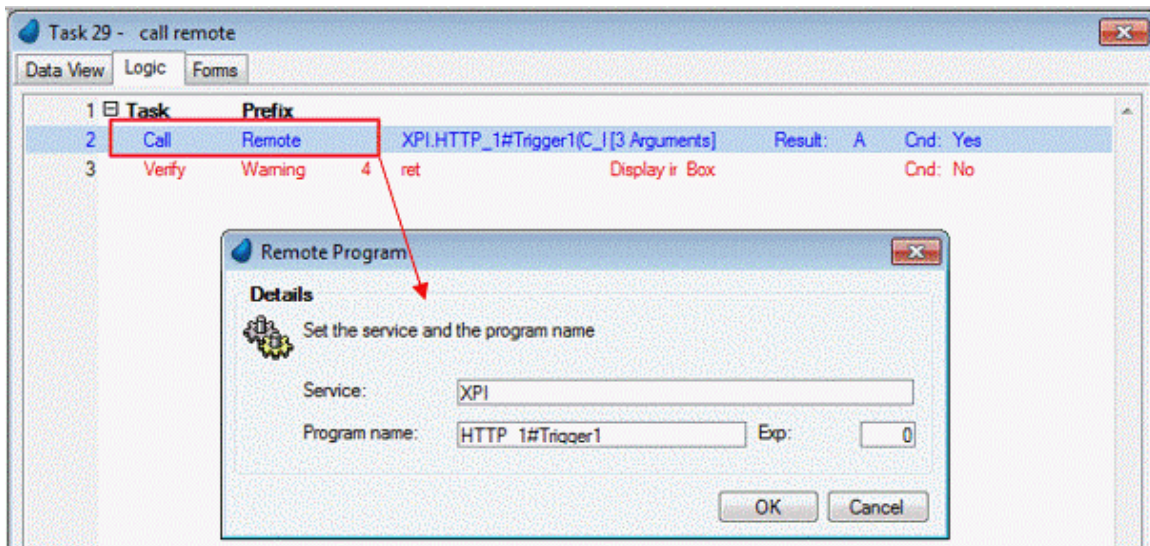
Magic xpa から Magic xpi の呼び出し（[コールリモート] 処理コマンド）

定義したサービス呼び出すため、Magic xpa の[コールリモート]処理コマンドを使用することができます。

注意: Magic xpi へのコールリモートは、文字と BLOB のタイプのみ戻すことができます。

コンポーネント SDK で作成された Magic xpi トリガを呼び出すか、Web リクエスト(IIS)を通さない k Magic xpi の HTTP トリガを呼び出すことができます。

- コンポーネント SDK で作成された Magic xpa トリガを使用している場合、次の行を Magic xpa プログラムに追加してください：<サービス名>.prgName(<Magic xpi プロジェクト名>)
- HTTP トリガを使用している場合、Magic xpi で定義されたエンドポイントの名前(例えば、HTTP_1#Trigger1)を使用してください。



Magic xpi で Magic xpa のユーザ定義関数を使用するには

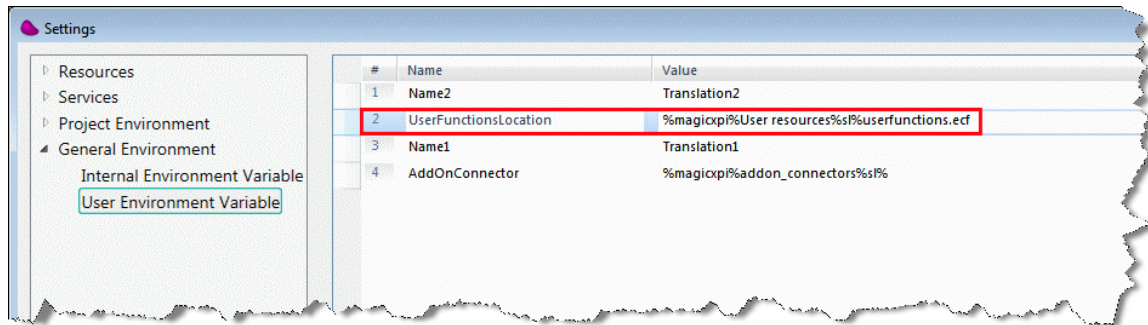
Magic xpa のユーザ定義関数を Magic xpi のプロジェクトで使用することができます。これを行うには、以下の手順に従ってください。

Magic xpi 逆引き辞典

Magic xpi

Magic xpi で、以下の手順に従ってください。

1. [プロジェクト] メニューから、[設定] を選択してください。[設定] ダイアログボックスが開きます。
2. [設定] ダイアログボックスの [一般的な環境] セクションで、[ユーザ環境変数] をクリックしてください。
3. ここに設定されたパスを記録してください。

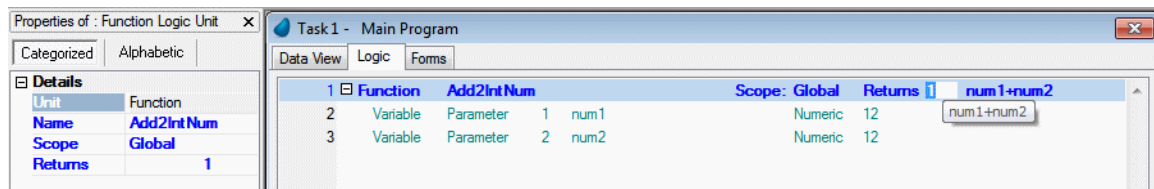


この設定は、Magic xpi の **Magic.ini** ファイル(<Magic xpi インストール>%Runtime%Magic xpa フォルダ)の [MAGIC_LOGICAL_NAMES] セクションで定義することもできます。

Magic xpa

Magic xpa で、以下の手順に従ってください。

1. 新しいプロジェクトを作成してください。
2. ユーザ定義関数を作成してください。



3. Magic xpi の **UserFunctionsLocation** 設定(上記で説明)で定義された名前と同じであることが確認されたこのプロジェクトのキャビネットファイルを作成してください。
4. ファイルを Magic xpi の **UserFunctionsLocation** 設定で定義されたパスに保存してください。例えば: **C:\Magic xpi 4.6\Runtime\User resources\UserFunctions.ecf**

Magic xpi

Magic xpi に戻って、以下の手順に従ってください。

1. **<Magic xpi インストール>\Studio\FunctionDescription.xml** ファイルを開き、必要であれば戻りタイプとパラメータを含めて、関連した詳細情報を定義した関数をリストに追加してください。以下のような既存の関数をコピーし、**DisplayName** や **ReturnValue**、**Description** の各タグ、**<Parameters>**、および他の関連したすべてのタグを編集することによって、これを行うことができます。

```
<Function>
  <DisplayName>ABS</DisplayName>
  <ReturnValue>N</ReturnValue>
  <HidelItem>N</HidelItem>
  <MinArgNum></MinArgNum>
  <MaxArgNum></MaxArgNum>
  <Obsolete></Obsolete>
  <HelpID>9032</HelpID>
  <FunctionSignature></FunctionSignature>
  <Description>Returns the absolute value of a real number, without regard to
sign.</Description>
  <Parameters>
    <Parameter>
      <Name>number</Name>
      <Type>N</Type>
      <Description>Any number.</Description>
    </Parameter>
  </Parameters>
</Function>
```

2. Studio を開き、新しく作成された関数が式エディタで利用可能であることを確認してください。
3. データマップで新しい関数を使用するために、**objectid** として関数を **comps.exp** ファイルに追加する必要があります。これは**<Magic xpi インストール>\Runtime\Templates\Mapper** フォルダに配置されています。テキストをファイルにコピーし、**Component name** や **MciFile val** タグ、および **PublicName val** タグの値を更新することで、これを行うことができます。

```
Component name="UserFunction">
  <Compld val="463816514-25434-19825-92354D96D"/>
  <MciFile val="C:\Magic xpi 4.6\Runtime\Magic
Xpa\Builders\ECI\userfunctions\UserFunction.eci"/>
  <ComponentType val="Magic xpa"/>
  <Type val="U"/>
  <CABINET_FILE val="%UserFunctionsLocation%"/>
  <LoadImmediate val="Y"/>
  <CtlIndex val="6"/>
  <LastISN val="1"/>
  <ComponentFuncs>
    <Object>
      <id val="1"/>
      <ItemIsn val="1"/>
      <PublicName val="userfunctionsTest"/>
    </Object>
  </ComponentFuncs>
```

Since version: 4.5

Magic xpi を使用して、Sugar にクオートを作成するには

Magic xpi を使用し、Sugar のクオートを作成するには、以下の手順に従ってください。

1. クオートを作成するために、**Sugar XML** インタフェースを使用してください。
2. 製品バンドルを作成するために、**Create Product Bundles** メソッドを使用してください。
3. 量や価格、および関連した割引率などの値を定義を確認し、クオートに格納したい製品を作成してください。
4. 製品バンドルと製品をリンクするために、**Link** メソッドを使用してください。
5. クオートと製品バンドルをリンクするために、**Link** メソッドを使用してください。

注意： クオートのシナリオは、コネクタのレガシーAPI の実装ではサポートされていません。

Apache の ActiveMQ を使用した JMS コンポーネントを設定するには

Magic xpi の JMS コンポーネントは、JNDI API を使用している Apache ActiveMQ サーバを使用して動作します。利用するには、ActiveMQ (バージョン 5.5.1) からメッセージを読み書きするためにプロジェクトを適切に設定する必要があります。

1. この例では、**ActiveMQ** は **QueTest** と呼ばれる一つのキューを持っています。
2. **ActiveMQ** と呼ばれるフォルダを Magic xpi のルートに追加します。
3. **lib** フォルダから、**jar** ファイルを **ActiveMQ** フォルダにコピーします。
4. 次に、**jar** ファイルを **Magic.ini** の **[CLASSPATH]** セクションに追加して、サーバがファイルを読み込むようにします。クラスパスのサイズを保存するために、**ActiveMQ** フォルダが追加され、相対パスは使用されます。

以下は、新しいクラスパスの **jar** ファイルの例です。

```
CLASSPATH=ActiveMQ/activation-1.1.jar;ActiveMQ/activemq-camel-5.5.1.jar;ActiveMQ/activemq-console-5.5.1.jar;ActiveMQ/activemq-core-5.5.1.jar;ActiveMQ/activemq-jaas-5.5.1.jar;ActiveMQ/activemq-protobuf-1.1.jar;ActiveMQ/activemq-web-5.5.1.jar;ActiveMQ/camel-core-2.7.0.jar;ActiveMQ/camel-jetty-2.7.0.jar;ActiveMQ/camel-jms-2.7.0.jar;ActiveMQ/camel-spring-2.7.0.jar;ActiveMQ/geronimo-j2ee-management_1.1_spec-1.0.1.jar;ActiveMQ/geronimo-jms_1.1_spec-1.1+1.1.jar;ActiveMQ/geronimo-jta_1.0.1B_spec-1.0.1.jar;ActiveMQ/jaxb-api-2.1.jar;ActiveMQ/jaxb-impl-2.1.6.jar;ActiveMQ/kahadb-5.5.1.jar;ActiveMQ/stax-1.2+0.jar;ActiveMQ/stax-api-1.0.1.jar;ActiveMQ/slf4j-log4j12-1.5.11.jar;ActiveMQ/log4j-1.2.14.jar;ActiveMQ/jcl-over-slf4j-1.5.11.jar;ActiveMQ/slf4j-api-1.5.11.jar;ActiveMQ/OR_JNDI.jar
```

5. 最後の **jar** ファイル (**OR_JNDI.jar** と呼ばれる) は、作成する必要があります。これは、**jndi.properties** ファイル (ActiveMQ ドキュメントによって必要となります) を保持します。例えば、**OR_JNDI.jar** ファイルの内容は次の通りです。

```

java.naming.factory.initial =
org.apache.activemq.jndi.ActiveMQInitialContextFactory
# use the following property to configure the default connector
#java.naming.provider.url = vm://localhost
java.naming.provider.url = tcp://localhost:61616
# use the following property to specify the JNDI name the connection factory
# should appear as.
connectionFactoryNames = connectionFactory, queueConnectionFactory,
topicConnectionFactory
# register some queues in JNDI using the form
# queue.[jndiName] = [physicalName]
queue.QueueTest = QueueTest
# register some topics in JNDI using the form
# topic.[jndiName] = [physicalName]
topic.MyTopic = example.MyTopic

```

6. **Magic.ini** ファイルの [JVM_ARGS] セクションで、以下の値を追加する必要があります。

```

Djava.naming.factory.initial=org.apache.activemq.jndi.ActiveMQInitial+
ContextFactory -Djava.naming.provider.url=tcp://localhost:61616

```

7. 次に、Magic xpi リソースを設定してください。Magic xpi プロジェクトで、リソースを定義します。**Connection Factory Name** パラメータを、**queueConnectionFactory** に設定しなければなりません（これは、**jndi.properties** ファイルの値と一致します）。

8. 最後に、**Queue Name** パラメータを **QueueTest** に設定する必要があります。

参考：

Apache ActiveMQ: <http://activemq.apache.org/>

Ver5.5.5 のダウンロード（このサンプルで使用）

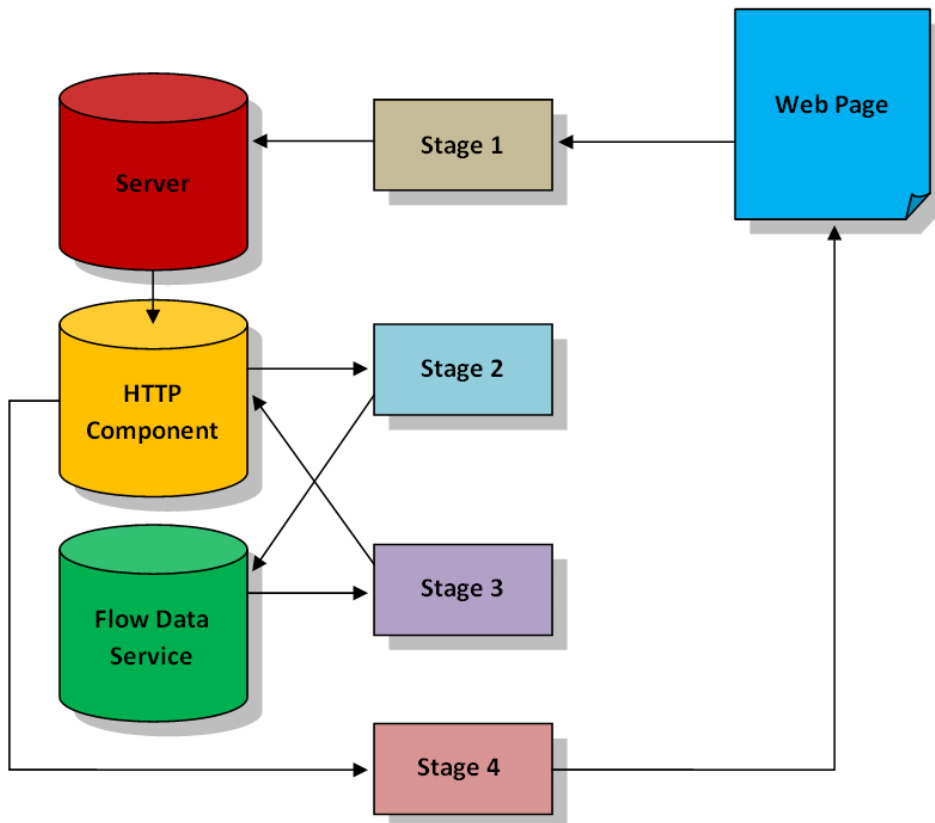
<http://archive.apache.org/dist/activemq/apache-activemq/5.5.1/apache-activemq-5.5.1-bin.zip>

HTTP トリガを作成するには

コンポーネントを Magic xpi エディタのメインエリア、またはトリガコンポーネントセクションに追加することができます。コンポーネントをトリガコンポーネントセクションに含めると、トリガコンポーネントが適切な情報を受信した場合、これらのコンポーネントはフローを起動します。このトピックでは、HTTP コンポーネントをトリガとして使用するフローの作成方法について説明しています。

以下は、適切な情報を HTTP トリガに返すフローとフローデータユーティリティを呼び出す、HTTP トリガを使用するプロセスの作成の実装ステージの説明です。

下の図はどのようにプロセスが動作するかを示しています。



プロセスは、以下の主要なステップを含んでいます。

開始前とステージ 1

- Magic xpi サーバへリクエストを送る Web ページが読み込まれます。
- リクエストは、HTTP トリガコンポーネントを実行するフォーマットで Magic xpi サーバに送られます。

ステージ 2

- HTTP トリガコンポーネントがフローを起動します。

ステージ 3

- フローは様々な"Update"アクションが定義されたフローデータユーティリティを含んでいます。
- フローデータユーティリティが終了すると、フローは HTTP トリガにユーザ BLOB で応答を返します。

ステージ 4

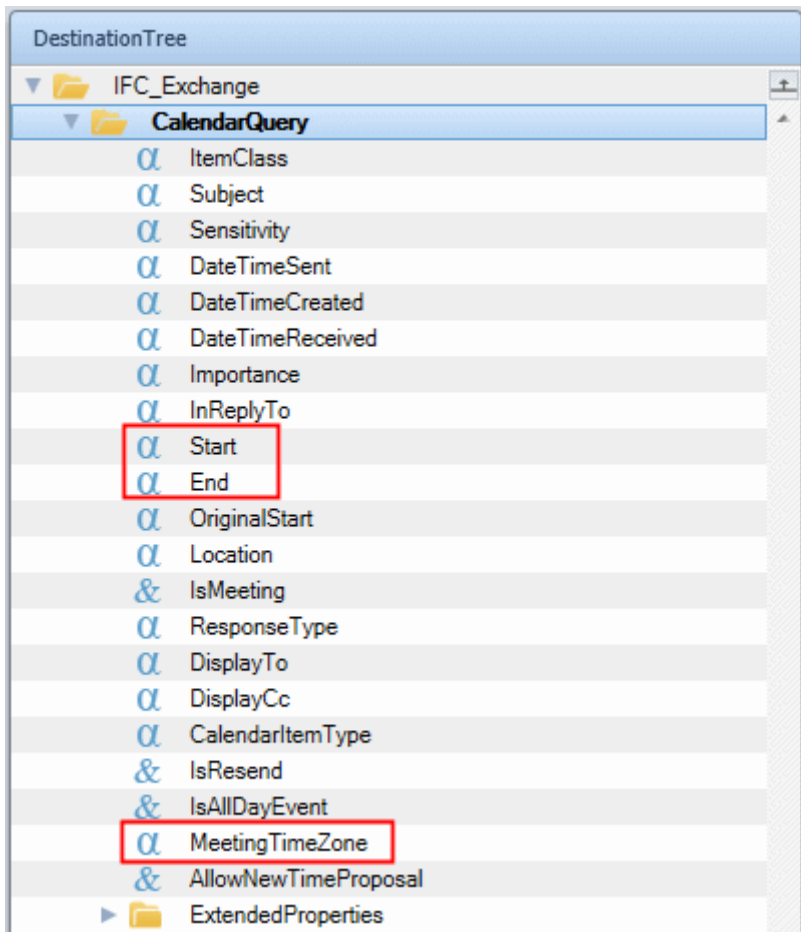
- プロセスが終了し、HTTP トリガは適切な情報を Magic xpi サーバの HTTP リクエストに送ります。



HTTP トリガを使用して動作する時に、戻される HTTP ヘッダをコントロールすることについてのビデオデモンストレーションを参照するには、[ここ](#)をクリックしてください。

Exchange コネクタを使用する時に、任意のタイムゾーンでミーティングを設定するには

Start と End の要素は、XML の dateTime フォーマットを使用します。これは、YYYY-MM-DDTHH:MM:SS または YYYY-MM-DD HH:MM:SS のどちらかです。2 番目の例は、一番目と同じ構文ですが、T が空白に置き換えられます。Start と End の要素では、タイムゾーンのオフセットを含めないでください。オフセットは、MeetingTimeZone 要素に設定する必要があります。



例えば、Start または End の値は、2015-01-31T09:00:00 または 2015-01-31 09:00:00 であるかもしれませんが、これは、XML の dateTime フォーマットで有効な構文であるため、日付と時間の区切り文字がハイフン(-)とコロン(:)であることに注意してください。Start と End 要素に設定した値がこの構文で評価されることを確認してください。

値がデータマップの MeetingTimeZone 要素に設定されない場合、ローカル PC のタイムゾーンが使用されます。値が要素に設定されている場合は、PC のタイムゾーンを上書きします。

明示的に UTC タイムゾーンを設定したい場合は、UTC を MeetingTimeZone 要素に入力してください。

- | | |
|----|--|
| 注 | • MeetingTimeZone 要素は、文字タイプです。 |
| 意： | • daylight saving または standard time は、MeetingTimeZone 要素に設定さ |

れた値に基づいて、自動的に計算されます。

- 以下は、**MeetingTimeZone** 要素で有効な構文です。
 - 以下は、リストに表示される j 値です。 <http://tutorials.jenkov.com/java-date-time/java-util-timezone.html>。値は大文字小文字を区別しません。

または:

- **+hh:ss**. これはタイムゾーンのオフセットです。この部分は、UTC を参照して常に計算されます。ここでは、常にコロン(:)が使用されなければならないことに注意してください。
- 無効な値が設定された場合、実行中を例外が発生します。

MSMQ を使用するには

MSMQ を使用するには

Magic xpi の MSMQ コンポーネントは、MSMQ サーバに接続することができます。MSMQ コンポーネントは、トリガおよびフロー内のステップとして使用することができます。コンポーネントをトリガとして使用する場合は、[ポーリング](#)機能を設定することができます。コンポーネントをステップとして使用する場合は、キューの開始と終了等、様々なアクティビティを実行させることができます。

このコンポーネントをフローエリアで使用した場合、以下のような簡単に使用することができるインターフェースを提供します。

- [ダイレクアクセスメソッド](#)
- [XML](#)

必要条件

MSMQ コンポーネントを使用するには以下の条件が必要です。

- MSMQ がインストールされ実行している。
- プロジェクトで使用されるユーザキューが存在するかの確認が必要です。

MSMQ 実装例

ここでは様々な方法で MSMQ キューにアクセスする手順の例を提供します。この例に基づいてプロジェクトを実行するには、以下のことをする必要があります。

1. MSMQ サービスを開始します。
2. 二つの新しいキューを追加します: **SampleQueue** と **SampleQueue1**
3. プロジェクトに数値タイプの **HandleQueue** というグローバル変数を追加し、デフォルト値を **100** に設定します。

T このトピックには、以下の例があります。

- [Send オペレーション](#)
- [Get オペレーション](#)
- [トリガとしての MSMQ](#)

MSMQ コンポーネントのメソッドインターフェースを使用するには

このセクションは、MSMQ コンポーネントのメソッドインターフェースをどのように使用するかについて説明しています。

1. [設定] ダイアログボックスの [リソース] セクションで **MSMQ** をリソースとして定義してください。詳細は、[ここをクリックしてください](#)。
2. **MSMQ** コンポーネントをフローエリアにドラッグしてください。
3. [インターフェース] を **メソッド** として定義し、[構成] をクリックします。[ダイレクトアクセスメソッド:**MSMQ**] ダイアログボックスが開きます。
4. [新規] をクリックし、使用したいメソッドを選択します。
5. [メソッド詳細] セクションで異なるパラメータの詳細を入力します。太字のパラメータは必須です。
6. 必要に応じて上記のステップ#3 と#4 を繰り返し、メソッドを追加します。

注 意 :

- メソッドは、[名前] カラムの並び順に順次実行されます。
- メッセージを送りたい場合は、その順序で **Open**、**Send**、そして **Close** を使用してください。

MSMQ コンポーネントの XML インターフェースを使用するには

このセクションは、MSMQ コンポーネントの XML インターフェースの使用方法について説明しています。

1. [設定] ダイアログボックスの [リソース] セクションで **MSMQ** をリソースとして定義してください。詳細は、[ここをクリックしてください](#)。
2. **MSMQ** コンポーネントをフローエリアにドラッグしてください。
3. **MSMQ** コンポーネントの [プロパティ] ペインで、インターフェースを **XML** と定義してください。

注 意 : 送り先オブジェクトは既に定義され、設定されています。これは送り先は MSMQ コンポーネントの XSD ファイルに基づいているからです。独自のソースファイルを使用したり、送り先を返す式を使用することもできます。

4. **MSMQ** コンポーネントをダブルクリックするか、右クリックして [設定] をクリックしてください。[データマップ] ウィンドウが開きます。

注 意 :

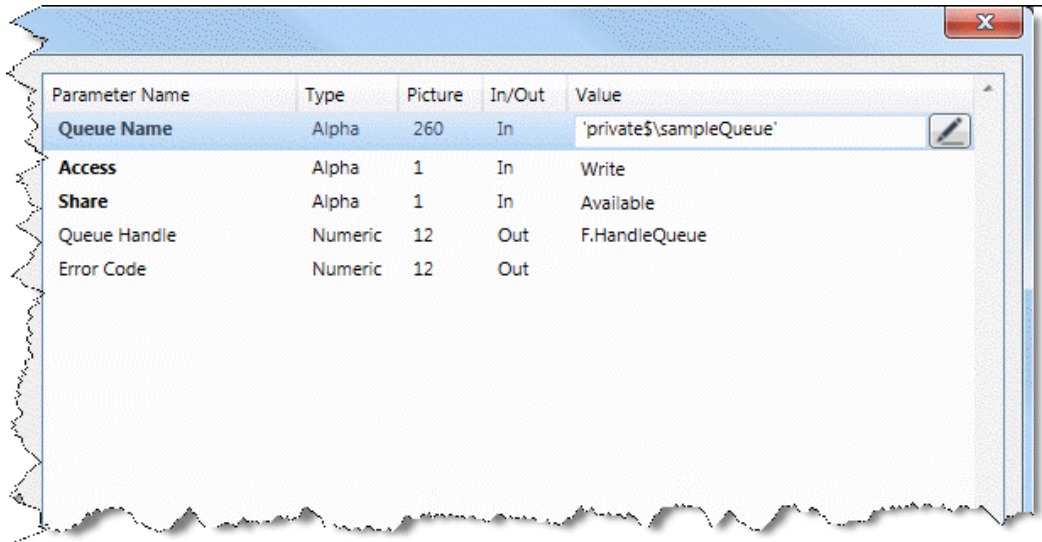
- データマップにはマッピングすることができる全てのパラメータを含んでいます。必須パラメータは表示されません。
- 各メソッドは、複合要素として表示されます。

メッセージをキューに送信するために MSMQ コンポーネントを使用するには

以下のステップは、どのように MSMQ コンポーネントを使用してメッセージをキューに送るかを説明します。

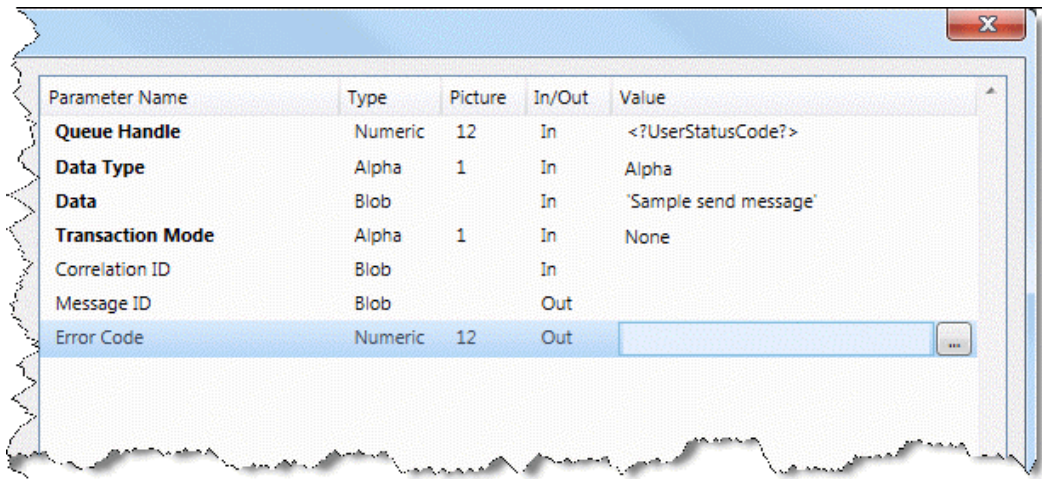
1. [設定] ダイアログボックスの [リソース] セクションで **MSMQ** をリソースとして定義してください。詳細は、[ここをクリックしてください](#)。
2. 新しいフローを作成します。
3. フローエリアに **MSMQ** コンポーネントをドラッグします。 .

4. [プロパティ] ペインの [設定] セクションで、**MSMQ** リソースが選択されていることを確認してください。
5. [プロパティ] ペインの [汎用] セクションで、[インタフェース] をメソッドと定義してください。
6. **MSMQ** コンポーネントをダブルクリックするか、右クリックして [設定] をクリックしてください。
7. [MSMQ 構成] ダイアログボックスで、[追加] をクリックし、**Open** メソッドを追加してください。以下にで表示されているように、パラメータに必要な値を入力してください。

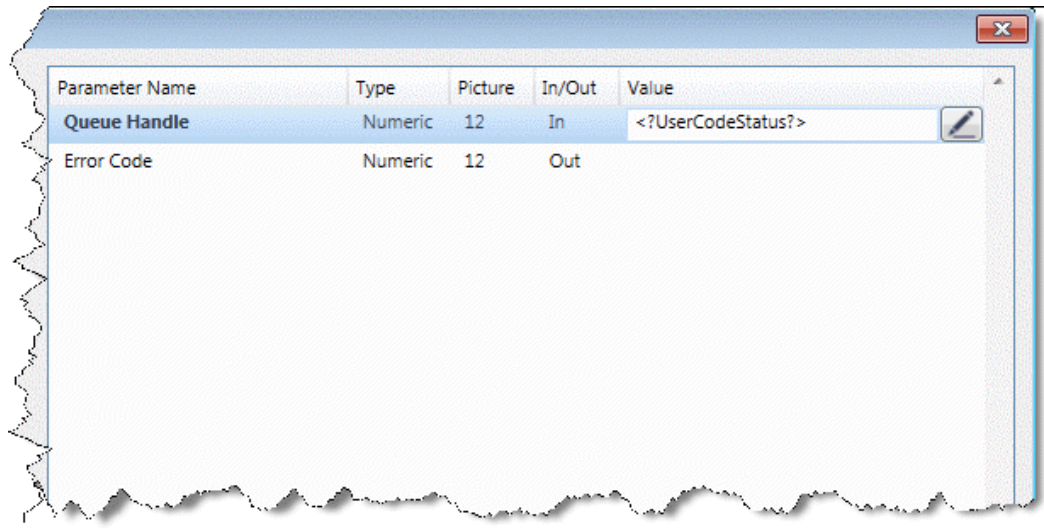


8. [追加] をクリックし **Send** メソッドを追加し、以下のようにパラメータの詳細を入力します。

注意: Queue Handle パラメータは、キューの最初から最後まで同一である必要があります。



9. [追加] をクリックして **Close** メソッドを追加し、以下のようにパラメータを入力します。

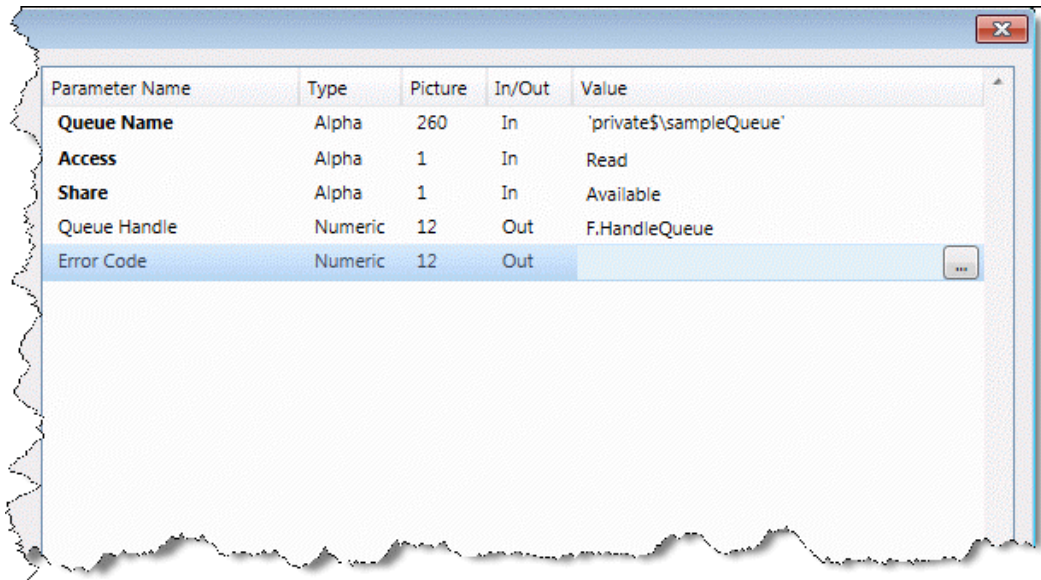


10. [OK] をクリックしてオペレーションの設定を確認します。
11. ソリューションエクスプローラで、フローで右クリックし、コンテキストメニューから [プロパティ] を選択してください。
12. フローの [プロパティ] ダイアログボックスで、**Auto start = Yes** でフローを定義してください。**Auto start = No** のメッセージを送るフローを作成してください。
13. キューの中身を確認するには、**マイコンピュータ** アイコンを右クリックし、[管理] を選択します。
 14. 以下のように進みます。サービスとアプリケーション > メッセージキュー > 専用キュー > SampleQueue > キューメッセージ
15. **Magic xpi モニタ** と **Magic xpi サーバ** を開始します。
16. キューメッセージでメッセージを確認することができるため、プロジェクトの動作をモニタで追跡することができます。

キューからメッセージを受信するために **MSMQ** コンポーネントを使用するには

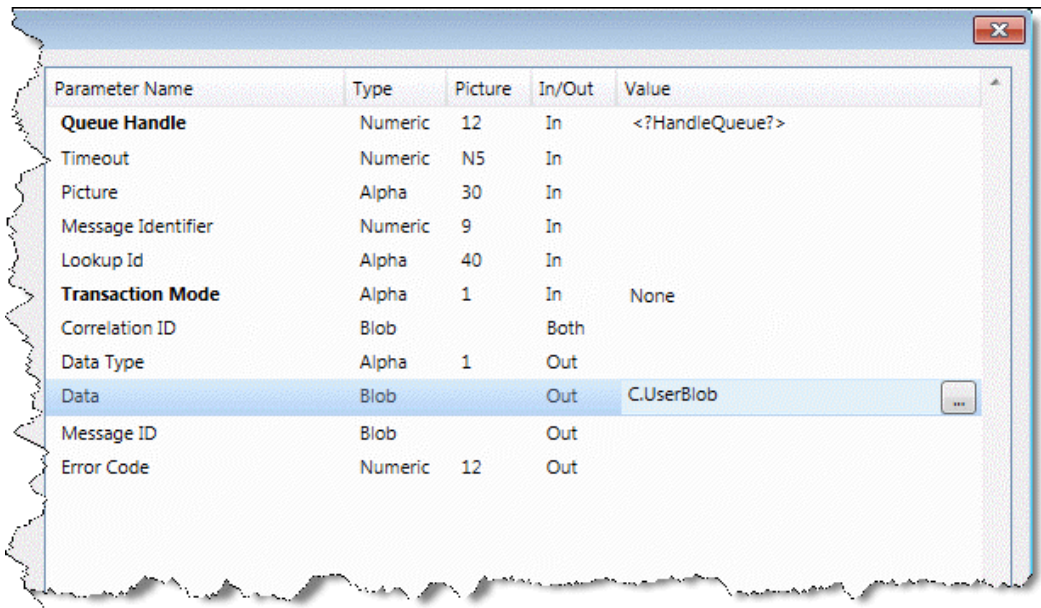
以下のステップは、どのように MSMQ コンポーネントを使用し、キューを取得するかを説明します。

1. [設定] ダイアログボックスの [リソース] セクションで **MSMQ** をリソースとして定義してください。詳細は、ここをクリックしてください。
2. プロジェクトに新規のフローを作成します。
3. フローエリアに **MSMQ** コンポーネントをドラッグします。
4. [プロパティ] ペインの [設定] セクションで、**MSMQ** リソースが選択されていることを確認してください。
5. [プロパティ] ペインの [汎用] セクションで、[インタフェース] をメソッドと定義してください。
6. **MSMQ** コンポーネントをダブルクリックするか、右クリックして [設定] をクリックしてください。
7. [MSMQ 構成] ダイアログボックスで、[追加] をクリックし、**Open** メソッドを追加してください。以下にで表示されているように、パラメータに必要な値を入力してください。

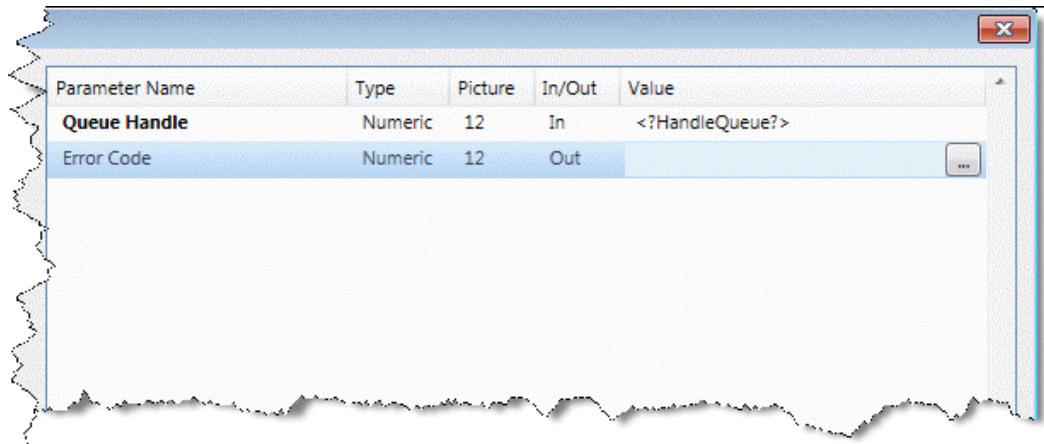


8. [追加] をクリックし **Get** メソッドを追加し、以下のようにパラメータの詳細を入力します。

注意: Queue Handle パラメータは、キューの最初から最後まで同一である必要があります。



9. [追加] をクリックして **Close** メソッドを追加し、以下のようにパラメータを入力します。



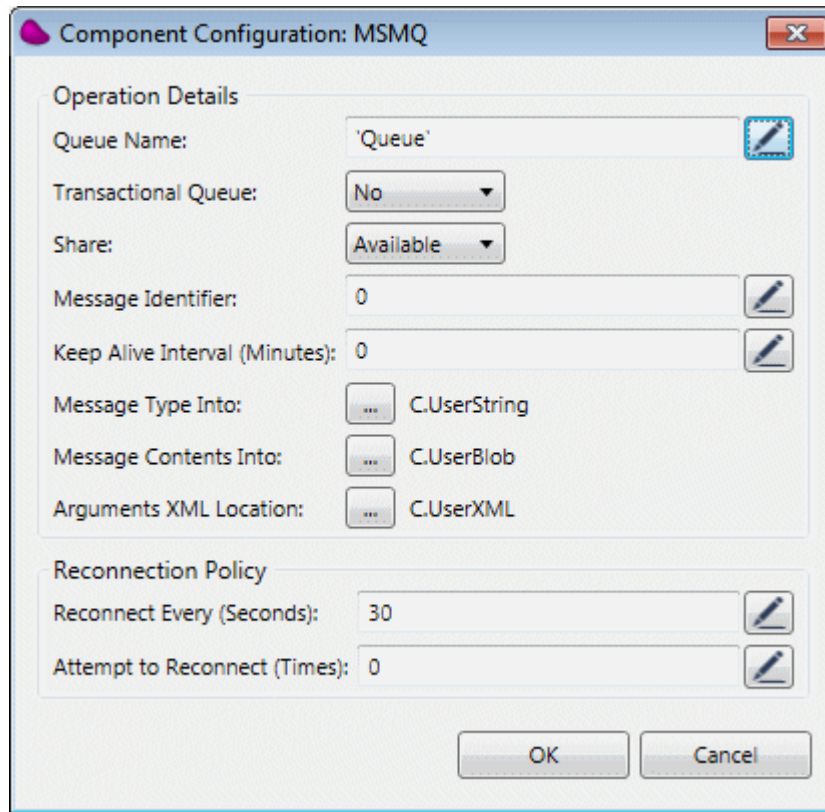
10. [OK] をクリックしてオペレーションの設定を確認します。
11. ソリューションエクスプローラで、フローで右クリックし、コンテキストメニューから [プロパティ] を選択してください。
12. フローの [プロパティ] ダイアログボックスで、**Auto start = Yes** でフローを定義してください。**Auto start = No** のメッセージを送るフローを作成してください。
13. キューの中身を確認するには、**マイコンピュータ**アイコンを右クリックし、[管理] を選択します。
14. 以下のように進みます。サービスとアプリケーション > メッセージキュー > 専用キュー > SampleQueue > キューメッセージ
15. **Magic xpi モニタ**と **Magic xpi サーバ**を開始します。
16. キューメッセージでメッセージを確認することができるため、プロジェクトの動作をモニタで追跡することができます。

トリガとして MSMQ コンポーネントを使用するには

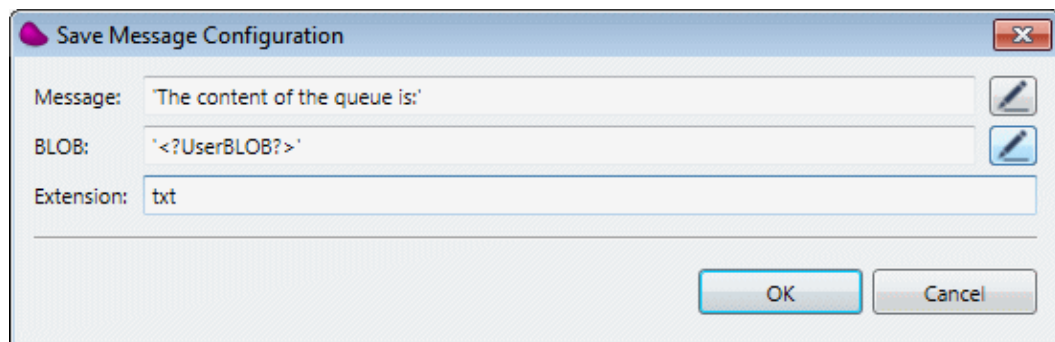
以下のステップは、MSMQ をトリガとしてどのように使用するかを説明しています。

MSMQ コンポーネントをトリガとして使用する場合、受信メッセージによってコンポーネントが実行され、それによってフローが起動されます。メッセージは、ユーザ BLOB 内のフローに送られます。

1. この例の目的のため、キューにメッセージを送るフローを定義する必要があります。フローを作成し、キュー名を **private&\SampleQueue1** に変更します。
2. [設定] ダイアログボックスの [リソース] セクションで **MSMQ** をリソースとして定義してください。詳細は、ここをクリックしてください。
3. [設定] ダイアログボックスの [サービス] セクションで **MSMQ** をサービスとして定義してください。詳細は、ここをクリックしてください。
4. **MSMQ** コンポーネントを **Magic xpi** のトリガエリアにドラッグします。
5. **MSMQ** コンポーネントをダブルクリックするか、右クリックして [設定] をクリックしてください。[コンポーネント構成 : MSMQ] ダイアログボックスが開きます。
6. 以下のようにパラメータの詳細を入力します。パラメータの説明は、こちらです。



7. [OK] をクリックしてオペレーションの設定を確認します。
8. **Save Message** ユーティリティをフローにドラッグし、以下のようにパラメータの詳細を入力します。



9. **Magic xpi モニタ**と **Magic xpi サーバ**を開始します。
10. **Magic xpi モニタ**のユーザメッセージを参照する場所でプロジェクトの動作を追跡することができます。メッセージをダブルクリックして参照することができます。

データマップ

データマップ

以下のトピックは、Magic xpi のでデータマップについて扱っています。

[XML 要素をチェックするには](#)

[データマップのデータベースウィザードを使用しないで、SQL ステートメントを作成するには](#)

[データマップを使用してデータベース処理を実行するには](#)
[DB2/400 データベースに対する言語サポートを作成するには](#)
[ダイナミックなテキストファイルを作成するには](#)
[テーブル内の他のカラムの値でテーブルカラムを更新するには](#)
[フラットファイル送り元から送り先に負数を設定するために、データマップを使用するには](#)
[Multi-Occurrence 複合物から特定のデータを取得するには](#)
[兄弟要素のデータに基づく Multi-Occurrence 複合物から特定のデータを取得するには](#)
[特定のインスタンスにマッピングするには](#)
[特定のインスタンスからマッピングするには](#)
[独立した送り元と順序/行構成にリンクするには](#)
[変換テーブルを使用するには](#)
[データ要素を統一するには](#)
[データ要素を分割するには](#)
[ヘッダと行レコードの両方を含むフラットな構造を階層的な XML にマッピングするには](#)
[送り元上で発生した繰返しをカウントするには](#)
[Multi-Occurrence 複合物から単一データ要素を取り出すには](#)
[データマップの特定の要素を検索するには](#)
[反復可能な複合物の中に、特定の数の項目を作成するには](#)

XML 要素をチェックするには

XML 内に要素が存在するかどうか調べるために、2 つの簡単なオプションがあります。

1. XML 処理コンポーネントの **Check Exists** メソッドの使用。
2. データマップの送り元として XML を使用。[送り先] ペインに論理変数を定義して、送り元内に要素が存在するかどうか調べるために、**XMLExist** 関数を使用します。

データマップのデータベースウィザードを使用しないで、SQL ステートメントを作成するには

Magic xpi データユーティリティは、データのマッピング時に、素早く簡単に使用する SQL 文を作成することができるウィザードを提供しています。開発の自由度を広げるために、ウィザードを使用せず SQL 文を手動で作成することもできます。

手動で SQL 命令文を作成するには、データマップユーティリティをフローにドラッグします。[送り元送り先]ダイアログボックスで送り元、または送り先としてデータベースを選択し、[プロパティ]をクリックし[データベーススキーマプロパティ]ダイアログボックスを開きます。ウィザードを実行し、必要なテーブルとカラムを選択します。次に[データベーススキーマプロパティ]を開き、ダイアログボックスの下部にある[SQL]をクリックし、[SQL 命令文]画面を開きます。基本的なデータベーススクリプトの規則に従って[SQL 命令文]フィールドに SQL 文を入力します。もし現在のオブジェクトがソースデータオブジェクトである場合、Select 文のみ使用できます。もし現在のオブジェクトが送り先データオブジェクトである場合は、Update、Insert、そして Delete 文のみ使用できます。

フロー変数が入力されると、このようにクエションマーク(?)と大小記号(<>)で囲まれます:<?フロー変数名?>。Example や Order By 等、他の有効な SQL 文を使用することもできます。

JOIN を使用して命令文で複数のテーブルを使用することもできます。ウィザードで複数のテーブルが選択された場合、ウィザードでの選択に従い、JOIN 文が自動的に作成されます。

データマッピングユーティリティでの SQL 文の注意点

- 全てのデータベースタイプをサポートするためにステートメントでは ANSI 記号を使用してください。
- データマッピングユーティリティは、ユニオンをサポートしていません。ビジュアルマップでは表示されません。
- データマップは、空白が含まれるフィールド名をサポートしていません。
- 文字列は、シングルクォーテーション(')で囲ってください。
- 他のデータベースに接続するにはデータベースのオーナーに通知してください。接続したいデータベースがプロジェクトで使用する Magic xpi の内部データベースである場合はその必要はありません。
- フィールドや関数にワイルドカードを使用することはできません。個別に指定するかエイリアスを作成してください。

例

この例は手動で SQL 文を作成し、以下のテーブルを更新する方法を説明します(データ マッピングユーティリティのウィザードは使用しません)。

Table_Key	Col1	Col2	Col3
1	A	12	F
2	B	14	Y
3	C	18	W

My_key という数値変数を追加するには：

1. 既存のフローを開くか新規のフローを作成し、データマッピングユーティリティをフローにドラッグします。
2. データマップをダブルクリックするか、右クリックして [設定] をクリックしてください。[データマップ] ウィンドウが開きます。
3. ツールボックスの [マップスキーマ] セクションから、データベースタイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
4. データベースソースの [プロパティ] ペインで、データベース処理とデータベース定義として **Update** を選択してください。
5. [ウィザード] をクリックし、ウィザードを実行します。
6. データベース送り先の [プロパティ] ペインで、[SQL ステートメント画面] を開くために [SQL] をクリックしてください。
7. 以下のステートメントを記述してください。

```
UPDATE ExampleTable SET ExampleTable.Col1='T', ExampleTable.Col2=20, ExampleTable.Col3='X', WHERE ExampleTable.Table_Key=<!?My_Key?!>
```

更新されるレコードは数値変数 **My_Key** の値によって決まります。テーブルカラムは以下の値で更新されます。

- Col1: T
- Col2: 20
- Col3: X

データマップを使用してデータベース処理を実行するには

データマップを使用してデータベース処理を実行するには

Magic xpi のデータマップユーティリティは、視覚的に XML 構造のソースファイルを送り先ファイルにマッピングさせて、XML もしくはデータベーススキーマを送り先 XML、あるいはデータベーススキーマへのマッピングを可能にします。以下のように Select、Insert、Update および Delete のデータベース操作を行うことができます。

- Select SQL ステートメント用のソースデータベーススキーマ
- insert、Update、Delete 操作の送り先データベーススキーマ

このトピックでは、データマップを使用した様々なデータベース操作の設定に必要なステップについて説明しています。

関連トピック：

[データマップでデータベース操作を設定するには](#)

[データマップの SQL ステートメントを作成してマッピング接続を行うには](#)


データマップでデータベース操作を設定するには

このセクションでは、データマップでデータベース運用の設定をする時に必要な情報を提供します。ここで提供されている情報は、どの操作においても共通です。

SQL 文の書き方とマッピング接続の作成の仕方については、[データマップの SQL ステートメントを作成してマッピング接続を行うには](#)を参照しています。


Insert 操作

以下のステップは、Insert 操作として実行するアクションについて説明します。

1. XML ファイルを送り元スキーマとして定義します。
2. データベースを送り先スキーマとして定義します。
3. 送り先スキーマで右クリックして [プロパティの表示] を選択します。
4. [ウィザード] プロパティで、ウィザードを開くために  をクリックしてください。
5. [DB オペレーション] フィールドで、**Insert** を選択してください。
6. 送り先スキーマのために、接続されたデータベースに基づいた有効な Insert の SQL 文を挿入するために、ウィザードを使用してください。
7. 送り元と送り先のオブジェクトを定義した後で、取得された情報を XML ファイルから Insert 文に接続する場所を示す [データマップ] 画面に進めることができます。


Update 操作

以下のステップは、Update 操作として実行するアクションについて説明します。

1. データベースのスキーマを送り元として定義します。この場合の SQL 文は、**Select** です。
2. データベースを送り先スキーマとして定義します。
3. 送り先スキーマで右クリックして [プロパティの表示] を選択します。
4. [ウィザード] プロパティで、ウィザードを開くために  をクリックしてください。
5. [DB オペレーション] フィールドで、**Update** を選択してください。
6. 送り先スキーマのために、接続されたデータベースに基づいた有効な **Update** の SQL 文を挿入するために、ウィザードを使用してください。
7. 送り元と送り先のオブジェクトを定義した後で、取得された情報を XML ファイルから **Update** 文に接続する場所を示す [データマップ] 画面に進めることができます。

Delete 操作

以下のステップは、Delete 操作とし実行するアクションについて説明します。

1. データベースのスキーマを送り元として定義します。この場合の SQL 文は、**Select** です。
2. データベースを送り先スキーマとして定義します。
3. 送り先スキーマで右クリックして [プロパティの表示] を選択します。
4. [ウィザード] プロパティで、ウィザードを開くために  をクリックしてください。
5. [DB オペレーション] フィールドで、**Delete** を選択してください。
6. 送り先スキーマのために、接続されたデータベースに基づいた有効な **Update** の SQL 文を挿入するために、ウィザードを使用してください。
7. 送り元と送り先のオブジェクトを定義した後で、取得された情報を XML ファイルから **Delete** 文に接続する場所を示す [データマップ] 画面に進めることができます。

Select 操作

以下のステップは、Select 操作とし実行するアクションについて説明します。

1. データベースのスキーマを送り元として定義します。この場合の SQL 文は、**Select** です。
2. XML ファイルを送り先スキーマとして定義します。
3. 送り元と送り先のオブジェクトを定義した後で、取得された情報を XML ファイルから **Select** 文に接続する場所を示す [データマップ] 画面に進めることができます。

データマップの SQL 文を作成してマッピング接続を行うには

このセクションでは、データマップを使用するときのように SQL 文を作成するか、そしてどのようにマッピング接続を作成するかについて説明しています。送り元/送り先のデータベースの定義の仕方については、[データマップでデータベース操作を設定するには](#)を参照してください。

SQL 文を準備する

送り元/送り先データベーススキーマは、以下のように定義することができます。

- **ダイレクト SQL ステートメント** SQL 文を手動で入力します。

- **ウィザード** "ウィザード"インターフェースを使用して必要なデータベース、テーブル、カラム等を選択します。

どちらの方法を使用しても結果は同じです。

SQL のいろいろな部分を動的に定義することができます。動的定義の値を以下のように決定することができます。

- **Runtime** グローバル/フロー変数の各タグは実行時しか値を持ちません。SQL 文中のこれらの変数の構文は以下のとおりです。:G.name1
- **Mapping** 動的タイプをどの送り元要素もしくは定数値から定義することもできます。SQL 文中のこれらの変数の構文は以下のとおりです。MyDel

以下は送り元/送り先のスキーマ用の様々な SQL 命令文の例です。

Insert 文

```
INSERT INTO customer
(city,fax_area,fax_num,house_num,name,number_id,phone_area,
phone_descr,phone_num,street,zip_code)
```

Update 文

```
SELECT customer.city,customer.street FROM customer WHERE
customer.name = G.name1
```

```
UPDATE customer SET city=city, street=street
WHERE customer.name = G.name2
```

Delete 文

```
SELECT customer.city FROM customer WHERE customer.city =
G.city
```

```
DELETE FROM customer WHERE customer.city = MyDel
```

Select 文

```
SELECT
customer.city,customer.fax_area,customer.fax_num,customer.house_num,customer.name,customer.number_id,customer.ph
FROM customer
```

マッピング接続を準備する

送り元/送り先スキーマを定義した後、マッピング段階に進むことができます。これは送り元要素(送り元スキーマから取り出された情報)と送り先要素を結びつける時です。

Magic xpi 逆引き辞典

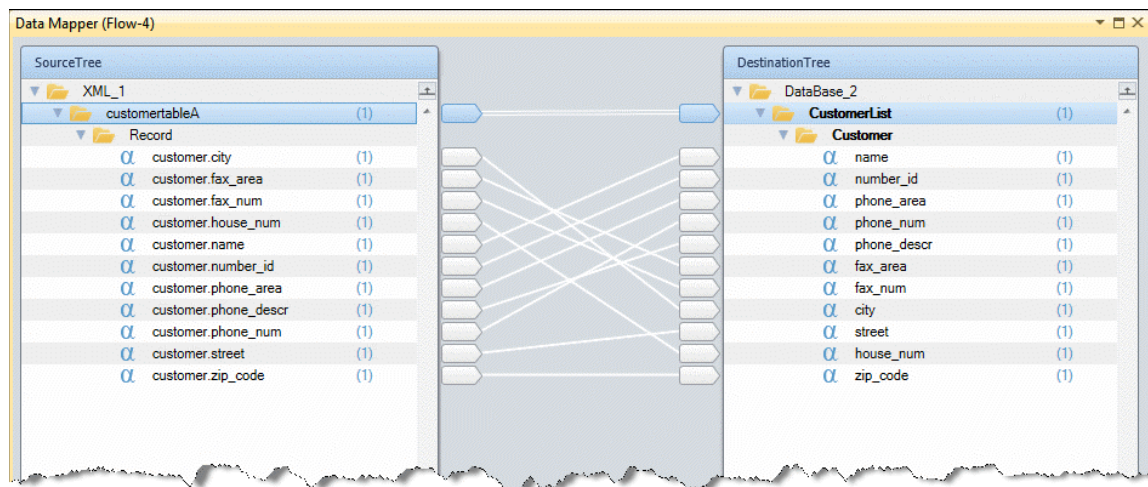
送り元と送り先のスキーマの間に様々なタイプの接続を定義することができます。以下がデータベース形式のスキーマに適切な接続です。

- **Simple Connection** 単一複合要素と送り先の単一複合要素を接続します。
- **Complex Connection** ひとつの複合要素と他の複合要素を接続します。
- **Dynamic Tag (Mapping) Connection** 動的タグ定義と定数、他の要素もしくはフ로그ローバル変数を接続します。

以下の図はサンプルプロジェクトで見ることができる様々なマッピング接続の例です。

Insert マッピング

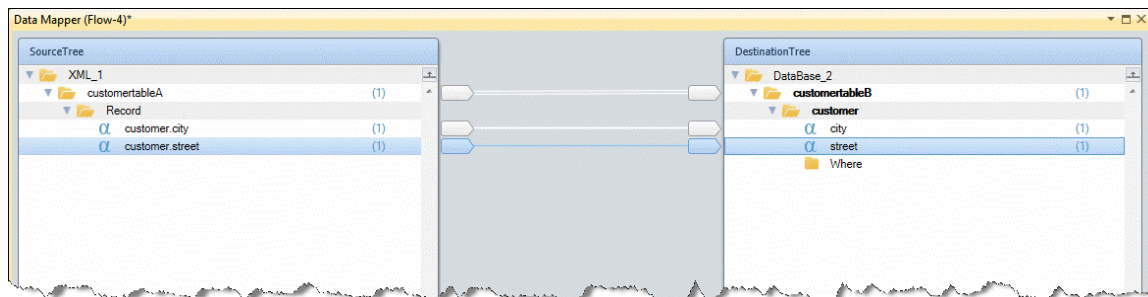
Insert 操作においては、単一要素が送り元 XML スキーマから送り先データベーススキーマへマッピングされます。



注 送り先スキーマの **Phone_desc** 要素はそれ自身に接続されます。つまり実行時に入力された値が計算された式から得られたということです。

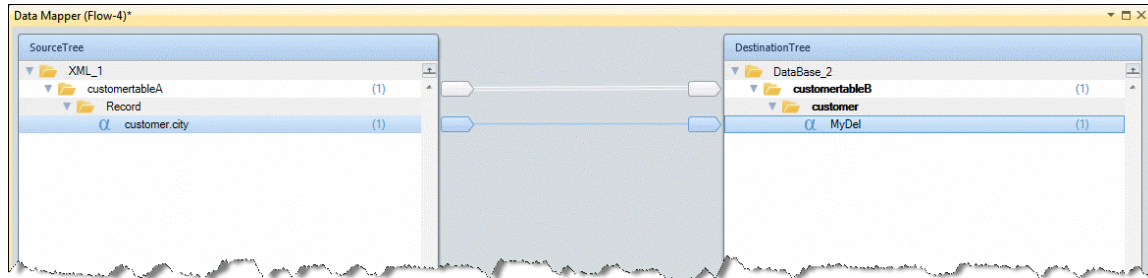
Update マッピング

Update 操作においては、送り先スキーマの **City** と **Street** 要素は動的タグです。これらのタグは UpdateSql 文の一部であり、SQL 文の定義の一部としてではなく、マッピングプロセス中に実際の値を受け取ります。



Delete マッピング

Delete 操作においては、送り先スキーマの **MyDel** 要素が動的なタグです。このタグは Delete 文と Where 句の一部であり、SQL 文の定義内ではなく、マッピング処理中に実際の値を受け取ります。



DB2/400 データベースに対する言語サポートを作成するには

以下の手順は、DB2 for IBM i データベース管理システムに言語サポートを追加するために必要な手順です。これらの手順は、Magic xpi スタジオとデータマップパサーサービスがインストールされている Windows OS 上で実行されます。

これらのステップは、データマップユーティリティが DB2 for IBM i データベースから、または DB2 for IBM i データベースへデータをマッピングする場合に必要なステップです。DB2 for IBM i データベースを使用してデータをマッピングするには、Windows PC は、DB2 for IBM i データベースがインストールされている IBM i サーバと同じネットワークになければなりません。

以下の変更を **Magic.ini** 設定ファイルに加えてください。

1. **Magic.ini** ファイルの [MAGIC_DBMS]へ進んでください。
2. **AS400** から始まる行で、**LANGUAGE=AS2EB.ENG** というエントリを探してください。もしシステム言語が英語でない場合は、*.ENG ファイル拡張子を DB2 for IBM i でサポートされている言語に対応するように変更してください。
3. **AS400** の行の最後にカンマ(,)を追加し、正しい **ASCII/EBCDIC** 変換パスを入力してください。例えば、日本語の場合は **support/ACS400.JPN** を追加します。
4. **DB2 for IBM i** から始まる行の最後にカンマ(,)を追加し、正しい **ASCII/EBCDIC** 変換パスを入力してください。例えば、日本語の場合は **support/ACS400.JPN** を追加します。

注: ステップ3とステップ4で **ASCII/EBCDIC** 翻訳パスに使用されている拡張子は、ステップ2で **LANGUAGE** エントリに追加された拡張子と同じでなければなりません。

5. カンマ (,) をもうひとつ追加し、正しい **ACS** ファイルを追加してください。例えば、日本語の場合、**support/ACS400.JPN** を追加します。
6. [MAGIC_LOGICAL_NAMES] セクションへ進みます。
7. **date_format** 設定をプラットフォームの日付フォーマットに変更します。例えば、IBM i の日付フォーマットは **YYYY-MM-DD** です。

以下の変更を Windows 環境設定変数に追加します。

1. [マイコンピュータ] アイコンで右クリックし、**プロパティ**を選択し、Windows のシステムプロパティを開きます。
2. [詳細設定] タブをクリックします。
3. [環境変数] をクリックします。

4. <コンピューター名>セクションのユーザ変数で **Magic xpi_LANGUAGE** を選択し [編集] をクリックします。
5. もしこの変数がリストにない場合、[新規] をクリックして **Magic xpi_LANGUAGE** を [変数名] フィールドに追加します。[編集] をクリックする必要はありません。次のステップで説明しているように [変数値] フィールドにパスを追加し、[OK] をクリックしダイアログボックスを閉じ最後のステップに進みます。
6. 正しい ASCII/EDBCDIC 翻訳パスを [変数値] フィールドに追加します。例えば日本語を使用する場合、**support/AS2EB.JPN** を追加します。
7. [OK] をクリックしダイアログボックスを閉じます。
8. [OK] をクリックし、[Windows 環境変数] ダイアログボックスを閉じます。

ユーザ ID を設定する

以下をチェックし、データマッピングユーティリティが動作しているかを確認してください。

- DB2 for IBM i データベースへそして DB2 for IBM i データベースからデータをマッピングするには、データベースの情報を [設定] ダイアログボックスの [リソース] セクションに入力しなければなりません。[リソース] セクションの [ユーザ] フィールドに入力された名前が、DB2 for IBM i データベースのある IBM i マシン上に存在していること確認してください。

IBM i 用 Magic xpi サーバを設定する

DB2 for IBM i データベースがインストールされる IBM i マシン上で、[リソース] セクションのデータベースエントリに入力されたユーザのユーザプロフィールが符号化された文字セット ID オプションの中に 65535 という数値が含まれていないことを確認してください。

ダイナミックなテキストファイルを作成するには

データマッピングユーティリティは、様々な送り元からのデータを様々なタイプのテキストファイルにマッピングできます。データをテキストファイルにマッピングするには、データ要素を定義する Magic xpi マージタグを使用してテンプレートファイルを作成する必要があります。データマッピングユーティリティは、このマージタグを使用して様々なタイプの送り元からデータをマッピングします。

注	マージタグが正規の ASCII 文字列として記述されている限り、テンプレートファイルはどのようなフォーマットでも構いません。
----------	--

もしテンプレートファイルをデータマッピングユーティリティの送り先と定義した場合、マージ機能はテンプレートファイル内のマージタグをスキャンします。スキャンされたタグは、[データマッピング] 画面の送り先セクションに表示されます。全ての送り元からこれらのタグにマッピングすることが可能です。例えば、XML あるいはデータベースのフィールドをタグにマッピングすることができます。

プロジェクトを実行すると、タグは定義した送り元マッピングに応じた値に置き換えられます。

ここには、いくつかのサンプルマージタグがあります。

<!**\$MG_NAME**> …… このタグはデータマッピングで定義されたデータ要素にマッチするタグ名を定義します。NAME の部分をタグの名前で置き換える必要があります。データマッピングがタグをスキャンした時に一致するものが見つかった場合、その値でタグを置き換えます。

<!**\$MGREPEAT**> …… このタグは繰り返し部分の開始場所を定義します。繰り返し部分は処理ごとに複製されるため数値が不明な行を扱うことができます。このタグは出力からは削除されます。

<!**\$MGENDREPEAT**> …… このタグは繰り返し部分の終了場所を定義します。このタグは出力からは削除されます。

例

この例はデータベーステーブル内容をテキストファイルに変換する方法を説明しています。最初はテンプレートを作成し、次にデータマッピングユーティリティを設定します。例として以下のサンプルテーブルを使用します。

Customer_ID	Order_ID	Product	Quantity
1	1	Chair	12
1	2	Table	12

ステップ 1：テンプレートファイルを作成する

以下の形式のテキストファイルを作成します。

ID,Order_ID,Product,Quantity

<!\$MGREPEAT>

<!\$MG_ID>,<!\$MG_OrderID>,<!\$MG_Product>,<!\$MG_QTY>

<!\$MGENDREPEAT>

ステップ 2：データマッピングユーティリティを設定する

以下のステップは、データマッピングユーティリティの設定の仕方を説明します。

1. データマッピングユーティリティをフローにドラッグします。
2. データマップをダブルクリックするか、右クリックして [設定] をクリックしてください。 [データマップ] ウィンドウが開きます。
3. ツールボックスの [マップスキーマ] セクションから、データベースタイプを [データマップ] ウィンドウの送り元エリアにドラッグしてください。
4. [データベース定義] ドロップダウンリストで、データベースを選択してください。
5. データベースウィザードを開くために、[ウィザード] をクリックしてください。
6. [有効なテーブル] の下の [テーブル選択] 画面で、サンプルテーブルやその他のテーブルを選択し、[追加] をクリックしてください。次に [次へ] をクリックして、[カラムの選択] 画面が開きます。
7. [全て追加] をクリックし、テーブルのカラムを [選択されたカラム] セクションに移動し、[b] をクリックして [Where 句] 画面を開いてください。
8. [次へ] をクリックして [ウィザード結果] ダイアログに移動します。次に [完了] をクリックしてウィザードを閉じてください。
9. ツールボックスの [マップスキーマ] セクションから、テンプレートタイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
10. テンプレート送り先の [プロパティ] ペインの [テンプレートファイル] フィールドで、ステップ 1 で作成したテンプレートファイルの位置を入力してください。
11. [データ送り先] のリストボックスで、ファイルを選択し、ファイルの位置を入力してください。次に [保存] をクリックしてください。
12. [データマップ] ウィンドウで、送り元内のテーブルカラムを適切な送り先にマッピングしてください。

13. プロジェクトを保存し実行してください。

テーブル内の他のカラムの値でテーブルカラムを更新するには

Magic xpi データマッピングユーティリティは、データベーステーブルのフィールドを更新することができます。同じテーブル内のフィールドを更新式で使用する場合、データマップの送り元、送り先の両方に同じテーブルを定義します。この場合、送り先に **Update** 操作を使用し、送り元で **Select** 操作を使用します。送り元から送り先へ必要なフィールドをマッピングします。

同じテーブルの別のカラムの値でテーブルのカラムを更新する別の方法は、データベースウィザードの最後の画面の中で SQL 文を変更することです(詳細は、[こちら](#)を参照してください)。

例

この例では Col1 の値を Col1 と Col2 の値の合計で更新します。


My_Key	Col1	Col2
1	12	56
2	666	19
15	23	97

送り元を定義する

1. Magic xpi で既存のフローを開くか新規のフローを作成します。
2. データマッピングユーティリティをフローにドラッグします。
3. データマップをダブルクリックするか、右クリックして [設定] をクリックしてください。[データマップ] ウィンドウが開きます。
4. ツールボックスの [マップスキーマ] セクションから、データベースタイプを [データマップ] ウィンドウの送り元エリアにドラッグしてください。
5. [データベース定義] ドロップダウンリストで、データベースを選択してください。
6. データベースウィザードを開くために、[ウィザード] をクリックしてください。
7. [テーブル選択] 画面で、[有効なテーブル] リストから **Example_Table** 選択し、[追加] をクリックして [選択されたテーブル] リストに追加します。[次へ] をクリックして [カラム選択] 画面を開きます。
8. Col1 を [有効なカラム] リストから選択し、[追加] をクリックし [選択カラム] リストに追加します。[次へ] をクリックして、[Where 句] ダイアログを開きます。
9. [次へ] をクリックして、[ウィザードの結果] ダイアログを開きます。[完了] をクリックして、ウィザードを閉じ [データマップ] ウィンドウに戻ります。

送り先を定義する

1. ツールボックスの [マップスキーマ] セクションから、データベースタイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
2. データベースウィザードを開くために、[ウィザード] をクリックしてください。
3. [DB オペレーション] リストで **Update** を選択します。

4. [テーブル選択] 画面で、[有効なテーブル] リストから **Example_Table** 選択し、[追加] をクリックして [選択されたテーブル] リストに追加します。[次へ] をクリックして [Where 句] ダイアログを開きます。
5. [次へ] をクリックして、[ウィザードの結果] ダイアログを開きます。[完了] をクリックして、ウィザードを閉じ [データマップ] ウィンドウに戻ります。
6. [データマップ] ウィンドウで、**Col1** と **Col2** を送り元から送り先の **Col1** にマッピングしてください。
7. 送り先リストの **Col1** で右クリックしてください。
8. [プロパティ] ペインで、[計算値] フィールドの右側の  をクリックして式エディタを開いてください。
9. 以下を行って、式を作成してください。
 - a. **Col1** を変数リストから選択します。
 - b. **+** を演算子リストから選択します。
 - c. **Col2** を変数リストから選択します。

作成した式は [式エディタ] の下側の [式] ウィンドウに表示されます。以下のように表示されます。

Col1+Col2

この処理は Col1 の値を Col1 と Col2 の値の和で更新します。

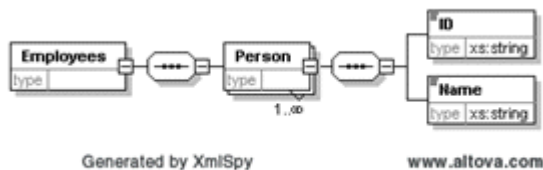
フラットファイル送り元から送り先に負数を設定するために、データマップを使用するには

送り先に負数かもしれない数値を持つフラットファイルにマッピングする必要があるとき、数値データタイプのフィールドを追加して、書式を 12 から N12 に修正することで、送り元のプロパティを修正してください。

Multi-Occurrence 複合物から特定のデータを取得するには

Multi-Occurrence 複合物から指定したデータを取得して、条件付マッピングを用いてフィルタリングするために、変数を使用することができます。

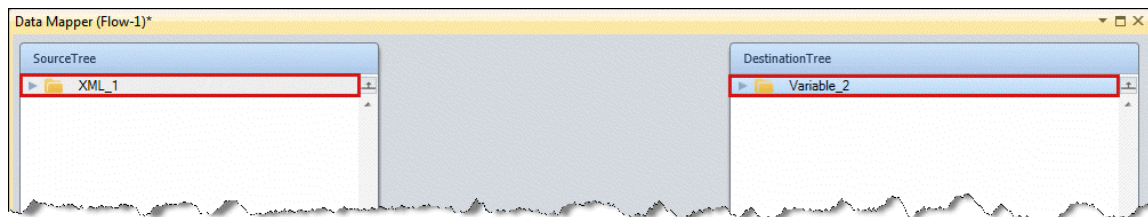
スキーマの例：



XML ドキュメントの例：

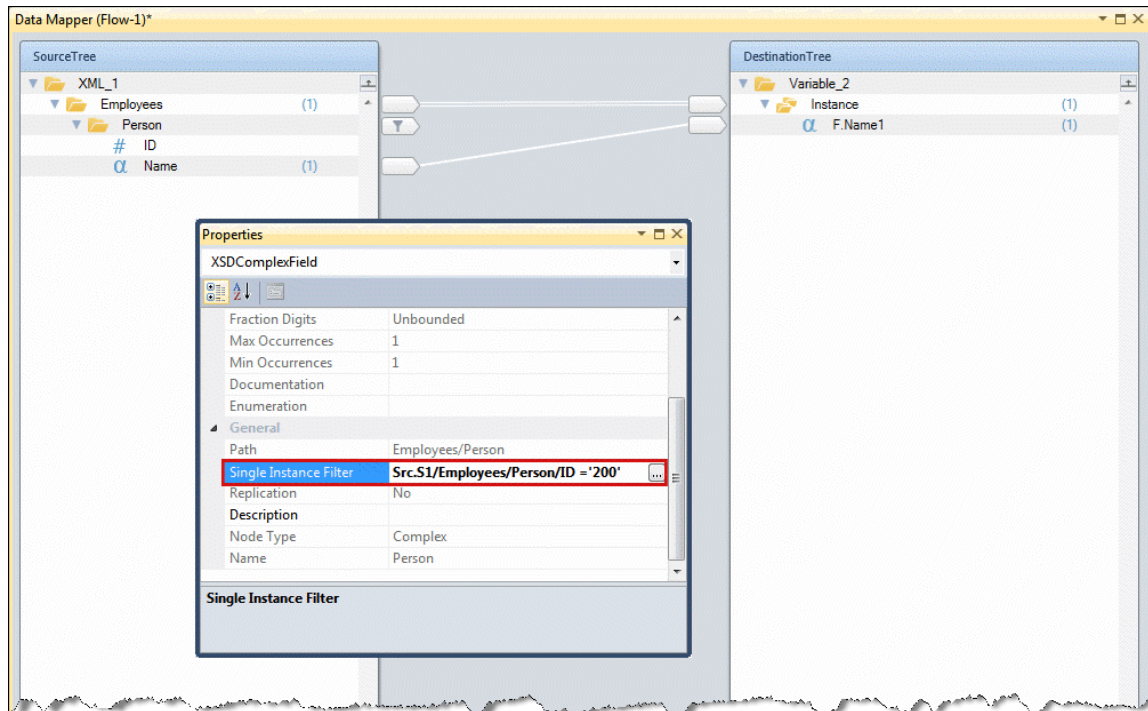
```
<?xml version="1.0" encoding="UTF-8"?>
<Employees xsi:noNamespaceSchemaLocation="filter10.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Person>
    <ID>100</ID>
    <Name>David</Name>
  </Person>
  <Person>
    <ID>200</ID>
    <Name>Alice</Name>
  </Person>
  <Person>
    <ID>300</ID>
    <Name>John</Name>
  </Person>
</Employees>
```

1. 取得されたデータを保持する数を定義してください。
2. データマッピングユーティリティをフローエリアにドラッグしてください。
3. ツールボックスの [マッピングスキーマ] セクションから、XML タイプを [データマッピング] ウィンドウの送り元エリアにドラッグしてください。
4. XML 送り元の [プロパティ] ペインで、サンプル XML 文書を選択してください。
5. ツールボックスの [マッピングスキーマ] セクションから、変数タイプを [データマッピング] ウィンドウの送り先エリアにドラッグしてください。
6. 変数の送り先の [プロパティ] ペインで、以前に定義した変数を選択してください。



7. [データマッピング] 画面で、送り元の Person 複合物の [single instance filter] を以下の値で定義してください。

Src.S1/Employees/Person/ID =' 200'



8. 送り元の **Name** 要素を送り先の **F.Name** 変数にマッピングしてください。

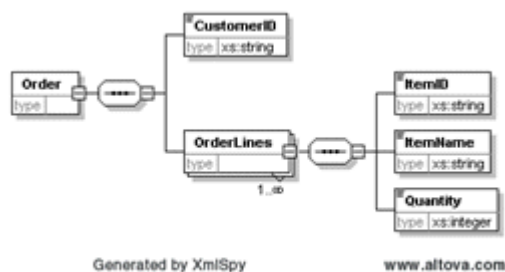
サンプル XML ドキュメントに基づいて、一旦マッピングが実行されると、変数 **F.Name** の値は **Alice** でなければなりません。

兄弟要素のデータに基づく **Multi-Occurrence** 複合物から特定のデータを取得するには

兄弟要素のデータに基づく Multi-Occurrence 複合物から必要なフィルタリング・データを取得するために、変数を使用することができます。次に、条件付マッピングを使用してフィルタリングを行うために、この変数を使用してください。

フィルタ値は、Multi-Occurrence 複合物からデータセットを取得するために連結されるかもしれません。

スキーマの例：



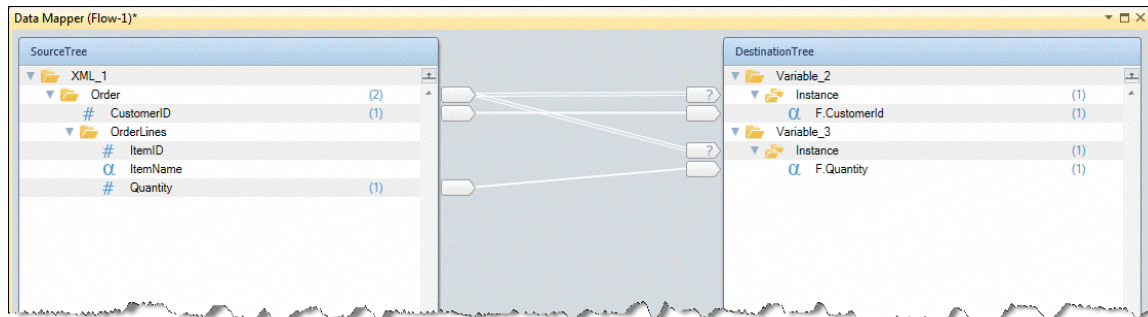
XML ドキュメントの例：

```
<?xml version="1.0" encoding="UTF-8"?>
<Order xsi:noNamespaceSchemaLocation="filter20.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <CustomerID>100</CustomerID>
  <OrderLines>
    <ItemID>200</ItemID>
    <ItemName>Knife</ItemName>
    <Quantity>2</Quantity>
  </OrderLines>
  <OrderLines>
    <ItemID>210</ItemID>
    <ItemName>Fork</ItemName>
    <Quantity>5</Quantity>
  </OrderLines>
</Order>
```

1. 取得されたデータを保持する数を定義してください。
2. データマッピングユーティリティをフローエリアにドラッグしてください。
3. ツールボックスの [マッピングスキーマ] セクションから、XML タイプを [データマッピング] ウィンドウの送り元エリアにドラッグしてください。
4. XML 送り元の [プロパティ] ペインで、サンプル XML 文書を選択してください。
5. ツールボックスの [マッピングスキーマ] セクションから、変数タイプを [データマッピング] ウィンドウの送り先エリアにドラッグしてください。
6. 最初の送り先は、**F.CustomerId** 変数を選択します。2 番目の送り先には、以下に示しているように **F.Quantity** 変数を選択します。



7. [データマッピング] 画面で、送り元 XML 複合物要素を最初の転送先の複合物要素に接続してください。これは変数の親になります。
8. 送り元上で一つ以上の要素に基づくフィルタ条件を定義してください。例：
Src.S1/Order/CustomerID ='100'
9. 送り先の変数に抽出したい単純要素を接続してください。必要に応じて、式を追加してください。
10. 子送り元の XML 複合物要素を 2 番目の送り先の複合物要素 (変数の親) に接続してください。
11. 送り元内の一つ以上の要素に基づいた **OrderLines** マッピング上で、フィルタ条件を定義してください。最適化のために、親から抽出される値に基づく論理式を含めることができます。
例：**Src.S1/OrderOrderLines/ItemID ='210' AND F.CustomerId <> ""**
12. 以下に示すように、送り先の変数を取得しようとする単純要素に接続してください。必要に応じて、式を追加してください。

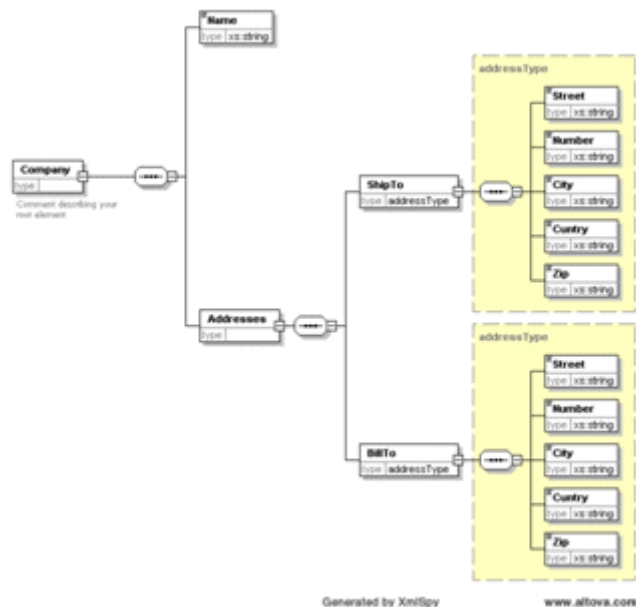


注 類似したケースとして、値が複数の送り元から来た場合、一連のイベントを保証することができないため、プロセスを2つの個別のデータマップに分割することが最善の方法です。

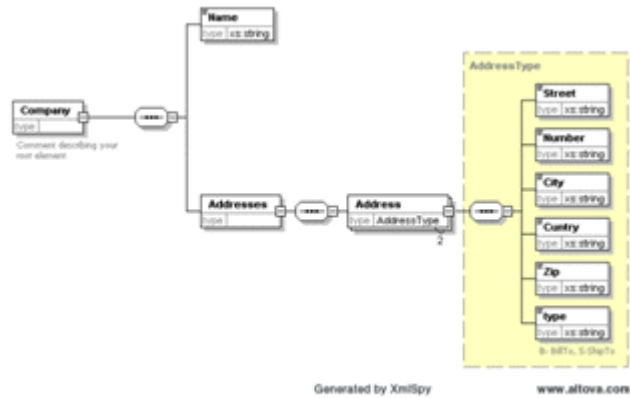
特定のインスタンスにマッピングするには

single-occurrence 複合物を Multi-Occurrence 複合物にマッピングするには、送り先合成物を増やすために、Instance Creation を使用してください。次に、ソースデータを適切に作成された複合物にマッピングしてください。

入力 **XML** ドキュメントのためのスキーマ例：



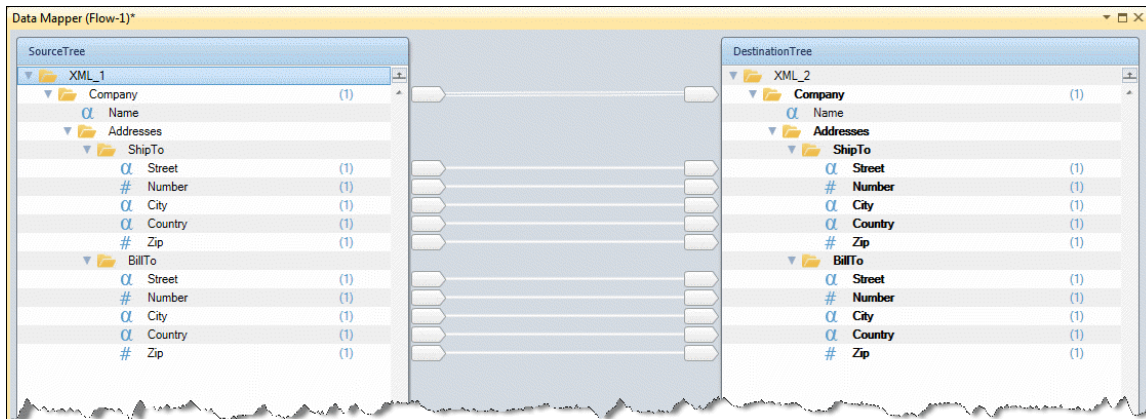
出力 **XML** ドキュメントのためのスキーマ例：



XML ドキュメントの例：

```
<?xml version="1.0" encoding="UTF-8"?>
<Company xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="CompanySource.xsd">
  <Name>StringName</Name>
  <Addresses>
    <ShipTo>
      <Street>String1</Street>
      <Number>String2</Number>
      <City>String3</City>
      <Country>String4</Country>
      <Zip>String5</Zip>
    </ShipTo>
    <ShipTo>
      <Street>String6</Street>
      <Number>String7</Number>
      <City>String8</City>
      <Country>String9</Country>
      <Zip>String10</Zip>
    </ShipTo>
  </Addresses>
</Company>
```

1. データマッピングユーティリティをフローエリアにドラッグしてください。
2. ツールボックスの [マップスキーマ] セクションから、XML タイプを [データマップ] ウィンドウの送り元エリアにドラッグしてください。
3. XML 送り元の [プロパティ] ペインで、サンプル XML 文書を選択してください。
4. ツールボックスの [マップスキーマ] セクションから、multi-occurrence 複合物(例えば、XML)をサポートするスキーマタイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
5. 送り先の [プロパティ] ペインで、サンプル XML 文書を選択してください。
6. [データマップ] 画面で、以下のように表示されたマッピングを定義してください。

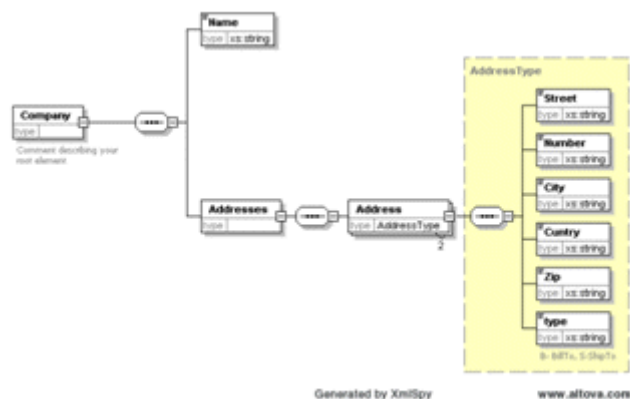


7. 送り先の複合物の [プロパティ] ペインで、[レプリケーション] パラメータのドロップダウンリストから **Yes** を選択して、[インスタンス作成] を有効にしてください。
8. [データマップ] 画面で複合物名が **Address#1** に変更されたことを確認してください。
9. 複合物ノードのコンテキストメニューから、[レプリケーション追加] を選択し、必要なレプリケーション数を選択してください。送り元内の単純要素を各々の指定した送り先インスタンスに接続してください。
 - 例：ShipTo 送り元の単純要素を指定した送り先インスタンス内の要素に接続してください。

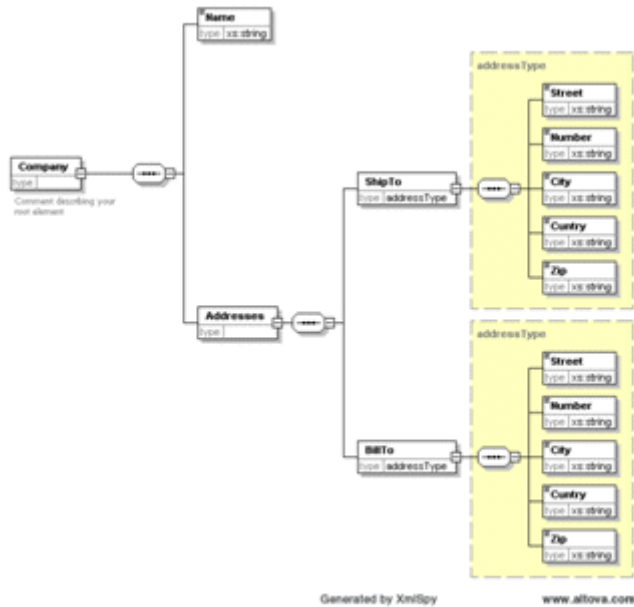
特定のインスタンスからマッピングするには

Multi-Occurrence 複合物を single-occurrence 要素にマッピングするには、送り元複合物を増やすために、レプリケーション機能を使用してください。次に、送り元データを送り先複合物にマッピングしてください。

入力 XML ドキュメントのためのスキーマ例：



出力 XML ドキュメントのためのスキーマ例：



XML ドキュメントの例:

```
<?xml version="1.0" encoding="UTF-8"?>
<Company xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="CompanySource.xsd">
  <Name>StringName</Name>
  <Addresses>
    <ShipTo>
      <Street>String1</Street>
      <Number>String2</Number>
      <City>String3</City>
      <Country>String4</Country>
      <Zip>String5</Zip>
    </ShipTo>
    <BillTo>
      <Street>String6</Street>
      <Number>String7</Number>
      <City>String8</City>
      <Country>String9</Country>
      <Zip>String10</Zip>
    </BillTo>
  </Addresses>
</Company>
```

1. データマップステップを作成してください。
2. 取得されたスキーマに基づいて、**XML** 送り元タイプを定義してください。
3. 取得された送り先スキーマに基づいて、**XML** 送り先タイプを定義してください。
4. [データマップ] 画面で、以下を実行してください。
 - 送り元複合物の [プロパティ] ペインの [レプリケーション] パラメータのドロップダウンリストから **Yes** を選択することで、[インスタンス作成] を有効にしてください。
 - [データマップ] 画面で複合物名が **Address#1** に変更されたことを確認してください。

- 複合ノードのコンテキストメニューから、[レプリケーション追加] を選択し、必要なインスタンス数を選択してください。
5. 複製された送り元要素を送り先の単純要素に接続してください。例：
- 単純な **Address** 要素を **ShipTo** に接続して、2 番目の要素を **BillTo** 送り先要素に接続してください。

注 送り元内のレプリケーションは、最初と 2 番目の複合物を提供します。レプリケーションインスタンス毎に必要な複合物が選択可能で有効なフィルタリングはありません。

独立した送り元と順序/行構成にリンクするには

2 つの独立した multi-occurrence 送り元の間で論理リンクを作成して、論理条件に合う複合データを取得するために、フラットファイルの multi-occurrence 送り元を経由して実行してください。繰り返しのために、SELECT 文がフラットファイルからデータベースに multi-occurrence の行送り元のために、フィルタ値付きで送られます。

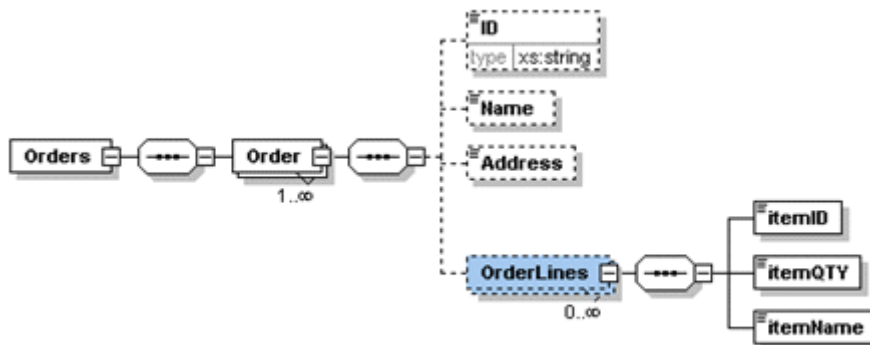
送り元フラットファイルの例

```
OrderID, Name, Address
1,Name1, address1
2,Name2, address2
```

データベーステーブルの例

orderID	itemID	name	qty
1	11	item1	2
1	33	item3	4
1	44	item4	1
2	22	item2	3
2	55	item5	5

送り先 XSD



1. データマッピングユーティリティをフローエリアにドラッグしてください。

2. ツールボックスの [マップスキーマ] セクションから、フラットファイルタイプとデータベースタイプを [データマップ] ウィンドウの送り元エリアにドラッグしてください。
3. 与えられたフラットファイルとテーブルの構造に従って、フラットファイルとデータベースソースを定義してください。
4. [データベースソース] プロパティに行き、ウィザードを開始してください。 **itemID**、**name**、および **qty** フィールドを **Select** し、**orderId=C.UserString** という **WHERE** 句を定義してください。結果の **SQL** 文は、これと同じように表示されるはずです。

```
SELECT [OrderLines].itemID , [OrderLines].name , [OrderLines].qty FROM  
dbo. [OrderLines] WHERE [OrderLines].orderId=<?C.UserString?>
```

5. 所定の **XSD** に基づいた **XML** として、送り先を定義してください。
6. フラットファイル・ソースの単純要素を、送り先の **Order** 複合単純要素にマッピングしてください。
7. フラットファイルから **Order compound** プロパティの **Advanced** タブで、**orderId** で **C.UserString** を更新してください。
8. データベース単純要素を **OrderLines** 複合単純要素にマッピングしてください。

これは、このプロセスの出力例です。

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<Orders>
  <Order>
    <ID>1</ID>
    <Name>Name1</Name>
    <Address>address1</Address>
    <OrderLines>
      <itemID>11</itemID>
      <itemQTY>item1</itemQTY>
      <itemName>2</itemName>
    </OrderLines>
    <OrderLines>
      <itemID>33</itemID>
      <itemQTY>item3</itemQTY>
      <itemName>4</itemName>
    </OrderLines>
    <OrderLines>
      <itemID>44</itemID>
      <itemQTY>item4</itemQTY>
      <itemName>1</itemName>
    </OrderLines>
  </Order>
  <Order>
    <ID>2</ID>
    <Name>Name2</Name>
    <Address>address2</Address>
    <OrderLines>
      <itemID>22</itemID>
      <itemQTY>item2</itemQTY>
      <itemName>3</itemName>
    </OrderLines>
    <OrderLines>
      <itemID>55</itemID>
      <itemQTY>item5</itemQTY>
      <itemName>5</itemName>
    </OrderLines>
  </Order>
</Orders>

```

説明

フラットファイル・レコード上で繰り返すたびに、SELECT ステートメントは orderID の値を含む Where 句でデータベースに送られます。そしてそれは、C.UserString で更新します。データベースはフィルタリングされていて、送り先複合物として選択されたレコードのマッピングを維持し続けます。

変換テーブルを使用するには

1 つの語彙またはシステムその他からデータを変換するときに、自動的にルックアップテーブルを利用するには、変換テーブルと Convert 関数を使用してください。

変換テーブルは、自動データ変換を容易にするために動的なルックアップテーブルとして使用することができます。変換テーブルには 2 つのタイプがあります。

- [スタティック](#)
- [ダイナミック](#)

スタティック変換テーブル

スタティック変換テーブルは、予め設定されたフラットファイル・データに基づいています。変換データは、サーバの起動によって読み込まれます。サーバがまだ実行中に、変換テーブルファイルのデータが修正されると、リフレッシュ変換のみが必要となります。

スタティック変換テーブルを使用するには：

1. ソリューションエクスプローラで、[データコンバータ] をダブルクリックし、スタティック変換テーブルを定義してください。
2. データマップステップを作成してください。
3. 複数のテキストフィールドを持つフラットファイルとして、送り元と送り先を定義してください。
4. 送り元から送り先にフィールドをマッピングし、必要な箇所で **Convert** 関数を使用してください。**Convert** 関数は、変換されたデータでユーザ文字列を返します。4 つの入力パラメータがあります。
 - a. 変換テーブル名
 - b. 入力テキスト（変換されたフィールド）
 - c. 入力フィールドが変換テーブル内に存在しないときに使用されるデフォルト値
 - d. 0/1 大文字と小文字の区別の指定

ダイナミック変換テーブル

下記の例では、ダイナミック変換テーブルは、XML 要素を変換するために使用されています。変換データはデータベーステーブルに保存され、実行時に変換テーブルに読み込まれます。ダイナミック変換テーブルが定義され、リフレッシュ変換テーブルサービスを使用してメモリに読み込まれます。

ダイナミック変換テーブルを使用するには：

1. [フラットファイル] ダイアログボックスで、フラットファイルの [プロパティ] ペインを通してアクセスされ、ダイナミック変換テーブルを、以下に表示されるように定義してください。

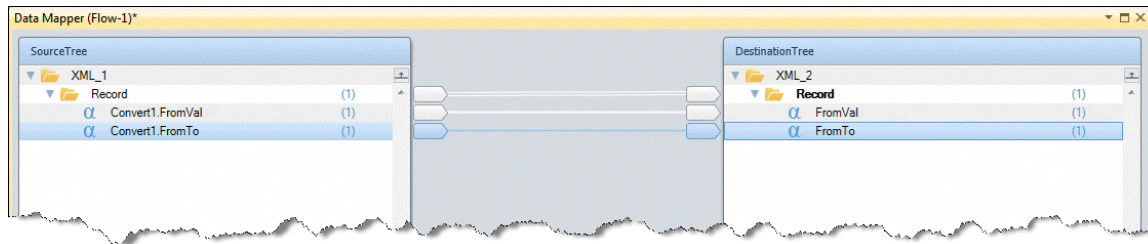
Flat File

Example File

Type **Delimited** Columns Delimiter Records Delimiter

#	Name	Data Type	Format	From	Length
1	FromVal	Alpha	100		30
2	ToVal	Alpha	100		30

2. 変換データを準備するために、**データマップ**ステップを作成してください。
3. **データベース**タイプとして送り元を定義してください。2つのテキストフィールドに**フラットファイル**タイプとして、送り先を定義してください。
4. 送り元から送り先に入力フィールドをマッピングしてください。送り先の2番目のフィールドは、変換値です。変換値を編集するために、式または追加された検索データを使用することができます。送り先ファイル名は動的な式です。これは実行時に評価されます。
5. データをリフレッシュするために、リフレッシュ変換テーブルサービスを使用してください。変換テーブルの名前は、データコンバータリポジトリで定義された名前です。ファイル名は、前のデータマップのステップで定義された送り先のファイルです。この時点で、変換テーブルデータは、メモリに読み込まれます。
6. 出力の変換を可能にするために、**データマップ**ステップを作成してください。
7. 送り元と送り先を **XML** ファイルとして定義してください。
8. 以下に表示されているように、必要な値をマッピングしてしてください。[スタティック変換テーブル](#)のセクションに記述されている **Convert** 関数を使用してください。

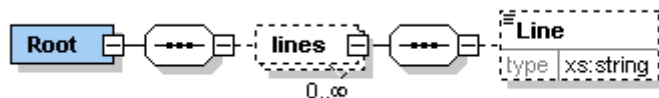


データ要素を統一するには


一つのデータ要素に multi-occurrence データ要素を統合するには、複合レベルの計算メカニズムを使用してください。統合された送り先データを作成するために送り元複合物を繰り返す間にこれは変数を更新します。

変数(F.Var1)は、送り元のデータを計算するために使用してください。

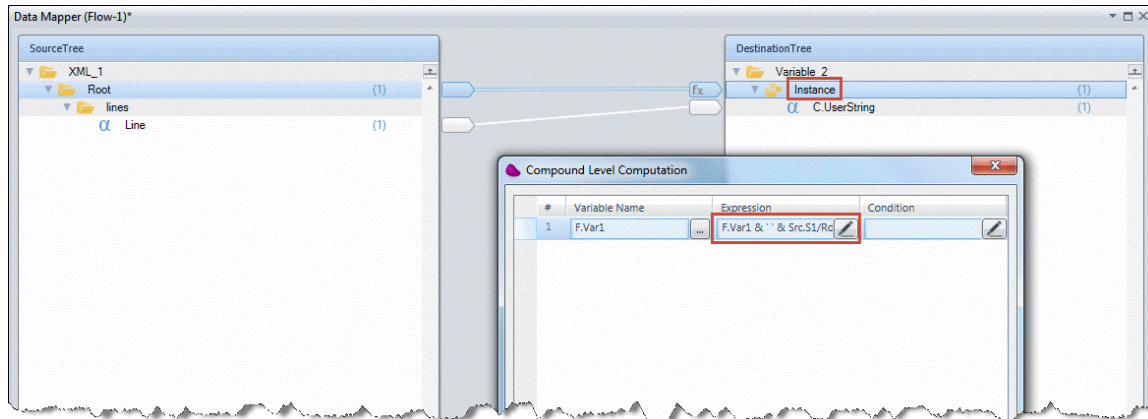
スキーマの例:



データ要素を統合するには :

1. データマッピングユーティリティをフローエリアにドラッグしてください。
2. ツールボックスの [マップスキーマ] セクションから、XML タイプを [データマップ] ウィンドウの送り元エリアにドラッグしてください。
3. XML 送り元の [プロパティ] ペインで、サンプル XML 文書を選択してください。
4. ツールボックスの [マップスキーマ] セクションから、変数タイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
5. 変数送り先の [プロパティ] ペインで、繰り返子を許可するために、ダミー変数として **C.UserString** を選択してください。
6. 以下に示すように、反復可能な単純要素の送り元複合物をインスタンス送り先にマッピングしてください。送り先の [プロパティ] ペインの [複数更新] パラメータで、をクリックして [複数更新] ダイアログボックスを開き、文字変数(F.Var1)を単一要素ソースデータの値で更新します。

F.Var1 & ' ' & Src.S1/Root/lines/Line



データ要素を分割するには

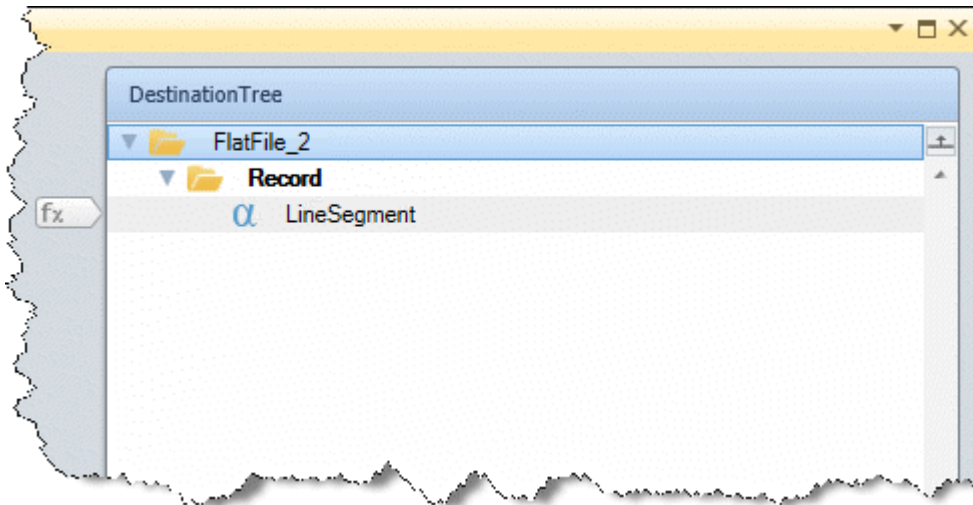
multi-occurrence の送り先要素 ([データ要素の統合処理](#)の反対)内の単一データ要素を分割するには、multi-occurrence データを含むフラットファイル・データを作成するフロー内のソースデータを前処理してください。次に、必要な multi-occurrence の送り先要素を作成するために、フラットファイルを送り先複合物にマッピングしてください。

解決策として、繰り返し回数をコントロールする変数を使用するループが必要となります。**追加**モードでフラットファイルの送り先を定義することで、各々の繰り返しでフラットファイルに要素を追加する必要があります。この代わりに、ローカルの OD 配列を使用することができます。

データ要素を分割するには：

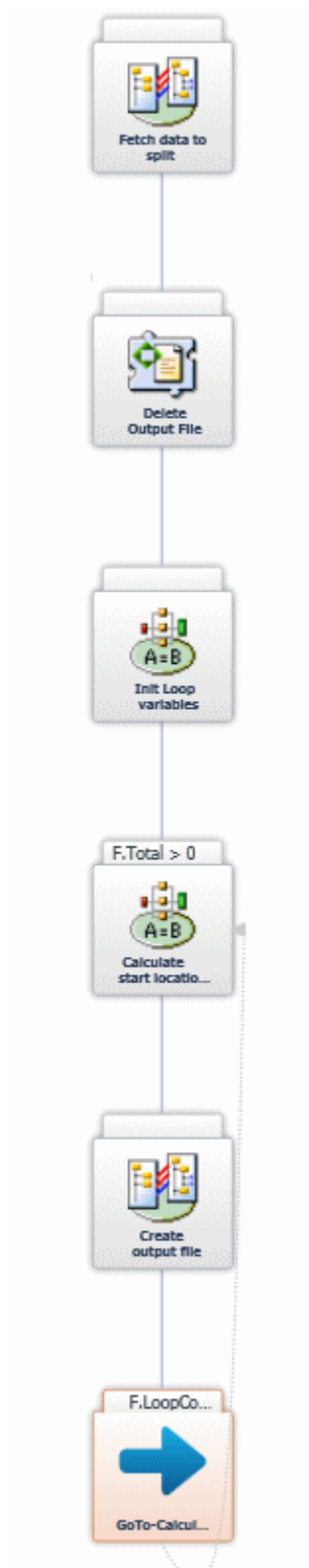
1. データマッパーティリティをフローエリアにドラッグしてください。
2. 送り元を **XML** タイプと定義し、送り先を変数と定義してください。計算は、同じデータに対する XML の度重なる読み込みを防止するために、この変数上で実行されます。
3. 文字変数への分割を希望する英数字値を保持する送り元の要素にマッピングしてください。
4. カウンタを更新するフローデータ・ステップを定義してください。
5. 分割処を実行するために、新しい**データマップ**ステップを定義してください。送り元は空で、送り先は**追加**モードの**フラットファイル**タイプです。
6. 以下に示すように、必要なデータを取得する送り先式を定義してください。例:

```
mid ( F.StringToProcess ,F.StartLocation ,F.LineSize )
```

7. 最後のステップは、**Calculate**（フローデータ）ステップに対する **GoTo** コマンドです。終了条件 (**F.Total**)が設定されます。以下のイメージでは、ループの終了条件は以下の通りです。

$F.\text{LoopCounter} < F.\text{Total}$



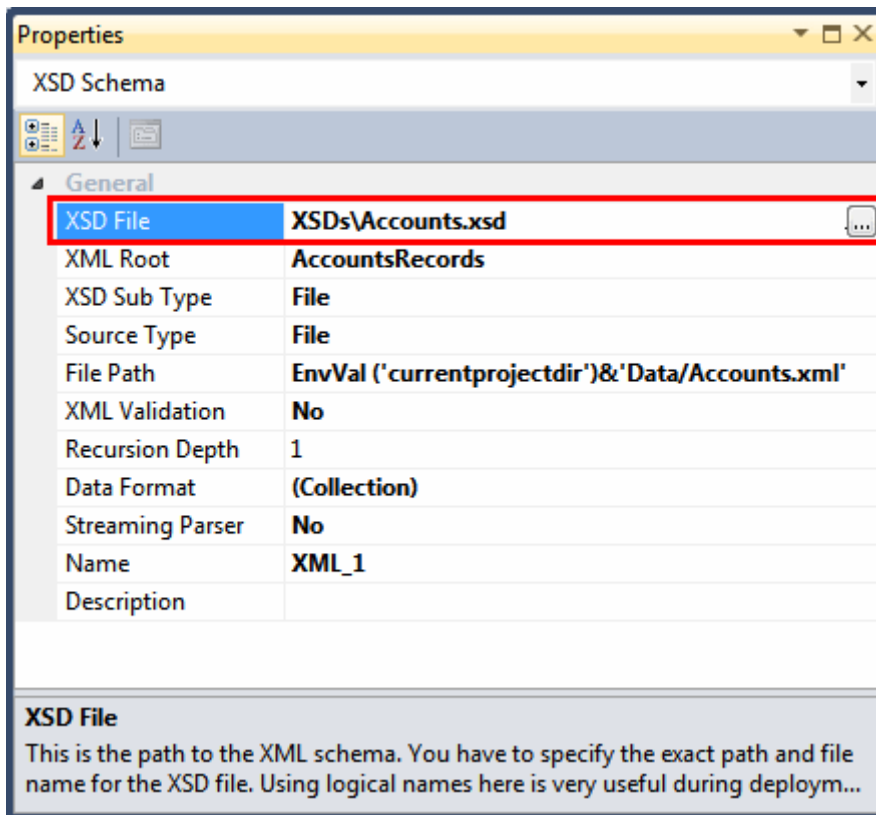
ヘッダと行レコードの両方を含むフラットな構造を階層的な XML にマッピングするには

Magic xpi でデータマップを使用する場合、フラットな構造を階層的な XML ファイルにマッピングする必要がある場合があるかもしれません。

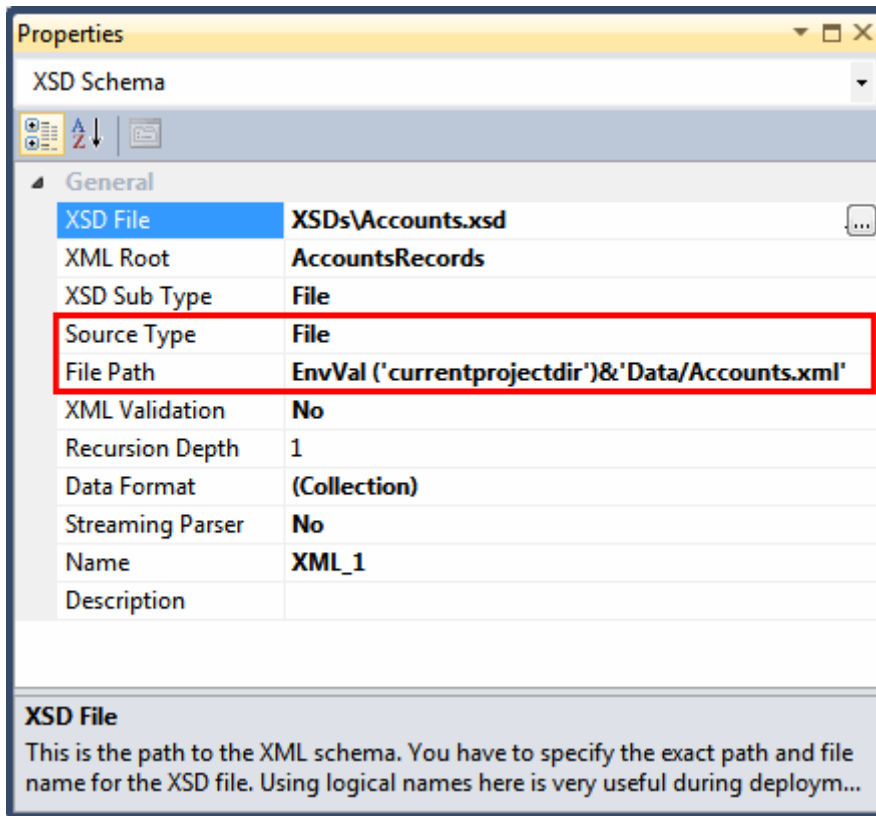
フラットな構造を階層的な XML ファイルにマッピングするには：

データマップを スタジオのフローエリアにドラッグした後に、最初にやるべき事は、次の通りにデータマップの送り元プロパティを設定することです。

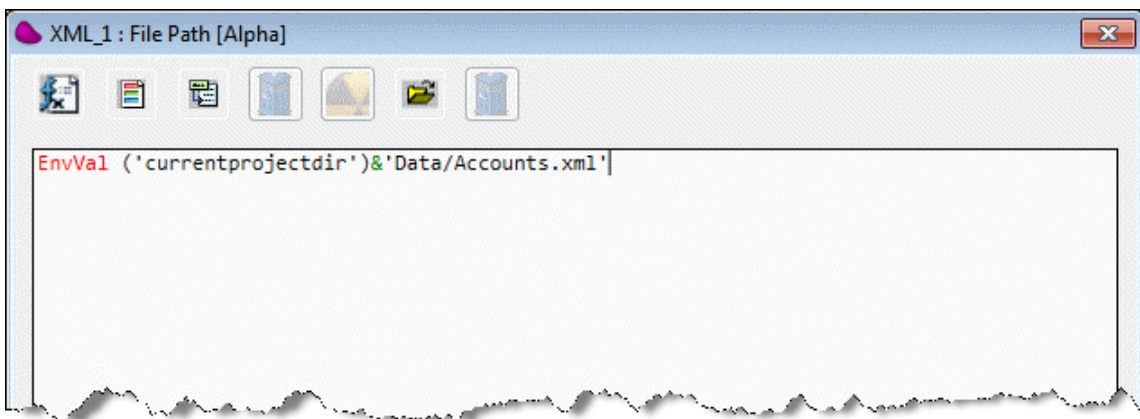
1. ツールボックスの [マップスキーマ] セクションから、XML タイプを [データマップ] ウィンドウの送り元エリアにドラッグしてください。
2. XML 送り元の [プロパティ] ペインで、必要な XSD ファイル(この場合、**Accounts.xsd**)を選択してください。



3. データをどこから取り出すかを定義する必要があります。この場合に、**Accounts.xml** と呼ばれるファイルになります。[送り元タイプ] パラメータで、ドロップダウンリストからファイルを選択してください。



4. [ファイルパス] パラメータで、式エディタを開くために...をクリックしてください。以下の式に入力してください: **EnvVal ('currentprojectdir')&'Data/Accounts.xml'**

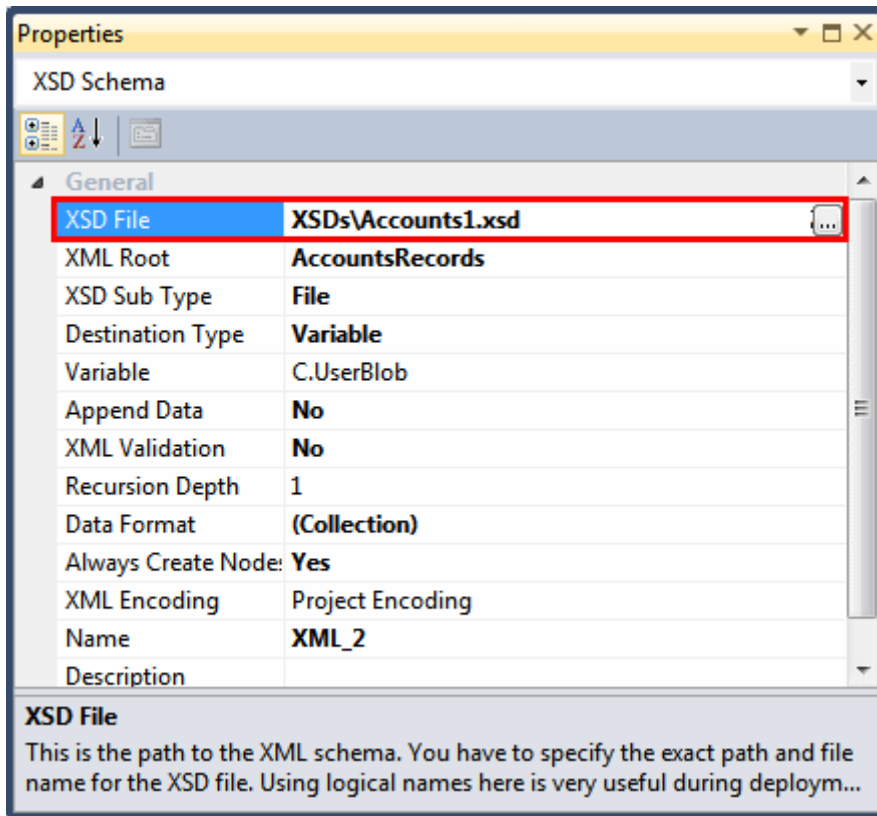


この式は送り元に、現在のプロジェクトディレクトリから **Accounts.xml** と呼ばれるデータファイルを探すように指示します。

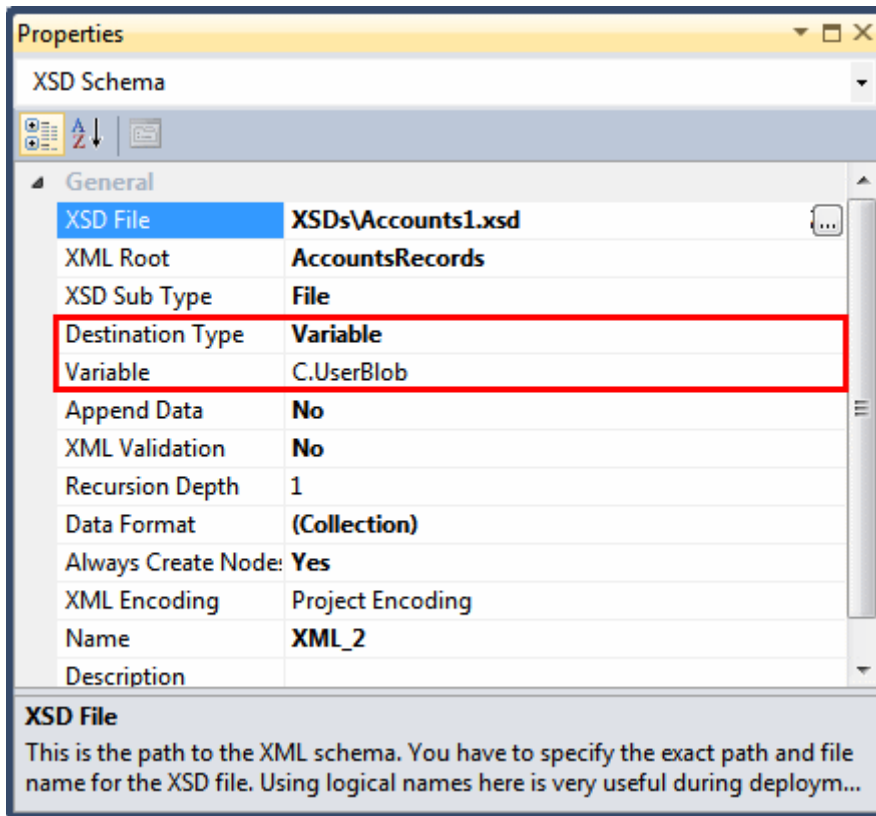
5. 式エディタを閉じるために、[OK] をクリックしてください。

データマップの送り元プロパティを設定し終わったら、次の通り送り先プロパティを設定する必要があります。

1. ツールボックスの [マップスキーマ] セクションから、XML タイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
2. XML 送り先の [プロパティ] ペインで、必要な XSD ファイル(この場合、**Accounts1.xsd**)を選択してください。



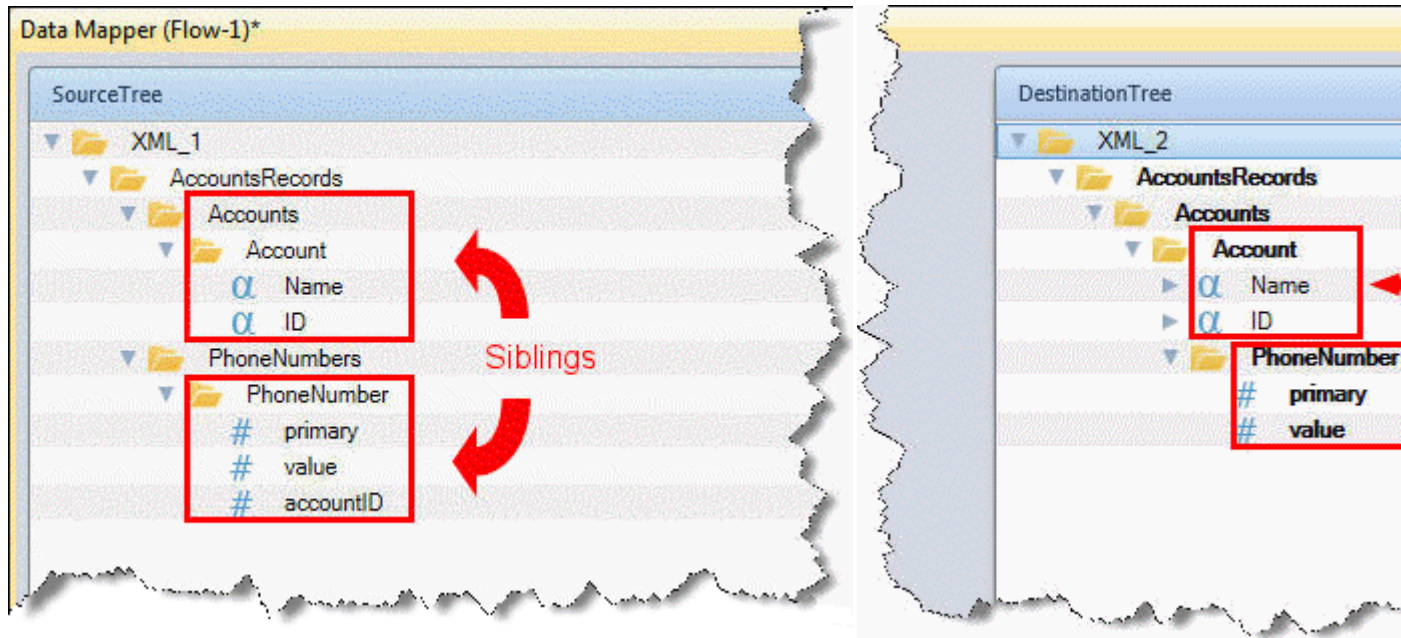
3. [送り先タイプ] パラメータで、ドロップダウンリストから変数を選択してください。
...をクリックし **C.UserBlob** を選択してください。これは、マッピングされたデータの送信先になります。



さて、データのより明確な書式を取得するために、[データマップ]ウィンドウで表示されるように、**Accounts.xsd** と **Accounts1.xsd** ファイルの構造を見てみましょう。これを行うには、送り元上と[データマップ]ウィンドウ内の送り先でパークし、それらで右クリックし、コンテキストメニューから[全て展開]を選択してください。

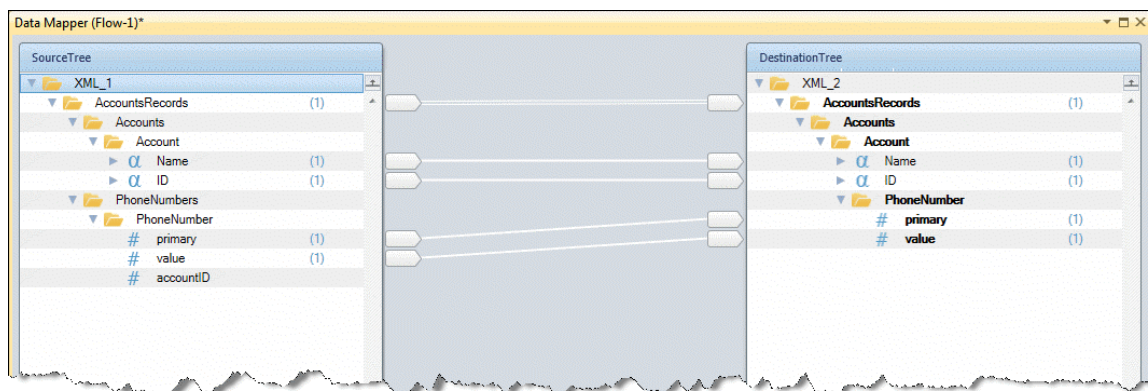
[送り元]ペイン(以下の左に表示されます)で、フラットな構造が **Accounts.xsd** ファイルの内容によって定義されていることを確認できます。これは **Account** と呼ばれる複合ノードから構成されています(それは **Name** と **ID** のノードから構成されています)。さらに、**PhoneNumber** と呼ばれる兄弟複合ノードがあります(それは **primary**、**value**、および **accountID** ノードから構成されています)。

2 つの複合ノードの間に親子関係が全くないことに注意してください。それらの間の唯一の接続は、**ID** ノードと **accountID** ノード経由です。[送り先]ペイン(以下の右に表示されます)で、階層的な親子構造が **Accounts1.xsd** ファイルによって定義されるのが確認できます。これは **Account** と呼ばれる親複合ノードから構成されています(それは **Name** と **ID** のノードから構成されています)。さらに、**PhoneNumber** と呼ばれる子複合ノードがあります(それは **primary** と **Valued** ノードから構成されています)。この場合に、アカウントと電話番号の間の関係がスキーマの構造によって制御されるため、**accountID** ノードは不要になります。



次に、次の通りソースデータを接続先データにマッピングする必要があります。

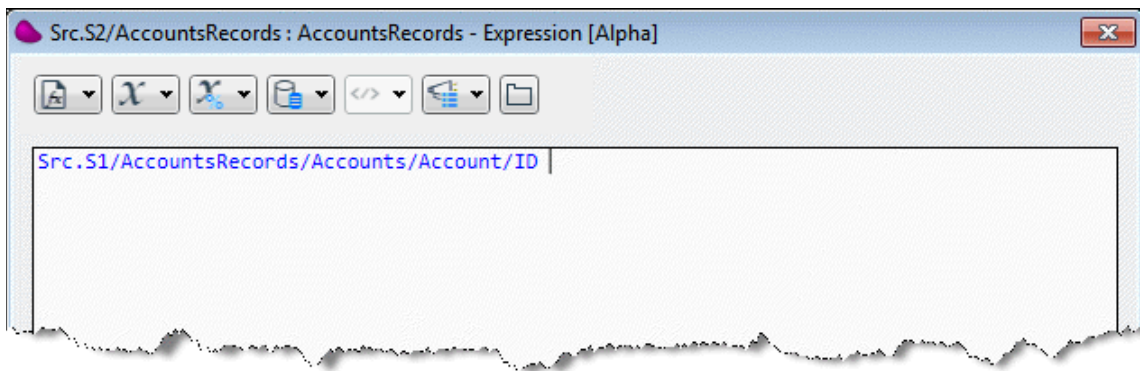
1. [送り元] ペインで、**Name** ノードを右クリックし、[接続] を選択してください。そして、マウスを [送り先] ペインの **Name** ノードにドラッグし、クリックしてください。
2. [送り元] ペインで、**ID** ノードを右クリックし、[接続] を選択してください。そして、マウスを [送り先] ペインの **ID** ノードにドラッグし、クリックしてください。
3. [送り元] ペインで、**Primary** ノードを右クリックし、[接続] を選択してください。そして、マウスを [送り先] ペインの **Primary** ノードにドラッグし、クリックしてください。
4. [送り元] ペインで、**value** ノードを右クリックし、[接続] を選択してください。そして、マウスを [送り先] ペインの **value** ノードにドラッグし、クリックしてください。



このマッピングについての問題は、各アカウントが自身の電話番号に加えてその時点でのすべてのアカウントの電話番号を保持するということです。これは、マッピングにフィルタが適用されなかったために発生します。これを防止するには、必要なフィルタを作成する必要があります。

1. フロー変数リポジトリを開き、**F.accountID** と呼ばれるフロー変数を作成してください。

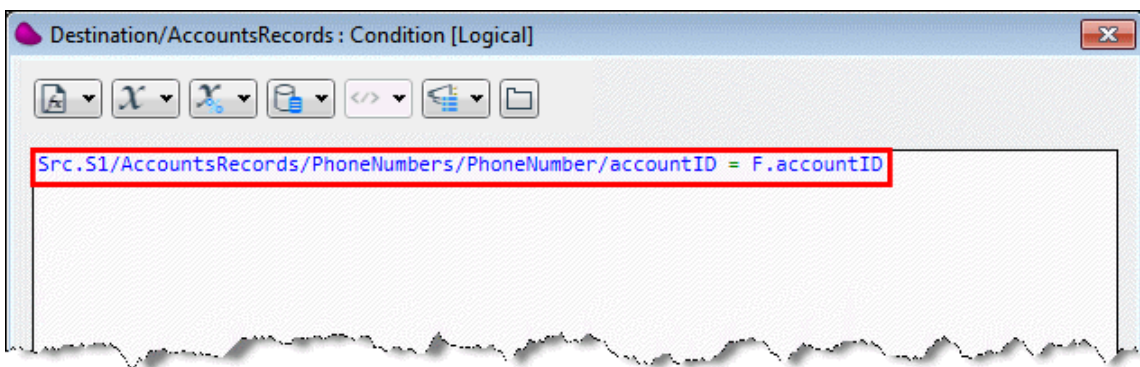
2. 各アカウントが自身の電話番号だけを保持し、他は保持しないことを確認するために、**複合レベルの計算**メカニズムを使用する必要があります。これを行うには、[送り先] ペインの **AccountsRecords** 複合ノードを右クリックし、[プロパティを表示] を選択してください。
3. [プロパティ] ペインの [複合更新] パラメータで、[複合更新] ダイアログボックスを開くために [...] をクリックしてください。そして、新規行を作成するために [追加] をクリックしてください。
4. [変数名] カラムで、[...] をクリックして **F.accountID** 変数を選択してください。
5. [式] カラムで、[編集] をクリックして式エディタを開いてください。[</>] をクリックし、リストから **ID** ノードのために正しいパスを選択してください。






この式の結果は、ソース側で、アカウントの各繰返しによって[ID]フィールドの値が取得されることです。

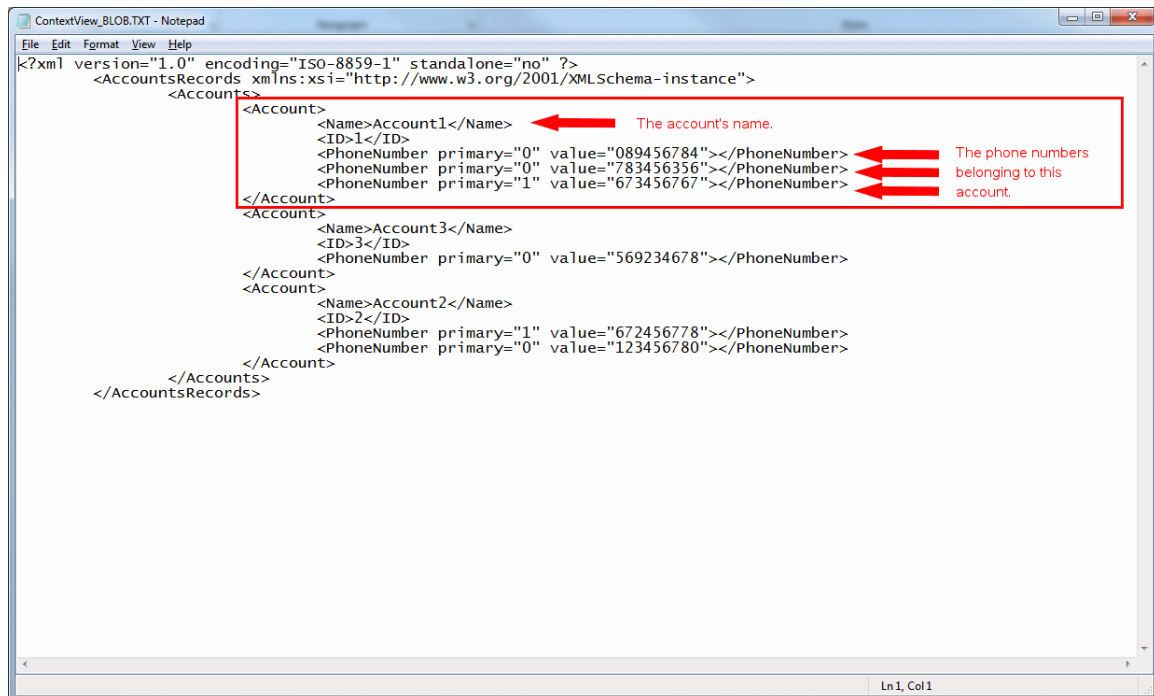
次に、接続先の**アカウント**と同じ **ID** を持つレコードだけがその特定のアカウントの配下の子 **PhoneNumber** レコードに送られるように、ソースの **PhoneNumber** レコードにフィルタをかける必要があります。

1. [送り先] ペインの **PhoneNumber** 複合ノードで右クリックし、[プロパティを表示] を選択してください。
2. [プロパティ] ペインの [条件] パラメータで、式エディタを開くために [...] をクリックしてください。[</>] をクリックしてリストから **accountID** ノードのために正しいパスを選択してください。



F.accountID は、アカウントの ID を保持するために、早い段階 c で作成されたフロー変数であることに注意してください。

4. [保存] をクリックしてください。
5. ブレイクポイントを設定してください、次に、ソリューションエクスプローラのフローで右クリックし、[デバッグ] を選択することで、フローをデバッグしてください。
6. ツールバー上の [プロジェクト実行/継続]  ボタンをクリックして、プロジェクトを実行してください。
7. [コンテキストビュー] ウィンドウを開くために、ツールバー上の [コンテキストビュー]  ボタンをクリックしてください。
8. ここで、[コンテキスト変数] タブを選択し、**C.UserBlob** 変数を探して [名前] カラムを見てください。結果データを保持するために、以前に [接続先プロパティ] ダイアログボックスの [データ接続先] パラメータでこの変数が選択されたことを覚えていてください。**BLOB** の内容にズームするために  をクリックしてください。
9. [ビュー変数 : **C.UserBlob**] ウィンドウで、ドロップダウンリストから **TXT** を選択し、[開く] をクリックして作成した実際の **XML** ファイルを参照してください。
10. **ContextView_BLOB.TXT XML** ファイルで、各アカウントがアカウントのエントリの配下で自身の電話番号がリストされていることが確認できます。



この XML は、以下の方法で作成されました。

- 各ソースの **Accounts** 要素のために、**Accounts** 要素は [送り先] ペインで作成されました。
- その後、**PhoneNumber** 要素は [送り先] ペインで作成されました。
- 次に、電話番号がアカウントに割り当てられました。
- しかし、作成したフィルタは、アカウント ID を接続先の要素と一致させる電話番号のマップを作成しただけです。
- これは、各アカウントが自身の電話番号だけを受け取ったことを意味していて、すべての番号がソースファイルに含まれているわけではありません。

送り元上で発生した繰返しをカウントするには

Magic xpi のデータマップにを使用する時、いくつかのソースレコードがあるかを知る必要がある場合があります。

また、レコードのすべての x 番号のために一定のフィールドで異なる値を定義したい場合があったり、レコードのすべての x 番号がヘッダ行で、行の残りが特定のヘッダと関連しているフラットファイルソースを持つかもしれません。

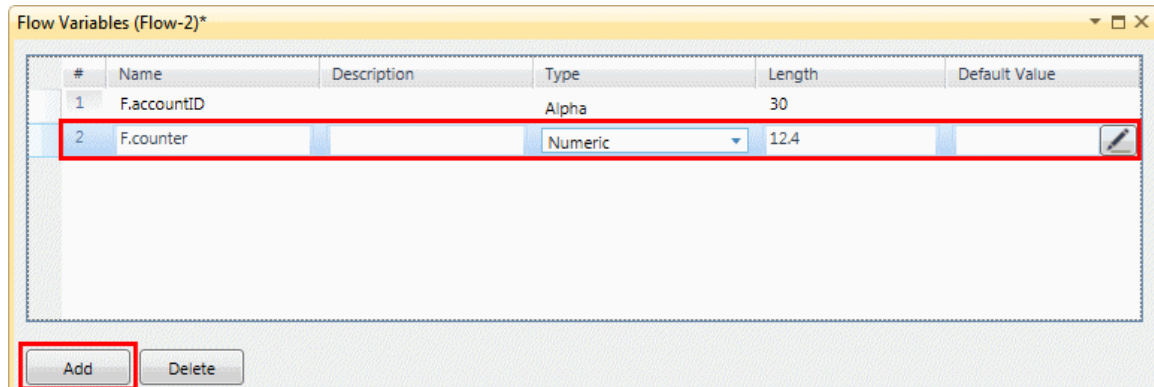
Magic xpi は、これらのソースレコードを簡単にカウントすることができます。

注 このトピックは、すでにビルドし、データマップを含むプロジェクトで動作していることを仮定しています。

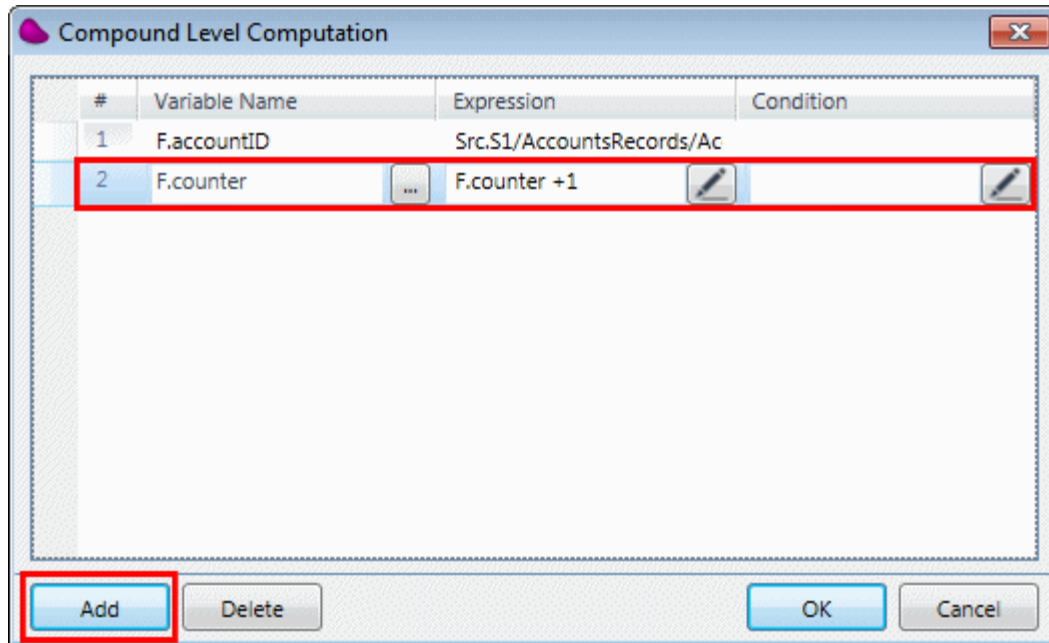
送り元上で発生した繰返し数をカウントするには：

送り元の繰返しをカウントしたい場所でプロジェクトを開いたら、以下の手順に従ってください。

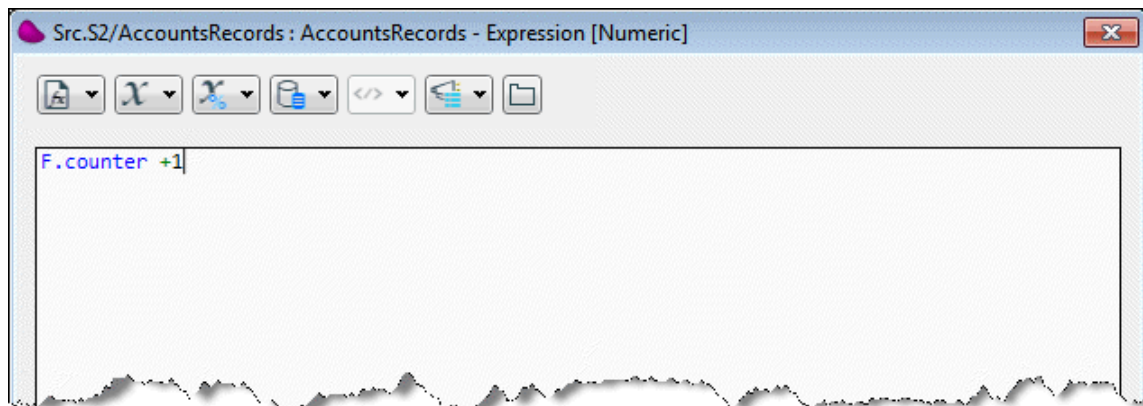
1. 関連するフローで、フロー変数リポジトリを開くために、[フロー変数]] をダブルクリックしてください。
2. [追加] をクリックして、**F.counter** と呼ばれる新しいフロー変数を作成してください。変数のタイプを**数値**に設定してください。



4. 送り元の繰返しをカウントしたい、データマップでダブルクリックして（または、右クリックして [構成] を選択して）、[データマップ] ウィンドウを開いてください
5. [データマップ] ウィンドウで、[送り先] ペインの親複合ノードで右クリックし、[プロパティを表示] を選択します。
6. [プロパティ] ペインの [複数更新] パラメータで、[複数更新] ダイアログボックスを開くために [...] をクリックしてください。そして、新規行を作成するために [追加] をクリックしてください。
7. [変数名] カラムで、[...] をクリックして **F.counter** 変数を選択してください。





9. [式] カラムで、 をクリックし次の式を入力してください: **F.counter +1**



これは、各繰返しのために、**1** の値が **F.counter** 変数に加算されることを意味しています。それは **0** から開始します。

注 意 : [送り先]ノード上で定義したかもしれない条件が実行される前に、この繰返しカウントは行われます。[送り先]ノード上で False の条件を定義した場合、代わりにマッピングを行った数を受け取ります。この問題を回避するためと、True の繰返しカウントを取得するために、ノードの[プロパティ]ペインの[条件]カラムにも同じ条件を定義する必要があります。

10. [保存] をクリックしてください。
11. ブレイクポイントを設定してください。次に、ソリューションエクスプローラのフローを右クリックし、[デバッグ] を選択することで、フローをデバッグしてください。
12. ツールバー上の [プロジェクトの実行/継続]  ボタンをクリックしてプロジェクトを実行してください。
13. [コンテキストビュー] ウィンドウを開くために、ツールバ上の [コンテキストビュー]  ボタンをクリックしてください。

14. [コンテキストビュー] ウィンドウの [フロー変数] タブで、**F.counter** 変数行の [値] カラムに値が表示されます。この値は、送り元上で発生した繰返しの数です。

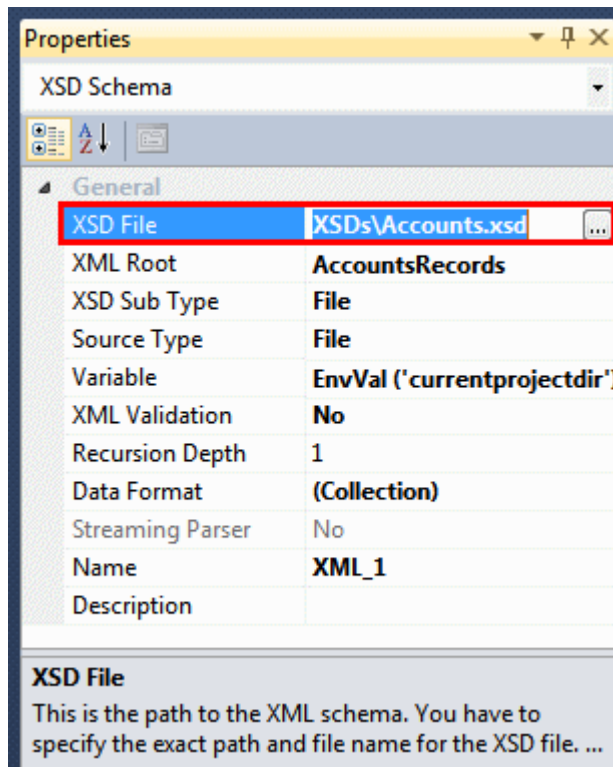
Multi-Occurrence 複合物から単一データ要素を取り出すには

時々、multi-occurrence 複合要素から、特定のレコードを取得する必要があるかもしれません。以下の例は、ソースファイルからどのように特定のアカウントの名前を取得するかについて説明しています。

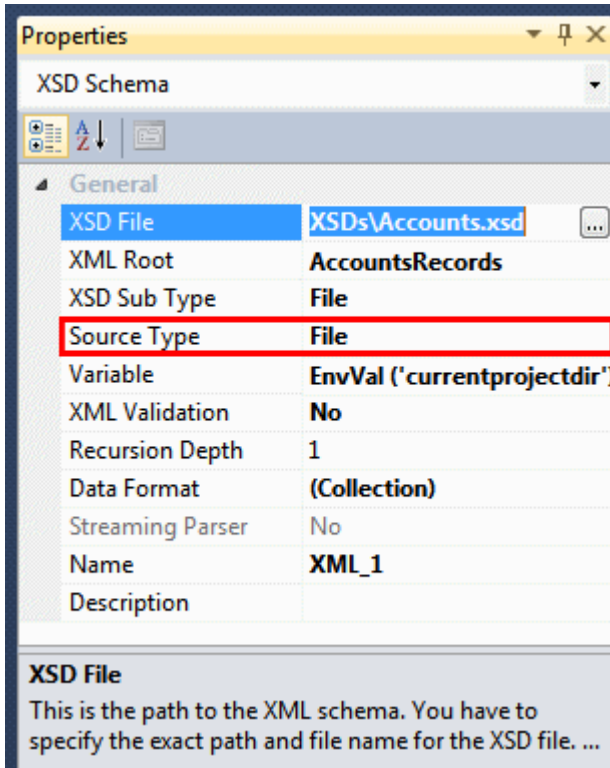
multi-occurrence 複合要素から単一のデータ要素を取り出すには：

データマップをスタジオのフローエリアにドラッグした後に、最初に行う事は、以下のようにデータマップのソースプロパティを設定することです。

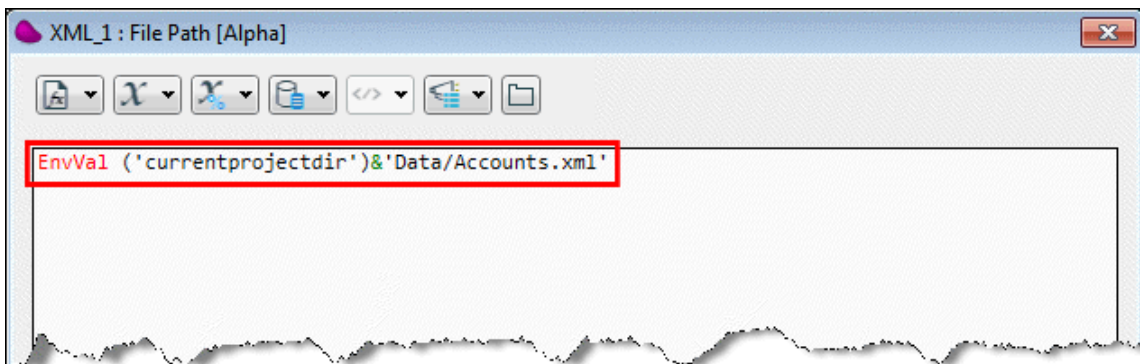
1. データマップをダブルクリックするか、右クリックして [構成] クリックしてください。 [データマップ] ウィンドウが開きます。
2. ツールボックスの [マップスキーマ] セクションから、XML タイプを [データマップ] ウィンドウの送り元エリアにドラッグしてください。
3. XML 送り元の [プロパティ] ペインで、必要な XSD ファイル(この場合、**Accounts.xsd**)を選択してください。



4. データをどこから取り出すかを定義する必要があります。この場合に、**Accounts.xml** と呼ばれるファイルになります。 [送り元タイプ] パラメータで、ドロップダウンリストからファイルを選択してください。



5. 式エディタを開くために  をクリックしてください。以下の式に入力してください:
EnvVal ('currentprojectdir')&'Data/Accounts.xml'

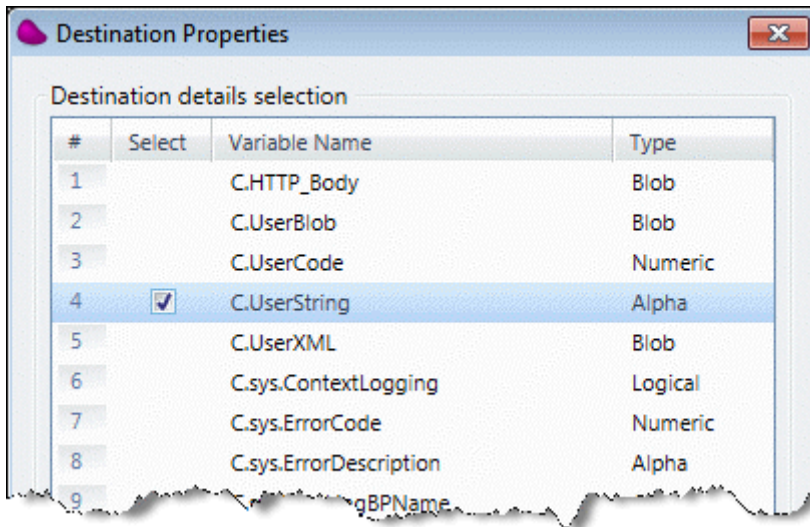


この式は送り元に、現在のプロジェクトディレクトリから **Accounts.xml** と呼ばれるデータファイルを探すように指示します。

6. 式エディタを閉じるために、**[OK]** をクリックしてください。.

データマップの送り元プロパティを設定し終わったら、次の通り送り先プロパティを設定する必要があります。

1. ツールボックスの [マップスキーマ] セクションから、変数タイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
2. 変数送り先の [プロパティ] ペインで、**C.UserString** コンテキスト変数を選択してください。

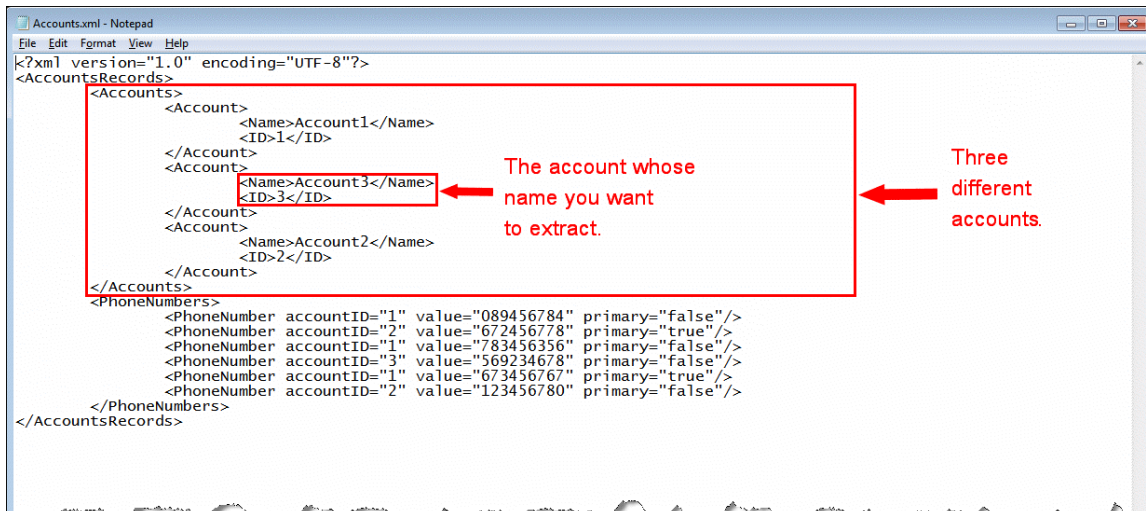


3. [送り先プロパティ] ウィンドウを閉じるために、[OK] をクリックしてください。

次に、次の通りソースデータを送り先データにマッピングする必要があります。

4. [データマップ] ウィンドウの [送り元] ペインで、**Name** ノードで右クリックし、[接続] を選択してください。そして、マウスを [送り先] ペインの **C.UserString** ノードにドラッグし、クリックしてください。[データマップ] ウィンドウで接続されるように、これらの簡単なノードは、それぞれの親の複合ノードとともに表示されます。

XML ソースファイルは、3 つの異なるアカウント(以下に表示されているような)を含んでいます。XML における Phone number は、この例では関連していないため無視することができます。



次のステップは、ID 値が 3 であるアカウントの名前を抽出するために送り先側の条件を作成することです。

1. [送り先] ペインのインスタンス複合ノードで右クリックし、[プロパティを表示] を選択してください。
2. ノードの [プロパティ] ペインの [条件] パラメータで、 ボタンをクリックして式エディタを開きます。
3. 式エディタで、 ボタンをクリックし、リストから **ID** ノードのために正しいパスを選択してください。

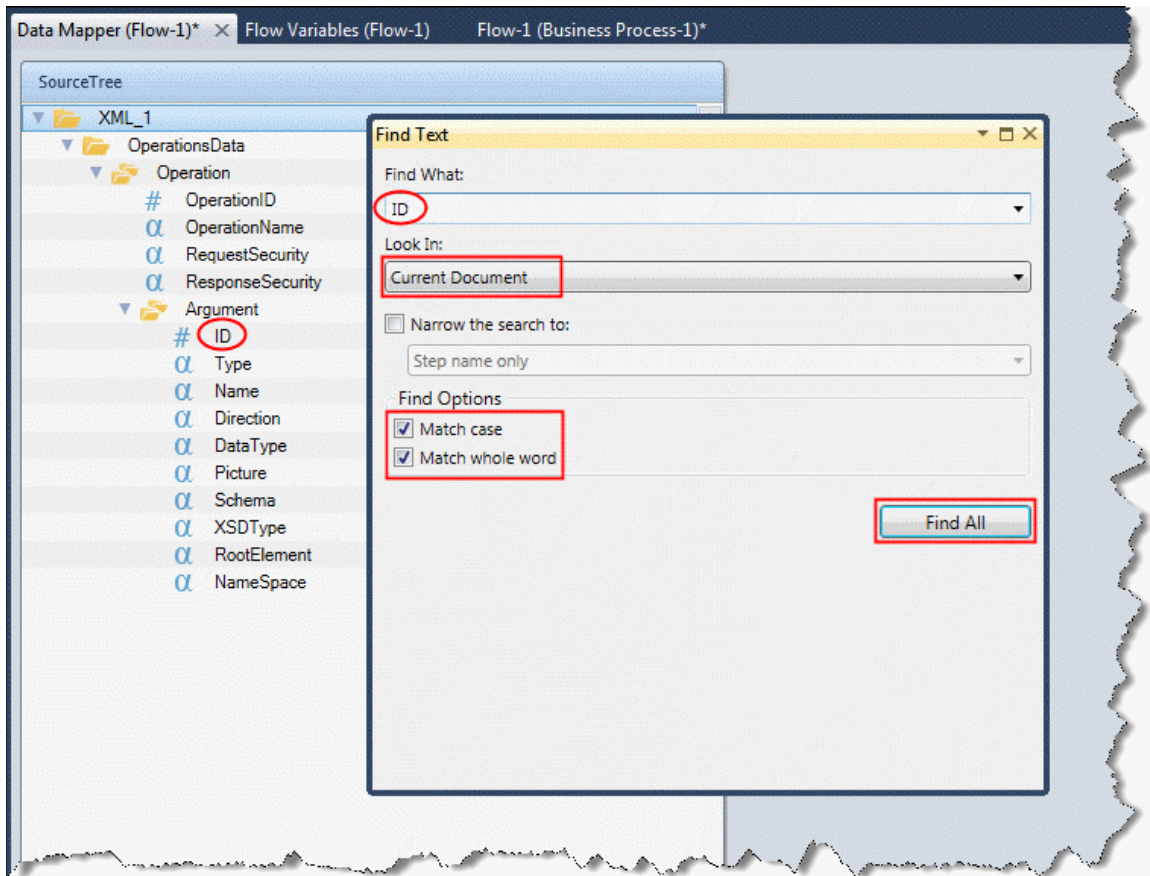
4. 式エディタを閉じるために、[OK] をクリックしてください。
5. フローエリアで、ブレイクポイントを設定してください。次に、ソリューションエクスプローラのフローで右クリックし、[デバッグ] を選択することによって、フローをデバッグしてください。
6. ツールバー上の [実行] ボタンをクリックして、プロジェクトを実行してください。
7. [コンテキストビュー] ウィンドウを開くために、ツールバー上の [コンテキストビュー] ボタンをクリックしてください。
8. ここで、[コンテキスト変数] タブを選択し、**C.UserString** 変数を探して [名前] カラムを見てください。結果データを保持するために、以前に [送り先プロパティ] ダイアログボックスでこの変数が選択されたことを覚えていてください。この変数は、その [値] カラムのソースファイルから取り出したかったアカウントの名前を表示します。この例では、アカウントの名前は **Account3** です。

データマップの特定の要素を検索するには

たくさんの送り元または送り先の要素を含むデータマップを使用する場合、マッピングしたい正確な要素を見つけることが難しい場合があります。これをより容易にするには、Magic xpi は、特定の要素を[データマップ]画面に配置する2つの異なる方法を提供します。

データマップの特定の要素を検索するには：

- 任意の要素上でクリックして、検索する要素の最初の文字を入力します。これによって、直接、入力した文字で開始される最初の要素に位置付けされます。
- [編集] メニューから、[検索] を選択し、[テキスト検索] を選択すると [テキスト検索] ダイアログボックスが開きます。ここで、検索したい要素の名前を入力し、[Look In] フィールドで**現在のドキュメント**を設定します。必要ならば、[大文字小文字を区別] や [完全に一致する単語] のチェックボックスを選択することもできます。そして、[全て検索] をクリックしてください。



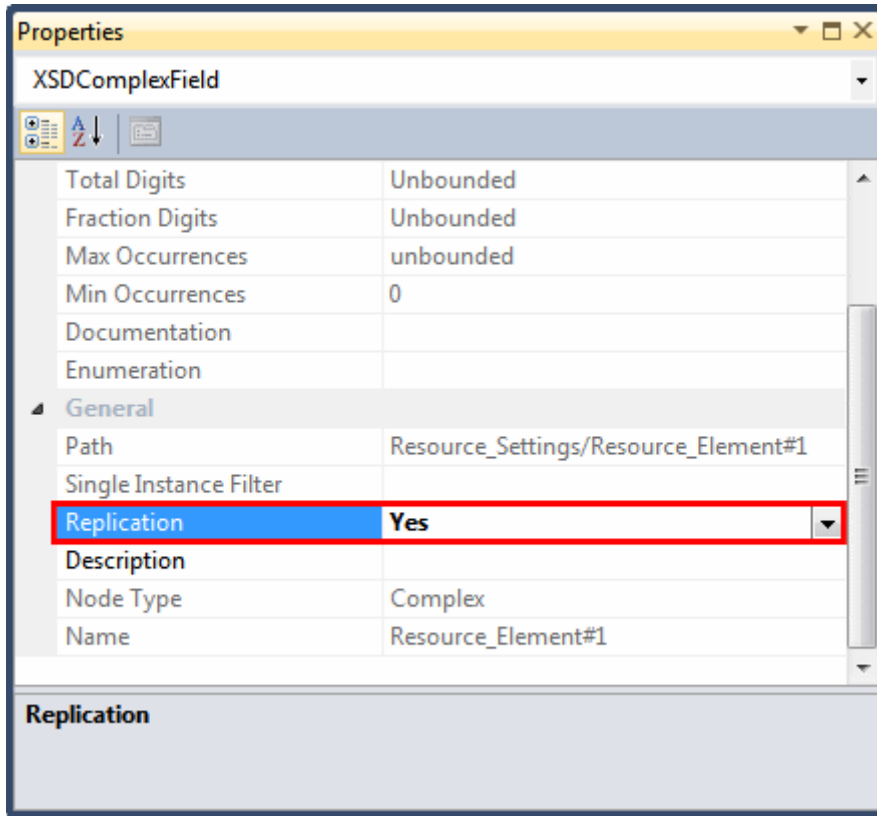
注 この機能は、[送り元]または[送り先]ペインのどちらかの検索を行うことはできますが、同時に両方のペインを検査することはできません。

反復可能な複合物の中に、特定の数の項目を作成するには

XML 送り元と送り先の中に、既存のデータマップ複合ノードを簡単に複製することができます。Magic xpi の専用のレプリケーション機能を使用してこれを行います。

反復可能な複合物内で項目を複製するには：

1. データマップを開き、複製したい複合ノードを選択してください。ノードで右クリックし、[プロパティ] ペインを表示するために、コンテキストメニューから [プロパティを表示] を選択してください。
2. [レプリケーション] パラメータで、ドロップダウンリストから **Yes** を選択してください。



3. 選択された複合ノードに **#1** という接尾辞が付加されることに気づくはずです。
4. [データマップ] 画面で、複製したい複合ノードで再び右クリックしてください。コンテキストメニューから [レプリケーションの追加] を選択してください。これによって、その下にエントリのある重複したい複合ノードを追加します。

複製された複合ノードには、**#2** という接尾辞が付加されます。この新しいノードは、複製した直後のノードのすぐ下に追加されます。オリジナルの複合ノードで右クリックするか、複製された任意のノード上で右クリックすることによって、制限無くレプリケーション処理を繰り返すことができます。新しいインスタンスには、連番で接尾辞が付加されます。

複製された複合ノードを削除するには：

- オリジナルのノードをまたは複製された複合ノードで右クリックし、コンテキストメニューから [レプリケーションの削除] を選択してください。これで、複製した最後の複合ノードが削除されます。

オリジナルのノード名を復元するには：

- オリジナルの複合ノード上でパークし、[レプリケーション] パラメータのドロップダウンリストから **No** を選択してください。

データベース

データベース

以下のトピックは、Magic xpi のデータベースについて扱っています。

[動的な Where 句を使用するには](#)

[ODBC を使用するには](#)

[Magic xpi の DateTime データベース項目を使用するには](#)

[Magic xpi の内部データベースを変更するには](#)

[How Do I Manually Create a Magic xpi Internal Database?](#)

[SQL Server Express を使用するとき、パフォーマンスの問題を解決するには](#)

[SQL トランザクション用にデフォルトの BLOB サイズを変更するには](#)

[データベース・トランザクション上でロールバックを開始するには](#)

[マッピングのために MSSQL の DateTime タイプの送り元を Oracle の DateTime タイプに変換するには](#)

[Magic xpi に Oracle のシーケンスを読み込むには](#)

[How Can I Call a Stored Procedure in Magic xpi?](#)

動的な Where 句を使用するには

Magic xpi データマッピングユーティリティを使用して SQL 文を作成することができます。静的と動的の二種類の Where 句があります。このテクニカルノートは、この命令文をいつどのように使用するかを説明します。

	静的 Where 句	動的 Where 句
適用先	送り元と送り先	送り先 (Update と Delete)
いつ使用する か	値が変数に入っている場合	値が変数に入っていない場合
「データマッ パ」画面での 表示	No	Yes
構文	Table_column = <?Variable_Name?>	table column = <!?Data_holder_name?!>
データタイプ	変数は Magic xpi でサポートされている どのデータ形式でも可	データフォルダは Magic xpi でサポートされてい るどのデータ形式でも可

例

以下の例では、動的な WHERE 句の作成方法を説明します。例の中のステップは二つのステージに分かれています。

- ステージ 1 送り元と送り先定義
- ステージ 2 データマッピング

このステップでは、プロジェクトで以下のプロパティを持った **exampleTable_Des** という名前のテーブルがあることを前提としています。

Table_Key	Col1	Col2	Col3
Key1	A	56	D
Key2	B	78	E
Key3	C	99	F

Magic xpi 逆引き辞典

また統合プロジェクトには、二つ目のテーブル、**exampleTable_Des** と同じ構造の **exampleTable_Sour** があることを前提としています。

この例の目的は、**exampleTable_Sour** テーブル内の **exampleTable_Des** テーブルからレコードを削除することです。

ステージ1：送り元と送り先を定義

送り元を定義する

1. Magic xpi スタジオでは、データマッピングユーティリティをフローにドラッグします。
2. データマッピングでダブルクリックするか、右クリックして [設定] をクリックしてください。 [データマッピング] ウィンドウを開きます。
3. ツールボックスの [マッピングスキーマ] セクションから、データベースタイプを [データマッピング] ウィンドウの送り元エリアにドラッグしてください。
4. [データベース定義] ドロップダウンリストで、データベースを選択してください。
5. データベースウィザードを開くために、 [ウィザード] をクリックしてください。
6. [テーブル選択] 画面で、**exampleTable_Sour** テーブルを選択し、 [次へ] をクリックして [カラム選択] 画面を開きます。
7. **Table_Key** カラムを選択し、 [次へ] をクリックして [Where 句] 画面を開きます。
8. [次へ] をクリックして [ウィザードの結果] 画面に移り、 [完了] をクリックしてウィザードを閉じます。

送り先を定義する

1. ツールボックスの [マッピングスキーマ] セクションから、データベースタイプを [データマッピング] ウィンドウの送り先エリアにドラッグしてください。
2. [データベース定義] ドロップダウンリストの中で、データベースを選択してください。
3. データベースウィザードを開くために、 [ウィザード] をクリックしてください。
4. [DB オペレーション] リストで、**Delete** を選択してください。
5. [テーブル選択] 画面で、**exampleTable_Des** テーブルを選択し、 [次へ] をクリックして [Where 句] 画面を開きます。
6. [使用可能カラム] リストから **Table_Key** カラムをダブルクリックし、 [Where 句テキスト] エリアに以下の内容を追加します。 : =<!?MyKey?!>.
テキストは以下のように表示されます : **Table_Key=<!?MyKey?!>**

注意: **MyKey** は定義済み変数ではなくデータホルダです。.

7. [次へ] をクリックし、 [ウィザード結果] ダイアログボックスへ進みます。 [完了] をクリックしてウィザードを閉じます。

ステージ2：データをマッピングする

- [データマッピング] 画面で、**Table_Key** カラムを **MyKey** にマップします。

これにより **exampleTable_Des** テーブル内のレコードのうち、**exampleTable_Sour** にもあるレコード全てを削除する設定が完了しました。

注 意 : この例で `example.Table_Des` テーブルのカラムをデータホルダにマッピングせずに変数を使用した場合、`exampleTable_Des` テーブルからは一つのレコードしか削除されないはずで
す。

ODBC を使用するには

Magic xpi は、ODBC データソースを使用し様々なデータベース管理システムのデータにアクセスすることができます。ODBC データソースでデータベースのデータにアクセスするには、互換 ODBC ドライバを設定しなくてはなりません。Windows®の ODBC データソースアドミニストレータで、ユーザもしくはシステムデータソースのドライバを設定します。プロジェクトのひとつ、もしくは両方のタイプのデータソースを設定するおとができます。

構成

ODBC の構成には三つの主なステージがあります。

- ステージ 1 ODBC データソースの追加
- ステージ 2 ドライバの互換性の確認
- ステージ 3 Magic xpi で ODBC の設定

ステージ 1 : ODBC データソースを追加する

ODBC データソースを追加するには、以下を行います。

1. Windows の [スタート] メニューから [コントロールパネル] をクリックし、[管理ツール] アイコンをダブルクリックします。
2. [データソース (ODBC)] アイコンをダブルクリックし、[ODBC データソースアドミニストレータ] ダイアログを開きます。
3. いずれかをクリックします：
 - [ユーザー DSN] タブをクリックすると、データが現在のユーザのみ参照可能で、現在の PC でのみで
使用できるようになります。
 - [システム DSN] タブをクリックすると、現在の PC の全てのユーザでデータを参照できるようになり
ます。
4. [追加] をクリックします。
5. データソースを設定したい適切なドライバを選択し、[完了] をクリックします。
6. [ODBC セットアップ] ウィザードで必要な情報を入力します。

ステージ 2 : ドライバ互換性を確認する

Mgchkdrv.exe ユーティリティを使用し、Magic xpi との互換性テストを行います。

1. <Magic xpi インストールフォルダ>\Runtime\Magic xpi\Gateways フォルダで、
Mgchkdrv.exe をダブルクリックします。
2. [Select Data Source] ボタンをクリックします。
3. [コンピュータデータソース] タブをクリックし、先ほど作成した DSN を選択し、
[OK] をクリックします。

ユーティリティがテストを行い、ログと併せて結果を表示します。

もしテストが失敗した場合、ODBC ドライバは Magic xpi と互換性がありません。

Magic xpi 逆引き辞典

ステージ 3 : Magic xpi で ODBC を設定する

以下のステップでは、Magic xpi で ODBC ゲートウェイを設定する方法を説明します。

1. Magic xpi スタジオを起動し、[プロジェクト] メニューで [設定] をクリックして [設定] ダイアログボックスを開きます。
2. [設定] ダイアログボックスの [リソース] セクションで、新しいデータベースリソースを作成してください。
3. [DBMS] フィールドから ODBC オプションを選択し、[OK] をクリックします。必要であれば、データソース名とユーザ ID/パスワードに DSN を入力します。
4. [検証] をクリックし、接続をチェックします。[適用] をクリックして終了します。

これで ODBC データベースを、データマッピングユーティリティの送り元と送り先として使用することができます。

トラブルシューティング

もし **Unknown database** エラーメッセージが表示された場合、以下のようにして **Magic.ini** ファイルの ODBC ゲートウェイ設定を有効にします。

1. Magic xpi インテグレートプラットフォームの一部となる全てのアプリケーションを閉じます。ここには、Magic xpi スタジオ、Magic xpi モニタ、Magic xpi サーバそして Magic xpa が含まれています。
2. Magic xpi のインストールフォルダにある **Magic.ini** ファイルを開きます。
3. [Magic_Gateways] セクションを探します。
4. ;MGDB19=mgodbc.dll 行へ進み行頭のセミコロン(;)を取り除きます。
5. 等号記号(=)の後に以下のパスを追加します。 \Magic xpi\Gateways\<.dll ファイルの名前>

注 意：	選択する.dll ファイルは、Magic xpix.x/Gateways フォルダ内にあるものでなければなりません。
---------	--

6. **Magic.ini** ファイルを保存し終了します。
7. Magic xpi を再起動します。

Magic xpi の DateTime データベース項目を使用するには

Magic xpi は、DateTime データ型をネイティブにはサポートをしません。Magic xpi のデータマップを使用してデータベースの DateTime 値で SELECT、INSERT、UPDATE を行うには、文字列としてこれらの値を処理する必要があります。

Magic xpi は、ノードレベルで、Magic xpi サーバに文字列として DateTime 値を処理するように指示する簡単な設定を提供します。この動作をコントロールするプロパティは、**StringDate** と呼ばれ、日付データ型で利用できます。

例

DATETIME フィールドを使用しているデータベース

日付と時刻を DATETIME データ型フィールドに挿入するには、[文字列日付]プロパティを **Yes** に設定してください。

注意： この変更後、全ての日付データベースフィールドは文字列として解釈されます。

例えば、MSSQL の DateTime フォーマットが 11:22:41 に 10/01/2007 の場合、計算値フィールドで以下の式を使用することができます。

DStr (Date (), 'DD/MM/YYYY') & ' ' & TStr (Time (), 'HH:MM:SS')

Oracle データベース

1. 設定

日付と時間を DATETIME データ型フィールドに挿入する場合は、[StringDate] プロパティを **Yes** に設定してください。

注意： この変更後、全ての日付データベースフィールドは文字列として解釈されます。

2. Magic xpi プロジェクトで:

上記で説明したように適切な変更が終了したら、Oracle の NLS_DATE_FORMAT を設定する必要があります。Oracle に DateTime 値を入力するには、二箇所の送り先 (同じデータマップかあるいは二つの個別のデータマップ) を修正する必要があります。

送り先 1:

- a. データマップユーティリティをフローエリアにドラッグしてください。
- b. データマップでダブルクリックするか、右クリックして [設定] をクリックしてください。 [データマップ] ウィンドウを開きます。
- c. ツールボックスの [マップスキーマ] セクションから、データベースタイプを [データマップ] ウィンドウの送り元エリアにドラッグしてください。
- d. [プロパティ] ペインの [ダイナミック SQL ステートメント] フィールドで、**Yes** を選択してください。
- e. [SQL ステートメント] ボタンをクリックし、以下の Oracle SQL コマンドを入力してください。

```
alter session set NLS_DATE_FORMAT='YYYY-MM-DD HH:MI:SS'
```

この SQL コマンドは、現在の Oracle セッションの NLS_DATE_FORMAT を次のように設定します: 'YYYY-MM-DD HH:MI:SS'。

- f. 変更内容を Magic xpi で有効にするには [保存] をクリックします。

送り先 2:

- a. データマップユーティリティをフローエリアにドラッグしてください。
- b. ツールボックスの [マップスキーマ] セクションから、データベースタイプを [データマップ] ウィンドウの送り元エリアにドラッグしてください。
- c. [プロパティ] ペインでウィザードを開いてください。
- d. [DB オペレーション] リストで、**Insert** を選択してください。
- e. Magic xpi ウィザードを使用して適切なテーブルとフィールドを選択します。

- f. 必要なデータをマッピングするか、直接入力します。
- g. [日付] フィールドには先ほどのステップで指定した書式で入力します。

例: '1980-10-16 18:18:18'

Magic xpi の内部データベースを変更するには

ユーザは、Magic xpi インストール中に導入されたデフォルト Magic xpi 内部データベースを別のサポートされているデータベースに変更することができます。このトピックは、Magic xpi 内部データベースを 1 つのサポートされたデータベースから別のサポートされたデータベースに変更するために必要な手順について紹介しています。

ステップ #1 : 新しい Magic xpi 内部データベースを作成する

新しい Magic xp の i 内部データベースは、Magic xpi のインストールによって提供されるデータベースに特有なスクリプトを使用して、バックエンドでデータベースサーバに作成されます。詳細は、[Magic xpi の内部データベースを手動で作成する](#) を参照してください。

ステップ #2 : Magic xpi の新しい内部データベースを参照するために、datasource.xml ファイルを変更する

新しい内部データベースを反映するために datasource.xml ファイルを変更する必要があります。ファイルは以下の配置されています: <Magic xpi インストールフォルダ>\¥Runtime¥Config このファイルを設定する方法を示した例は、この中のファイルを参照してください。

1. **datasource.xml** ファイルでデータベース設定を行ってください。
2. MSSQL を除いたすべてのデータベースでは、以下のフォルダに JDBC ドライバ-jar ファイルをコピーしてください: <Magic xpi インストールフォルダ>\¥Runtime¥java¥DatabaseDrivers。MSSQL の JDBC ドライバ(jar ファイル)だけが Magic xpi のインストール時に提供されます。他の DBMS に対しては、JDBC ドライバの jar ファイルを提供する必要があります。
3. **datasource.xml** で定義された **driverClassName** が JDBC ドライバと互換であることを確認してください。

ステップ #3 : Magic xpi の新しい内部データベースを参照するために、Magic.ini ファイルを変更する

1. Magic xpi スタジオ、Magic xpi サーバ、および Magic xpi モニタが動作していないことを確認してください。
2. 修正する前に、**Magic.ini** ファイルをバックアップしてください。
3. **Magic.ini** ファイルの[MAGIC_LOGICAL_NAMES]セクションに、以下の論理名を追加してください。

date_format=YYYY-MM-DD

DB_NAME= <スクリプトを使用し作成したデータベース名>

DB_SERVER_NAME= <データベースサーバ名>

DB_USER= <スクリプトを使用して作成されたユーザ名>

DB_PASSWORD= <スクリプトを使用して作成されたユーザのパスワード>

4. MSSQL ゲートウェイのエントリを有効にします。

Magic.ini ファイル内の[MAGIC_GATEWAYS]セクションのMSSQL ゲートウェイのエントリが有効になっており、ゲートウェイの dll ファイルへのフルパスが指定されていることを確認してください。

もしこのエントリがコメントアウトされていた場合（行頭にセミコロンがついている）、以下のように行頭のセミコロンを外してください。

```
[MAGIC_GATEWAYS]
```

```
MGDB20=D:¥Magic xpi¥Gateways¥mgms7.dll
```

5. **Magic.ini** ファイルを保存し、Magic xpi サービスを開始してください。

エラーメッセージが表示されない(データベースサーバとの接続について)場合、Magic xpi の内部データベースの変更が成功したことになります。

ステップ#4 : Oracle データベースのために、magicxpi-setenv.bat ファイルを更新します。

- **oracle.net.tns_admin jvm** の引数を持つ **GigaSpaces/bin/magicxpi-setenv.bat > ADDITIONAL_OPTIONS** プロパティを更新してください。これは **TNSNAMES.ORA** ファイルを含むフォルダへのパスが含まれています。

例えば : **set ADDITIONAL_OPTIONS=%ADDITIONAL_OPTIONS% -
Dcom.magicsoftware.ibolt.home=%MAGIC_XPI_HOME%\runtime -
Doracle.net.tns_admin=\$ORACLE_HOME/network/admin**

ステップ#5 : Oracle12c 以降のデータベースのため、Oracle12c のセッションをプラグブルデータベース(PDB)に切り換える

Oracle12c は、顧客で作成されたプラグブルデータベース(PDB)を含めることができるマルチテナントコンテナデータベース(CDB)です。Oracle12c のデータベースのスクリプトを実行することで、Magic xpi に、PDB を作成させたい場合があります。

スクリプトを実行する前に、次の通りに Oracle セッションをその PDB に切り換える必要があります。

1. **dba** として **sqlplus** と接続してください。例えば : **connect system/manager as sysdba**
2. コマンドを実行してください : **SHOW CON_NAME**

現在のセッションがどこで実行しているかを確認する必要があります。例えば、次のセッションはルートセッションとして動作しています。

```
CON_NAME
-----
CDB$ROOT
```

3. データベースサーバ上にどのようなタイプのデータベースを持っているかを確認するには、以下のコマンドを実行する必要があります: **SELECT PDB FROM V\$SERVICES;**

出力例:

```
PDB
-----
PDBORCL
CDB$ROOT
CDB$ROOT
CDB$ROOT
CDB$ROOT
```

これは、PDBORCL である PDB を持っていることを意味しています。

4. コマンドを実行してください: **alter session set container=PDBORCL;**

出力は次のようになるはずです: **Session altered.**

5. セッションが切り替わったことを確認するには、再び以下のコマンドを実行してください: **SHOW CON_NAME**

出力は次のようになるはずです: :

```
CON_NAME
-----
PDBORCL
```

6. これで、Oracle ユーザを作成するためにスクリプトを実行することができます。

トラブルシューティング

特定の RDBMS で内部データベース用に Magic xpi を作成した時にエラーが発生した場合、SQL スクリプトの実行時に作成されたエラーログを参照してください。これは、Magic xpa が使用している RDBMS と関連した問題を特定する場合に役立ちます。

例 - MSSQL データベース

〈Magic xpi インストール〉¥db¥MSSQL 内の MSSQL.log ファイルをチェックして、MSSQL サーバに接続して SQL ステートメントを実行する際に記録されたエラーを確認してください。

Windows OS の PATH 環境変数にパス(MSSQL¥Binn フォルダ)のエントリが含まれていることを確認してください。MSSQL サーバ上でデータベースやユーザを作成することが可能な権利を持つユーザ ID があることを確認してください。

Magic xpi の内部データベースを手動で作成するには

Magic xpi の内部データベースを手動で作成するには、〈Magic xpi インストールフォルダ〉¥DB フォルダ内にある j スクリプトを実行する必要があります。各データベース用に、それぞれのフォルダに必要なスクリプトがあります。

Rundb.bat ファイルを編集し、それを実行するか、データベースを設定するためのコマンドラインを実行する必要があります。

Rundb.bat ファイルを編集するには:

1. %1 を SQL Server 名に置き換えてください。

2. **%2** をユーザ名と置き換えるか、ユーザが無ければ、空の""ブラケットを使用してください。
3. **%3** をパスワードと置き換えるか、パスワードが無ければ、空の""ブラケットを使用してください。
4. コマンドが Windows 認証用の場合は、**%4** を「1」と置き換えます。そうでなければ空の""ブラケットを使用してください。
5. **%5** をデータベースのタイプと置き換えてください : MSSQL、ORACLE、DB2、DB2400、または MYSQL。

コマンドラインからファイルを実行するには :

1. コマンドプロンプトを開いてください。
 2. **DB** フォルダへ移動してください。
 3. コマンドラインで、**rundb** を入力し、スペースを入力後に以下のパラメータを追加してください。
- SQL Server の名前。
 - ユーザ(ユーザが無ければ、空の""ブラケットを使用してください)の名前。
 - パスワード(パスワードが無ければ、空の""ブラケットを使用してください)。
 - コマンドが Windows 認証用の場合は「1」を、そうでなければ、空の""ブラケット。
 - データベースのタイプ : MSSQL、ORACLE、DB2、DB2400、または MYSQL。

例えば : C:\Magic xpi 4.6\DB>RunDB MSSQLServer sa p@ssword "" MSSQL

4. **Enter** を押下して、パラメータを指定してバッチを実行してください。

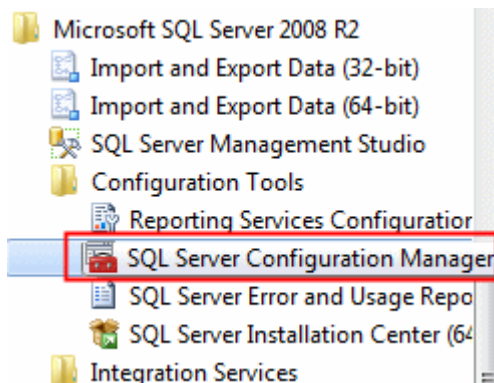
ファイルを実行すると、プロセスは選択した DB タイプに基づいたログファイルを作成します。

SQL Server Express を使用するとき、パフォーマンスの問題を解決するには

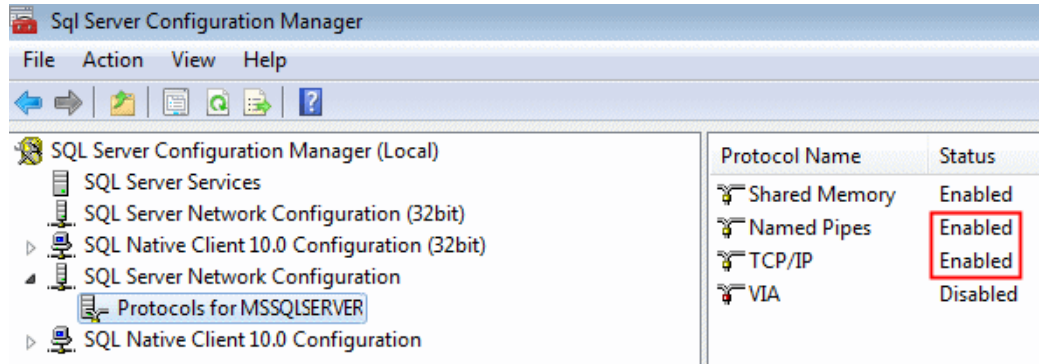
Magic xpi が内部データベースとして MS SQL Server Express を使用してインストールされるとき、時々パフォーマンスの問題が発生することがあります。たとえば、[データマップ]画面を開くときに、このような問題を経験するかもしれません。

これらの問題を防ぐには、MS SQL Server の設定マネージャで、以下の変更を行ってください。

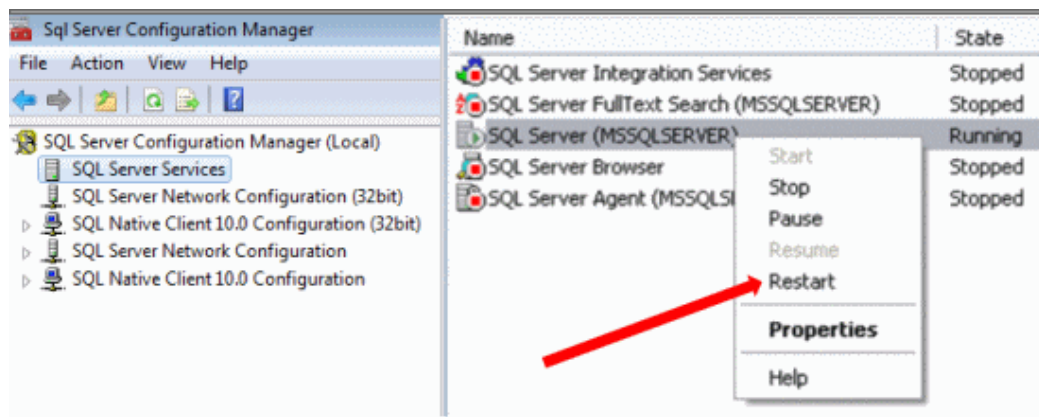
1. **SQL Server 構成マネージャ**を開いてください。 .



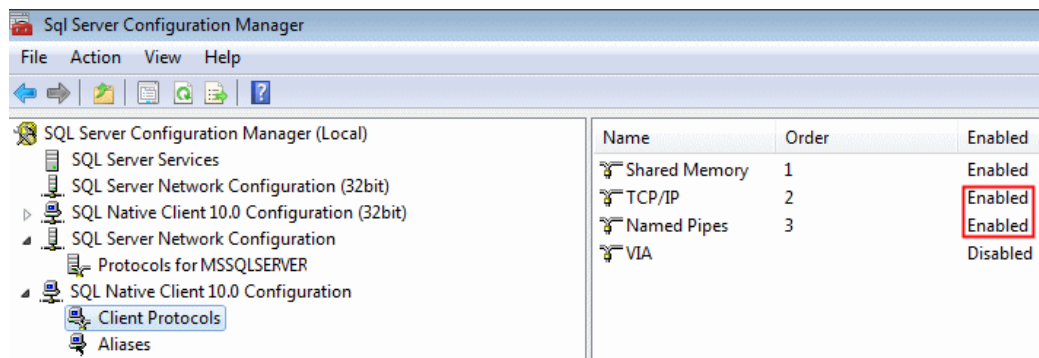
2. サーバのプロトコルとして **TCP/IP** と名前付きパイプを有効にしてください。



3. これらの変更をした後、サーバを再起動してください。 **SQL Server** のサービスに移動し、サーバを選択して、（以下の図で示すように）右クリックで再起動します。



4. 次に、クライアント側で同じプロトコルを有効にしてください。



SQL トランザクション用にデフォルトの **BLOB** サイズを変更するには

`sql_blobsize` フラグを使用して、これを行うことができます。このフラグは、**Magic.ini** ファイルの `[MAGIC_LOGICAL_NAMES]` セクションにあります。デフォルト値は、**1000000** に設定されます。これは BLOB のサイズを **1MB** と定義しています。BLOB のサイズを変更するためにこのフラグの値を変更することができます。

データベース・トランザクション上でロールバックを開始するには

ロールバックを作成するために、フローデータユーティリティを使用することができます。論理変数を以下の式で更新するフローデータ・ステップを作成してください。

Rollback ('FALSE'Log, 0)

この関数は、オープンされたすべてのデータベースのトランザクションをロールバックします。

マッピングのために MSSQL の DateTime タイプの送り元を Oracle の DateTime タイプに変換するには

MSSQL の DateTime タイプは、Magic xpi の文字型タイプ (String Date)として読み出されます。

上記のマッピングを実行するために、Oracle のフォーマットと互換性を持つように、String Date を変換するために Oracle の **to_date** 関数を使用する必要があります。

例：

- **Insert** の代わりにデータマップの動的な DB オペレーションを使用してください。
- 送り元ノードで [StringDate] プロパティが **Yes** に設定されていることを確認してください。
- ダイナミックな SQL 文の“プレースホルダ”として使用するために、2つの新しい変数（例、**F.source** と **F.format**）を追加してください。
- 例えば、以下のような SQL 文を使用してください。
`"insert into table_name values(to_date('<!?F.source?!>','<!?F.format?!>'))"`
- 送り元の MSSQL DateTime を **F.source** 変数にマッピングしてください。
- 以下のフォーマットを **F.format** 変数にマッピングしてください。
`'yyyy/mm/dd hh:mi:ssam'` (これは、Oracle の Date タイプのデフォルトフォーマットです)

結果として、以下のような SQL 文が作成されます。

```
insert into table_name values(to_date('2012/01/15 11:05:04','yyyy/mm/dd hh:mi:ssam'))
```

Magic xpi に Oracle のシーケンスを読み込むには

以下の SQL 文を数値変数にマッピングすることで、Oracle のシーケンスを取得することができます。

select pubs1.nextval from dual

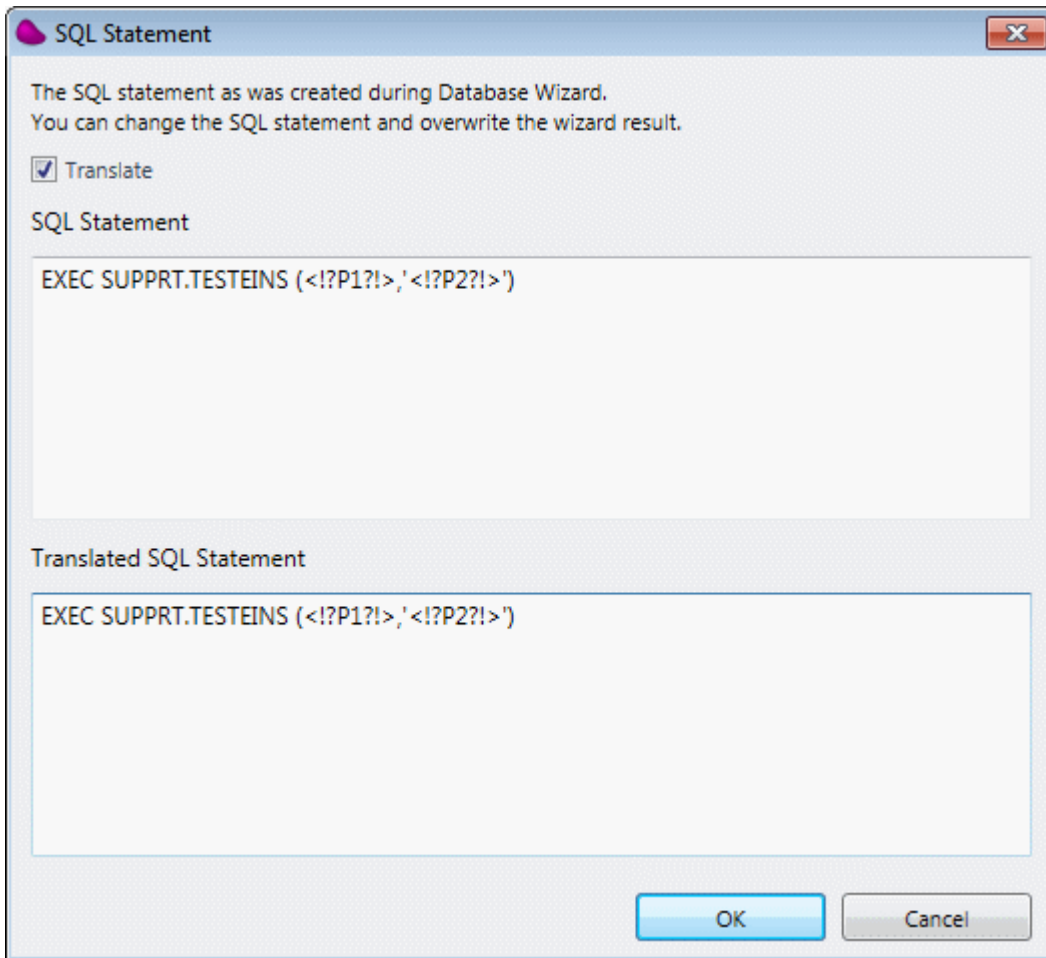
(*pubs1 は、シーケンスの名前です。)

SQL 文は、手動で記述する必要があります。この場合、ウィザードを使用することができません。

Magic xpi でストアードプロシージャを呼び出すには

Magic xpi でストアードプロシージャを呼び出すためにデータマップを使用することができます。

データの戻りが内ストアードプロシージャを呼び出すために動的な送り先の SQL 文を使用することができ、また、関連データを送るためにプレースホルダーを使用することができます。

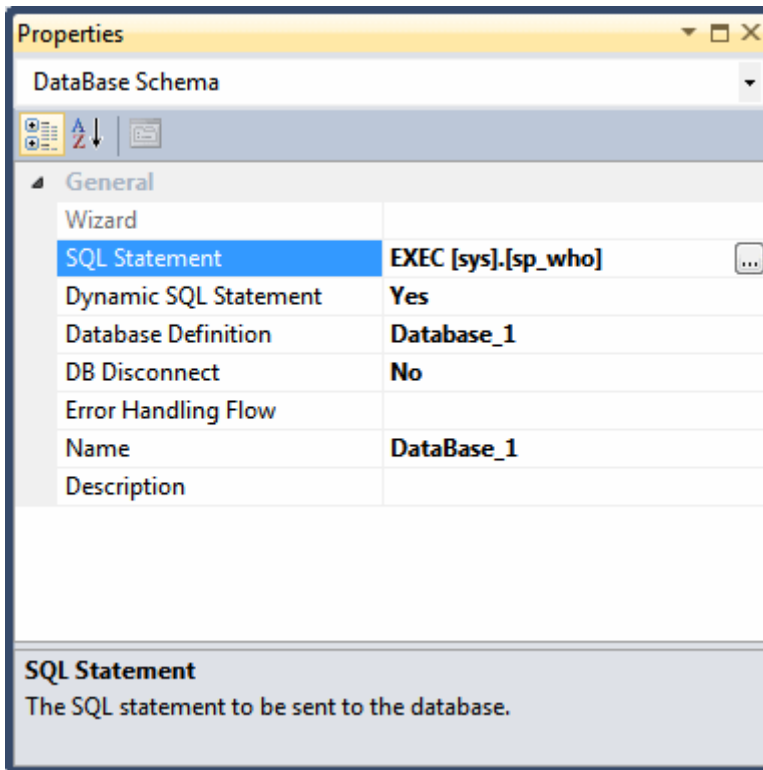


ストアードプロシージャがデータを返すと、それが結果セットを返すことを確認してください。

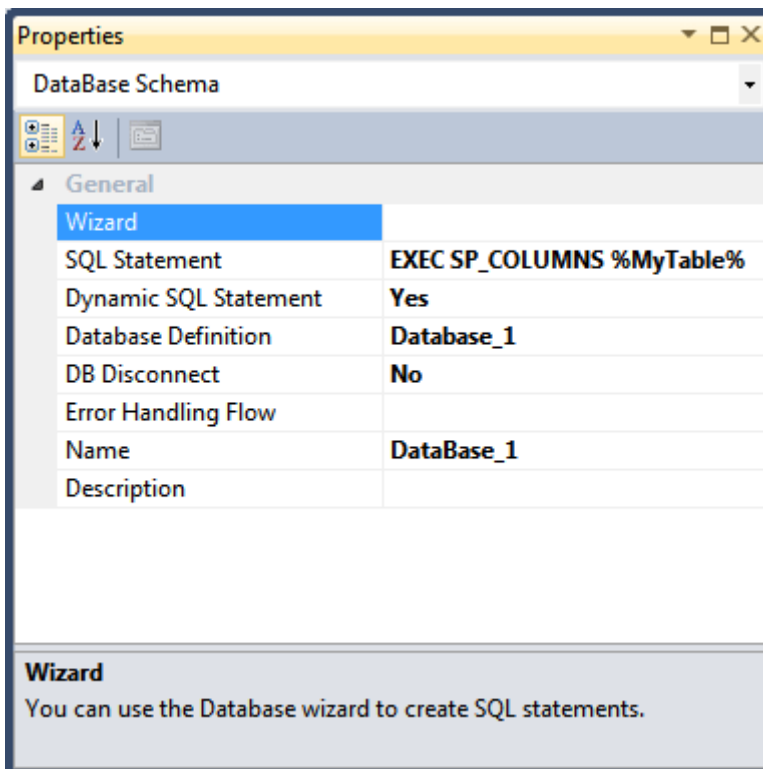
パラメータをストアードプロシージャに送る必要がある場合、ハードコードされた値または環境変数を使用してください。

注 環境変数を使用するのであれば[データマップ]画面に入るために最初にハードコードされた値を使用してください。マッピングが完了したら、これらの値を環境変数に置き換えることができます。これをしないと、[データマップ]画面を開くことができません。

以下のスクリーンショットは、パラメータなしでのストアードプロシージャの呼び出しを表示しています。



以下のスクリーンショットは、パラメータ付きでストアードプロシージャを呼び出す場合を表示しています。



日付と時刻

日付と時刻

以下のトピックは、Magic xpi の中で日付と時刻の扱いについて説明しています。

[Magic xpi で DateTime の日付タイプを使用するには](#)

[Unix 時刻を計算するには](#)

[現在の日付を取得するには](#)

[現在の時刻を取得するには](#)

[日付値を加算するには](#)

[時効値を加算するには](#)

[指定された日付に対応する月の最初の日付を取得するには](#)

[指定された日付に対応する年の最初の日付を取得するには](#)

[指定された日付に対応する月の最後の日付を取得するには](#)

[指定された日付に対応する年の最後の日付を取得するには](#)

[指定された日付に対応する曜日名を取得するには](#)

[指定された日付に対応する月名を取得するには](#)

[指定された日付の年/月/日の各部分を取得するには](#)

[指定された日付の週単位の日数を取得するには](#)

[日付値を文字列に変換するには](#)

[文字列で格納された日付を日付値に変換するには](#)

[XML DateTime フォーマットを計算するには](#)

[ユリウス歴の日付を計算するには](#)

Magic xpi で DateTime の日付タイプを使用するには

Magic xpi は、DateTime データタイプをネイティブにサポートしていません。DateTime データタイプを使用する場合は、文字列として処理する必要があります。Magic xpi データマップを使用して、データベース間で DateTime 値 に対する SELECT、INSERT、または UPDATE を行いたい場合はこれを行う必要があります。

Magic xpi は、ノードレベルで簡単に設定することができます。これによって Magic xpi サーバに、文字列として DateTime 値を処理するように指定します。この動作をコントロールするプロパティは、**String Date** と呼ばれて、日付データタイプで利用できます。

Unix 時間を計算するには

Unix 時間は、協定世界時(UTC)1970 年 1 月 1 日 00:00:00 以降からの経過した秒数です。何秒が一定のイベントが発生してから経過時間を追跡するためにこの数値を使用することができます。

プロジェクトで Unix 時間を使用したい場合、UTC が以下の式を使用するタイムゾーンであるという前提で、Unix 時間の導関数を計算することができます。

(Date ()-DVal ('01/01/1970','MM/DD/YYYY')-1)*86400+(Time ()-TVal('00:00:00','HH:MM:SS'))

それでは、この式はどのようなものかを見てみましょう。

Date() - …… これは日付関数で、システム日付を返し、以下を減算します。

DVal('01/01/1970','MM/DD/YYYY')-1 …… DVal 関数は、日付'01/01/1970'を日付値に変換し、Magic xpi に、MM/DD/YYYY という書式であることを指定します。次に、1 の値は、1970 年の最初の日を削除するために戻り値から減算します。

***86400** …… 86,400 (これは 1 日の秒数です)で戻り値の日付を乗算します。

ここまでで在、1970 年 1 月 1 日から現在の日付(システム日付)までの秒数を持っています。

次に以下を加算します。

Time() - …… システム時刻で以下を減算します。

TVal('00:00:00','HH:MM:SS') …… TVal 関数は、時間'00:00:00'を日付値に変換します。HH: MM:SS は、時刻の書式を Magic xpi に指定します。しかし、この部分を空白のままにすると、Magic xpi が書式を 'HH:MM:SS' と扱われるため、この部分は省略することもできます。

これは Magic xpi に、この日付値が日付の最初に開始されることを指定しています。

式全体がユニット全体としてそれをマークしている括弧に取り囲まれています。

現在の日付を取得するには

Magic xpi で現在の日付を取得するには、Date() 関数を使用します。**Date()** 関数は、Magic xpi が現在実行中の PC のシステム日付を返します。

日付の格納形式

Magic xpi での日付は、0001/01/01 以降からの日数を表す数値で格納されています。変換はすべて自動的に行われるため、開発者が意識する必要はありませんが、内部処理上は数値として扱われるということです。従って、5 日を追加する場合、もとの日付から 5 日を進めることになります。「月末の日付」や「うるう年」のような問題に関して考慮する必要はありません。

Date() 関数を使用するには :

Date()関数は、データマップでしばしば使用されます。これは受渡日として現在の日付を入力し、レコードがいつ作成され修正されたかというタイムスタンプを設定する場合に利用できます。

これを使用する場合は、**Date()**を入力するだけです。実行時に、関数は現在の日付に置き換えます。

アメリカ(MM/DD/YY)またはヨーロッパ(DD/MM/YY)などの日付のフォーマットを変更した場合、DStr()関数を使用することができます。この詳細については、[日付値を文字列に変換するには](#)のトピックの **DStr()** 関数を使用するを参照してください。

Date()+10 のように、この関数に数値式を加算したい場合は、AddDate()関数を使用してください。これは、Magic xpi が日付フィールドで数値式が利用できないからです。

現在の時刻を取得するには

Magic xpi で現在の時間を取得するには、Time() 関数を使用します。**Time()** 関数は、Magic xpi が実行している PC のシステム時間を返します。

Time() 関数を使用するには :

Magic xpi 逆引き辞典

Time()関数は、データマップでしばしば使用されます。これは受渡日として現在の時刻を入力し、レコードがいつ作成され修正されたかというタイムスタンプを設定する場合に利用できます。

これを使用する場合は、**Time()**を入力するだけです。実行時に、関数は現在の時刻に置き換えます。

より精密な時間を扱う場合は、**mTime** 関数を使用してください。

日付値を加算するには

Magic xpi における日付データは 0 年以降の日付を表す数値が格納されています。従って、日付を加算するには、日数を追加するだけで可能になります。例えば：

`Date() + 5`

この式では、現在から 5 日後の日付が返ります。

しかし、年数や月数および日数をもとに加算処理を行う場合は、**AddDate()**関数を使用します。この関数は、以下の4つのパラメータを使用します。

AddDate(<i>Date, Years, Months, Days</i>)	
<i>Date</i>	加算/減算対象の日付データ
<i>Years</i>	加算/減算する年数を表す数値
<i>Months</i>	加算/減算する月数を表す数値
<i>Days</i>	加算/減算する日数を表す数値

例えば、以下のような式を作成した場合：

`AddDate(F.myDate,0,-1,0)`

指定された日付から 1 ヶ月が減算されます。従って、例えば 2008 年 7 月 6 日をもとに計算を行った場合、**AddDate()** 関数は 2008 年 6 月 6 日を返します。

時効値を加算するには

Magic xpi における時刻データは、0 時以降の秒数を表す数値が格納されています。従って、時刻を加算するには、秒数を追加するだけで可能になります。例えば：

`Time() + 5`

この式では、現在から 5 秒後の時刻が返ります。

しかし、時数や分数および秒数をもとに加算処理を行う場合は、**AddTime()** 関数を使用します。この関数は、以下の4つのパラメータを使用します。

AddTime(<i>Time, Hours, Minutes, Seconds</i>)	
<i>Time</i>	加算/減算対象の時刻データ
<i>Hours</i>	加算/減算する時数を表す数値
<i>Minutes</i>	加算/減算する分数を表す数値
<i>Seconds</i>	加算/減算する秒数を表す数値

例えば、以下のような式を作成した場合：

`AddTime(F.myTime,-1,0,0)`

指定された日付から 1 時間が減算されます。従って、例えば 14:08:03 をもとに計算を行った場合、**AddTime()** 関数は 13:08:03 を返します。

指定された日付に対応する月の最初の日付を取得するには

帳票処理のための日付範囲を求めるために、月の最初の日付を取得する必要がある場合があります。
BOM() 関数を使用することで、このようなことが実現できます。

BOM() 関数を使用するには：

BOM() 関数の構文は以下の通りです。

BOM(*date*)

date …… 日付項目 (05/06/2008'DATE のように直接日付を指定するか、日付データを返す式を指定)

この関数は、月の最初の日付を返します。

上記の 05/06/2008'DATE 中の文字列'DATE'は、Magic xpi のリテラルです。このリテラルを使用すると、文字列は日付と解釈されます。

これが数値として内部で扱われるため、四則演算で使用することができます。例えば、
01/01/97'DATE+14 は、日付'15/01/97' が算出される有効な式です。

参照: [指定された日付に対応する月の最後の日付を取得するには](#)

指定された日付に対応する年の最初の日付を取得するには

帳票処理のための日付範囲を求めるために、年の最初の日付を取得する必要がある場合があります。
BOY() 関数を使用することで、このようなことが実現できます。

BOY() 関数を使用するには：

BOY() 関数の構文は以下の通りです。

BOY(*date*)

date …… 日付項目 (05/06/2008'DATE のように直接日付を指定するか、日付データを返す式を指定)

この関数は、年の最初の日付を返します。

上記の 05/06/2008'DATE 中の文字列'DATE'は、Magic xpi のリテラルです。このリテラルを使用すると、文字列は日付と解釈されます。

これが数値として内部で扱われるため、四則演算で使用することができます。例えば、
01/01/97'DATE+14 は、日付'15/01/97' が算出される有効な式です。

参照: [指定された日付に対応する年の最後の日付を取得するには](#)

指定された日付に対応する月の最後の日付を取得するには

帳票処理のための日付範囲を求めるために、月の最後の日付を取得する必要がある場合があります。
EOM()関数を使用することで、このようなことが実現できます。

EOM() 関数を使用する利点の一つは、うるう年を正しく処理することです。

EOM() 関数を使用するには：

EOM()関数の構文は以下の通りです。

Magic xpi 逆引き辞典

`EOM(date)`

date …… 日付項目 (05/06/2008'DATE のように直接日付を指定するか、日付データを返す式を指定)
この関数は、月の最後の日付を返します。

上記の 05/06/2008'DATE 中の文字列'DATE'は、Magic xpi のリテラルです。このリテラルを使用すると、文字列は日付と解釈されます。

これが数値として内部で扱われるため、四則演算で使用することができます。例えば、
01/01/97'DATE+14 は、日付'15/01/97' が算出される有効な式です。

参照: [指定された日付に対応する年の最初の日付を取得するには](#)

指定された日付に対応する年の最後の日付を取得するには

帳票処理のための日付範囲を求めるために、年の最後の日付を取得する必要がある場合があります。
EOY()関数を使用することで、このようなことが実現できます。

EOY() 関数を使用するには :

EOY()関数の構文は以下の通りです。

`EOY(date)`

date …… 日付項目 (05/06/2008'DATE のように直接日付を指定するか、日付データを返す式を指定)
この関数は、年の最後の日付を返します。

上記の 05/06/2008'DATE 中の文字列'DATE'は、Magic xpi のリテラルです。このリテラルを使用すると、文字列は日付と解釈されます。

これが数値として内部で扱われるため、四則演算で使用することができます。例えば、
01/01/97'DATE+14 は、日付'15/01/97' が算出される有効な式です。

参照: [指定された日付に対応する年の最初の日付を取得するには](#)

指定された日付に対応する曜日名を取得するには

指定された日付が何曜日になるかを取得し、テキストに変換することは非常に難しいことかもしれません。
しかしこれを行うには、CDOW() 関数を使用することができます。

この関数を使用する利点の一つは、プロジェクトが他の国で使用された場合、ユーザの選択した言語で曜日名を返すということです。

ヒント :	週単位の日付計算を行う必要がある場合は、DOW() 関数を使用することができます。この関数は週単位で日数(日曜日を 1 とした、日数)が返ります。フロー内で使用するには、こちらの方が便利です。
--------------	--

参照: [日付値を文字列に変換するには](#) トピックの **日付書式** セクション。

CDOW() 関数を使用するには :

CDOW()関数の構文は以下の通りです。

`CDOW(date)`

date …… 日付項目 (05/06/2008'DATE のように直接日付を指定するか、日付データを返す式を指定)
この関数は、曜日の名前(英語)を含む文字列を返します。

上記の 05/06/2008'DATE 中の文字列'DATE'は、Magic xpi のリテラルです。このリテラルを使用すると、文字列は日付と解釈されます。

これが数値として内部で扱われるため、四則演算で使用することができます。例えば、
01/01/97'DATE+14 は、日付'15/01/97' が算出される有効な式です。

指定された日付に対応する月名を取得するには

指定された日付に対応する月名を取得し、テキストに変換することは非常に難しいことかもしれません。しかしこれを行うには、CMonth() 関数を使用します。

この関数を使用する利点の一つは、プロジェクトが他の国で使用された場合、ユーザの選択した言語で月の名前を返すということです。

ヒント: 月数の計算を行う必要がある場合は、Month() 関数を使用することができます。この関数は、名前ではなく月数を数値で返します。フロー内で使用するには、こちらの方が便利です。

CMonth() 関数を使用するには:

CMonth()関数の構文は以下の通りです。

CMonth(*date*)

date …… 日付項目 (05/06/2008'DATE のように直接日付を指定するか、日付データを返す式を指定)
この関数は、月名(英語)を含む文字列を返します。

上記の 05/06/2008'DATE 中の文字列'DATE'は、Magic xpi のリテラルです。このリテラルを使用すると、文字列は日付と解釈されます。

これが数値として内部で扱われるため、四則演算で使用することができます。例えば、
01/01/97'DATE+14 は、日付'15/01/97' が算出される有効な式です。

参照: [指定された日付の年/月/日の各部分を取得するには](#)

指定された日付の年/月/日の各部分を取得するには

フロー内で日付データを使用する場合、日付データを分離する必要があるかもしれません。例えば、その年の1ヶ月間の全レコードを要約するような場合が考えられます。

日付データを分離するには、以下の3つの関数があります。

- **Day(*date*)** …… その月の日数を表す数値を取得します。
- **Month(*date*)** …… 月数を表す数値を取得します。
- **Year(*date*)** …… 年数を表す数値を取得します。

date …… 日付項目 (05/06/2008'DATE のように直接日付を指定するか、日付データを返す式を指定)

上記の 05/06/2008'DATE' 中の文字列'DATE'は、Magic xpi のリテラルです。このリテラルを使用すると、文字列は日付と解釈されます。

これが数値として内部で扱われるため、四則演算で使用することができます。例えば、01/01/97'DATE'+14 は、日付'15/01/97' が算出される有効な式です。

- ヒント：**
- 月数に基づいた計算をする必要がある場合は、異なる関数、**Month()**を使用してください。これは、フロー内で月数の数値を返します。
 - 月の名前や曜日の名前を表示させたい場合は、**CMonth()** 関数や **CDOW()** 関数を使用します。これらは、文字列で名前を返します。参照：[指定された日付に対応する月名を取得するには](#)

指定された日付の週単位の日数取得するには

プログラムの内容によっては、週単位で処理を行う必要があるかもしれません。例えば、金曜日に給料が配達される場合、指定された日付が金曜日か否かを確認する必要があるはずです。

値が数値で返るようであれば、曜日を確実にチェックできます。特に、アプリケーションが他言語で実行されるような場合、このオプションはスペルや大文字／小文字の問題を低減させることになります。

指定された日付から週番号を取得するには、**DOW()**関数を使用します。

DOW() 関数を使用するには：

DOW(date) は、週単位での日数取得します。返される値は、1(日曜日)から 7(土曜日)までです。

例えば、2007/08/19 は日曜日なので **DOW('2007/08/19'DATE)** は、1 を返します。

ヒント： 指定された日付の曜日名(英語)を取得する場合は、**CDOW()**関数を使用します。

参照：[指定された日付に対応する曜日名を取得するには](#)

日付値を文字列に変換するには

Magic xpi における日付値は基本的には数値で、0001/01/01 以降の日数として扱われます。しかし、日付を文字列に含めたい場合、日付を文字列に変換する必要があります。このような場合、**DStr()** 関数を使用して行います。

日付書式

日付を表示する場合、Magic xpi では複数の書式を指定することができます。日付の書式は、どのように表示されるかを指定します。書式は、日付を解釈する一種のテンプレートです。文字で示されたプレースホルダは、どのように日付を解釈するかを示すために使用されます。例えば、2007 年 6 月 30 日 は以下のように扱われます。

書式	どのように表示されるか	コメント
YY/MM/DD	07/06/30	
YYYY/MM/DD	2007/06/30	
YYYYMMDD	20070630	
YYYY-MM-DD	2007-06-30	

YYYY	2007	
YY	07	
MMMM	June	
DDD	181	ユリウス歴での年内の日数
DDDD	30th	
WWW	Sat	
WWWWWWWWW	Saturday	
##/##/####	Depends	Magic.ini ファイルで設定されている DateMode で定義された値に依存します。

DStr() 関数を使用する：

DStr()関数の構文は以下の通りです。

DStr (date, picture)

戻り値: 指定された書式による文字列。DStr() 書式を使用します。

この関数がどのように動作するか、いくつかの例があります。

Magic.ini ファイルの中で設定している DateMode が **A** (アメリカ)と設定されている場合。

DStr ('01/28/92'Date,'WWWWWWWWW, MMMMMMMM DD, YYYY')

戻り値: 'Tuesday, January 28, 1992'.

DateMode が **E** (ヨーロッパ)と設定されている場合, DStr('28/01/96'Date,'DD/MM/YY')

戻り値: 28/01/96.

参照: [文字列で格納された日付を日付値に変換するには](#)

文字列で格納された日付を日付値に変換するには

通常 Magic xpi は、日付変換処理を行います。すなわち、フォーム上に日付を表示したり、フォームの日付を読み込んだり、データソースから読み込む場合、手動で変換する必要はありません。しかし、文字列を日付として取得するためにその文字列を解析する必要がある場合、若干のプログラムを組む必要があります。

このような場合、DVal() 関数を使用する必要があります。この関数は文字列を解析して日付データに変換するために指定された書式を使用します。書式と文字列の整合性は、開発者に依存しています。例えば以下ようになります。

式	結果
DVal('06/30/07', 'MM/DD/YY')	June 30, 2007
DVal('06/30/07', 'DD/MM/YY')	0

参照: [日付値を文字列に変換するには](#)

XML DateTime フォーマットを計算するには

Magic xpi は、XML DateTime フォーマットを簡単に計算することができます。これを行うために必要なものは、提供されている 4 つの Magic xpi 関数のうちの 1 つを使用します。

DateTimeFormat, GetDate, GetTime, or GetTimeZone.

DateTimeFormat() 関数を使用するには :

DateTimeFormat() の構文は以下の通りです。

DateTimeFormat(*date*, *time*, *timezone*, *format*)

date …… 日付項目 (05/06/2008'DATE のように直接日付を指定するか、日付データを返す式を指定)

注 意: 上記の 05/06/2008'DATE 中の文字列'DATE'は、Magic xpi のリテラルです。このリテラルを使用すると、文字列は日付と解釈されます。

これが数値として内部で扱われるため、四則演算で 사용할 ことができます。例えば、01/01/97'DATE+14 は、日付'15/01/97' が算出される有効な式です。

time …… 時刻項目 ('16:10:14'TIM のように直接時刻を指定するか、時刻データを返す式を指定)

timezone …… 世界標準時として使用したいタイムゾーンです。

format …… 使用したい XML DateTime のフォーマットで、YYYY-MM-DDThh:mm:ssTZD またはユリウス歴の日付フォーマットです。

ここに、この関数がどのように動作するかを示す例があります。

式: DateTimeFormat('29/04/2008'Date, '10:45:30'Time, '+05:30',1)。戻り値: 2008-04-29T10:45:30+5:30.

GetDate() 関数を使用するには :

この関数は 1 つの DateTime 文字列を受け入れ、文字フォーマットで入力値の日付セクションを返します。

GetDate() の構文は以下の通りです:

GetDate(*datetime*, *format*)

datetime …… DateTime 式です。

format …… 使用したい XML DateTime のフォーマットで、YYYY-MM-DDThh:mm:ssTZD またはユリウス歴の日付フォーマットです。

ここに、この関数がどのように動作するかを示す例があります。

式: GetDate('2008-04-29T10:45:30+5:30',1)。戻り値:2008-04-29.

GetTime() 関数を使用するには :

この関数は 1 つの DateTime 文字列を受け入れ、文字フォーマットの中で入力値の時間セクションを返します。

GetTime() の構文は以下の通りです:

GetTime(*datetime*, *format*)

datetime …… DateTime 式です。

format …… 使用したい XML DateTime のフォーマットで、YYYY-MM-DDThh:mm:ssTZD またはユリウス暦の日付フォーマットです。

ここに、この関数がどのように動作するかを示す例があります。

式: `GetTime('2008-04-29T10:45:30+5:30',1)`。戻り値: 10:45:30.

GetTimeZone() 関数を使用するには :

この関数は、日付と時間の入力値を受け入れ、文字フォーマットで入力値のタイムゾーンセクションを返します。

GetTimeZone() の構文は以下の通りです:

`GetTimeZone(datetime, format)`

datetime …… DateTime 式です。

format …… 使用したい XML DateTime のフォーマットで、YYYY-MM-DDThh:mm:ssTZD またはユリウス暦の日付フォーマットです。

ここに、この関数がどのように動作するかを示す例があります。

式: `GetTimeZone('2008-04-29T10:45:30+5:30',1)`。戻り値: +5:30.

参照: [ユリウス暦の日付を計算するには](#)

ユリウス暦の日付を計算するには

JD Edwards を使用する場合、すべての日付を 6 桁に変換するユリウス日付の特別なタイプを使用する必要があります。Magic xpi は、ユリウス日付の JD Edwards タイプを計算します。これによって Magic xpi の JD Edwards Enterprise One と JD Edwards World のコネクタの使用が簡単になります。

この目的のために、Magic xpi は 4 つの関数を提供します。

`DateTimeFormat`, `GetDate`, `GetTime`, or `GetTimeZone`.

これらの関数の各構文についての詳細は、[XML DateTime フォーマットを計算するには](#)のトピックを参照してください。

関数は同じ方法で動作しますが、[フォーマット]パラメータで、2 を入力します。これによって、関数に、ユリウス日付の値を返すように指示することになります。

このタイプのユリウス日付のためのフォーマットは、CYDYYY です。C が世紀(0=1900 と 1=2000)、YY は年数、DDD は日数です。

従って、111081 の場合、1 は世紀を示し、11 は年(2011)、081 はその年内の 81 日目となります。

ここで、関数のいくつかの例を示します。

式	結果
<code>DateTimeFormat('22/03/2011'Date, '', '', 2)</code>	111081
<code>GetDate('111081',2)</code>	2011/03/22
<code>GetTime('111081',2)</code>	10:45:30

GetTimeZone('111081',2)	+5:30
-------------------------	-------

配備

配備

以下のトピックは、Magic xpi プロジェクトの配備について扱っています。

[自動的に開始するようにプロジェクトを設定するには](#)

[配備する PC に、どのファイルをコピーする必要があるか](#)

[プロジェクトを配備環境に移動させる時にどの構成を修正する必要があるか](#)

[配備のために使用する実行可能ファイルを作成するには](#)

[プロジェクトのディレクトリスキーマコンポーネントがネットワークドライブからファイルにアクセスすることができることを確認するには](#)

[Web サーバのエイリアスを定義する時に、クライアントのユーザとパスワードを取得するには](#)

[配備されたプロジェクトで HTTP リクエストのボディを受け取るには](#)

[HTTP リクエストの HTTP Verb を取得するには](#)

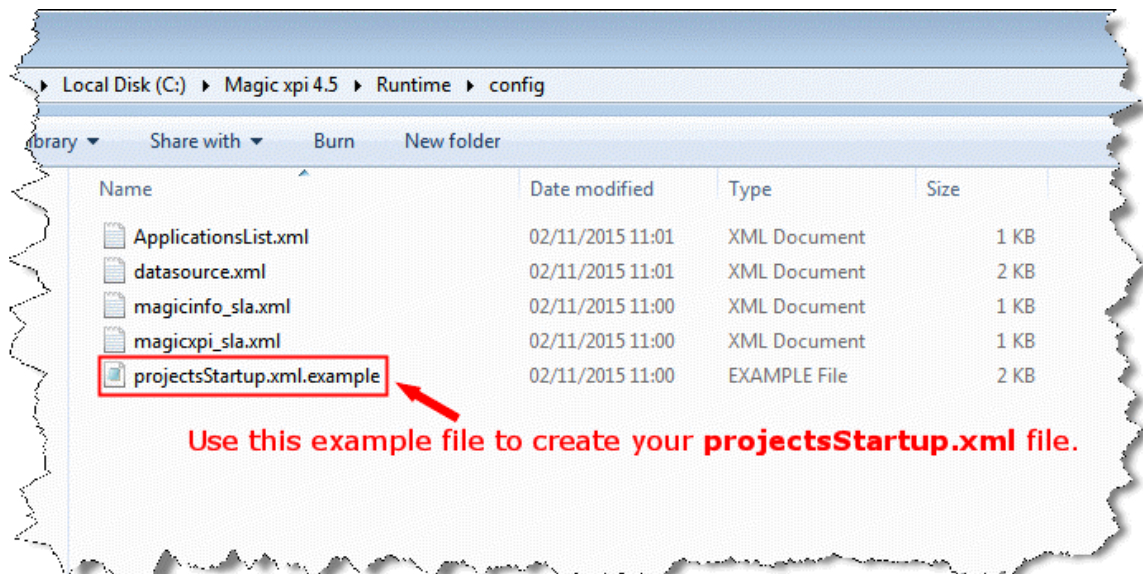
[運用環境でコンポーネントのロギングを保持する必要があるか](#)

[Web サービスのトリガを配備するには](#)

[ユーザ定義のコンポーネントを配備環境に移動するには](#)

[自動的に開始するようにプロジェクトを設定するには](#)

プロジェクトのスタートアップ構成は、**projectsStartup.xml** ファイルで定義されます。ここには、自動的に開始されるプロジェクトのリストが含まれています。**projectsStartup.xml.example** と名付けられたこのファイルのサンプルは、**<Magic xpi インストール>¥Runtime¥Config** フォルダ内に提供されています。**<Magic xpi インストール>¥Runtime¥Config** フォルダに置く必要のある **projectsStartup.xml** ファイルを作成するには、**projectsStartup.xml.example** ファイルを使用してください。



Magic xpi サービスが Magic Space を開始し配備すると、自動的に **projectsStartup.xml** ファイルにリストされるプロジェクトとサーバが開始されます。

注 意： **projectsStartup.xml** ファイルの名前は変更しないでください。名前を変更をすると、GSA は、Magic Space に **projectsStartup.xml** ファイルの情報を配備するすることができなくなります。

配備する PC に、どのファイルをコピーする必要があるか

プロジェクトの開発を終了したら、先に進み、フィールド内のプロジェクトの配備を準備することができます。適切に装備するには、配備する PC へコピーする内容を確認する必要があります。

これらは、以下の通りです。

- 開発 PC のプロジェクトフォルダ。
- プロジェクトで作成したユーザコンポーネント。

また、Systinet に Web サービスプロバイダの Jar を配備する必要があります。

プロジェクトを配備環境に移動させる時にどの構成を修正する必要があるか

プロジェクトを配備環境に移動する時、適切に動作させるためにプロジェクトのために構成を部分的に修正する必要があります。

- プロジェクトで使用され、**Magic.ini** ファイルで定義された環境変数は、配備 PC の **Magic.ini** ファイルにコピーされ、配備環境に対応するように修正する必要があります。
- プロジェクトの **ifs.ini** ファイルで定義された環境変数は、配備環境に対応するように修正する必要があります。
- **Magic.ini** ファイルの内部のデータベースの環境変数は、配備 PC の内部データベースを示すように修正してください。
- **start.xml** ファイルは、プロジェクトフォルダの新しい位置と新しいホストに対応するように修正してください。
- 手動での起動とシャットダウンができるように、**<My Documents>\Magic\projects\<プロジェクト名>\<現在のプロジェクト>**にある開始と終了のリンクを新しいプロジェクトの位置を示すように修正してください。
- 自動起動する新しいプロジェクトをリストに反映するために、**<Magic xpi インストール>\Runtime\Config** フォルダの **projectsStartup.xml** ファイルを修正してください。
- サーバの **ifs.ini** ファイルの **LicenseName** 特性は、運用ライセンス(WindowsPC の場合は **IBPRSRVI**、非 Windows マシンの場合は **IBPRSRVN**)に設定してください。

配備のために使用する実行可能ファイルを作成するには

Magic xpi スタジオ を使用している場合、Magic xpi プロジェクトまたは **.mgxpiproj** ファイルにアクセスしています。この **.mgxpiproj** ファイルは、ソースファイルのコレクションを参照します。これは **<My Documents>\Magic\projects\<プロジェクト名>\<現在のプロジェクト>\Source** に XML フォーマットで保持されています。

しかし、プロジェクトが完成したら、インストール用にパッケージ化したいはずです。パッケージ化されたファイルは、Magic xpi の実行可能ファイルまたは **.ibp** として知られていて、ビルドまたはリビルドの処理によって作成されます。**.mgxpiproj** と **.ibp** ファイルの主要な違いは以下の通りです。

.mgxpiproj	.ibp
Magic xpi プロジェクト	Magic xpi の実行可能ファイル
Magic xpi スタジオを開きます。	統合プロジェクトを実行します。

XML ソースファイルが存在している必要があります。	XML ソースのパッケージされたバージョンが含まれています。
----------------------------	--------------------------------

プロジェクトを配備する時は、**.ibp** ファイルと他のサポートファイル(フォント、色、コンポーネントなど)だけを配備する必要があります。

また、配備の際、Magic xpi の実行版だけをインストールするだけで十分です。

実行可能な**.ibp** ファイルは、簡単に作成することができます。

実行可能ファイルを作成するには：

1. プロジェクトを開いてください。
2. [ビルド] メニューを開き、[ソリューションのビルド] または [ソリューションのリビルド] のどちらかを選択してください。

ダイアログボックスが閉じたら、**.ibp** ファイルが <My Documents>¥Magic¥projects¥<プロジェクト名>に作成され、利用できるようになります。

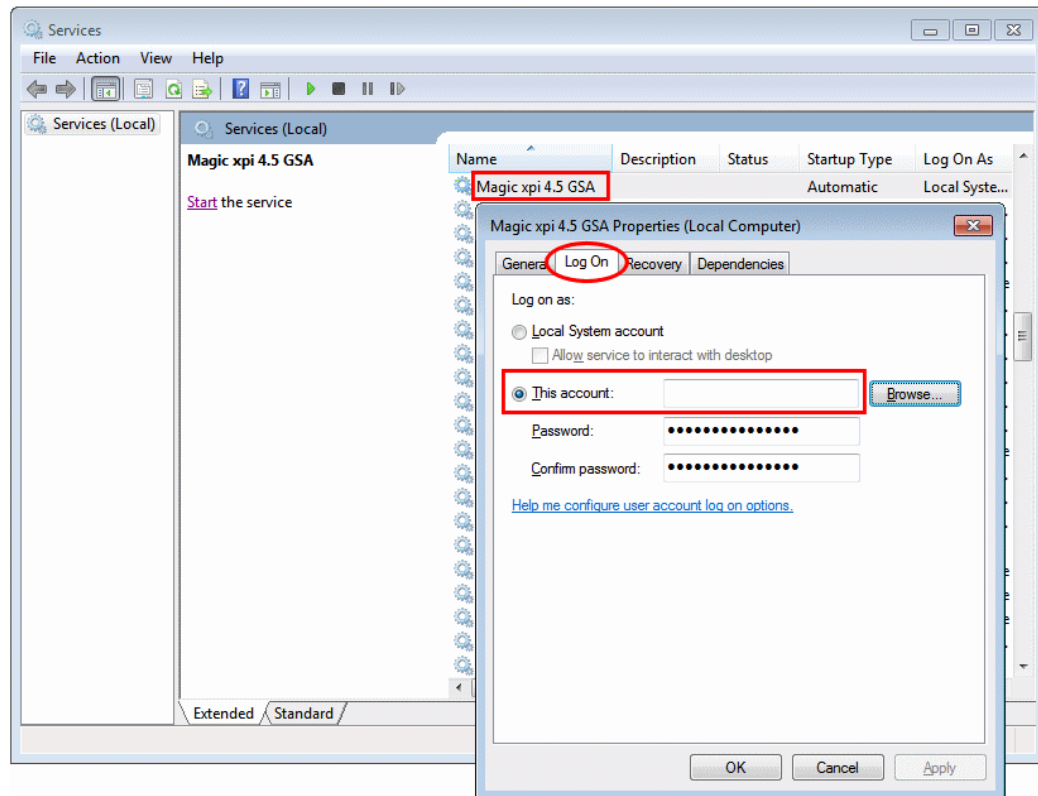
プロジェクトのディレクトリスキャナコンポーネントがネットワークドライブからファイルにアクセスすることができることを確認するには

プロジェクトのディレクトリスキャナコンポーネントがネットワークドライブからファイルにアクセスすることができることを確認するためには、ネットワークリソース(システム管理者によって定義されます)にアクセスするために、証明書を持っていることを確認する必要があります。その際、必要な位置でファイルの削除や書込、リネームが可能です。

これを行うには、以下のようにします。

1. [スタート] メニューから [ファイル名から実行] をクリックしてください。[ファイル名から実行] ダイアログボックスが開きます。
2. [ファイル名から実行] ダイアログボックスで **services.msc** を入力し、[OK] をクリックしてください。[サービス] ダイアログボックスが開きます。
3. [サービス] ダイアログボックスで、**Magic xpi4.6GSA** サービスをダブルクリックしてください。[サービスプロパティ] ダイアログボックスが開きます。
4. [ログオン] タブをクリックしてください。

5. [アカウント] オプションを選択し、アカウント名とパスワードを入力してください。



6. [OK] をクリックしてください。

Web サーバのエイリアスを定義する時に、クライアントのユーザとパスワードを取得するには

IIS の認証設定が **BASIC 認証** に設定されている場合、Magic xpi への HTTP リクエストはユーザとパスワードを必要とします。指定されない場合、リクエストは受け取りません。

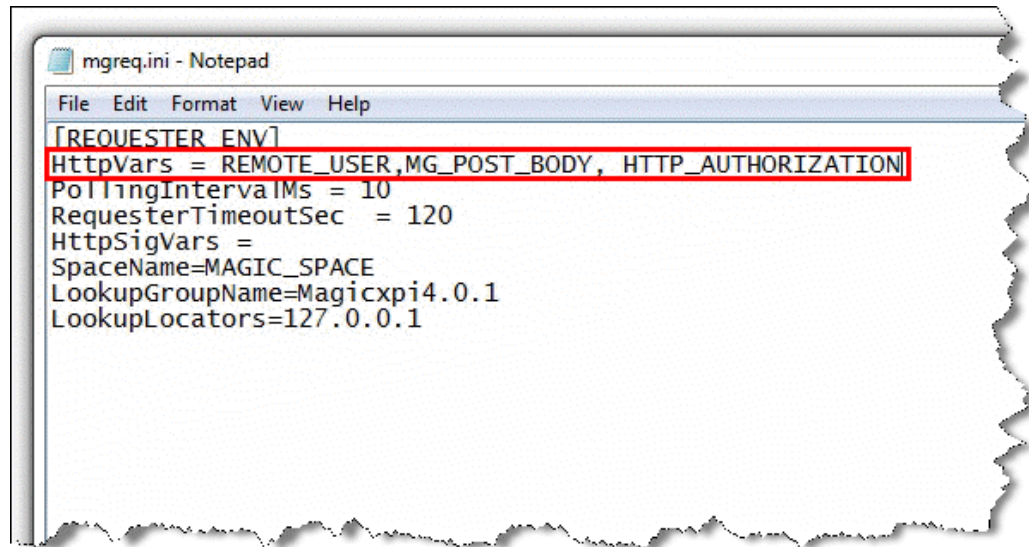
これらの詳細は、HTTP_AUTHORIZATION リクエストの HTTP ヘッダで利用可能です。

mgreq.ini ファイルで HTTP 変数を定義することで保持することができ、その情報を検索するために GetParam 関数を式エディタに手動で入力することができます。(この関数は、Magic xpi の関数リストでは選択できませんが、Magic xpa の機能を利用しています。)

これを行うには、以下のようになります。

1. **mgreq.ini** ファイル(**Runtime\scripts\config** ディレクトリ)に以下のように HTTP 変数を追加してください。

HttpVars = REMOTE_USER,MG_POST_BODY, HTTP_AUTHORIZATION



2. IIS を再起動してください。
3. プロジェクトを開いてください。
4. HTTP トリガを含むフローにおいて、以下の関数を使用して、任意の変数を更新してください。

GetParam ('HTTP_AUTHORIZATION')

変数には、Base64 フォーマットのユーザ ID とパスワードが含まれています。(例えば: 'Basic XXXXXXXX')

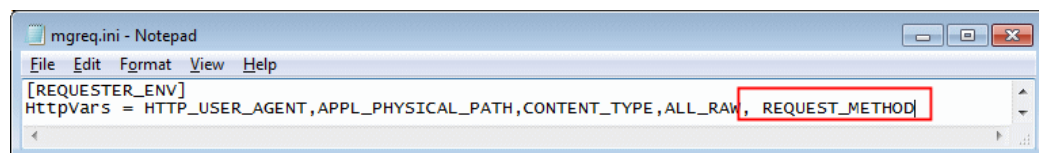
配備されたプロジェクトで HTTP リクエストのボディを受け取るには

配備されたプロジェクト内で HTTP リクエストのボディを受け取るには、<Magic xpi インストール>\Runtime\scripts\config フォルダ内にある mgreq.ini ファイルの HttpVars パラメータに MG_POST_BODY プロパティを追加して、IIS を再起動してください。

実行時に、HTTP ボディは C.HTTP_Body コンテキスト変数で有効になります。

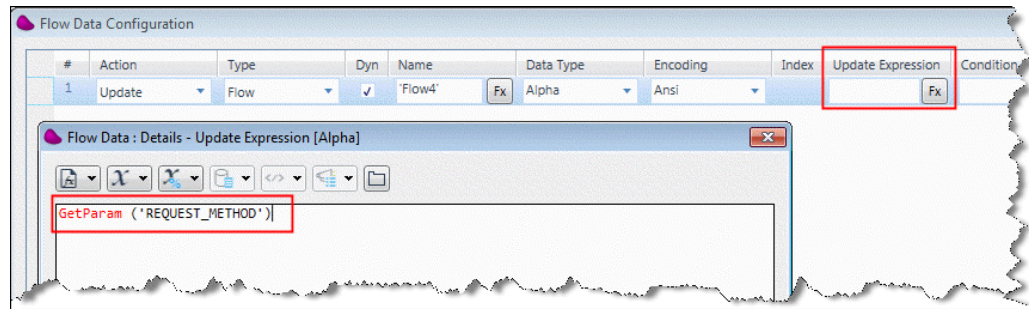
HTTP リクエストの HTTP Verb を取得するには

1. <Magic xpi インストール>\Runtime\scripts\config フォルダ内の mgreq.ini ファイルで、HttpVars パラメータを有効にし、変数 REQUEST_METHOD を追加してください。



2. IIS を再起動してください。

3. HTTP トリガを含むフローにおいて、**verb** を受け取るために、フローデータユーティリティの「更新式」カラムで以下の式を使用してください。 **GetParam ('REQUEST_METHOD')**



プロジェクトを実行すると、GET、POST、PUT、DELETE、および HEAD などの値が取得されます。

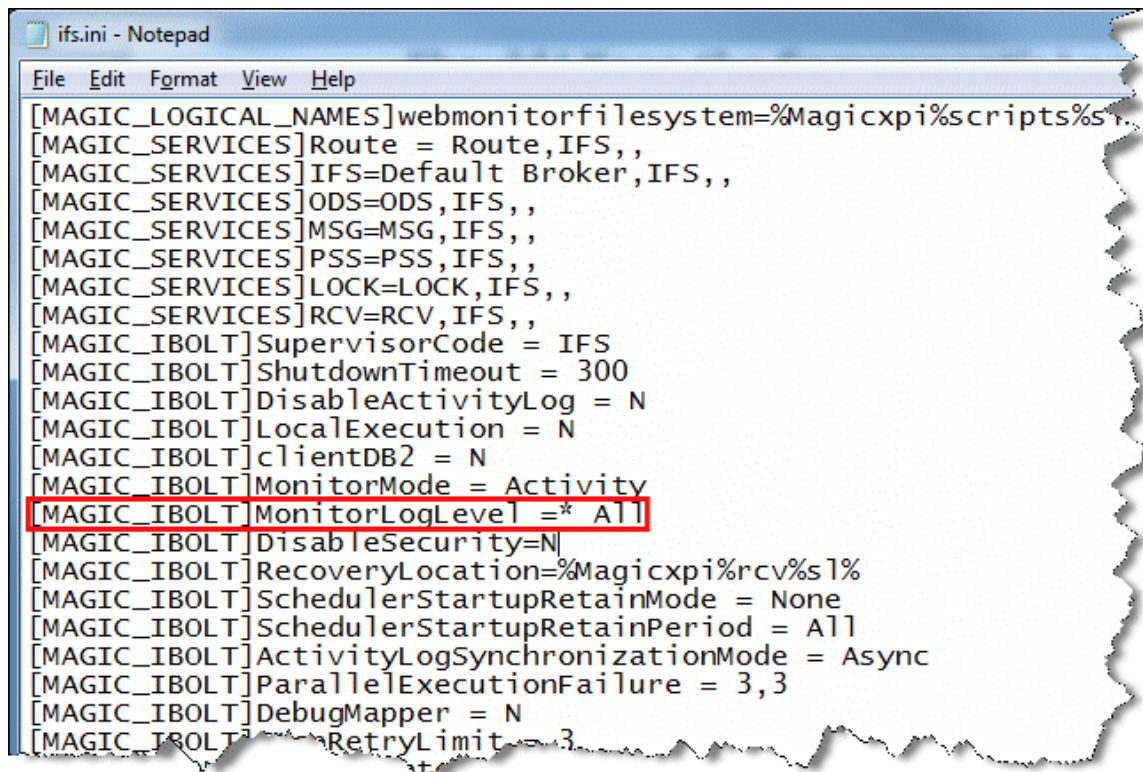
運用環境でコンポーネントのロギングを保持する必要があるか

Magic xpi は、プロジェクトについての追加情報を取得する様々なロギングオプションを提供しています。しかし、ロギングを有効にするとプロジェクトの実行の速度は低下します。

従って、最適なプロジェクト性能を得るために、特定のログタイプを無効にすることができます。デバッグする必要がある特定の問題がない限り、**Mapper.log** と **SFDC** ログなどのログをデバッグすることは運用環境では使用しないでください。

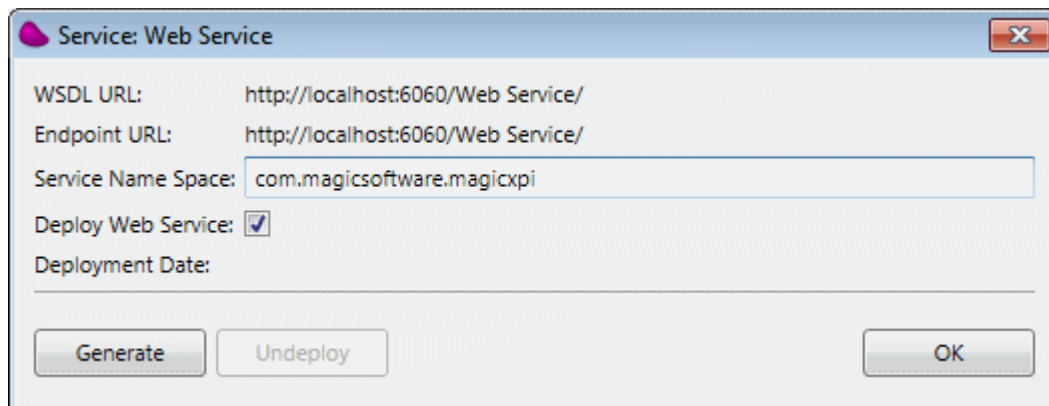
さらに、内部データベースに書き込んだり、モニタに表示されるロギングメッセージのいくつかを無効にすることができます。これらのフィルタリング選択を更新した場合、適用するためにサーバを再起動する必要があります。

ifs.ini ファイルの `MonitorLogLevel= flag` によってモニタのロギングレベルを制御することもできます。しかし、このフラグに変更を加えた場合、変更内容を反映させるためにサーバを再起動する必要があります。



Web サービスのトリガを配備するには

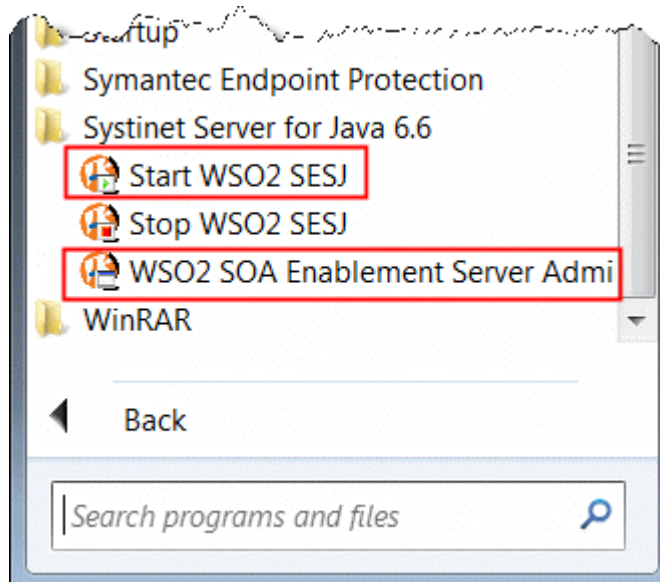
Magic xpi で Web サービスサーバを作成する場合、[サービス:Web サービス]ダイアログの、[Web サービスの配備]のチェックボックスを選択することで Systinet 内に自動的に Web サービスを配備させることができます。



代わりに、以下の手順に従うことで Systinet から手動でモジュールを配備することができます。

1. Systinet コンソールを開いてください。2つの方法のうちのどれかを行ってください。
 - a. 以下の URL を使用してください: **http://<Systinet host>:6060**。例えば:
http://localhost:6060。

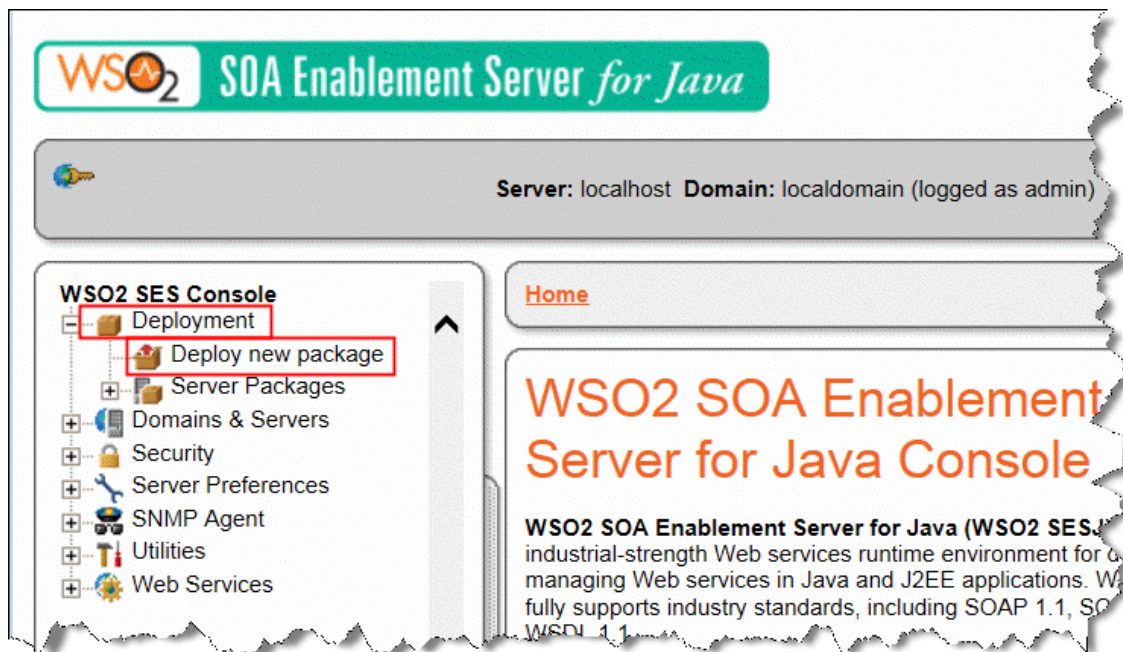
- b. Windows の [スタート] メニューを開き、 **Start WSO2 SESJ** オプションと **WSO2 SOA Enablement Server Administration Console** オプションをクリックしてください。



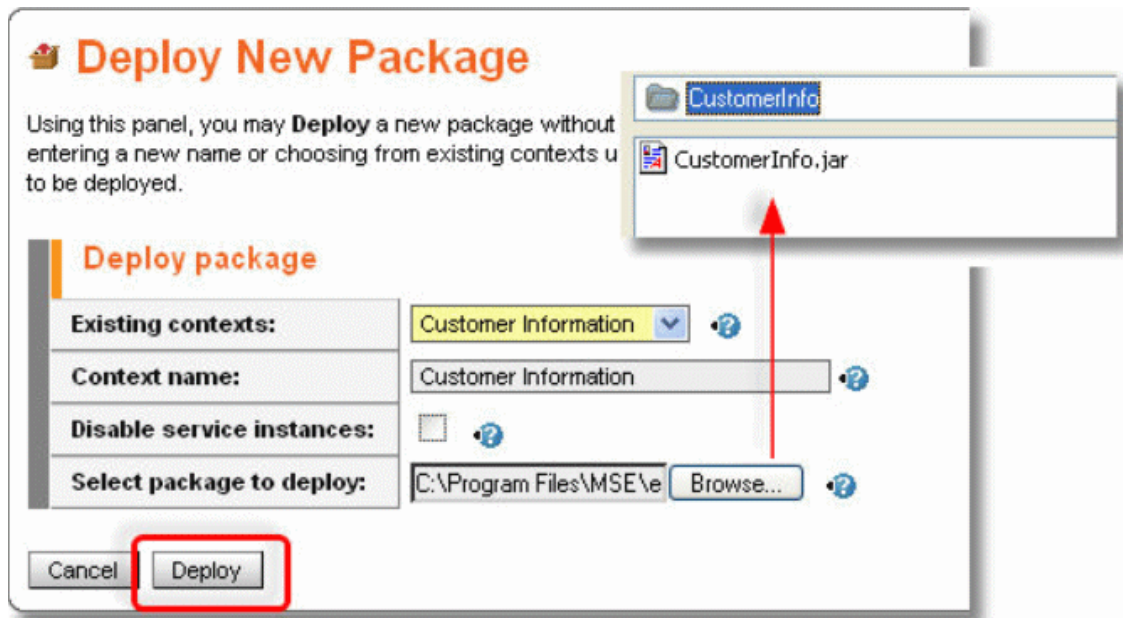
- c. **stinet host>:6060**. 例えば : **http://localhost:6060**.

注意 : Systinet へのデフォルトログインオプションは、User Name = admin と Password = changeit です。

2. **Deployment** をクリックして、**Deploy new package** に続けます。



3. コンテキストとコンテキスト名に入力してください。



4. 配備したいパッケージを選択してください。これは Magic xpi によって作成された.jar ファイルです。これは、**project\service\<サービス名>** フォルダにあります。

5. [Deploy] ボタンをクリックしてください。

これで Web サービスサーバが利用できます。

ユーザ定義のコンポーネントを配備環境に移動するには

.NET ベースのユーザ定義コンポーネント

Magic xpi の.NET ユーティリティを定義すると、.NET ベースのユーザ定義コンポーネントが開発 PC 上のプロジェクトフォルダに作成されます。推奨方法として、プロジェクトフォルダをそのまま配備 PC に移動すると、プロジェクトを配備すると、利用可能なユーザ定義.NET コンポーネントが自動的に定義されることになります。そうしない場合は、開発 PC 上でコンポーネントを手動でプロジェクトのフォルダに移動する必要があります。

Java ベースのユーザ定義コンポーネント

Java ベースのユーザ定義コンポーネントを作成するためにコネクタビルダを使用することができます。コンポーネントのファイルは、開発 PC 上の<Magic xpi>%Runtime%addon_connectors フォルダに保存されます。開発 PC 上で同じ階層を持つこれらのコンポーネントを手動で同じフォルダに移動させる必要があります。

Java の実装が含まれた jar は、開発 PC からコピーする必要があります。jar は、**Magic.ini** ファイルまたは OS の classpath のどちらかで参照されるようにすべきで、それは<Magic xpi>%Java%lib フォルダに配置されています。

注	コネクタビルダを使用してユーザがユーザ定義コンポーネントを修正することを防止するには、metadata.xml ファイルのないコネクタフォルダをユーザに送ってください。
---	--

開発時の検討事項

開発時の検討事項

以下のトピックは、Magic xpi での開発時に検討する項目について扱っています。

[配備をより簡単にするために開発ガイドラインとは](#)

[Web サービスコンシューマを配備 PC に移行するには](#)

配備をより簡単にするために開発ガイドラインとは

プロジェクトの開発を始める前に、プロジェクトを配備したい時に助けとなる様々な選択を考慮しておいてください。

- データマップによって使用されたスキーマなどのファイルをプロジェクトフォルダにすべてを保持しておいてください。この方法によって、一方の中心的な位置の必要なファイルとプロジェクトフォルダに関連している他のファイルのほとんどを持つことになります。これによって、開発環境から配備環境にそれらを簡単にコピーすることができます。
- ファイルやコンポーネントに名前を割り当てる場合、小文字だけを使用してください。Windows OS ではファイルとコンポーネントの名前は大文字小文字を区別しません。しかし、UNIX または IBM i などの他の OS では大文字小文字を区別します。プラットフォームを横断して名前が利用できるように、小文字を使用してください。
- 環境変数は、開発環境から配備環境にプロジェクトをよりスムーズに移行できるようにします。物理的な記憶媒体、または OS の命名規則との明示的な関係を持つことなくアプリケーション要素のために名前を作成することができます。次のような環境変数を使用してください。
 - 外部のプロジェクトのフォルダ内のファイルの物理パス これは Magic xpi 内のすべてのコネクタと関連した一般的なガイドラインです。
 - FTP コンポーネントとディレクトリスキャナコンポーネント内の FTP パス。
 - Web サービスコンシューマ WSDL 位置とエンドポイント。参照：[Web サービスコンシューマを配備 PC に移行するには](#)
 - サーバ、メッセージングキュー、およびエイリアスの名前またはアドレス。
- ネットワークドライブで環境変数を使用することを推奨します。
- それは、プロジェクトのロジック(例えば、ハードコードされたリトライのループ数を定義する代わりに、この数値を保持する環境変数を使用し、環境変数が存在している場合、ハードコードされたデフォルトを定義します)と関連した構成。
- 原文のデータのために ANSI を使用しないでください。代わりに Unicode を使用してください。詳細は、推薦とヒントを参照してください。
- Magic xpi の開発と運用環境の両方で同じデータベースタイプを使用してください。

参照: サーバ性能の考察

Web サービスコンシューマを配備 PC に移行するには

Web サービスコンシューマのエンドポイントは、すでに移植可能で、エンドポイントを環境変数として定義することで、またはリソースから値を変更することによって変更することができます。

配備を成功するための唯一の問題と注意する重要な点は、WSDL の位置です。サービスをロードした後に、変更することはできません。

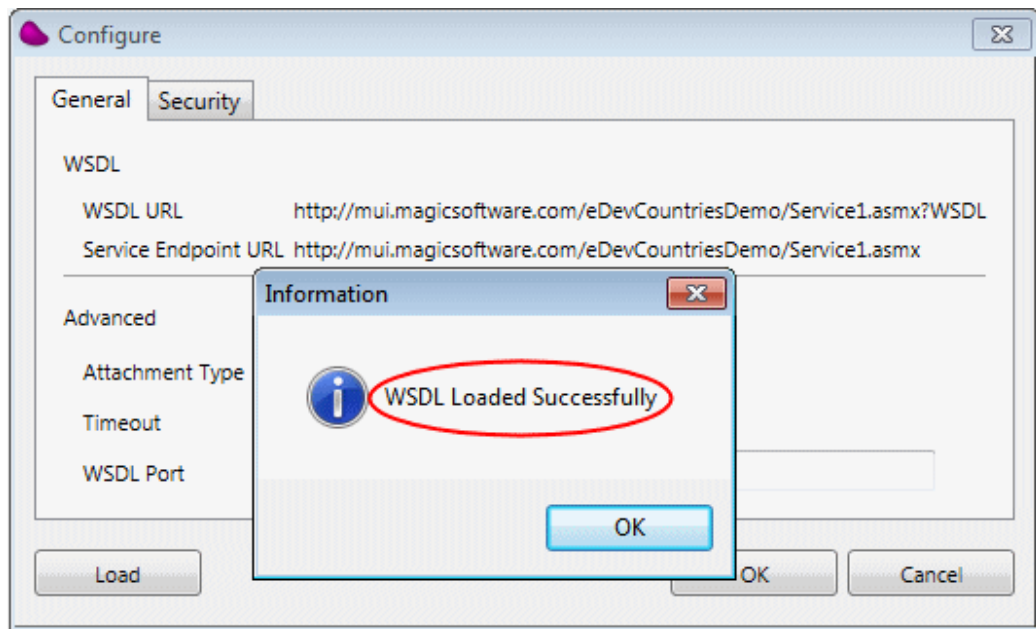
この問題を解決するには、WSDL URL のために環境変数を使用して、それを運用環境に対応した値に変更することを確認してください。

このためのステップは以下の通りです。

1. [設定] ダイアログボックスの [環境] セクションの中で、**ユーザ環境変数** の下の [追加] をクリックして、WSDL URL のために新しい環境変数を作成します。ここで、以下の URL へのパスを指定します。

http://mui.magicsoftware.com/eDevCountriesDemo/Service1.asmx?WSDL

2. [OK] をクリックしてください。
3. [設定] ダイアログボックスの [リソース] セクションで、Web サービスクライアントリソースの中で、[WSDL URL] 設定に移動し、パーセント記号(%)を含めた新しい環境変数を入力してください。ここでは、**%WsdIURL%** が入力されています。これは、上記で定義された環境変数です。
4. [構成] ボタンをクリックして、[設定] ダイアログボックスを開き、[読込] をクリックしてください。すべてが適切に設定されると、"WSDL の読み込みに成功しました"というメッセージが表示されます。



ディレクトリスキャナコンポーネント

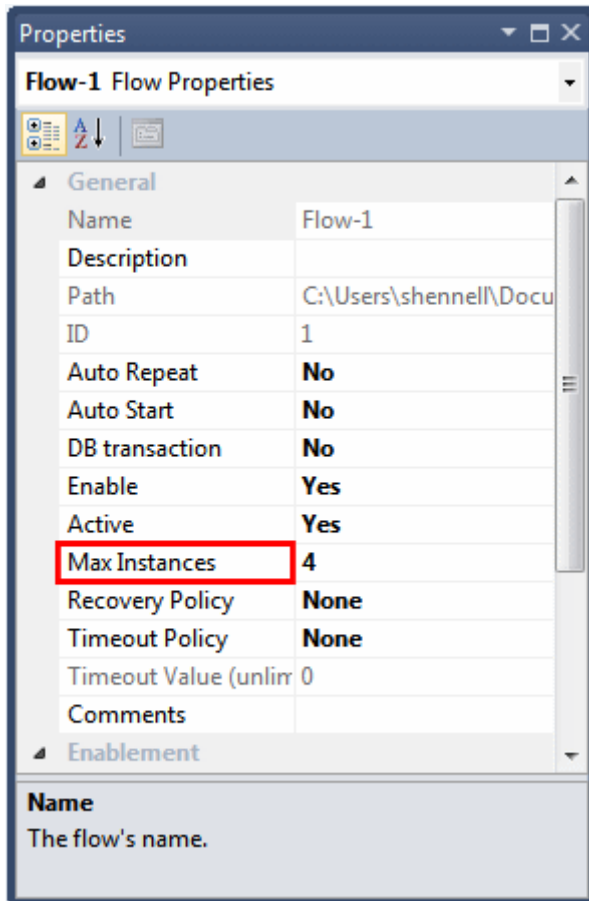
ディレクトリスキャナコンポーネント

以下のトピックは、Magic xpi のディレクトリスキャナコンポーネントについて扱っています。

[ディレクトリスキャナのトリガによって処理されたファイルの数を制御するには](#)

ディレクトリスキャナのトリガによって処理されたファイルの数を制御するには

ディレクトリスキャナのトリガによって処理されたファイル数を制限するには、フローの[プロパティ]ペインで設定する[最大インスタンス数]を使用してください。



この設定は、フローが持つことができる並行インスタンスの数を制限します。

しかし、ディレクトリスキャナは、まだ、TriggersBufferSize= flag(デフォルト値は 10)で定義された制限でファイルを取得し、Space

にそれらをロードし続けることに注意してください。

Dynamics CRM コネクタ

Dynamics CRM コネクタ

以下のトピックは、Magic xpi の Dynamics CRM コネクタについて扱っています。

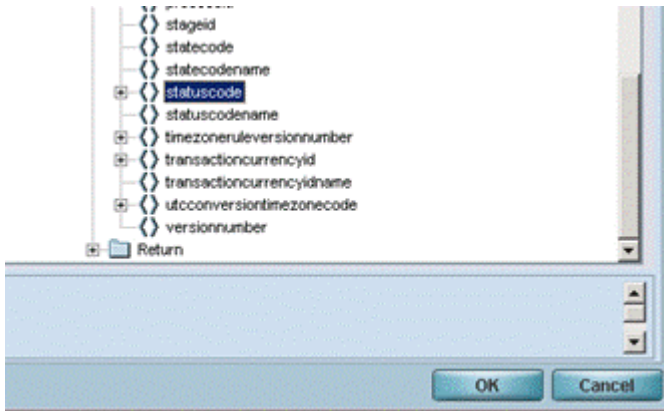
[非アクティブな状態にあるオブジェクトを更新するには](#)

[Dynamics CRM を使用して PhoneCall\(アクティブ\)を作成するには](#)

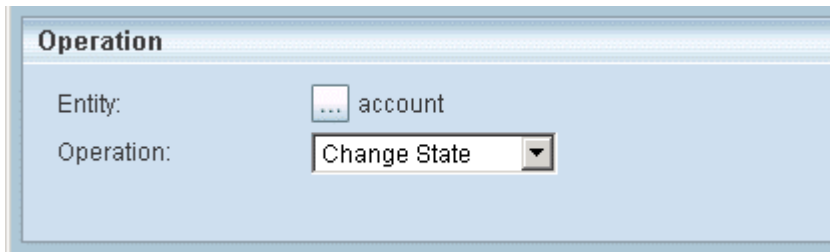
非アクティブな状態にあるオブジェクトを更新するには

オブジェクトを変更したり、更新したりした場合、そのオブジェクトが非アクティブ、またはクローズ状態にある時は、常に最初にステータスを更新する必要があります。

これは、新しいステータスを **StateCode** オブジェクトまたは **statuscode** フィールドにマッピングすることでは行われません。



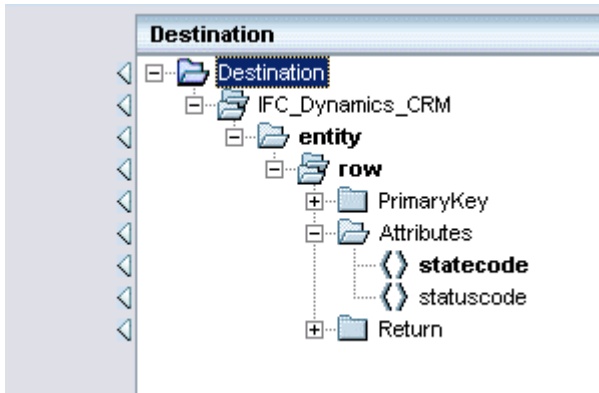
代わりに、**Change State** オペレーションを使用してこれを行います。



このオペレーションは、エンティティの状態を変更するために使用されます。各エンティティは、このオペレーションによって変更することができる Active、Inactive そして Open などのステータスがあります。

このオペレーションを使用する時、データマップには 2 つのノードがあります。

- **statecode** ステータスです。
- **statuscode** ステータスの意味です。



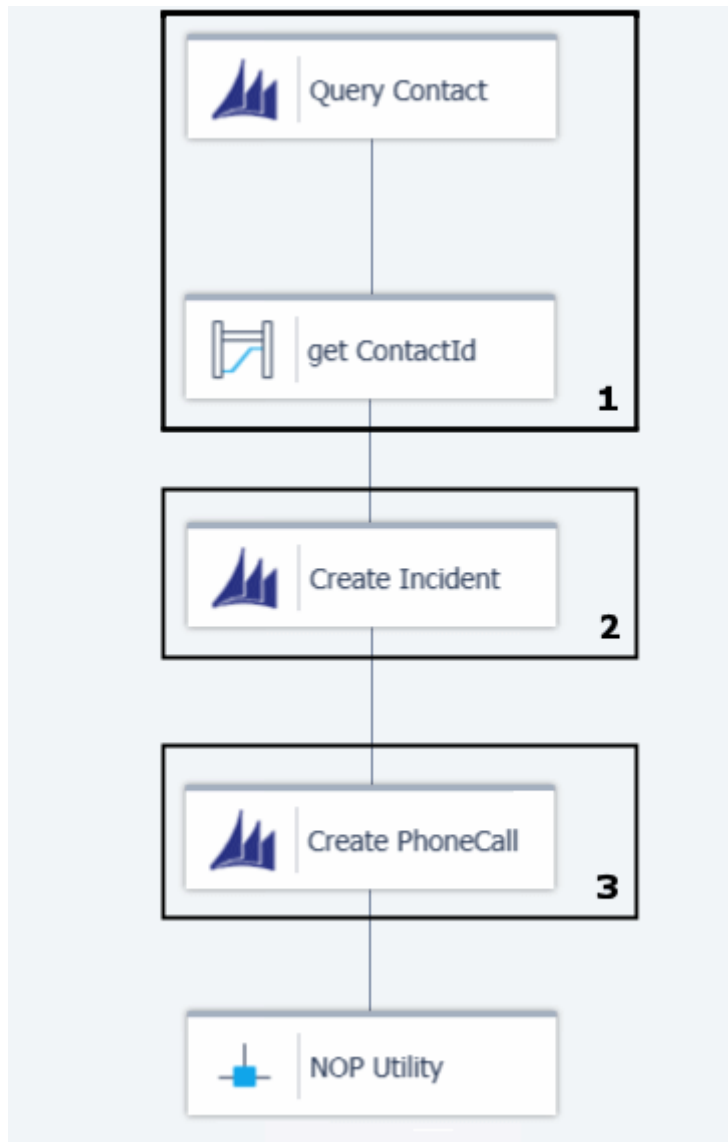
statecodes はステータスを表す数値ではなく、ステータスの文字列(**Active** や **Inactive** など)として使用する必要があります。Dynamics CRM はただ、特定の **statecode** のために有効な **statuscodes** を受け入れ、正しい数値を **statuscodes** に設定したことを確認しなければなりません。例えば、0(Open)の **statecode** を持つ読み込みエンティティを持つことができます。この **statecode** は、1(New)または 2(Contacted)の **statuscode** を持つことができます。

statecodes と **statuscodes** のリストについては、次を参照してください:[https://technet.microsoft.com/en-us/library/dn531157\(v=crm.7\).aspx](https://technet.microsoft.com/en-us/library/dn531157(v=crm.7).aspx).

Dynamics CRM を使用して PhoneCall(アクティブ)を作成するには

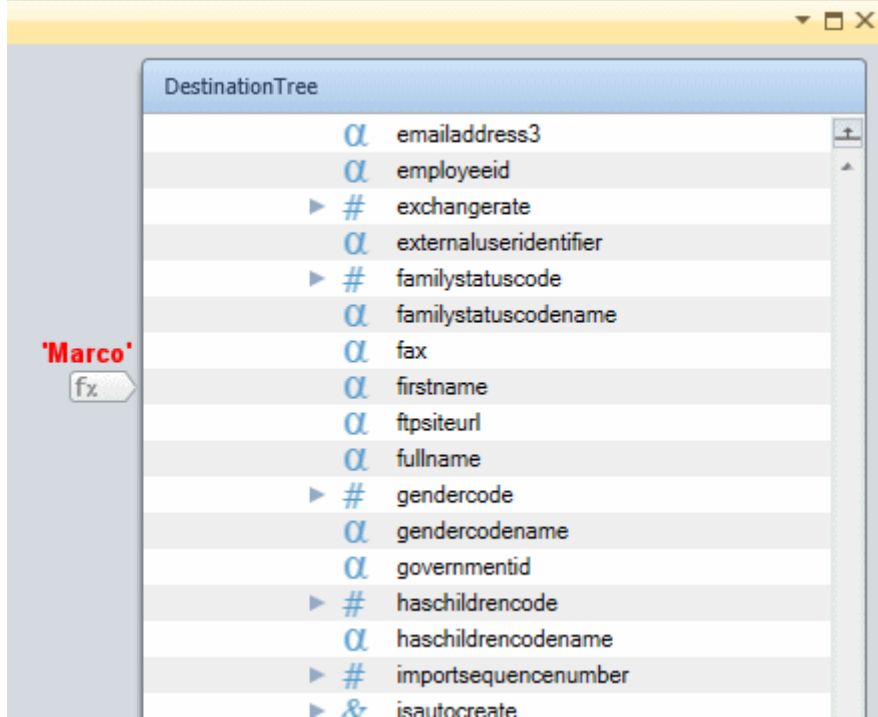
PhoneCall オブジェクトは、Case オブジェクトと関連しているアクティビティのタイプです。Case オブジェクトは、Contact オブジェクトまたは Account オブジェクトと関連することができます。従って、PhoneCall を作成するには、最初に、Contact/Account を定義し、次にアクティビティにドリルダウンする必要があります。

この例は、Contact のケースのために、どのように PhoneCall を作成するかについて説明しています。

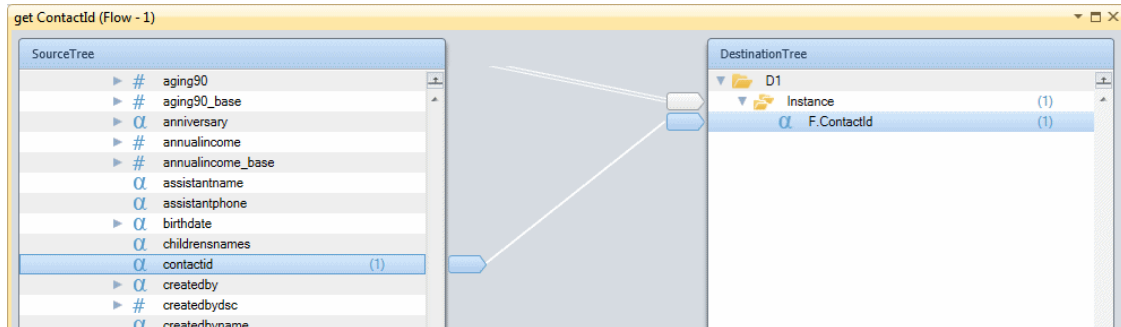


1. Contact の ID を取得します。

- a. この例において、ファーストネームが **‘Marco’** である **Contact** の ID を取得してください。

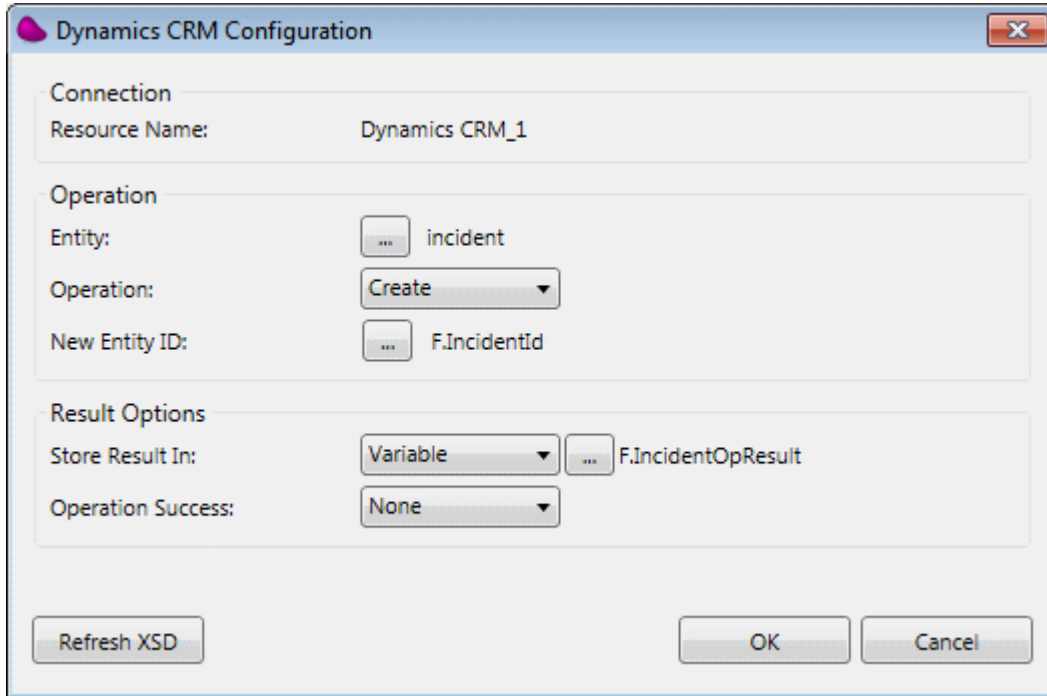


- b. 次に、**Contact** の ID を **F.ContactId** 文字変数に保存するために、データマップを使用してください。



2. その **Contact** のためにインシデントなオブジェクトを作成してください。Dynamics CRM では、インシデントは **Case** と呼ばれます。

- a. 新しく作成されたインシデント ID を **F.IncidentId** 文字変数に保存してください。



The image shows a 'Dynamics CRM Configuration' dialog box with the following fields and values:

Connection	
Resource Name:	Dynamics CRM_1

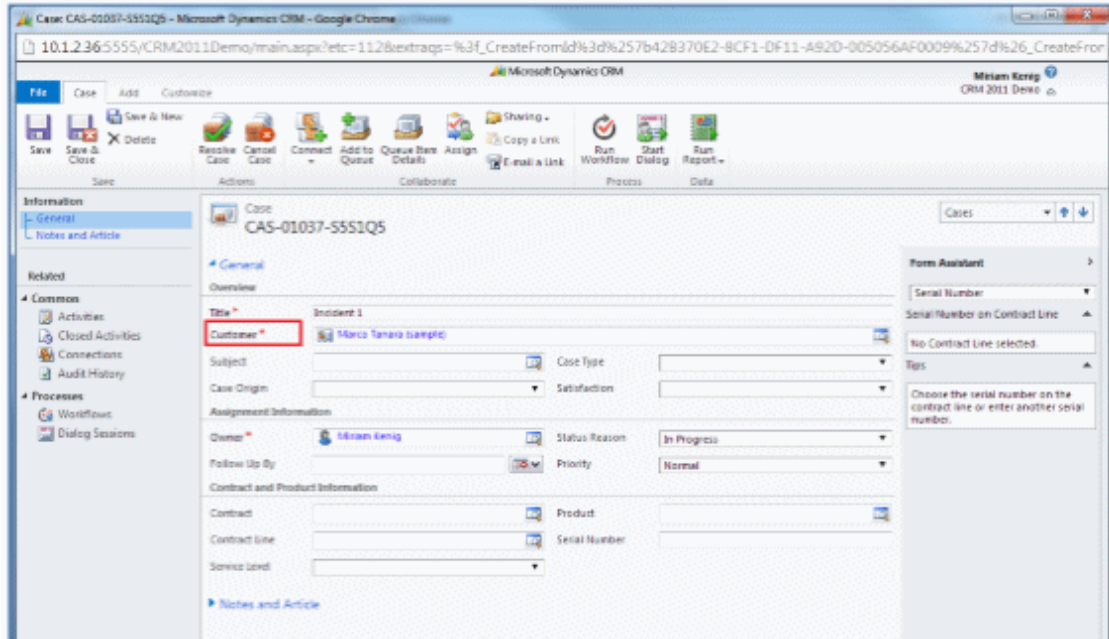
Operation	
Entity:	incident
Operation:	Create
New Entity ID:	F.IncidentId

Result Options	
Store Result In:	Variable F.IncidentOpResult
Operation Success:	None

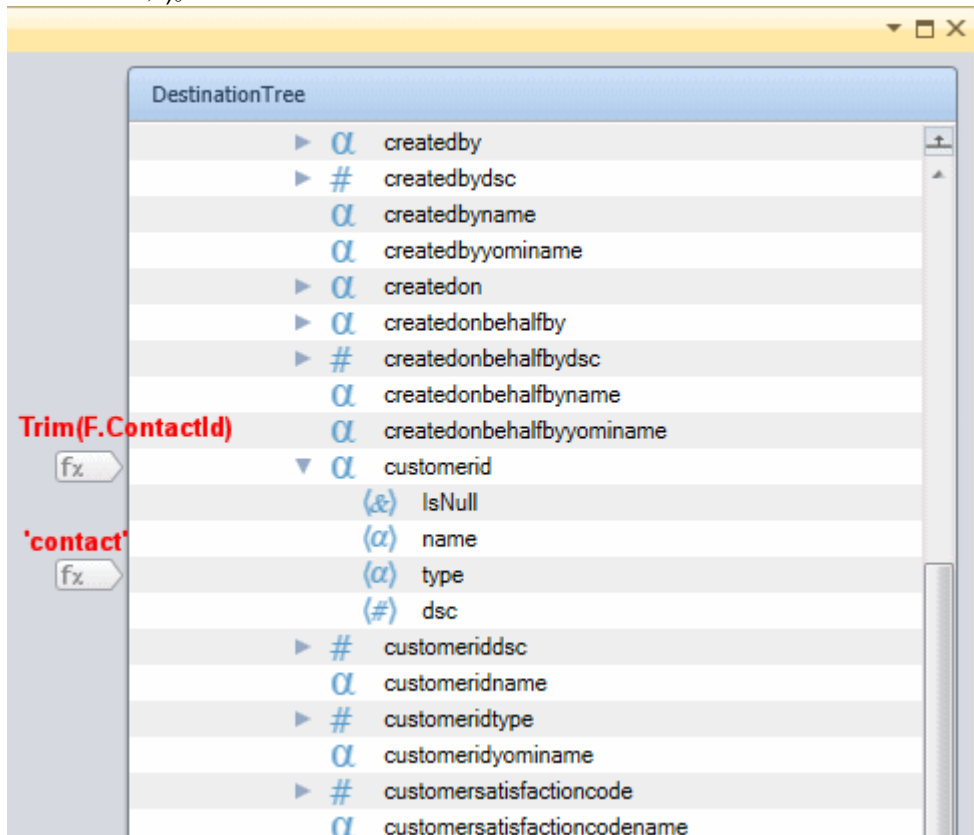
Buttons at the bottom: Refresh XSD, OK, Cancel.

Magic xpi 逆引き辞典

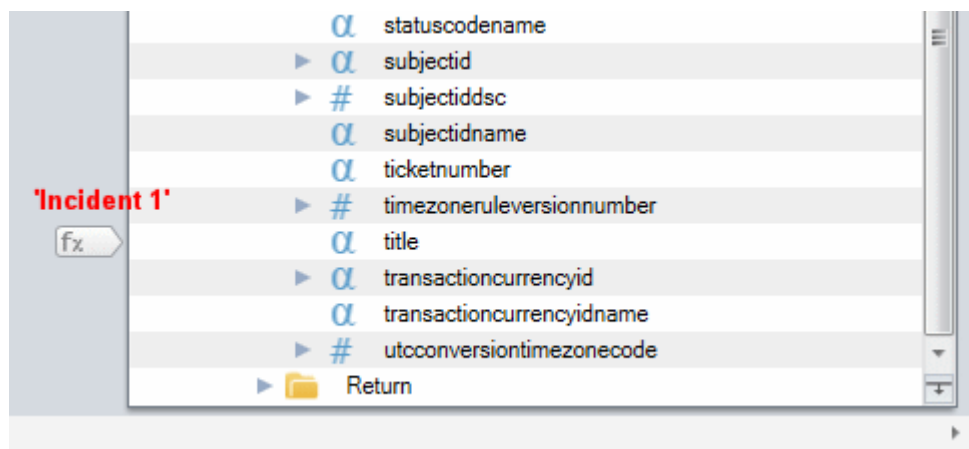
- b. インシデントを作成する時に、それを **Customer** に照会しなければなりません。
Web ベースの UI(以下)の場合、[Customer] フィールドのためにアスタリスクが必須になります。



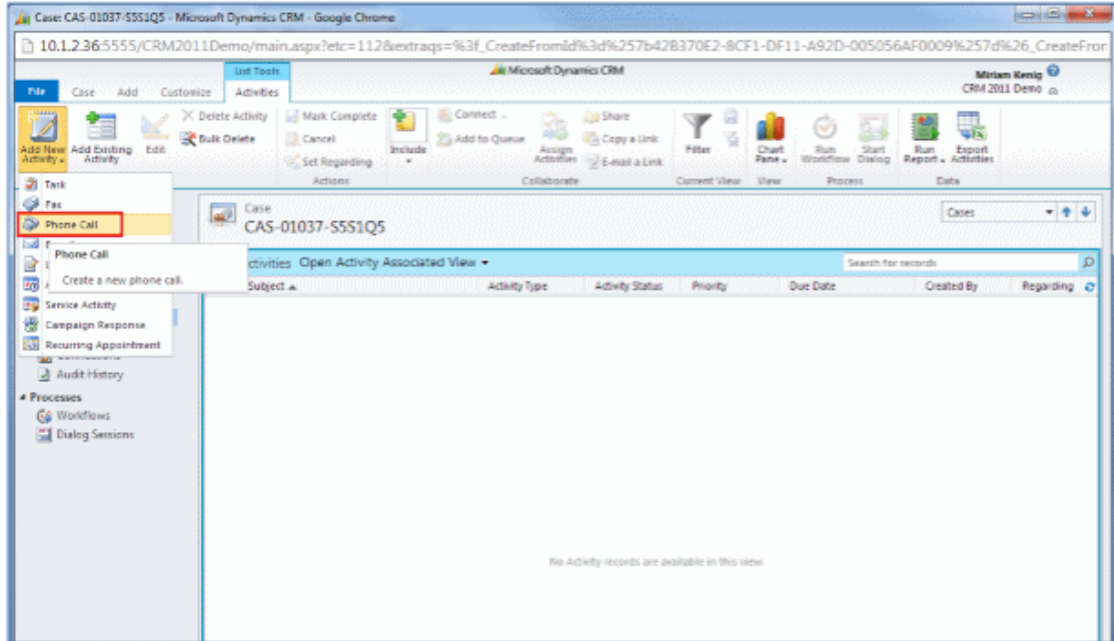
Magic xpi では、CustomerId を提供し、顧客のタイプを定義する必要があります。この場合に 'contact' です。



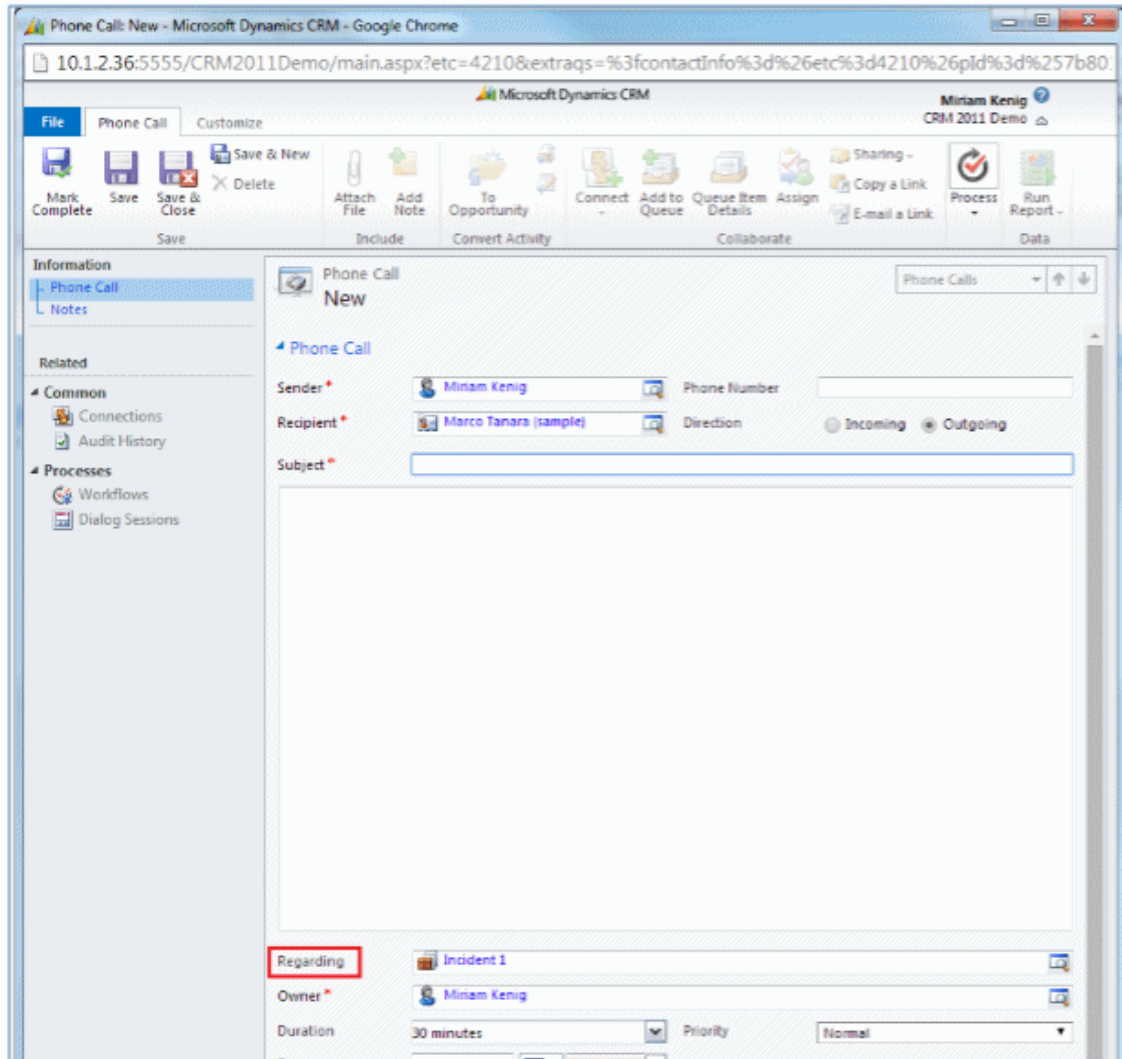
さらに、新しく作成されたインシデントのために名前を指定する必要があります。この例では、それを **'Incident 1'** と呼びます。



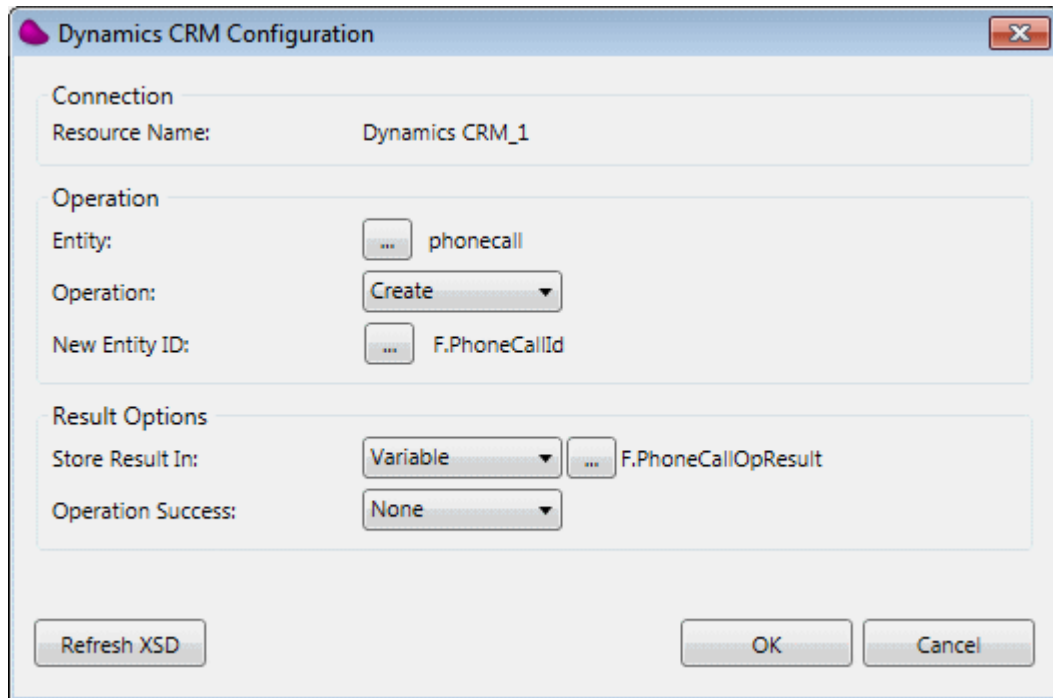
3. 以下のイメージで示すように、作成されたインシデントのために、**PhoneCall** オブジェクトを作成してください。



Web ベースの UI を考慮したフィールド名であることに注意してください。後で、このトピックは Magic xpi 経由でその属性にどのように値を設定するかを説明します。



Magic xpi で、新しく作成された PhoneCall の ID を **F.PhoneCallId** 文字変数に保存してください。



The image shows a 'Dynamics CRM Configuration' dialog box. It has three main sections: 'Connection', 'Operation', and 'Result Options'. In the 'Connection' section, 'Resource Name' is set to 'Dynamics CRM_1'. In the 'Operation' section, 'Entity' is 'phonecall', 'Operation' is 'Create', and 'New Entity ID' is 'F.PhoneCallId'. In the 'Result Options' section, 'Store Result In' is 'Variable' with a value of 'F.PhoneCallOpResult', and 'Operation Success' is 'None'. At the bottom, there are buttons for 'Refresh XSD', 'OK', and 'Cancel'.

Connection	
Resource Name:	Dynamics CRM_1

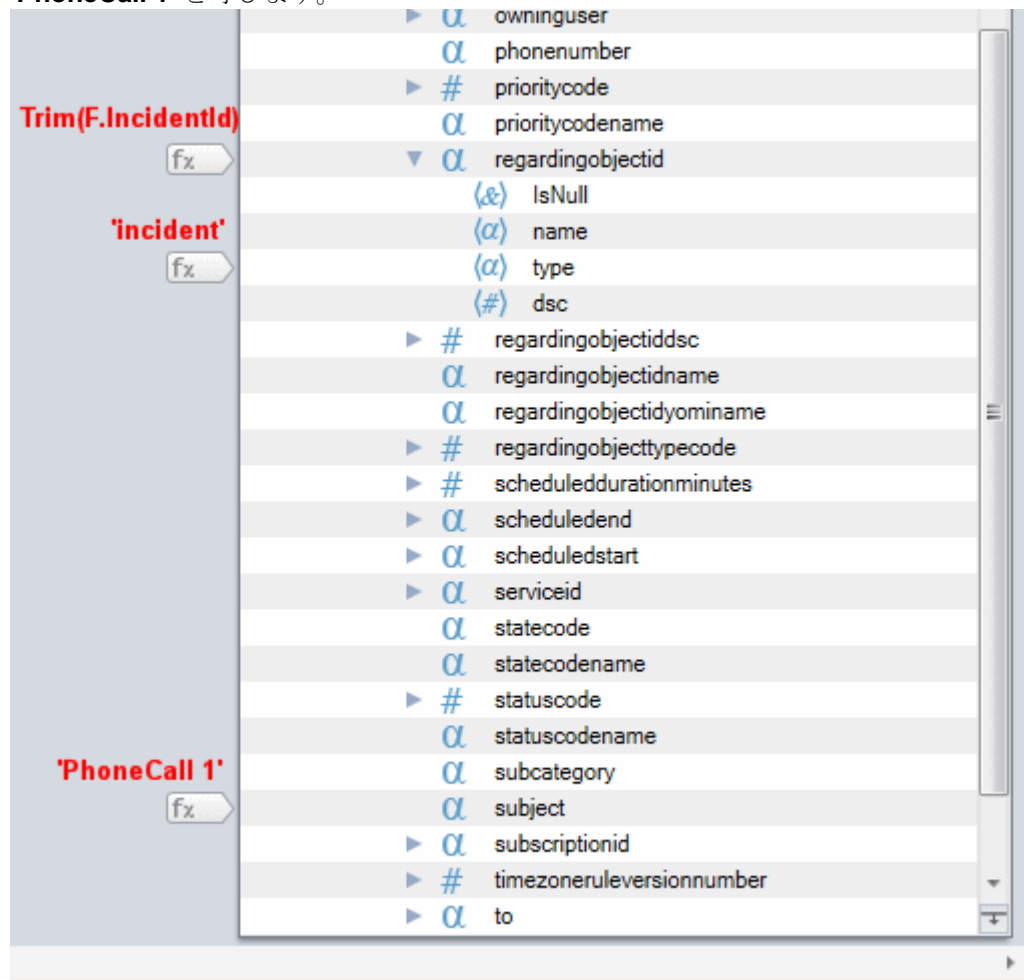
Operation	
Entity:	phonecall
Operation:	Create
New Entity ID:	F.PhoneCallId

Result Options	
Store Result In:	Variable F.PhoneCallOpResult
Operation Success:	None

Buttons: Refresh XSD, OK, Cancel

次に、**RegardingObjectId** とそのタイプを提供する必要があります。この例では **'Incident'** です。さらに、新しく作成された **PhoneCall** に名前を指定する必要があります。ここでは、

'PhoneCall 1' と呼びます。



エンコード

エンコード

以下のトピックは、Magic xpi でのエンコードについて扱っています。

[ファイルが指定されたエンコードで定義されることを確認するには](#)

[指定されたエンコードでファイルを作成するには](#)

[Unicode か ANSI、UTF-8 のファイルをメモリに読み込ませるには](#)

[データマップを使用する場合、Unicode か ANSI、UTF-8 のフラットファイルでレコードを処理するには](#)

[Unicode か UTF-8 BOM を BLOB ファイルに追加するには](#)

[UTF-8 の内容を含むバイナリ BLOB を Unicode BLOB に変換するには](#)

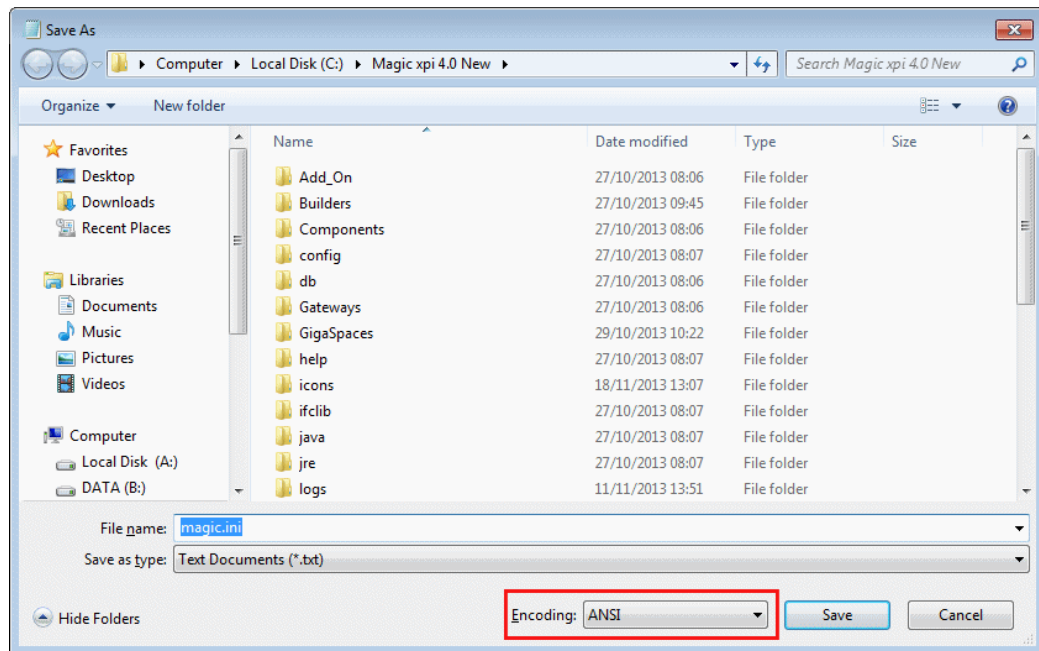
[UTF-8 の内容を含む Unicode BLOB をバイナリ BLOB に変換するには](#)

ファイルが指定されたエンコードで定義されることを確認するには

Magic xpi で指定したファイルのエンコーディングは簡単に確認することができます。ここでは、テキストと XML ファイルのエンコーディングを確認する方法について説明します。

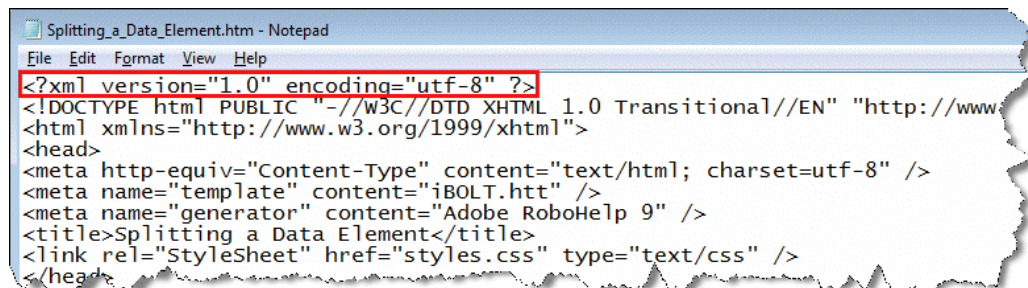
テキストファイルのエンコーディングを確認するには：

1. 関連するテキストファイルで右クリックして、[プログラムから開く] を選択し、続いて **Notepad** を選択します。テキストファイルが開きます。
2. テキストファイルの [ファイル] メニューから、[名前を付けて保存] を選択してください。
3. [名前を付けて保存] ダイアログボックスで、下のイメージで示するように [保存] ボタンのそばにある [エンコーディング] をクリックしてください。ここには、選択されたテキストファイルで使用するエンコーディングを表示しています。この場合、エンコードタイプは、**ANSI** です。



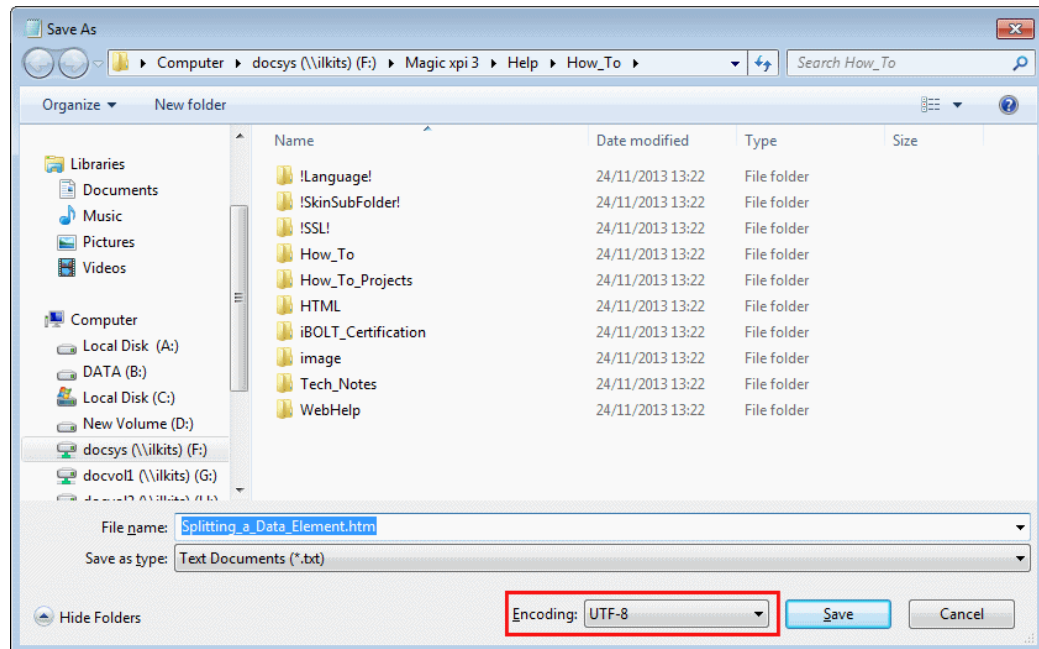
XML ファイルのエンコーディングを確認するには：

1. 関連する XML ファイルで右クリックしてください。[プログラムから開く] を選択し、**Notepad** を選択します。XML ファイルが開きます。
2. 下のイメージで示すように、ファイルのエンコーディングを参照するために、XML ヘッダをチェックしてください。この場合、エンコードタイプは、**UTF-8** です。

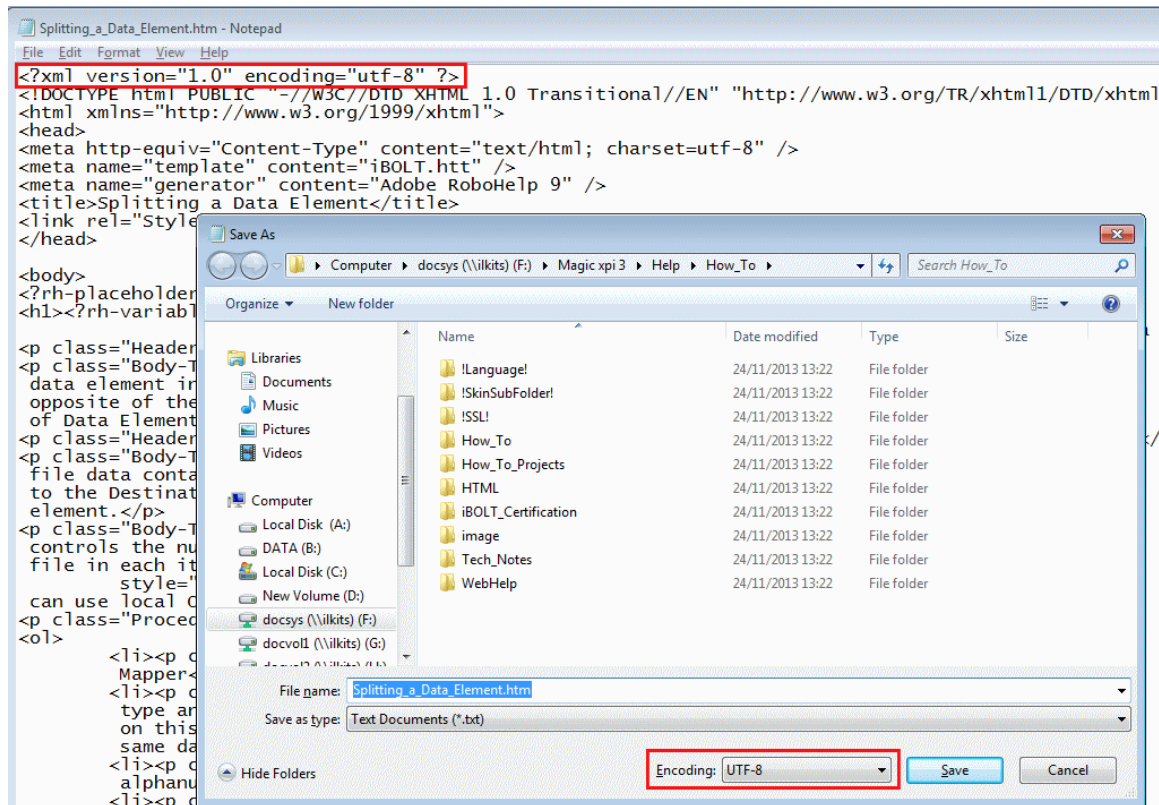


3. ヘッダのない XML ファイルを選択した場合、[ファイル] メニューをクリックして、[名前を付けて保存] を選択してください。

4. [名前を付けて保存] ダイアログボックスで、下のイメージで示するように [保存] ボタンのそばにある [エンコーディング] をクリックしてください。これは、**XML** ファイルで使用されるエンコードタイプを示しています。この場合、エンコードタイプは、**UTF-8** です。



注 XML ヘッダのエンコードタイプを、[名前を付けて保存]ダイアログボックスで指定されたエンコードタイプと比較してください。下のイメージで示すように、これらのタイプは、常に同じでなければなりません。

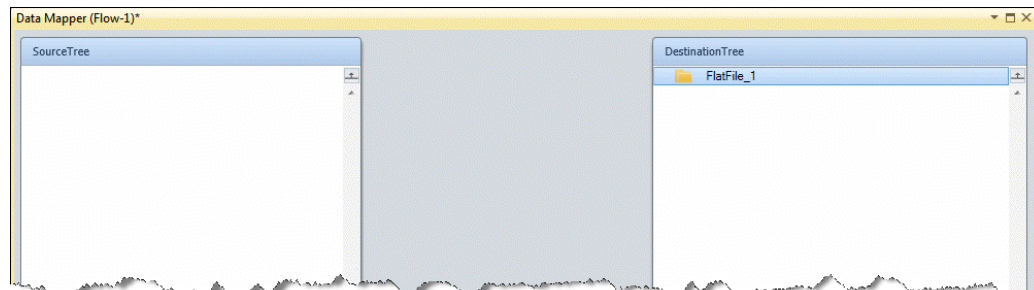


指定されたエンコードでファイルを作成するには

Magic xpi では、非常に簡単に指定したエンコードタイプでファイルを作成することができます。エンコードタイプとして選択することができる 2 つの異なるファイルタイプがあります。テキストと XML です。

指定したエンコードタイプでテキストファイルを作成するには：

1. フローにデータマップをドラッグしてください。
2. データマップをダブルクリックするか、右クリックし、コンテキストメニューから [構成] を選択してください。[データマップ] ウィンドウが開きます。
3. ツールボックスの [マップスキーマ] セクションから、フラットファイルタイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。



4. フラットファイルで右クリックし、[プロパティ] ペインを開くために、コンテキストメニューから [プロパティを表示] を選択してください。

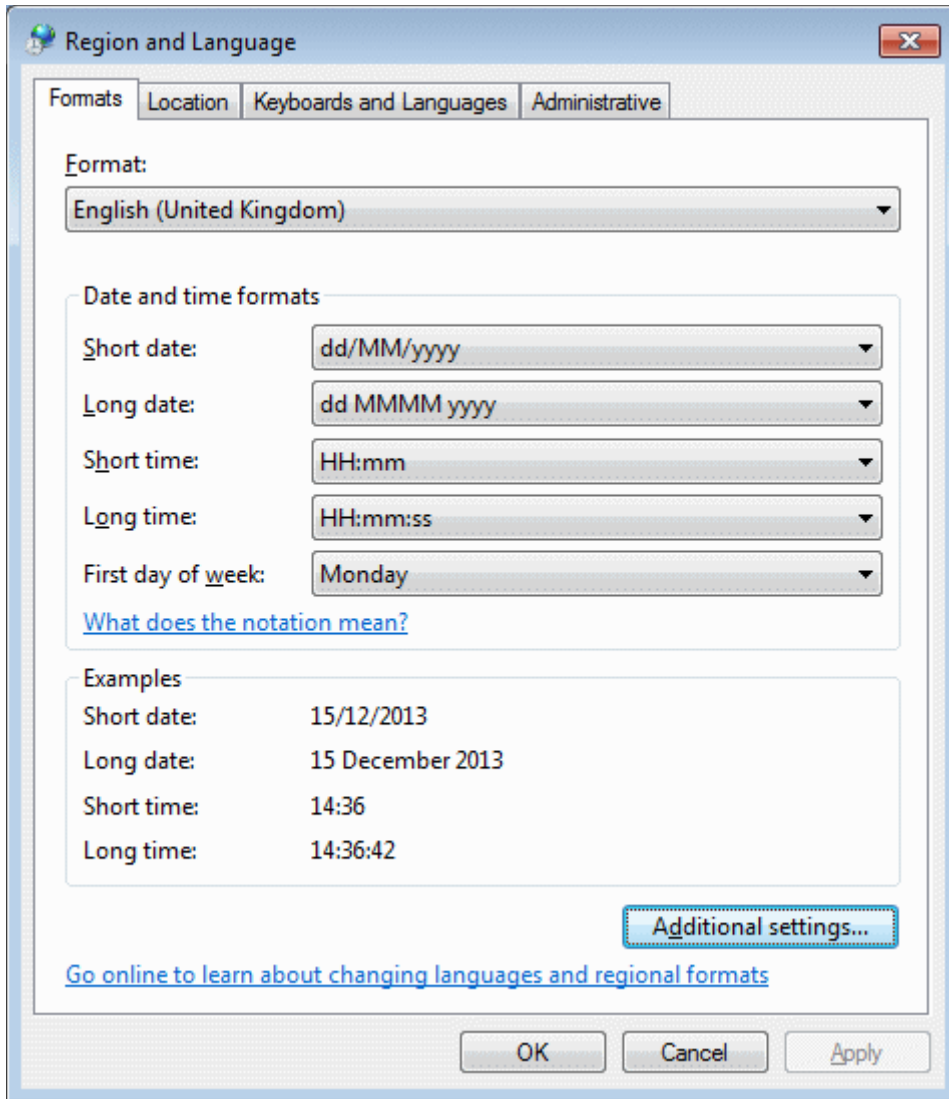
5. [プロパティ] ペインの [データ送り先エンコーディング] フィールドで、必要なエンコーディングタイプを選択してください。デフォルトは **ANSI** です。

The screenshot shows a 'Properties' dialog box for a 'Flat File Schema'. The 'General' tab is active, displaying a table of properties. The 'Data Destination Encoding' property is highlighted with a red box, showing 'ANSI' selected from a dropdown menu. Other properties include 'Destination Type', 'Variable', 'Append Data', 'Type', 'Columns Delimiter', 'Record Delimiter', 'Include Delimiter In Data', 'Create Header Line', 'Lines', 'Name', and 'Description'.

Destination Type	Variable
Data Destination Encoding	ANSI
Variable	C.UserBlob
Append Data	No
Type	Delimited
Columns Delimiter	,
Record Delimiter	
Include Delimiter In Data	No
Create Header Line	No
Lines	(Collection)
Name	FlatFile_1
Description	

6. [OK] をクリックして構成内容を保存して、開いていたダイアログボックスを閉じてください。

注 ドロップダウンリストから ANSI を選択すると、エンコードタイプは PC のコントロールパネルで定義されている [地域と言語] の設定に基づいて定義されます。

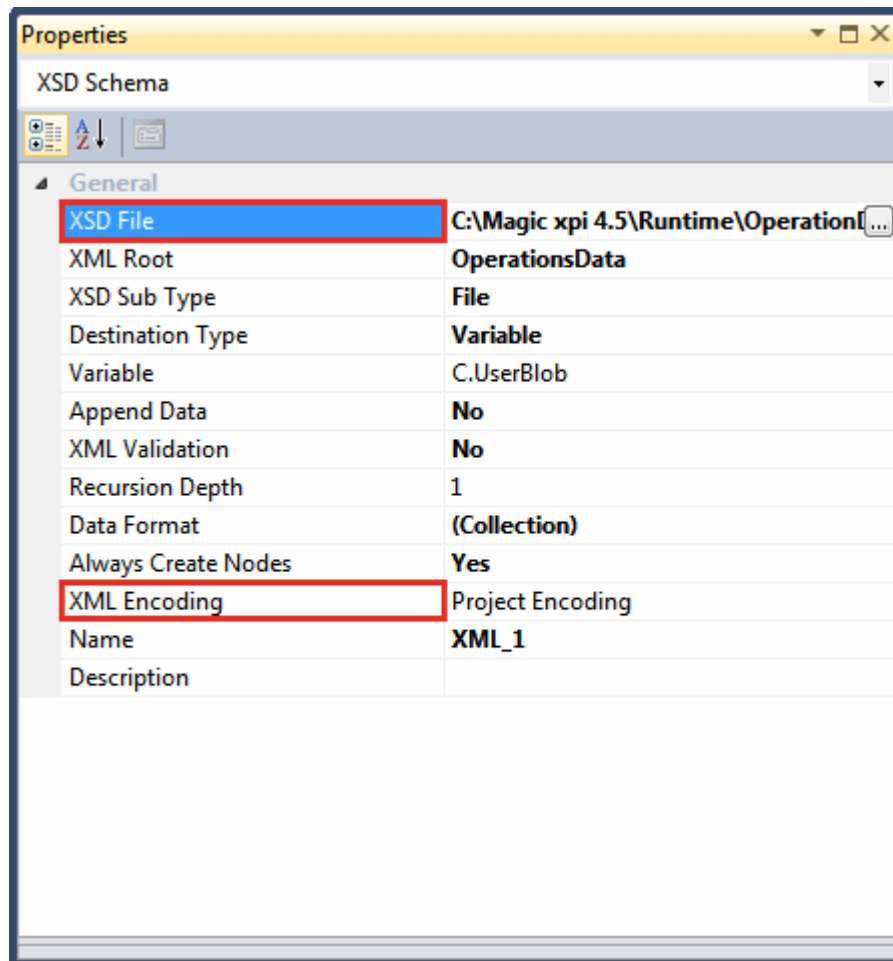


指定したエンコードタイプで XML ファイルを作成するには：

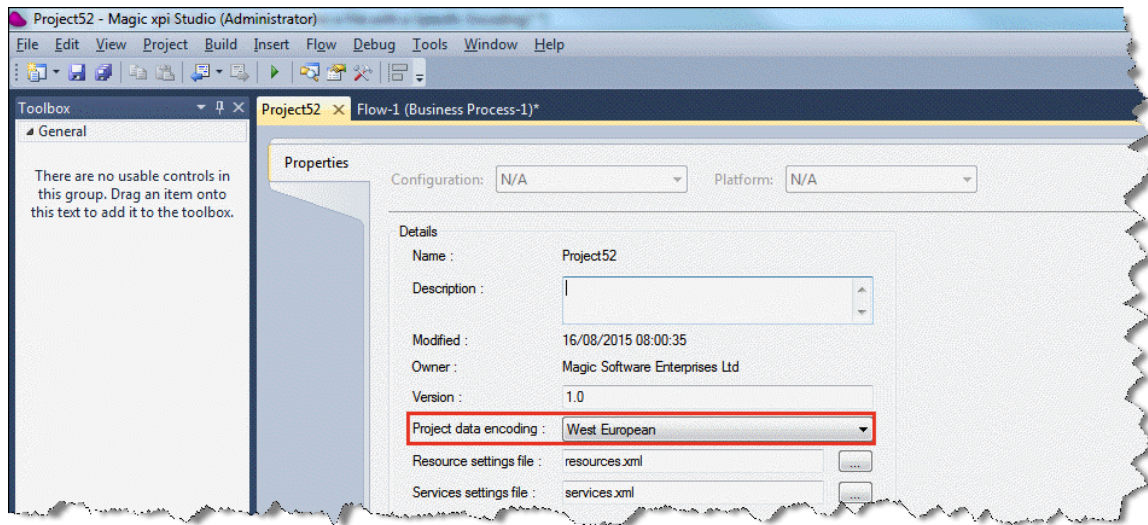
1. フローにデータマップをドラッグしてください。
2. データマップをダブルクリックするか、右クリックし、コンテキストメニューから [構成] を選択してください。 [データマップ] ウィンドウが開きます。
3. ツールボックスの [マップスキーマ] セクションから、XML ファイルタイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。 .



4. XML ファイルで右クリックし、[プロパティ] ペインを開くために、コンテキストメニューから [プロパティを表示] を選択してください。
5. [プロパティ] ペインの [XSD ファイル] フィールドで、必要な XSD ファイルを選択してください。
6. [プロパティ] ペインの [XML エンコーディング] フィールドで、必要なエンコーディングを選択してください。デフォルトは、プロジェクトエンコーディングです。



注意： ドロップダウンリストから**プロジェクトエンコーディング**を選択すると、エンコード値がプロジェクトレベルの設定から取得されます。プロジェクトレベルのこのエンコード値を変更するには、[プロジェクト]メニューを選択して、[プロジェクト<名前>プロパティ]をクリックして[プロジェクトプロパティ]ダイアログボックスを開きます。[詳細]タブの[プロジェクトデータエンコーディング]フィールドで必要なエンコーディングを選択します。





7. [保存] をクリックして構成を保存してください。

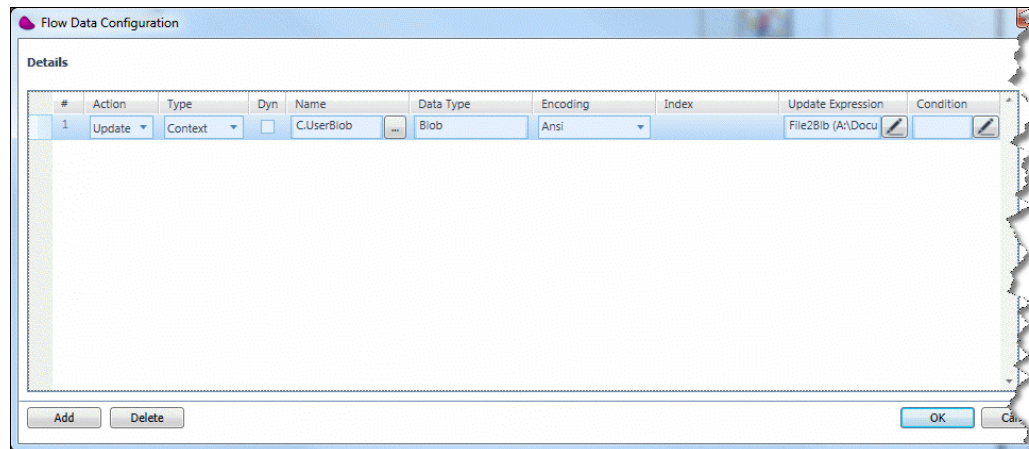
Unicode か ANSI、UTF-8 のファイルをメモリに読み込ませるには

Unicode、ANSI または UTF-8 ファイルをメモリに読み込ませるには、File2Blob 関数とファイル管理コンポーネントを使用して行われます。

File2Blob 関数を使用して内容をメモリに読み込ませるには：

1. フローにフローデータユーティリティをドラッグしてください。
2. [追加] をクリックして [フローデータ構成] ダイアログボックスを開きます。
3. [タイプ] カラムで、**コンテキスト**を選択してください。
4. [名前] カラムで、関連する **BLOB** を選択してください。
5. [エンコーディング] カラムで、ファイルに必要なエンコードタイプを選択してください。
6. [更新式] カラムで、 をクリックして式エディタを開いてください。
7. 式エディタで、**File2Blob** 関数を選択して続けて左括弧 "(" を入力します。
8.  をクリックして、メモリに読み込ませたい内容をクリックして、右括弧 ")" を続けて入力してください。

9. [OK] をクリックして [フローデータ構成] ダイアログボックスを閉じます。フローを実行すると、ファイルの内容がメモリに読み込まれます。



ファイル管理コンポーネントで内容をメモリに読み込ませるには：

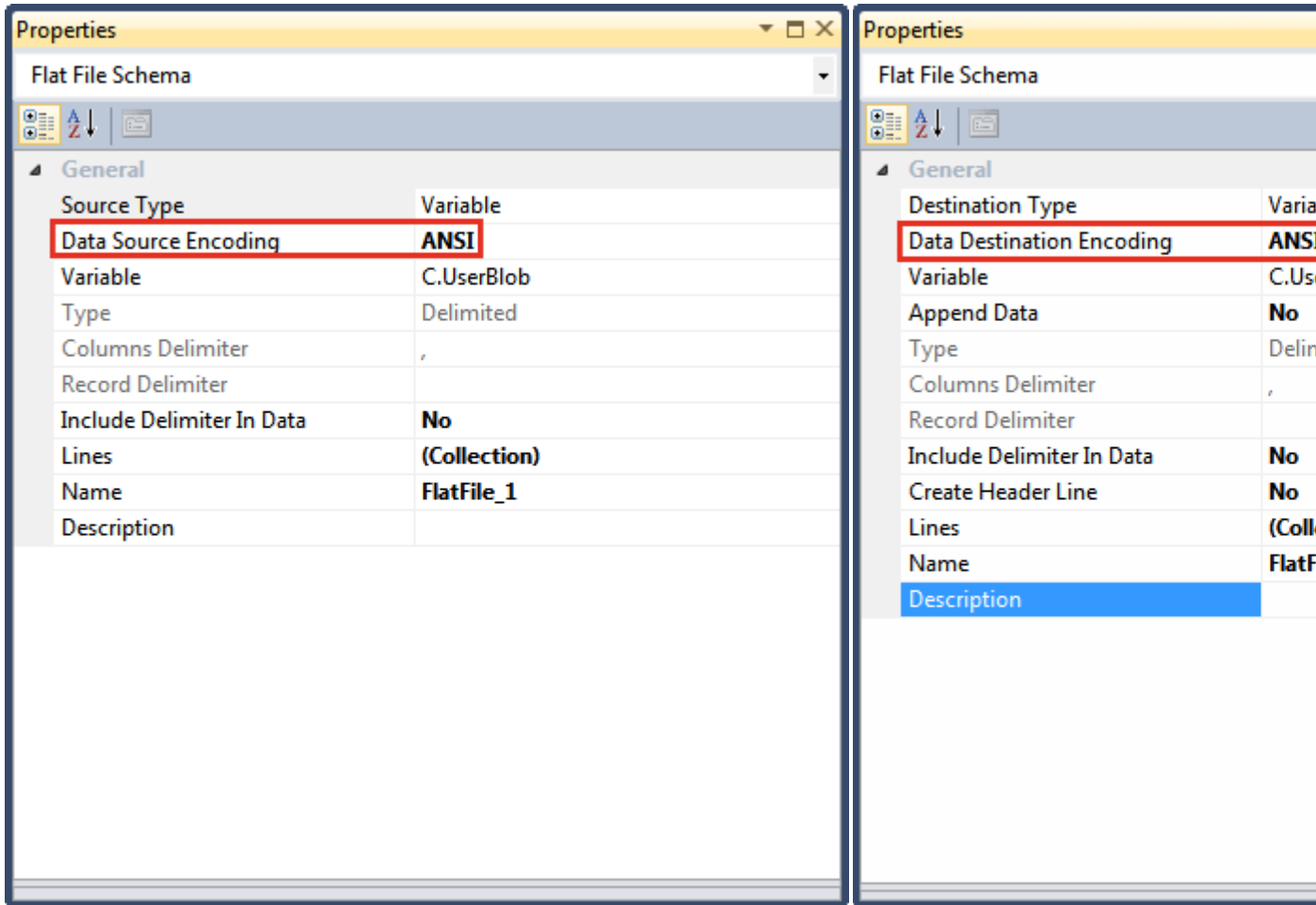
1. フローにファイル管理コンポーネントをドラッグしてください。
2. [メソッド] インターフェースを選択し、[設定] をクリックしてください。
3. [ダイレクトアクセスメソッド：ファイル管理] ダイアログボックスで、[新規] をクリックしてください。
4. ドロップダウンリストから **Read File** メソッドを選択してください。
5. [OK] をクリックして、[ダイレクトアクセスメソッド：ファイル管理] ダイアログボックスを閉じてください。フローを実行すると、ファイルはエンコードタイプに基づいてメモリに自動的に読み込まれます。

データマップを使用する場合、**Unicode** か **ANSI**、**UTF-8** のフラットファイルでレコードを処理するには

データマップは、フラットファイルを使用して動作するとき、異なるエンコーディングでレコードを処理する方法を定義します。

送り元と送り先レコードの処理：

1. フローにデータマップをドラッグしてください。
2. データマップをダブルクリックするか、右クリックし、コンテキストメニューから [設定] を選択してください。[データマップ] ウィンドウが開きます。
3. ツールボックスの [マップスキーマ] セクションから、フラットファイルタイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
4. フラットファイルで右クリックし、[プロパティ] ペインを開くために、コンテキストメニューから [プロパティを表示] を選択してください。
5. [プロパティ] ペインの [データ送り元エンコーディング] パラメータまたは [データ送り先エンコーディング] フィールドで、ドロップダウンリストから必要なエンコーディングタイプを選択してください。ここでは、**ANSI**、**Unicode**、または **UTF-8** から選択することができます。



送り元レコード例


送り先レコード例

Unicode か UTF-8 BOM を BLOB ファイルに追加するには

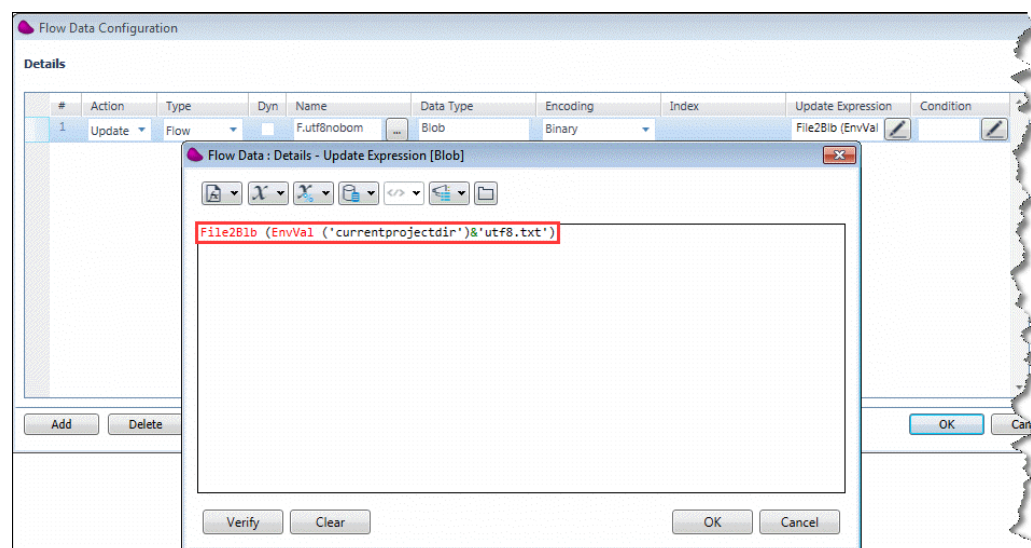
バイトオーダーマーク(BOM)は、テキストファイルのバイト順序を示すために使用されます。エンコーディングの検出を支援するために BOM を BLOB ファイルに追加したい場合があります。BlobAddBOM 関数を使用することで Unicode または UTF-8 BOM を BLOB ファイルに簡単に追加することができます。


BlobAddBOM 関数を使用して、Unicode または UTF-8 BOM を BLOB ファイルに追加するには：

1. この例の目的のために、サンプルの BLOB ファイルを作成することから開始してください。<Magic xpi のインストールフォルダ>\Magic xpa\projects<現在のプロジェクト>フォルダに保存してください。このファイルを **utf8.txt** と呼びます。
2. 次に、この例で使用する変数を作成してください。変数リポジトリを開いてください。
3. ここで、**F.utf8nobom** と呼ばれる BLOB 変数と **F.logic** と呼ばれる論理変数を作成してください。
4. フローにフローデータユーティリティをドラッグしてください。

5. BOM を追加する処理を開始するために、選択されたファイルを **BLOB** に変換する必要があります。[フローデータ構成] 設定ダイアログボックスで、[追加] をクリックしてください。
6. [アクション] カラムで、ドロップダウンリストからの **Update** を選択します。
7. [タイプ] カラムで、**Flow** を選択します。
8. [名前] カラムで、以前作成した **F.utf8nobom** という BLOB 変数を選択します。
9. [エンコード] カラムで、**Binary** を選択します。
10. [更新式] カラムで、 をクリックして式エディタを開いてください。
11. 式エディタでは、以下の式 (以下のように)を入力し、[OK] をクリックしてください。

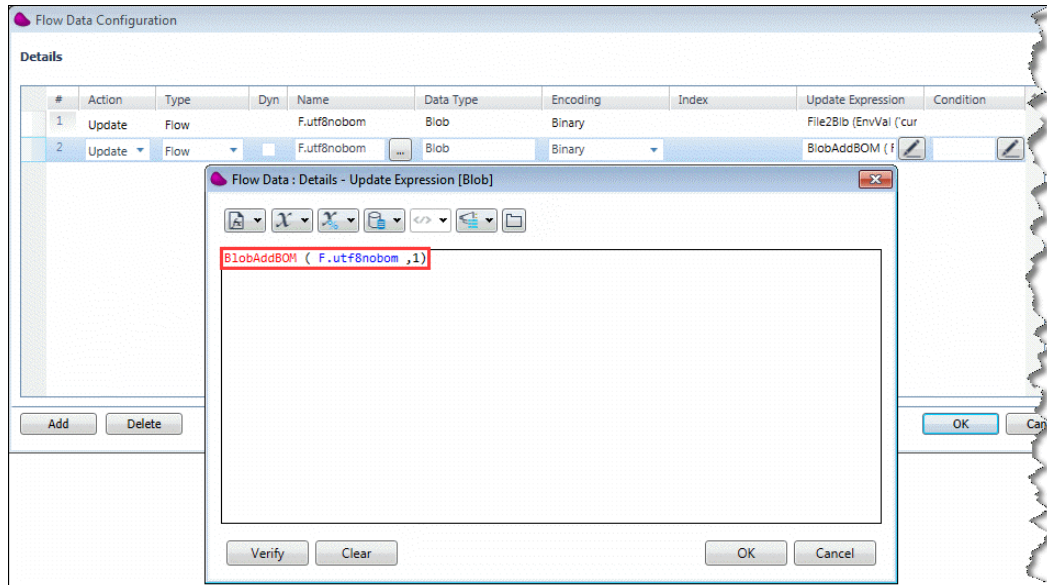
File2B1b (EnvVal ('currentprojectdir')&'utf8.txt')




12. 次のステージは、BOM を新しく作成された **BLOB** に追加することです。これは、**BlobAddBOM** 関数を使用することで可能です。[フローデータ構成] ダイアログボックスで、[追加] をクリックしてください。
13. 手順#6～9（上記の）で入力した値と同じパラメータを入力してください。
14. [更新式] カラムで、 をクリックして式エディタを開いてください。
15. 式エディタでは、以下の式 (以下のように)を入力し、[OK] をクリックしてください。

BlobAddBOM (F.utf8nobom ,1)

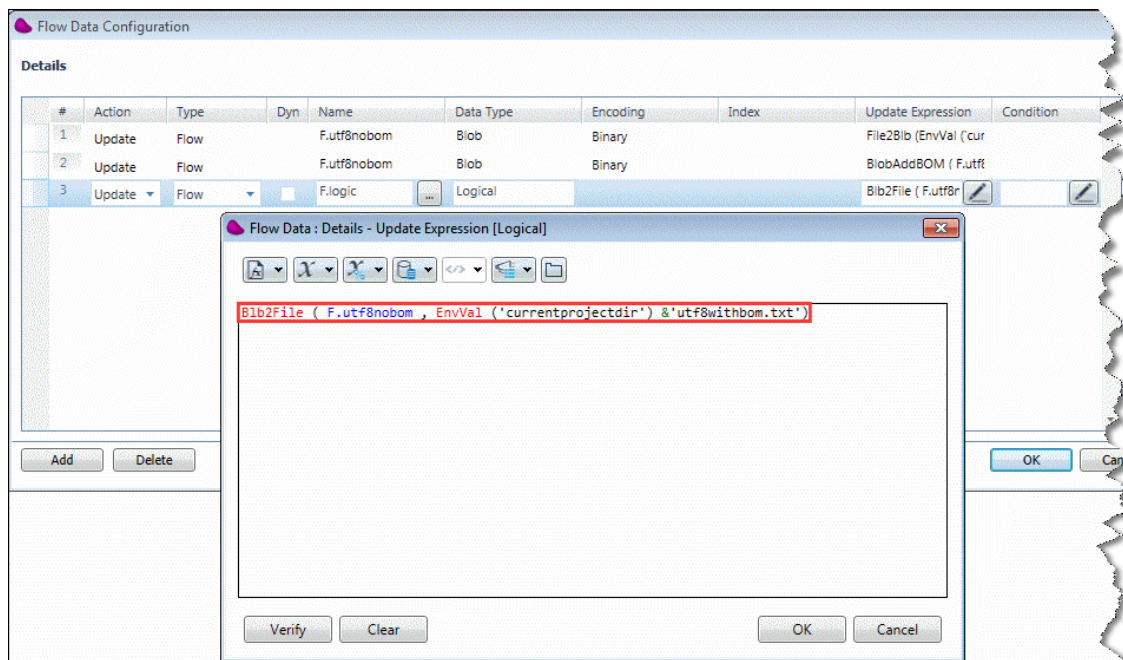
注 意: UTF-16 BOM を追加するには、次の式を使用してください: **BlobAddBOM (F.utf8nobom ,2).**



16. 次に、新しく **BOM** が追加された **BLOB** を最初のファイル形式に戻す必要があります。これを行うには、**Blob2File** 関数を使用します。[フローデータ構成] ダイアログボックスで、[追加] をクリックしてください。
17. [アクション] カラムで、ドロップダウンリストからの **Update** を選択します。
18. [タイプ] カラムで、**Flow** を選択します。
19. [名前] カラムで、以前作成した **F.logic** という論理型変数を選択します。
20. [更新式] カラムで、 をクリックして式エディタを開いてください。

21. 式エディタでは、以下の式（以下のように）を入力し、[OK] をクリックしてください。

Blob2File (F.utf8nobom , EnvVal('currentprojectdir') &'utf8withbom.txt')




22. [OK] をクリックしてフローデータユーティリティを閉じてください。
23. プロジェクトをビルドしてください。
24. これで、プロジェクトのデバッグや実行を行うと、<Magic xpi のインストールフォルダ>\Magic\projects\<現在のプロジェクト>フォルダ内に **utf8withbom.txt** と呼ばれるファイルが作成されます。このファイルは、BOM が追加されたファイルの元の BLOB ファイルです。ファイルの名前は、手順#20 で入力した式から取得されます。バイナリエディタを使用することで、この BOM を参照することができます。

UTF-8 の内容を含むバイナリ BLOB を Unicode BLOB に変換するには

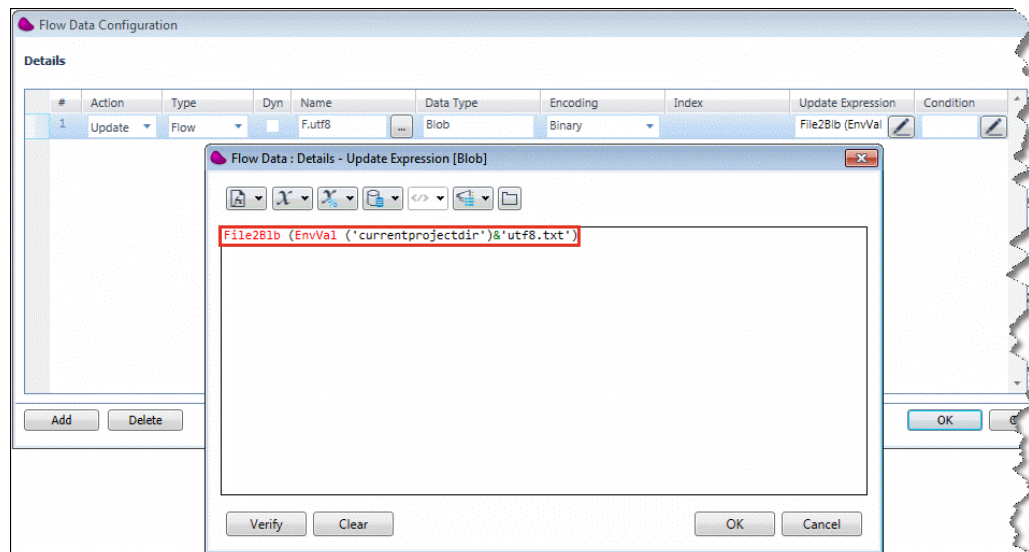
UTF-8 の内容を含むバイナリ BLOB を Unicode BLOB に変換するには、UTF8toUnicode 関数を使用する必要があります。

UTF-8 の内容を含むバイナリ BLOB を UTF8toUnicode 関数を使用して Unicode BLOB に変換するには：


1. この例では、バイナリの BLOB ファイルを作成することから始めます。<My Documents>\Magic\projects\<現在のプロジェクト>フォルダ内にこのファイルを UTF-8 エンコーディングで保存してください。このファイルを **utf8.txt** と呼びます。
2. この例で使用する変数を作成するために、フロー変数リポジトリを開いてください。
3. ここで、**F.utf8** と **F.utf16** と呼ばれる 2 つの BLOB 変数を作成してください。そして、**F.log** と呼ばれる論理変数を作成してください。
4. 次に、この例で使用する変数を作成してください。フロー変数リポジトリを開き、**F.utf8** と **F.utf16** と呼ばれる 2 つの BLOB 変数を作成してください。そして、同じタブで **F.log** と呼ばれる論理変数を作成してください。
5. Drag a Flow Data u フローデータユーティリティをフローにドラッグしてください。

6. 変換処理を開始するために、**File2B1b** 関数を使用して、ファイルシステムからバイナリの **BLOB** ファイルを読み込む必要があります。[フローデータ構成] ダイアログボックスで、[追加] をクリックしてください。
7. [アクション] カラムで、ドロップダウンリストから **Update** を選択してください。
8. [タイプ] カラムで、**Flow** を選択してください。
9. [名前] カラムで、以前に作成した **F.utf8 BLOB** 変数を選択してください。
10. [エンコーディング] カラムで、**UTF-8** をそのまま表示することがないため、**Binary** を選択してください。
11. [更新式] カラムで、 をクリックして式エディタを開きます。
12. 式エディタで、以下の式(以下のイメージに示するような)を入力してください、次に [OK] をクリックしてください。

File2B1b (EnvVal ('currentprojectdir')&'utf8.txt')

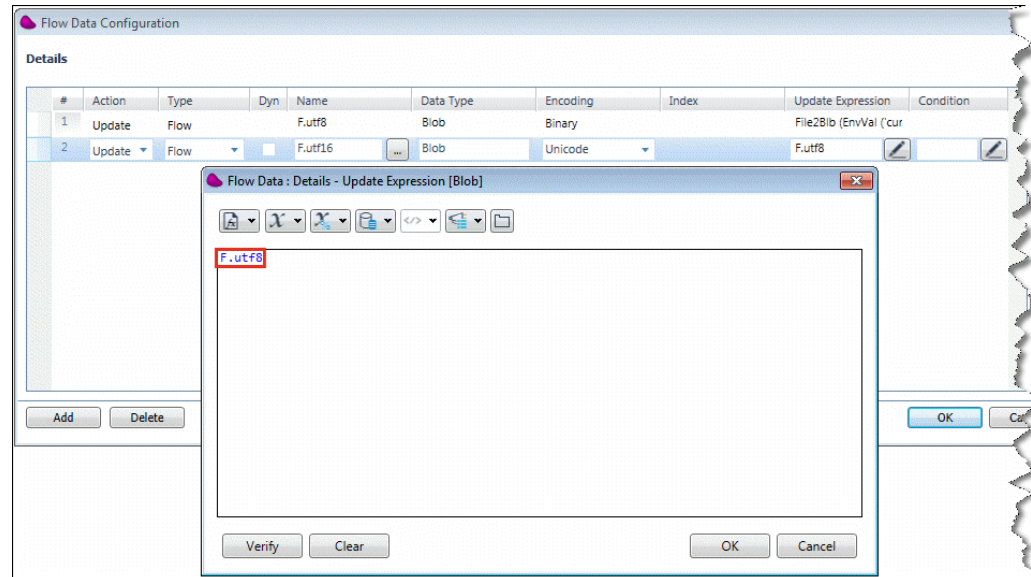


実例として、UTF8toUnicode 関数を使用しない簡単な方法で **utf8.txt** ファイルを Unicode に変換しようとする場合、どうなるかを見てみましょう。この場合、UTF-8 がサポートされないためにファイルが破損することになります。


1. [フローデータ構成] ダイアログボックスで、[追加] をクリックしてください。
2. [アクション] カラムで、ドロップダウンリストから **Update** を選択してください。
3. [タイプ] カラムで、**Flow** を選択してください。
4. [名前] カラムで、以前に作成した **F.utf16 BLOB** 変数を選択してください。
5. [エンコーディング] カラムで、**Unicode** を選択してください。
6. [更新式] カラムで、 をクリックして式エディタを開きます。

7. **F.utf8** BLOB 変数の内容によって **F.utf16** BLOB 変数を更新するために、以下の式を式エディタ(以下のイメージで示すように)に入力してください。次に **[OK]** をクリックしてください。

F.utf8

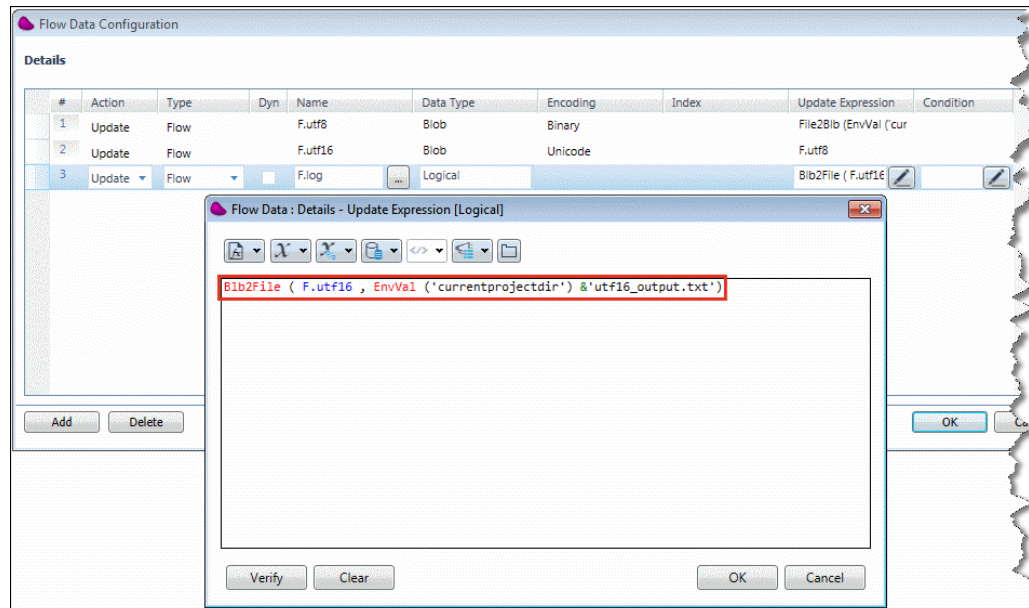


次のステージは、**F.utf16** BLOB 変数の内容をファイルシステム(**utf16_output** という名前のファイル)に保存することです。**Blb2File** 関数でこれを行います。

1. [フローデータ構成] ダイアログボックスで、[追加] をクリックしてください。
2. [アクション] カラムで、ドロップダウンリストから **Update** を選択してください。
3. [タイプ] カラムで、**Flow** を選択してください。
4. [名前] カラムで、以前に作成した **F.log** 論理変数を選択してください。
5. [更新式] カラムで、 をクリックして式エディタを開きます。

- 式エディタで、以下の式(以下のイメージに示するような)を入力してください、次に [OK] をクリックしてください。

B1b2F11e (F.utf16 , EnvVal('currentprojectdir') &'utf16_output.txt')




次に、ステップ 6(上記)で定義した **utf16_output.txt** ファイルで何が作成されるかを確認するため、プロジェクトを実行してください。

- [フローデータ構成] ダイアログボックスを閉じるために、[OK] をクリックしてください。
- プロジェクトをビルドしてください。
- Magic xpi モニタを開き、プロジェクトを開始します。
- <My Documents>\Magic\projects\<現在のプロジェクト>フォルダを開いてください。
utf16_output.txt ファイルが作成されていることを確認できます。

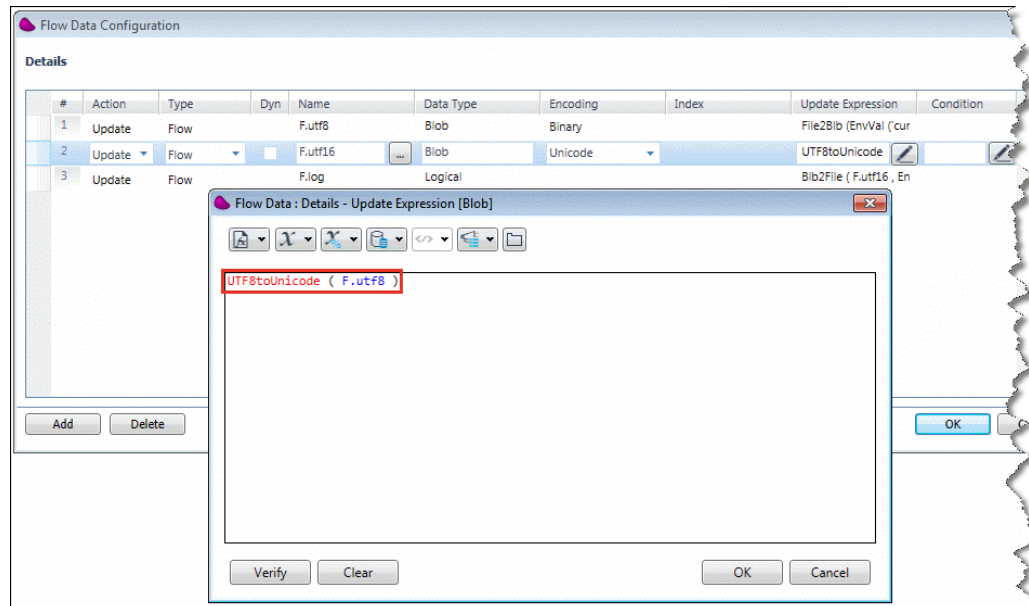
utf16_output.txt ファイルを開くと、その内容が破損していることに注意してください。


さて、UTF8toUnicode 関数を使用して、UTF-8 ファイルを Unicode に変換するために以下の変更をする必要があります。これによって、正しくエンコードされた内容を受け取ることができます。

- [フローデータ構成] ダイアログボックスの 2 番目の行に移動し、[削除] をクリックしてください。
- 新規行を作成するために、[追加] をクリックしてください。
- [アクション] カラムで、ドロップダウンリストから **Update** を選択してください。
- [タイプ] カラムで、**Flow** を選択してください。
- [名前] カラムで、以前に作成した **F.utf16 BLOB** 変数を選択してください。
- [エンコーディング] カラムで、**Unicode** を選択してください。
- [更新式] カラムで、 をクリックして式エディタを開きます。

8. 式エディタで、以下の式(以下のイメージに示するような)を入力してください、次に [OK] をクリックしてください。

UTF8toUnicode (F.utf8)



9. テーブルの 2 番目の行までこの新規エントリを移動させるために、[フローデータ構成] ダイアログボックスの右側の上カーソルキーを使用してください。
10. [フローデータ構成] ダイアログボックスの 3 番目のエントリで [更新式] カラムに移動して  をクリックして式エディタを開きます。
11. 式を以下のように変更してください。

Bib2File (F.utf16 , EnvVal('currentprojectdir') &'utf16_output_valid.txt')

さて、ステップ 11(上記)で定義した `utf16_output_valid.txt` ファイルがどのように作成されるかを確認するために、再びプロジェクトを実行してください。

1. [フローデータ構成] ダイアログボックスを閉じるために、[OK] をクリックしてください。
2. プロジェクトをビルドしてください。
3. Magic xpi モニタを開き、プロジェクトを開始します。
4. `<My Documents>\Magic\projects<現在のプロジェクト>` フォルダを開いてください。
`utf16_output.txt` ファイルが作成されていることを確認できます。

`utf16_output_valid.txt` ファイルを開くと、その内容が正しくエンコードされることが確認できるはずです。これは、UTF-8 の内容を Unicode に変換するために、UTF8toUnicode 関数を使用したためです。

関連トピック：

[UTF-8 の内容を含む Unicode BLOB をバイナリ BLOB に変換するには](#)


UTF-8 の内容を含む Unicode BLOB をバイナリ BLOB に変換するには

Magic xpi は、簡単に Unicode BLOB を UTF-8 の内容を含んだバイナリ BLOB に変換することができます。UTF8fromUnicode 関数を使用してこれを行うことができます。

このトピックは、[UTF-8 の内容を含むバイナリ BLOB を Unicode BLOB に変換するには](#)で作成された utf16_output_valid.txt Unicode BLOB を使用して説明します。このトピックのために新しいプロジェクトを作成すると、新しいプロジェクトの<My Documents>%Magic%projects% フォルダに utf16_output_valid.txt ファイルをコピーしてください。

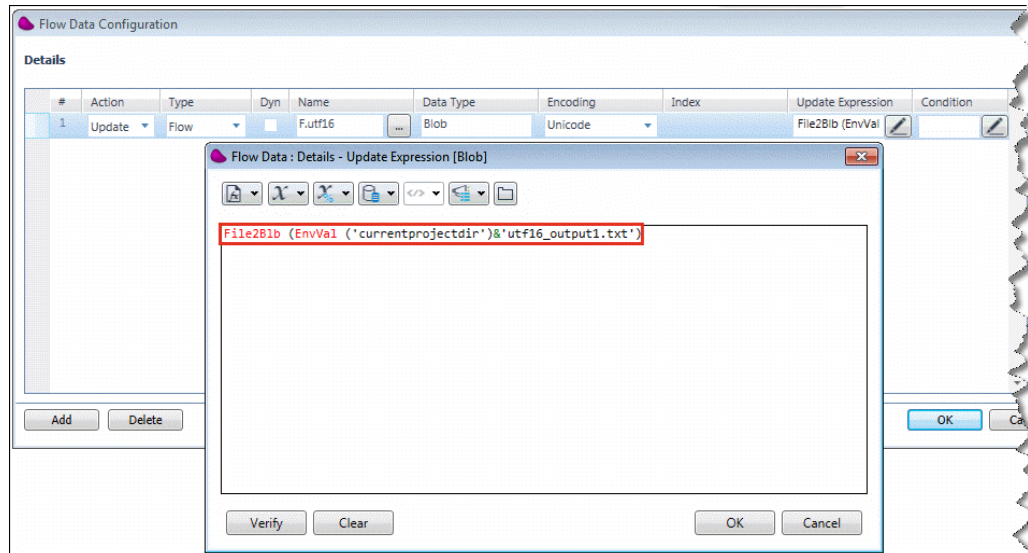
[UTF-8 の内容を含むバイナリ BLOB を Unicode BLOB に変換するには](#)トピックのプロジェクトが動作しない場合は、このサンプルで動作するような独自の Unicode BLOB ファイルを作成する必要があります。そしてこのファイルを <My Documents>%Magic%projects%<現在のプロジェクト> フォルダに保存してください。このファイルを utf16_output1.txt と呼びます。

UTF8fromUnicode 関数を使用して Unicode BLOB を UTF-8 の内容を持つバイナリ BLOB に変換するには：

1. この例で使用する変数を作成するために、フロー変数リポジトリを開いてください。
2. そこで、**F.utf8** と **F.utf16** と呼ばれる 2 つの BLOB 変数を作成してください。また、**F.log** と呼ばれる論理変数を作成してください。
3. フローデータユーティリティをフローにドラッグしてください。
4. 変換処理を開始するために、File2Bib 関数を使用して、ファイルに Unicode BLOB を読み込む必要があります。[フローデータ構成] ダイアログボックスで、[追加] をクリックしてください。
5. [アクション] カラムで、ドロップダウンリストから **Update** を選択してください。
6. [タイプ] カラムで、**Flow** を選択してください。
7. [名前] カラムで、以前に作成した **F.utf16** BLOB 変数を選択してください。
8. [エンコーディング] カラムで、**Unicode** を選択してください。
9. [更新式] カラムで、 をクリックして式エディタを開きます。

10. 式エディタで、以下の式(以下のイメージに示するような)を入力してください、次に [OK] をクリックしてください。

File2B1b (EnvVal ('currentprojectdir')&'utf16_output1.txt')

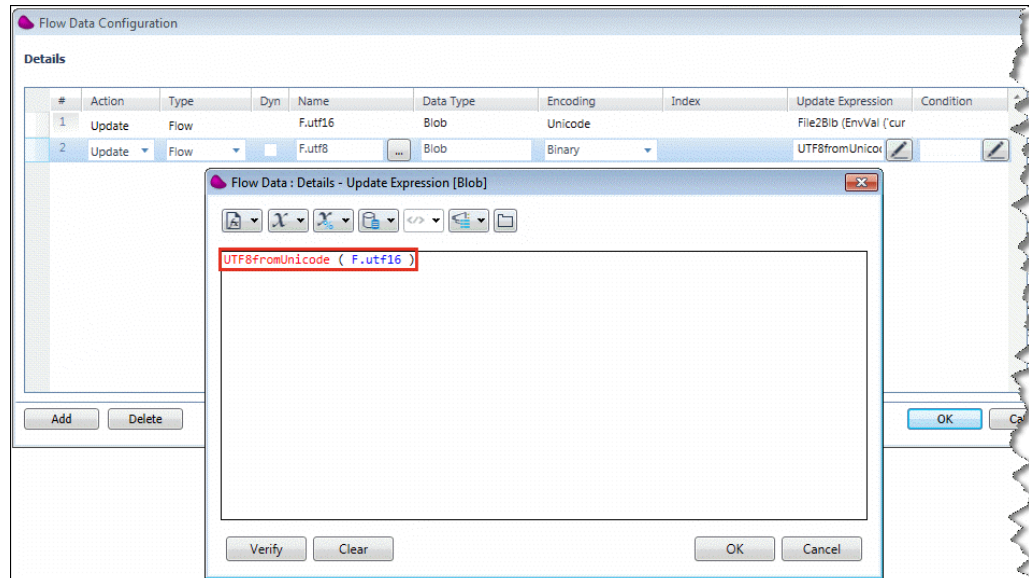


次のステージは、UTF8fromUnicode 関数を使用して、**utf16_output1.txt** ファイルを UTF-8 BLOB 変数に変換することです。


1. [フローデータ構成] ダイアログボックスで、[追加] をクリックしてください。
2. [アクション] カラムで、ドロップダウンリストから **Update** を選択してください。
3. [タイプ] カラムで、**Flow** を選択してください。
4. [名前] カラムで、以前に作成した **F.utf8 BLOB** 変数を選択してください。
5. [エンコーディング] カラムで、**Binary** を選択してください。
6. [更新式] カラムで、 をクリックして式エディタを開きます。

7. 式エディタで、以下の式(以下のイメージに示するような)を入力してください、次に [OK] をクリックしてください。

UTF8fromUnicode (F.utf16)

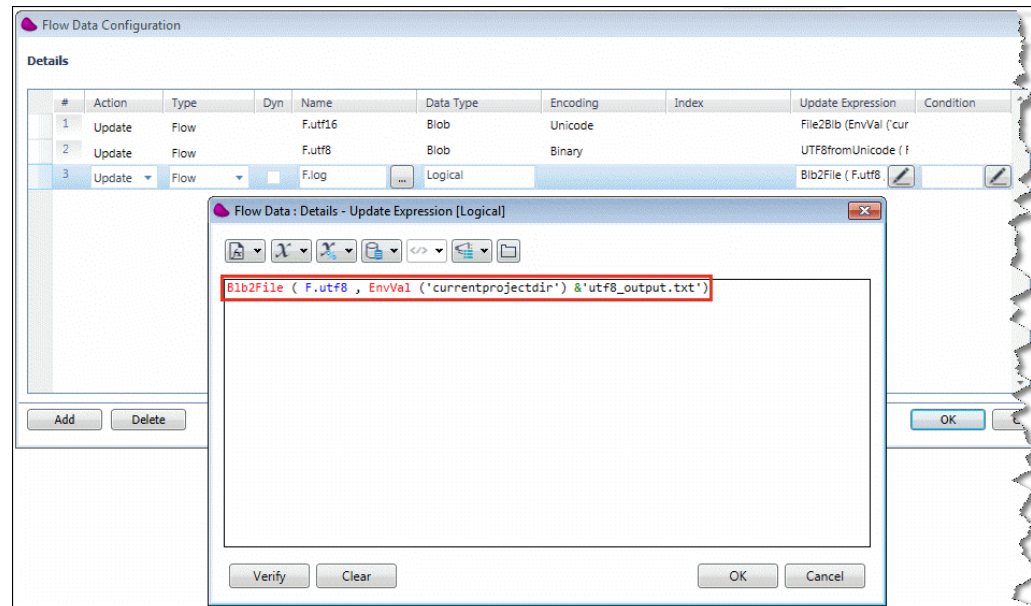


次に、ファイルシステムに UTF-8 BLOB 変数を書き込む j 必要があります。

1. [フローデータ構成] ダイアログボックスで、[追加] をクリックしてください。
2. [アクション] カラムで、ドロップダウンリストから **Update** を選択してください。
3. [タイプ] カラムで、**Flow** を選択してください。
4. [名前] カラムで、以前に作成した **F.log** 論理変数を選択してください。
5. [エンコーディング] カラムで、**Binary** を選択してください。
6. [更新式] カラムで、 をクリックして式エディタを開きます。

- 式エディタで、以下の式(以下のイメージに示するような)を入力してください、次に [OK] をクリックしてください。

Bib2File (F.utf8 , EnvVal('currentprojectdir') &'utf8_output.txt')




次に、ステップ 6(上記)で定義した **utf8_output.txt** ファイルに何が作成されるかを確認するために、プロジェクトを実行してください。

- [フローデータ構成] ダイアログボックスを閉じるために、[OK] をクリックしてください。
- プロジェクトをビルドしてください。
- Magic xpi モニタを開き、プロジェクトを開始します。
- <My Documents>\Magic\projects<現在のプロジェクト>フォルダを開いてください。
utf8_output.txt ファイルが作成されていることを確認できます。

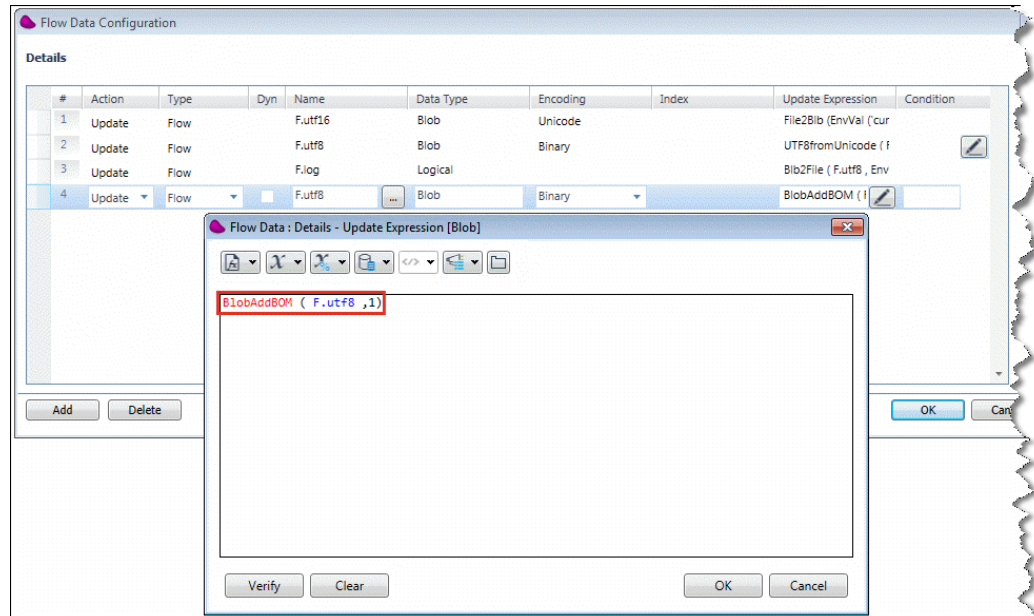
utf8_output.txt ファイルを開くと、その内容は UTF-8 で正しくエンコードされます。バイナリエディタを使用して内容をチェックすると、ファイルに BOM が含まれていないことが確認できます。

次のステージは、BOM を BLOB 変数に追加する方法についてを説明します。

- [フローデータ構成] ダイアログボックスで、[追加] をクリックしてください。
- [アクション] カラムで、ドロップダウンリストから **Update** を選択してください。
- [タイプ] カラムで、**Flow** を選択してください。
- [名前] カラムで、以前に作成した **F.utf8** BLOB 変数を選択してください。
- [エンコーディング] カラムで、**Binary** を選択してください。
- [更新式] カラムで、 をクリックして式エディタを開きます。

- 式エディタで、以下の式(以下のイメージに示するような)を入力してください、次に [OK] をクリックしてください。

BlobAddBOM (F.utf8 ,1)



- テーブルの 3 行目までにこの新規エントリを移動させるために、[フローデータ構成] ダイアログボックスの右側の上カーソルキーを使用してください。
- 既存の utf8_output.txt ファイルを<My Documents>\Magic\projects<現在のプロジェクト>フォルダから削除してください。

さて、utf8_output.txt ファイルで何が作成されるかを確認するために、再びプロジェクトを実行してください。

- [フローデータ構成] ダイアログボックスを閉じるために、[OK] をクリックしてください。
- プロジェクトをビルドしてください。
- Magic xpi モニタを開き、プロジェクトを開始します。
- <My Documents>\Magic\projects<現在のプロジェクト>フォルダを開いてください。utf8_output.txt ファイルが作成されていることを確認できます。

utf8_output.txt ファイルを開くと、その内容が UTF-8 で正しくエンコードされることが確認できるはずです。それだけでなく、バイナリエディタを使用して、内容をチェックするとファイルに BOM が含まれていることが確認できます。

関連トピック：

[UTF-8 の内容を含むバイナリ BLOB を Unicode BLOB に変換するには](#)

エラー処理

エラー処理

以下のトピックは、Magic xpi の中のエラー処理について扱います。

リカバリ

[クラッシュ後にワークプロセスを再起動するには](#)
[事前に定義された後にワークプロセスの実行を停止するには](#)
[クラッシュから回復する時にリソースを消去することができるか](#)
[フローにタイムアウトを設定するには](#)

フローのエラー処理

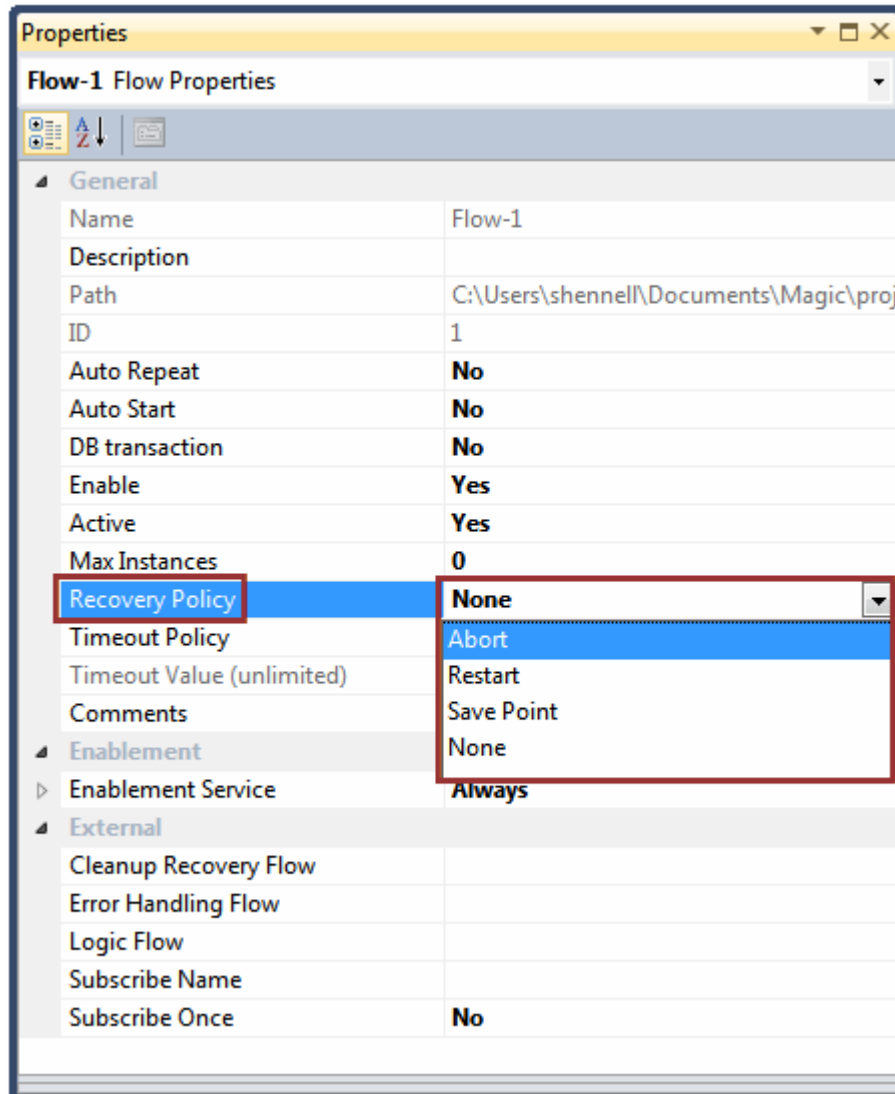
[エラーが発生したときに、フローを中断するには](#)
[エラーを受け取った時に、E メールを受け取るようにするには](#)
[エラーフローでエラーについての追加情報を取得するには](#)
[エラーが発生したら、ステップのリトライ回数をコントロールするには](#)
[特定のエラーが発生した後に、フローを異なる場所に移動するには](#)
[どのフローがエラーを呼び出したかを識別するには](#)
[どのステップがエラーを呼び出したかを識別するには](#)
[どのトリガがエラーを呼び出したかを識別するには](#)
[どの最終ステップ ID がエラーを呼び出したかを識別するには](#)
[どの最終フローID がエラーを呼び出したかを識別するには](#)
[エラー処理フローからのフローにエラーを返すには](#)

データマッピングエラー処理

[データベースエラーが発生した場合に、データマッピングを終了するには](#)
[例外が発生したデータベースのレコードフィールドを取り出すには](#)
[フローにタイムアウトを設定するには](#)

クラッシュ後にワークプロセスを再起動するには

プロジェクト開発フェーズの間、Magic xpi はエラーがプロジェクトのフローで発生した場合に、動作内容を定義することができます。これはリカバリポリシーとして知られて、フローの[プロパティ]ペイン(以下のイメージに示しています)の[リカバリポリシー]プロパティで定義されます。



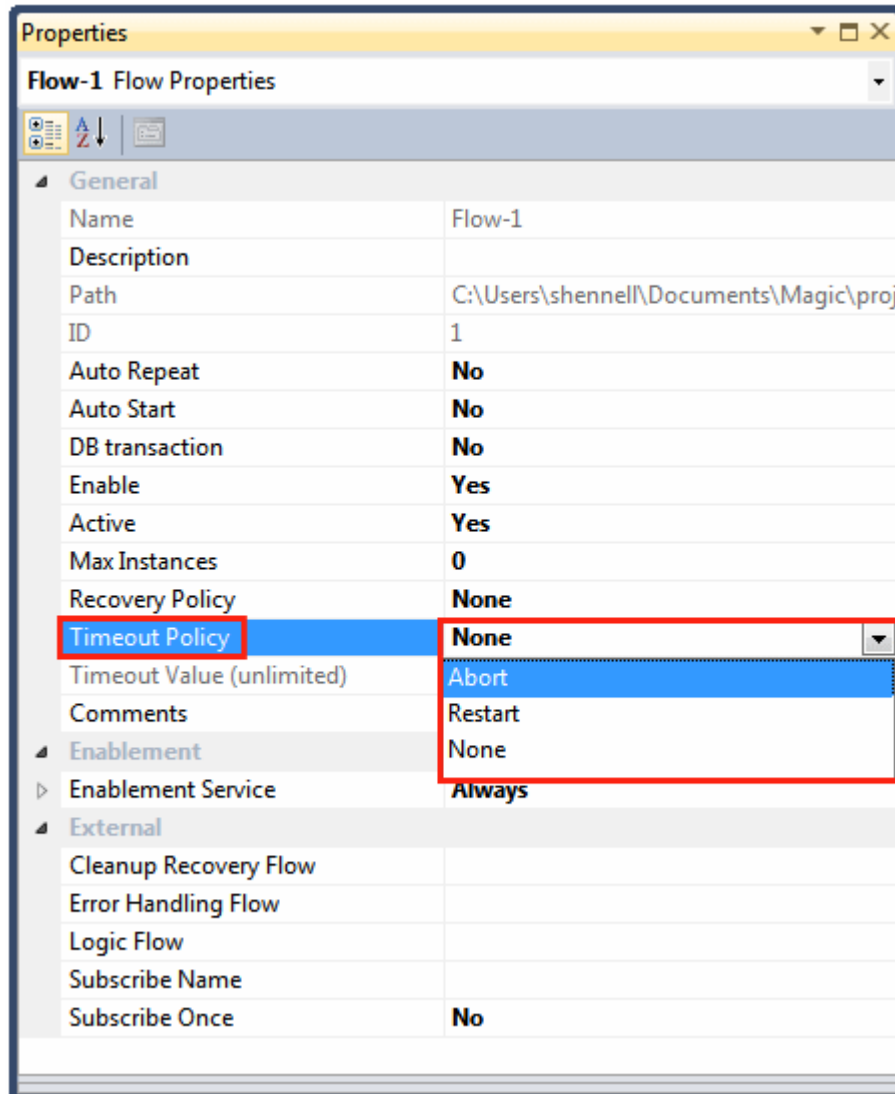
クラッシュの後にワークプロセスを再起動するするには、[リカバリポリシー]プロパティから**再起動**を選択してください。

クラッシュの後にワークプロセスを停止するには、[リカバリポリシー]プロパティから**アボート**を選択してください。

事前に定義された後にワークプロセスの実行を停止するには

ワークプロセスは、ルートフロー全体(その子フローを含みます)の実行ツリーです。ワークプロセスは全体として1つの企業取引として扱われます。

フローの[プロパティ]ペイン(以下のイメージで示しています)の[タイムアウトポリシー]プロパティは、Magic xpi が終了する前に、フローがどれくらい長く実行することができるかを定義することを可能にします。



これは、IBM i とコマンドラインなどの外部システムと接続する場合にフローが無制限に応答を待つことを防止するために、フローにタイムアウトを設定することが推奨されます。

タイムアウトをフローに設定する時、タイムアウトに達したら何を行うかを定義することができます。[タイムアウトポリシー]プロパティのオプションは以下の通りです。

- アボート
- 再起動
- なし

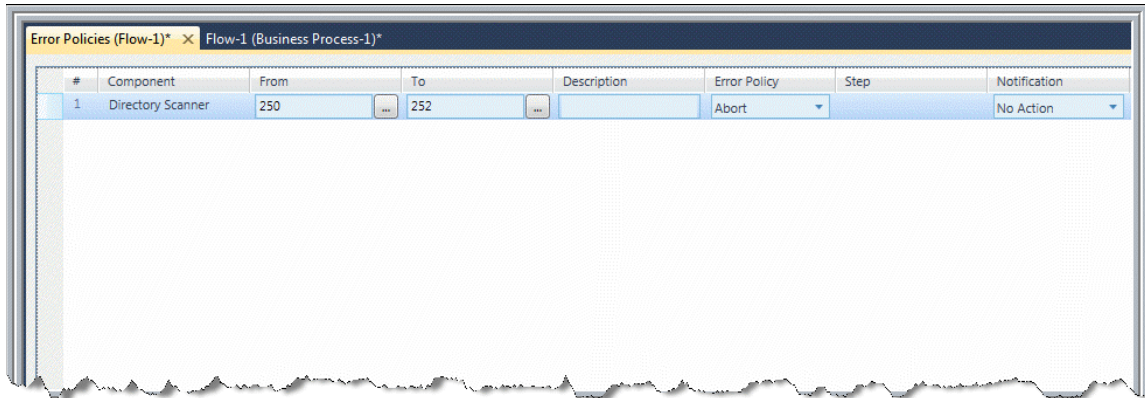
タイムアウトをフローに設定する時、最初に、フローの実行にどれだけの時間がかかるかをチェックする必要があります。フロータイムアウトがルートフロー(他のフローから呼ばれないフロー)にだけ作用することに注意してください。これは、呼び出すフローと呼び出されるフローでタイムアウトが定義されると、呼び出すフローのタイムアウトだけが実施されることを意味しています。

フロータイムアウトは、ユーザ定義の猶予期間の後にスレッドを停止させます。HTTPトリガまたは Web サービストリガは、フロータイムアウトを問わずあらかじめ決められたタイムアウトの後にタイムアウトメッセージを返します。フローがタイムアウトになり、再起動し、その時だけ応答を返すようにすることができます。

フローは、新しいスレッドで再起動します。

エラーが発生したときに、フローを中断するには

Magic xpi は、特定のエラーコードのまたはエラーコードの範囲でエラーポリシーを定義することができます。エラーが発生すると、これらのエラーポリシーは何を行うかを Magic xpi に指示します。Magic xpi に、フロー中止するように指示することができます。これは、[エラーポリシー]タブ(以下のイメージで示しています)を通して行います。



エラーが発生した時にフローを中断するには：

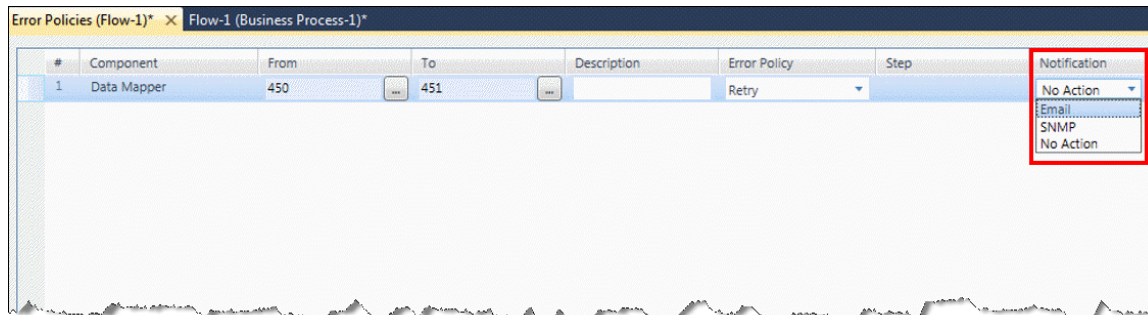
1. ソリューションエクスプローラの関連したフロー下で、エラーポリシーをダブルクリックすることによって、[エラーポリシー] タブを開いてください。
2. [追加] をクリックしてください。
3. エラーリポジトリを開くために、[開始] カラムの右側で...をクリックしてください。
4. エラー範囲の開始値としてエントリを選択し、[OK] をクリックしてください。これを行うと、[コンポーネント] カラムは選択されたエラーに関連しているコンポーネントの名前が表示されます。
5. エラーリポジトリを開くために、[終了値] カラムの右側で...をクリックしてください。
6. エラー範囲の終了値としてエントリを選択し、[OK] をクリックしてください。
7. [開始値] カラムに入力した同じ値を、[終了値] カラムに入力することでエラーポリシーを 1つのエラーコードに設定することができます。
8. [エラーポリシー] カラムで、**アボート**を選択してください。
9. 上記に定義したエラーコードをフローが返す場合、これは、サーバに、実行してほしい動作になります。
10. 新しいエラーポリシーを保存するために、[保存] をクリックしてください。

これで、定義したエラー範囲内でエラーが発生すると、フローは中断されます。

エラーを受け取った時に、E メールを受け取るようにするには

Magic xpi に、特定のエラーが発生したときに常に Eメールの通知を送るように指示することができます。これを行うには、[エラーポリシー]タブで特定のエラーコードまたはエラーコードの範囲をエラーポリシーに定義します。エラーが発生すると、これらのエラーポリシーによって何を行うかを Magic xpi に指示します。

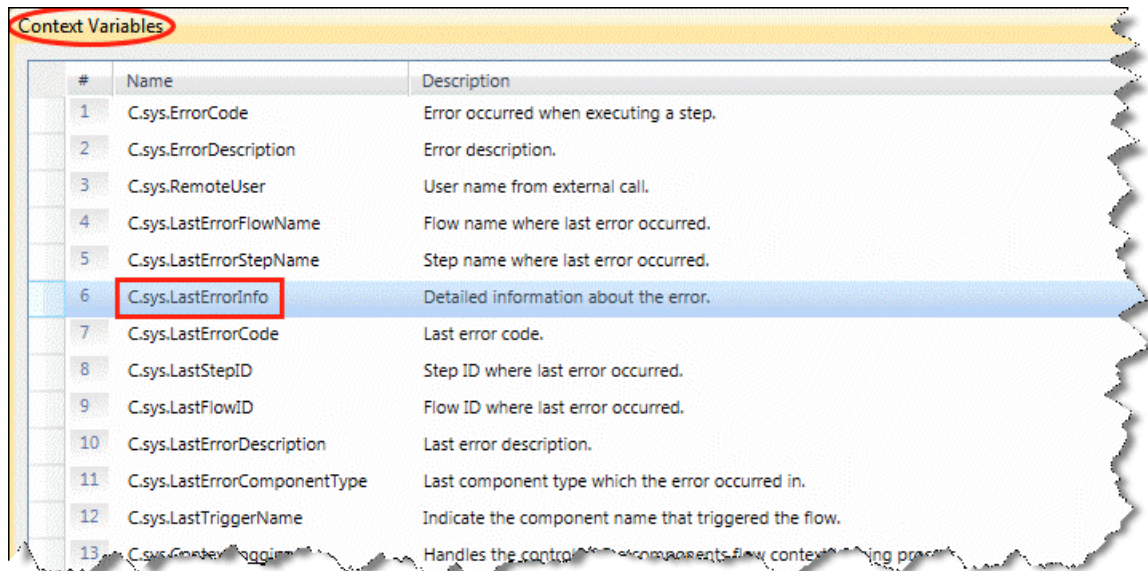
エラーポリシーを定義する時、[通知]カラム(以下のイメージに示しています)でメールを選択してください。



これで、エラーが発生すると常に、[E メール送信先]で定義された電子メールアドレスに、E メール通知が送られます。

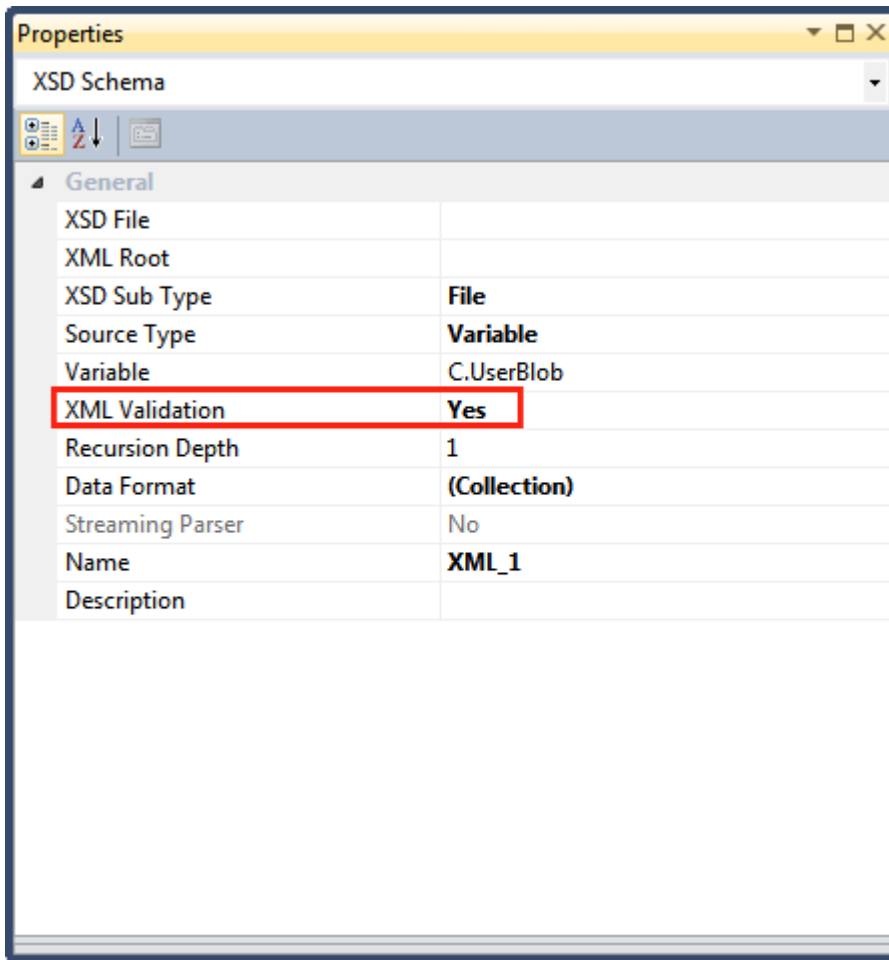
エラーフローでエラーについての追加情報を取得するには

C.sys.LastErrorInfo コンテキスト変数を使用することで、エラーフローでエラーについての詳細情報を取得することができます。この変数は BLOB タイプで、SQL ステートメントや Java エラー、およびその他を含めることができます。



エラーがデータベースで発生した場合、問題の多い SQL ステートメントは、**sys.LastErrorInfo** 変数に出力されます。

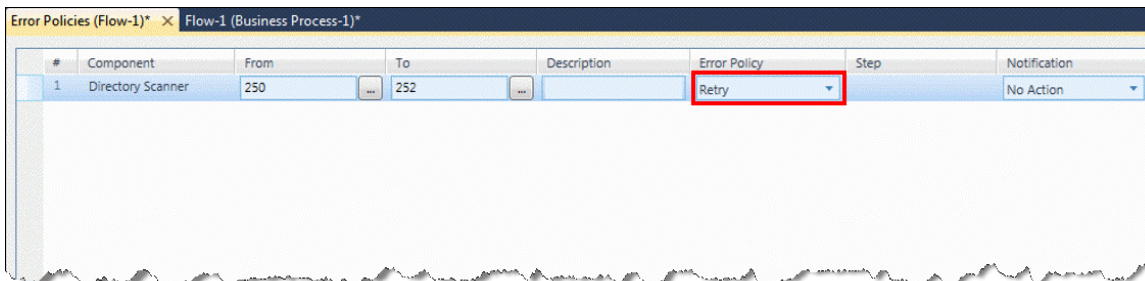
XML 送り元または送り先を使用している場合、XML の[プロパティ]ペイン(以下のイメージで示しています)の[XML 検証]プロパティで Yes を選択すると、**C.sys.LastErrorInfo** 変数に、XML が有効でない理由を含めます。



注意： SQL ステートメントを構文解析せずに、ステートメントの値を取得するノードの[プロパティ]ペインの[ログエラー]プロパティを使用することができます。

エラーが発生したら、ステップのリトライ回数をコントロールするには

Magic xpi は、エラーの発生時にステップを繰り返して実行する回数を定義することができます。最初にすべき事は、[エラーポリシー]タブの[エラーポリシー]プロパティをリトライ(以下のイメージに示すように)に設定することです。



次に、実行するリトライ回数を定義する必要があります。これは、ifs.ini ファイルの StepRetryLimit= flag(以下のイメージで示しています)で行います。デフォルトは、3 回です。

```

[MAGIC_IBOLT]MonitorLogLevel = * All
[MAGIC_IBOLT]DisableSecurity=N
[MAGIC_IBOLT]RecoveryLocation=%Magicxpi%rcv%sl%
[MAGIC_IBOLT]SchedulerStartupRetainMode = None
[MAGIC_IBOLT]SchedulerStartupRetainPeriod = All
[MAGIC_IBOLT]ActivityLogSynchronizationMode = Async
[MAGIC_IBOLT]ParallelExecutionFailure = 3,3
[MAGIC_IBOLT]Debugger = N
[MAGIC_IBOLT]StepRetryLimit = 3
[MAGIC_IBOLT]TimeoutInterval = 1
[MAGIC_IBOLT]ThreadExceptionInterval = 60
[MAGIC_IBOLT]RuntimeVariableTypeValidation = Y
[MAGIC_IBOLT]CleanTemp = N
[MAGIC_IBOLT]JMSStartupConnectionFailurePolicy = Retry
[MAGIC_IBOLT]SchedulerPersistency = N
[MAGIC_IBOLT]SchedulerRecoveryLocation = %currentprojectdir%Recovery%sl%
[MAGIC_IBOLT]DebuggerLicenseMaxRetry = 30
[MAGIC_IBOLT]SAPB1RemoveEmptyNodes = N
[MAGIC_IBOLT] FlowTimeoutGraceTime = 300
[MAGIC_DATABASES]MemoryLarge = DBMS,22, , , , , , NoMagicRecordLock, ChangeFileInToolkit, N
[MAGIC_DATABASES]MemoryXL = DBMS,22, , , , , , NoMagicRecordLock, ChangeFileInToolkit, N
[MAGIC_SPECIALS]SpecialBackgroundDelayActionInterval=25
[MAGIC_SPECIALS]CtxMaxShutdownTime=2
[MAGIC_SPECIALS]SpecialEventQSizeLimit = 100
[MAGIC_SPECIALS]SpecialEventQMaxInterval = 10
[MAGICXPI_GS]ReservedLicenseThreads=0

```

これで、定義されたエラー範囲で、エラーが発生すると、指定したリトライ回数に応じて、Magic xpi は、再びステップを実行しようとします。

特定のエラーが発生した後に、フローを異なる場所に移動するには

エラーが発生すると、Magic xpi に、フロー内の異なる場所に移動するように指示することができます。これは、[エラーポリシー]タブ(以下のイメージに示すような)で行います。


#	Component	From	To	Description	Error Policy	Step	Notification
1	Directory Scanner	250	252		Jump	3	Email

Tells Magic xpi to jump to another step if an error occurs.

Tells Magic xpi which step to jump to if an error occurs.

エラーが発生した時に、フローを異なる場所に移動するには：

- ソリューションエクスプローラの関連したフロー下で、エラーポリシーをダブルクリックすることによって、[エラーポリシー]タブを開いてください。
- [追加] をクリックしてください。
- エラーリポジトリを開くために、[開始] カラムの右側で...をクリックしてください。
- エラー範囲の開始値としてエントリを選択し、[OK] をクリックしてください。これを行うと、[コンポーネント] カラムは選択されたエラーに関連しているコンポーネントの名前が表示されます。
- エラーリポジトリを開くために、[終了値] カラムの右側で...をクリックしてください。
- エラー範囲の終了値としてエントリを選択し、[OK] をクリックしてください。

7. [開始値] カラムに入力した同じ値を、[終了値] カラムに入力することでエラーポリシーを 1 つのエラーコードに設定することができます。
8. [エラーポリシー] カラムで、**ジャンプ**を選択してください。
9. 上記に定義したエラーコードをフローが返す場合、これは、サーバに、実行してほしい動作になります。
10. [ステップ] カラムで、ステップリストを開くために  をクリックしてください。ここで、エラーが発生した時のフローの移動先のステップを選択してください。
11. 新しいエラーポリシーを保存するために、[保存] をクリックしてください。

これで、定義したエラー範囲内でエラーが発生すると、フローは定義されたステップに移動します。

どのフローがエラーを呼び出したかを識別するには

たくさんのフローを含んでいるプロジェクトを作成する時に、エラーが発生した特定のフローを識別することができることは非常に便利です。**C.sys.LastErrorFlowName** コンテキスト変数を使用することで、この情報を取得することができます。サーバがステップを実行するたびに、**C.sys.LastErrorFlowName** 変数は、エラーコードが返った時にエラーが発生したフロー名で更新されます。

Context Variables		
#	Name	Description
1	C.sys.ErrorCode	Error occurred when executing a step.
2	C.sys.ErrorDescription	Error description.
3	C.sys.RemoteUser	User name from external call.
4	C.sys.LastErrorFlowName	Flow name where last error occurred.
5	C.sys.LastErrorStepName	Step name where last error occurred.
6	C.sys.LastErrorInfo	Detailed information about the error.
7	C.sys.LastErrorCode	Last error code.
8	C.sys.LastStepID	Step ID where last error occurred.
9	C.sys.LastFlowID	Flow ID where last error occurred.
10	C.sys.LastErrorDescription	Last error description.
11	C.sys.LastErrorComponentType	Last component type which the error occurred in.
12	C.sys.LastTriggerName	Indicate the component name that triggered the flow.
13	C.sys.ContextLogging	Handles the control of the components flow context logging process.
14	C.sys.InvokingBPName	This variable will hold the calling BP name when using: Invoke Flow, Call Flow, or Error Flow.
15	C.UserString	Passes alpha-numeric information between components in flows.
16	C.UserCode	Passes only numeric information between components in flows.

どのステップがエラーを呼び出したかを識別するには

たくさんのフローを含んでいるフローを作成する時に、エラーが発生した特定のステップを識別する必要があるかもしれません。**C.sys.LastErrorStepName** コンテキスト変数を使用することで、この情報を取得することができます。サーバがステップを実行するたびに、**C.sys.LastErrorStepName** 変数は、エラーコードが返った時にエラーが発生したステップ名で更新されます。

Context Variables		
#	Name	Description
1	C.sys.ErrorCode	Error occurred when executing a step.
2	C.sys.ErrorDescription	Error description.
3	C.sys.RemoteUser	User name from external call.
4	C.sys.LastErrorFlowName	Flow name where last error occurred.
5	C.sys.LastErrorStepName	Step name where last error occurred.
6	C.sys.LastErrorInfo	Detailed information about the error.
7	C.sys.LastErrorCode	Last error code.
8	C.sys.LastStepID	Step ID where last error occurred.
9	C.sys.LastFlowID	Flow ID where last error occurred.
10	C.sys.LastErrorDescription	Last error description.
11	C.sys.LastErrorComponentType	Last component type which the error occurred in.
12	C.sys.LastTriggerName	Indicate the component name that triggered the flow.

どのトリガがエラーを呼び出したかを識別するには

エラーを呼び出すトリガについての情報を取得するには、**C.sys.LastTriggerName** コンテキスト変数を使用する必要があります。この変数は、フローを起動したコンポーネント名を保持し、各トリガの呼出しによって更新されます。変数は、どのトリガが起動したかをチェックするためにフローによって使用されます。

Context Variables		
#	Name	Description
1	C.sys.ErrorCode	Error occurred when executing a step.
2	C.sys.ErrorDescription	Error description.
3	C.sys.RemoteUser	User name from external call.
4	C.sys.LastErrorFlowName	Flow name where last error occurred.
5	C.sys.LastErrorStepName	Step name where last error occurred.
6	C.sys.LastErrorInfo	Detailed information about the error.
7	C.sys.LastErrorCode	Last error code.
8	C.sys.LastStepID	Step ID where last error occurred.
9	C.sys.LastFlowID	Flow ID where last error occurred.
10	C.sys.LastErrorDescription	Last error description.
11	C.sys.LastErrorComponentType	Last component type which the error occurred in.
12	C.sys.LastTriggerName	Indicate the component name that triggered the flow.
13	C.sys.ContextLogging	Handles the control of the components flow context logging process.
14	C.sys.InvokingBPName	This variable will hold the calling BP name when using: Invoke Flow, Call Flow, or Error Flow.
15	C.UserString	Passes alpha-numeric information between components in flows.
16	C.UserCode	Passes only numeric information between components in flows.

どの最終ステップ ID がエラーを呼び出したかを識別するには

エラーを呼び出したステップの ID または名前を見つけ出すには、**C.sys.LastStepID** コンテキスト変数を使用してください。この変数は最後に実行されたステップのステップ ID を保持し、各コンポーネントの実行後に自動的に設定されます。従って、エラーがその特定のステップで発生した場合、この変数はそのステップの ID を提供します。

Context Variables		
#	Name	Description
1	C.sys.ErrorCode	Error occurred when executing a step.
2	C.sys.ErrorDescription	Error description.
3	C.sys.RemoteUser	User name from external call.
4	C.sys.LastErrorFlowName	Flow name where last error occurred.
5	C.sys.LastErrorStepName	Step name where last error occurred.
6	C.sys.LastErrorInfo	Detailed information about the error.
7	C.sys.LastErrorCode	Last error code.
8	C.sys.LastStepID	Step ID where last error occurred.
9	C.sys.LastFlowID	Flow ID where last error occurred.
10	C.sys.LastErrorDescription	Last error description.
11	C.sys.LastErrorComponentType	Last component type which the error occurred in.
12	C.sys.LastTriggerName	Indicate the component name that triggered the flow.
13	C.sys.ContextLocation	Handles the control of the components flow context logging process.

どの最終フローID がエラーを呼び出したかを識別するには

エラーを呼び出したフローの ID または名前を見つけ出すには、**C.sys.LastFlowID** コンテキスト変数を使用してください。この変数は最後に実行されたステップのフローID を保持し、各コンポーネントの実行後に自動的に設定されます。従って、エラーがその特定のフローで発生した場合、この変数はそのフローの ID を提供します。

Context Variables		
#	Name	Description
1	C.sys.ErrorCode	Error occurred when executing a step.
2	C.sys.ErrorDescription	Error description.
3	C.sys.RemoteUser	User name from external call.
4	C.sys.LastErrorFlowName	Flow name where last error occurred.
5	C.sys.LastErrorStepName	Step name where last error occurred.
6	C.sys.LastErrorInfo	Detailed information about the error.
7	C.sys.LastErrorCode	Last error code.
8	C.sys.LastStepID	Step ID where last error occurred.
9	C.sys.LastFlowID	Flow ID where last error occurred.
10	C.sys.LastErrorDescription	Last error description.
11	C.sys.LastErrorComponentType	Last component type which the error occurred in.
12	C.sys.LastTriggerName	Indicate the component name that triggered the flow.
13	C.sys.ContextLocation	Handles the control of the components flow context logging process.

エラー処理フローからのフローにエラーを返すには

Magic xpi は、2 つの専用のコンテキスト変数、**C.sys.LastErrorCode** と **C.sys.LastErrorDescription** を持っています。これは、新しいエラーが発生するまで、どのジョブがフローで発生した最後のエラーを保持しているかを提供します。

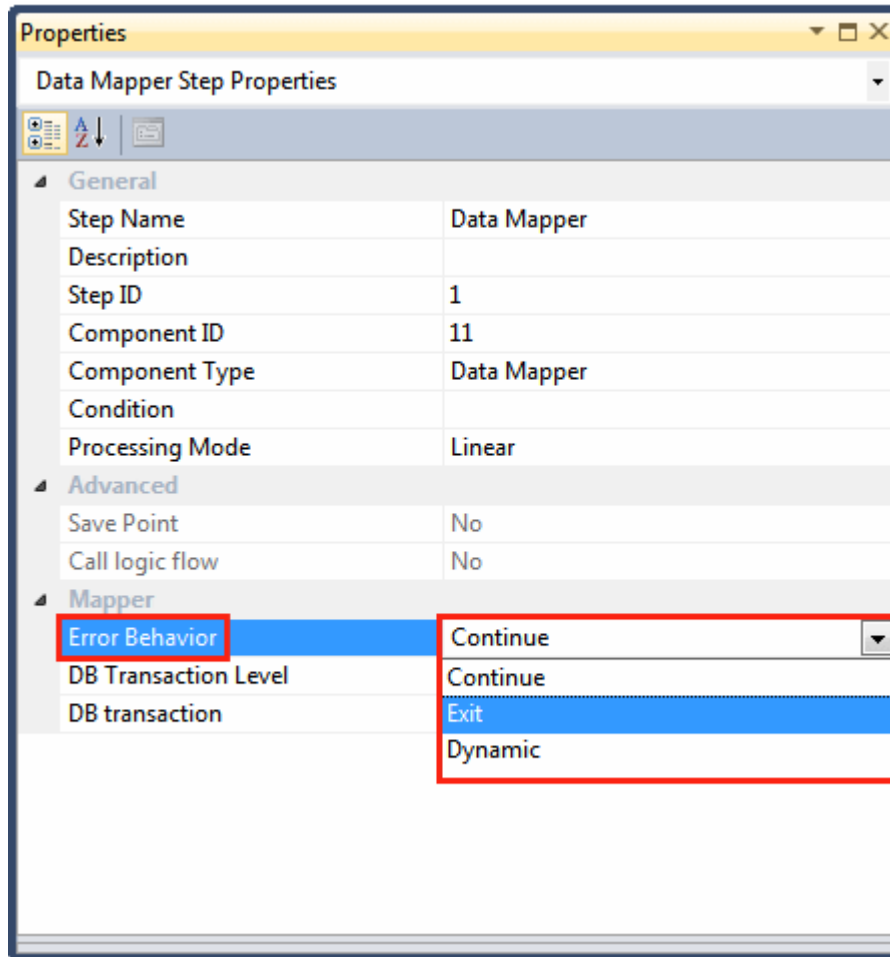
Context Variables		
#	Name	Description
1	C.sys.ErrorCode	Error occurred when executing a step.
2	C.sys.ErrorDescription	Error description.
3	C.sys.RemoteUser	User name from external call.
4	C.sys.LastErrorFlowName	Flow name where last error occurred.
5	C.sys.LastErrorStepName	Step name where last error occurred.
6	C.sys.LastErrorInfo	Detailed information about the error.
7	C.sys.LastErrorCode	Last error code.
8	C.sys.LastStepID	Step ID where last error occurred.
9	C.sys.LastFlowID	Flow ID where last error occurred.
10	C.sys.LastErrorDescription	Last error description.
11	C.sys.LastErrorComponentType	Last component type which the error occurred in.
12	C.sys.LastTriggerName	Indicate the component name that triggered the flow.

エラーの詳細は、**C.sys.LastErrorCode** 値によって **C.sys.ErrorCode** コンテキスト変数を自動的に更新することでフローに返されます。

注 意： 専用のエラー処理フローを作成していない場合、ステップの実行を許すか防止するための条件として、**C.sys.ErrorCode** を使用することができます。

データベースエラーが発生した場合に、データマップを終了するには

フローにおいてステップとしてのデータマップを設定する時に、データベースエラーがそのステップで発生した場合の動作を定義することができます。データマップの[プロパティ]ペイン(以下のイメージに示すような)で[エラー動作]プロパティを使用して、これを行います。



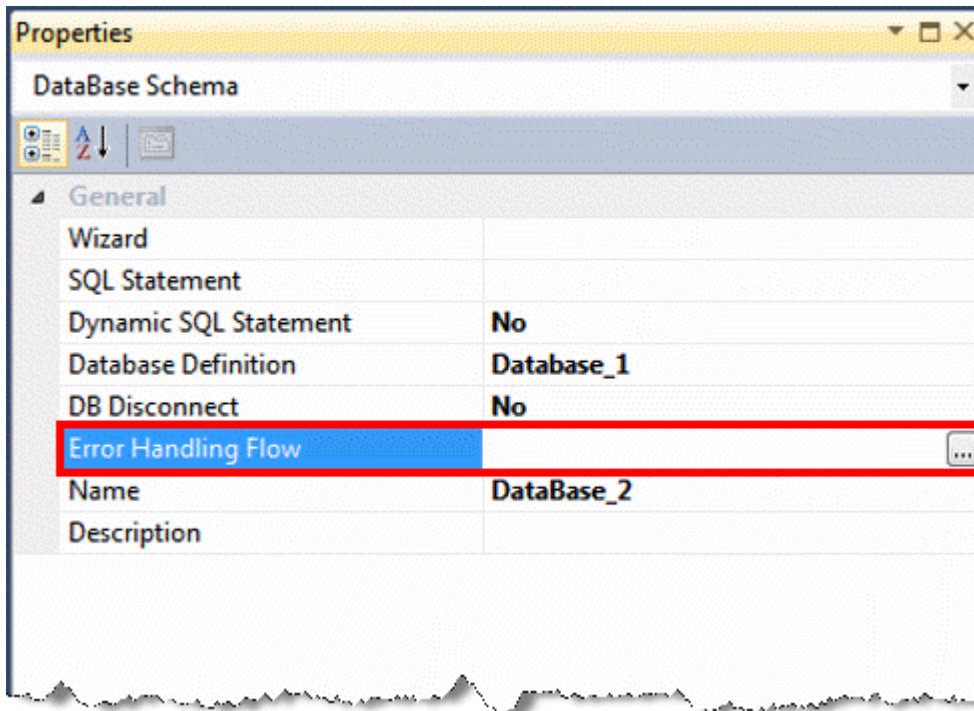
データベースエラーが発生したときにデータマッパを終了するには、[エラー動作]プロパティを**終了**または**ダイナミック**に設定してください。

終了は直ちにステップ実行を終了させ、フローに定義されたエラーポリシーが実行されます。エラーが発生したポイント以降に定義されたマッピングは実行されません。

ダイナミックは、エラーフローがエラーで終了した場合、直ちにステップ実行を終了させます。エラーフローがエラーをクリアすると、フローは継続されます。これは、データベース送り先にみの関連していることに注意してください。

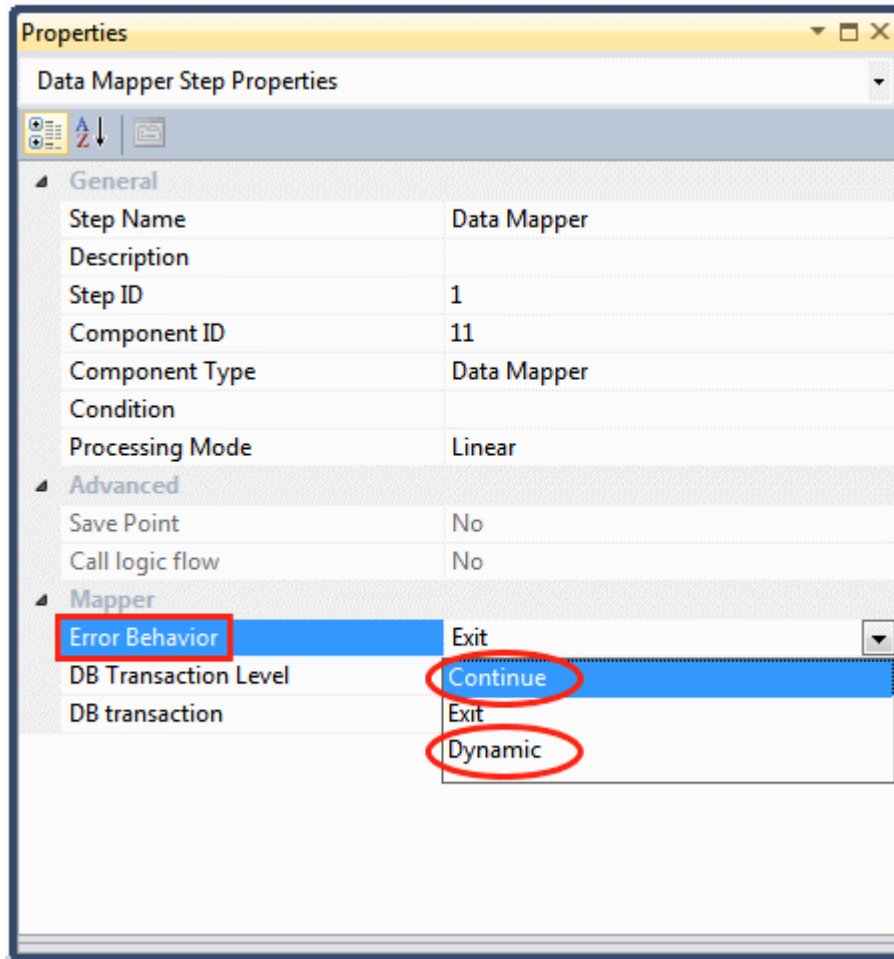
例外が発生したデータベースのレコードフィールドを取り出すには

例外が発生したデータベースのレコードフィールドを取得するためには、データベースの[プロパティ]ペインの[エラー処理フロー]プロパティを使用してください。このプロパティは、データベース送り先レベルで発生する任意のエラーを処理するためにエラーフローを選択します。失敗ステートメントからエラー処理フローのフロー変数にレコードフィールドをマッピングするエラーフローを作成することができます。このようにして、例外が発生したレコードフィールドを確認することができます。



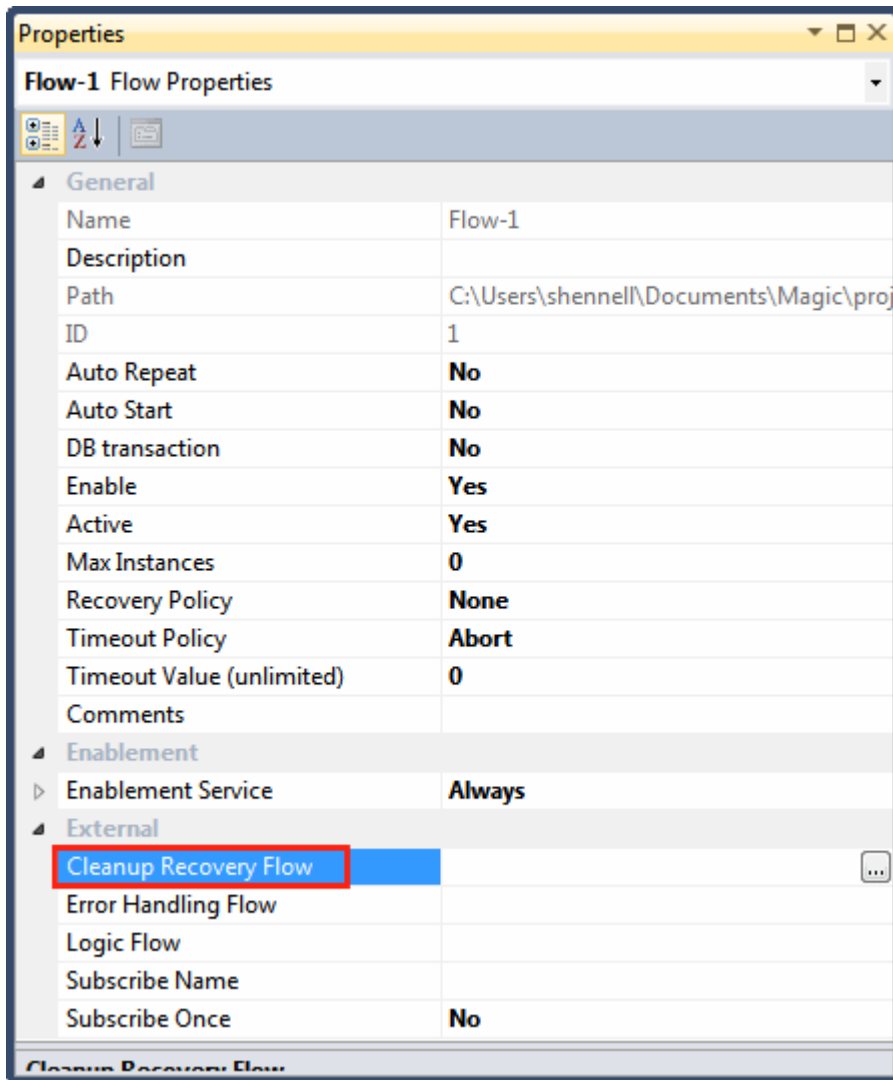
エラーを無視し、次のレコードに継続するには

デフォルトでは、**C.sys.ErrorCode** と **C.sys.ErrorDescription** 変数は、ステップを入力することで自動的にクリアされます。エラーフローが定義されていない場合、Magic xpi は関連したエラーポリシーがあるかどうかを確認します。特定のエラーコード、またはエラーコード範囲でエラーポリシーが定義されていない場合、データマップの[プロパティ]ペインの[エラー動作]プロパティで**継続**か**ダイナミック**のどちらかが選択された場合に、次のレコードへ移動します。



クラッシュから回復する時にリソースを消去することができるか

クラッシュから回復する時にリソースをクリアにするには、リソースをクリアにする開放するためにクリーンアップリカバリフローを作成する必要があります。フローに対して専用のクリーンアップリカバリフローを定義してください。これは、フローの[プロパティ]ペイン(以下のイメージに示するような)の[クリーンアップリカバリフロー]プロパティで定義されます。



クリーンアップリカバリフローは、フローが中断されるか、再起動されるか、セーブポイントから開始される直前に、リカバリプロセスの一部として呼び出されます。

フローにタイムアウトを設定するには

フローの[タイムアウトポリシー]プリシーは、Magic xpi がフローを終了する前に、どれくらい実行するかを定義することができます。

これは、IBM i とコマンドラインなどの外部システムと接続する場合にフローが無制限に応答を待つことを防止するために、フローにタイムアウトを設定することが推奨されます。

タイムアウトをフローに設定する時、タイムアウトに達したら何を行うかを定義することができます。[タイムアウトポリシー]プロパティのオプションは以下の通りです。

- アボート
- 再起動
- なし

タイムアウトをフローに設定する時、最初に、フローの実行にどれだけの時間がかかるかをチェックする必要があります。フロータイムアウトが無制限のフロー(他のフローから呼ばれないフロー)にだけ作用することに注意してください。これは、呼び出すフローと呼び出されるフローでタイムアウトが定義されると、呼び出すフローのタイムアウトだけが実施されることを意味しています。

フロータイムアウトは、ユーザ定義の猶予期間の後にスレッドを停止させま、[タイムアウトポリシー]プロパティで定義されたリカバリが適用されます。

式

式

以下のトピックは、Magic xpi の中のエラー処理について扱います。

[式ウィンドウで式の体制を整えるには](#)

[自動的にタイプ入力された関数の名前を完了するには](#)

[関数のヘルプページを表示するには](#)

[式エディタから関数リストを開くには](#)

[式の中で文字列値を定義するには](#)

式ウィンドウで式の体制を整えるには

式の内容によっては記述内容が長く複雑になる場合があります。例えば、ネストされた IF 文を定義する場合は以下ようになります。

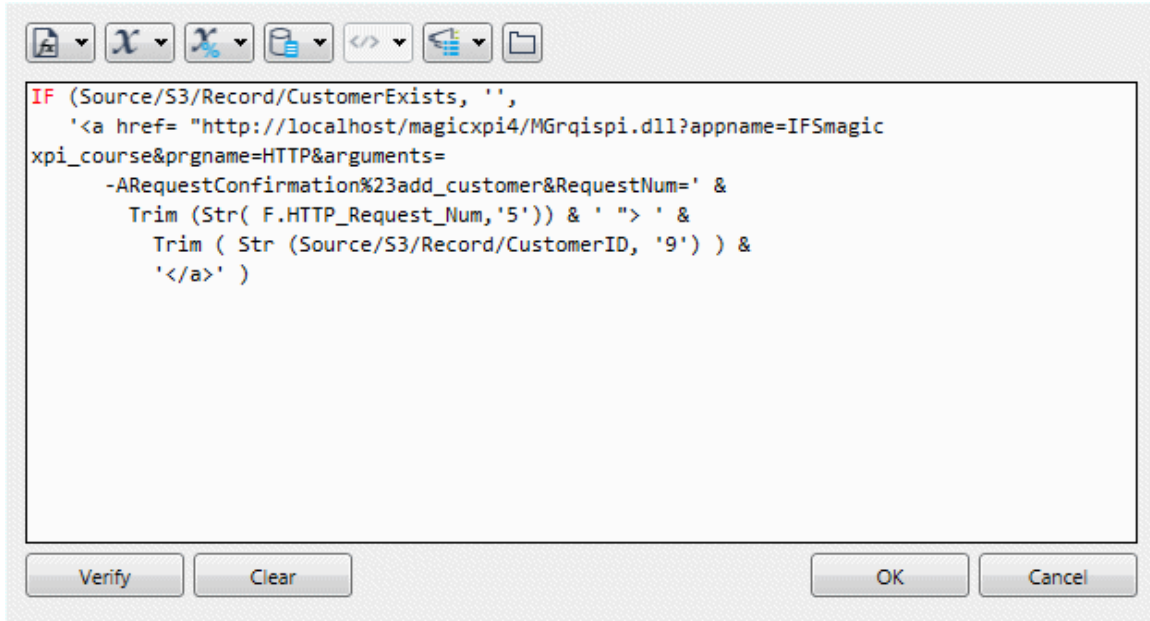
```
IF (Source/S3/Record/CustomerExists, '', '<a href=
"http://localhost/magicxpi4/MGrqispi.dll?appname=IFSmagic_xpi_course&prgname=HTTP&arguments=
-AResultConfirmation%23add_customer&RequestNum=' & Trim (Str( F.HTTP_Request_Num,'5')) & ' ">
' & Trim ( Str (Source/S3/Record/CustomerID, '9')) & '</a>')
```

構文上この式が正しくても、人が理解するには難しいものがあります。この式に改行や空白を追加することで、より理解しやすいものにすることができます。例えば以下ようになります。

```
IF (Source/S3/Record/CustomerExists, '',
    '<a href= "http://localhost/magicxpi4/MGrqispi.dll?appname=IFSmagic
xpi_course&prgname=HTTP&arguments=
    -AResultConfirmation%23add_customer&RequestNum=' &
    Trim (Str( F.HTTP_Request_Num,'5')) & ' "> ' &
    Trim ( Str (Source/S3/Record/CustomerID, '9')) &
    '</a>')
```

ここでは、Magic xpi で式の体裁を整える方法について説明します。

[式の体裁を整えるには：](#)



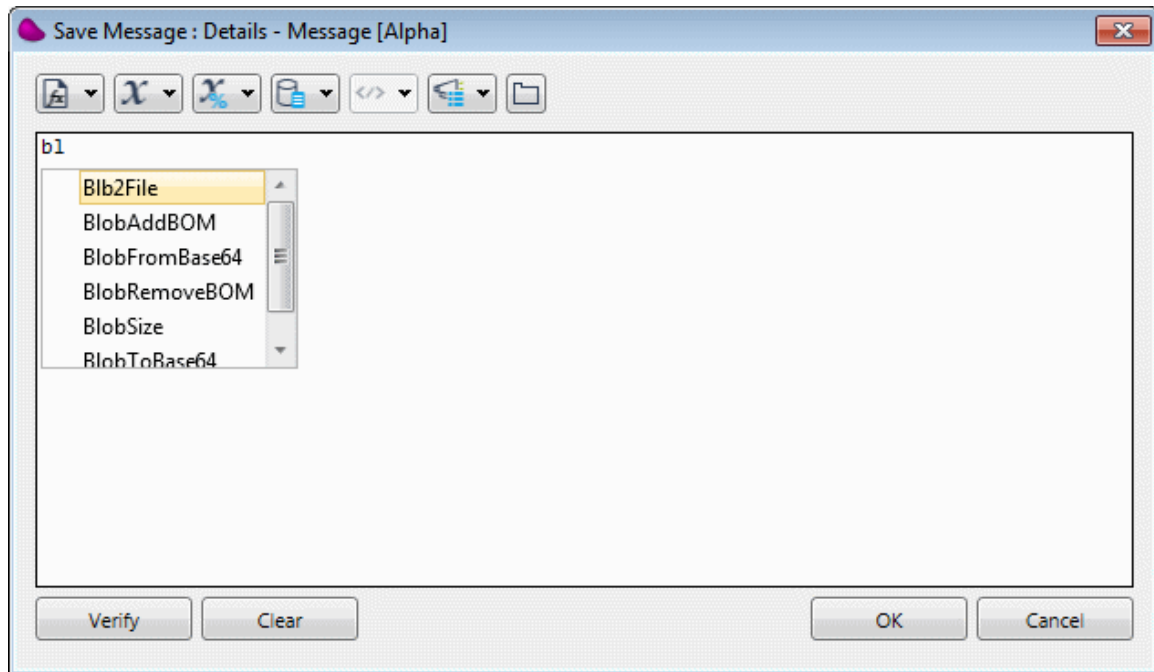
1. ボタンから [式エディタ] にズームしてください。
2. **Enter** キーを押下することで式が改行表示されます。また、必要に応じてインデントを加えるために空白を入力することもできます。
3. 編集が終了したら、[OK] ボタンや右上の **X** ボタンをクリックするか、**Esc** キーを押下します。

注 改行を入力すると式の中に CRLF コードが入力されます。これは Magic xpi の式内であれば問題はありません。

自動的にタイプ入力された関数の名前を完了するには

キー操作を減らすために、Magic xpi には関数名に対して オートコンプリート機能が備わっています。関数名を入力する場合、入力操作のほとんどを Magic xpi に依存させることができます。

オートコンプリート機能を使用するには：

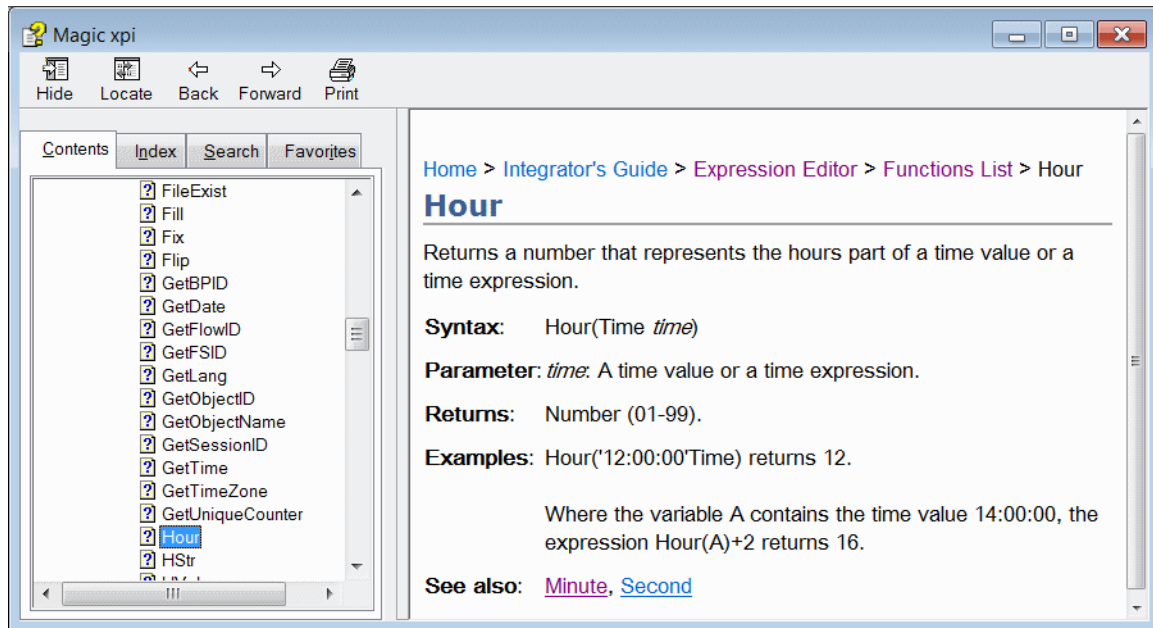


1. 入力したい関数名の最初の数文字を入力します。この例では、'bl' が入力されています。大文字小文字は区別しません。
2. **CTRL+スペースバー**を押下してください。関数リストが表示され、名前の最初の文字が入力した文字と一致する関数に位置付けされます。
3. カーソルを下に動かして使用したい関数に位置付けます。この場合、カーソルキーを使用して動かすこともできますが、関数名を続けて入力することでも自動的にカーソルが移動します。
4. 使用したい関数名にカーソルが位置付いたら、**Tab** を押下します。関数名が自動的に追加されます。

注 意 :

- 入力文字に該当する関数が 1 つだけの場合は、リストボックスが開かず自動的にその関数が挿入されます。

関数のヘルプページを表示するには



どのような言語で開発する場合でも、全ての関数(メソッド)がどのように動作するものかをすべて覚えていることは難しいものです。Magic xpi では、プログラムを作成中に該当するオブジェクトについて説明されたヘルプトピックを表示する機能があります。ヘルプを表示させるには、2つの方法があります。

任意の場所から関数のヘルプトピックを探すには：

スタジオでは、どこからでも関数のヘルプトピックを表示させることができます。

1. **F1** を押下します。これによってオブジェクトに対応したヘルプトピックが表示されます。
2. [インデックス] タブをクリックします。
3. 関数の名前を入力します。

入力した関数に対応するトピックが表示されます。また、[検索]タブを使用することでこの関数名を使用しているトピックを探すことができます。

ヒント： 開発作業中(特に Magic xpi の勉強中)は、ヘルプを開いたままにしておくことで参照にかかる時間が早くなります。

式エディタでヘルプを使用するには：

さらに、関数が含まれている式から、その関数に対応するヘルプトピックを表示させることができます。

1. [式] エディタを開きます。
2. 関数の記述場所にカーソルを置きます。
3. **F1** キーを押下します。

これで、パークした関数に対するヘルプトピックが表示されます。

関数リストからヘルプを表示するには：

Magic xpi 逆引き辞典

関数を入力する便利な機能の1つに関数リストがあります。ここでは簡単にヘルプを表示させることができるだけでなく、関数を選択すると、括弧やパラメータのプレースホルダが自動的に入り、関数入力が楽になります。

関数の入力方法については、[式エディタから関数リストを開くには](#)を参照してください。

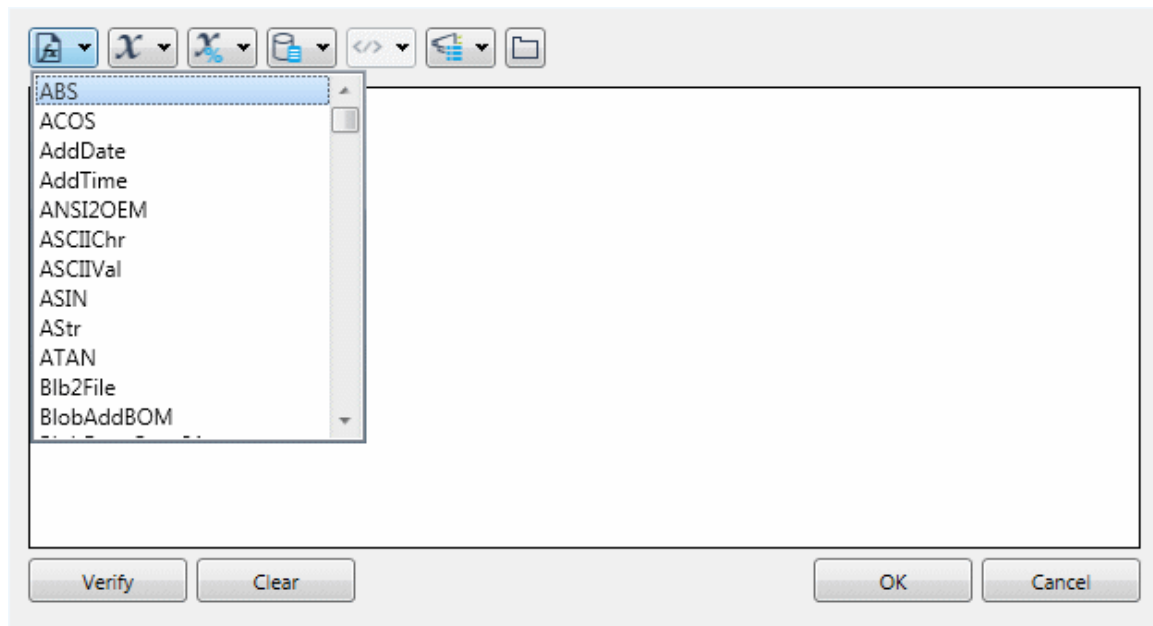
式エディタから関数リストを開くには

式エディタ で編集集中に、以下の操作を行うことで関数リストを表示させることができます。

- **Ctrl+SPACE** の押下
- エディタの上部で  ボタンをクリックします。

関数リストが開いたら、以下の手順で関数を選択することができます。

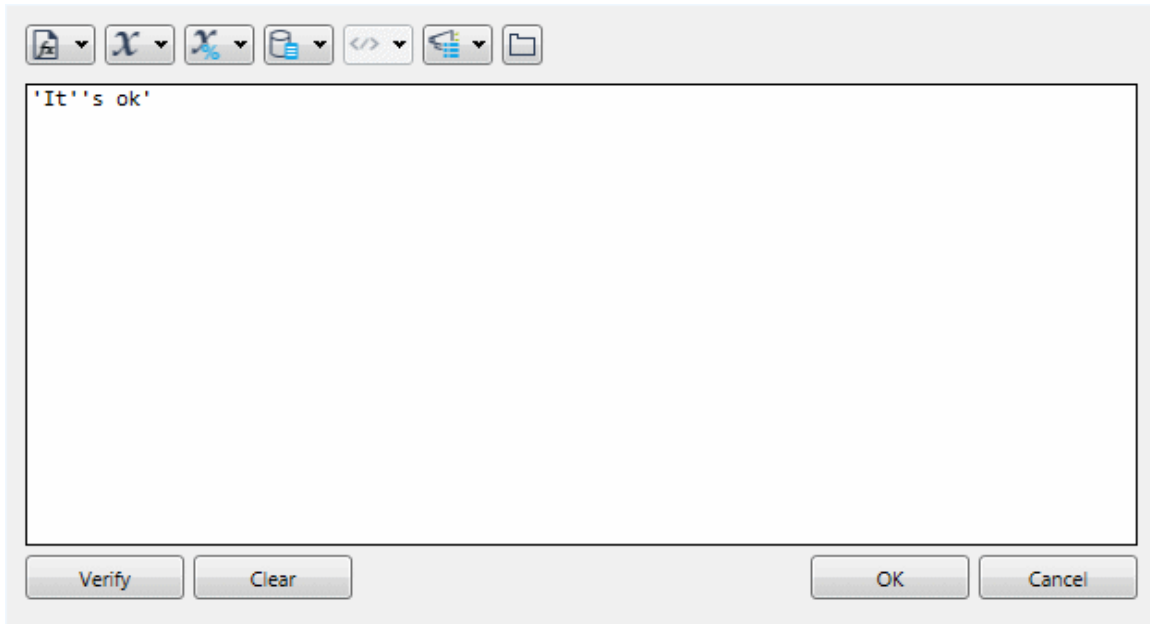
関数リストから関数を選択するには：



1. 式エディタ で、関数を設定したい場所にカーソルを置きます。
2. 上記で説明されたオプションの 1 つを使用して、関数リストを開いてください。
3. 使用したい関数名に位置付けると、関数のパラメータを表示するためにツールティップを使用することができます。どのような関数を使用すべきか確信がない場合は、必要なものを見つけるまで、この処理を繰り返すことができます。
4. 使用したい関数が見つかったら、そこでダブルクリックするか、**Enter** キーを押下してください。
5. これで、関数は式に書き込まれます。

式の中で文字列値を定義するには

式を Magic xpi に入力する際、すべての文字列はシングルクォーテーションで囲む必要があります。この指定をしない場合、文字列は項目のシンボル名か関数名として認識されます。



式の中で文字としてシングルクォーテーションを入力する必要がある場合は、2つのシングルクォーテーションを連続して入力します。

例:

'It''s ok'

は、次のように表示されます。

It's ok

JD Edwards Enterprise One コネクタ

JD Edwards Enterprise One コネクタ

以下のトピックは、Magic xpi の JD Edwards Enterprise One コネクタについて扱っています。

[Magic xpi で JD Edwards Enterprise One の呼び出しを暗号化するには](#)

[JD Edwards Enterprise One で新しい顧客レコードを作成するために Magic xpi を使用するには](#)

Magic xpi で JD Edwards Enterprise One の呼び出しを暗号化するには

Magic xpi は、JD Edwards Enterprise One ダイナミック Java コネクタを通して、JD Edwards サーバと接続します。サーバによって行われる通信は、動的な Java コネクタと JDE サーバの間で行われます。

セキュリティを適用する方法があるのであれば、そこで設定する必要があります。、JDE ダイナミック Java コネクタの通信をセキュリティ保護することを JDE のエキスパートに相談する必要があります。

JD Edwards Enterprise One で新しい顧客レコードを作成するために Magic xpi を使用するには

基本的な顧客を作成するために、以下の 2 つの標準の JDE のビジネス関数を呼ぶ必要があります。

- N0100041.AddressBookMasterMBF

- N0100042.MBFCustomerMaster

N0100041.AddressBookMasterMBF

最初に、住所録レコード(通常は、C タイプのレコード)を作成し、住所録番号(AN8)を取り出すために N0100041.AddressBookMasterMBF 関数を使用する必要があります。

1. JD Edwards Enterprise One コネクタをフローにドラッグしてください。
2. ステップ上で右クリックし、[構成] を選択してください。
3. [モジュール名] プロパティで、**N0100041** を入力してください。
4. [ビジネス関数] プロパティで、**AddressBookMasterMBF** を選択してください。
5. データマップを開き、以下のマッピングを接続側(最初の 5 つは必須です j)に入力してください。

入力は以下の通りです。

cActionCode = 'A' (追加の意味)

cUpdateMasterFile = '1'

cProcessEdits = '1'

szSearchType = 'C' (顧客の意味)

szAlphaName = [住所録アカウント名]

szSecondaryAlphaName = [2 番目の住所録アカウント名]

szMailingName = [メールの名前]

szSecondaryMailingName = [2 番目ののメール名前]

szAddressLine1 = [アドレス]

szPostalCode = [郵便番号]

szCity = [都市]

szState = 国コード、有効な国コードを指定。

szCountry = 会社のためにデフォルト国を使用している場合、szCountry = You は Null のままにすることができます。

szPhoneNumber1 = [電話番号]

AN8 を含む出力フィールドは以下の通りです。

mnAddressBookNumber

N0100042.MBFCustomerMaster

F03012 に顧客レコードを作成するために、N0100042.MBFCustomerMaster 関数を呼び出すステップで AN8 を使用します。

1. JD Edwards Enterprise One コネクタをフローにドラッグしてください。
2. ステップ上で右クリックし、[構成] を選択してください。
3. [モジュール名] プロパティで、**N0100042** を入力してください。
4. [ビジネス関数] プロパティで、**MBFCustomerMaster** を選択してください。
5. 住所録ファンクションコールから AN8 値を使用する必要があります。

6. AN8 値を接続側の **mnCustomerNumber** にマッピングしてください。
7. 接続側で、以下の最小のマッピングを入力してください。

```
mnCustomerNumber - the AN8 (Address Book # in the mnAddressBookNumber node)
from the previous function
cActionCode - 'A' (for Add)
cUpdateMasterFile - '1'
cProcessEdits - '1'
```

ライセンス

ライセンス

以下のトピックは、Magic xpi ライセンスについて扱っています。

[プロジェクトのためにスレッドを確保するには](#)

[プロジェクトの潜在的な並列実行機能を制限するには](#)

[プロジェクトが運用ライセンスを使用していることを確認するには](#)

[プロジェクトのライセンスを変更するには](#)

[ライセンスをリフレッシュするには](#)

[複数のサーバを使用するためにライセンスを設定するには](#)

プロジェクトのためにスレッドを確保するには

Magic xpi がマルチプロジェクトモードで動作する場合、プロジェクトを適切に実行するために一定のスレッド数を確保する必要があるかもしれません。例えば、複数のプロジェクトを実行していて、いくつかのプロジェクトが、ストレス条件(例えば、多くの Web サービスまたは HTTP リクエストがあることで)のもとですべてのスレッドを消費するかもしれない場合、継続的に動作させる必要がある重大なプロジェクトのために一定の(最小)のライセンススレッドを確保することを考慮する必要があります。

プロジェクトを実行する時、Magic xpi のワーカは、プロパティで定義されたライセンスのみをチェックアウトしますが、ライセンスを返すことはしません。ワーカがフローを完了しても、ライセンスはリリースしません。プロジェクトのワーカの残りは、プールからライセンスを使い続けることになります。

サーバが手動でシャットダウンされたり、モニタ経由で停止された場合、確保されたライセンスはプールに戻ります。そのプロジェクトを処理しているサーバはフローティングライセンスを消費し続けます。しかし、サーバが再起動されたり、そのプロジェクトのために別のサーバが起動された時は、確保されたライセンスは再度チェックアウトされます。

[設定]ダイアログボックスの[プロジェクト環境]セクションの[一般設定]の[予約ライセンススレッド]エントリに必要なライセンス値を設定します。プロジェクトの **ifs.ini** ファイルの[MAGICXPI_GS]セクションで、**ReservedLicenseThreads= flag** に値を設定することも可能です。フラグの値が、**0**(デフォルト値)に設定された場合、プロジェクトはライセンスを確保せず、ライセンスプールで利用可能なライセンスを使用します。


```
[MAGIC_IBOLT]ParallelExecutionFailure = 3,3
[MAGIC_IBOLT]DebugMapper = N
[MAGIC_IBOLT]StepRetryLimit = 3
[MAGIC_IBOLT]TimeoutInterval = 1
[MAGIC_IBOLT]ThreadExceptionInterval = 60
[MAGIC_IBOLT]RuntimeVariableTypeValidation = Y
[MAGIC_IBOLT]CleanTemp = N
[MAGIC_IBOLT]JMSStartupConnectionFailurePolicy = Retry
[MAGIC_IBOLT]SchedulerPersistency = N
[MAGIC_IBOLT]SchedulerRecoveryLocation = %currentprojectdir%Rec
[MAGIC_IBOLT]DebuggerLicenseMaxRetry = 30
[MAGIC_IBOLT]SAPBIRemoveEmptyNodes = N
[MAGIC_IBOLT]FlowTimeoutGraceTime = 300
[MAGIC_DATABASES]MemoryLarge = DBMS,22, , , , , , , NoMagicR
[MAGIC_DATABASES]MemoryXL = DBMS,22, , , , , , , NoMagicReco
[MAGIC_SPECIALS]SpecialBackgroundDelayActionInterval=25
[MAGIC_SPECIALS]CtxMaxShutdownTime=2
[MAGIC_SPECIALS]SpecialEventQSizeLimit = 100
[MAGIC_SPECIALS]SpecialEventQMaxInterval = 10
[MAGICXPI_GS]ReservedLicenseThreads=0
```

プロジェクトの潜在的な並列実行機能を制限するには

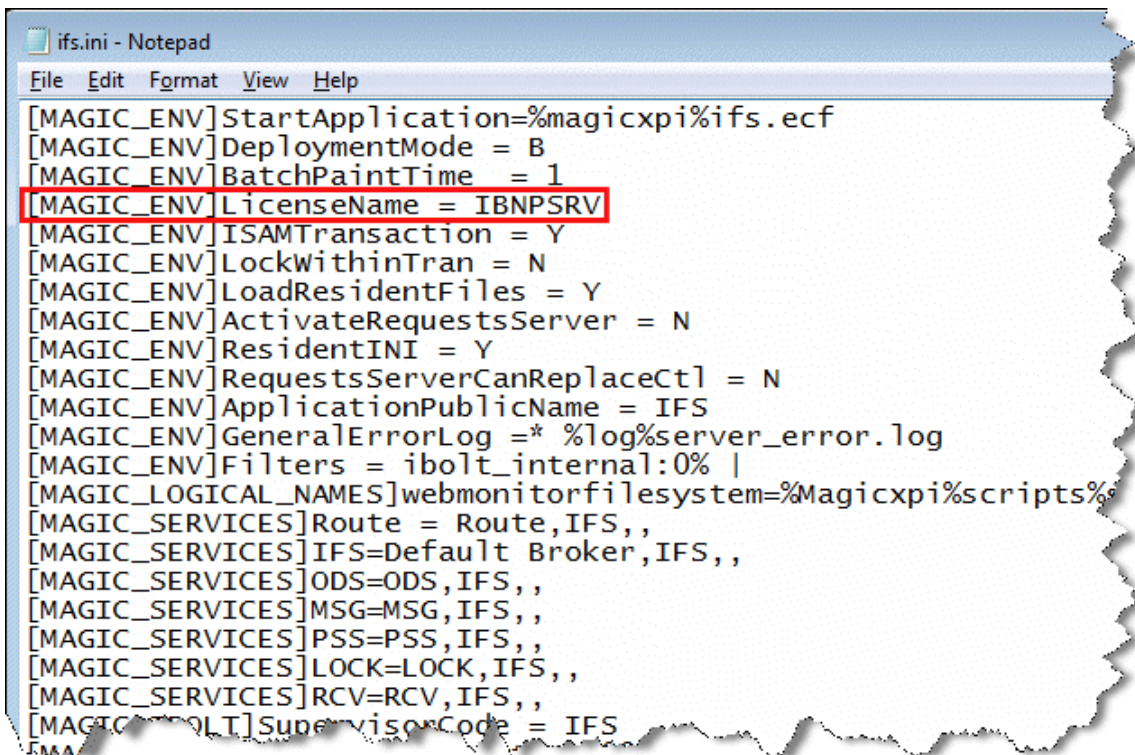
ワーカのプロジェクトの総数は、潜在的な最大の並列実行機能を定義します。

100 個のワーカを持つことができますが、ライセンスに基づいて実行するため、「潜在的」という言葉が使用されています。

プロジェクトが運用ライセンスを使用していることを確認するには

プロジェクトのために運用環境でのライセンスを使用していることを確認するには、ライセンスファイルに正しいライセンス名が定義されていることをチェックする必要があります。これは、Windows プラットフォームの場合が IBPRSRVI で、非 Windows プラットフォームの場合が IBPRSRVN になります。

ifs.ini ファイルの[MAGIC_ENV]LicenseName= flag に定義されたライセンス名がライセンスファイルに存在していなければなりません。一致していなければ、プロジェクトの ifs.ini ファイルでライセンス名を変更してください。また<Magic xpi インストール>\¥Runtime¥Magic xpa フォルダでライセンス名を変更した場合、作成する新しいプロジェクトでこのライセンス名を使用するようにしてください。



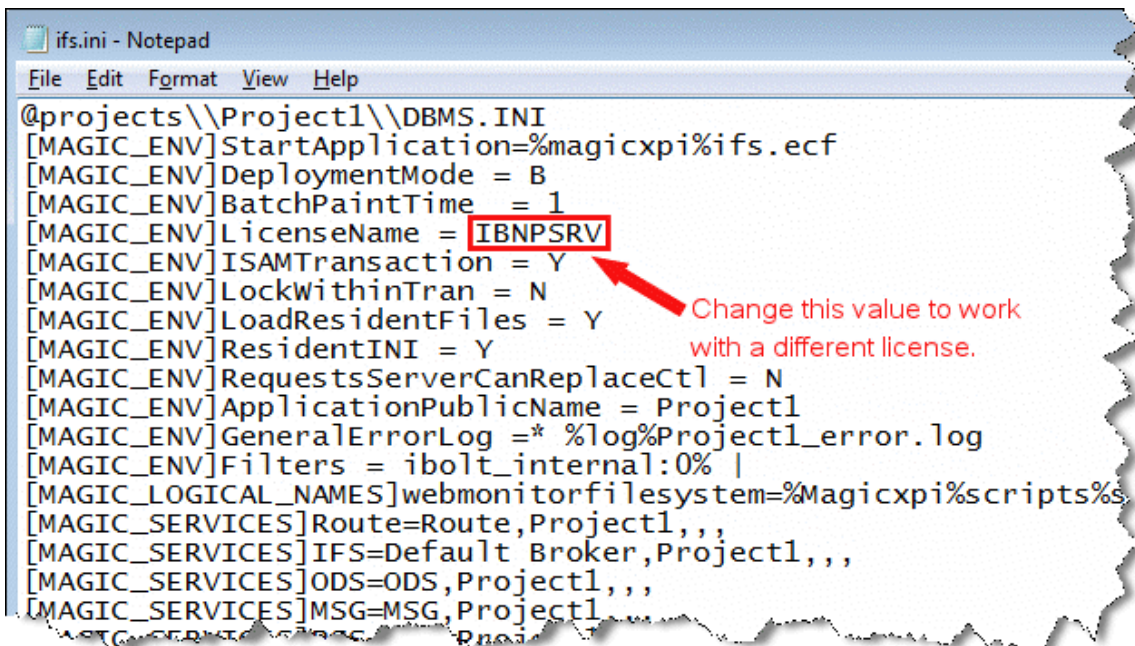
```

ifs.ini - Notepad
File Edit Format View Help
[MAGIC_ENV]StartApplication=%magicxpi%ifs.ecf
[MAGIC_ENV]DeploymentMode = B
[MAGIC_ENV]BatchPaintTime = 1
[MAGIC_ENV]LicenseName = IBNPSRV
[MAGIC_ENV]ISAMTransaction = Y
[MAGIC_ENV]LockWithinTran = N
[MAGIC_ENV]LoadResidentFiles = Y
[MAGIC_ENV]ActivateRequestsServer = N
[MAGIC_ENV]ResidentINI = Y
[MAGIC_ENV]RequestsServerCanReplaceCtl = N
[MAGIC_ENV]ApplicationPublicName = IFS
[MAGIC_ENV]GeneralErrorLog =* %log%server_error.log
[MAGIC_ENV]Filters = ibolt_internal:0% |
[MAGIC_LOGICAL_NAMES]webmonitorfilesystem=%Magicxpi%scripts%
[MAGIC_SERVICES]Route = Route,IFS,,
[MAGIC_SERVICES]IFS=Default Broker,IFS,,
[MAGIC_SERVICES]ODS=ODS,IFS,,
[MAGIC_SERVICES]MSG=MSG,IFS,,
[MAGIC_SERVICES]PSS=PSS,IFS,,
[MAGIC_SERVICES]LOCK=LOCK,IFS,,
[MAGIC_SERVICES]RCV=RCV,IFS,,
[MAGIC_SERVICES]SupervisorCode = IFS

```

プロジェクトのライセンスを変更するには

プロジェクトのライセンスを変更するには、projects/<プロジェクト名>フォルダに置かれている ifs.ini ファイルを開いてください。そこで、[MAGIC_ENV]LicenseName= flag の値を、使用したいライセンス名に変更するだけです。



```

ifs.ini - Notepad
File Edit Format View Help
@projects\\Project1\\DBMS.INI
[MAGIC_ENV]StartApplication=%magicxpi%ifs.ecf
[MAGIC_ENV]DeploymentMode = B
[MAGIC_ENV]BatchPaintTime = 1
[MAGIC_ENV]LicenseName = IBNPSRV
[MAGIC_ENV]ISAMTransaction = Y
[MAGIC_ENV]LockWithinTran = N
[MAGIC_ENV]LoadResidentFiles = Y
[MAGIC_ENV]ResidentINI = Y
[MAGIC_ENV]RequestsServerCanReplaceCtl = N
[MAGIC_ENV]ApplicationPublicName = Project1
[MAGIC_ENV]GeneralErrorLog =* %log%Project1_error.log
[MAGIC_ENV]Filters = ibolt_internal:0% |
[MAGIC_LOGICAL_NAMES]webmonitorfilesystem=%Magicxpi%scripts%
[MAGIC_SERVICES]Route=Route,Project1,,
[MAGIC_SERVICES]IFS=Default Broker,Project1,,
[MAGIC_SERVICES]ODS=ODS,Project1,,
[MAGIC_SERVICES]MSG=MSG,Project1,,

```

ライセンスをリフレッシュするには

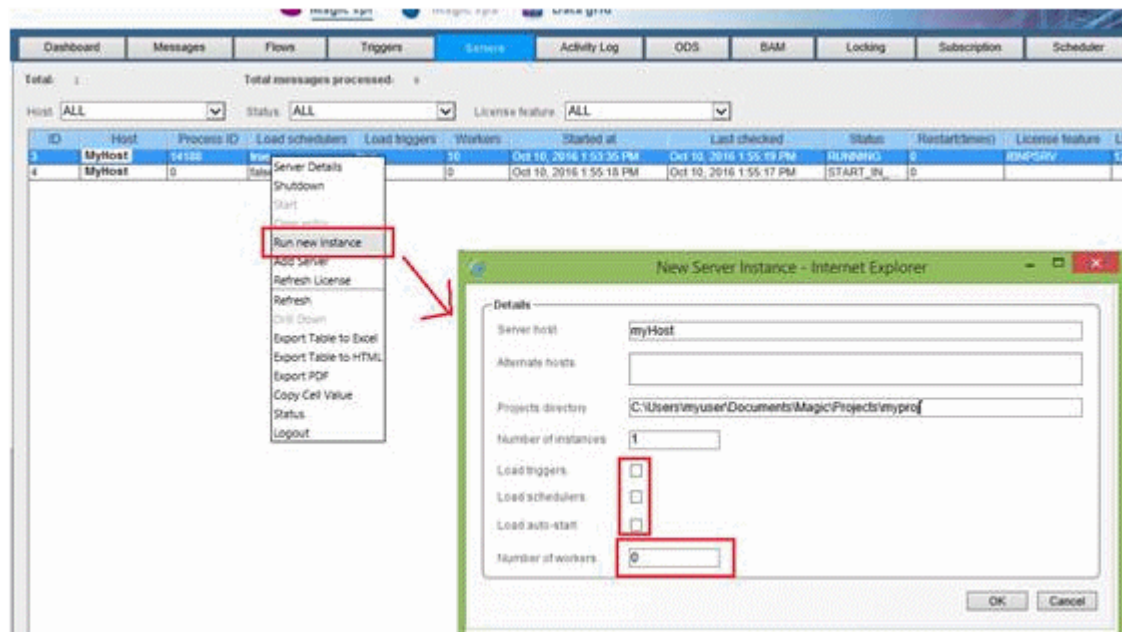
新しいライセンス(有効期限の延長や、スレッド数を増やす場合も含まれます)を受け取った場合にライセンスレコードをリフレッシュする必要があります。

リフレッシュ処理は簡単です。ただ、ライセンスファイルで定義されたホスト ID と一致しているホストで、新しいエンジンを起動させる必要があります。

前より少ないライセンス数や有効期限が前のライセンスより前の場合、ライセンスは更新されません。

ライセンスをリフレッシュするには：

1. <Magic xpi インストール>\Runtime\Magic xpa フォルダの既存のライセンスファイルを、新しいライセンスファイルに置き換えてください。
2. 0 個のワーカ、そしてトリガやスケジューラ、または自動起動のない設定でプロジェクトのサーバ(エンジン)を定義してください。
3. Magic xpa モニタで、ステップ 2 で定義したサーバを右クリックし、[新規インスタンスの実行] を選択してください。
4. エンジンが、実行ステータス([ステータス] カラムで **Running** が表示されます)に鳴るまで待ってください。
5. 再度サーバ上で右クリックし、シャットダウンを選択してください。



複数のサーバを使用するためにライセンスを設定するには

複数のサーバを使用する時、各プロジェクトは、**license.dat** ファイルにアクセスして読み込む必要があります。これは、以下の通りおこないます。

1. 購入後に受け取ったライセンスファイルをコピーし、サーバのすべて、または共有フォルダにコピーしてください。ライセンスファイルは、各サーバ毎に独自のライセンスファイルが必要ではないため、すべてのサーバでアクセスできる共有フォルダに置くことがよい選択です。

- 共有されたライセンスファイルを指定するために、**Magic.ini** ファイルの **[MAGIC_ENV] LicenseFile** のエントリを更新して、ライセンスファイルの位置を修正してください。例えば：
LicenseFile =\\10.1.1.6\\licenses\\License.dat
- プロジェクトの ifs.ini ファイルが、Windows 用で **[MAGIC_ENV] LicenseName = IBPRSRVI** などの製品ライセンスが設定されることを確認してください。

注 ライセンスの **HOSTID** フラグでライセンスを制限しているサーバは別として、ロードする他のサーバは実際のライセンスを追加しません。従って、ライセンスカウントは 0 になります。

ナビゲーションとワークスペース

ナビゲーションとワークスペース

以下のトピックは、Magic xpi のナビゲーションとワークスペースについて扱っています。

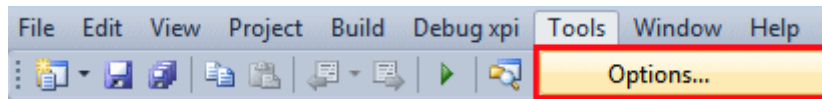
[Magic xpi スタジオのデフォルトのオープニング表示を定義するには](#)
[プロジェクトのオブジェクト内のテキストを検索するには](#)
[プロジェクトのエラーをチェックするには](#)
[ツールティップをより長く表示させるには](#)

Magic xpi スタジオのデフォルトのオープニング表示を定義するには

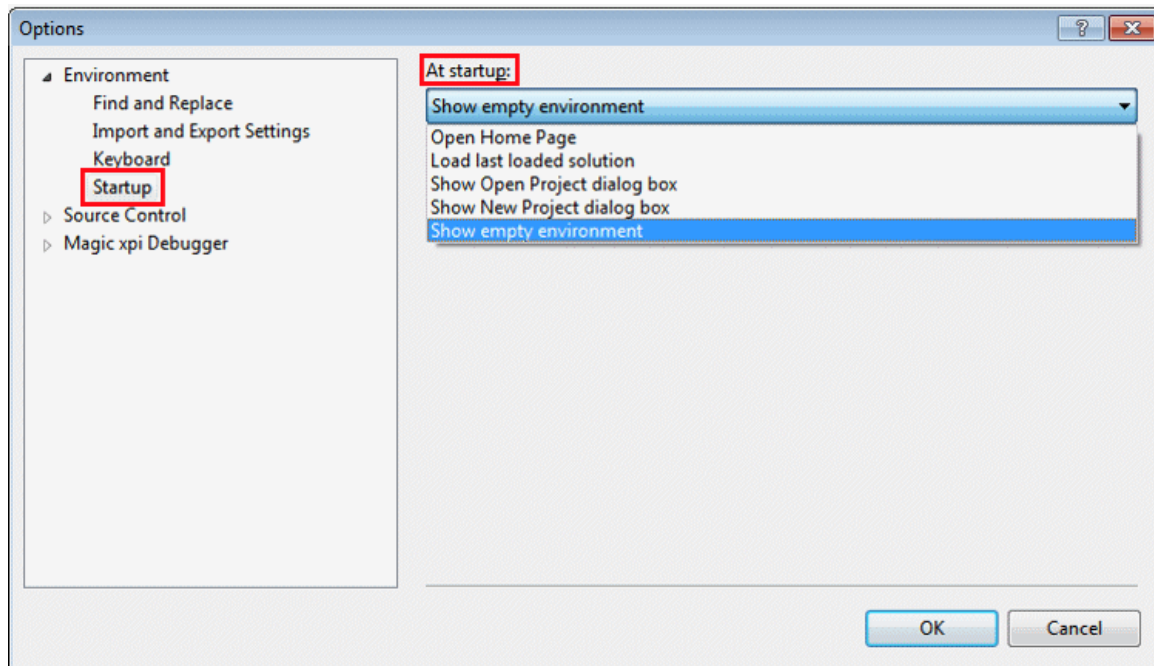
Magic xpi は、スタジオを最初に開く時にどのように表示させるかを定義することができます。

スタジオのデフォルトのオープニング表示を定義するには：

1. [ツール] メニューで、[オプション] を選択してください。



2. [環境] の配下の [オプション] ダイアログボックスで、**Startup** をクリックしてください。



3. [起動時] のドロップダウンリストで、以下のオプションのどれかを選択してください。
 - ホームページを開く
 - 前回読み込んだソリューションを読み込む
 - [プロジェクトを開く] ダイアログボックスを表示する
 - [新規プロジェクト] ダイアログボックスを表示する
 - 空の環境を表示する
4. [OK] をクリックしてください。

プロジェクトのオブジェクト内のテキストを検索するには

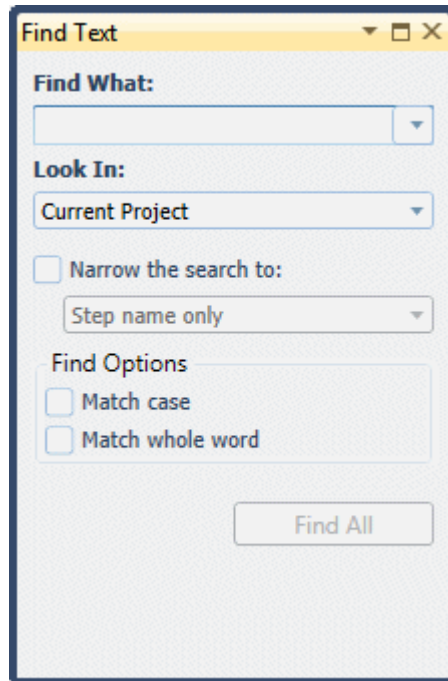
変更を行うためにプロジェクト内のテキストを検索すること便利です。特定のデータの名前を変更する必要が発生するかもしれません。例えば「Salesman」の代わりに「Salesrep」を使用する場合などです。ひょっとしたら、ハードコードされた会社名を変更する必要があるかもしれません。

Magic xpi には、以下で説明するようなテキストを検索するための便利な機能があります。

特定のオブジェクトの名前をテキスト検索するには：

ビジネスプロセスやフロー、フロー内のステップ、およびデータマップノードの名前をテキスト検索することができます。

1. **編集 > 検索 > テキスト検索** を選択するか、**Ctrl+SHIFT+F** を押下してください。[テキスト検索] ダイアログボックスが開きます。



2. 検索したいテキストを入力してください。
3. どこで検索を実行するかを、[Look In] ドロップダウンリストから指定してください。
4. [Narrow the search to] チェックボックスを選択し、次にドロップダウンリストからオプションの1つを選択することで、検索条件を絞り込んでください。
5. さらに、[検索オプション] の下のチェックボックスのどちらかを選択して、検索条件を絞ってください。
6. [全て検索] をクリックしてください。結果は、[テキスト検索結果] ウィンドウに表示されます。ここから、直接、特定の結果項目をダブルクリックするか、その項目上で右クリックし、コンテキストメニューから[移動]を選択することで検索結果と関連した[プロパティ] ペインに移動することができます。
7. 次の検索項目に続くには、**F3**(または、**編集 > 次を検索**)を押下します。前の項目に戻る場合は、**Ctrl+F3**(**編集 > 前を検索**)を押下してください。

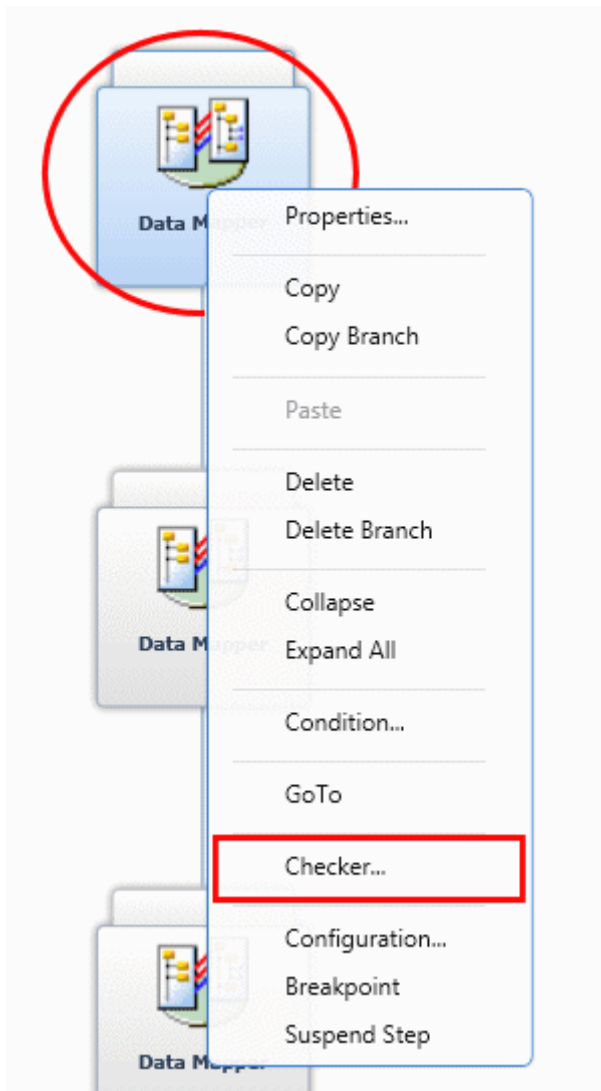
プロジェクトのエラーをチェックするには

Magic xpi で、デバッグ段階または運用段階にかかわらず、プロジェクトをビルドする時、プロジェクトは自動的にチェックされます。

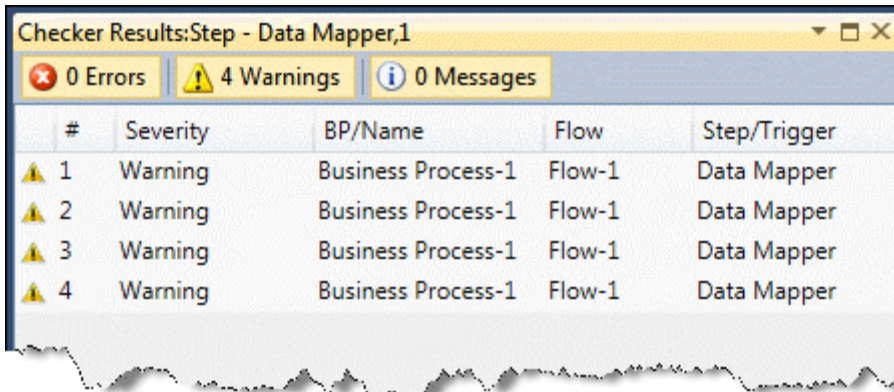
自身でチェックを実行させることもできます。Magic xpi プロジェクトの構文チェックに数秒かかるだけです。チェッカーツールは、例えば、設定されていないステップや互換性のないタイプを見つけます。構文チェックが完了すると、エラー(もしあれば)のリストが表示されます。エラーをクリックすると、そのエラー箇所に移動することができるので、それを修正することができます。

構文チェックを処理する様々なオプション(プロジェクト全体、ビジネスプロセス、特定のフロー、ステップ、またはトリガ)があります。

1つのアイテムの構文をチェックを行うには：

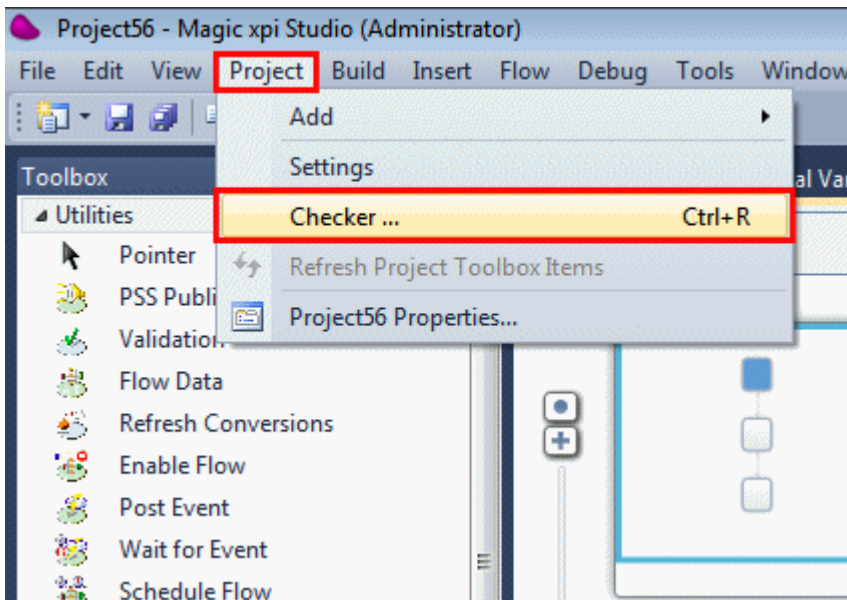


1. チェックしたい項目上でパークしてください。
2. 項目を右クリックし、コンテキストメニューから「チェッカ」を選択してください。
以下のどちらかになります。
 - 「フローに何も結果は発見されませんでした」というメッセージが表示されます。
 - 「チェッカ結果」画面が表示され、エラーメッセージが表示されます。

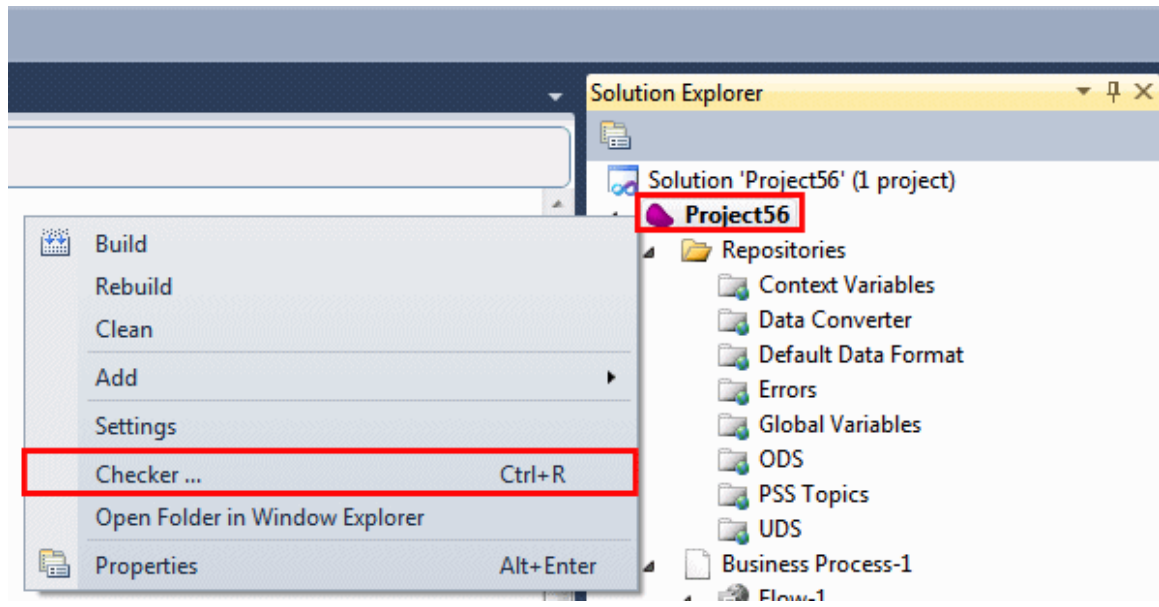


3. エラーがある場合、[チェック結果] リストを使用して、エラーのリストから修正作業を行うことができます。
4. [チェック結果] 画面から、画面のエントリをダブルクリックすることによって、問題が存在している **Magic xpi** 中のエリアに直接移動することができます。
5. 画面のトップでエラーや警告、および情報のボタンをクリックすることでどのエラーを表示するかをカスタマイズすることもできます。

プロジェクト全体をチェックするには：

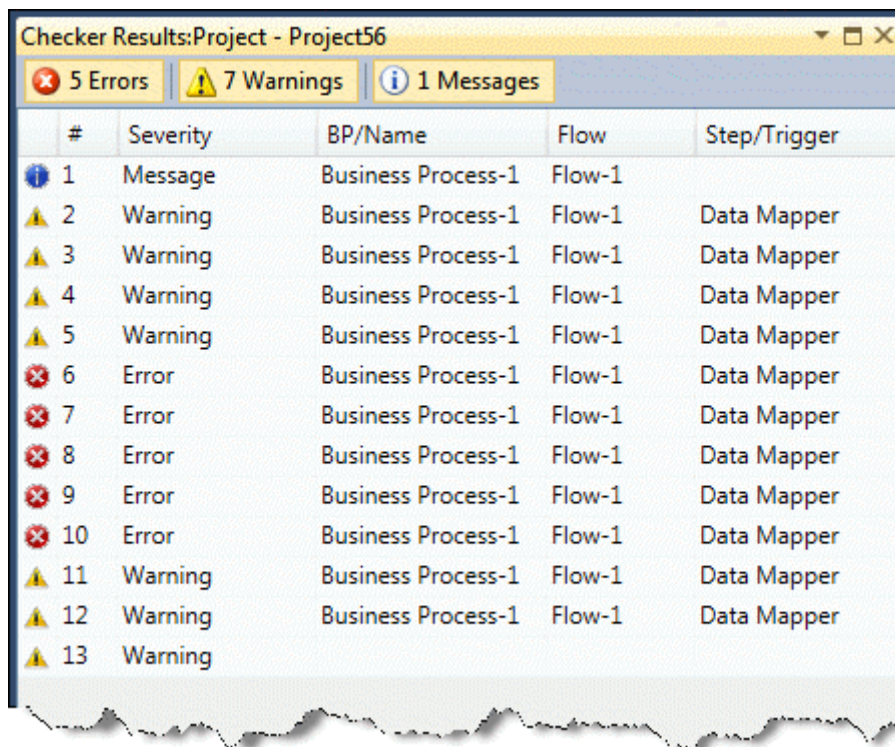


または：



1 度にプロジェクト全体をチェックすることができます。エラーメッセージは、すべて[チェッカー結果]画面にグループ化され、1 つのプログラムに対してチェックする場合と同じ方法でリストを通して修正作業を行うことができます。

1. プロジェクトの任意の場所から [プロジェクト] メニューを選択し、[チェッカ] を選択してください。



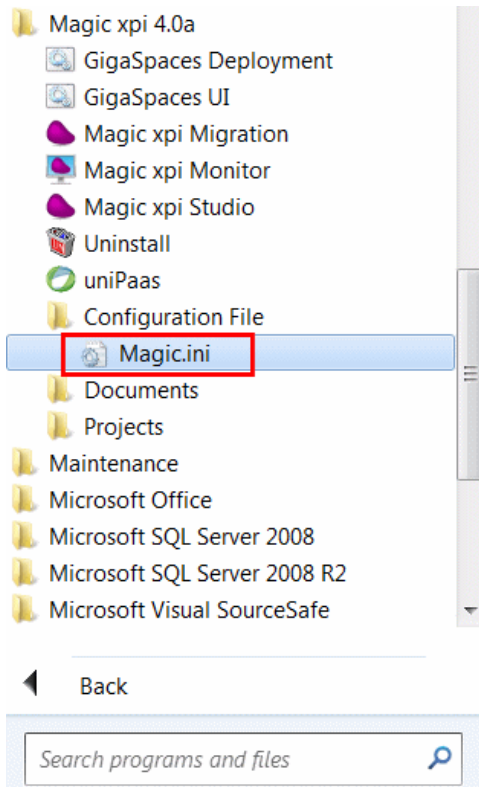
2. 構文チェッカーはプロジェクト全体で実行され、エラーは [チェッカ結果] 画面にリスト表示されます。
3. エラーがある場合、[チェッカ結果] リストを使用して、エラーのリストを通して修正作業を行うことができます。

4. [チェッカ結果] 画面から、画面のエントリをダブルクリックすることによって、問題が存在している **Magic xpi** の中のエリアに直接移動することができます。

ツールティップをより長く表示させるには

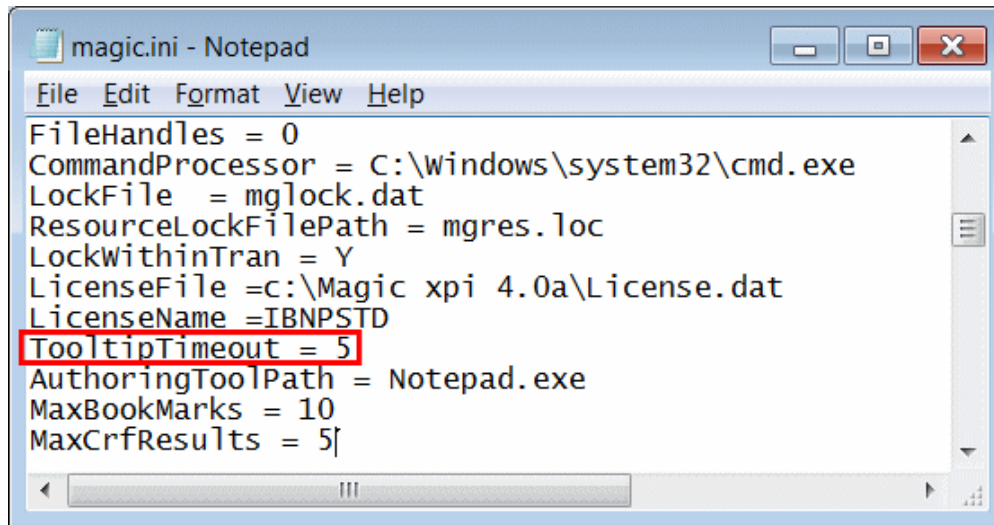
デフォルトでは、Magic xpi のツールティップは、5 秒間表示されるように設定されています。より長く表示させたい場合は、簡単に変更することができます。

Magic xpi のインストールフォルダにある **Magic.ini** を開くか、Windows の[スタート]メニューからショートカットを選択してください。



[MAGIC_ENV]セクションで、**TooltipTimeout** という名前のエントリが見つかります。その値が **5** に設定されることが確認できます。それは 5 秒を意味しています。それを **0** から **31** の間の任意の数値に変更することができます。これによって、ツールティップを **31** 秒までの表示させることができるようになります。変更をしたら、プロジェクトを再起動してください。そうすると、変更内容が反映されます。

値を **0** に設定した場合、ツールティップは表示されなくなります。



プロジェクトのデバッグ

プロジェクトのデバッグ

以下のトピックは、Magic xpi のデバッグ機能について扱っています。

[デバッガを使用して、プロジェクトをデバッグするには](#)

[ブレイクポイントをプロジェクトに設定するには](#)

[データベースゲートウェイによって記録された情報を制御するには](#)

[動的なゲートウェイログを作成するには](#)

[SAP R/3\(JCO\)コンポーネントのためにログファイルを作成するには](#)

[Magic xpi をデバッグするためにデバッグツールを使用するには](#)

[Magic xpi をデバッグするために User-Mode Dump Heap \(UMDH\)を使用するには](#)

[Magic xpi で .NET ユーティリティコンポーネントをデバッグするには](#)

デバッガを使用して、プロジェクトをデバッグするには

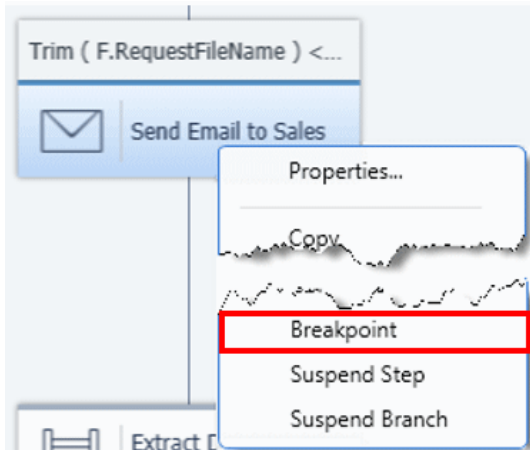
Magic xpi のデバッガは、開発段階の間、統合プロジェクトをテストするための強力なデバッグツールです。デバッガは、プロジェクトを実行したり、運用プロジェクトに付属して、プロジェクトがどのように実行されるかを制御します。実行順序やフロー変数、サービス、およびブレイクポイントをコントロールし表示します。デバッガのコンテキストツリーを見ることによって、サーバがどのようにプロジェクトでフローを実行しているかを確認できます。

このトピックは、デバッガをどのように使用するかについて説明しています。

ブレイクポイントを設定する

デバッガを実行する前に、どこで停止したらいいかを指定する必要があります。ブレイクポイントを設定することによって、これを行うことができます。これによって、特定のステップ上のフロー実行を停止させることができるので、ブレイクポイントはツールのデバッグに役立ちます。ブレイクポイントに達すると、次のステップが実行される前に、スレッドと分岐などのすべての実行されたコンテキストは停止します。

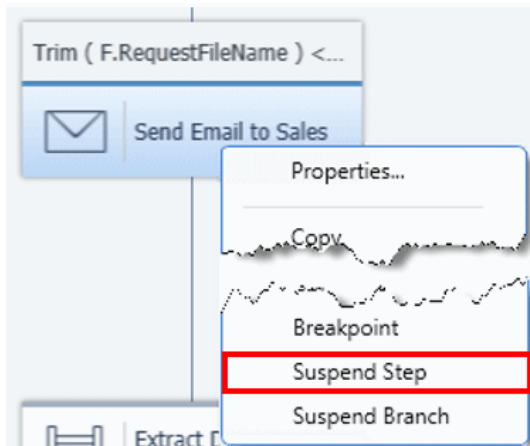
ブレイクポイントを設定するには、関連したステップで右クリックし、コンテキストメニューから[ブレイクポイント]を選択してください。




ステップ、フロー、またはブランチの一時中断

プロジェクトをデバッグする時、任意の問題に対してピンポイントで狙うことを容易にするために特定の要素を中断することができます。中断されたステップは無視され、ステップが実行されたかのように、バグは次のステップに続きます。中断されたフローは実行されず、そのモードから解放されるまで、「停止される」ように扱われます。

ステップ、フロー、または分岐を中断するために、そこで右クリックし、コンテキストメニューから[中断]を選択してください。



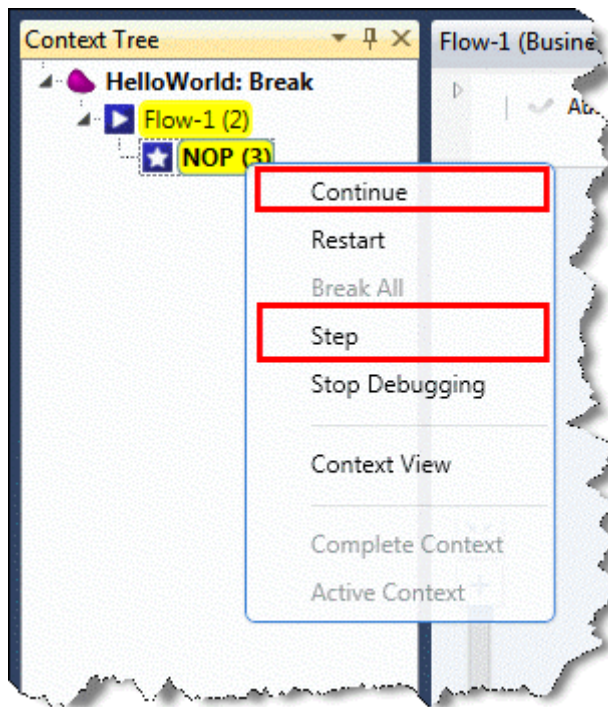
デバッガによる実行を開始する

さて、デバッガによる実行を開始する必要があります。[デバッグ xpi]メニューをクリックし、[デバッグ開始]を選択するか F5 を押下することで、プロジェクト全体をデバッグすることができます。ツールバー上の  アイコンをクリックすることもできます。

代わりに、関連するフローで右クリックし、コンテキストメニューから[デバッグ]を選択することで特定のフローをデバッグするよう設定できます。このように特定のフローをデバッグする時、プロジェクトの他のすべてのフローは無効になります。

プロジェクト内をステップ動作する

ブレイクポイントを設定し、デバッガを開始したら、プロジェクトの中で1つずつステップ実行を行うことができます。コンテキストツリーのステップで右クリックし、コンテキストメニューから[ステップ]または[継続]を選択することによって、これを行うことができます。



ここからコンテキストビューにアクセスすることもできます。コンテキストビューは、現在動作しているビューのために格納されたすべての内部情報を提供し、変数の値を変更することもできます。

デバッグ処理を終了する

デバッグを終了する場合は、[デバッグ xpi]メニューをクリックし[デバッグ停止]を選択するか、**Shift+F5**を押下することによって、簡単に処理を停止することができます。コンテキストツリー内の任意のステップで右クリックし、コンテキストメニューから[デバッグ停止]を選択することもできます。

ブレイクポイントをプロジェクトに設定するには

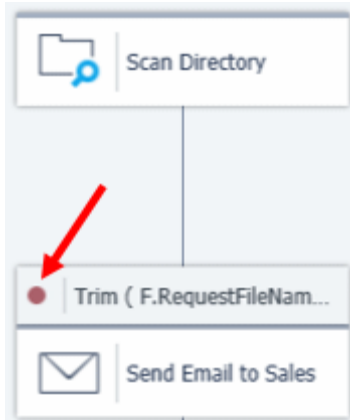
ブレイクポイントは、選択したステップ上のフローの実行を停止させることができ、デバッグのために非常に便利なツールです。

ブレイクポイントに達したら、次のステップを実行する前に、スレッドと分岐などの実行されたコンテキストのすべては停止します。コンテキストツリーから[ステップ]または[継続]のどちらかを選択するまで、デバッグはその時点で待機します。

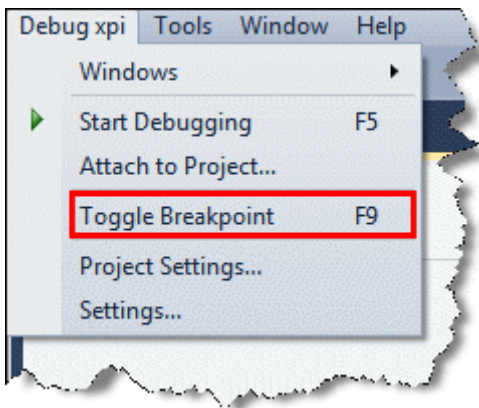
ブレイクポイントの設定または削除するには：

1. ブレイクポイントを特定のステップに設定するには、そこで右クリックし、コンテキストメニューから [ブレイクポイント] を選択してください。
2. ブレイクポイントを特定のステップから削除するには、そこで右クリックし、コンテキストメニューから [ブレイクポイント] を選択してください。

ステップ上にブレイクポイントが定義されている場合、ブレイクポイントのイメージ(●)が隣に表示されます。以下のイメージのように、ブレイクポイントとしてマークされたステップが表示されます。



[デバッグ xpi]メニューからアクセスされた[ブレイクポイント設定/解除]オプション(以下のイメージを参照)は、現在パークしているステップからブレイクポイントを追加したり、削除することができます。ブレイクポイントがステップ上に存在していない場合、ブレイクポイントが作成されます。ブレイクポイントが存在していて、現在使用可能な場合、それは削除されます。



データベースゲートウェイによって記録された情報を制御するには

ゲートウェイログ

特定のデータマップのステップをデバッグするか、一定の間隔でログの動作をおこなうようにに指示するなどの場合に、動的にゲートウェイログを使用可能にしたり無効にするために、**Logging** 関数を使用することができます。

これをどのように行うかについての詳細は、[ここ](#)をクリックしてください。

注 意：

- デバッグ段階が完了しアプリケーションを配備する前に、ロギングを停止することを推奨します。ロギングは、プログラムの実行速度を低下させます。
- デバッグ段階で失敗するすべてのオペレーションはまたは動作は、**MagicXPI.Log** ファイルに記録されます。メッセージボックスで、要求された処理で発生したエラーが表示されます。どのようなエラーが発生したか、そしてどこで発生したかを確認するには、ログを参照してください。

動的なゲートウェイログを作成するには

場合によっては、データベースと接続する特定のデータマッピングステップをデバッグする必要があるったり、一定の間隔で実行するためにログが必要になるかもしれません。これらのインスタンスで、動的にゲートウェイログを使用可能にしたり、無効にするために **Logging** 関数を使用することができます。これは **Magic.ini** ファイルの中では ロギング設定を変更することができません。

構文は以下の通りです。

Logging(*startstop*, *filter*)

パラメータ:

- **startstop** 'True'Log の場合、ロギングを開始します。
- **filter** どのロギングエントリを開始/停止するかを指定します。

いくつかの例があります:

Logging 式	効果
Logging('True'Log, 'Levels')	Levels に対してロギングを開始します。 ,
Logging('False'Log, 'Recompute')	Recompute の動作に対してロギングを停止します。
Logging('False'Log, 'ALL')	すべてのロギングをを停止します。
Logging('True'Log, 'RESET')	Magic.ini ファイルに格納された値にロギングを戻します。
Logging('True'Log, 'Oracle=D')	Oracle のロギングを開発者レベルに設定します。

動的なゲートウェイログを作成するには、以下の手順に従ってください。

1. 論理タイプの変数を定義してください。
2. 以下の行を含むフローデータステップを追加してください。
 - a. 変数を以下の式で更新します **Logging ('TRUE'LOG,'MicrosoftSQLServer=D')**. この設定は **MSSQL** 用です。別のデータベースタイプが必要な場合、それに応じて名前を変更してください。
 - b. 変数を以下の式で更新します **Logging ('TRUE'LOG,'Gateway')**.
 - c. 変数を以下の式で更新します **Logging ('TRUE'LOG,'ExecutionLogFileName=< file name >')**.
パスが指定されていない場合、ファイルは<Magic xpi インストール>%Runtime フォルダに作成されます。
3. ゲートウェイログを停止するには、別のフローデータステップを作成し、論理変数を定義し、次の式で更新してください: **Logging ('FALSE'LOG,'MicrosoftSQLServer=D')**.

SAP R/3(JCO)コンポーネントのためにログファイルを作成するには

デフォルトでは、SAP R/3(JCO)コンポーネントのために作成されるログファイルはありません。

そのようなログを作成するには、**log4j.xml** ファイルを修正する必要があります。これは **%java%classes%** フォルダにあります。

log4j.xml ファイルを開き、以下の 2 つのエントリの "level value" を変更してください。

```
<logger name="com.magicsoftware.ibolt.sapr3.IBSapR3" additivity="false">
    <level value="off"/>
    <appender-ref ref="ifc.sapr3"/>
</logger>
```

そして:

```
<logger name="com.magicsoftware.ibolt.sapr3.IBSapR3Server" additivity="false">
    <level value="off"/>
    <appender-ref ref="ifc.sapr3Trigger"/>
</logger>
```

エントリは以下のどれかの値に変更することができます。

- all
- debug
- info
- warn
- error
- fatal
- off
- null

(通常、**debug** を選択することで十分です。)

注 意:	このログは、Magic xpi サーバ側に作成されます。これは、SAP R/3 のトレースファイルではありません。SAP R/3 トレースを有効にするには、SAP R/3 リソースに移動し、[SAP トレース] パラメータの値を Yes に変更してください。結果として生じるトレースファイルは、Magic xpi のインストールフォルダに作成されます。
-----------------------	---

Magic xpi をデバッグするためにデバッグツールを使用するには

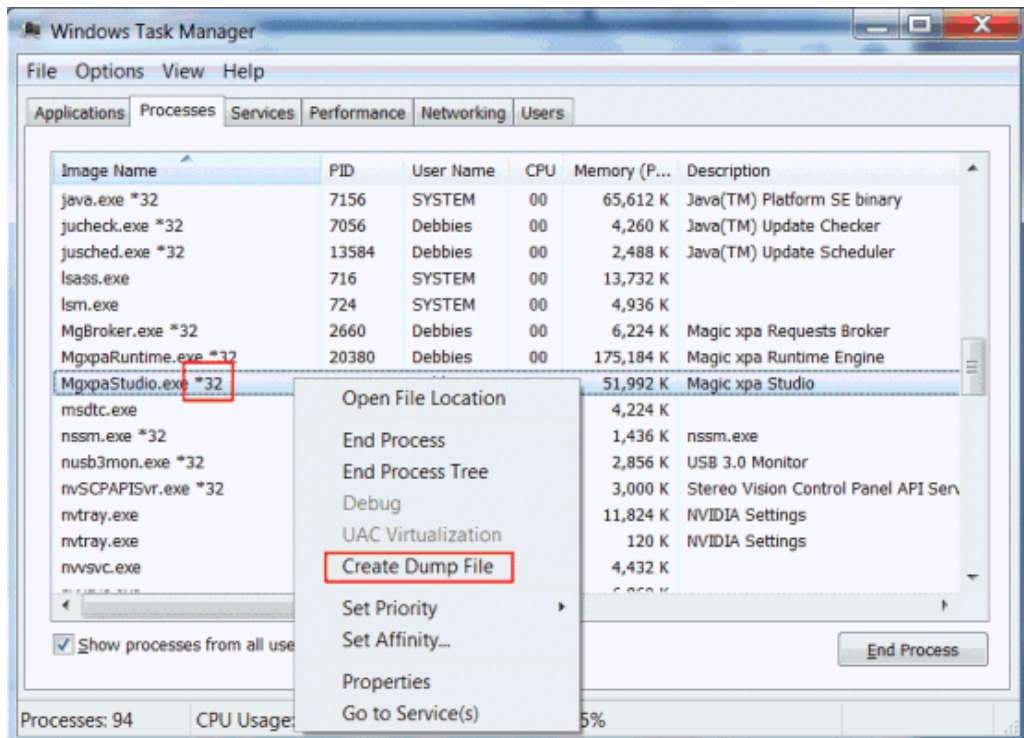
Magic xpa または Magic xpi のエンジンをトラブルシューティングする必要がある場合、問題が再現できない時、ログファイルが必要になるかもしれません。Magic xpa のテクニカルサポート部門がそれらを要求した場合、このトピックは、Magic xpa(uniPaaS)または Magic xpi(jBOLT)のために、ダンプログファイルを作成するために、デバッグツールをどのように使用するかについて説明しています。

64 ビットの Windows OS 上で 32 ビットのダンプファイルを作成する

以下のどれかの方法で、64 ビット OS 上で 32 ビットのダンプを作成することができます。

1. **タスクマネージャを使用する:** タスクマネージャからダンプを作成することが可能です。しかし、32 ビット版のタスクマネージャを使用する必要があります。
 - a. トラブルシューティングしたいプログラムを実行してください。
 - b. **C:\Windows\SysWOW64\taskmgr.exe** からタスクマネージャを開始してください。これは、32 ビット版のタスクマネージャです。タスクマネージャの 32 ビット版を実行していることを確認したい場合、**taskmgr.exe** がタスクマネージャ自身で*32 とリスト表示されていることでチェックすることができます。以下のイメージに表示されているように、プロセス名の隣の*32 は、プロセスが 32 ビットアーキテクチャを使用していることを示しています。

- c. トラブルシューティングしたいプロセス上で右クリックして [ダンプファイルの作成] を選択してしてください。



2. ProcDump ユーティリティを使用する :

- a. **procdump.exe** ファイルを次のサイトからダウンロードしてください :
<http://technet.microsoft.com/en-us/sysinternals/dd996900.aspx>. インストールする必要はありません。ダウンロードして実行するだけです。
- b. コマンドラインから以下の構文を使用して実行してください。

```
procdump -e -ma [プロセス名] <ダンプファイルのフルパス>
```

または

```
procdump -e -ma <PID> <ダンプファイルのフルパス>
```

-e = 処理されない例外を待ちます

-ma = ダンプファイル名

Microsoft の WinDbg ツールを使用してダンプファイルを作成する



WinDbg を使用して Magic xpa をデバッグするビデオデモンストレーションを参照する場合は、[ここ](#)をクリックしてください。

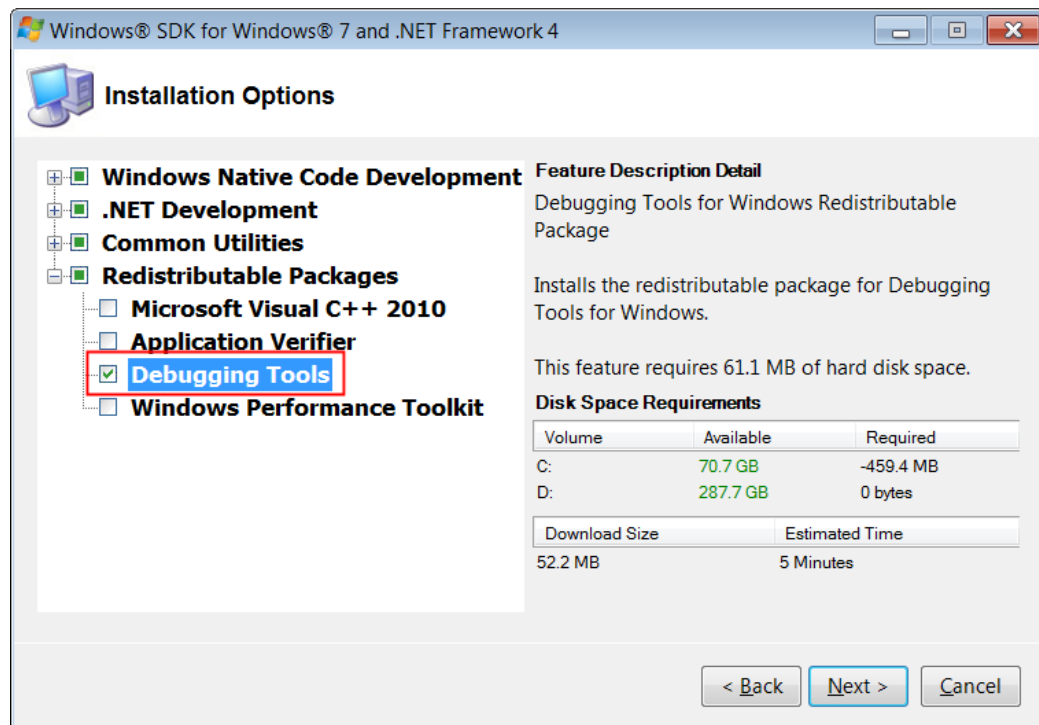
WinDbg ツールをインストールし設定する

Microsoft の Windows デバッガ(WinDbg)を使用するには、Magic xpi がインストールされた PC で、以下を実行してください。

1. **C ドライブ**(C ドライブ上になければならないことに注意してください)上に、**mgdbg** と呼ばれる新規フォルダを作成してください。
2. **Debugging Tools for Windows** フォームからダウンロードして実行してください。

<http://www.microsoft.com/en-us/download/confirmation.aspx?id=8279>

3. Windows SDK セットアップウィザードから、[Next] ボタンをクリックして [Installation Options] ダイアログに移動し、[Debugging Tools] チェックボックスをチェックし、他のオプションのチェックをすべてを外します。インストール処理が終了するまで [Next] ボタンをクリックして、[Finish] をクリックしてください。
すでに Windows SDK がインストールされている場合は、このダイアログのようにメンテナンスオプションを選択してください。



4. それを保存したら、プロセスがかなり大きなサイズの複数のファイルを作成するため、ハードディスク上の **C** ドライブに十分な空きスペースがあることを確認してください。
Magic xpa のバージョンに対応した WinDbg の内部構成ファイルをダウンロードするため、以下のリンクを使用してください。

Ver9 に対応したファイルは、[ここ](#)をクリックしてください。

Ver10、uniPaaS、および Magic xpa に対応したファイルは、[ここ](#)をクリックしてください。

jBOLT と Magic xpi に対応したファイルは、[ここ](#)をクリックしてください。

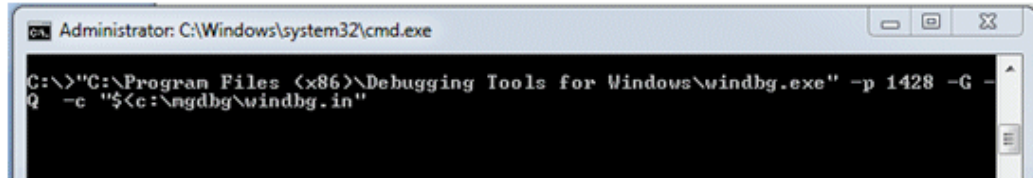
5. ファイルを解凍し、ファイルを **windbg.in** にリネームし、それを **c : \mgdbg** フォルダに保存してください。

オプション 1 : CMD を使用してダンプファイルを作成する

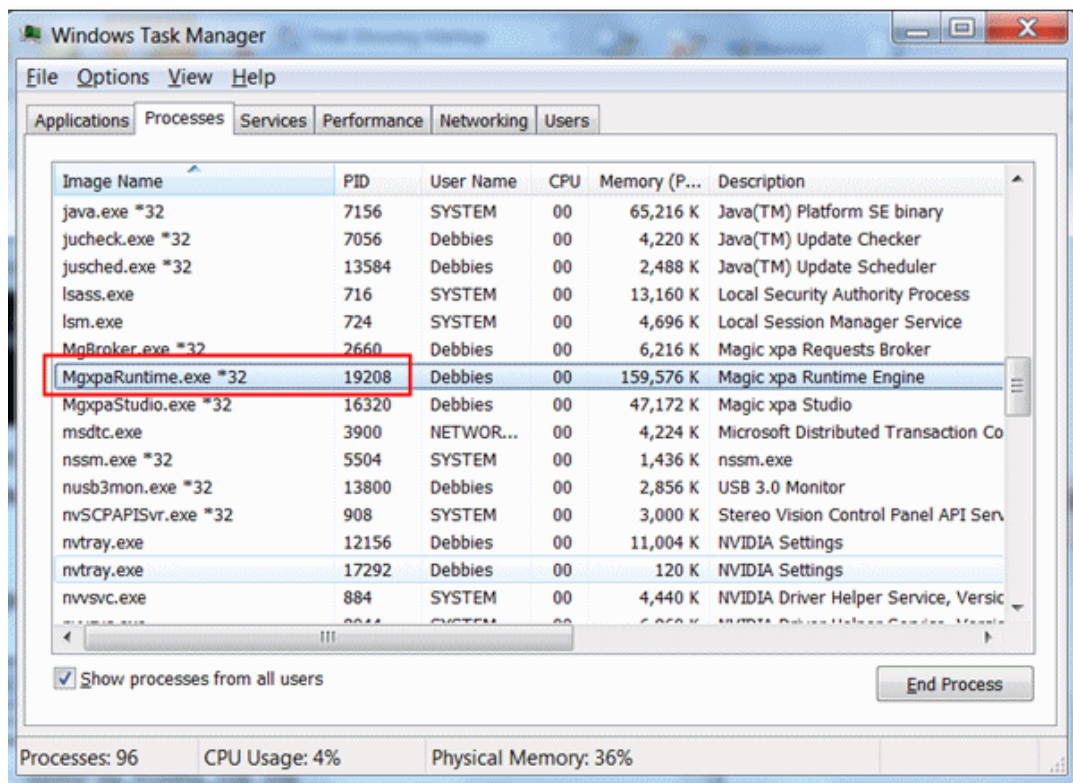
Magic xpi 逆引き辞典

1. 管理者権限で CMD ウィンドウを開いてください。
2. CMD ウィンドウで以下の構文を使用してください。

[windbg.exe のフルパス名] -p [プロセス id] -G -Q -c "\$<c:\mgdbg\windbg.in"



[プロセス id] は、Windows のタスクマネージャの[プロセス]タブに表示される Magic xpa 実行エンジンのプロセス ID です。



デバッガウィンドウが開きます。

```

0:012> * The command to run :
0:012> *windbg -p [decimal untrte.exe pid] -G Q -c "<c:\mgdbg\windbg.in"
0:012> *open a log
0:012> .logopen /t /u c:\mgdbg\ag_windbg_log.log
Opened log file 'c:\mgdbg\ag_windbg_log_1990_2013-01-14_21-20-12-373.log'
0:012> *reset all exceptions to default
0:012> *srx
sx state reset to defaults
0:012> SXI *
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" av
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" da
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" dz
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" gp
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" ii
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" iov
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" ip
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" isc
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" sbo
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" sov
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" aph
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" ch
0:012> g

```

3. トラブルシューティングしたい Magic xpa アプリケーションを実行してください。
4. エンジンがクラッシュしたら、以下のイメージに示されているように「**Dump successfully written**」という表示を確認してください。これは、ダンプファイルを初期に作成したフォルダに作成します。この場合、C ドライブ上の mgdbg フォルダになります。

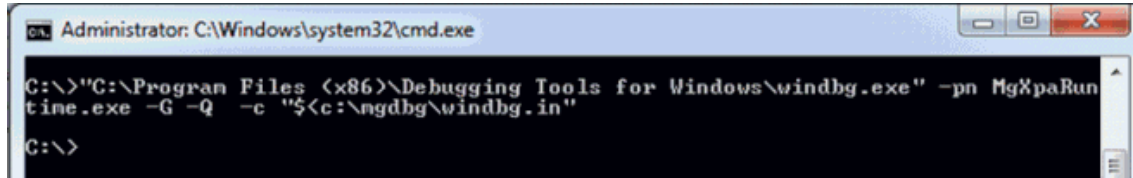
```

0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" ip
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" isc
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" sbo
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" sov
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" aph
0:012> SXE -c ".dump /ma /u c:\\mgdbg\\ag_crsh.dap:gn" ch
0:012> g
ModLoad: 1ed90000 1eeec000 C:\Windows\SysWOW64\ole32.dll
ModLoad: 76030000 760b3000 C:\Windows\syswow64\CLBCatQ.DLL
ModLoad: 20020000 201ca000 C:\Program Files (x86)\Chilkat Software Inc\Chilkat F
ModLoad: 203f0000 205dd000 C:\Windows\SysWow64\ComEm1.dll
ModLoad: 73d10000 73d26000 C:\Windows\SysWOW64\CRYPTSP.dll
ModLoad: 73cd0000 73d0b000 C:\Windows\SysWOW64\rsaenh.dll
(14bc.12d8): Access violation - code c0000005 (first chance)
Dump successfully written
(14bc.12d8): Access violation - code c0000005 (!!! second chance !!!)
eax=00000000 ebx=0028d614 ecx=00d79640 edx=fffffe4 esi=0028de74 edi=0028dfe8
eip=77c815de esp=0028d600 ebp=0028dad0 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ntdll!ZwRaiseException+0x12:
77c815de 83c404         add     esp,4
0:000> |g

```

5. PC 上で 1 つの Magic xpi エンジンしか実行していない場合は、以下の構文を使用することもできます。

[Full pathname of windbg.exe executable] -pn [.exe name] -G -Q -c "\$c:\mgdbg\windbg.in"



.exe 名は以下の通りです。

mgrntw.exe Ver9
eDevRTE.exe Ver10
uniRTE.exe Magic xpa
MgxpaRuntime.exe Magic xpi
iBoltServer.exe jBOLT versions
MgxpiServer.exe Magic xpi versions

.exe 名/プロセス ID を変更することで、開発環境のクラッシュを捕らえることもできます。

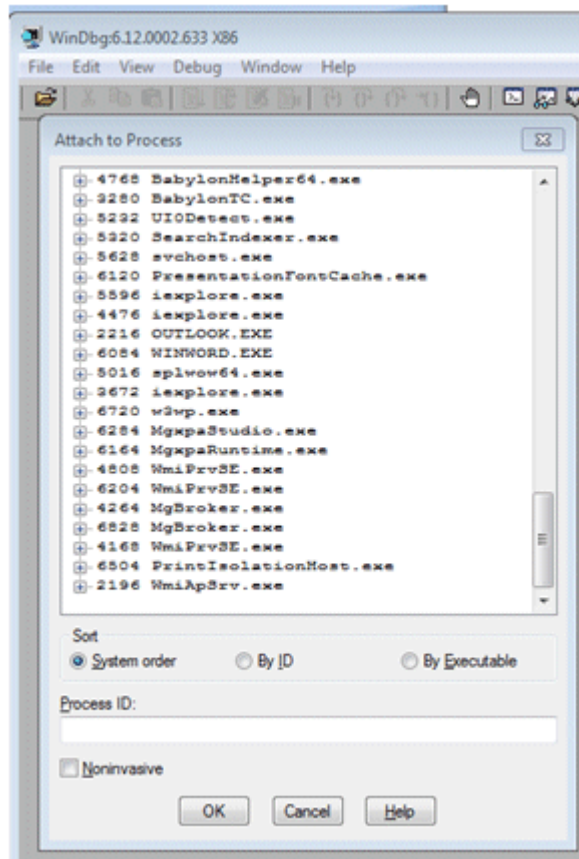
重 要：	マルチスレッドの Magic xpa エンジンデバッグするためには、ダンプファイルを作成するために、1 つのスレッドのクラッシュによって実行エンジン全体がクラッシュすることを確認する必要があります。これは、 Magic.ini ファイルの定義を使用しています。 [MAGIC_SPECIALS] ExceptionMessageBoxDisplay = Y
-----------------	---

オプション 2 : WinDbg から直接ダンプファイルを作成する

コマンドラインにアクセスする権限がない場合、次の通りに直接 WinDbg ツールでダンプファイルを作成することができます。

1. アプリケーションをロードする前に、**WinDbg** ユーティリティ(管理者として)を実行してください : **スタート > すべてのプログラム > Debugging Tools for Windows > WinDbg**。

2. [ファイル] メニューから、[Attach to a Process] オプションを選択し、Magic xpa エンジンにアクセスします。

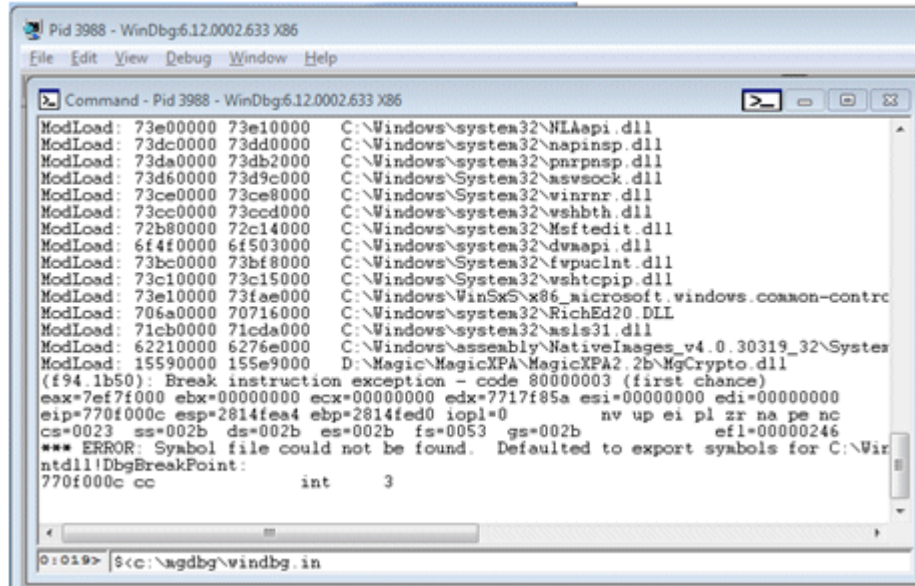


注意：アルファベット順に並べられているため、実行可能な名前でプロセスをソートすることができます。

3. デバッグしたい.exe が実行していることを確認してください。
4. プロセスリストから、デバッグしたい.exe ファイルを選択するか、プロセス ID を入力して、[OK] をクリックしてください。
5. 再度、トラブルシューティングしたいアプリケーションを開き、実行してください。

Magic xpi 逆引き辞典

6. .exe ファイルがクラッシュしたら、コマンド `$c:\mgdbgwindbg.in` を入力し、**Enter** を押下してください。これによって、**mgdbg.in** のスクリプトが読み込まれ、ダンプファイルが作成されます。

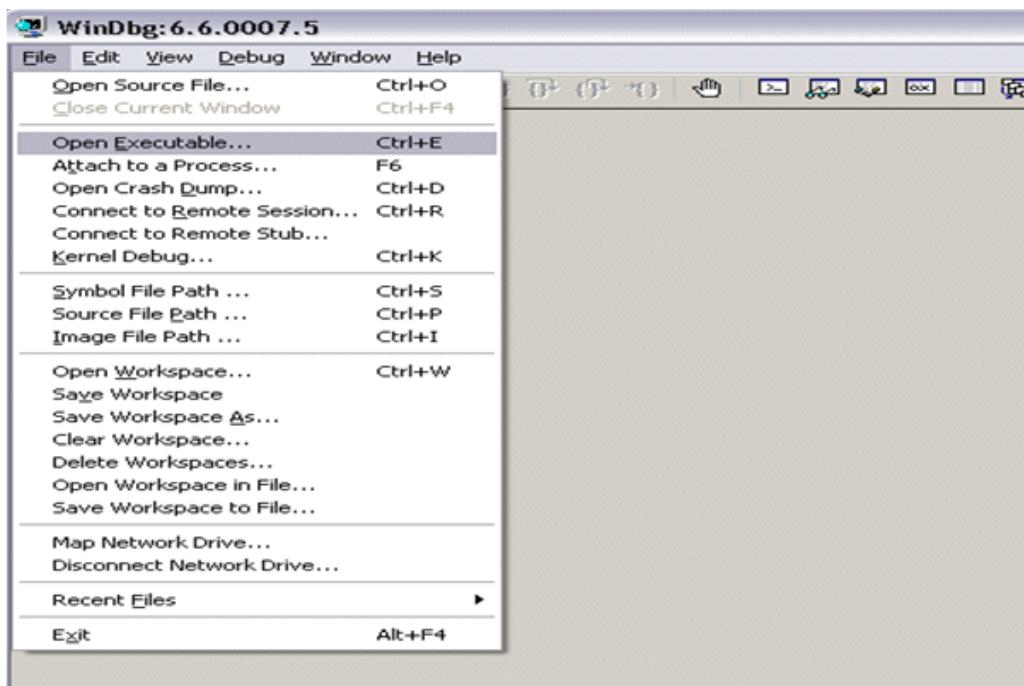


起動時にエンジンがクラッシュした場合にダンプファイルを作成する

起動時に実行エンジンがクラッシュする場合があり、Windows デバッガを既存のプロセスに割り当てることは不可能です。

この場合に、WinDbg ツールで以下のようにします。

1. [ファイル] メニューから、Magic xpa エンジンをロードするために [Open Executable] オプションを選択してください。



2. 実行エンジンの .exe ファイルを選択し、開いてください。
3. .exe ファイルがロードされたら、コマンド `$<c:\mgdbg\windbg.in` を入力してしてください。

これは、**mgdbg.in** でスクリプトを読み込みます。

注意：すでに一度ツールを実行した場合、パスを入力することなく **g** という文字を入力し、**Enter** を押下するだけですみます。

```

Command - D:\Magic\MagicXPA\MagicXPA2.2b\MgxpRuntime.exe - WinDbg.6.12.0002.633 X86
CommandLine: D:\Magic\MagicXPA\MagicXPA2.2b\MgxpRuntime.exe
Symbol search path is: *** Invalid ***
*****
* Symbol loading may be unreliable without a symbol search path.
* Use .symfix to have the debugger choose a symbol path.
* After setting your symbol path, use .reload to refresh symbol locations.
*****
Executable search path is:
ModLoad: 00200000 00266000 MgxpRuntime.exe
ModLoad: 77c60000 77de0000 ntdll.dll
Page heap: pid 0x7D4: page heap enabled with flags 0x3.
ModLoad: 74fb0000 75010000 C:\Windows\syswow64\verifier.dll
Page heap: pid 0x7D4: page heap enabled with flags 0x3.
ModLoad: 70c50000 70c9a000 C:\Windows\SysWOW64\MSCOREE.DLL
ModLoad: 75c70000 75d80000 C:\Windows\syswow64\KERNEL32.dll
ModLoad: 75fd0000 76017000 C:\Windows\syswow64\KERNELBASE.dll
(7d4.1ae4): Break instruction exception - code 80000003 (first chance)
eax=00000000 ebx=00000000 ecx=85060000 edx=0016dc78 esi=ffffffe edi=00000000
eip=77d00fab esp=0047f648 ebp=0047f674 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
*** ERROR: Symbol file could not be found. Defaulted to export symbols for ntdll.dll
ntdll!LdrVerifyImageMatchesChecksum+0x96c:
77d00fab cc                int     3
0:000> $<c:\mgdbg\windbg.in

```

4. **Enter** をクリックするとデバッグが可能になります。
5. クラッシュする瞬間まで、アプリケーションを実行してください。「**Dump successfully**」というメッセージが表示され、ダンプファイルが **mgdb** フォルダに作成されます。

```

Command - Pid 5288 - WinDbg.6.12.0002.633 X86
ModLoad: 63550000 6364b000 C:\Windows\SysWOW64\WindowsCodecs.dll
ModLoad: 76030000 760b3000 C:\Windows\syswow64\CLBCatQ.DLL
ModLoad: 0ee30000 0efda000 C:\Program Files (x86)\Chilkat Software Inc\Chilkat F
ModLoad: 0f0e0000 0f2cd000 C:\Windows\SysWOW64\ComEm1.dll
(14a8.1924): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=0678d841 ebx=00000000 ecx=ffffffe4 edx=0a3ae29c esi=07cc44c8 edi=0c4fd73c
eip=060bb25b esp=0c4fd020 ebp=0c4fd034 iopl=0         nv up ei ng nz ac po cy
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00010293
*** ERROR: Symbol file could not be found. Defaulted to export symbols for D:\Mag
MgCore!get_ctrl_hwnd_by_name+0x14febf:
060bb25b 0fb6510c          movzx   edx,byte ptr [ecx+0Ch]      ds:002b:ffffff0=??
0:010> dump (c:\mgdbg\mg_crsh_1f78_2013-01-14_22-16-14-775_14a8.dmp)
Creating c:\mgdbg\mg_crsh_1f78_2013-01-14_22-16-14-775_14a8.dmp - mini user dump
Dump successfully written
eax=00000000 ebx=0c4fcb6c ecx=ffffffe4 edx=0a3ae29c esi=07cc44c8 edi=0c4fd73c
eip=77c815de esp=0c4fcb58 ebp=0c4fd034 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ntdll!ZwRaiseException+0x12:
77c815de 83c404          add     esp,4
0:010>

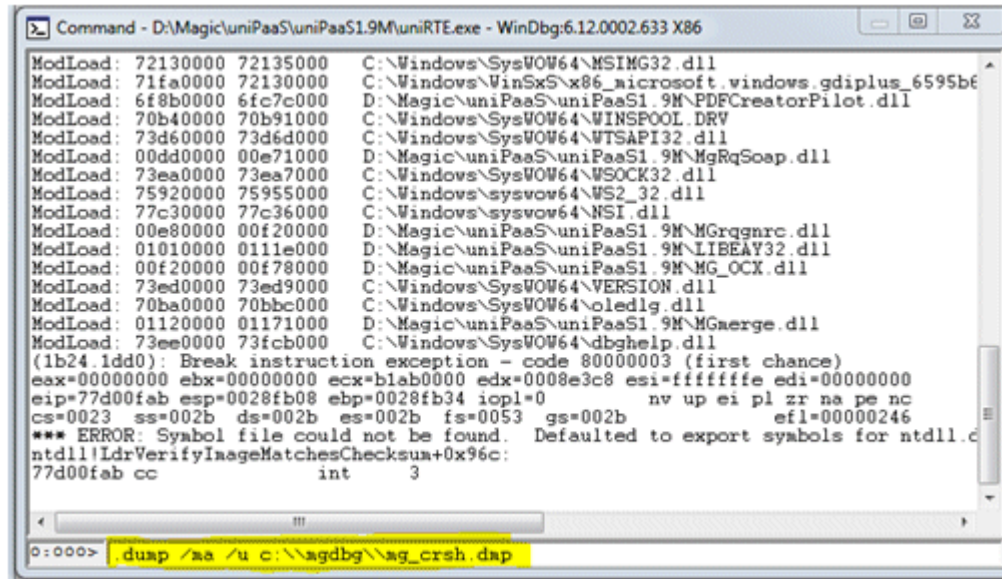
```

Magic xpi 逆引き辞典

全くクラッシュがない時にダンプファイルを作成する

異常終了が全くない場合:

1. **windbg.in** ファイルを開いてください。
2. **.dump** で始まる行ををコピーしてください。例えば: **.dump /ma /u c:\mgdbg\mg_crsh.dmp;gn.**
3. [コマンドプロンプト] 画面の下にそれを貼り付けて、**Enter** を押下してください。



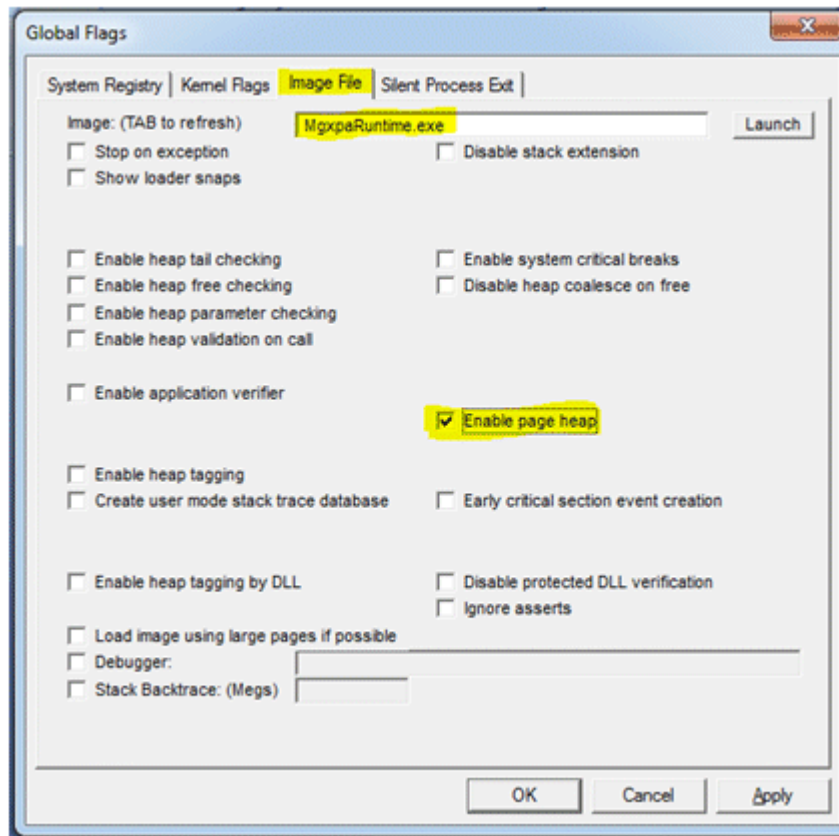
```
Command - D:\Magic\uniPaaS\uniPaaS1.9M\uniRTE.exe - WinDbg:6.12.0002.633 X86
ModLoad: 72130000 72135000 C:\Windows\SysWOW64\MSIMG32.dll
ModLoad: 71fa0000 72130000 C:\Windows\WinSxS\x86_microsoft.windows.gdiplus_6595b6
ModLoad: 6f8b0000 6fc7c000 D:\Magic\uniPaaS\uniPaaS1.9M\PDFCreatorPilot.dll
ModLoad: 70b40000 70b91000 C:\Windows\SysWOW64\WINSPOOL.DRV
ModLoad: 73d60000 73d6d000 C:\Windows\SysWOW64\WTSAPI32.dll
ModLoad: 00dd0000 00e71000 D:\Magic\uniPaaS\uniPaaS1.9M\MgRqSoap.dll
ModLoad: 73ea0000 73ea7000 C:\Windows\SysWOW64\WSOCK32.dll
ModLoad: 75920000 75955000 C:\Windows\syswow64\WS2_32.dll
ModLoad: 77c30000 77c36000 C:\Windows\syswow64\NSI.dll
ModLoad: 00e80000 00f20000 D:\Magic\uniPaaS\uniPaaS1.9M\MGrqgnrc.dll
ModLoad: 01010000 0111e000 D:\Magic\uniPaaS\uniPaaS1.9M\LIBEAY32.dll
ModLoad: 00f20000 00f78000 D:\Magic\uniPaaS\uniPaaS1.9M\MG_OCX.dll
ModLoad: 73ed0000 73ed9000 C:\Windows\SysWOW64\VERSION.dll
ModLoad: 70ba0000 70bbc000 C:\Windows\SysWOW64\oledlg.dll
ModLoad: 01120000 01171000 D:\Magic\uniPaaS\uniPaaS1.9M\MGmerge.dll
ModLoad: 73ee0000 73fcb000 C:\Windows\SysWOW64\dbghelp.dll
(1b24.1dd0): Break instruction exception - code 80000003 (first chance)
eax=00000000 ebx=00000000 ecx=blab0000 edx=0008e3c8 esi=ffffffe edi=00000000
eip=77d00fab esp=0028fb08 ebp=0028fb34 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
*** ERROR: Symbol file could not be found.  Defaulted to export symbols for ntdll.d
ntdll!LdrVerifyImageMatchesChecksum+0x96c:
77d00fab cc                int     3
0:000> .dump /ma /u c:\mgdbg\mg_crsh.dmp
```

グローバルなフラグ

ダンプファイルに(クラッシュの原因を理解するために)必要な情報が保存されない場合があります。従って、グローバルなフラグ(gflags)情報を保持するダンプファイルを作成するように要求されます。その他の情報を作成するために、**Debugging Tools for Windows**を設定するために、以下のステップを実行してください。(注:グローバルフラグを設定した後で、.exe ファイルを実行するようにしてください。)

1. **Global Flags** ユーティリティを実行してください: (スタート > すべてのプログラム > **Debugging Tools for Windows** > **Global Flags**)
2. **Yes** クリックしてください。
3. フォーカスを **Image File** と呼ばれる 3 番目のタブに置いてください。
4. デバッグしたい.exe ファイルの名前を入力してください。フルパスではなく **image name**(以下に表示されているような)のみを使用してください。
5. [Tab] ボタンを押下して、画面の上の **Option** を有効にしてください。

6. [Enable page heap] チェックボックスとテクニカルサポートから選択するように指示された他のすべてのチェックボックスを選択してください。



重 要 : メモリの不要な消費がないように、ダンプファイルが作成されたら、**グローバルフラグ**の定義を削除してください。再度ツールを開き、[Enable page heap] チェックボックスをクリアすることで、これを行うことができます。

WinDbg をデフォルトの Windows の 事後分析デバッガとして設定する方法

WinDbg はデフォルトのデバッガとして動作しないため、端末セッションでユーザダンプをキャプチャすることは困難です。このような場合、以下の方法でダンプファイルを作成することができます。

この方法は、様々なアプリケーションのどのようなクラッシュに対しても自動的にダンプファイルを作成します。

この方法は、レジストリの変更を必要としています。他に選択肢がない限りは使用しないでください。

1. 最新の **Debugging Tools for Windows** をインストールしてください : :
<http://www.microsoft.com/whdc/devtools/debugging/default.mspx>.
2. 以下のコマンドを実行して、WinDbg をデフォルトデバッガとして設定してください : : **WinDbg -I**

注意 : **I** は大文字で入力してください。

3. ダンプが保存されるフォルダを作成し、ユーザとリモートデスクトップユーザに対しフルコントロールの権限を与えてください。

この例では、**c:\TEMP** フォルダが使用されます。

4. 以下のキーを探し、まだない場合は追加してください：

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\AeDebug

例えば、それは以下の値を持っています：

"C:\Program Files\Debugging Tools for Windows\windbg.exe" -p %ld -e %ld -g

現在のキーの値を取得し、以下の文字列を追加してください：

-c '.dump /o /f c:\TEMP\new.dmp; q' -Q -QS -QY -QSY

新しいキーは、以下の値を設定する必要があります：

"C:\Program Files\Debugging Tools for Windows\windbg.exe" -p %ld -e %ld -g -c '.dump /o /f c:\TEMP\new.dmp; q' -Q -QS -QY -QSY

64 ビットの Windows 上で、64 ビットの **"Debugging Tools for Windows"** を使用しています。しかし、32 ビットプロセス(タスクマネージャで ***32** として表示されます)からダンプを保存したい場合、**Wow6432Node** のレジストリーハイクを変更する必要があります。32 ビット版の **"Debugging Tools for Windows"** から 32 ビット版の **WinDbg.exe** ファイルを使用する必要があります。

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\AeDebug

セッション中に例外がある時は常に、ダンプが **Temp** フォルダに作成されます。

注意： 以前に保存されたダンプは上書きされます。

5. Windows Server2003、Windows XP 以降の OS 上では、ハンドルデータとスレッド時刻の情報などの追加のデバッグ情報を保存するために **/f** の代わりに **/ma** スイッチを使用することができます。

"C:\Program Files\Debugging Tools for Windows\windbg.exe" -p %ld -e %ld -g -c '.dump /o /ma c:\TEMP\new.dmp; q' -Q -QS -QY -QSY

6. ダンプがまだ発生しない場合、ローカルセキュリティポリシーにおいて [詳細追跡] を有効にしてください。そして、事後分析デバッグ起動されたことを確認するために、セキュリティイベントログのプロセス作成イベントを捜してください。

注意： この方法は PC 上の様々なクラッシュに対するダンプファイルを作成し、**c : \TEMP** フォルダへの既存のファイルを上書きします。

事後分析デバッガとして WinDbg を削除する

1. [スタート] を選択し、[ファイルを指定して実行] をクリックしてください。
2. レジストリエディタを開くために、**RegEdit** を入力してください。

3. **HKEY_LOCAL_MACHINE** を展開し、**Software** に移動してください。 **Microsoft > WindowsNT > Current Version > AeDebug** をクリックしてください。
4. ウィンドウの右側のペインで、**Debugger** という名前のレジストリキーをダブルクリックしてください。
5. エントリを、"**drwtsn32 -p %ld -e %ld -g**" に変更します。クォーテーションマークを含めて、事後分析デバッガをデフォルトに戻します。
6. **[OK]** をクリックし、レジストリエディタを閉じてください。

関連トピック：

[Magic xpi をデバッグするために User-Mode Dump Heap \(UMDH\)を使用するには](#)

Magic xpi をデバッグするために User-Mode Dump Heap (UMDH)を使用するには

メモリリークは、トレースしづらいかもしれません。Microsoft は、開発者に、特定のプロセスの Windows ヒープ アロケーションを分析するように、OS を使用して動作する便利なユーティリティを提供しました。

UMDH(User-Mode Dump Heap)と呼ばれるこのユーティリティは、Windows のためのデバッグツールの一部です。一般的な用法は、"quiet mode"でメモリ状態のスナップショットを取り、メモリリークを再生し、別のスナップショットを取り、それから 2 つのショットを比較することです。

User-Mode Dump Heap (UMDH)を使用し Magic xpi をデバッグするには、以下の手順に従ってください。

1. **Debugging Tools for Windows** フォームからダウンロードしてください：

<http://archive.msdn.microsoft.com/debugtoolswindows/Release/ProjectReleases.aspx?ReleaseId=4912>

dbg_x86_6.12.2.633.msi と呼ばれるファイルをダウンロードしてください。

2. コマンドプロンプトを開き、"**Debugging Tools for Windows**"のインストールディレクトリ (通常は、**ProgramFiles%\Debugging Tools for Windows** (64 ビット OS の場合、**"(*86)"** が追加されます))を開いてください。
3. **pdb** ファイルのディレクトリを参照するため、環境変数 **_NT_SYMBOL_PATH** を定義してください。
4. ユーザースタックトレースを有効にしてください：**gflags -i +ust**

注意： グローバルなフラグ UI を経由して行うこともできます。

5. 実行エンジンを起動してください。そして、**Magic xpi** アプリケーション/サーバを実行してください。
6. 開始点に達してください。実行形式で実行した後で、再び使用する予定のタスクを調べてください。
7. **begin.txt** と呼ばれるファイルでスナップショットを作ってください。

umdh -p: -f:

例えば：**umdh -p:1234 -f:d:\temp\begin.txt**

8. メモリリークを作成するために、あと数回同じプログラムを実行してください。次に、最初のスナップショットを実行した時に存在していたのと同じ **Open Windows** の状態にしてください。

再び、**end.txt** と呼ばれるスナップショットを作成してください。

umdh -p:1234 -f:d:\temp\end.txt (これは今回異なるファイルです.)

さて、ファイル間で比較してください。

umdh -d d:\temp\begin.txt d:\temp\end.txt -f:d:\temp\diff.txt

9. **diff.txt** を送ってください。
10. ユーザスタックトレースを無効にしてください: **gflags -i -ust**

Magic xpi で .NET ユーティリティコンポーネントをデバッグするには

Magic xpi で .NET ユーティリティコンポーネントをデバッグするには :

1. .NET コードをリビルドし、関連した **pdb** ファイルが作成されたことを確認してください。
 2. .NET コードの中で、ブレイクポイントを定義してください。
 3. デバッガでプロジェクトを開始し、ブレイクポイントを .NET のステップに設定してください。
 4. Visual Studio を、サーバ(以下のリンクを参照してください)に接続してください。
<http://msdn.microsoft.com/en-us/library/vstudio/3s68z0b3.aspx>
- デバッグする .NET コードを選択してください。
5. Magic xpi スタジオ で、ステップモードでプロジェクトを実行してください。

プロジェクト

プロジェクト

以下のトピックは、Magic xpi でのプロジェクトについて扱っています。

[新しいプロジェクトを作成するには](#)

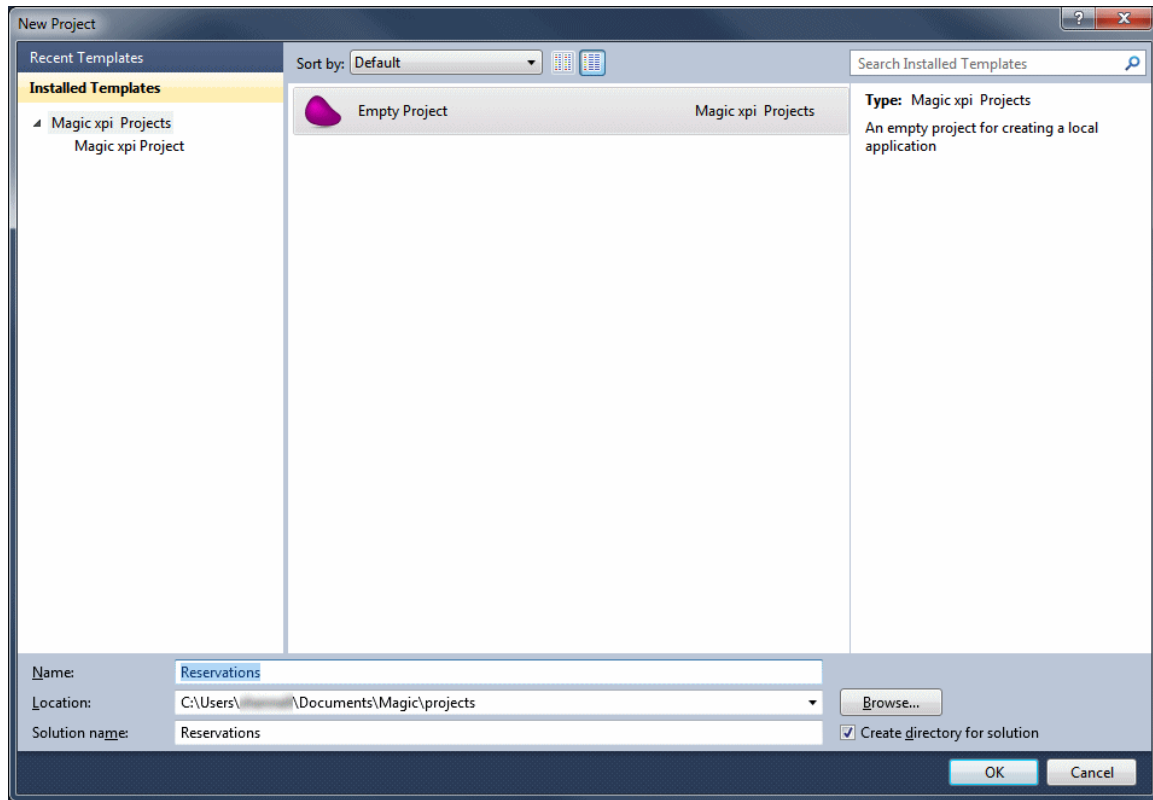
[既存のプロジェクトを開くには](#)

[プロジェクトをバックアップするには](#)

[Magic xpi プロジェクトを削除するには](#)

[プロジェクトをリネームするには](#)

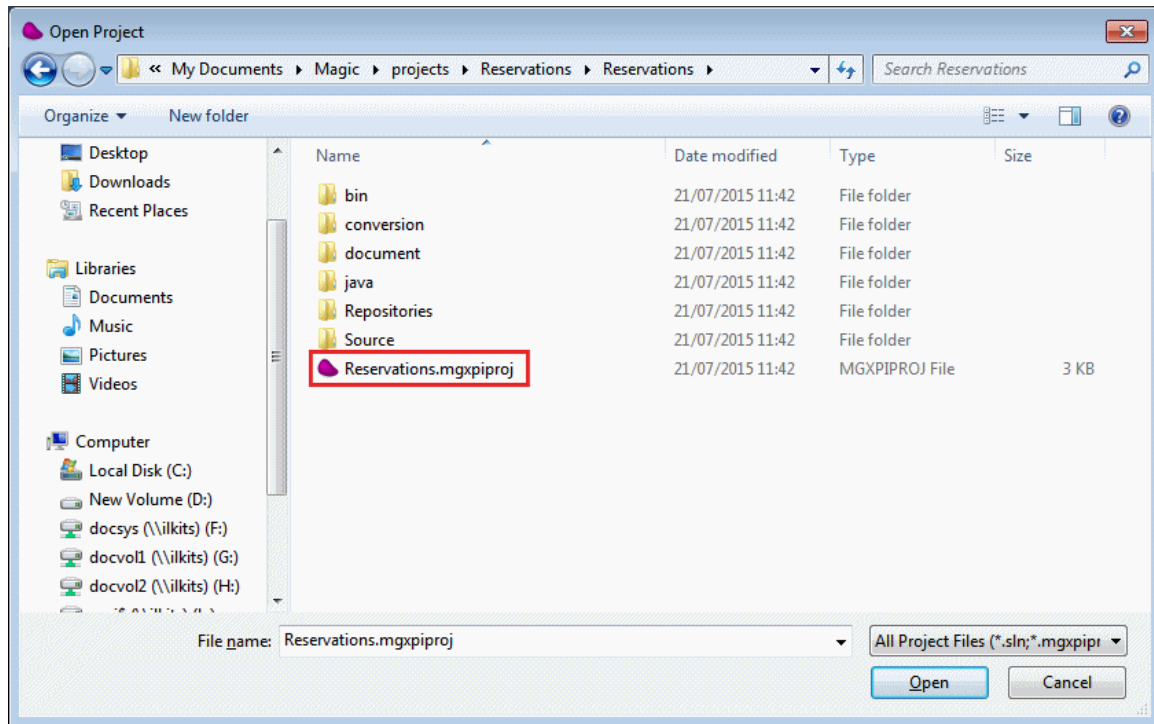
新しいプロジェクトを作成するには



新しいプロジェクトを作成するには：

1. ファイル > 新規作成 > プロジェクト を選択するか、**Ctrl+SHIFT+N** を押下してください。すでにプロジェクトを開いている場合、それは閉まり、新しいプロジェクトが作成されます。
2. プロジェクト名を入力してください。
3. 新しいプロジェクトを作成したい位置を選択してください。
4. ソリューション名は、デフォルトで新しいプロジェクトに割り当てた名前と同じになります。**Create directory for solution** チェックボックスを選択することで、変更することができます。
5. [OK] をクリックしてください。

Magic xpi は、[位置]フィールドで指定した位置に新しいフォルダを作成します。



プロジェクトが作成されると、入力した位置パスの新しいサブディレクトリに作成されます。プロジェクト名はディレクトリ名になり、また Magic xpi プロジェクトのファイル名になります。拡張子は **.mgxpiproj** になります。

関連トピック：

[既存のプロジェクトを開くには](#)

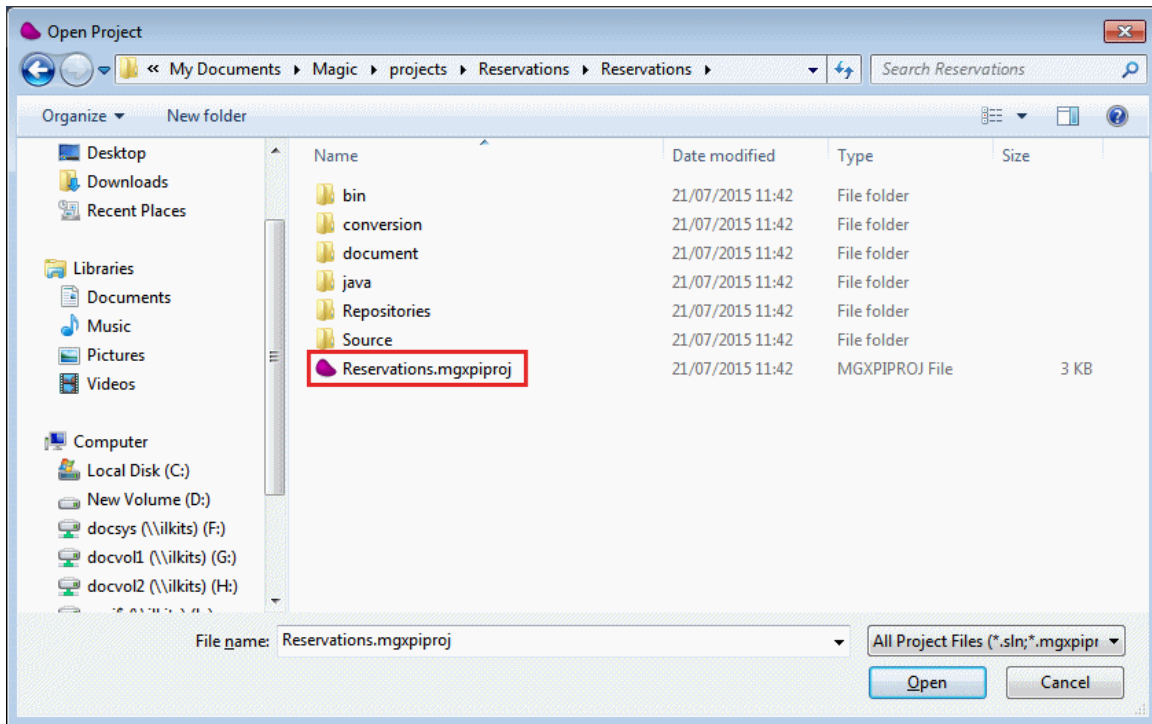
既存のプロジェクトを開くには

簡単に Magic xpi スタジオ でプロジェクト間を移動を行ったり、Windows からプロジェクトを開くことができます。既存のプロジェクトを開く 2 つの基本的な方法は、以下の通りです。

【プロジェクトを開く】ダイアログボックスを使用する

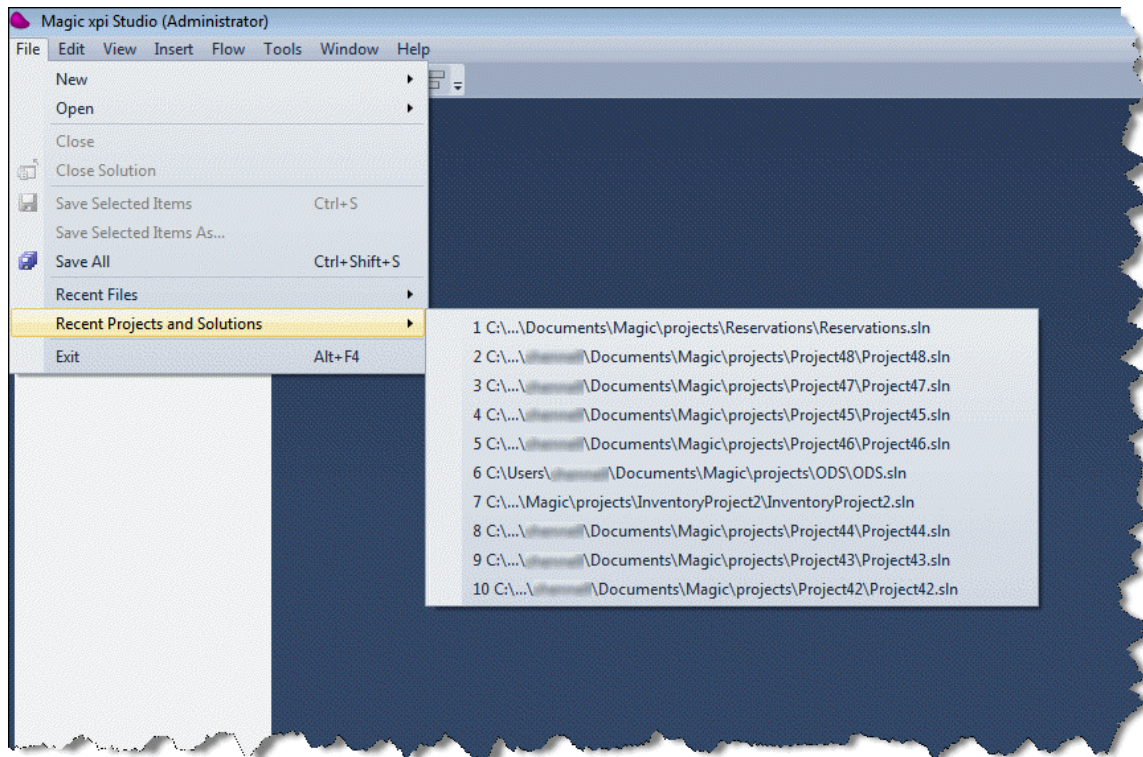
1. 上部のメニューから、**ファイル > 開く > プロジェクト/ソリューション**を選択するか、**Ctrl+SHIFT+O** を押下してください。
2. Windows の [プロジェクトを開く 7] ダイアログボックスが表示されます。要求されたプロジェクトに対応した **.mgxpiproj** ファイルを選択してください。
3. [開く] をクリックしてください。

新しいプロジェクトが開きます。



最近使用したプロジェクト

現在のプロジェクトを閉じる必要はありません。それは自動的に閉まります。プロジェクトに変更を加えた場合、[確認メッセージ]ボックスが表示され、変更を確認します。自身がプロジェクトのセットを使用して動作していることに気付くかもしれません。**Ctrl+SHIFT+O**(ファイル > 開く > プロジェクト/ソリューション)を使用してネットワーク上の任意のプロジェクトを開くことができます。しかし、最近作業したものを開く方が簡単です。



1. ファイル > 最使ったプロジェクトとソリューションを選択してください。
2. 必要なプロジェクトを見つけてください。
3. 開きたいプロジェクトをクリックしてください。

今まで行っていた作業は保存され、選択されたプロジェクトに切り替わります。プロジェクトに変更を加えた場合、[確認メッセージ]ボックスが表示され、変更を確認します。

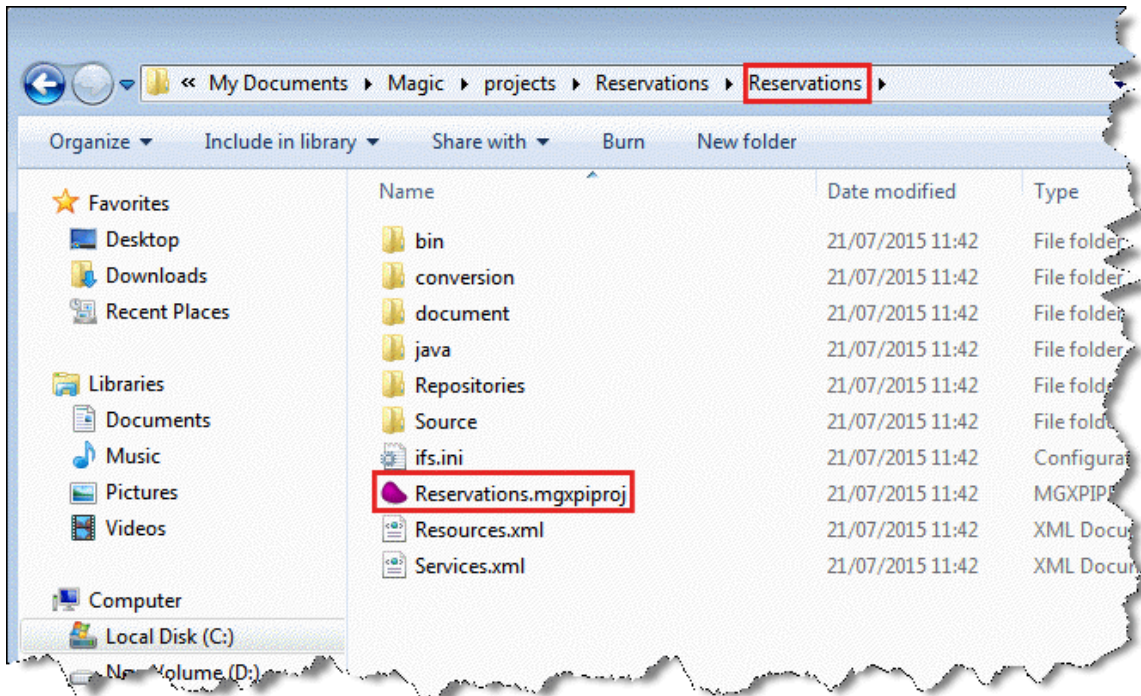
プロジェクトをバックアップするには

作業内容をバックアップすることは常に重要です。自動的にバックアップ処理を行うことができるソースコントロール製品を持っていてもよいですが、このようなものがない場合、よいバックアップ処理のために 2、3 の簡単なオプションを備えています。

- OS 上のファイルのバックアップ処理を作成する。
- 作業中にフローのコピーを作成する。
- プロジェクトのコピーを保存する。

これらについて、以下で紹介しています。

OS 上のファイルのバックアップする



Magic xpi が新しいプロジェクトを作成する時、プロジェクトの構造はルートレベルから構成されています。これは、**.mgxpiproj**(プロジェクト)ファイル、2 つの **.xml** ファイル、およびいくつかのサブディレクトリによって構成されています。

従って、プロジェクト全体をバックアップする最も迅速で、最も簡単な方法は、最上層(この例では "Reservations")で単にフォルダを圧縮することです。必要であれば、プロジェクトの複数の圧縮されたバージョンを保持することができます。この方法は、現在のプロジェクトの完全なスナップショットを確実に持つことになります。

ハードウェア上の障害の場合に対応するため他のメディア(CD または外部ハードディスクなど)にこのディレクトリをコピーすることもできます。

作業中にフローのコピーを作成する

フローの作業を始める前に、フローで右クリックし、[コピー]を選択することで常にそのバックアップコピーを作成することができます。フローのビジネスプロセスでパークし、右クリックしてください、[貼り付け]を選択してください。これはプロジェクト全体をバックアップするほどの広範囲なものではありませんが、オリジナルのコピーに戻ることができる確実に迅速な方法です。何か実験的なことを行っている場合は、これは特に便利です。

プロジェクトのコピーを保存する

新しい名前で既存のプロジェクトのコピーを保存することもできます。詳細は、[ここ](#)をクリックしてください。

Magic xpi プロジェクトを削除するには

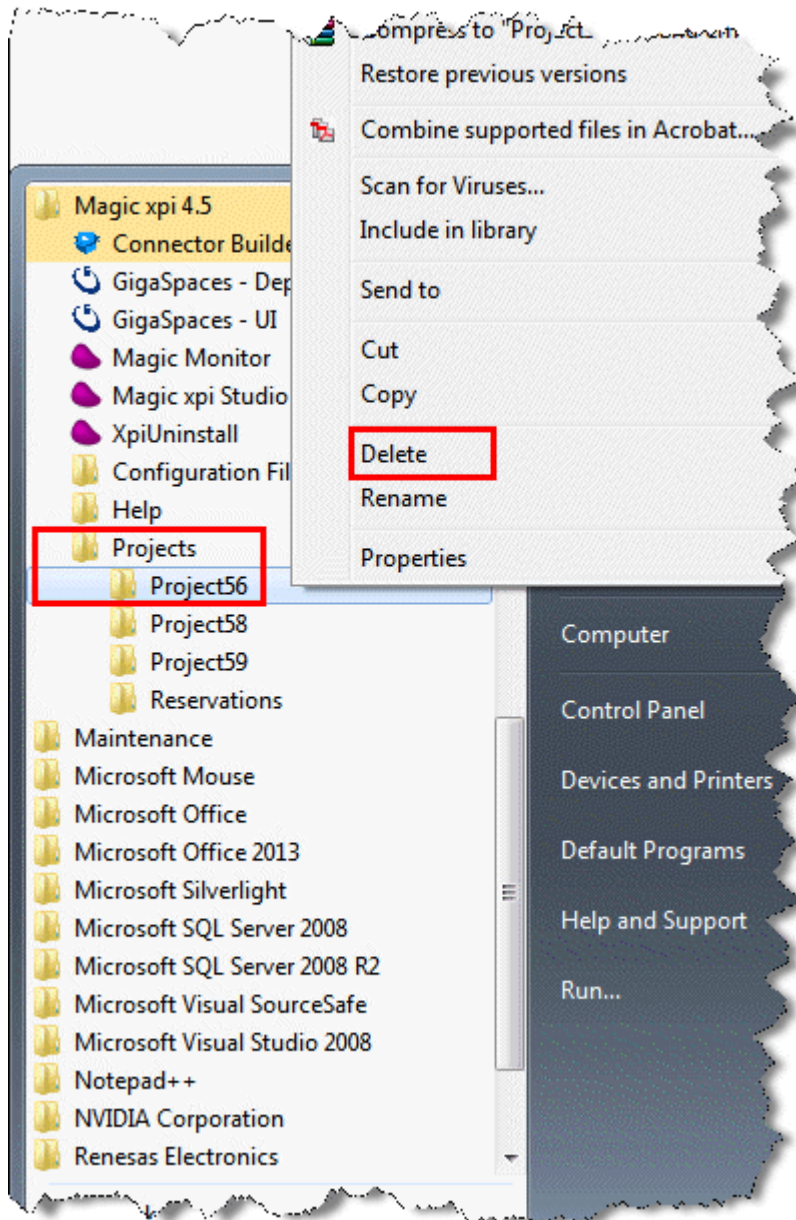
必要なくなったプロジェクトを削除するには、そのディレクトリを <My Documents>¥Magic¥projects ディレクトリより削除します。

例えば、Test という名前のプロジェクトを削除するには、C:¥Magic xpi¥projects¥Test ディレクトリを削除します。

Magic xpi 逆引き辞典

もしプロジェクトに対して[スタート]メニューにショートカットがある場合、必要であればそれも削除してください。削除するには、[スタート]メニューを選択し、該当する Magic xpi インストールを右クリックし、コンテキストメニューから**削除**を選択します。

注： この操作はプロジェクトへのショートカットを削除するだけで、実際のプロジェクトファイルは削除されません。



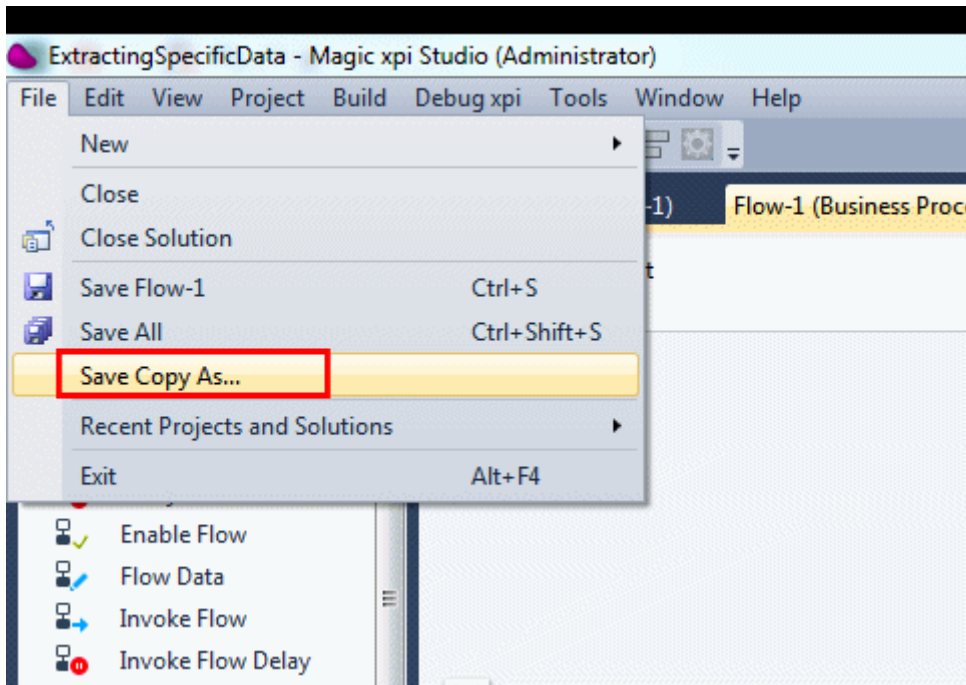
プロジェクトをリネームするには

Magic xpi は、新しい名前で既存のプロジェクトのコピーを保存することができます。リネームされたプロジェクトは、オリジナルのプロジェクトと同じ位置に保存されます。

新しい名前プロジェクトを保存するには：

1. [ファイル] メニューをクリックしてください。

2. 「別名で保存」を選択してください。



Salesforce コネクタ

Salesforce コネクタ

以下のトピックは、Magic xpi の Salesforce コネクタプロジェクトについて扱っています。

[Magic xpi の Salesforce アプリをインストールするには](#)

[最初の Salesforce プロジェクトを作成するには](#)

[setRefbyExternalId 関数を使用して Salesforce における Update と Upsert を行うには](#)

[Salesforce を Web サービスと呼ぶために XMLSpy を使用するには](#)

Magic xpi の Salesforce アプリをインストールするには

Salesforce のために Magic xpi のライセンスを使用することができるため、Magic xpi4.5a で始めて、Salesforce AppExchange 上にある Magic xpi アプリをインストールする必要があります。

アプリをインストールするには：

1. AppExchange アプリケーションをインストールするための権限を持つアカウントで、<https://login.salesforce.com> にログインしてください。
2. 新しいブラウザタブで、次の URL を開いてください：
<https://appexchange.salesforce.com>
3. **Magic xpi Integration Platform for Salesforce** アプリを検索してください。
4. Magic xpi アプリのリンクをクリックしてください。
5.  をクリックしてください。
6. (必要ならば)上記と同じ Salesforce アカウントで、AppExchange にログインしてください。

7. 条件をチェックし、[Install from provider's site] ボタンをクリックしてください。

Install from Magic Software Enterprises's Website

☒ I have read and agree to the [terms and conditions](#) .

Install from provider's site

You will be taken to the provider's website to continue installation. By submitting, you agree to share this information with salesforce.com and Magic Software Enterprises.

8. 必要な情報を入力し、最後のフィールドで、アプリをどこにインストールするかを選択してください。

Yes - Install in Production

Are you already a Magic xpi user?

Yes - Install in Sandbox

Yes - Install in Production

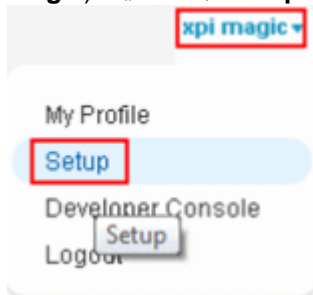
No - Talk to a Representative

9. [Submit] をクリックしてください。
10. どのユーザのためにインストールするかを選択してください。すべてのユーザのためにインストールすることを推奨します。
11. [Install] (または、以前にアプリをインストールしている場合は、[Upgrade])をクリックしてください。
12. [Done] をクリックしてください。

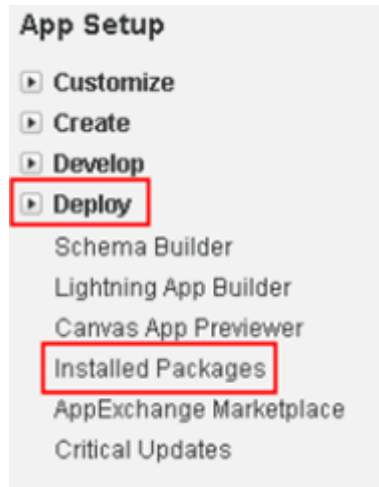
インストールがされる、正しくインストールされたパッケージがあり、次のような Salesforce アカウントでカスタムオブジェクトがあることをチェックしてください。

インストールされたパッケージ

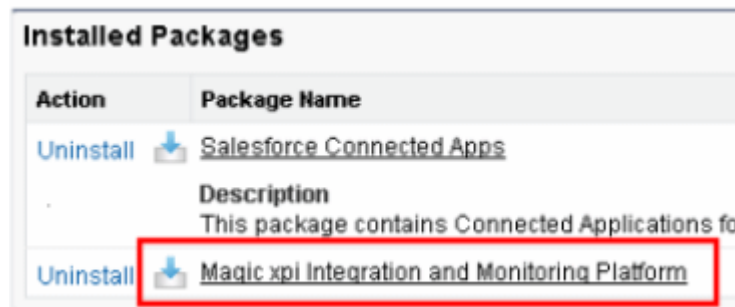
1. インストールされたパッケージを見つけるために、ユーザ(以下のイメージでは、**xpi magic**)に移動し、**Setup** を選択してください。



2. 画面の左側の [App Setup] セクションで、[Deploy] オプションを展開し、インストールされたパッケージを選択してください。

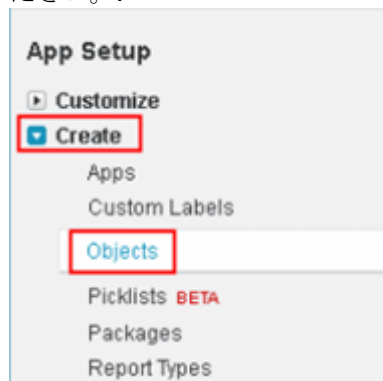


3. Magic xpi インテグレーションとモニタリングプラットフォームのインストールされたパッケージがあることをチェックしてください。

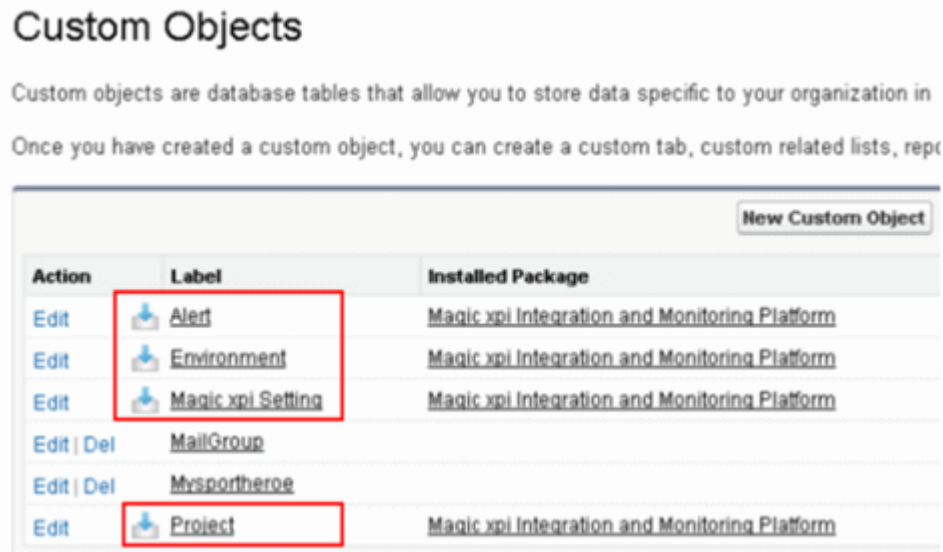


カスタムオブジェクト

1. [App Setup] セクションで、[Create] オプションを展開し、Objects を選択してください。



2. アプリが以下の 4 つのオブジェクトにインストールされていることをチェックしてください: **Alert**, **Environment**, **Magic xpi Setting**, と **Project**.



関連トピック:

[Salesforce を使用してプロジェクトをモニタするには](#)

最初の **Salesforce** プロジェクトを作成するには

このトピックは、Magic xpi の Salesforce プロジェクトを作成する場合の最初のステップを説明します。

Salesforce アカウントを作成するには:

1. <http://www.salesforce.com/developer/> を開いてください。
2. サイト上でサブスクリプションフォームに必要事項を入力して、**Salesforce** アカウントを開いてください。
3. ログイン URL が含まれた E メールを受け取ります。パスワードを変更するために、URL をクリックしてください。
4. パスワードを変更した後に、セキュリティトークンを要求してください。
 - a. ページのトップの **Setup** リンクをクリックしてください。
 - b. [My Personal Information] セクションで、[reset your security token link] をクリックして、セキュリティトークンを含む E メールを受け取るようにします。セキュリティトークンをパスワードに追加し、Magic xpi の [Connections Salesforce] ダイアログボックスの [Password] フィールドに結合された結果を入力すると、後でこれを使用することができます。例えば、パスワードが **1234** で、セキュリティトークンが **AABBCC** である場合、Magic xpi の [パスワード] フィールドで **1234AABBCC** を入力する必要があります。

注 salesforce.com の Web サイトにログインする場合は、セキュリティトークンは必要ありません。
意: Magic xpi などの外部アプリケーションを使用する時だけトークンは使用されます。

5. <https://login.salesforce.com/> を開いてください。ここで、**Salesforce** アプリケーションにログインすることができます。



簡単な Salesforce プロジェクトを作成するには：

この例では、Query オペレーションは、すべて **Channel Customers** アカウントタイプで実行されます。

Salesforce との接続を設定する：

1. Magic xpi スタジオを開き、新しいプロジェクトを作成してください。
2. [プロジェクト] メニューから [設定] を選択して、[設定] ダイアログボックスを開いてください。
3. [設定] ダイアログボックスの [リソース] セクションで、[追加] をクリックしてください。[新規リソース] ダイアログボックスが開きます。
4. ドロップダウンリストから **Salesforce** のリソースタイプを選択してください。[リソース名] フィールドで、新しい **Salesforce** リソースの名前を入力し、[OK] をクリックしてください。
5. 関連したフィールドで **Salesforce** サーバのユーザ名とパスワードを入力してください。パスワードは、**Salesforce** パスワードと **salesforce.com** によって供給されたセキュリティトークンの組み合わせです。
6. 新しい接続をテストするために、[検証] をクリックしてください。
7. 接続が確認されたら、[オブジェクトリスト] をクリックしてアップデートされたサポートされたオブジェクトリストを取得してください。
8. 接続定義を保存するために、[適用] をクリックしてください。

Salesforce コネクタを設定する：

1. **Salesforce** コネクタをフローエリアにドラッグしてください。
2. データマップをダブルクリックするか、右クリックして [構成] をクリックし、[Salesforce 構成] ダイアログボックスを開いてください。
3. [オブジェクト] フィールドの隣で  をクリックし、**アカウントオブジェクト** を選択してください。
4. [オペレーション] フィールドで、ドロップダウンリストから **Query** を選択してください。
5. フロー変数リポジトリを開くために、**CTRL+L** を押下してください。**F.ResultBlob** と呼ばれる新しい **BLOB** 変数を定義してください。
6. [Store Result In] フィールドの隣で  をクリックし、戻りの **XML** データを保持するために、**F.ResultBlob** を選択してください。
7. [データマップ] ウィンドウを開くために、[OK] をクリックしてください。
8. 送り先側で、タイプ要素をダブルクリックして、[プロパティ] ペインの [計算値] フィールドのチャンネルに **Customer - Channel** を入力してください。これは、**channel customers** だけを検索することを可能にします。
9. [OK] をクリックして構成画面を終了してください。

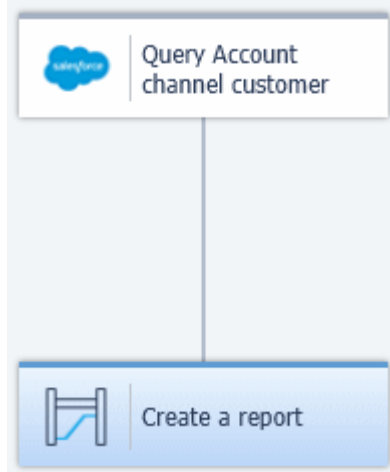
Salesforce XML のクエリ結果をフラットファイルにマッピングする：

1. データマップコンポーネントをフローに **Salesforce** コネクタの子としてドラッグします。

2. データマップをダブルクリックするか、右クリックし [設定] をクリックしてください。 [データマップ] ウィンドウが開きます。
3. ツールボックスの [マップスキーマ] セクションから、XML タイプを [データマップ] ウィンドウの送り元エリアにドラッグします。
4. XML 送り元の [プロパティ] ペインで、アカウント XSD ファイルを選択してください。
<プロジェクト名>/salesforce/XSD/<Salesforce.com 接続名>/Account.xsd の下でこれを見つけることができます。
5. データソースとして **F.ResultBlob** を選択してください。
6. ツールボックスの [マップスキーマ] セクションから、フラットファイルタイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
7. データベースソースの [プロパティ] ペインの [データ送り先] フィールドで、ドロップダウンリストから **ファイル** を選択してください。それに以下の値を設定してください：**EnvVal('currentprojectdir')&'ChannelAccounts.txt'**
8. テーブルエリアで、3つの文字フィールドを作成するために、[新規] を3回クリックしてください。
9. 新規フィールドを、**Name**、**Country**、および **Website** にリネームしてください。
10. [OK] をクリックしてください。
11. [データマップ] ウィンドウで、[Name] 送り元ノードを [Name] 送り先ノードにマッピングしてください。
12. [データマップ] ウィンドウで、[BillingCountry] 送り元ノードを [Country] 送り先ノードにマッピングしてください。
13. [データマップ] ウィンドウで、[Website] 送り元ノードを [Website] 送り先ノードにマッピングしてください。
14. [保存] をクリックしてください。

フローの実行：

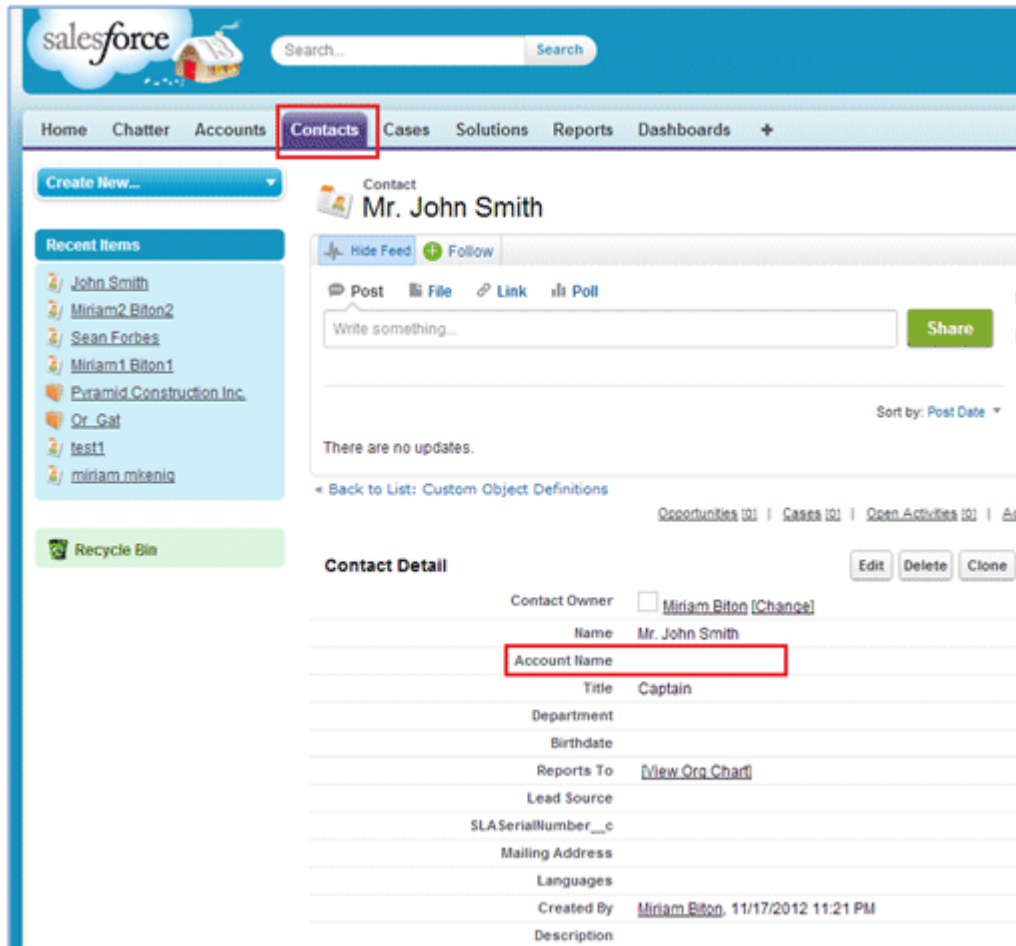
1. デバッガモードでプロジェクトを実行してください。Salesforce クエリ結果に続き、**ChannelAccounts.txt** ファイルが利用可能で、すべてのチャンネルレコードを含んでいることをチェックするために、ステップを使用してください。
2. 最終的なフローは、以下の表示表示されます。



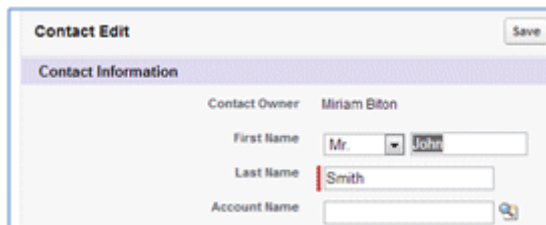
setRefbyExternalId 関数を使用して Salesforce における Update と Upsert を行うには

setRefbyExternalId 関数を使用している Salesforce で Update と Upsert を行うには、以下の手順に従ってください。

1. Magic xpi を使用して、Salesforce でコンタクトを作成してください。すべての Salesforce コンタクトには、**ルックアップ アカ운ツフィールド**(以下にで表示されているように **Account Name** と名付けられています)があります。



コンタクトの編集モードでは、Account Name はルックアップフィールドです。



2. **Setup > App Setup > Customize > Accounts > Fields** を選択します。以下のイメージのように、アカウントには、API 名 **SLASerialNumber__c** が設定された 外部 ID フィールドがあります。

Action	Field Label	Field Name	Data Type
	Account Name	Name	Name
Edit	Account Number	AccountNumber	Text(40)
Edit	Account Owner	Owner	Lookup(User)
Edit	Account Record Type	RecordType	Record Type
Edit	Account Site	Site	Text(80)
Replace Edit	Account Source	AccountSource	Picklist
Edit	Annual Revenue	AnnualRevenue	Currency(18, 0)
	Billing Address	BillingAddress	Address

Action	Field Label	API Name	Data Type	Controlling Field
Edit Del Replace	Active	Active__c	Picklist	
Edit Del	custom_date	custom_date__c	Date	
Edit Del	custom_datetime	custom_datetime__c	DateTime	
Edit Del Replace	Customer Priority	CustomerPriority__c	Picklist	
Edit Del	my_field	my_field__c	Text(40)	
Edit Del	Number of Locations	NumberOfLocations__c	Number(3, 0)	
Edit Del Replace	SLA	SLA__c	Picklist	
Edit Del	SLA Expiration Date	SLAExpirationDate__c	Date	
Edit Del	SLA Serial Number	SLASerialNumber__c	Text(10)	External ID

3. アカウント名 **Pyramid Construction Inc** フィールドを示し、**Account ID** を知る必要が無い **Contact “John Smith”** フィールドを更新するには、**SLASerialNumber__c** の値を使用する事ができます。この場合に、値は **1981** になります。

Field	Value
Account Owner	Miriam Biton (Change)
Account Name	Pyramid Construction Inc. (View Hierarchy)
Parent Account	
Account Number	OC213425
Account Record Type	
Account Site	
Type	Customer - Channel
Industry	Construction
Annual Revenue	\$950,000.00
my_field	
custom_date	
custom_datetime	
Billing Address	2 Place Jussieu Paris, 75251 France
Shipping Address	2 Place Jussieu Paris, 75251 France
Customer Priority	
SLA Expiration Date	11/30/2009
Number of Locations	17
SLA	Silver
SLA Serial Number	1981
Upsell Opportunity	Maybe

4. Salesforce コネクタでコンタクトを更新する時、**AccountId** フィールドはデータマップに表示されます。コンタクトの [Account Name] 前フィールドがルックアップフィールドであるため、以下の詳細を **AccountId** にマッピングすることができます。

'setRefbyExternalId (Account, SLASerialNumber __c, 1981)'

実行中、AccountIdに挿入される値は、SLA シリアル番号が 1981 と同じアカウントの ID となります。SLA シリアル番号が 1981 に設定される複数のアカウントがある場合、処理は失敗します。

同じ方法は、Upsert オペレーション (唯一の違いは、Upsert では Upsert 条件を指定する必要があることです) や Create オペレーション (Create オペレーションの必須フィールドを指定する必要があります) で有効です。

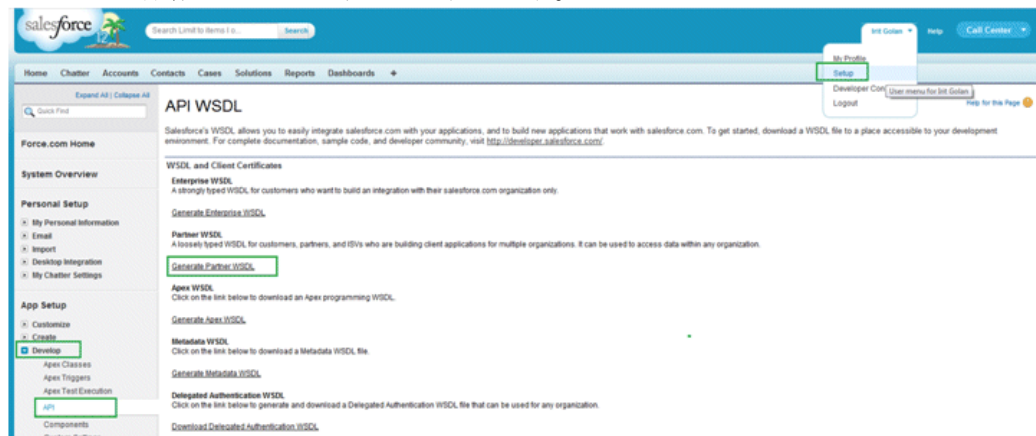
Salesforce を Web サービスと呼ぶために XMLSpy を使用するには

このトピックは、テストツールとして Magic xpi の外部でどのように Salesforce Web サービスを呼び出すか、そして Salesforce API の動作を確認するかについて説明しています。

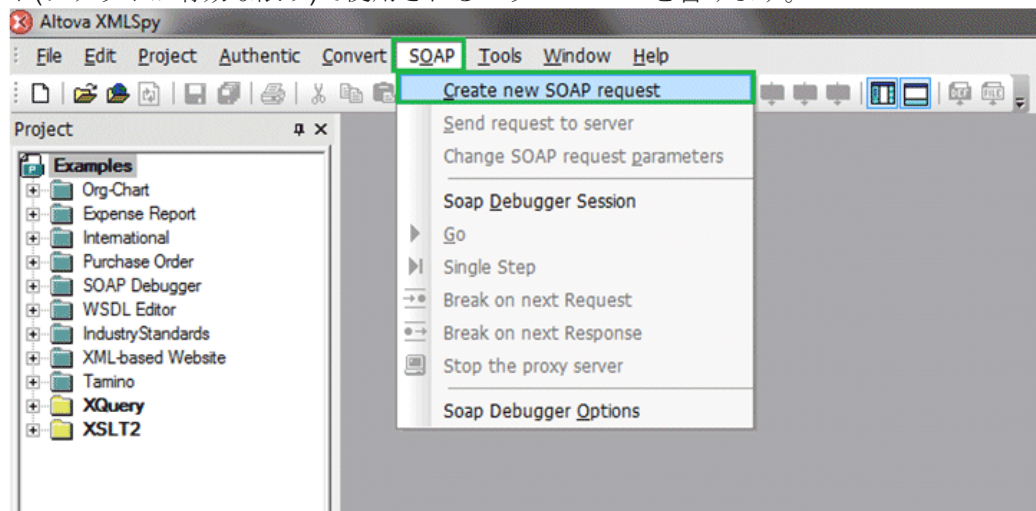
これは、SOAP リクエスト (例えば、soapUI) を送ることが可能な様々なツールを使用するで行うことができます。このトピックは、XMLSpyR を使用して、どのようにリクエストを Salesforce に送るかを、ログイン問題をチェックし、SOQL クエリを実行し、`describeObjects()` オペレーションを実行し、Upsert オペレーションを使用したり、`getServerTimestamp` を使用することが可能になることを実証しています。

Salesforce Web サービスを呼び出すために XMLSpy を使用するには：

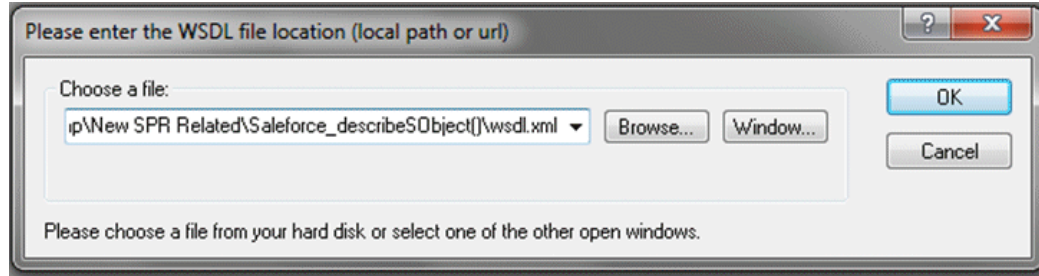
1. Salesforce WSDL ファイルをダウンロードし、それをローカルで簡単にアクセス可能な場所に保存してください。salesforce.com にログインし、**Setup** を選択してください。
2. **App Setup > Develop > API** を選択し、**Generate Partner WSDL** を選択してください。ローカルに保存した XML ファイルが開きます。



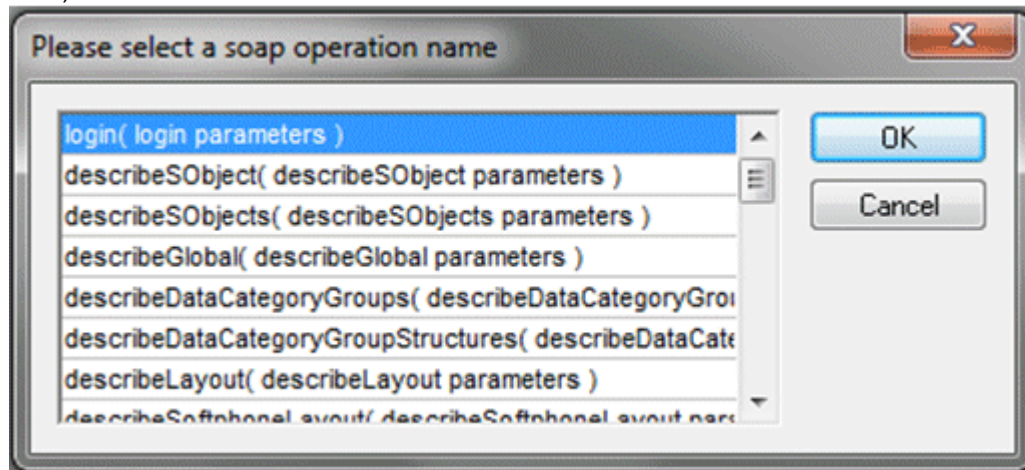
3. XMLSpyを開き、ログイン処理の **Create new SOAP request** を使用して、Salesforce にログインしてください。Web サービスの応答は、さらなる Web サービスリクエスト (チケットが有効な限り) で使用されるセッション ID を含みます。



1. [WSDL file location] ウィンドウが開きます。ここで、以前にローカルに保存した Salesforce WSDL を選択してください。



2. WSDL を選択すると [Select Soap Operation Name] ウィンドウが開きます。次に、必要な処理コマンドを選択してください。(最初に **Salesforce** でチケットを開くためにログイン処理を選択する必要があるであることに注意してください。)



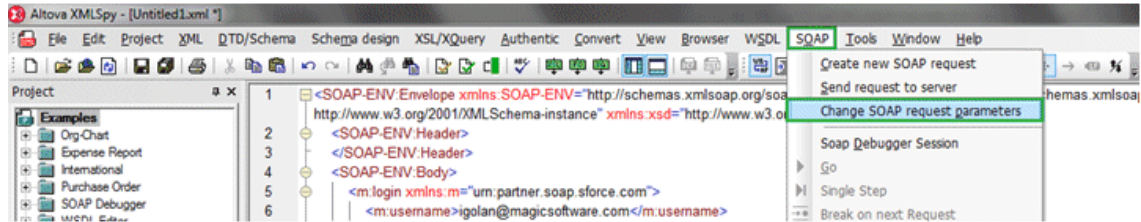
3. ログイン XML が開きます。
 - a. 関連しない全ての要素を削除することができます。
 - b. 以下を有効な **Salesforce** 証明書と置き換えてください。
 - i. `<m:username> String</m:username>`
 - ii. `<m:password> String</m:password>`

ログインリクエストは、以下のように表示されるはずです：

```
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Header>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <m:login xmlns:m="urn:partner.soap.sforce.com">
      <m:username>userName</m:username>
      <m:password>Password+Token</m:password>
    </m:login>
  </SOAP-ENV:Body>
```


</SOAP-ENV:Envelope>

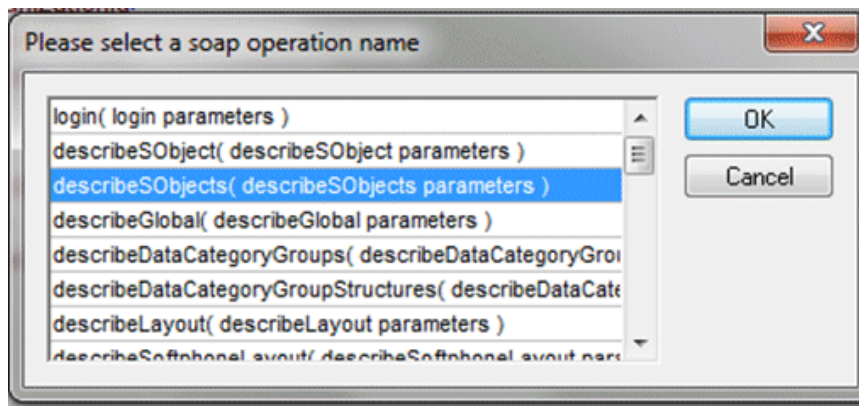
7. リクエストを Salesforce に送ってください。



4. より多くの Web サービスリクエストを Salesforce に送る必要がある場合、以下を行う必要があります。

- a. 新規の SOAP リクエスト(ステップ 3 を参照)を作成してください。
- b. 要求された SOAP オペレーションを選択してください。

8. 以下のステップは、オブジェクトのメタデータを取得するための **describeSObjects** オペレーションを示しています。



5. XML が開かれます。:

6. 関連しない全ての要素を削除することができます。

・ **<m:sessionId>String</m:sessionId>** をログイン WS 結果 XML からのセッション ID 文字列に置き換えます。

- a. **<m:sObjectType>String</m:sObjectType>** をメタデータを取得したいオブジェクトに置き換えます。例えば、 **Account**。

7. **describeSObjects** リクエストは、以下のように表示されるはずですが。

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
```

```
<SOAP-ENV:Header>
```

```
<m:SessionHeader xmlns:m="urn:partner.soap.sforce.com">
```

```
<m:sessionId>00D70000000Jv4S!ARoAQB5ijhZCQhFpql4UFcs_qro5mQFg1d9A6_k8aDxniMM4WlaRnq_KOZudBnsau2GZwLGpBEmhf_PCKbfX7.InihBsPet5</m:sessionId>
```

```
</m:SessionHeader>
```

```
</SOAP-ENV:Header>
```

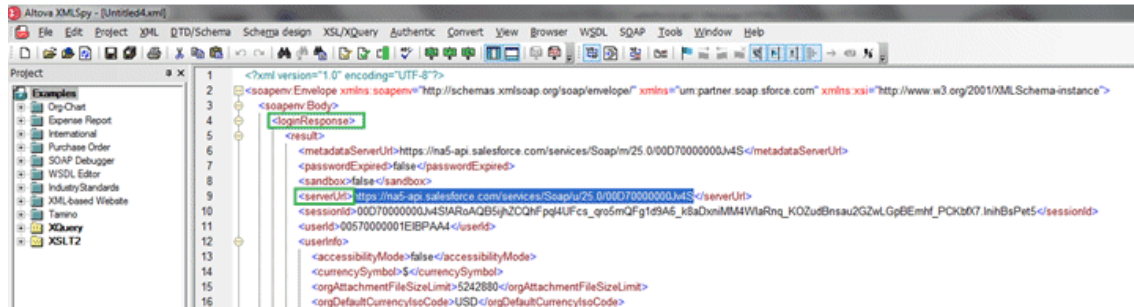
```
<SOAP-ENV:Body>
```

```
<m:describeSObjects xmlns:m="urn:partner.soap.sforce.com">
```

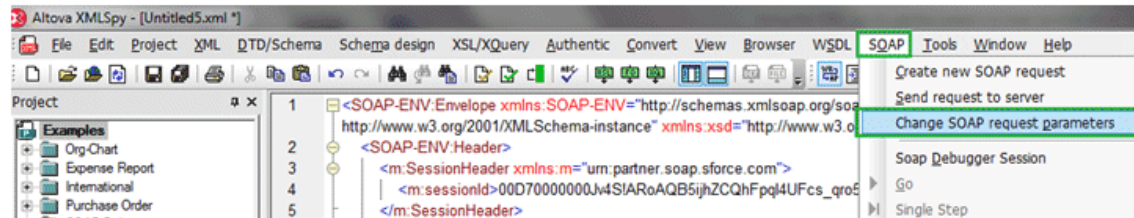
```
</m:describeObjects>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

8.

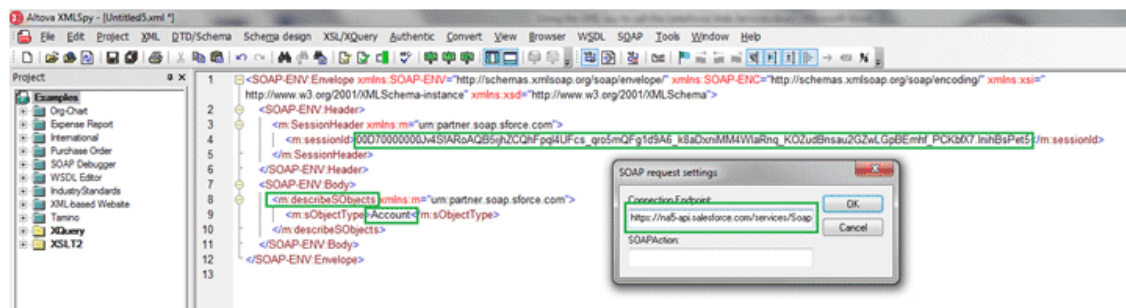
11. login Web service response に移動し、サーバ URL の文字列をコピーしてください。



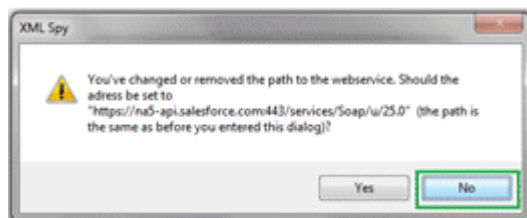
12. **describeObjects** リクエスト XML 上でパークし、[SOAP] メニューを選択して、[Change SOAP request parameters] を選択します。



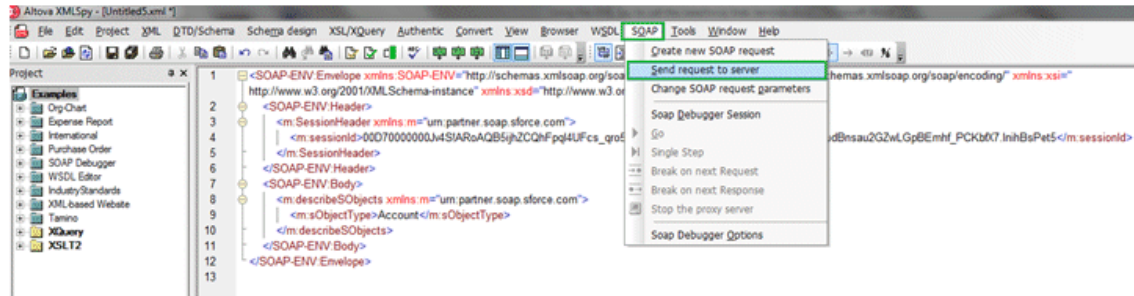
SOAP リクエスト設定において、接続エンドポイントを、**login Web service response**(ステップ 11)からコピーしたサーバ URL 文字列で置き換えてください。



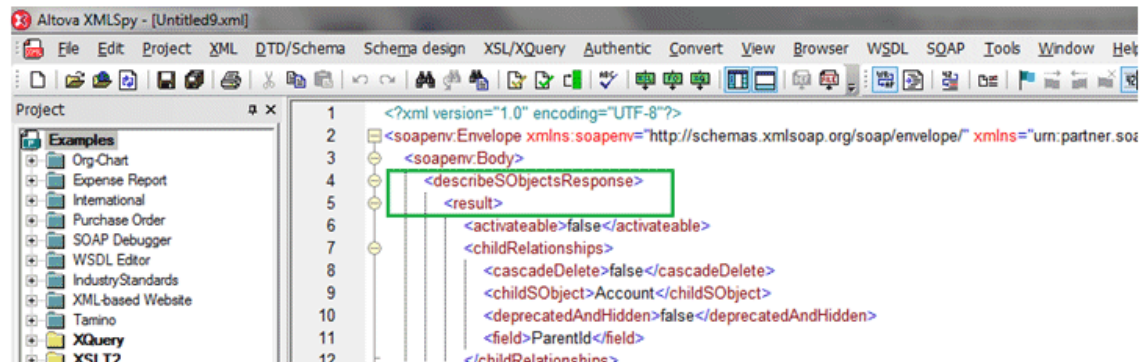
メッセージウィンドウが開きます。**No** を選択します。



13. リクエストを送ってください。



別の XML が開きます。これは Web サービス応答です。



注意： 上記の方法は、SOAP リクエスト (soapUI などの) を送ることが可能な様々なツールを使用して実行することができますが、このドキュメントは XMLSpy のみ使用しています。

SAPB1 コネクタ

SAPB1 コネクタ

以下のトピックは、Magic xpi の SAPB1 コネクタについて扱っています。

[SAPB1 コネクタで複数オブジェクトを照会するには](#)

[SAPB1 コネクタで複数のオブジェクトを追加するには](#)

[SAPB1 コネクタをアップグレードするには](#)

[SAPB1 フィールド内の値をクリアにするには](#)

[SAPB1 HANA を使用するには](#)

SAPB1 コネクタで複数オブジェクトを照会するには

SAPB1 コネクタで複数オブジェクトを照会するには

SAPB1 データインターフェース (DI) API は、一度に一つのオブジェクトを処理します。一方、Magic xpi のデータマッピングユーティリティは、一つのステップ内で複数のオブジェクトの照会を行うことができます。ステップを構成するとき、Magic xpi は DI で複数のオブジェクトを照会できるようにします。

このトピックの終わりに、SAPB1 のメインオブジェクトの複合要素に関する、特別な注意事項があります。

注意： ユーザーテーブルシステム内の追加または更新中に、SAP B1 によって管理されている 'like Object, UserSign, Series, Period, Transferred, CreateDate, CreateTime and DataSource' の

各フィールドは、ユーザによって更新することはできません。

例

複数の SAPB1 オブジェクトを照会するには、三つの主要なステージがあります。

- ステージ 1: Magic xpi の SAPB1 コンポーネントを構成します。
- ステージ 2: 複数照会の結果をマッピングします。
- ステージ 3: マップされた複数照会の結果を表示します。

ステージ 1 : SAPB1 コンポーネントを構成する

この例では Business Partners の一覧がテキストファイルにある場合、複数の BusinessPartners を照会する方法について説明します。この例では、以下の情報を使用してサンプルテキストファイルを作成します。

Business Partner ファイルテンプレートは、以下のようになります。

BPName,CardCode,CardType,Tel1,Tel2,Contact,email

テキストファイル内のデータは、以下のようになります。

Test 00001,M00001,cCustomer,111-00001,222-00001,Contact 00001,1@mail.com


Test 00002,M00002,cCustomer,111-00002,222-00002,Contact 00002,2@mail.com

Test 00003,M00003,cCustomer,111-00003,222-00003,Contact 00003,3@mail.com

Test 00004,M00004,cCustomer,111-00004,222-00004,Contact 00004,4@mail.com

Test 00005,M00005,cCustomer,111-00005,222-00005,Contact 00005,5@mail.com

以下のステップは、Business Partners の一覧がテキストファイルにある場合、どのように複数の BusinessPartners を照会するかを説明しています。

1. [設定] ダイアログボックスの [リソース] セクションで **SAPB1** リソースを定義してください。
2. **SAPB1** コネクタをフローにドラッグします。
3. **SAPB1** コネクタの [プロパティ] ペインで、インタフェースとして **XML** を選択してください。
4. [プロパティ] ペインの [設定] セクションで、以前に [設定] ダイアログボックスの [リソース] セクションで定義した **SAPB1** リソースを選択してください。まだ **SAPB1** リソースを定義していない場合、**SAPB1** コネクタの設定を継続することができません。
5. **SAPB1** コネクタをダブルクリックするか、右クリックして [設定] をクリックします。設定ウィンドウが開きます。
6. データオブジェクトリストを開くために、 をクリックしてください。ビジネスパートナーを選択し、[OK] をクリックしてください。
7. ツールボックスの [マップスキーマ] セクションから、**フラットファイルタイプ** [データマップ] ウィンドウの送り元エリアにドラッグしてください。
8. 送り元の [プロパティ] ペインで、[Example] フィールドに最初の頃に作成したテキストファイルパスをコピーし、データソースを選択してください。

9. [リフレッシュ] をクリックして、サンプルファイルを読み込みます。データマップは、サンプルファイル内の情報を使用してフラットファイルのデータを調整するかを定義します。
10. [保存] をクリックします。
11. [データマップ] ウィンドウで、関連したクエリパラメータ接続を作成してください。この例では二つの **CardCode** 要素を接続します。複数のオブジェクトを照会するには、**Record** 要素を **BO** 要素に結び付けます。

Magic xpi は、複数のオブジェクトのデータを格納する単一の XML ファイルを作成します。Magic xpi は、このファイルを使用して SAPB1 の DI の照会を行います。XML ファイルは、ユーザ BLOB 変数に保存されます。



注 照会結果を保存したい場所を指定することができます。このオプションは、[SAPB12005 構成]ダイアログボックスの[アドバンスド照会オプション]セクションの[結果を保存]ドロップダウンリストにあります。

XML コードの照会結果のサンプルを参照するには、[こちら](#)をクリックしてください。

ステージ 2: 複数照会の結果をマッピングする


複数照会の結果は、Magic xpi の ODS テーブルに保存されます。Business Partners の名前一覧は、ODS 文字列フィールドに保存されます。

照会結果をマッピングするには、以下のようにします。

1. データマッパーサービスを **SAPB1** コンポーネントにドラッグします。
2. データマップをダブルクリックするか、右クリックして [設定] をクリックします。
[データマップ] ウィンドウが開きます。
3. ツールボックスの [マップスキーマ] セクションから、以下の送り元の 1 つを、[データマップ] ウィンドウ(以下の例は **QueryResult_BP** という名前を使用しています)の送り元エリアにドラッグしてください。
 - **XML** 送り元タイプを XML に設定し、以下にある **Business partner XSD** を選択します。
<current project>\SAPB1_2004\XSD\oBusinessPartners(2).xsd, または
 - **IFC Model** 送り元タイプを IFC モデルに設定し、 をクリックしてコンポーネントリストを開き **SAPB1** コネクタを選択します。そして、 をクリックしてスキーマリストを開き、**Business Partners** を選択します。データソースリストボックスから **UserBLOB** を選択します。
4. ツールボックスの [マップスキーマ] セクションから、**ODS** タイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。
5. 送り先の [プロパティ] ペインで、ODS リポジトリを開くために [ODS] ボタンをクリックしてください。
6. [新規] をクリックして、キー名を入力します。この例では **BP_Name** という名前を使用します。チェックボックスにチェックを入れ [OK] をクリックします。
7. [データマップ] ウィンドウで、前の段階で作成された **BO** 要素を **BP_Name** 要素と接続してください。そして、**CardName** 要素を **Alpha** 要素と接続してください。

ステージ 3: マップされた複数照会の結果を見る

Magic xpi 逆引き辞典

1. デバッガを実行します。[デバッグ xpi] メニューの [システムテーブル] で [ODS] をクリックします。
2.  をクリックして [ODS 詳細] 画面を開き、照会結果を表示します。

SAPB1 のメインオブジェクト複合要素

各オブジェクトの SAPB1 スキーマでは、実際に XML でこの複合要素の 1 つを持つ場合だけ、メインのオブジェクト(例:Business Partner/row)の要素は、複数の occurrence としてスキーマ内に誤って定義されます。

マップ規則に従うことで複数の occurrence 要素(行)を一つの occurrence 要素(QueryParams)にもマッピングできないため、SAPB1 コンポーネント内で複数の更新、参照、削除を行おうとした場合、この定義では問題が発生します。また、送り元は、SAPB1 オブジェクトの XML になります。

この問題に対して 2 つの解決策があります。

- 一つの occurrence 要素とするために、行レベル (常に True になっています)上で、フィルタを定義する。
- 手動で XSD を修正する。

XML コードのサンプル

```

<?xml version="1.0" encoding="UTF-16" standalone="no" ?>
<BOM>
  <B0>
    <AdmInfo>
      <Object>2</Object>
      <Version>2</Version>
    </AdmInfo>
    <BusinessPartners>
      <row>
        <CardCode>M00001</CardCode>
        <CardName>Test 00001</CardName>
        <CardType>cCustomer</CardType>
        <Phone1>111-00001</Phone1>
        <Phone2>222-00001</Phone2>
        <EmailAddress>1@mail.com</EmailAddress>
      </row>
    </BusinessPartners>
    <ContactEmployees>
      <row>
        <Name>Contact 00001</Name>
      </row>
    </ContactEmployees>
  </B0>
  <B0>
    <AdmInfo>
      <Object>2</Object>
      <Version>2</Version>
    </AdmInfo>
    <BusinessPartners>
      <row>
        <CardCode>M00002</CardCode>
        <CardName>Test 00002</CardName>
        <CardType>cCustomer</CardType>
        <Phone1>111-00002</Phone1>
        <Phone2>222-00002</Phone2>
        <EmailAddress>2@mail.com</EmailAddress>
      </row>
    </BusinessPartners>
    <ContactEmployees>
      <row>
        <Name>Contact 00002</Name>
      </row>
    </ContactEmployees>
  </B0>

```

SAPB1 コネクタで複数のオブジェクトを追加するには

SAPB1 データインターフェース(DI)API は、一度に一つのオブジェクトを処理します。一方、Magic xpi のデータマッピングユーティリティは、一つのステップ内で複数のオブジェクトの照会を行うことができます。ステップを構成するとき、Magic xpi は DI で複数のオブジェクトを照会できるようにします。

複数の SAPB1 オブジェクトを追加する場合、まず最初に XML インターフェースを使用して Magic xpi の SAPB1 コンポーネントを構成します。そして送り元複合物を SAPB1 BO 複合物に接続します。これにより同時に複数のレコードを追加することができます。この処理は以下の例で説明されています。

例

この例では、Business Partners の一覧がテキストファイルになっている場合、複数の Business Partners を追加する方法を説明しています。この例においては、以下の情報を格納したサンプルテキストファイルを作成します。

Business Partners ファイルテンプレートは、以下のようになります。

BPName,CardCode,CardType,Tel1,Tel2>Contact,email

テキストファイルのデータは、以下のようになります。

Test 00001,M00001,cCustomer,111-00001,222-00001>Contact 00001,1@mail.com


Test 00002,M00002,cCustomer,111-00002,222-00002>Contact 00002,2@mail.com

Test 00003,M00003,cCustomer,111-00003,222-00003>Contact 00003,3@mail.com

Test 00004,M00004,cCustomer,111-00004,222-00004>Contact 00004,4@mail.com

Test 00005,M00005,cCustomer,111-00005,222-00005>Contact 00005,5@mail.com

以下のステップは、Business Partners の一覧がテキストファイルにある場合の複数の Business Partners の追加の方法です。

1. [設定] ダイアログボックスの [リソース] セクションで **SAPB1** リソースを定義してください。
2. **SAPB1** コネクタをフローにドラッグします。
3. **SAPB1** コネクタの [プロパティ] ペインで、インターフェースとして **XML** を選択してください。
4. [プロパティ] ペインの [設定] セクションで、以前に [設定] ダイアログボックスの [リソース] セクションで定義した **SAPB1** リソースを選択してください。まだ **SAPB1** リソースを定義していない場合、**SAPB1** コネクタの設定を継続することができません。
5. **SAPB1** コネクタをダブルクリックするか、右クリックして [構成] をクリックします。[構成] ウィンドウが開きます。
6. [オペレーション] ドロップダウンリストから、**Add** を選択してください。
7. データオブジェクトリストを開くために、 をクリックしてください。ビジネスパートナーを選択し、[OK] をクリックしてください。
8. ツールボックスの [マップスキーマ] セクションから、**フラットファイルタイプ** を [データマップ] ウィンドウの送り元エリアにドラッグしてください。
9. 送り元の [プロパティ] ペインで、ソースパスとして上記で取得したテキストファイルを使用してください。
10. [保存] をクリックします。

11. [データマップ] ウィンドウで、関連した接続を作成してください。複数のオブジェクトを追加するために、**Record** 要素を **BO** 要素と接続してください。

注 Magic xpi は複数のオブジェクトを格納した一つの XML ファイルを作成し、UserXML を更新します。Magic xpi はこのファイルを使用して SAPB1 DI へオブジェクトを追加します。UserXML コンテンツを PC に保存すると、XML ファイルを参照することができます。この情報は、プロジェクトのデバッグの時などに便利 k です。

XML コードのサンプルを参照するには、[こちら](#)をクリックしてください。

SAPB1 コネクタをアップグレードするには

Magic xpi は、アップグレードを行った後でも既存のコネクタで動作し続けることができます。たとえば、SAPB1 8.8 に接続するために SAPB1 2007 コネクタを使用し続けることもできます。しかし、新しいオブジェクトやデータベースタイプのように、新しい SAPB1 バージョンの新機能が必要な場合、コネクタの新しいバージョンを使用する必要があります。

SAPB1 コネクタをアップグレードすることを選択した場合、手動でステップを再現する必要があります。

SAPB1 フィールド内の値をクリアにするには

値を SAPB1 フィールドからクリアしたい場合、NULL 値を持つ関連要素(フィールド)で XML を作成する必要があります。結果 XML に NULL 値を作成するには、以下のようにする必要があります。

1. クリアしたい値の要素に **NULL** 値をマッピングしてください。
2. [データマップ送り先プロパティ] ダイアログボックスの [常にノード作成] チェックボックスを選択します。これによって、その要素のために空のノードが作成されます。

SAPB1 HANA を使用するには

SAPB1 HANA を使用するには、最初に以下を行う必要があります。

- SAP HANA DI API をインストールしてください。
- SAP HANA ODBC ドライバをインストールしてください。
- ODBC データソースを作成してください。
- SAP HANA トリガのために ODB 接続文字列を定義してください。

注 SAP HANA DI API をインストールすると、インストールされた場所から特定のバージョンだけを使用することができます。他の SAPB1 バージョンと接続することはできません。

SAP HANA ODBC ドライバをインストールする

データベースと接続することが必要な SAP HANA ODBC ドライバは、SAP HANA クライアントソフトウェアと一緒に提供されます。クライアントソフトウェアをダウンロードし、Magic xpi がインストールされた PC にインストールする必要があります。

ODBC データソースの作成

必要な ODBC ドライバをインストールしたら、接続したい各データベースのために専門の ODBC データソースを作成する必要があります。データソースは、データベースの名前とパスワードとユーザ名などの SAP HANA データベースを位置付け、アクセスするために使用する接続証明書を保持します。

Magic xpi 逆引き辞典

Windows PC 上で、ODBC リソースを作成する以下の手順に従ってください。

1. [スタート] メニューから、**コントロールパネル**と [管理ツール] を選択してください。
2. [管理ツール] ダイアログボックスで、[ODBC データソース(32 ビット)] をダブルクリックしてください。
3. [ODBC データソースアドミニストレータ] ダイアログボックスで、[システム DSN] タブをクリックしてください。
4. ここで、SAP HANA クライアントソフトウェアによってダウンロードした **HDBCODBC32** ドライバを使用して、32 ビット Windows のために ODBC データソースを作成してください。

注 意: 64 ビット OS を使用していて、[管理ツール]ダイアログボックスで ODBC データソース(32 ビット)オプションが表示されない場合、`%windir%\sysWOW64\odbcad32.exe` コマンドを実行して [ODBC データソースアドミニストレータ]ダイアログボックスを開いてください。

SAP HANA トリガのために ODB 接続文字列を定義する

SAP HANA トリガを使用したい場合、[ODBC データソースアドミニストレータ]ダイアログボックスで定義された、ODBC 接続文字列を取り出し、それを SAPB1 マルチ リソースリストの[接続プロパティ]カラムに入力する必要があります必要なフォーマットは、**<接続文字列>¥<db owner>**です。バックスラッシュ(¥)が入力されない場合、入力全体が接続文字列とみなされ、db owner の値は SAPB1 リソースの[Company Database]プロパティから取り出されます。

Since version: 4.5a

SAP R/3 コネクタ

SAP R/3 コネクタ

以下のトピックは、Magic xpi の SAP R/3 コネクタについて扱っています。

[Magic xpi を起動させるために SAP R/3 システムを設定するには](#)

[マスタデータ分布のために SAP R/3 システムを設定するには](#)

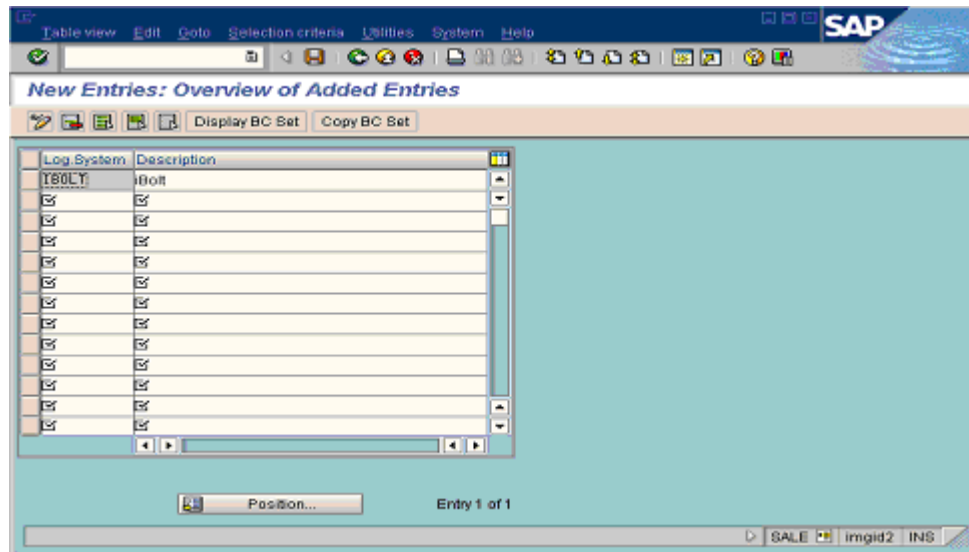
Magic xpi を起動させるために SAP R/3 システムを設定するには

SAP R/3 システムは、特定のイベントが発生すると論理システムに IDoc(Intermediate Document)を送るよう設定することができます。SAP R/3 ALE (Application Link Embedding)メカニズムは、IDoc を送信できるように SAP R/3 システムを設定する場合に使用されます。

SAP R/3 システムを設定するには：

1. 論理システムを定義する

SAP R/3 システムと Magic xpi 間の通信を可能にするためには、SAP R/3 システムに Magic xpi と SAP R/3 クライアントの論理システムを設定しなければなりません(SAP R/3 クライアントは通常定義済みです)。

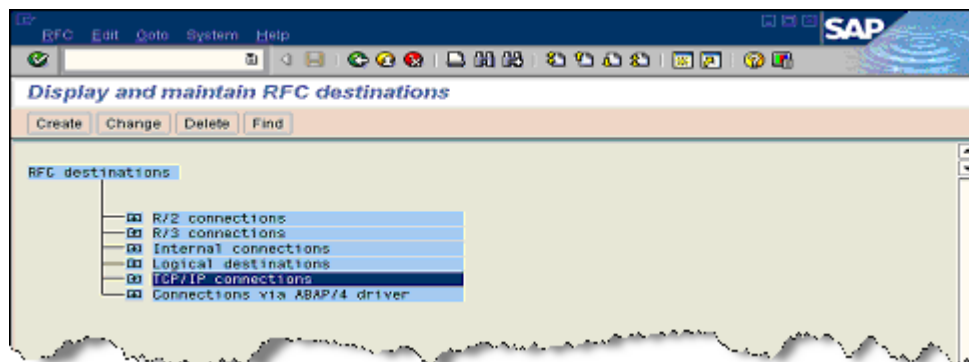


アプリケーションリンク *Enabling* > *Sending and Receiving Systems* > *Logical systems* > *Define Logical systems*.

代わりに 使用するトランザクション: *SALE*

2. RFC 接続先を定義する

RFC の接続先は、SAP R/3 から Magic xpi にデータを転送するために使用されます。

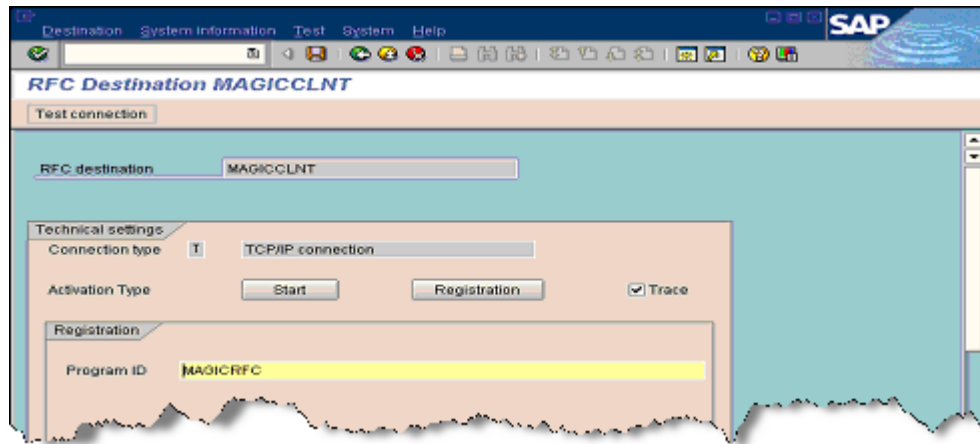


アプリケーションリンク *Enabling (ALE)* > *Sending and Receiving Systems* > *Systems in Network* > *Define Target Systems for RFC Calls*

代わりに 使用するトランザクション: *SM59*

以下の点に注意してください。

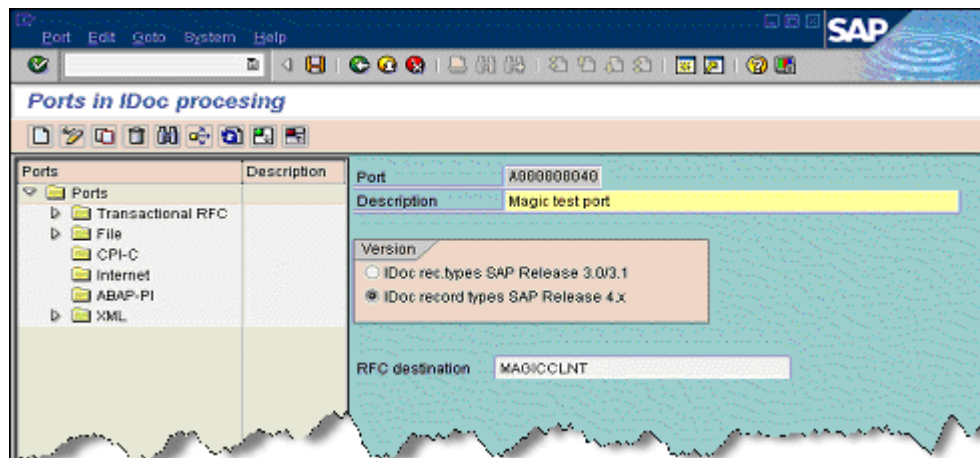
- RFC 接続先のコネクションタイプは、TCP/IP にしてください。
- 新しい RFC 接続先を定義する場合は、プログラム ID 値を必ず Magic xpi のトリガ設定の **Progid** 設定値と同じにしてください。



論理システムへの RFC 接続先が定義されました。

3. 論理ポートを定義する

低レベルネットワーク運用においては、RFC 接続先とシステムポート番号が関連付けられている必要があります。論理ポートは、Magic xpi ヘーダが送られるポートを定義します。論理ポートは RFC 接続先が既に作成されている場合のみ使用できます。

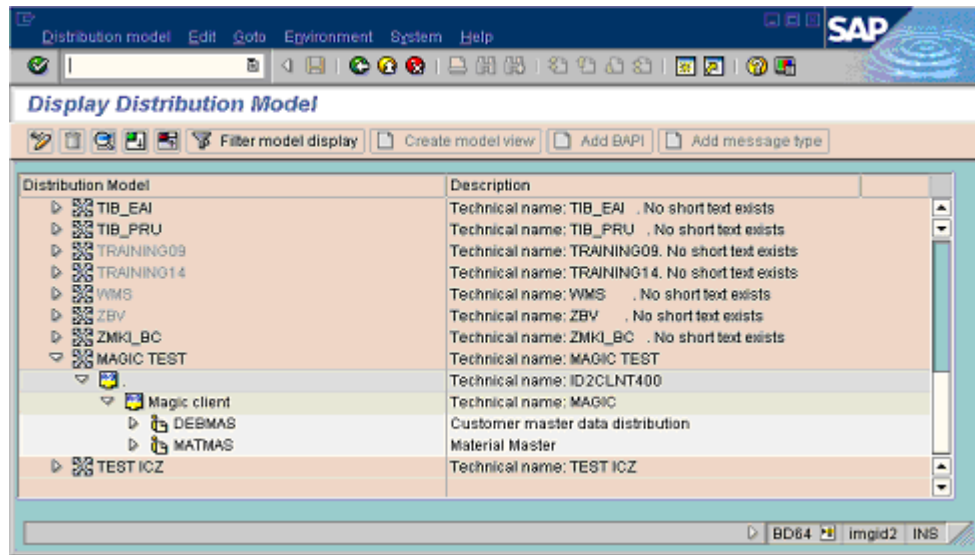


アプリケーションリンク *Enabling (ALE) > Sending and Receiving Systems > Systems in Network > Asynchronous Processing > Assigning Ports > Define Port*

代わりに 使用するトランザクション: WE21

4. 配布モデルを定義する

SAP R/3 システムと Magic xpi 間の IDoc 送信を有効にするには、配布モデルを定義する必要があります。配布モデルは、必要なメッセージタイプ全てを含んでいる必要があります。パートナプロファイルも作成する必要があります。

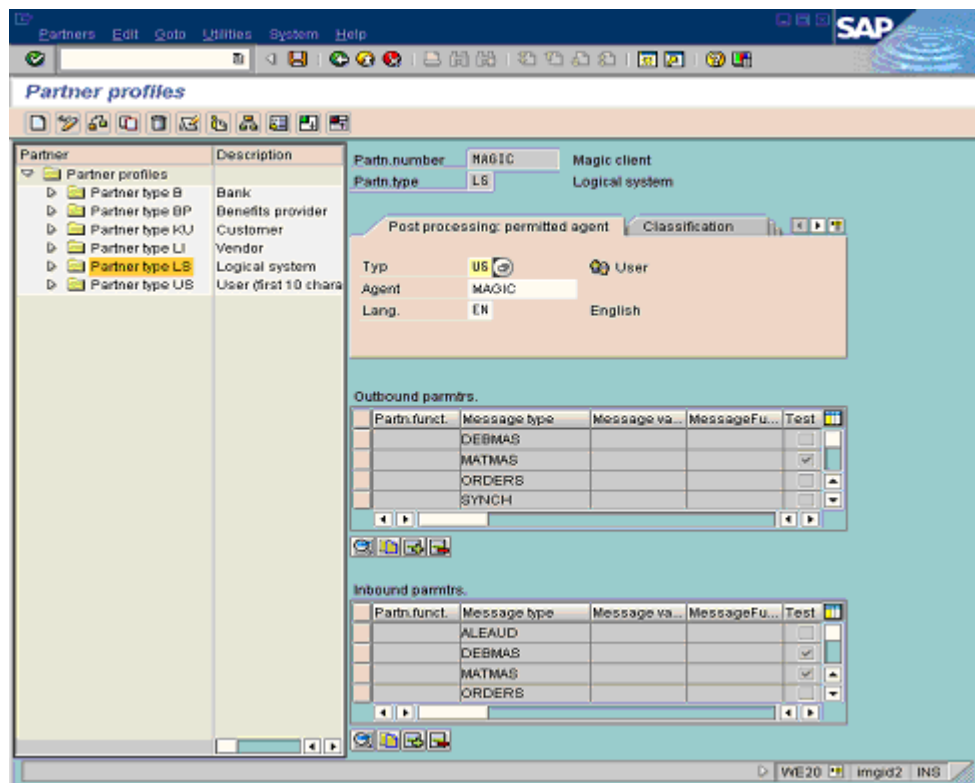


アプリケーションリンク *Enabling > Modeling and Implementing Business Process > Maintain Distribution Model and Distribute View*

代わりに 使用するトランザクション: BD64

5. パートナプロフィールを定義する

マスタデータ配布シナリオのための送出メッセージをコントロールするためのパートナプロフィールを作成します。



アプリケーションリンク *Enabling > Modeling and Implementing Business Process > Partner Profiles and Time of Processing > Maintain Partner Profile*

代わりに 使用するトランザクション: WE20

IDoc を手動で送信し、トリガ設定をテストする：

1. Magic xpi の SAP R/3 コネクタトリガを設定する

このトリガは、SAP R/3 システムから手動で送られる DEBMAS IDoc タイプのデータを受信待機するために使用されます。

The screenshot shows the 'SAP R/3 Trigger Configuration' dialog box. It has two main sections: 'Service Definition' and 'Result Definitions'. In the 'Service Definition' section, 'Service' is set to 'SAP R3_1'. Below this is a table with columns: 'Message Type', 'IDoc Type', 'Extension', and 'ALE Update Status'. The first row shows 'DEBMAS', 'DEBMAS04', and 'No'. There are 'New' and 'Delete' buttons at the bottom right of this section. The 'Result Definitions' section has two fields: 'Store trigger IDoc in:' with a dropdown set to 'C.UserBlob', and 'Store trigger information in:' with a dropdown set to 'String' and a field set to 'C.UserString'. There is a 'Refresh XSD' button and 'OK' and 'Cancel' buttons at the bottom.

Message Type	IDoc Type	Extension	ALE Update Status
DEBMAS	DEBMAS04		No

2. SAP R/3 システムを使用して DEBMAS メッセージを送信する

IDoc を確実に Magic xpi に送るために、パートナープロファイルの送付パラメーターに DEBMAS メッセージタイプが設定されているかを確認してください。

上記の設定ステップで定義した通りの適切な Magic xpi の論理システム名を入力してください。

The screenshot shows the 'Send Customers' dialog box in SAP. It has a menu bar (Program, Edit, Goto, System, Help) and a toolbar. The main area has several input fields: 'Customer' (with a search icon), 'Class' (with a search icon), 'Output type' (set to 'DEBMAS'), and 'Logical system' (set to 'MAGIC'). Below these is a 'Parallel processing' section with 'Server group' and 'No. of customers per process' (set to '20').

トランザクション: BD12

マスタデータ分布のために SAP R/3 システムを設定するには

このトピックは、マスタデータ配布のための SAP R/3 設定方法について説明しています。マスタデータが更新されると、配布モデルの定義に従ってマスタデータ IDoc を作成したり、Magic xpi に配布することができます。

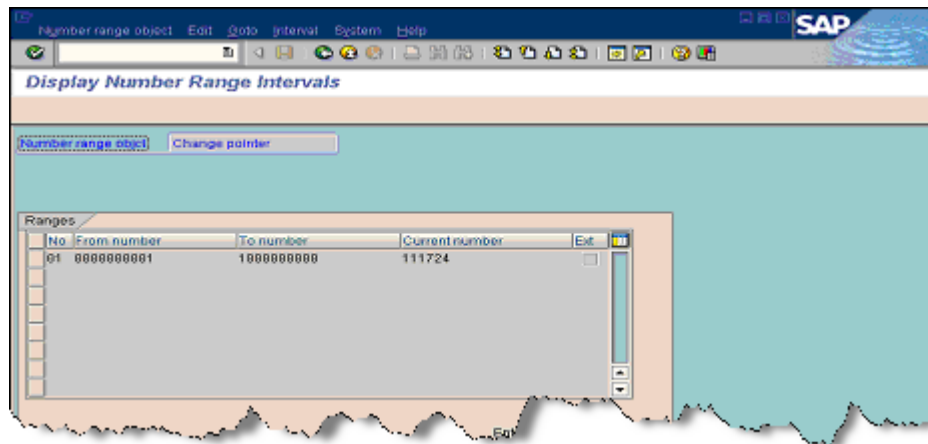
最初に [Magic xpi を起動させるために SAP R/3 システムを設定するには](#) のトピックを理解してください。

チェンジポインタは、shared master data(SMD)ツールに含まれるメカニズムにより SAP R/3 システムで管理されており、change document (CD)オブジェクトに基づいています。

SAP R/3 システムのマスタデータへの全ての変更は、change documents に記録されます。change documents から IDocs を生成するにはチェンジポインタを作成しなくてはなりません。IDocs はバックグラウンド処理でこれらの change pointers により作成されます。

チェンジポインタを設定するには：

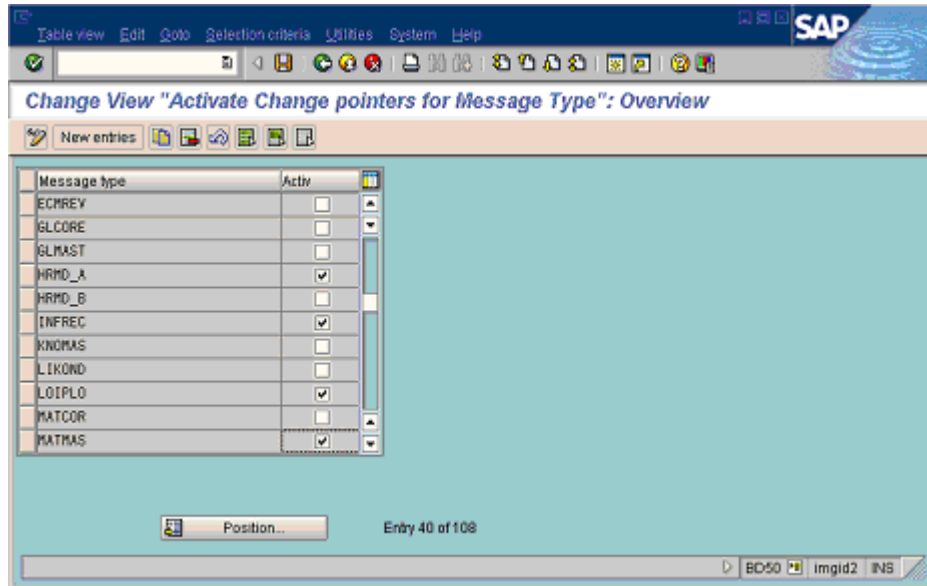
1. チェンジポインタの数値範囲を管理する



アプリケーションリンク Enabling > Master Data Distribution > Replication of Modified Data > Maintain Number Range for Change Pointers

2. メッセージタイプのチェンジポインタを有効にする

例えば、MATMAS (material master data) あるいは DEBMAS (customer master data) メッセージタイプのメッセージポインタ書き出しを開始できます。



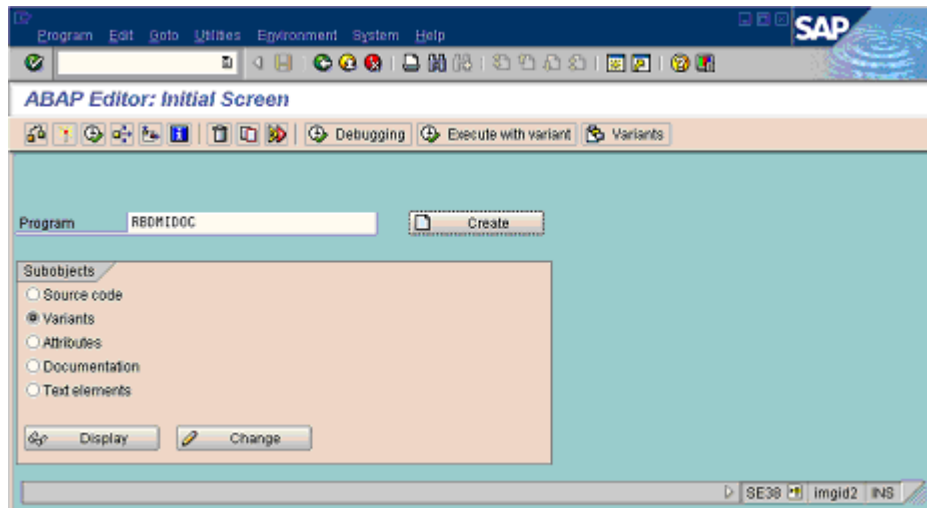
アプリケーションリンク *Enabling > Master Data Distribution > Replication of Modified Data > Activate Change Pointers for Message Types*

代わりに使用するトランザクション: BD50

3. レポートのバリエントを定義する

マスタデータ変更を配布するには、チェンジポイントを読み込むレポートをスケジュールし、それらから IDocs を作成する必要があります。

バリエント作成には、RGDMIDOC プログラムを指定します。バリエント作成は、特定メッセージタイプのために行われます。



アプリケーションリンク *Enabling > Master Data Distribution > Replication of Modified Data > Creating IDocs from change pointers > Define Variants*

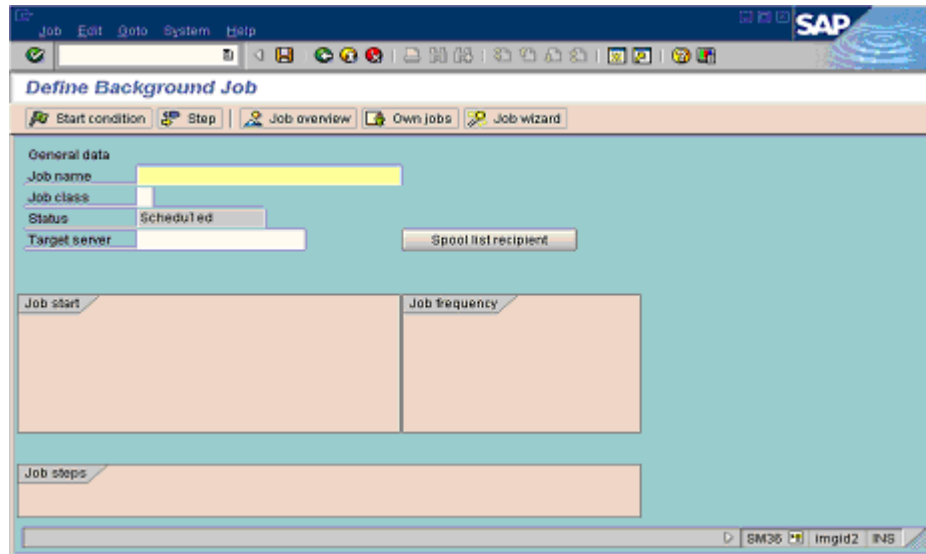
代わりに使用するトランザクション: SE38

4. スケジュールジョブ

マスタデータのための IDocs を定期的に変成するジョブをスケジュールしなくてはなりません。スケジュールは定義されたバリエーションごと、すなわちメッセージタイプごとに起動されます。

例えば、顧客マスタを配布したい場合、DEBMA5 メッセージタイプ用のジョブを計画しなくてはなりません。これは定期的に変成をチェックし、IDocs を生成します。

ジョブステップを定義する場合、**ABAP** プログラム名、**RBDMIDOC** を入力し、先ほどのステップで作成したバリエーションを選択してください。



アプリケーションリンク *Enabling > Master Data Distribution > Replication of Modified Data > Creating IDocs from change pointers > Schedule Job*

代わりに使用するトランザクション: **SM36**

サーバ

サーバ

以下のトピックは、Magic xpi のサーバについて扱っています。

[Magic xpi のサーバメッセージを管理するには](#)

[Magic xpi サーバ用のゲートウェイログを作成するには](#)

Magic xpi のサーバメッセージを管理するには

このトピックは、Magic xpi サーバのメッセージの変更方法を説明します。

ログ情報は、プロジェクトの設計やデバッグ中に必要となります。しかし、過剰な量の情報を出力しながらデバッグを行うと、システムのパフォーマンスを低下させてしまいます。このためログ情報は製品環境では減らすべきです。

解説:

Magic xpi サーバは、稼動中 `ifs_actlog` という名前の内部メッセージテーブルにメッセージを送ります。

Magic xpi モニタはこのテーブルを参照し、メッセージを表示します。

ユーザが設定を変更することにより特定のタイプのメッセージに対して以下の変更ができます。

1. メッセージが Magic xpi モニタに表示されない。
2. メッセージが **ifs_actlog** テーブルに書き込まれない。

メッセージが書き込まれないようにする

メッセージを内部メッセージテーブルに書き込まないようにする方法は三つあります。

A. ifs.ini ファイルの特定フラグを変更する

以下のフラグにより Magic xpi サーバから内部メッセージテーブルへ書き込まれるメッセージが既定されます。これらのフラグはプロジェクトの **ifs.ini** ファイルに設定します。

注意: ifs.ini ファイルの変更を有効にするにはサーバを再起動する必要があります。

[MAGIC_IBOLT]MonitorLogLevel

このフラグの値として、以下の値を指定することができます(カンマ区切り)。

User …… ユーザ定義メッセージを表示します。

Error …… エラー及びエラーコンポーネントの呼び出しを表示します。

Exec …… サーバやフローなどの起動を表示します。このスイッチは User、Error メッセージを含みます。

Service …… サービスを表示します(ODS, Lock, PSS)。

All …… 全てのメッセージを表示します(フラグが存在しない場合これがデフォルトです)。

None …… メッセージを表示しません。

[MAGIC_IBOLT]DisableActivityLog

値: Y or N

このフラグが **Y** になっている場合、動作ログは取られず、内部メッセージテーブルには何も書き込まれません。もちろんモニタにも何も表示されません。

[MAGIC_IBOLT]MonitorMode

このフラグの値として、以下の値を指定することができます。

Activity …… モニタは動作ログを表示しますが統計表示は行いません。

Statistic …… モニタは統計表示を表示し、動作ログは表示しません。このオプションはかなりの CPU 能力を消費します。

Both …… モニタは動作ログと統計表示両方を表示します。

B. モニタ経由のメッセージを停止する

注意: モニタへの変更は即時に反映されます。

1. モニタを開きます。
2. [設定] メニューから [アクティビティログ] を開きます。

3. [アクティビティログ] ダイアログの [サーバ設定] ボタンをクリックし、[サーバログ設定] ダイアログを開きます。

ここで選択されたメッセージはサーバから送られなくなります。

C. アクティビティログメッセージファイルを変更する

Magic xpi で、`Actlog_Write_Settings.xml` と `Actlog_Display_Settings_<user>.xml` ファイルにアクティビティログメッセージを格納します。

例

1. Magic xpi にエラーメッセージとユーザメッセージのみを送って欲しい場合、プロジェクト内の `ifs.ini` ファイルの `[MAGIC_IBOLT]MonitorLogLevel` フラグを以下のように変更してください。

`[MAGIC_IBOLT]MonitorLogLevel = Error , User`

2. もし **Flow component started** と **Flow component completed** を送信して欲しくない場合、`ifs_mytp` テーブルでこれらのメッセージを `Suspend` として選択するか、Magic xpi モニタで `suspend` を設定してください。

Magic xpi サーバ用のゲートウェイログを作成するには

Magic xpi サーバとプロジェクトのデータベース間で接続とデータ転送に関して詳細な状況を把握する必要がある場合があります。ゲートウェイのログを作成することによって、この状況を把握することができます。

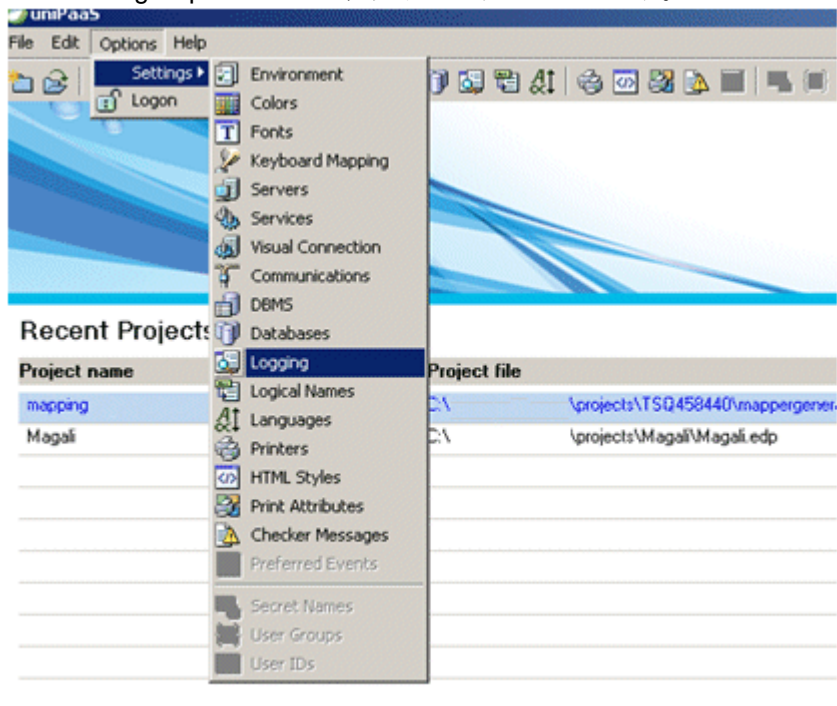
ゲートウェイのログを作成するには：

1. **Magic.ini** ファイルを開きます。そして、使用するデータベースゲートウェイ (.dll) が有効になっていることを確認してください。たとえば、Oracle ゲートウェイのログを有効にするには：
 - a. **Magic.ini** ファイルの `[MAGIC_GATEWAYS]` セクションで、`MGDB13=gateway\mgOracle.dll` を見つけてます。
 - b. 先頭のセミコロンを外してください：これは以下のようなイメージになります。

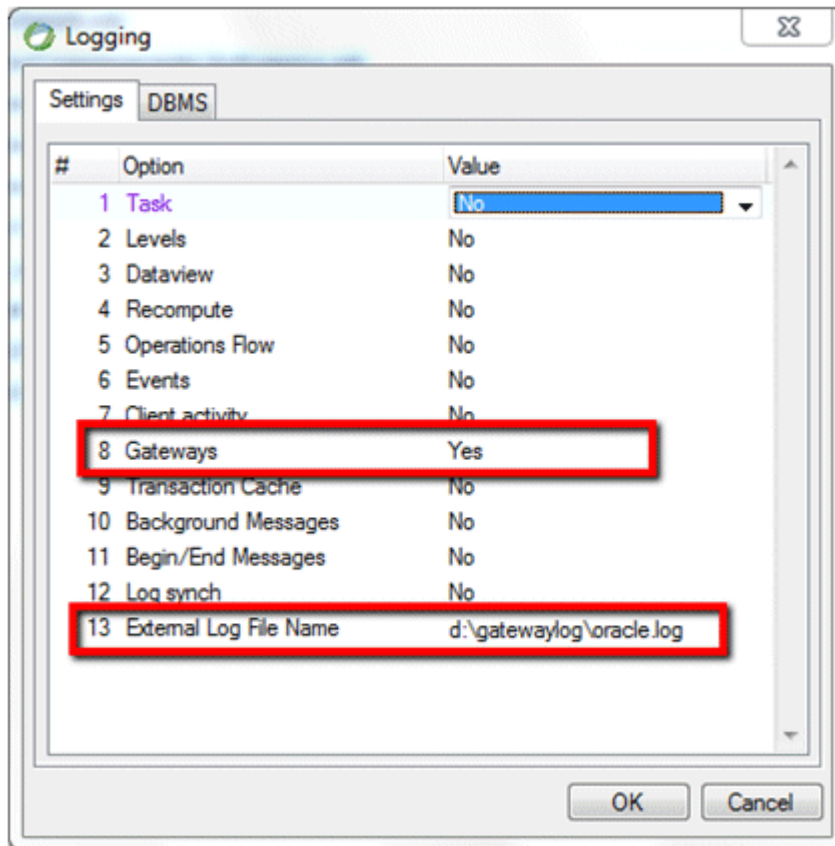
```
[MAGIC_GATEWAYS]
MGDB00=d:\          \Gateways\MGBtrieve.dll
MGDB01=d:\          \Gateways\MGPervasiveSQL.dll
;MGDB03=MGMySQL.dll
MGDB06=d:\          \Gateways\mgdb2400.DLL
MGDB09=mgsql32.dll
MGDB13=d:\          \Gateways\mgOracle.dll
MGDB15=d:\          \Gateways\mgOracle32.dll
;MGDB18=mgdb2.DLL
MGDB19=d:\          \Gateways\mgodbc.dll
MGDB20=d:\          \Gateways\mgmssql.dll
MGDB21=d:\          \Gateways\mgmemory.dll
```

2. Magic xpa Studio を起動します。これは、Magic xpi のインストール環境から利用できます。

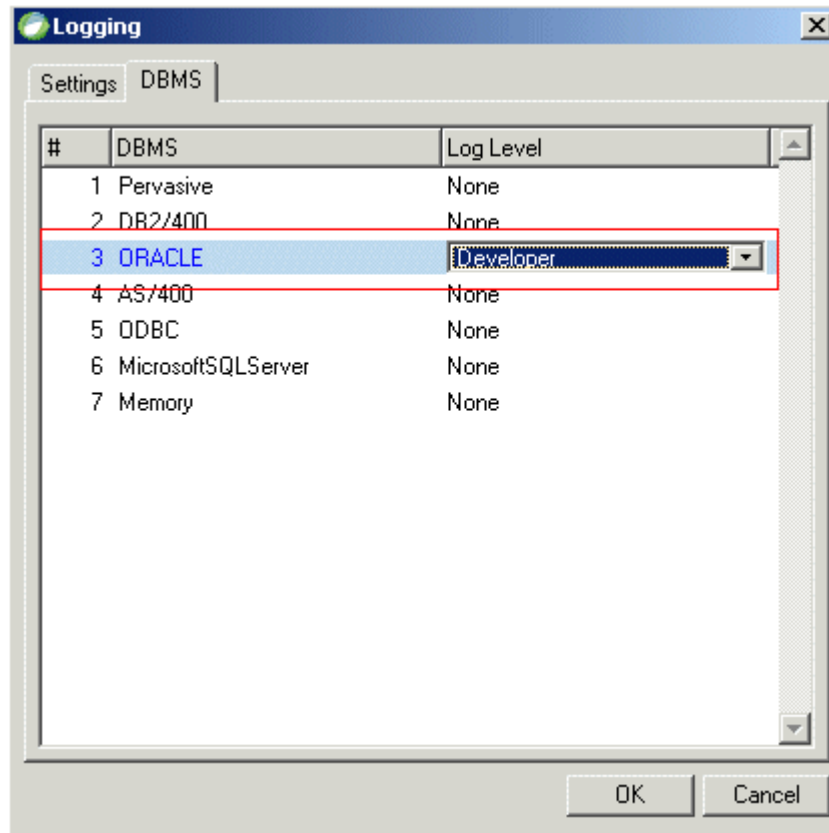
3. Magic xpa Studio で、以下のように選択します。オプション>設定>ロギング



4. [ロギング] ダイアログボックスの[設定]タブでは、ゲートウェイを **Yes** に設定します。そして、[外部ログファイル名] にファイル名を入力してください。残りのオプションが **No** に設定されることを確認してください。



5. [ロギング] ダイアログボックスの [DBMS] タブで、データベースのエントリ（例、Oracle）を**開発者**に設定してください。.



注意：次の3つのログレベルが利用できます：開発者、サポート、ユーザ。

6. [OK] をクリックして終了します。

7. 次に、**Magic.ini** ファイルで行われた変更内容を確認します。たとえば、以下のイメージのように、**Oracle** ゲートウェイのログは開発者レベルで設定されています。そして、**SQLite** はゲートウェイのログが設定されていません。

```
[MAGIC_DBMS]
Btrieve = 1, NotAllowNull, 10.3, Btrieve Parameters, N, , NotLogSync, 0, +
0, NotCheckExist,
Pervasive = 2, NotAllowNull, 10.3, Pervasive Parameters, N, , NotLogSync, +
0, 0, NotCheckExist,
MySQL = 4, NotAllowNull, 10.3, MySQL Parameters,,,0,0,,
DB2/400 = 7, NotAllowNull, 10.3, DB2/400 Parameters FORCE_TRANS, N, , +
NotLogSync, 35, 0, NotCheckExist, support\ACS400.eng
Oracle = 14, NotAllowNull, 10.3, Oracle Parameters, D, , NotLogSync, +
9999, 0, NotCheckExist,
AS400 = 17, NotAllowNull, 10.3, LANGUAGE=Support\AS21.eng NAMING=SQL
NOMSGBOX, N, , NotLogSync, 35, 0, NotCheckExist, support\ACS400.eng
DB2 = 19, NotAllowNull, 10.3, DB2 Parameters,,,,1,,
ODBC = 20, NotAllowNull, 10.3, ODBC Parameters, N, , NotLogSync, 0, 0, +
NotCheckExist,
MicrosoftSQLServer = 21, NotAllowNull, 10.3, MicrosoftSQL Parameters, N, +
, NotLogSync, 888, 0, NotCheckExist,
Memory Tables = 22, NotAllowNull, 10.3, INITFILE=1000, , NotLogSync, +
0, 0, NotCheckExist,
SQLite = 10, NotAllowNull, 10.3, SQLite Parameters, N, , NotLogSync, 0, +
1, CheckExist,
```

Magic.ini ファイルの **[MAGIC_LOGGING]** セクションでは、ログファイルのパス設定を確認することができます。:

```
[MAGIC_LOGGING]
Task = N
Levels = N
DataView = N
Recompute = N
Flow = N
Events = N
LogClient = N
TransCache = N
LogSynch = N
BeginEndMsg = N
Gateway = Y
BackgroundMsg = N
ExternalLogFileName = d:\gatewaylog\oracle.log
```

ソース管理

ソース管理

以下のトピックは、Magic xpi のソース管理について扱っています。

[ソース管理を使用して作業する環境を設定するには](#)

[ソース管理によって管理されるプロジェクトを作成するには](#)

[既存のプロジェクトをソース管理データベースに追加するには](#)

[プロジェクトをソース管理から削除するには](#)

[ソース管理プロバイダの定義、変更を行うには](#)

[オブジェクトを以前のコピーにロールバックするには](#)

[変更内容をトラッキングするには](#)



Magic xpi でソース管理製品として **TFS** を使用する場合はビデオデモンストラーションを参照するには、[ここ](#)をクリックしてください。

ソース管理を使用して作業する環境を設定するには

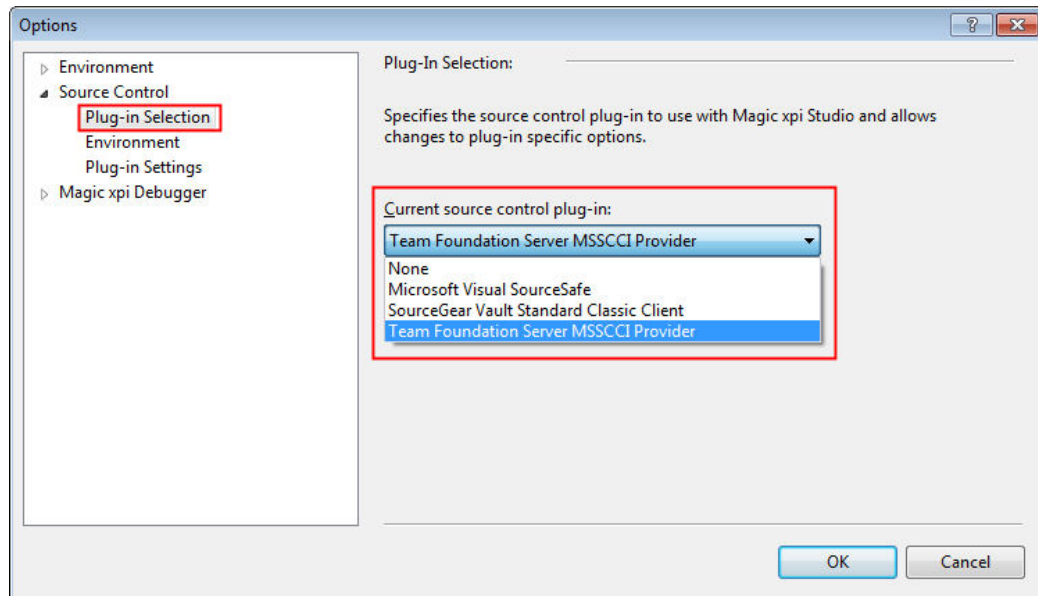
必要条件：

Magic xpi と一緒に以下を PC にインストールする必要があります。

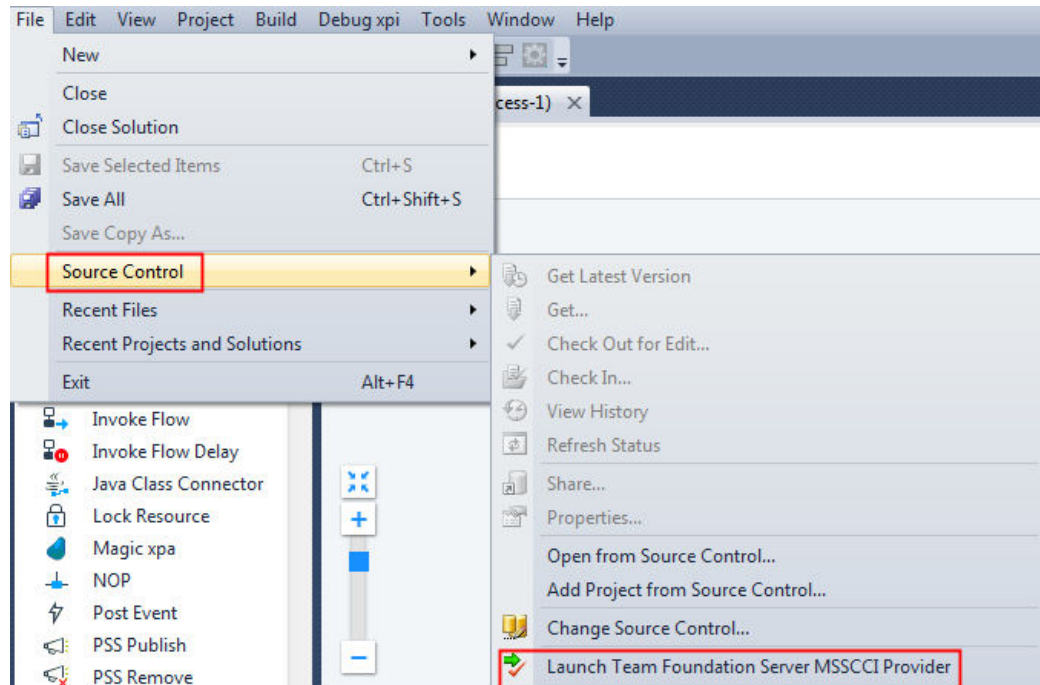
- TFS クライアント(クライアントか、Visual Studio2010 以上をインストールしてください)などのソース管理クライアント
- Visual Studio 用の Team Foundation Server MSSCCI Provider 32-bit プラグインのようなソース管理のための API

ソース管理を使用して作業を開始するには：

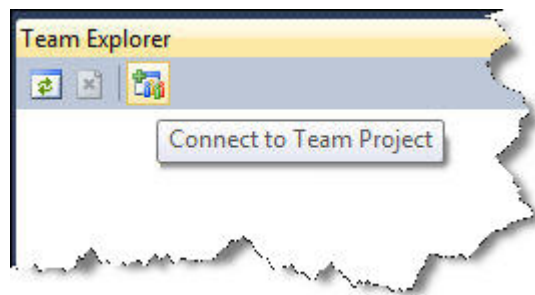
1. [ツール] メニューから、[オプション] を選択してください。
2. [オプション] ダイアログボックスで、[ソース管理] をクリックしてください。
3. [ソース管理] の下の、[プラグインの選択] をクリックしてください。
4. [現在のソース管理プラグイン] フィールドで、使用したいソース管理製品を選択し、[OK] をクリックしてください。



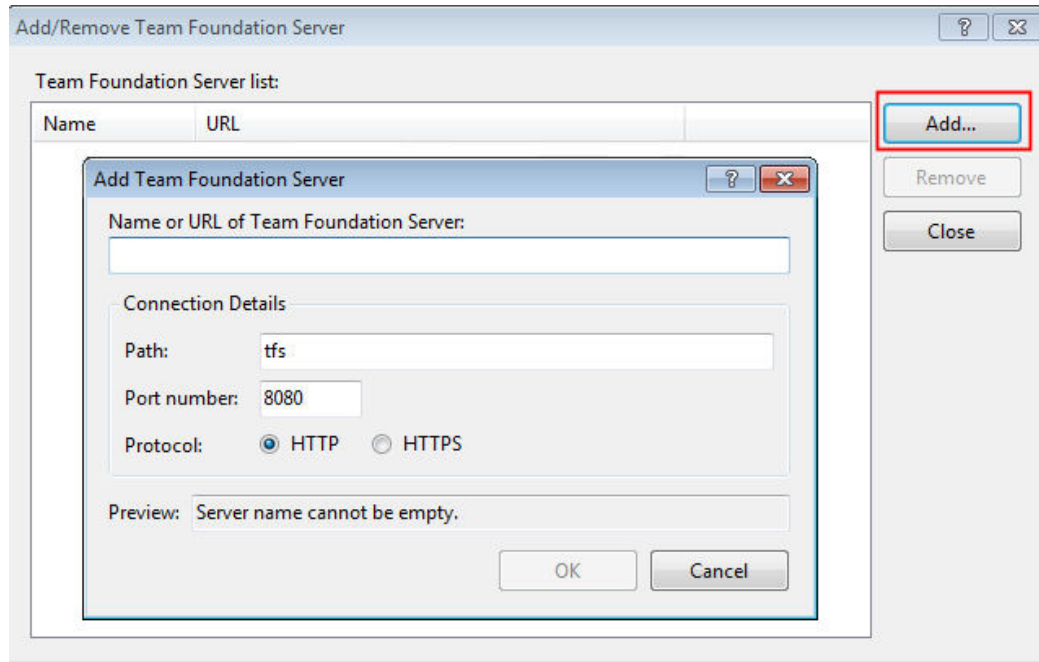
- 最初に、**Magic xpi** でソース管理を使用する場合は、[ファイル] メニューから [ソース管理] を選択し、次に [<ソース管理プロバイダ>を起動] を選択してください。以下のイメージでは、このオプションは [Team Foundation Server MSSCCI Provider を起動] になっています。



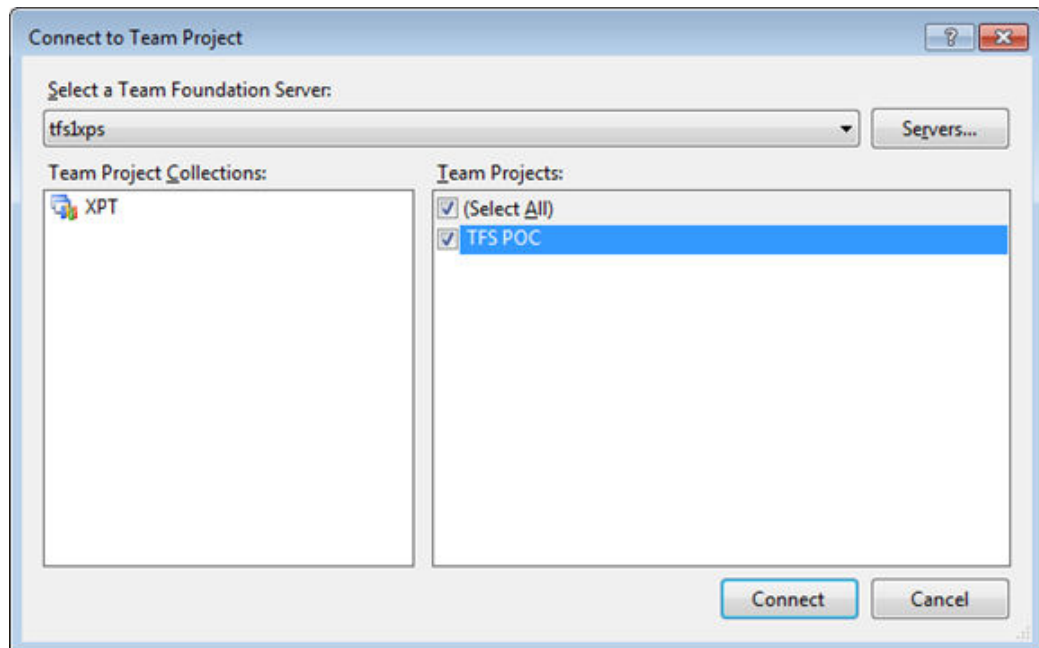
- ソース管理クライアントで、**チームエクスプローラ**を開き、[チームプロジェクトへの接続] アイコンをクリックしてください。



7. [追加] をクリックし、[Team Foundation Server の追加] ダイアログボックスで、Team Foundation Server の名前か URL を追加してください。



8. [OK] をクリックしてください。
9. [Team Foundation Server への接続] ダイアログボックスで、サーバとチームプロジェクトを選択し、[接続] をクリックしてください。



これで終了です。PC、ソース管理を使用して動作する用意ができています。

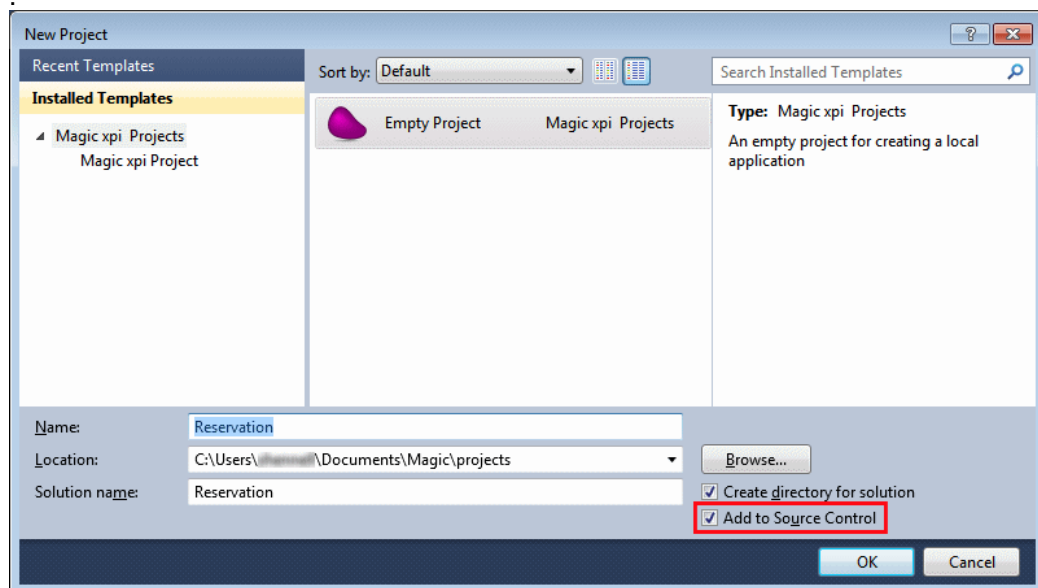
ソース管理によって管理されるプロジェクトを作成するには

Magic xpi で新しいプロジェクトを作成する時、最初からソース管理を使用して管理させるように選択することができます。

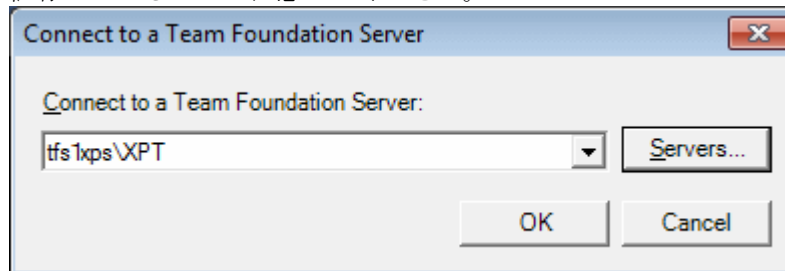
注意： ソース管理を使用して作業するには、PC のセットアップを行う必要があります。[ソース管理を使用して作業する環境を設定するには](#)を参照してください。

ソース管理によって管理されるプロジェクトを作成するには：

1. [ファイル] メニューまたはツールバーボタンのどちらかを使用して、通常の方法で新規プロジェクトを作成してください。
2. 通常の方法で [新規プロジェクト] ダイアログボックスを埋め、[ソース管理に追加] のチェックボックスをチェックしてください。

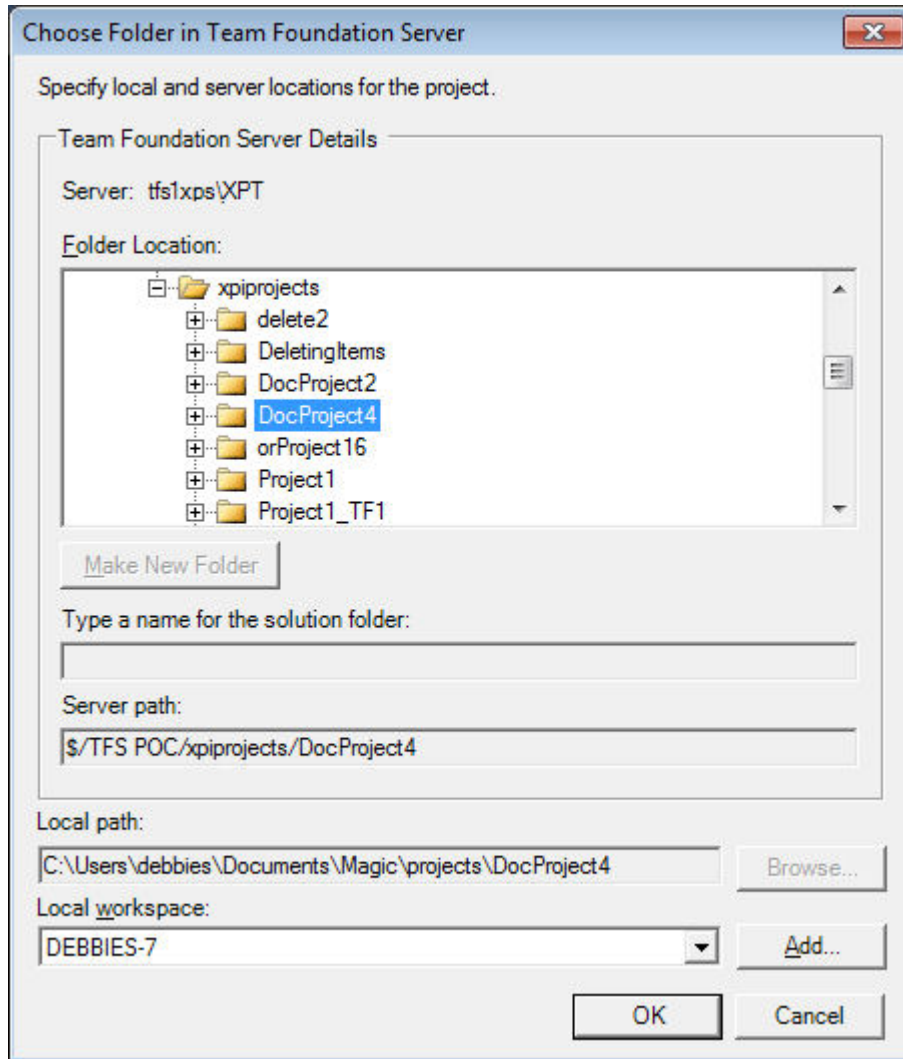


3. その時、使用したいソース管理データベースを定義するように促されます。以下のステップで表示される実際の画面とオプションは、使用するソース管理製品に依存していることに注意してください。

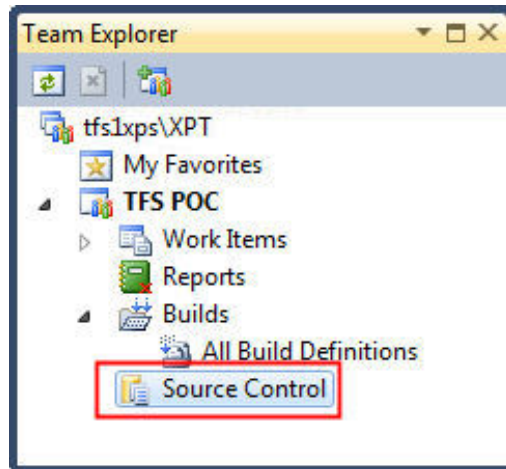


4. [OK] をクリックしてください。

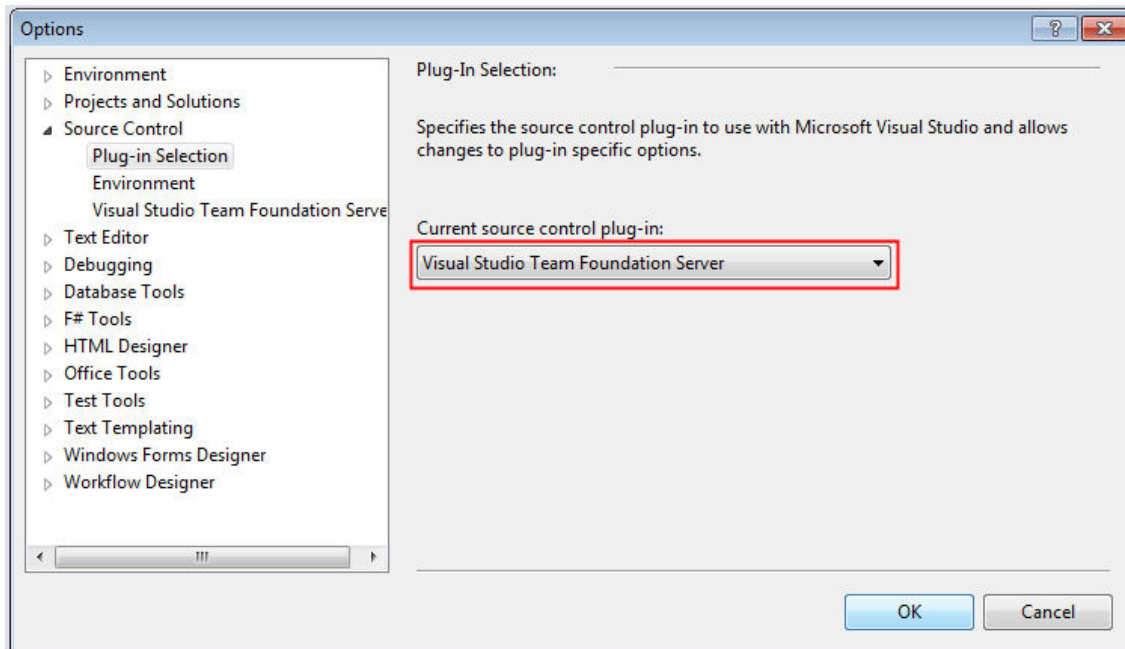
5. [Choose Folder in Team Foundation Server] ダイアログボックスでフォルダで、ソース管理データベース内のプロジェクトの位置を選択してください。



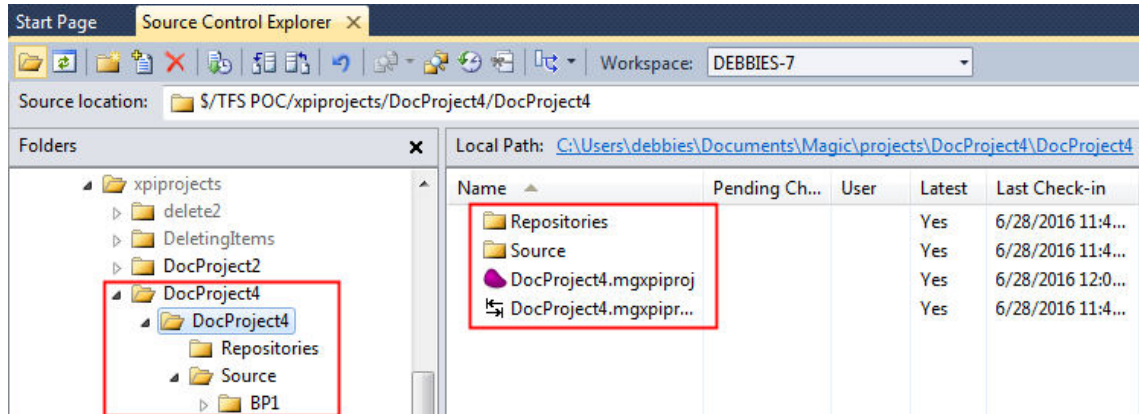
6. [OK] をクリックしてプロジェクトをソース管理に追加します。
7. プロジェクトがソース管理データベースに格納されると、それと関連するすべてのファイルとともにソース管理ツールに確認することができます。チームエクスプローラで、[Source Control] をクリックしてください。



8. ソース管理を選択していないというメッセージが表示されたら、**Visual Studio Team Foundation Server**を選択してください。



9. [OK] をクリックすると、チェックアウトした項目が表示されます。



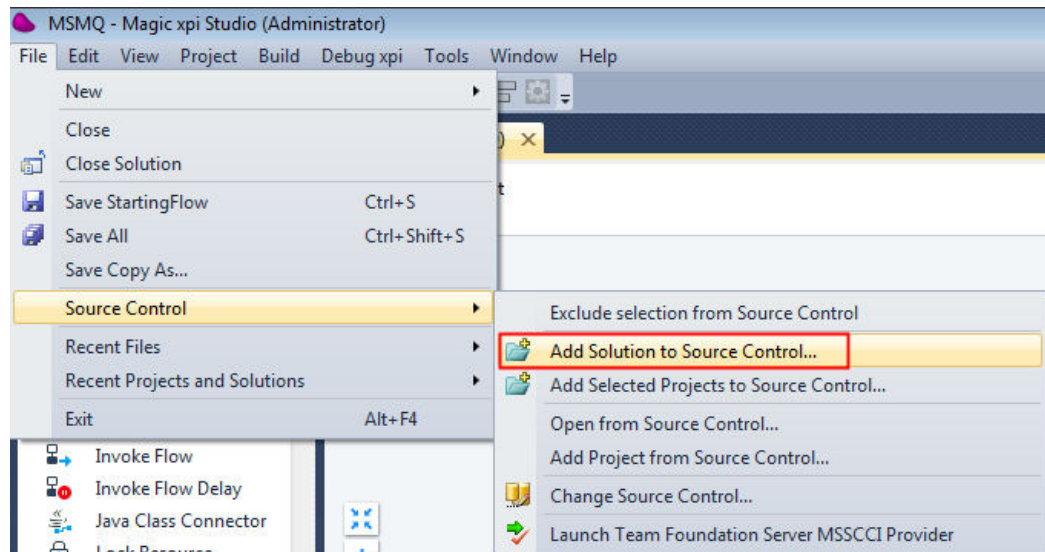
既存のプロジェクトをソース管理データベースに追加するには

すでに開発中のプロジェクトがある場合、いつでもソース管理データベースにそれをチェックすることができます。

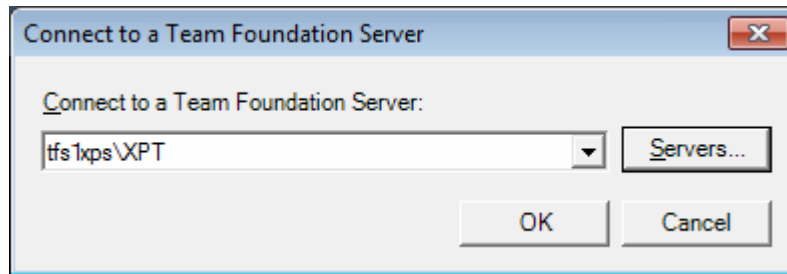
注 ソース管理を使用して作業するには、PC のセットアップを行う必要があります。[ソース管理](#)
意 : [を使用して作業する環境を設定するには](#)を参照してください。

既存のプロジェクトをソース管理データベースに追加するには :

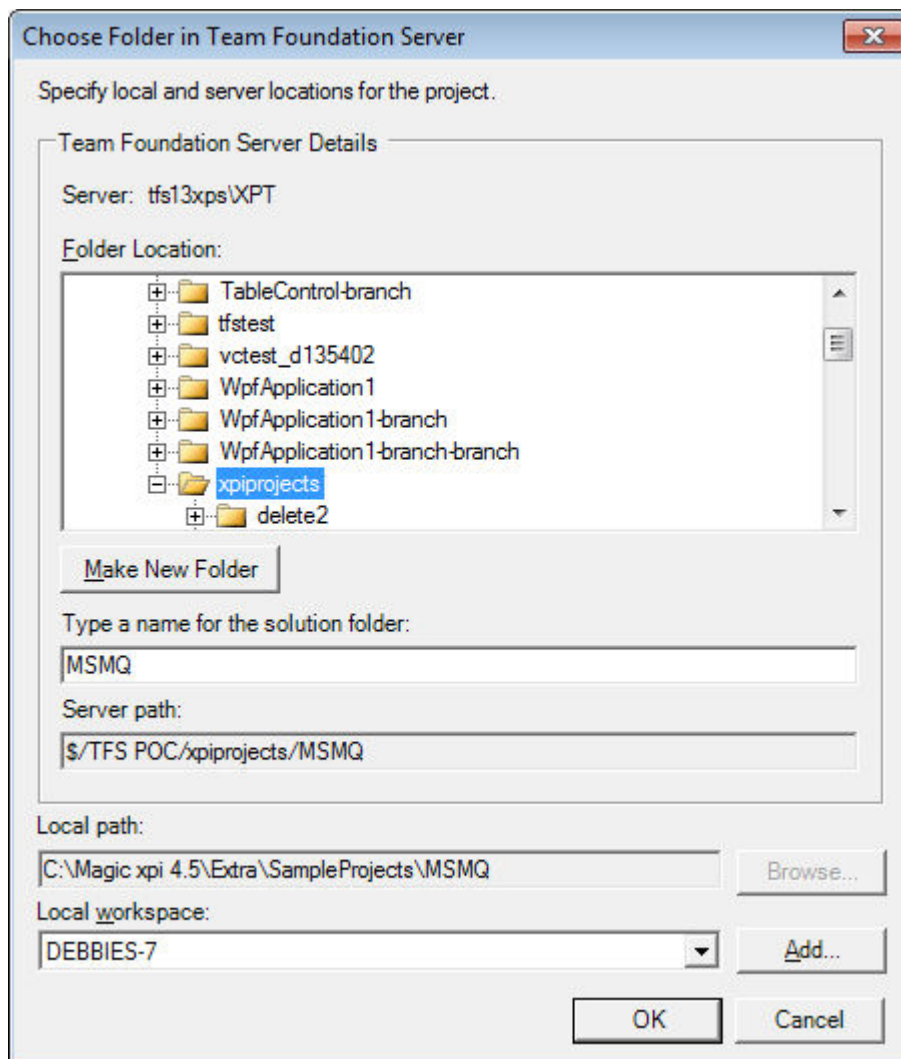
1. ソース管理コントロールの配下に置きたいプロジェクトを開いてください。
2. [ファイル] メニューから、[ソース管理]、次に [ソース管理にソリューションを追加] を選択してください。



3. その時、使用したいソース管理データベースを定義するように促されます。以下のステップで表示される実際の画面とオプションは、使用するソース管理製品に依存していることに注意してください。



4. [OK] をクリックしてください。
5. [Choose Folder in Team Foundation Server] ダイアログボックスでフォルダで、ソース管理データベース内のプロジェクトの位置を選択してください。

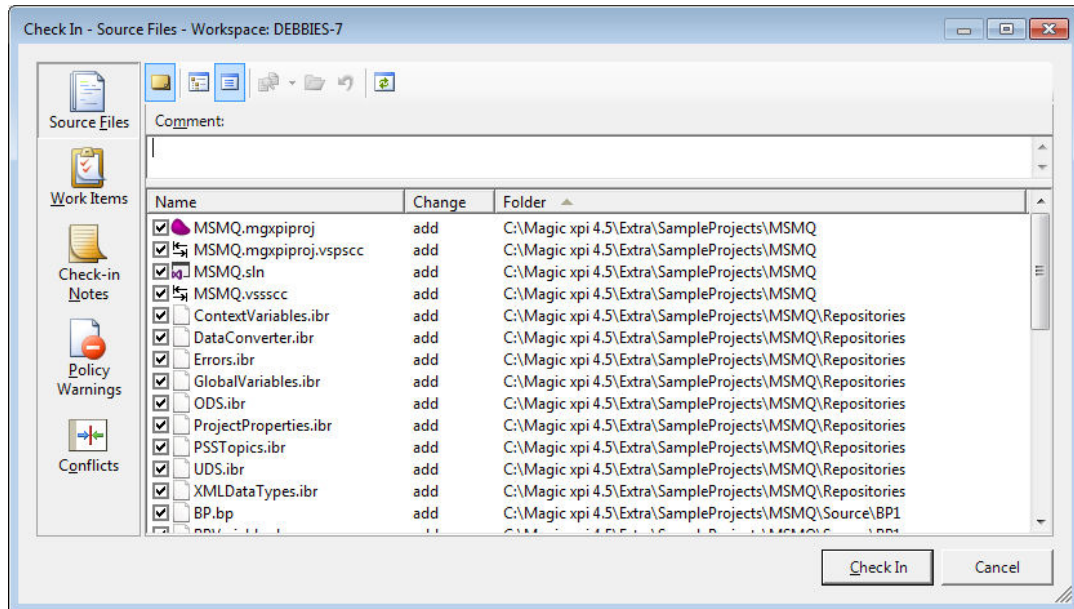


6. [OK] をクリックしてプロジェクトをソース管理に追加します。


Magic xpi 逆引き辞典

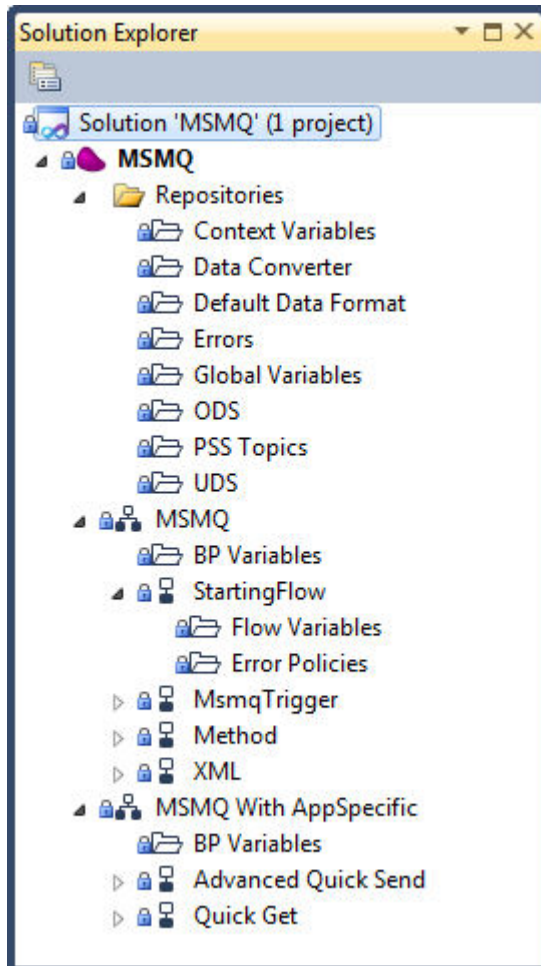
7. [チェックイン - ソースファイル] は、**Magic xpi** がデフォルトでソース管理内で処理する全てのソースファイルが表示されます。チェックボックスのすべてを選択した状態のままにして、[チェックイン] ボタンをクリックしてください。これで、**Magic xpi** がデフォルトでのソースコントロールにおいて、扱う全てのソースファイルをチェックインします。後で、他の項目を追加することができます。

注 意： **ifs.ini**、**Resources.xml**、および **Services.xml** などの **Magic xpi** のファイルは、ソース管理で処理されません。



これで項目はチェックインされ、使用したい項目があれば、最初にそれらをチェックアウトする必要があります。

8. Magic xpi のツールボックスでは、全ての項目がロックされている状態(隣に  ロックアイコンが表示されます)で表示されます。



プロジェクトをソース管理から削除するには

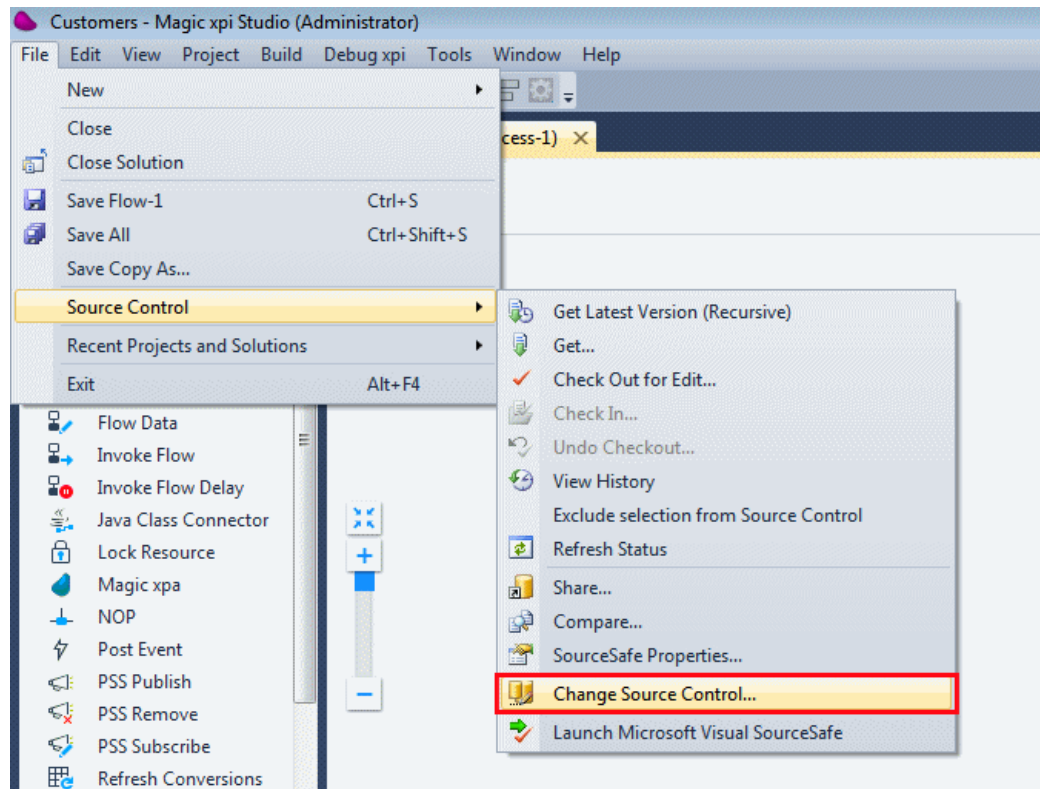
特定のプロジェクトでソース管理を使用しないことを決めた場合、それをソース管理サーバから削除することができます。

注 以下のステップで表示される実際の画面とオプションは、使用するソース管理製品に依存しています。ご注意ください。

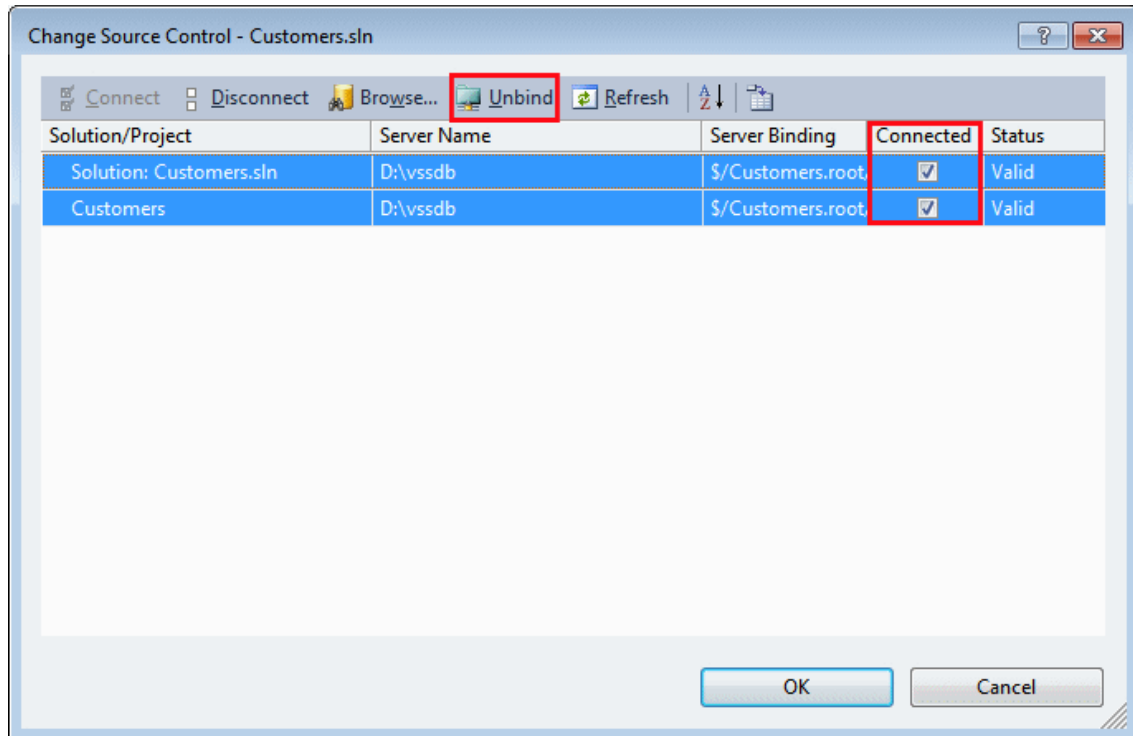
プロジェクトをソースコントロールから削除するには：

Magic xpi 逆引き辞典

1. [ファイル] メニューから、[ソース管理]、そして[ソース管理の変更]を選択してください。



2. [ソース管理の変更] ダイアログボックスで、ソース管理から削除したいソリューションやプロジェクトを選択し、[解除] をクリックしてください。



3. [OK] をクリックして除去プロセスを完了します。

注 このよにプロジェクトを解除しても、そのコピーはまだソース管理サーバ上に残っています。
注意: 再びソース管理の使用を開始する場合は、このコピーを削除する必要があります。

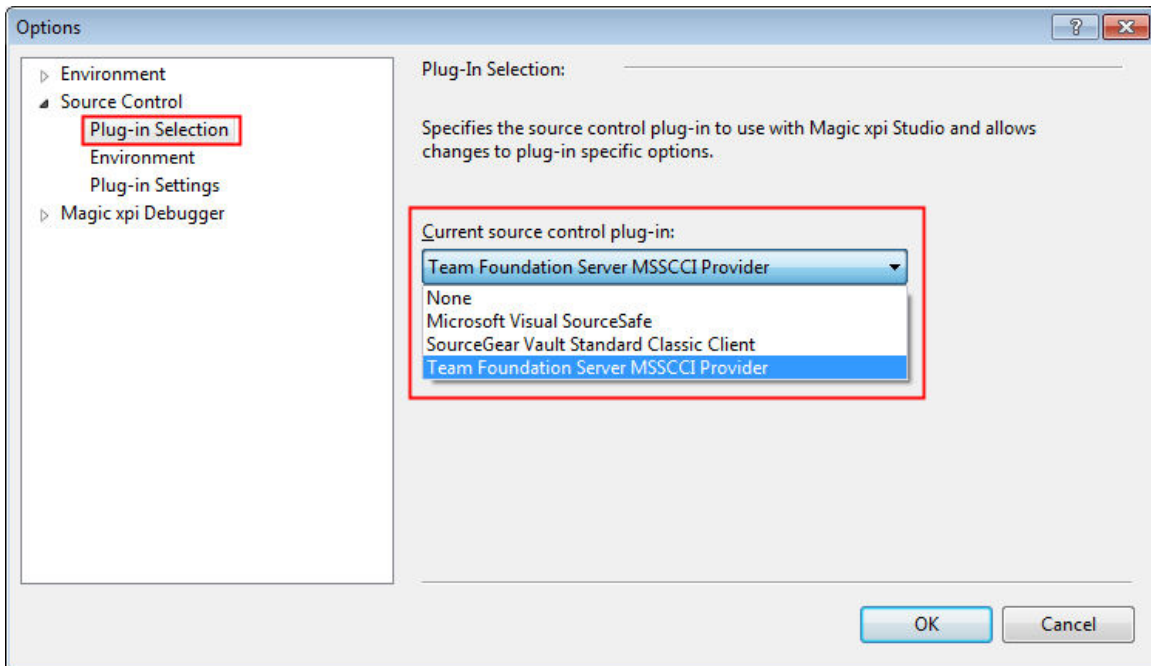
ソース管理プロバイダの定義、変更を行うには

特定のプロジェクトで使用されているソース管理製品は、Magic xpi の[オプション]ダイアログボックスに表示されています。他のソース管理製品を PC にインストールして、そちらに切り換えたい場合は、ソース管理製品を変更することもできます。

プロジェクトによって使用されるソース管理製品を定義したり、変更するには：

1. [ツール] メニューから、[オプション] を選択してください。
2. [オプション] ダイアログボックスで、[ソース管理] をクリックしてください。
3. [ソース管理] の配下で、[プラグインの選択] をクリックしてください。

[現在のソース管理プラグイン]フィールドで、プロジェクトが使用しているソース管理製品が表示されています。必要に応じて、ドロップダウンリストから異なるソース管理製品を選択し、[OK]をクリックしてください。

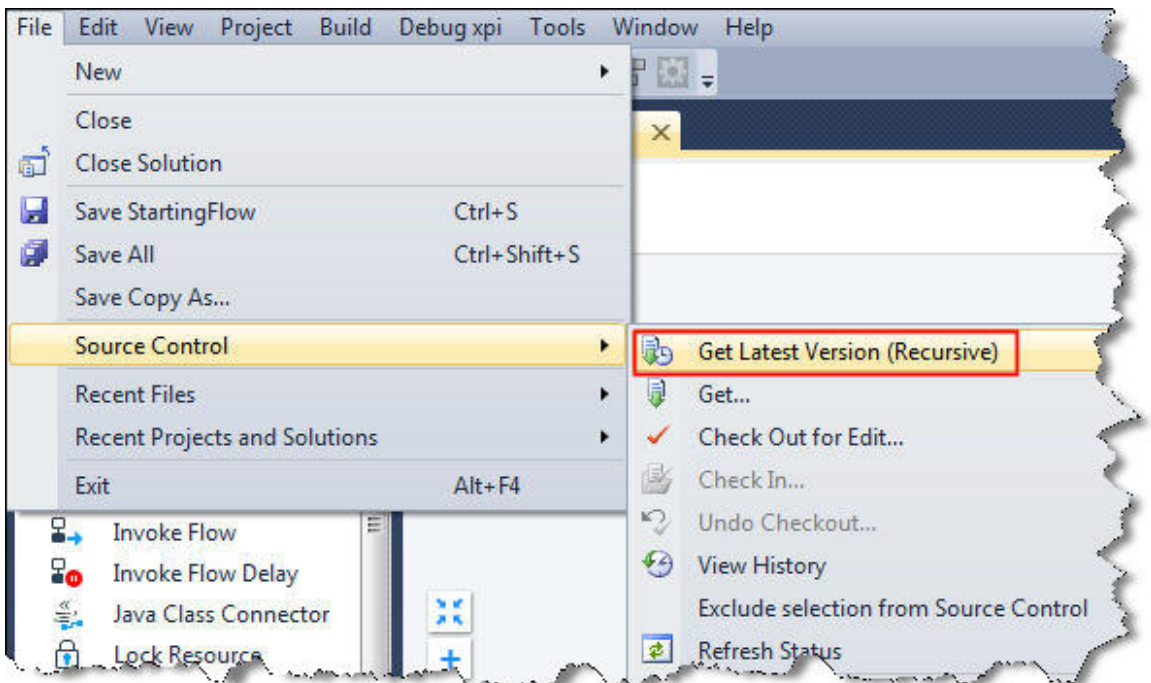


オブジェクトを以前のコピーにロールバックするには

コンテキストメニューの「ロールバック」は、オブジェクトを以前のバージョンに戻すことを意味しています。

オブジェクトの最新バージョンを取得するには：

- オブジェクト上でパークし [ファイル] を選択してください。[ソース管理] から [最新バージョンの取得(再帰)] を選択します。



これで、そのオブジェクトのバージョンの中で以前にチェックインされたものを取得します。

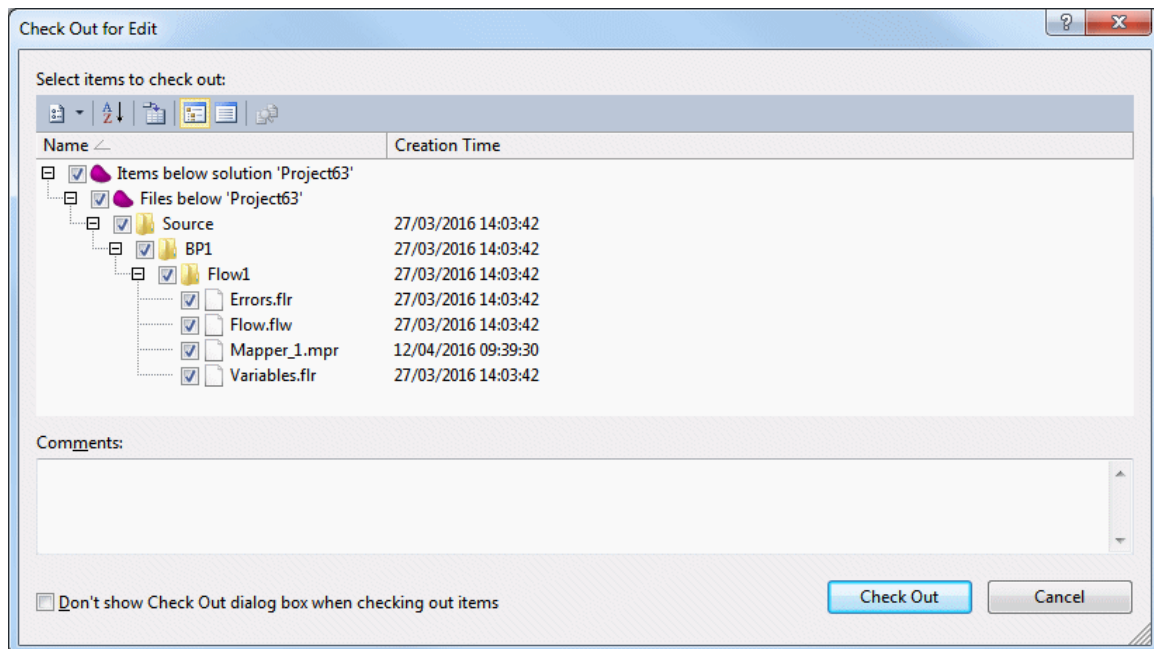
変更内容をトラッキングするには

Magic xpi のアイテムでチェックアウト／チェックインするたびに、ソース管理ツールは、何が変更されたかを記録しています。それらの変更がなぜ発生したかについて短い注意書きを入力することができます(そして、そうすべきです)。

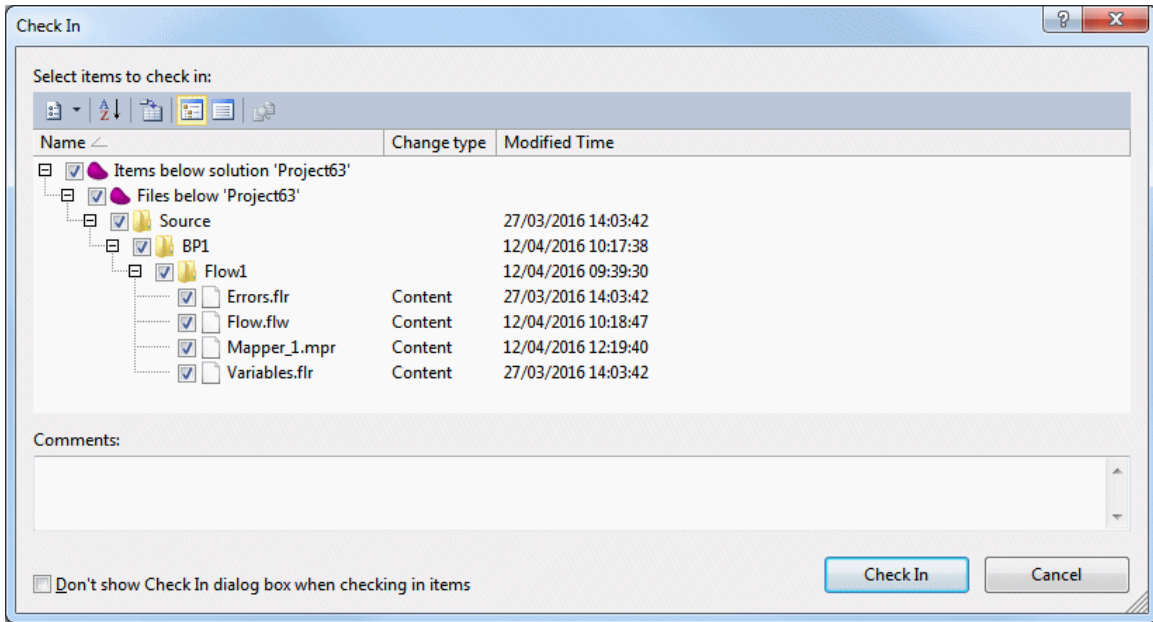
さて、どのような変更を行ったか、いつ、そしてどのような理由かを確認したい時、ソリューションエクスプローラの関連する項目に位置付けてください。そして、[ファイル]メニューから[ソース管理]そして[履歴表示]を選択してください。結果は、どのソース管理ツールを使用しているかに依存します。

項目のチェックインとチェックアウト

項目に対する処理を行いたい場合、[ファイル]メニューから[ソース管理]、そして[編集のためにチェックアウト]を選択します。コメントの入力が要求され、項目は編集用に開きます。



終了したら、[ファイル]メニューから[ソース管理]、そして[チェックイン]を選択します。再び、コメントの入力要求が表示されます。

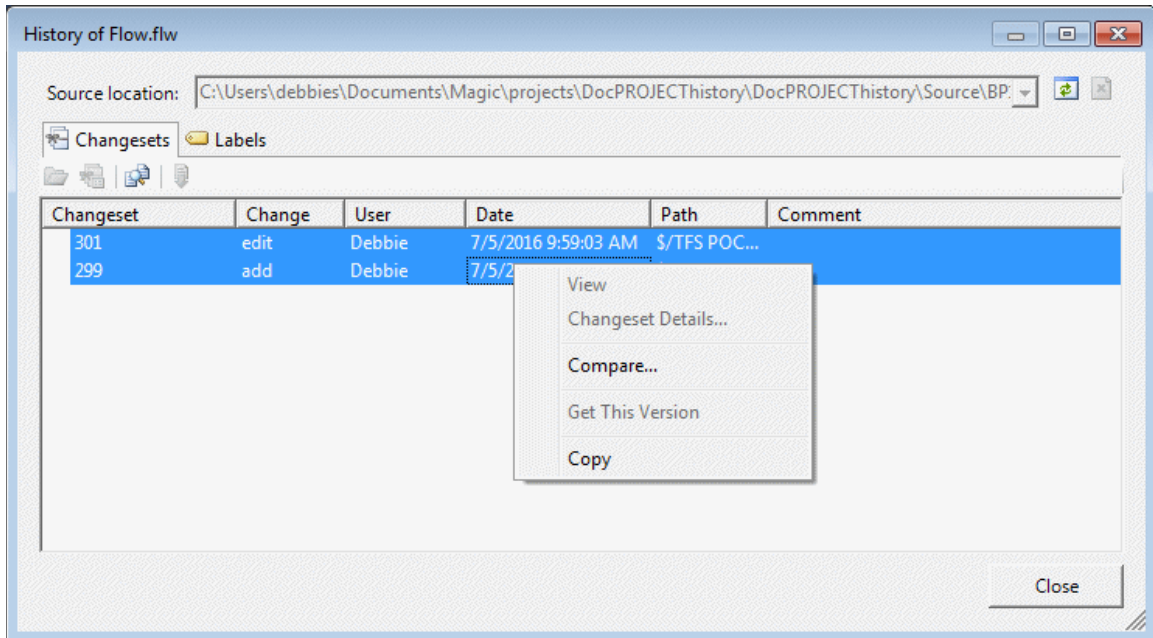


これらのコメントは、オブジェクトの変更履歴に表示されます。

履歴

どのような変更を行ったか、いつ、そしてどのような理由かを確認したい場合は、[ファイル]メニューから[ソース管理]そして[履歴表示]を選択してください。

変更内容のリストが表示されます。TFS の場合は、2 つの変更内容を選択することができ、右クリックで[コンペア]を選択します。



ここで、比較された2つのファイルが並列して表示されます。変更や追加、削除された行が色分けされます。

20	<HasDBTransaction>false</HasDBTransaction>	20	<HasDBTransaction>false</HasDBTransaction>
21	<DBTransactionString />	21	<DBTransactionString />
22	<RecoveryPolicy>6</RecoveryPolicy>	22	<RecoveryPolicy>6</RecoveryPolicy>
23	<isSecuredCall>false</isSecuredCall>	23	<isSecuredCall>false</isSecuredCall>
24	<ErrorFlow>	24	<ErrorFlow>
25	<BPID>0</BPID>	25	<BPID>1</BPID>
26	<FlowID>	26	<FlowID>
27	<ExpectedType>N</ExpectedType>	27	<ExpressionVal>PCMHJUAkPmdldEZsb3d0YW11KDIPCMHJUAkPg==</ExpressionVal>
28	<Direction>I</Direction>	28	<ExpectedType>N</ExpectedType>
29	</FlowID>	29	<Direction>I</Direction>
30	</ErrorFlow>	30	</FlowID>
31	<LogicFlow>	31	</ErrorFlow>
32	</BPID>	32	<LogicFlow>
33	</BPID>	33	<BPID>0</BPID>

Space

Space

以下のトピックは、Magic xpi の Space 機能について扱っています。

[Magic xpi 4.x のために Unicast Discovery の設定を行うには](#)

[1 台の PC 上のサーバの複数インスタンスを実行するには](#)

[モニタから新規サーバインスタンスを追加するには](#)

[モニタからサーバインスタンスをシャットダウンするには](#)

[Salesforce を使用してプロジェクトをモニタするには](#)

Magic xpi 4.x のために Unicast Discovery の設定を行うには

注 ユニキャストは、一つの開発 PC で動作している時を含むすべての配備シナリオで使用し
意： なければなりません。

ユニキャストは、ルックアップサービス (LU) が指定した PC 上で位置付けられ、PC のホスト名または IP アドレスを知っていることを意味します。このトピックは、Magic xpi 4.x で動作するためにユニキャストディスカバリの設定方法について説明しています。

ユニキャストディスカバリの設定は、手動でも自動的にでも行うことができます。

ルックアップ環境の手動設定

手動でユニキャストディスカバリの設定を行うには、以下に示すような、ファイル内で同じ Lookup Group エントリと Lookup Locator エントリの定義を含むいくつかのファイルを必要とします。

1. **magicxpi-setenv.bat** ファイル (以下のフォルダ内にあります : **<Magic xpi のインストールフォルダ>\Runtime\Gigaspaces\bin**)
 - a. **MAGIC_MULTICAST_ENABLED** エントリを **False** に設定します。
 - b. **LOOKUPLOCATORS** エントリにルックアップサービス (LU) を実行するホストに設定してください。これらの値は、カンマで区切ってください。
 - c. **LOOKUPGROUPS** のエントリの値はそのままにしてください。
2. **Magic.ini** ファイルの **[MAGICXPI_GS]** セクション
 - ・ **LookupLocators** エントリをルックアップサービス (LU) を実行するホストに設定してください。
 - a. **LookupLocators** エントリが **Magic.ini** ファイルに存在しない場合、それを作成してください。
 - b. **LookupGroupName** のエントリの値はそのままにしてください。

3. **Magic.ini** ファイルと同じ設定 (手順#2)で、**Mgreq.ini** ファイル (<Magic xpi のインストールフォルダ>\Runtime フォルダ上)を設定してください。
4. **Magic.ini** ファイルと同じ設定 (手順#2)で、インターネットリクエストの **Mgreq.ini** ファイル (scripts\config フォルダ上)を設定してください。
5. Systinet は、**Mgreq.ini** ファイル (<Magic xpi のインストールフォルダ>\Runtime フォルダ上)から、設定設定を使用します。**Mgreq.ini** ファイルの位置は-**Dcom.magicsoftware.requester.conf** という JVM プロパティによってコントロールされます。そして、それは%WASP_HOME%\bin\Install_GSA_service.bat ファイルと%WASP_HOME%\bin\Server.bat ファイルにあります。**Install_GSA_service.bat** ファイルを変更した場合、以下のステップを行わなければなりません。
 - a. Systinet サービスをアンインストールしてください。これを行うには、管理者としてコマンドプロンプト (CMD)を開いてください。そして、%WASP_HOME%\bin に移動してください。次に、**Uninstall_GSA_service.bat** ファイルを実行してください。.
 - b. Systinet サービスをインストールしてください。これを行うには、管理者としてコマンドプロンプト (CMD)を開いてください。そして、%WASP_HOME%\bin に移動してください。に、**Install_GSA_service.bat** ファイルを実行してください。

注意： サービスのアンインストールやインストールを行うには、管理者権限が必要です。

グリッドサービスエージェント (GSA)を設定する

ユニキャストディスカバリ・モードで実行すると、LUS は **LookupLocators** エントリに定義された PC 上でローカルに開始されなければなりません。残りのインストール処理は、指定した LUS を示すための設定を行う必要があり、その結果、開始しないようにしてください。

1. グリッドサービス・エージェント (GSA)が LUS を読み込むため、GSA を設定する必要があります。これは、**magicxpi-gs-agent.bat** ファイルで行います (以下のフォルダにあります: <Magic xpi のインストールフォルダ>\Runtime\Gigaspace\bin)。
LUS を開始するインストール処理を設定するには、**magicxpi-gs-agent.bat** ファイルに、以下の変更を行ってください。
 - a. Notepad でファイルを開いてください。
 - b. **call gs-agent.bat** で開始される行をコメント化してください。行の先頭に **Rem** を追加することでコメントになります。
 - c. **any machine with lus service** から開始されるコメント行に位置付けてください。
 - d. call 処理からテキストをコピーして、スクリプトの最後にペーストしてください。

行は次の通りでなければなりません: **call gs-agent.bat gsa.gsc 2 gsa.global.gsm 1 gsa.lus 1 gsa.global.lus 0 gsa.mgdeploy 1**

gsa.lus 1 要素は、LUS が 1 インスタンスから開始されなければならない、GSA によってローカルで管理されなければならない、グリッド上でグローバルでないことを示しています。

gsa.global.lus 0 要素は、グローバルな LUS が開始してはいけないことを示しています。

2. LUS を使用するだけのようにインストール処理を設定するには、それを開始することなく、**magicxpi-gs-agent.bat** ファイルに以下の変更を行います。
 - a. Notepad でファイルを開いてください。

- b. **call gs-agent.bat** で開始される行をコメント化してください。行の先頭に **Rem** を追加することでコメントになります。
- c. **any machine without lus service:** から開始されるコメント行に位置付けてください。
- d. **call** 処理からテキストをコピーして、スクリプトの最後にペーストしてください。

行は次の通りでなければなりません: **call gs-agent.bat gsa.gsc 2 gsa.global.gsm 1 gsa.lus 0 gsa.global.lus 0 gsa.mgdeploy 1**

gsa.lus 0 要素は、ローカルの LUS が開始してはいけないことを示しています。**gsa.global.lus 0** 要素は、グローバルな LUS が開始してはいけないことを示しています。

1 台の PC 上のサーバの複数インスタンスを実行するには

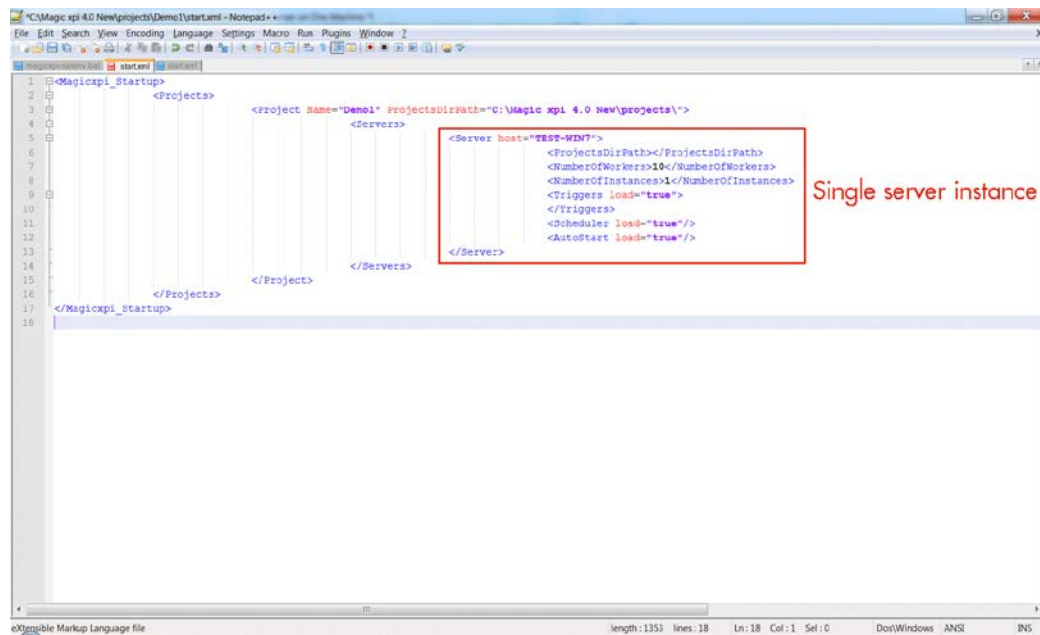


1 台の PC 上のサーバで複数のインスタンスを実行するデモを参照する場合は、[ここ](#)をクリックしてください。

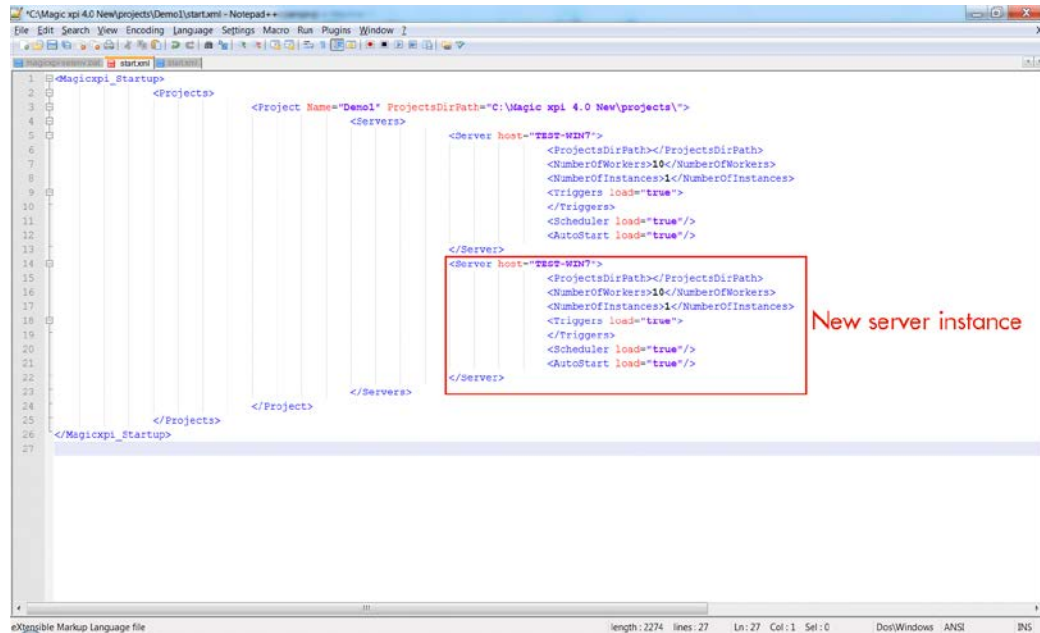
Magic xpi サーバのプロセス(MagicxpiServer.exe)は、1 つの OS プロセスです。プロジェクトで処理される大量のデータがある場合、1 つのプロセスがそのリソースを使い果たした場合に、複数の OS プロセスで処理を分割したら、より速くなるかもしれません。

1 台の PC 上のサーバで複数のインスタンスを実行するには：

1. テキストエディタを使用して **start.xml** ファイルを開いてください。このファイルは、**<My Documents>\Magic\projects\<プロジェクト名>\<current project>** フォルダにあります。
2. **start.xml** ファイルでは、以下のイメージで示すように、一つのサーバインスタンスが定義されています。



- 次に、以下のイメージで示すように、このサーバインスタンスをコピーして、既存のインスタンスの下に新しいサーバインスタンスとして、ペーストしてください。



- 次に必要に応じて新しいサーバインスタンスの **NumberOfWorkers** と **NumberOfInstances** の値を変更することができます。**Triggers**、**Scheduler** と **Autostart** の値を **True** または **False** に設定します。
- 必要な値を設定したら、**start.xml** ファイルを保存して閉じてください。
- プロジェクトを開始すると、モニタでサーバのインスタンス数を確認することができます。

注 意： 異なる PC 上でプロジェクトのサーバプロセスのための複数インスタンスを実行させることもできます。これによって、リニアなスケーラビリティと同様に、より多くのメモリと CPU が利用できるようになります。より多くのリソースをプロジェクトに追加することができて、唯一の共有リソースがネットワークのため(プロジェクトは異なる PC 上で実行するため、すべてのエンジンは各 PC の CPU とメモリを利用します)にこれが可能になります。

モニタから新規サーバインスタンスを追加するには



モニタから新しいサーバインスタンスを追加するデモを参照する場合は、[ここ](#)をクリックしてください。

モニタから複製のサーバインスタンスを追加するには：

- プロジェクトを作成してビルドしてください。
- モニタを開いてプロジェクトを開始してください。
- モニタから、[サーバ] タブを開いてください。
- サーバの詳細がタブに表示されます。

5. サーバエントリで右クリックし、コンテキストメニューから [新規インスタンスの実行] を選択してください。[新規サーバインスタンス] ダイアログボックスが開きます。
6. [新規サーバインスタンス] ダイアログボックスのサーバ詳細は、既存サーバと同じです。必要な変更を行って、[OK] をクリックして新しいサーバインスタンスを作成してください。
7. [リフレッシュ] をクリックして、新しいサーバの詳細を確認してください。

モニタから異なる情報を持つサーバインスタンスを追加するには：

1. プロジェクトを作成してビルドしてください。
2. モニタを開いてプロジェクトを開始してください。
3. モニタから、[サーバ] タブを開いてください。
4. [ナビゲーション] ペインのサーバエントリで右クリックし、コンテキストメニューから [サーバ追加] を選択してください。[新規サーバインスタンス] ダイアログボックスが開きます。
5. [新規サーバインスタンス] ダイアログボックスで必要なサーバ詳細を入力し、[OK] をクリックしてください。
6. [リフレッシュ] をクリックして、新しいサーバの詳細を確認してください。

モニタからサーバインスタンスをシャットダウンするには



モニターからサーバインスタンスをシャットダウンするデモを参照するには、[ここ](#)をクリックしてください。

モニタからサーバインスタンスをシャットダウンするには：

1. プロジェクトを作成して、ビルドしてください。
2. モニタを開いて、プロジェクトを開始してください。
3. モニタから、このプロジェクトの [サーバ] タブを開いてください。
4. 終了させたいサーバを選択してください。
5. このサーバ上で右クリックし、コンテキストメニューから [シャットダウン] を選択してください。[シャットダウンパラメータ] ダイアログボックスが開きます。
6. [OK] をクリックして [シャットダウンパラメータ] ダイアログボックスを閉じます。
7. [リフレッシュ] をクリックします。サーバのステータスは、**STOP_REQUESTED** に変更されます。
8. もう一度 [リフレッシュ] をクリックします。サーバは停止します。
9. **Windows のタスクマネージャー**を開いて、[プロセス] タブを選択します。停止したサーバは、リストには表示されないはずです。
10. モニターの [プロジェクトビュー] 画面で、サーバのうちの 1 台が実行していないという警告メッセージが表示されます。これは、サーバを停止したからです。
11. モニタを開いて、[ナビゲーション] ペインの [サーバ] タブを選択してください。

12. たった今停止したサーバ上で右クリックし、コンテキストメニューから [エントリの消去] を選択してください。
13. [リフレッシュ] をクリックします。サーバがリストから消えます。
14. モニタの [プロジェクトビュー] 画面で、[リフレッシュ] をクリックしてください。リストからのシャットダウンされたサーバを削除したため、警告は表示されなくなります。

Salesforce を使用してプロジェクトをモニタするには

Magic xpi の モニタユーティリティは、Magic xpi クラスタの動作状況をの高水準なビューを表示します。

Magic xpi の環境をモニタし、実行プロジェクトを表示し、警告を受け取り、様々なイベントに関するチャット通知を受け取ることができます。このトピックは、モニタユーティリティのセットアップ方法や使用方法について説明しています。

Salesforce モニタユーティリティを設定するには：

1. Salesforce AppExchange から Magic xpi アプリをインストールしてください。インストールの方法についての情報は、[ここ](#)をクリックしてください。
2. Magic xpi のインストールフォルダで、**SFDC_Monitor_Utility** フォルダを開き、**SFDC_MonitorSetup.exe** ファイルをダブルクリックしてください。[Salesforce モニタ構成] ダイアログボックスが開きます。このダイアログボックスには、以下のプロパティが含まれています。
 - **Environment Name** …… Salesforce において環境を表すために使用する名前。
 - **Environment ID** …… ランダムに作成された 30 文字の文字列が表示される読み込み専用フィールドです。これはユニークな環境キーとして Salesforce で使用されます。
 - **Endpoint URL** …… Salesforce 組織のログインエンドポイント URL。このフィールドのデフォルト値は：**https://login.salesforce.com.**
 - **Proxy Host:Port** …… 実行部分とモニタユーティリティの構成部分の両方で使用されるプロキシ定義。デフォルト値は、空白です。.
3. 必要なパラメータを入力したら、OAuth 認証フローを完了する必要があります。これを行うには、[OAuth2] ボタンをクリックしてください。
4. [OAuth2View] ウィンドウで、Salesforce の証明書を入力してください。証明書が受け入れられたら、緑色のチェックマークが [Salesforce モニタ構成] ダイアログボックスの [OAuth リフレッシュトークンステータス] フィールドの隣に表示されます。
5. [検証] ボタンをクリックして Salesforce との接続を確認してください。

Salesforce モニタユーティリティを使用するには：

1. Magic xpi プロジェクトを実行してください。
2. Magic xpi のインストールフォルダで、**SFDC_Monitor_Utility** フォルダを開き、**start.bat** ファイルをダブルクリックしてください。これによって、Salesforce に Magic xpi プロジェクトデータを送り出すランタイムユーティリティが起動します。Windows OS のサービスとしてユーティリティを実行したい場合は、最初に、<Magic xpi インストール>\SFDC_Monitor_Utility\OS_Service フォルダに置かれている、**SFDCserviceInstall.bat** ファイルを実行する必要があります。これで、手動または自動的に実行する **MgcxpiSFDCUtil** サービスを作成します。

3. 証明書を使用して <https://login.salesforce.com> にログインしてください。
 4. [環境] タブをクリックしてください。このタブが有効でない場合は、それを追加することができます。これを行うには、**AppSetup** エリアに移動し、[Create] をクリックしてください、次に、[Tabs] を選択してください。
 5. [Environment Name] の下の **Recent Environments** エリアで、セットアップ手順(上記)で作成された環境の名前が表示されます。
 6. Environment Name のリンクをクリックすると、モニタ情報が表示されます。
 7. [Environment Detail] 画面は、環境の名前やステータスなどの情報を表示し、また、専用のセクションで以下のデータを表示します。
 - **Space Information**
 - **System Information**
 - **Alerts Report** …… このセクションは、以下の情報を表示するチャートを表示します。
 - レコード数
 - 重大度レベル
 - **Alerts** …… このセクションは、以下の問題についての情報を表示します。
 - **checkNotAllServersAreRunning:** …… モニタ、**start.xml** ファイル、または **projectsStartup.xml** ファイルによって開始されたサーバが動作していない。
 - **checkTriggerNotRunning:** …… プロジェクトがトリガを使用しているが、トリガなどを読み込んでいないサーバが存在しない。
 - **checkIfPrimaryAndBackupOnSameMachine:** …… プライマリとバックアップのパーティションが同じ PC 上にない。
 - **checkMessagesWaitingMoreThan:** …… メッセージが指定された時間に処理されない。
 - **addGSAAlerts:** …… メモリと CPU の消費問題などの GigaSpaces の内部の警告が発生した。
 - **Projects**
 8. [Environment Detail] 画面の [Projects] セクションで、特定のプロジェクト名をクリックすると、以下の情報が表示されます。
 - **Project Detail**
 - **Activities**
 - **System Information**
 - **Alerts Report**
 - **Alerts**
- 以下のようにすることで、Salesforce モニタユーティリティのリフレッシュレートを変更することができます。
1. [Magic xpi 設定] タブをクリックしてください。
 2. [Magic xpi 構成] 画面で、[編集] をクリックしてください。
 3. [インターバル (分)] プロパティで、必要なリフレッシュレートに設定してください。
 4. [保存] をクリックしてください。

Web サービス

Web サービス

以下のトピックは、Magic xpi の Web サービスについて扱っています。

[Web サービスのトリガを使用するには](#)

[Web サービスのコンシューマとして Magic xpi を設定するには](#)

[統合 Windows 認証によって保護されている Web サービスを呼び出すには](#)

[Magic xpi 中の Web サービスに基本認証を提供するには](#)

[WS-Security で動作している Web サービスプロバイダとして、Magic xpi を設定するには](#)

[HTTP コンポーネントを使用して Web サービス呼び出しを行うには](#)

[手動で Systinet Server for Java をインストールするには](#)

[Magic xpi で Web Services Over SSL のコンシューマとして使用するには](#)

[IIS7 で証明書を処理して、Systinet に SOAP リクエストを送るには](#)

Web サービスのトリガを使用するには

Web サービスのトリガを使用するには

Web サービストリガから Magic xpi を起動させ、Magic xpi を Web サービスプロバイダとして機能させるには、以下のステップを行ってください。

- インストールとセットアップ
- Magic xpi プロジェクトの [ビルド](#)
- Web サービスユーティリティの定義
- サービス操作の定義
- Systinet へのサービスの作成と実行
- リクエストを処理し結果を返すフローの定義
- 以前定義済みのサービスを元にフローの Web サービストリガを定義
- Web サービスカスタマを使用したトリガのテスト

トリガは、以下の処理を行います。

1. Web サービスコンシューマからリクエストを受け取ります。
2. XSD タイプを Magic xpi データタイプにマッピングします。
3. リクエストの発行者がフロー実行の権限を持つかどうかをチェックします。プロジェクトプロパティ内の「認証システム」プロパティがなしに設定されている場合は実行されません。
4. リクエストの内容を含む XML ドキュメントを作成します。ドキュメントはメソッドに渡される引数の値に依存します（トリガ定義時に作成される XML スキーマによります）。
5. フローを起動します。

6. 戻り値として割り当てられたフロー変数の値を **Web** サービスコールに返します。II.

フローの作成

フローは以下のことを行います。

1. トリガによって作成され、引数の値を格納している **XML** ドキュメントを受け取ります。これはデータマップ経由で行われます。
2. トリガの戻り値として割り当てられた変数を更新します。

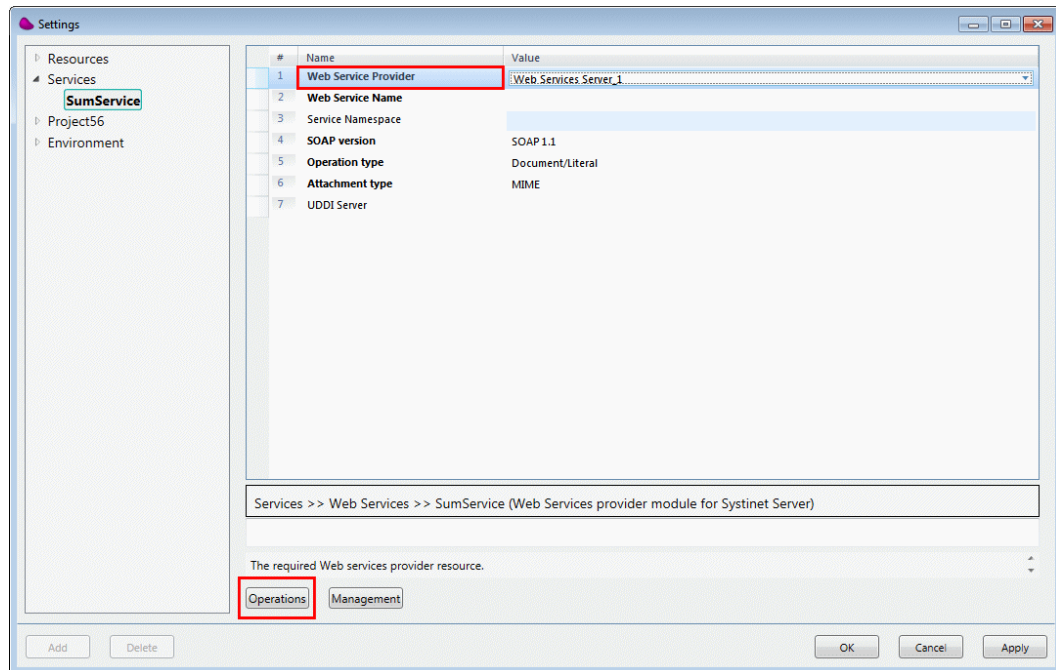
関連トピック：

[Web サービストリガを実装するには](#)

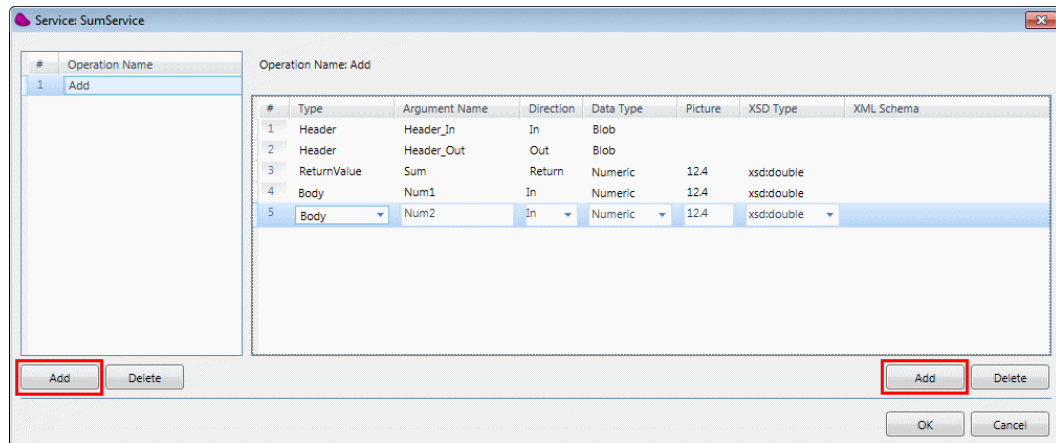
Web サービストリガを実装するには

このトピックでは、Web サービス・ロバイダである Magic xpi プロジェクトをビルドする方法について説明しています。このサービスは、Web サービスコンシューマ(クライアント)から二つの数値引数を受け取ります。Magic xpi がリクエストを受け取り、二つの数値の合計を計算し、それをコンシューマに返します。

1. [設定] ダイアログボックスの [リソース] セクションを開き、**Systinet** 接続詳細を含んだ新しい **Web** サービスサーバリソースを定義してください。デフォルトの接続値は以下の通りです。：
 - サーバーURL : **http://localhost:6060**
 - ユーザ名 : **admin**
 - パスワード : **changeit**
2. [設定] ダイアログボックスの [サービス] セクションを開き、**SumService** という名前で、新しい **Web** サービスサービスを定義してください。
3. [サービス設定] ダイアログボックスで、**Web** サービスプロバイダとして以前に作成された **Web** サービスサーバリソースを選択してください [Web サービス名] を入力し、パラメータの残りをデフォルト値のままにして、以下に表示されているように、[オペレーション] ボタンをクリックしてください。

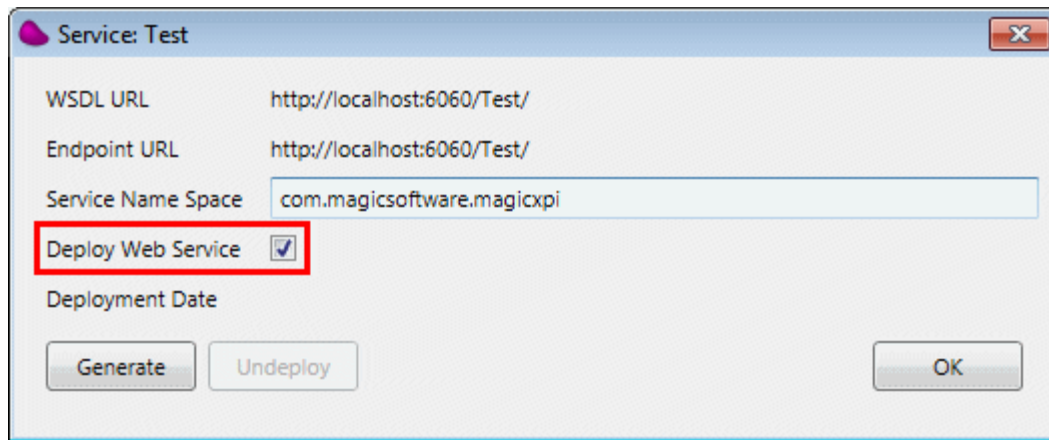


4. [オペレーション] ダイアログボックスで、**Add** オペレーションとその引数を定義してください。[Add] をクリックして新しいオペレーションを作成し、**Add** という名前を指定します。
5. [オペレーション] ダイアログボックスの[引数] セクションで、戻り値の名前を **Sum** に変更し、タイプを **Numeric** に変更してください。
6. さらに二つの引数を追加し、それらを **Num1** と **Num2** と名付けてください。以下のよう
に [方向] は **In**、[データタイプ] は **Numeric** に設定します。



[OK] をクリックして、[設定] ダイアログボックスに戻ります。

7. [設定] ダイアログボックスで [管理] ボタンをクリックし、[管理] ダイアログボックスを開きます。
8. [管理] ダイアログボックスで **Magic xpi** サービスと **Systinet** の **.jar** ファイルを生成することができます。[Web サービスのデプロイ] チェックボックスをチェックすると、サービスは **Systinet** サーバに配備され、機能するようになります。



9. [生成] ボタンをクリックし新しいサービスを実行します。

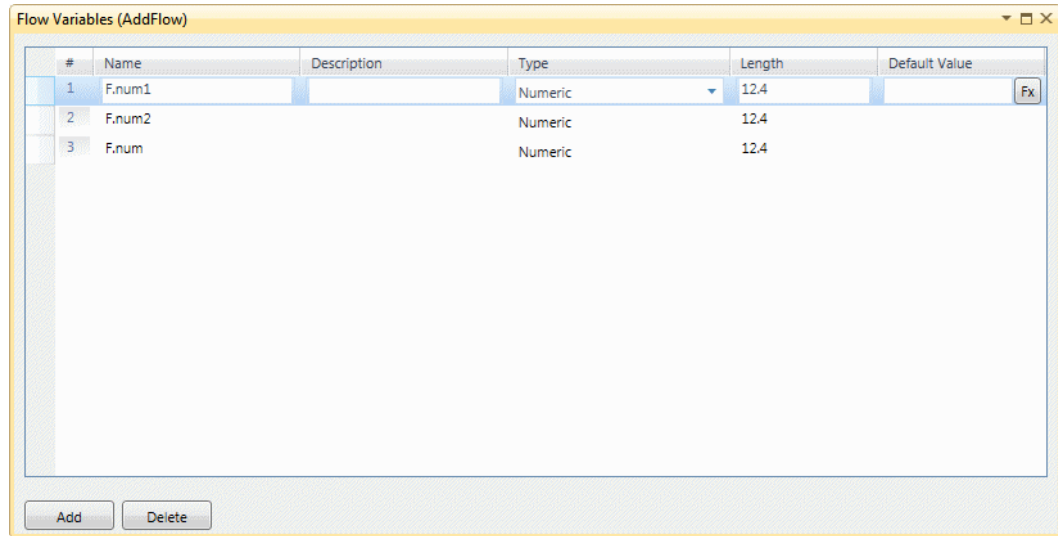
サービスが正しく配備されると、以下のメッセージが表示されます。また何かおかしい場合は、詳細を含んだエラーメッセージが表示されます。

エラーメッセージが表示されたり、**Not Authorized 404** メッセージが発生した場合、**Web サービスサーバ** リソースの接続詳細が正しいか確認してください。また、同じユーザ名とパスワードで Systinet コンソールにログインしてみてください。

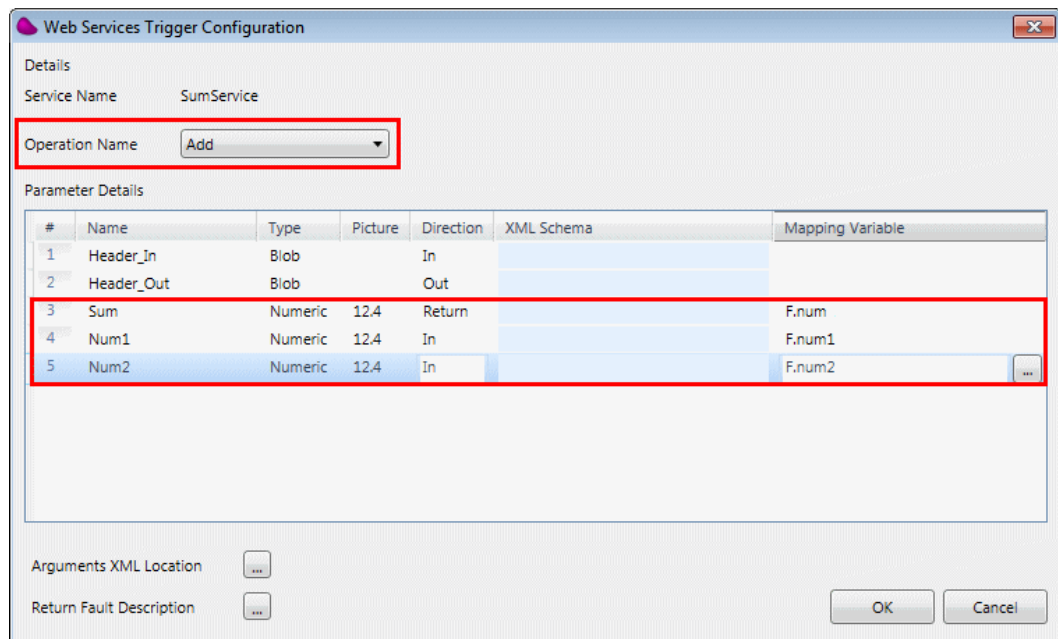
サービスの生成と配備に成功したら、その新しいサービスをフローで使用することができます。以下のステップを行ってください。

1. ソリューションエクスプローラで、フローを挿入したいビジネスプロセスを選択してください。
2. P [プロジェクト] メニューから、[Add]、そして [フロー] をクリックしてください。新しいフローを **AddFlow** と名付けてください。
3. ソリューションエクスプローラの新しいフローの配下で、[フロー変数] をダブルクリックしてフロー変数リポジトリを開いてください。

変数名	タイプ	説明
F.num1	Numeric	最初の値
F.num2	Numeric	二番目の値
F.num	Numeric	二つの値の合計

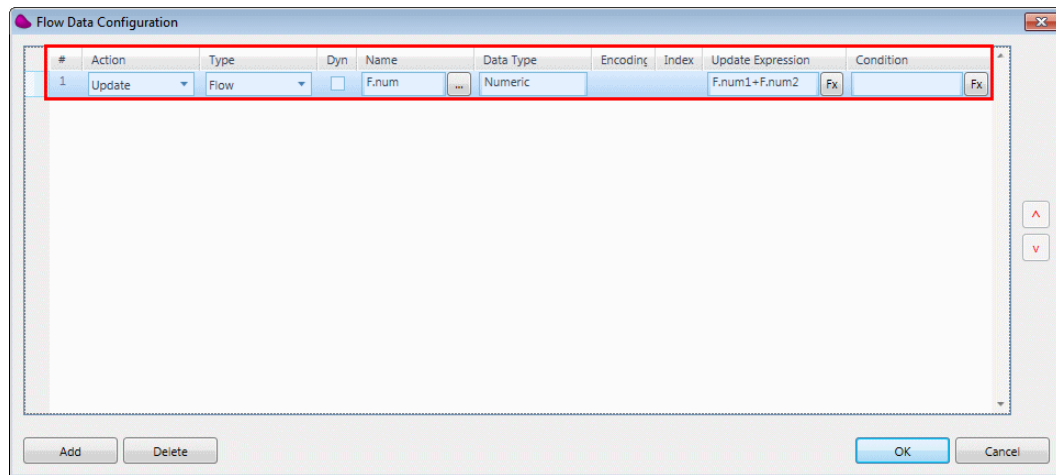


3. Web サービスユーティリティをフロートリガエリアにドラッグします。
4. Web サービスユーティリティの [プロパティ] ペインで、[サービス名] ドロップダウンリストから **SumService** を選択してください。
5. Web サービスユーティリティをダブルクリックするか、右クリックしてコンテキストメニューから [構成] を選択してください。[Web サービストリガー構成] ダイアログボックスが開きます。
6. [Web サービストリガー構成] ダイアログボックスで、[オペレーション名] ドロップダウンリストから **Add** を選択してください。
7. 戻り値の式で、**F.num** 変数を使用してください。
8. **Num1** と **Num2** のパラメータには以下のようにそれぞれ **F.num1** と **F.num2** 変数を使用します。

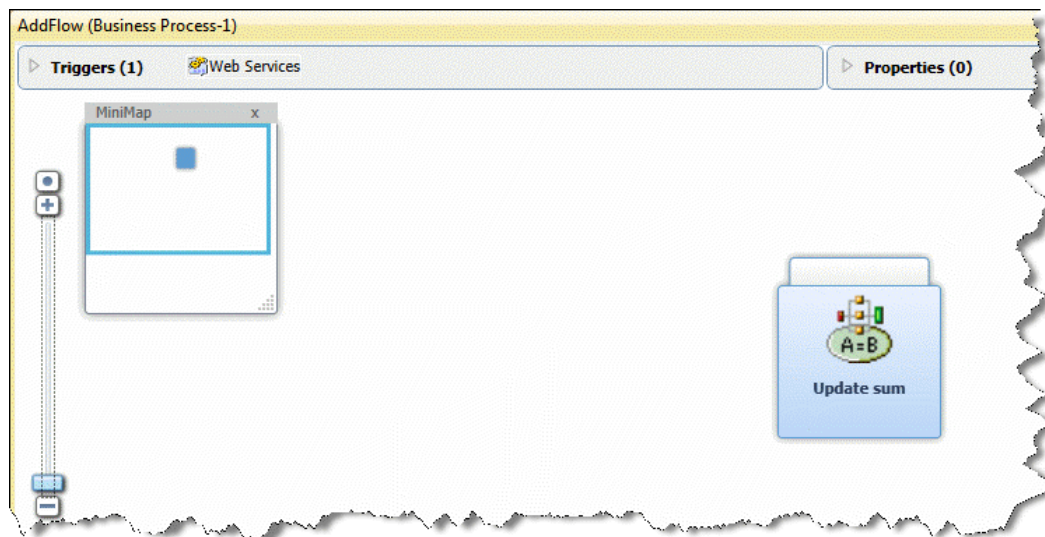


9. [OK] をクリックしてコンポーネントの構成を終了します。

10. 次にフローエリアにフローデータユーティリティをドラッグし、以下のように **f.sum** 式を **F.num1+F.num2** の合計値で更新します。



結果としてフローは以下のように表示されます。



11. [ビルド] メニューから [ソリューションをビルド] を選択し配備用フローを作成します。

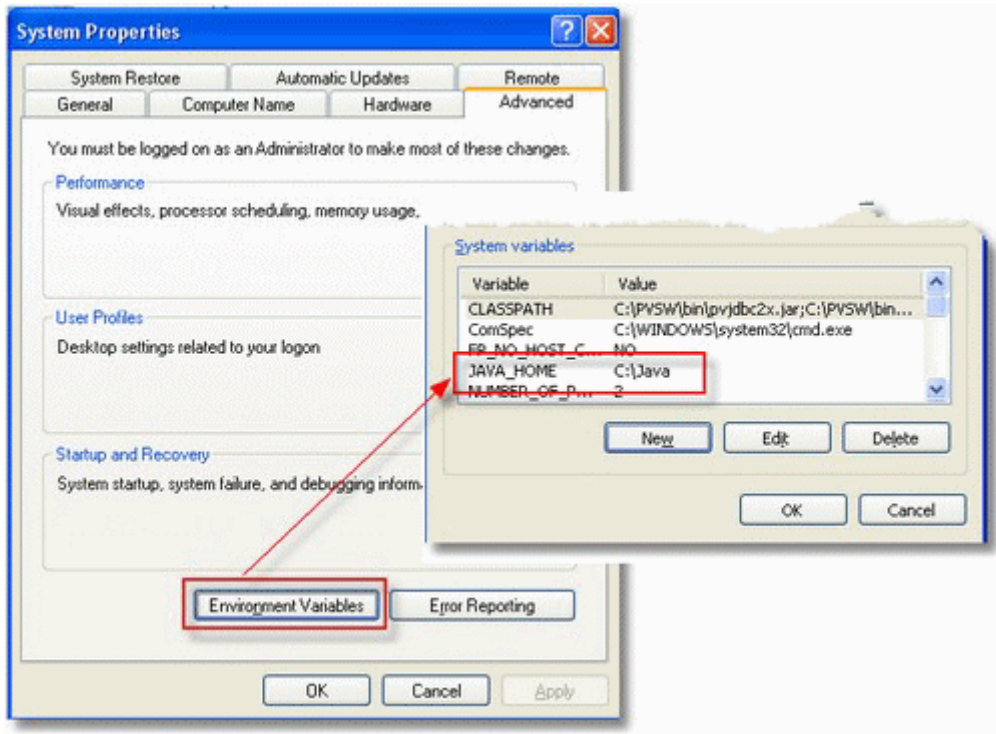
サービスのテスト

新しい Web サービスをテストするには、以下のステップを実行します。

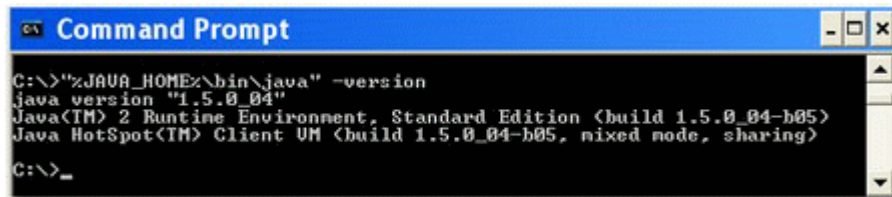
1. Java ランタイムエンジンがインストールされているか システム変数をチェックして確認します。
 - a. スタート > マイ コンピューター と選択し、右クリックして [プロパティ] を選択します。

Magic xpi 逆引き辞典

- b. [詳細設定] タブで [環境変数] を選択します。**JAVA_HOME** が一覧の中に表示されているはずです。

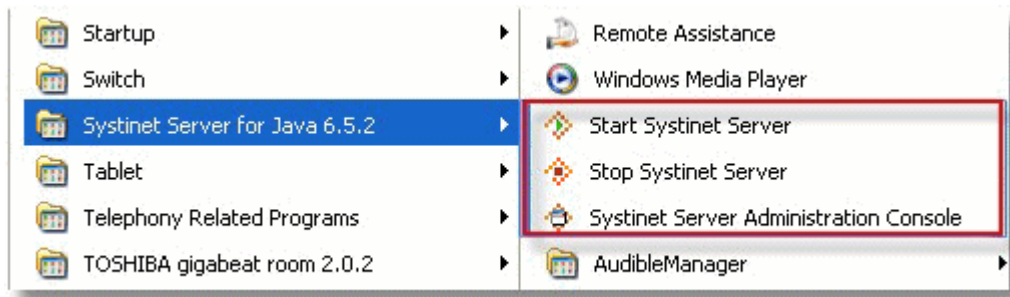


コマンドプロンプトで "**%JAVA_HOME%\bin\java**" -version コマンドを実行してみることも可能です。



どちらもうまくいかない場合、インストールに問題がある可能性があります。Magic xpi の再インストールで問題が解決するかもしれません。

- プロジェクトのスタートリンク、または Magic xpi モニタからサーバのプロジェクトを起動し、正しく起動するか確認します。

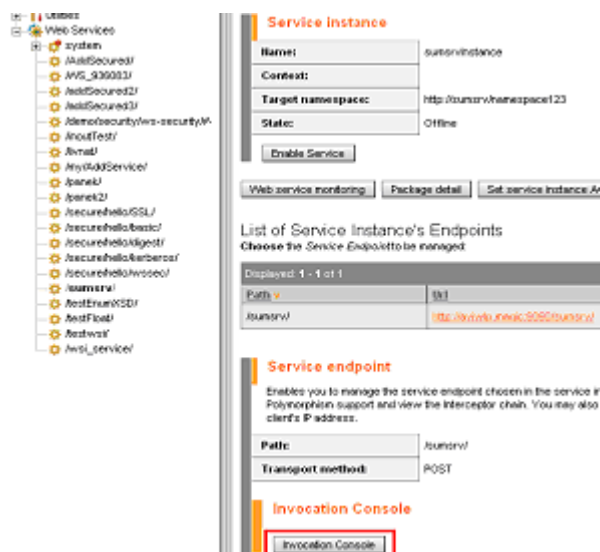


- インターネットブラウザ（Internet Explorer 又は Mozilla Firefox）を開き、Systinet コンソール URL (<http://localhost:6060/admin/console>)を開きます。以下の画面が見えるはずです。

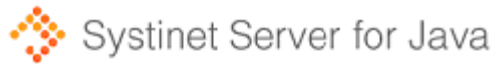


Systine の ャユーザ名とパスワードでログインします。

- Web Service Tree** であなたのサービスを探し出し、以下のように [Invocation Console] ボタンをクリックします。



- テストしたいオペレーションを選択し、テストデータを入力し、以下のように [Perform call] ボタンをクリックします。



Web Invocation Console

Service URL:

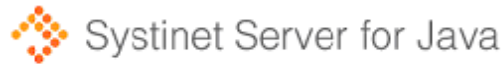
<http://sviwp.magic:8080/sumsrv/>

Operation:

Service	Port	Operation	Style	Encoding
SumsrvImpl	SumsrvImpl	Add(float, float)	doc	literal

Num1: float	<input checked="" type="checkbox"/>	<input type="text" value="5"/>
Num2: float	<input checked="" type="checkbox"/>	<input type="text" value="6"/>

6. コールが成功すると以下のように表示されます。



Web Invocation Console

Input message

Following message was sent to the service at <http://sviwp.magic:8080/sumsrv/>

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:e="http://schemas.xmlsoap.org/soap/envelope/">
  <e:Body>
    <ns0:Num1 xmlns:ns0="http://systinet.com/xsd/SchemaTypes/">5</ns0:Num1>
    <ns0:Num2 xmlns:ns0="http://systinet.com/xsd/SchemaTypes/">6</ns0:Num2>
  </e:Body>
</e:Envelope>
```

Output message

Following message has been received

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:d="http://www.w3.org/2001/XMLSchema" xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns:vm0="http://systinet.com/xsd/SchemaTypes/"
  xmlns:vm1="http://ideoz.com/interface">
  <e:Body>
    <vm0:float_Response i:type="d:float">11.0</vm0:float_Response>
  </e:Body>
</e:Envelope>
```

注意: (XML Spy のような) 外部ツールや Web サービススタジオやコードからサービスをテストすることもできます。

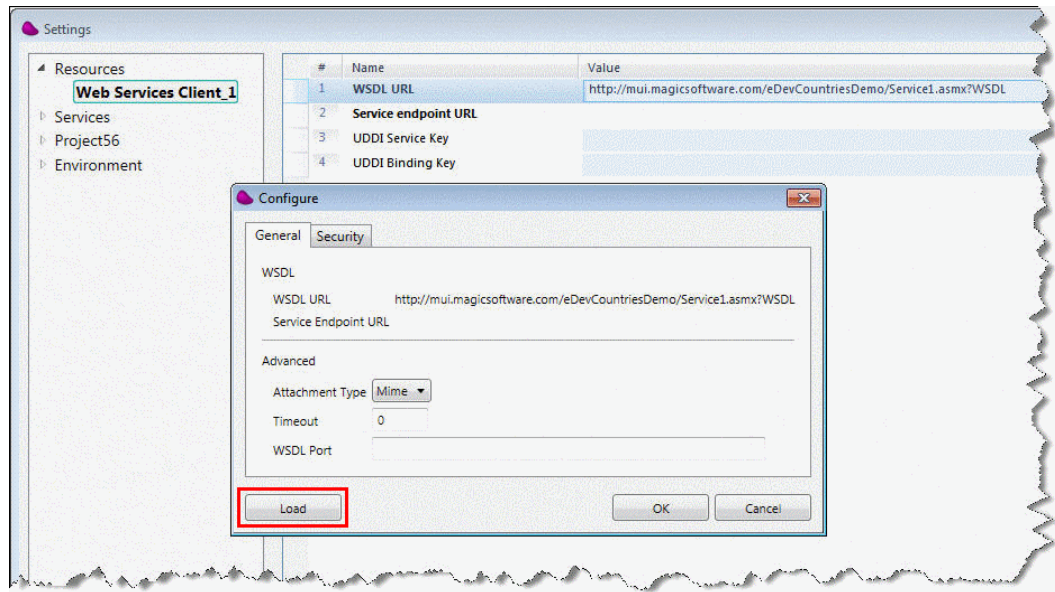
Web サービスのコンシューマとして Magic xpi を設定するには

Magic xpi Web サービスコンシューマは、インターネット上の外部 Web サービスプロバイダにアクセスすることができます。

以下のステップに従い、Countries Demo Web サービスにサービスの WSDL URL を使用してアクセスします。

1. 新しいプロジェクトまたは新しいフローを作成します。

2. 新しい Web サービスクライアントリソースを定義します。
3. 以下の値を [WSDL URL] フィールドに入力します：
http://mui.magicsoftware.com/eDevCountriesDemo/Service1.asmx?WSDL
4. [設定] をクリックして WSDL をロードします。

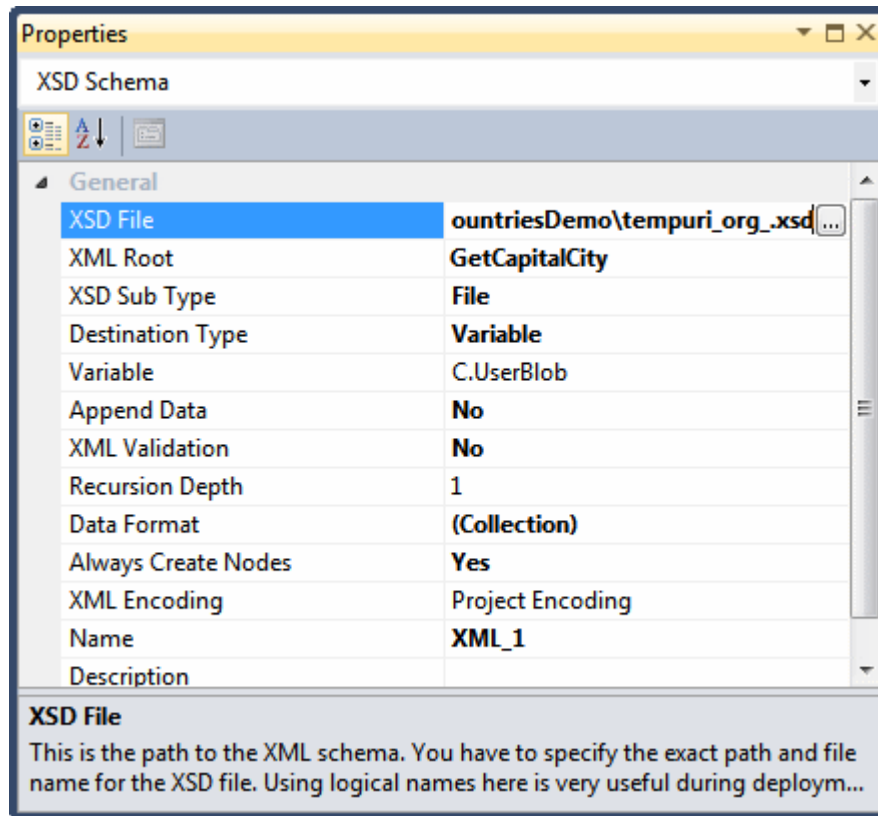


5. 以下のフロー変数を定義します。

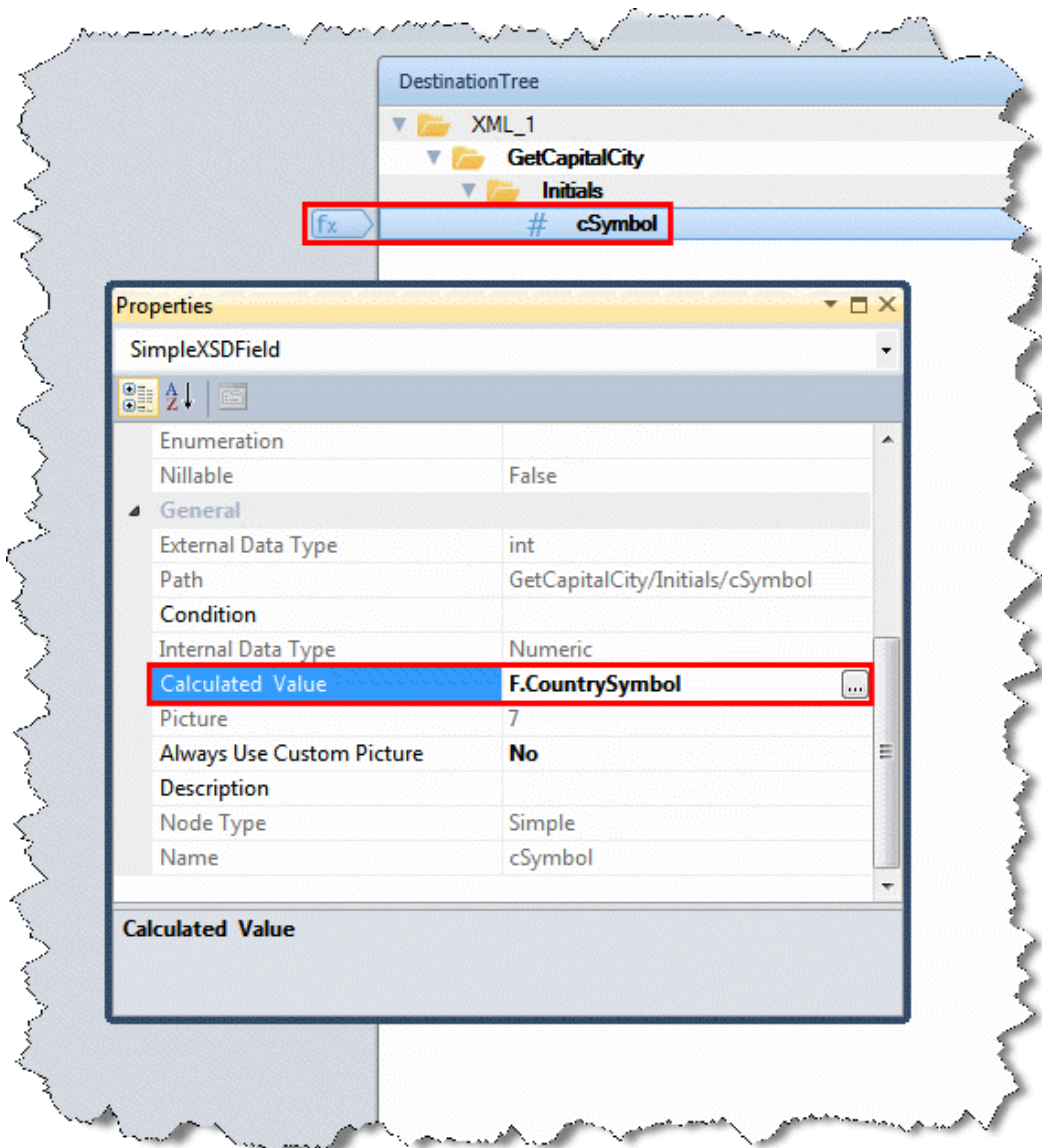
名前	タイプ	書式	デフォルト値
F.Result	BLOB		
F.OUT	BLOB		
F.Fault	BLOB		
F.CountrySymbol	Alpha	2	US

6. データマップユーティリティをフローエリアにドラッグします。
7. Web サービスユーティリティをフローエリアのデータマップステップにドラッグしてします。
8. Web サービスユーティリティをダブルクリックするか、右クリックしてコンテキストメニューから [構成] を選択してください。[Web サービス構成] ダイアログボックスが開きます。
9. **GetCapitalCity** という名前のオペレーションを選択します。
10. [GetCapitalCity] パラメタで **F.OUT** を、[GetCapitalCityResponse] パラメタで **F.Result**、[Fault] パラメタで **F.Fault** (画面左下) を選択します。[OK] をクリックします。
11. データマップステップをダブルクリックするか、右クリックしてコンテキストメニューから [構成] を選択してください。[データマップ] ウィンドウが開きます。
12. ツールボックスの [マップスキーマ] セクションから、XML タイプを [データマップ] ウィンドウの送り先エリアにドラッグしてください。

13. XML 送り元の [プロパティ] ペインで、以下の XSD ファイルエントリを選択してください : **tempuri_org_.xsd**。このファイルは、 **<Magic xpi>\Projects\<プロジェクト名>\resources\<サービス名>** にあります。
14. **F.OUT** フロー変数をデータ送り先として選択します。
15. **GetCapitalCity** を XML ルートとして選択します。



16. [保存] をクリックしてください。
17. [データマップ送り先] ペインで、ツリーを全て展開し **cSymbol** 要素を公開してください。
18. **cSymbol** 要素をダブルクリックし、計算値として **F.CountrySymbol** を選択してください。



19. [保存] をクリックしてください。
20. メッセージ保存ユーティリティをフローにドラッグしてください。
21. [メッセージ] パラメータに **WS call result** を入力してください。
22. [BLOB] パラメータでは、**F.Result** 変数を選択してください。
23. [ファイル拡張子] パラメータに、**xml** を入力してください。コンシューマフローは、**自動起動**に設定してください。
24. プロジェクトを保存してください。プロジェクトは以下のように表示されるはずです。



25. デバッガまたはサーバを使用してフローを実行します。

注
意：

- Web サービスの起動に失敗すると **Fault (F.Fault)** にマッピングされている変数に障害の応答が入力されます。
- Web サービス呼出しからの XML 結果は、データマップを使用して、リクエストで使用されたものと同じ XSD で解析する必要があります。しかし、**GetCapitalCityResponse** ルートを使用してください。
- 最初のデータマップステップの前に **F.CountrySymbol** に適切な値を入れる必要があります（デフォルトは **US** です）。

統合 Windows 認証によって保護されている Web サービスを呼び出すには

このトピックは、統合 Windows 認証 (IWA) を必要とする Web サービスのコンシューマを有効にするための環境設定を行う方法について説明しています。

IWA の概要は、以下を参照してください：

http://en.wikipedia.org/wiki/Integrated_Windows_Authentication

セットアップ手順

1. 以下を含む **jaasconfig.conf** という名前のテキストファイルを作成してください。

```
KrbCredentials {
```

```
com.sun.security.auth.module.Krb5LoginModule required debug=true
doNotPrompt=false
useTicketCache=true;
};
```

2. **Magic.ini** ファイルの**[MAGIC_SPECIALS]**セクションに、以下の 3 つのエントリを追加します。

- a. OverrideSoapSpyForIntegratedAuthentication=Y
- b. Java1.6Home = <Magic xpi installation>\Runtime\JRE
- c. IntegratedAuthenticationJvmArgs= -
Djava.security.auth.login.config=c:\temp\jaasconfig.conf -Dsun.security.jgss.debug=true -
Dsun.security.krb5.debug=true

設定ファイルのパス (太字) は、PC 上のファイルの位置と合っていなければなりません。

コメント

- このセットアップの後で Web サービスを呼び出すと、現在の Windows ユーザーの Windows ログイン証明書が各呼び出し (Kerberos でエンコードされています) に追加されます。
- 呼び出し側とプロバイダは、同じアクティブディレクトリを使用しなければなりません。

デバッグ

呼び出しを行う Java プロキシサーバで、トレースを可能にするには、以下のように特殊フラグを修正してください:

```
[MAGIC_SPECIALS]
IntegratedAuthenticationJvmArgs=-Djava.security.auth.login.config=C:\temp\jaasconfig.conf -
Dsun.security.jgss.debug=true -Dsun.security.krb5.debug=true -
Dcom.magicsoftware.ssj.integratedauth.debug=true
```

これで、WS の呼び出しで、**%TEMP%**フォルダの **WsInteAuthXXXXXX.log** という名前のログファイルが作成されます。

LAN 上のプロキシサーバを使用する

プロキシサーバを使用した通信を可能にするためには、特殊フラグのうちの 1 つを以下のように変更する必要があります:

```
[MAGIC_SPECIALS]
IntegratedAuthenticationJvmArgs=-Dhttp.proxyHost=Proxy server IP -Dhttp.proxyPort=Proxy
Server port -Djava.security.auth.login.config=jaasconfig.conf -Dsun.security.jgss.debug=true -
Dsun.security.krb5.debug=true
```

Magic xpi の中の Web サービスに基本認証を提供するには

Magic xpi で、Web サービスに基本認証を提供するには、以下を行う必要があります。

Systinet の Admin コンソール:

Magic xpi 逆引き辞典

1. Web service を開き、[Set endpoint security] をクリックしてください。
2. [Authorization Required] チェックボックスを選択してください。
3. [Custom Security Providers] ボタンをクリックしてください。
4. [HttpBasic] チェックボックスを選択してください。

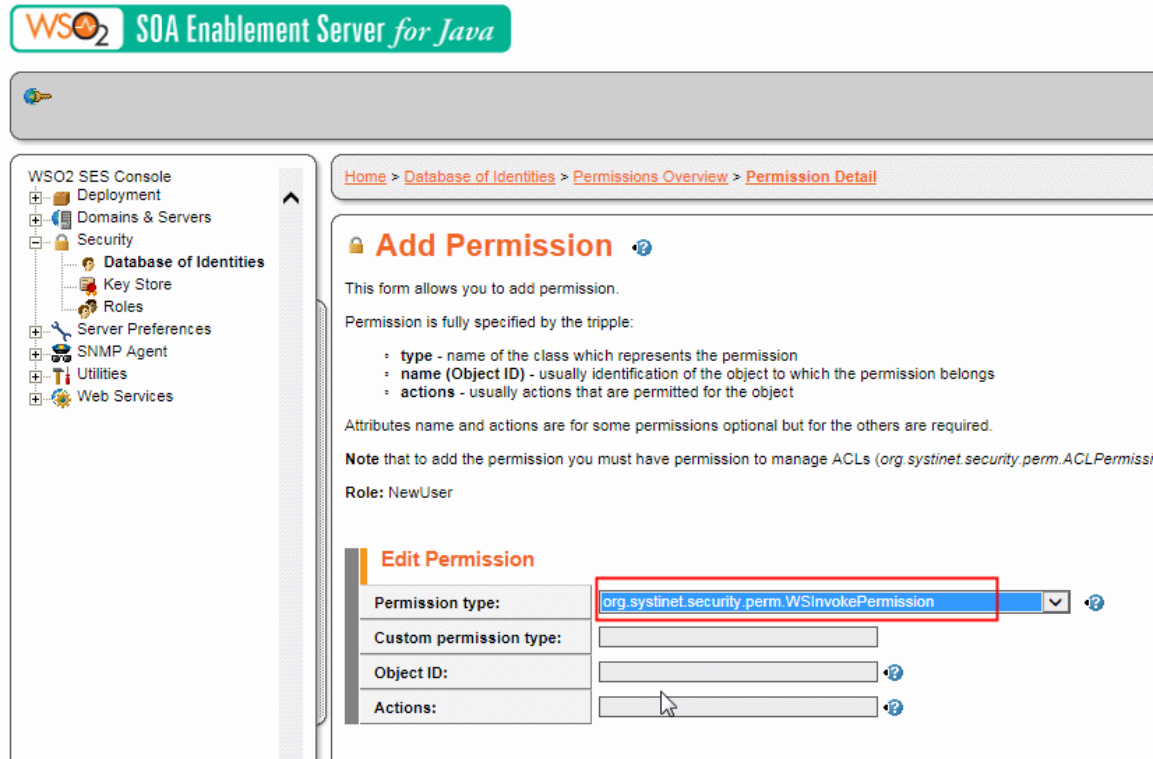
Magic xpi :

次の通り Web サービスクライアントリソースを設定する必要があります。

1. Web サービスクライアントリソースで、WSDL URL を入力し、[構成] ボタンをクリックしてください。
2. [汎用] タブで、[検証] ボタンをクリックしてください。
3. [セキュリティ] タブで、[セキュリティレベル] を **Transport**、[認証タイプ] を **BASIC** に変更してください。
4. [OK] をクリックしてください。
5. Web サービスステップを追加し、[Web サービス構成] で、[Identity] ボタンをクリックしてください。
6. ユーザ名とパスワード(**admin** と **changeit** など)を入力してください。

注意: Systinet にユーザと役割を追加することができます。password という 名前のプロパティを Systinet に追加するために必要なパスワードを定義して、Web サービスを呼び出す許可を与える場合は、以下のようしてください。

1. Systinet の **Permissions overview** で、[Add permission] ボタンをクリックしてください。
2. [Permission type] フィールドで、**org.systinet.security.perm.WSInvokePermission** を選択してください。



WS-Security で動作している Web サービスプロバイダとして、Magic xpi を設定するには

WS-Security は、Web サービスのリクエストと応答の両方でデジタル署名される高水準のセキュリティプロトコルです。

これは、他の「ハンドシシェイク」のセキュリティプロトコル (SSL) とは異なり、クライアントとサーバで相互で信用するには十分でないことを意味します。各リクエストと応答は、証明書キーを使用して署名されなければなりません。

パート 1—エイリアスと証明書の設定

上記の説明に基づき、最初にプロバイダとコンシューマの間で証明書を渡す必要があります。

応答にデジタル署名するために使用することができる証明書を読み込む必要があります。このようにするには、以下の手順に従ってください。

1. Systinet Pstore で新しいエイリアスを作成します (この例では : **WsSecurityAlias**)。
2. 使用する必要がある証明書を読み込むか、証明書のリクエストを作成します。次に、リクエストを証明書 “generator” に送信します。
3. 一旦、証明書を受け取ると、作成した新しいエイリアスを使用して読み込ませます。

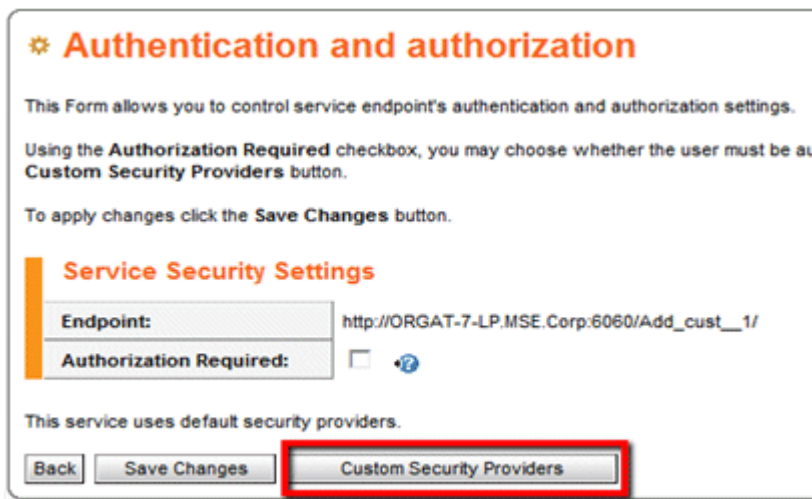
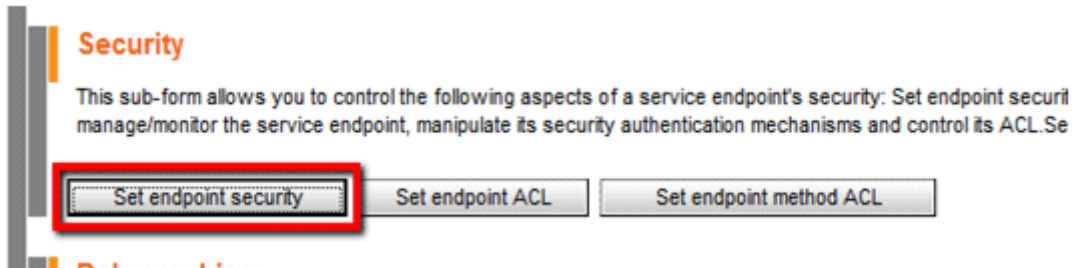
新しい証明書を読む方法については、[ここ](#)を参照してください。

証明書を受け取り、Pstore のエイリアス (**WsSecurityAlias**) に読み込むと、WS-Security プロトコルを使用するために Web サービスを設定する必要があります。

パート 2 - (Systinet コンソールから)WS-Security プロトコルを設定する

WS-Security プロトコルを設定するために、以下の手順に従ってください。

1. Magic xpi で Web サービスのサービスを作成して、それを配備してください。
2. Systinet コンソールから、作成された Web サービスを見つけ、以下の設定を行います：
 - a. **[Set endpoint security]** をクリックして、次に **[Custom Security Providers]** をクリックします。



- b. **[WS-Security]** チェックボックスを選択し **[Properties]** をクリックします。

Accepting Security Providers:

Record count: 6		
Name v	Accepting	
HttpBasic	<input type="checkbox"/>	N/A
HttpDigest	<input type="checkbox"/>	Properties
Kerberos	<input type="checkbox"/>	Properties
Steminder	<input type="checkbox"/>	N/A
SSL	<input type="checkbox"/>	N/A
WS-Security	<input checked="" type="checkbox"/>	Properties

- c. **[Custom Settings]** をクリックします。

Security Provider Properties

This page can be use for setup security provider properties.

Service Endpoint: /Add_cust__1/
Security Provider Name: WS-Security

Custom Settings Back

3. 新しいセクションが"WS-Security configuration of message"というページに追加されます。ここから、新しいトークンと証明書を以下のように設定します：

- a. [Create Configuration] をクリックします。

Secured outgoing messages

#	WS-Security configuration of message
1	WS-Security configuration of message Create Configuration

Add message configuration

- b. [Add Token] をクリックして、[Add Signature] に移ります。.

Security tokens

#	order	properties
Add Token		

signatures

#	canonicalization	KeyInfo	order	references
Add Signature				
timestampId:		<default>		
version:		<default>		
WS-Security namespace:		<default>		
WS-Utility namespace:		<default>		

Add message configuration

estampHeaderExpiration: <default>

4. セキュリティトークンのために、証明書 "generator" によって必要される公開鍵基盤 (PKI)を設定する必要があります。たとえば、X.509 v3 デジタル証明書の構造については、以下の設定を使用してください。

- a. **Security tokens** のパラメータに以下を入力してください。

i.order: 100

ii.type: **X509v3**

iii.wsu:Id: **SigningSecurityToken-1**

iv.methods: Web サービスのサービスメソッドの各メソッド名

#	order	properties	type	wsu:Id						
1	100	<table border="1"><thead><tr><th>#</th><th>name</th><th>value</th></tr></thead><tbody><tr><td></td><td></td><td></td></tr></tbody></table>	#	name	value				X509v3	SigningSecurityToken-1
#	name	value								

#	name
1	opt1

b. signatures でのパラメータ

- i. token Id: **SigningSecurityToken-1**
- ii. mode: **reference**
- iii. order: **0**
- iv. method: **http://www.w3.org/2000/09/xmldsig#rsa-sha1**
- v. body: チェック
- vi. wsu:id: **Signature-1**

注意 :

- このセクションで記述される設定は、**wasp.xml** ファイルの **%WASP_HOME%\work\<your WS name>.jar\WASP-INF** の配下で保存されます。
- Web サービスを再作成すると、デフォルト設定で作成されます。これは、上記の設定が元に戻ることを意味します。

パート 3 - エイリアス (パート 1) を **WS-Security** 設定の **WS エンドポイント** (パート 2) に接続する

これまで、新しいエイリアスと新しい証明書を設定して、Web サービスのサービスを利用するために、WS-Security プロトコルと構造を設定しました。

次に、エイリアスと WS-Security 設定を結びつけることによってこれらの 2 つの要素を接続する必要があります。これを行うには、次の手順に従ってください:

1. **"Providers manager"** と呼ばれているユーティリティを開きます。これは、**SSJ\bin** フォルダの下にインストールされています。Windows の [スタート] ボタンから、[ファイル名を指定して実行] を起動し、次のコマンドを入力します。
%WASP_HOME%\ProvidersManager.bat --gui.
2. **Providers manager** ユーティリティを使用することで、エンドポイントで Pstore から証明書エイリアスを結びつけることができます。この証明書は、Web サービスの応答を署名するために使用されます。

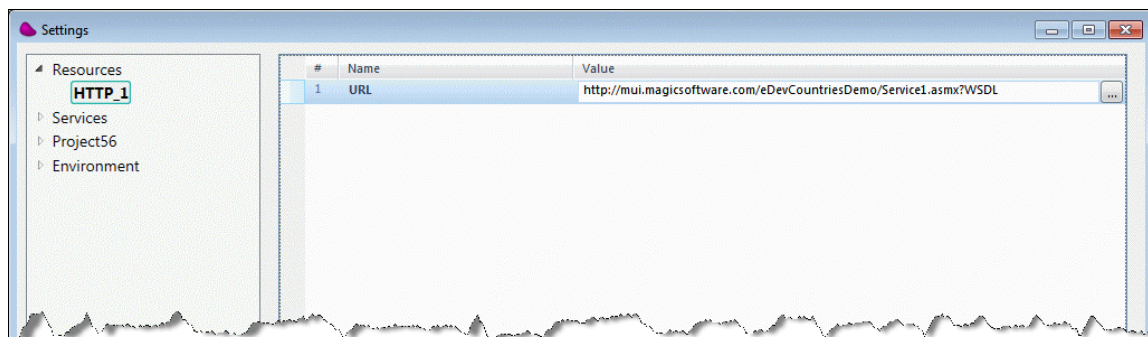
3. [File] メニューから、[Open Admin Service] を選択します。
4. [Endpoints] タブを選択します。
5. 左側の [Service] ウィンドウから利用する Web サービスを見つけてください。
6. [WS-Security] チェックボックスを選択します。
7. ユーザー名とパスワードを以下のように設定してください。
 - a. ユーザー名は、Pstore からサーバ証明書のエイリアスの名前でなければなりません。
 - b. パスワードは、その証明書に格納されたパスワードでなければなりません。

HTTP コンポーネントを使用して Web サービス呼び出しを行うには

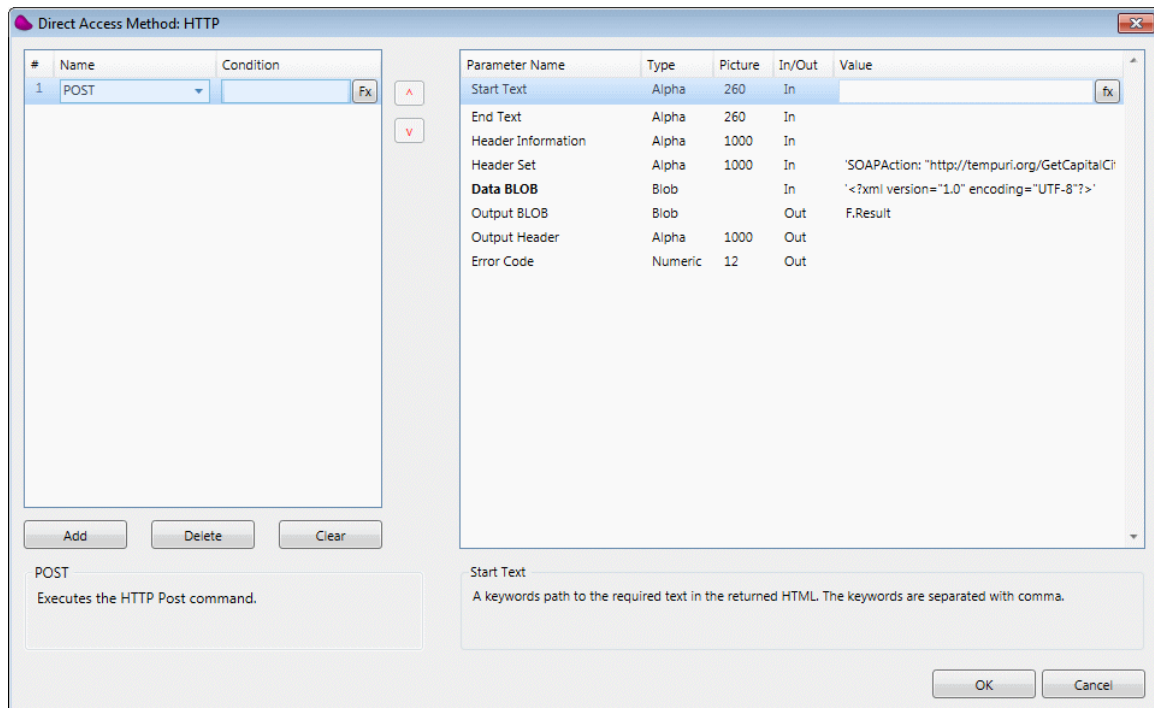
時々、Web サービス・コンポーネントの代わりに HTTP コンポーネントを使用して Web サービス呼び出しを送る要求や必要があるかもしれません。

このトピックでは、以下の通りに HTTP ステップ ([Web サービスのコンシューマとして Magic xpi を設定するには](#) の例を参照)を設定する方法について説明しています。

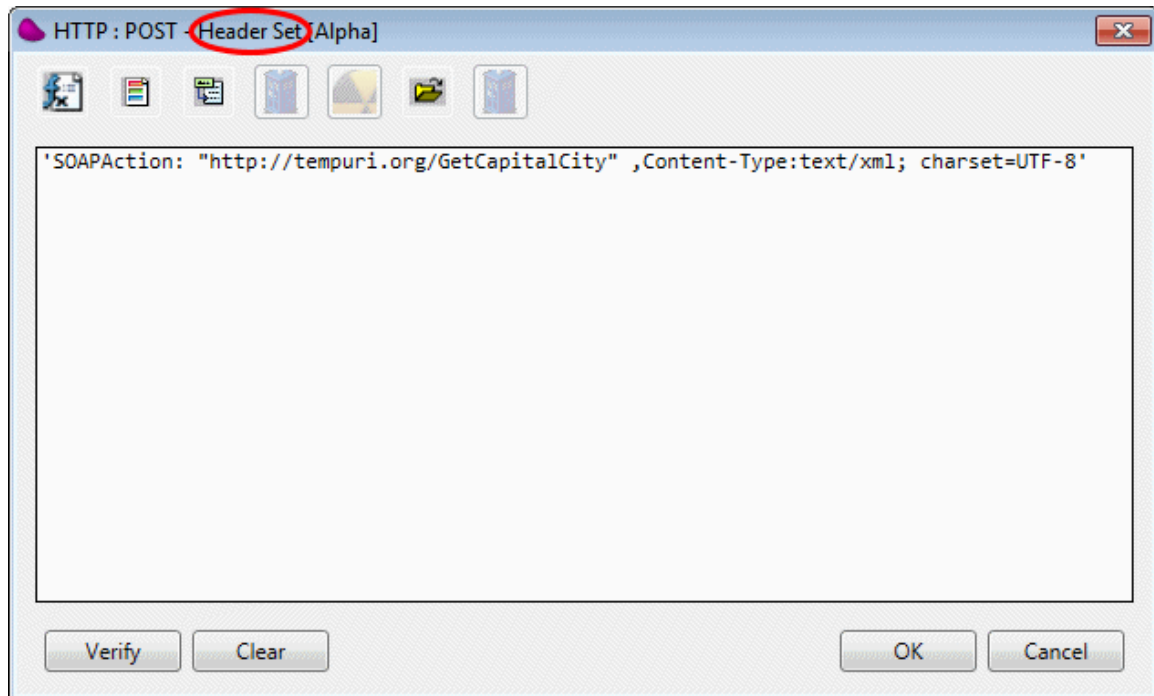
1. HTTP リソースを作成してください。

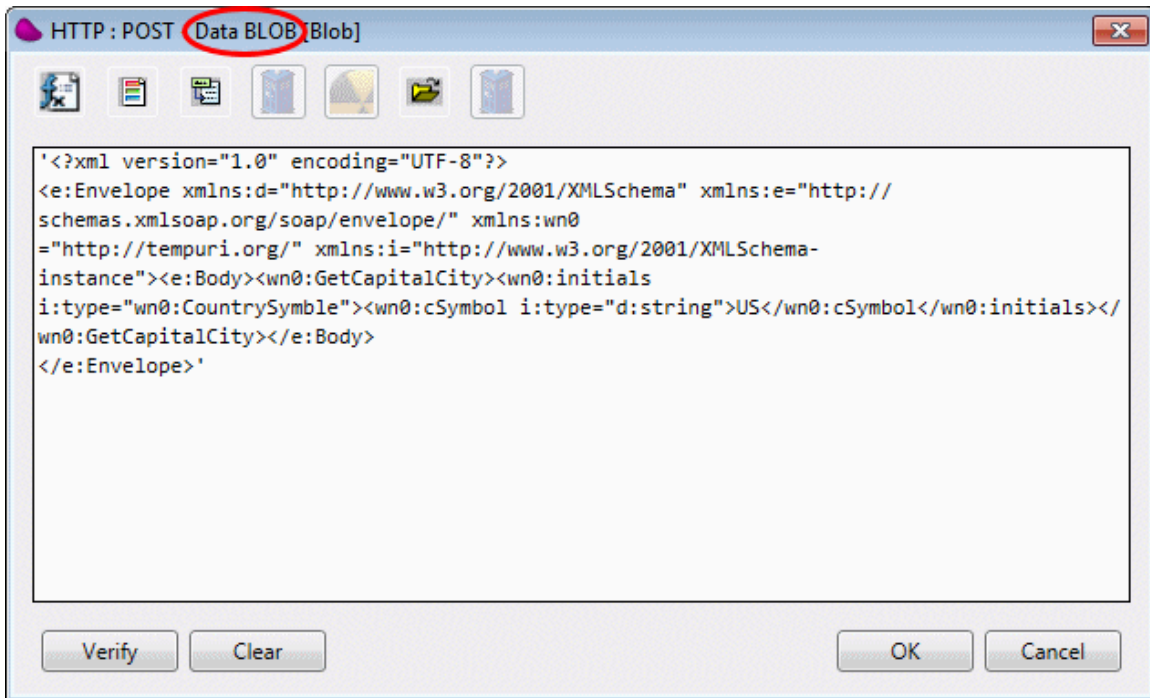


2. HTTP コンポーネントをフローエリアにドラッグ&ドロップして、**Post** メソッドを設定してください。



3. HTTP リクエストのボディの内容を参照するには、XMLSpy や SoapUI またはサードパーティのツールを使用して、リクエストをトレースしてください。そして、そこから HTTP ヘッダを取得してください。
4. 送信されたパラメータがこの例で何であるのか（リクエスト・ヘッダとデータ BLOB は、以下の画面を参照してください）について知るために、全部のヘッダとボディは、SoapSpy からそのままコピーされます。





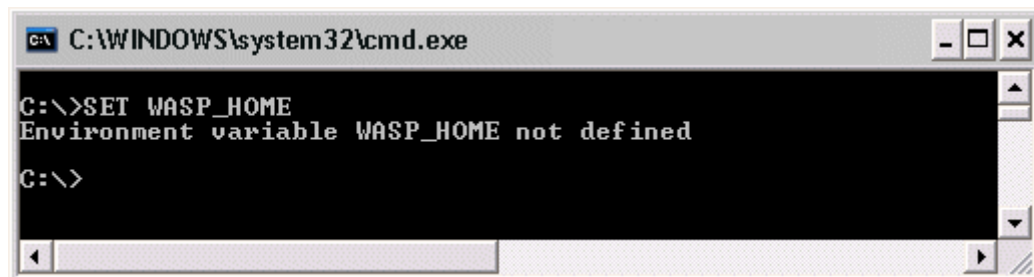
注 WSDL が Magic xpi で読み込むことができない場合、SoapSpy を使用することはできません。呼び出しをトレースするために SoapUI と SOAP ツールキットを使用することができます。

手動で Systinet Server for Java をインストールするには

Magic xpi のインストーラが設定上の問題で Systinet をインストールすることが場合があります。このトピックでは、Systinet をインストールする上での必要条件と検証方法とインストール方法について説明します。

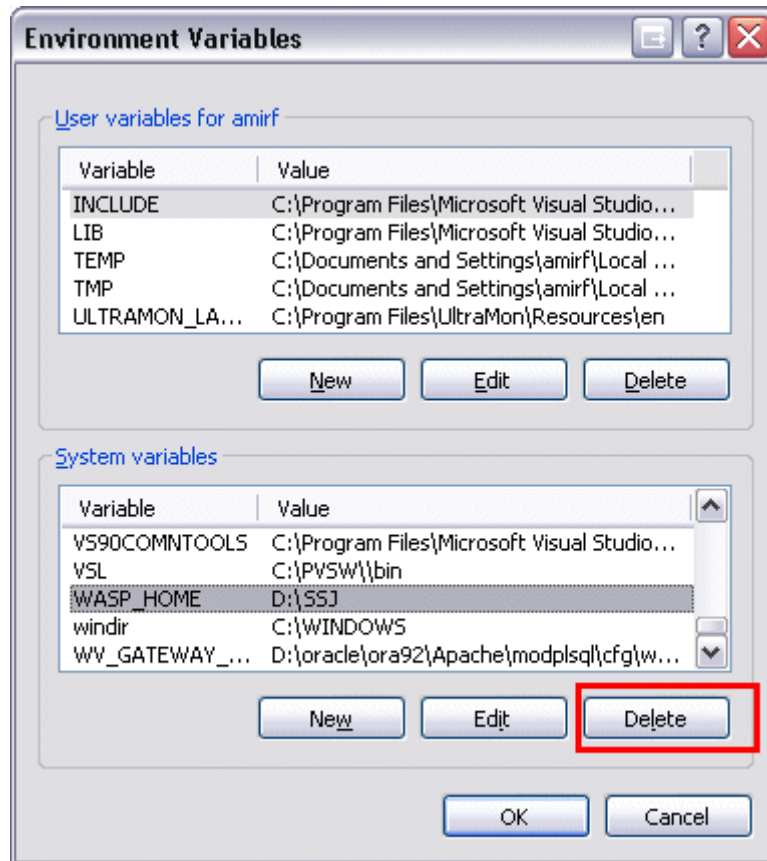
必要条件

1. 環境変数として WASP_HOME が定義されないことを確認してください。
SET WASP_HOME コマンドを使用することで確認できます。

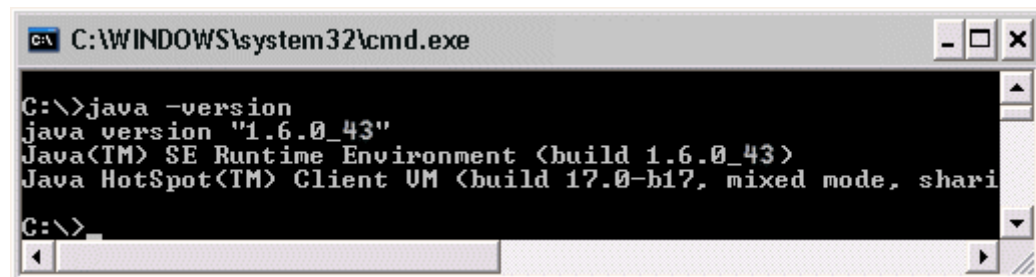


2. 環境変数が存在する場合は、以下のようにして環境変数ダイアログを開き、WASP_HOME を選択したら、[削除]をクリックしてください。

コンピュータ アイコンで右クリック > プロパティ > システム詳細設定 > 詳細設定タブ > 環境変数



3. PC で Java が正しくインストールされていることを確認してください。Systinet サーバーは Java を利用しており、Ver1.6 以上を必要とします。これは、以下に示すように、コマンドプロンプトで `Java -version` を実行することで確認できます。



`Java -version` コマンドは、PATH 環境変数で定義されたパスで位置づけられた `java.exe` ファイルのインストール・バージョンを表示します。

たとえば、`Java.exe` ファイルが、PATH 環境変数 `….; F:\magic¥ Java¥bin;….` によって利用できるように設定されます。

これは、**Java.exe** ファイルが以下のファイル・パスを指定していることを意味します。**F:**
%magic% Java%bin%;.....。
 以下のどちらかがインストール可能です。

- **JRE** Magic xpi の実行エンジンに必要な **Java** 実行時環境です。
- **JDK**クライアント・モジュールまたはプロバイダ・モジュールを作成するうえで必要な **Java** の開発キットです。JDK がインストールされている場合は、**JRE** も含まれます。JDK は、開発環境では必須になります。

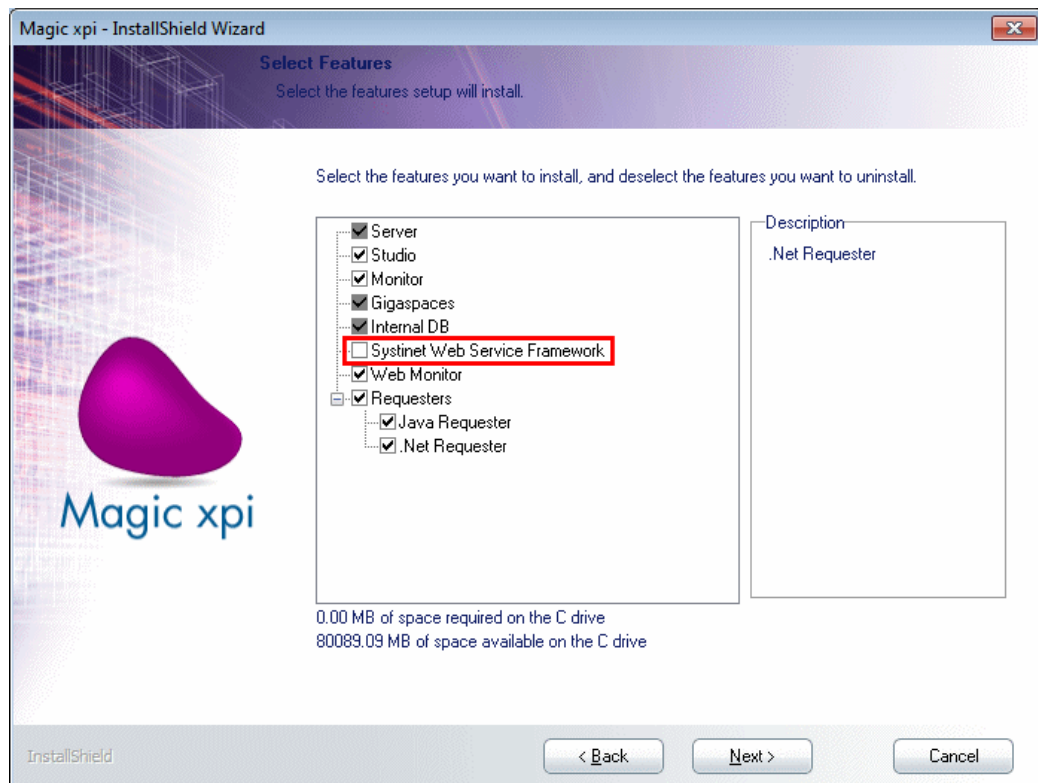
JDK/JRE のインストーラーは、以下からダウンロードすることができます：
<http://java.sun.com/javase/downloads/index.jsp>

Java のインストーラは、Magic xpi のインストール CD にもバンドルされています。

4. 環境変数 **JAVA_HOME** が **Java -version** コマンドで確認したバージョンと合っていることを確認してください。
5. **Java -version** コマンドによって特定された **java.exe** ファイルの位置を特定するには、**where java** コマンドを使用します。最初の結果は、PC の **PATH** 環境変数に **java.exe** ファイルの位置が返ります。

インストール方法

1. 以下に示すように、**Web サービス・フレームワーク** を選択しないで、Magic xpi をインストールしてください。このコンポーネントはデフォルトで選択されるので、チェックボックスをクリアにする必要があります。



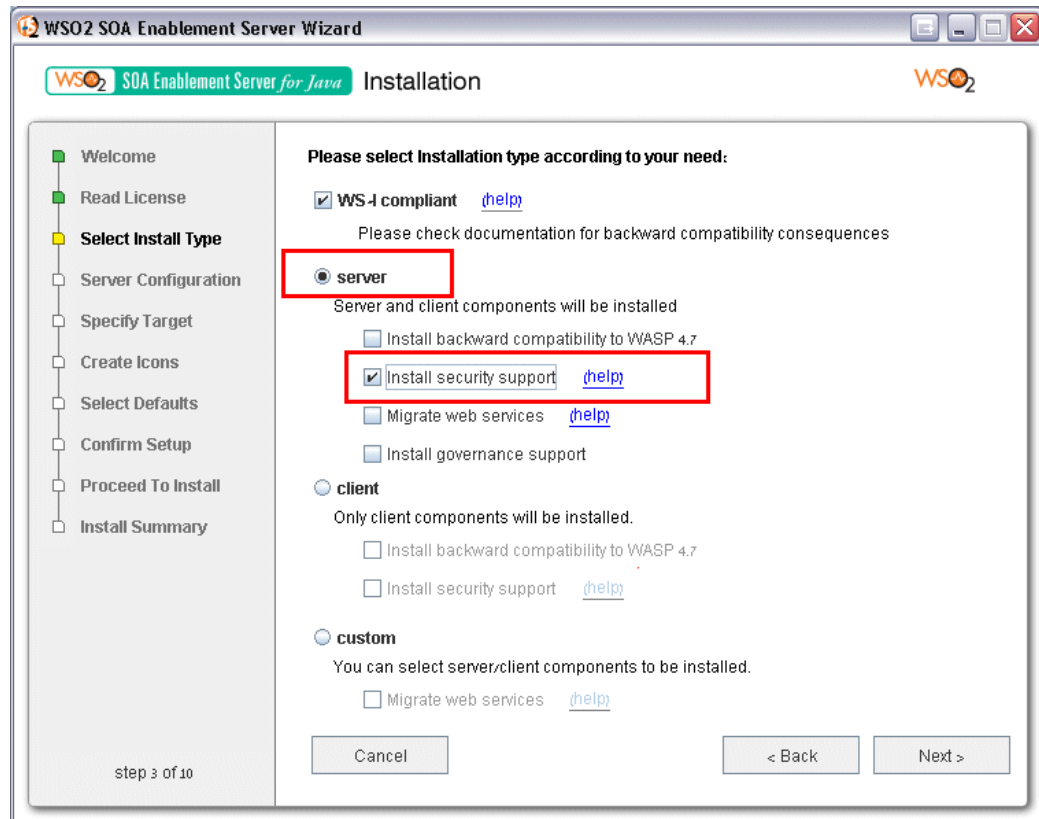
2. 以下のコマンドを入力することによって、手動で Systinet をインストールしてください。**java -jar wso2-soa-enablement-server-java-6.6.jar**.

wso2-soa-enablement-server-java-6.6.jar ファイルは、Magic xpi のインストール CD 内にあります。

このファイルは、Systinet 6.6 のインストーラです。
旧バージョンのファイル名は、**systinet-server-java-6.5.2.jar** です。

Systinet 6.6 は、Java Ver1.6 以上を使用します。

3. S 以下に示すように、**Install Security Support** オプションを選択してください。

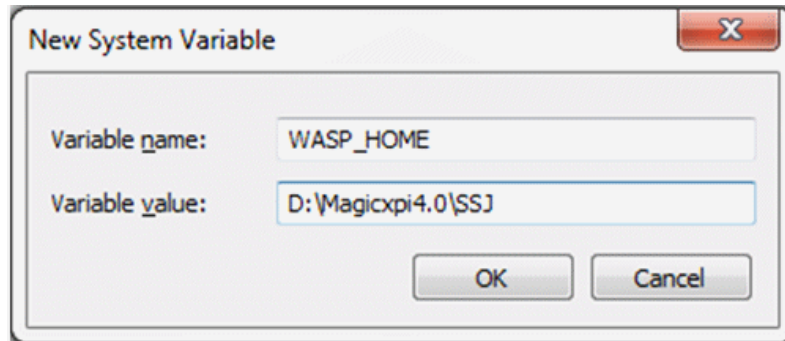


インストールのステップ#3の後、**Next** をクリックしてデフォルト指定を使用してください。

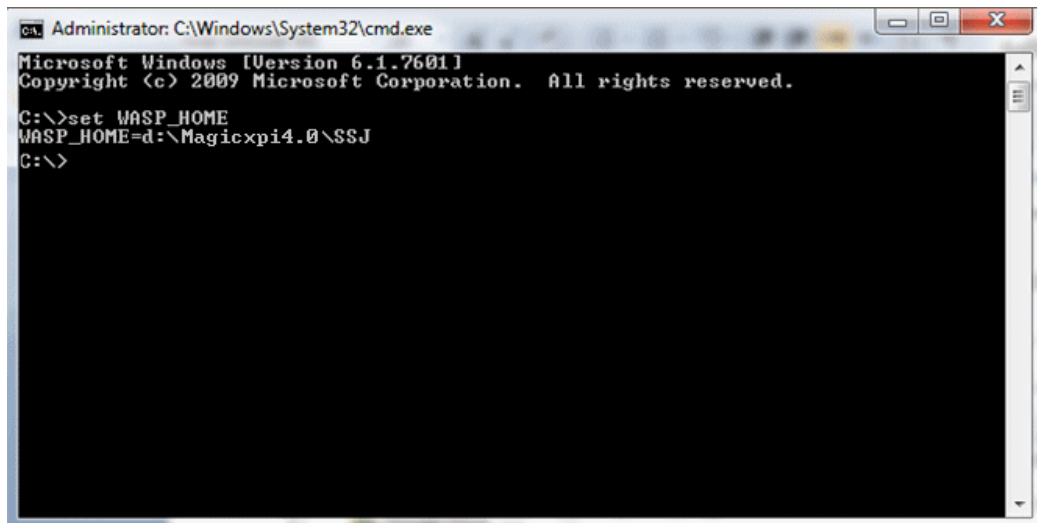
4. 以下のようにして環境変数ダイアログを開き、**WASP_HOME** に Systinet のインストールパスを定義してください。

コンピュータ アイコンで右クリック > プロパティ > システム詳細設定 > 詳細設定タブ
> 環境変数

ユーザ環境変数でなく、システム環境変数に定義してください。



以下のイメージは、予想される結果を表示しています（PC を再起動する必要があるかもしれません）。



5. 以下の **JAR** ファイルを **%WASP_HOME%\lib** ディレクトリ内にコピーしてください。

- <Magic xpi のインストールフォルダ>\Runtime\Magic Xpa\Support\MgSSJ.jar
- <Magic xpi のインストールフォルダ>\Runtime\Magic Xpa\Support\MgRequester.jar
- <Magic xpi のインストールフォルダ>\Runtime\Magic xpa\Support\saaj_utils.jar
- <Magic xpi のインストールフォルダ>\Runtime\java\lib\uniRequester.jar
- <Magic xpi のインストールフォルダ>\Runtime\java\lib\mgxpi-commons.jar

6. log4j.config ファイル（SSJ\config）に以下の行を追加してください。

log4j.category.magicxpi=INFO,magicFile

```
log4j.additivity.magicxpi=false
# file appender
log4j.appender.magicFile=org.apache.log4j.RollingFileAppender
log4j.appender.magicFile.File=${wasp.location}/log/magic.log
log4j.appender.magicFile.MaxFileSize=1MB
log4j.appender.magicFile.MaxBackupIndex=5
log4j.appender.magicFile.layout=org.apache.log4j.PatternLayout
log4j.appender.magicFile.layout.ConversionPattern=%p %d [%c] %m%n
```

7. log4j.jar ファイル (%WASP_HOME%\lib) を以下のようにして Magic xpi の同一ファイルで置き換えてください。:

- a. <Magic xpi のインストールフォルダ>\java\lib\log4j-1.2.16.jar を %WASP_HOME%\lib ディレクトリにコピーしてください。
- b. %WASP_HOME%\lib から元の log4j.jar ファイルを削除してください。
- c. %WASP_HOME%\lib にある log4j.jar を、log4j-1.2.16.jar にリネームしてください。

8. %WASP_HOME%\lib で、GigaSpaces という名前の新しいフォルダを作成してください。そして、ここに以下のファイルをコピーしてください。:

- <Magic xpi のインストールフォルダ>\java\lib\mgxpi-gs.jar
- <Magic xpi のインストールフォルダ>\GigaSpaces\lib\required (すべてのファイルをこのフォルダからコピーします)

9. %WASP_HOME%\bin\Server.bat ファイルを編集してください。これは、Web サービスを提供する上で必要です。以下の 2 つの変更が必要です。:

8.

- a. **WASP_CLASSPATH** を以下のように設定します。

```
set
WASP_CLASSPATH=%WASP_CLASSPATH%;%WASP_HOME%\lib\uniSSJ.jar;%WASP_
HOME%\lib\uniRequester.jar;%WASP_HOME%\lib\saa_j_utils.jar;%WASP_HOME%\lib\mg
xpi-commons.jar;%WASP_HOME%\lib\GigaSpaces\com.springsource.org.aopalliance-
1.0.0.jar;%WASP_HOME%\lib\GigaSpaces\commons-
logging.jar;%WASP_HOME%\lib\GigaSpaces\dsl.jar;%WASP_HOME%\lib\GigaSpaces\gs-
s-openspaces.jar;%WASP_HOME%\lib\GigaSpaces\gs-
runtime.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.aop-
3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.asm-
3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.aspects-
3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.beans-
3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.context.support-
3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.context-
3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.core-
3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.expression-
3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.transaction-
3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\mgxpi-gs.jar;
```

以下のように、**CALL** から始まり、**SET PARAMS** から始まる行のの後に配置します。

```
CALL <SSJ installation>\bin\env.bat
Set WASP_CLASSPATH=.....
SET PARAMS=%*
```

- b. **%JAVA_CMD%** セクションでは、次を追加してください。 "-

```
Dcom.magicsoftware.xpi.SpaceName=MAGIC_SPACE " "-
Dcom.magicsoftware.xpi.LookUpGroupName=<LookupGroupName> " "-
Dcom.magicsoftware.xpi.LookUpLocators = <host:port>,<host:port>"
```

9. Systinet サーバのサービスを作成してください。

このセクションは、必須ではありません。サーバは、以下を入力することで呼び出すことができます。

```
%wasp_home%\bin\StartService.bat
```

%WASP_HOME%\bin\InstallService.bat を以下のように修正してください。

- a. **CALL env.bat** の行を以下の行に差し替えてください。

```
set JAVA_HOME= The JAVA_HOME value
set WASP_HOME= The WASP_HOME value
set JAVA_CMD= The JAVA_HOME value%bin%java
set BUILD_FILE=build-default.xml
```

- b. ステップ#9 で定義した場合と同 CLASSPATH を指すように、**Djava.class.path** を変更してください。

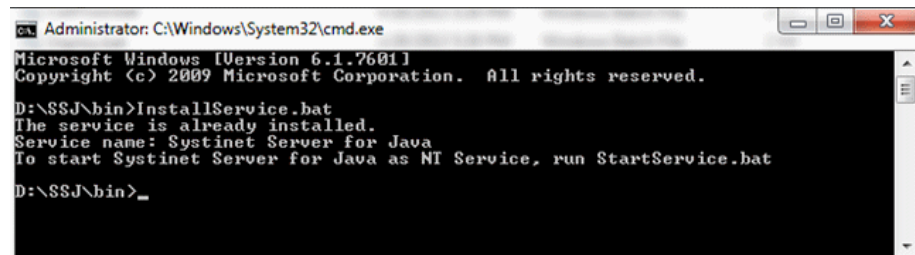
"_

```
Djava.class.path=%WASP_HOME%\lib\wasp.jar;%WASP_HOME%\lib\uniSSJ.jar;%WASP_HOME%\lib\uniRequester.jar;%WASP_HOME%\lib\saa_j_utils.jar;%WASP_HOME%\lib\mgxpi-commons.jar;%WASP_HOME%\lib\GigaSpaces\com.springsource.org.aopalliance-1.0.0.jar;%WASP_HOME%\lib\GigaSpaces\commons-logging.jar;%WASP_HOME%\lib\GigaSpaces\dsl.jar;%WASP_HOME%\lib\GigaSpaces\gs-openspaces.jar;%WASP_HOME%\lib\GigaSpaces\gs-runtime.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.aop-3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.asm-3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.aspects-3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.beans-3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.context.support-3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.context-3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.core-3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.expression-3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\org.springframework.transaction-3.2.4.RELEASE.jar;%WASP_HOME%\lib\GigaSpaces\mgxpi-gs.jar;%CLASSPATH%;
```

- c. 以下の行 (正しい Space 関連の環境で)を追加してください。

```
Dcom.magicsoftware.xpi.SpaceName=MAGIC_SPACE " "-
Dcom.magicsoftware.xpi.LookUpGroupName=<LookupGroupName>" "-
Dcom.magicsoftware.xpi.LookUpLocators = <host:port>,<host:port>" "-
直後に
"-Dlog4j.configuration=%WASP_HOME%\conf\log4j-NTservice.config"
```

- d. スクリプトを保存して、実行してください (**set NT_SERVICE_NAME=...**の行を変更することでサービス名を変えることができます)。



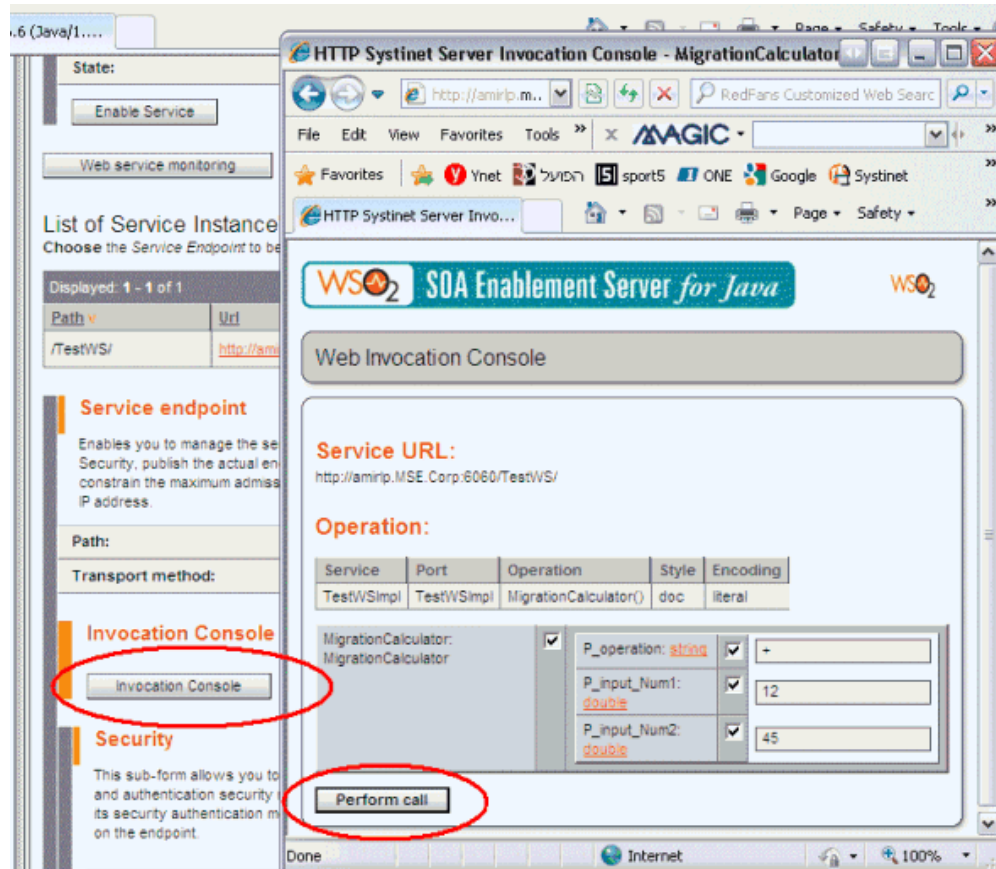
```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

D:\SSJ\bin>InstallService.bat
The service is already installed.
Service name: Systinet Server for Java
To start Systinet Server for Java as NT Service, run StartService.bat
D:\SSJ\bin>_
```

インストールのテスト

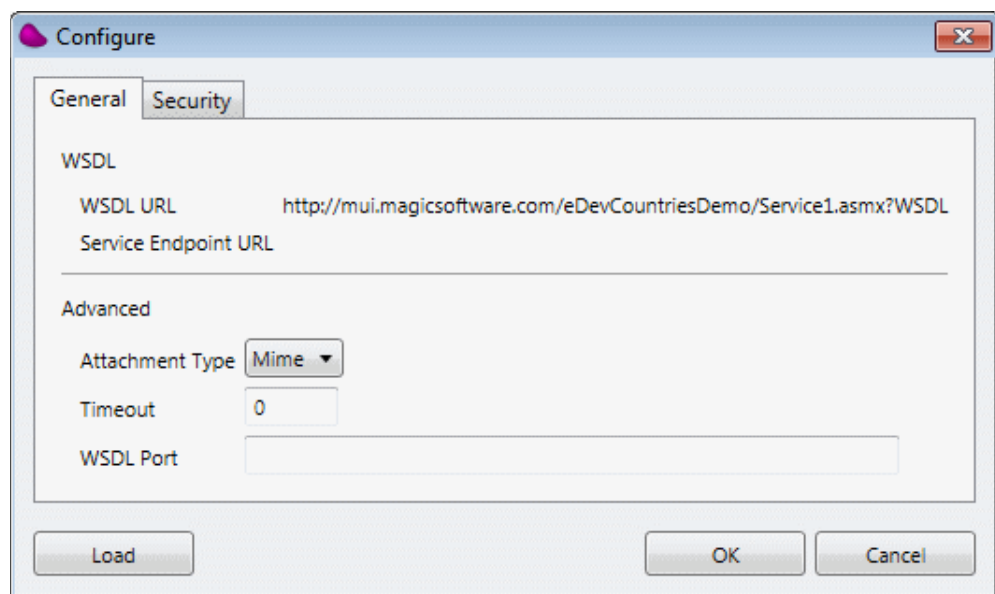
1. プロバイダ側のテスト

- Magic xpi で Web サービスを作成し配備します。
- 以下に示すように、Systinet Server の **Web Invocation Console** から、Web サービスを呼び出してください。



2. コンシューマ側のテスト

- 「設定」ダイアログボックスの「サービス」セクションの中にサービスをロードしてください。
- Web サービスを呼び出すプログラムを実行してください。



Magic xpi で Web サービス Over SSL のコンシューマとして使用するには

このトピックでは、SSL 経由の Web サービスをプロバイダやコンシューマとして使用するために Magic xpi と Systinet を設定する方法について説明しています。さらに、SSL サーバの証明書の変更が記述される追加された SSL 関連の設定方法も含んでいます。

セキュアな方法で動作する時だけ、いくつかの Web サービスは利用できます。最も一般的なプロトコルのうちの 1 つが、SSL (Secure Socket Layer) です。例えば、次のように URL に 's' が付加されていることで識別できます。https://www.example_url.com

このトピックは、そのような Web サービスをプロバイダやコンシューマとするための、Magic xpi と Systinet Server での設定内容について説明しています。

Magic xpi は、いくつかの設定で SSL で動作することをサポートします。

- サーバは信頼されたサーバでなければならないが、クライアント認証を必要としない場合
- 証明書を使用して、サーバがクライアントを信頼し、その引き替えにクライアントがサーバを信頼するような相互認証を使用する場合。

このトピックは、Systinet のインストールによって提供されるデフォルト証明書を他の証明書に変更するために必要な方法を説明しています。

Web サービス over SSL のコンシューマにする：

Web サービス over SSL のコンシューマの場合、Web サービス・ベンダによって提供される証明書を使用するために Systinet Server を設定する必要があります。

場合によっては、正しい証明書が使用されていることを確実にするために、Magic xpi と関連するリソースで特定の環境が必要となります。

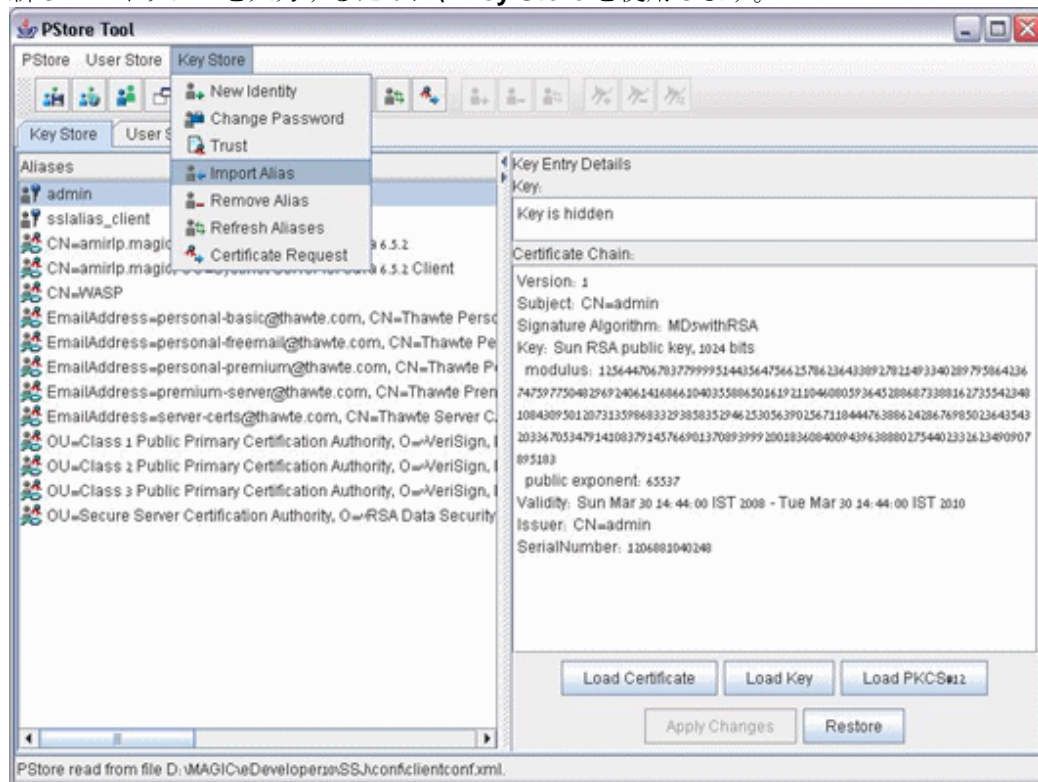
相互認証のためにクライアント証明書を設定する：

クライアント証明書が提供された場合、クライアント証明書を Systinet に読み込ませるために以下のステップが必要です。

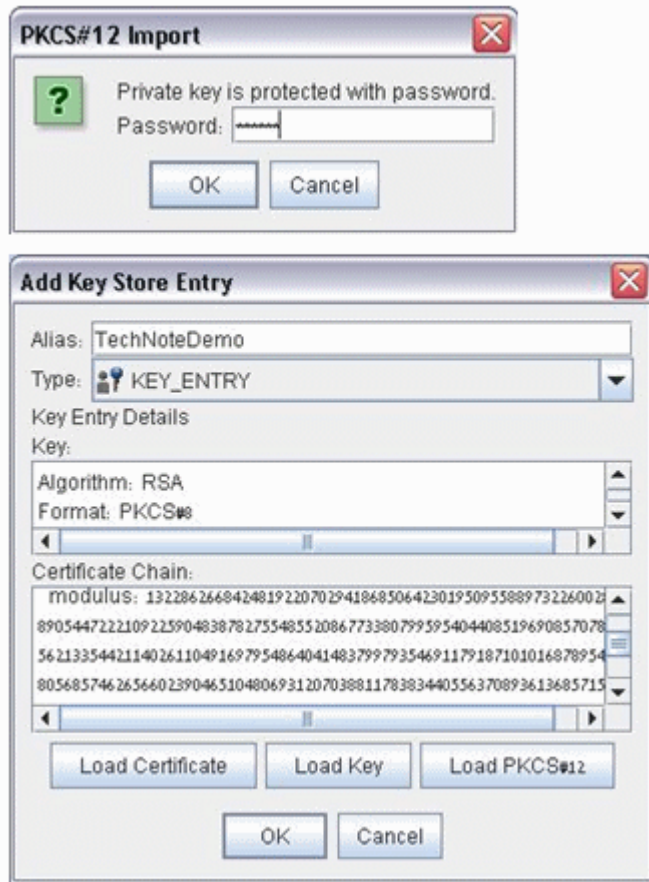
注	WSDL を読み込むときは、クライアント証明書が使用されないため、WSDL のローカルコピーを保存して、そこから動作することが推奨されます。
----------	--

1. コマンドプロンプトで以下のコマンドを実行して、PStore ツールを開いてください。**%WASP_HOME%\bin\PStoreTool.bat --gui**
2. PStore メニューを使用して、**%WASP_HOME%\conf\clientconf.xml** を選択することで、クライアント設定ファイルを開きます。

3. 新しいエイリアスを入力するために、**Key Store** を使用します。



4. 新しいエイリアスに名前 (Magic xpi で使用するため)を設定し、**Load PKCS#12** をクリックします。クライアント証明書 (.pfx または.cer ファイル)とそのパスワードを指定してください。



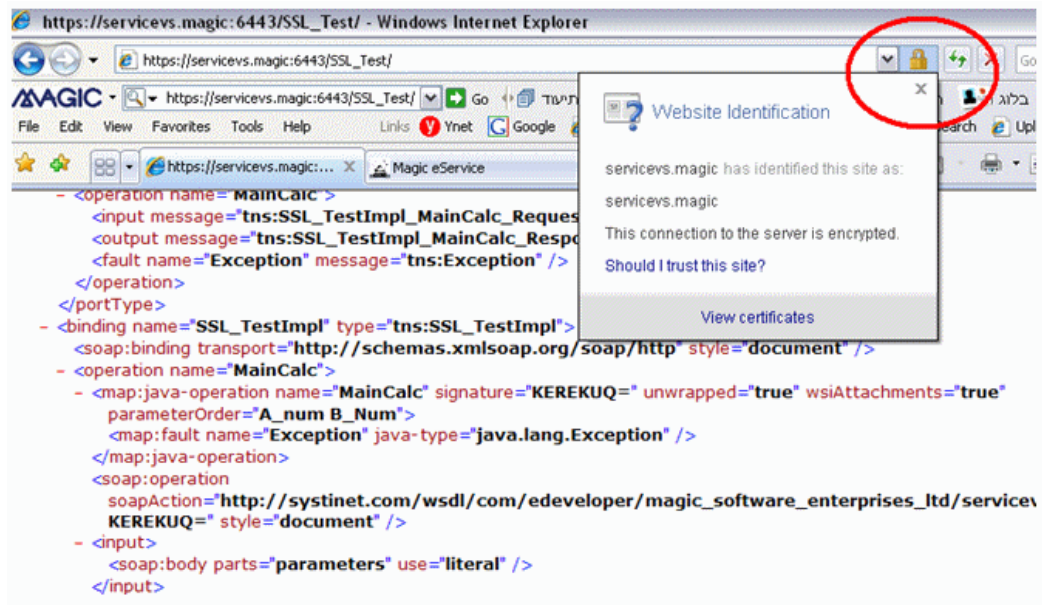
5. このエイリアス名とパスワードは、Web サービスのステップで [Show Identity details] 画面で使します。

サーバ証明書を設定する：

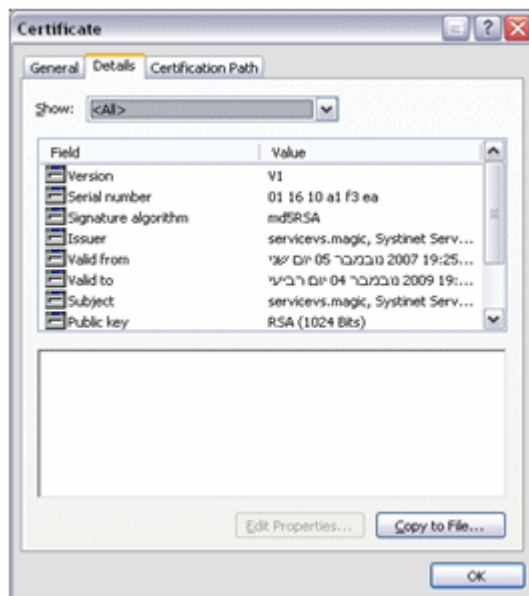
ベンダがすべてのクライアントでサーバを”信頼”させる必要がある場合、この方法が必要です。

1. Web ブラウザ (相互に認証された Web サービスは、この段階でベンダの s 証明書を必要とします)を使用して、WSDL URL にアクセスしてください。

2. セキュリティ・アイコンを見つけ、それをクリックすると、証明書が表示されます。

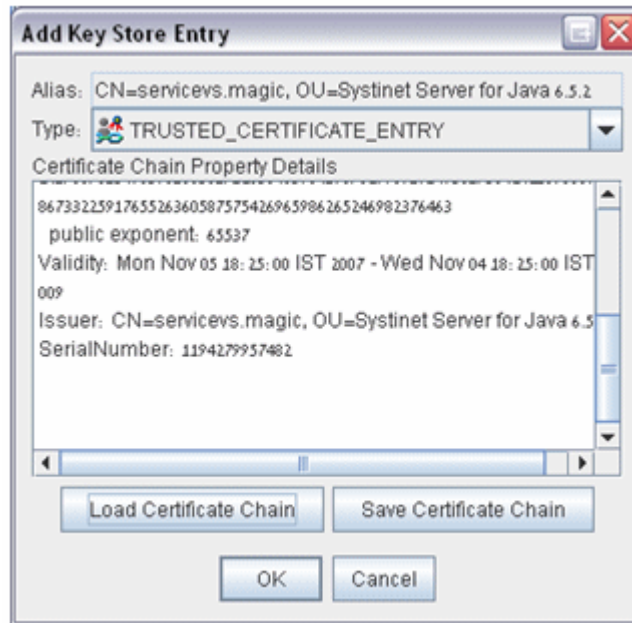


3. [証明書を見る] をクリックして、[詳細] タブに移動します。



4. [Copy to File] (デフォルトを使用)をクリックして証明書をファイルで保存してください。
5. コマンド%WASP_HOME%\bin\PStoreTool.bat --gui を使用してクライアント側のPStore ツールを開きます。
6. [PStore] メニューをクリックして **Open from file** を選択します。
7. **SSJ\Conf** ディレクトリ内の **clientConf.xml** ファイルを開いてください。
8. [Key Store] メニューから、**Import Alias** を選択します。

9. **Type** を **TRUSTED_CERTIFICATE_ENTRY** に変更し、[Load Certificate Chain] をクリックします。先ほど保存した *.cer ファイルを指定します。



10. [OK] をクリックします。これで、サーバはクライアントに"信頼"されます。

Magic xpi でクライアント証明書を設定する：

Magic xpi 側で Web サービス over SSL のコンシューマとなるために必要な設定は、以下の通りです。

1. Web サービスのステップでは、関連する ID の詳細を送るために、[Identity] ボタンを使用してください。必要なパラメータは、ユーザ名パラメータとして **Pstore** エイリアスとパスワードパラメータとして証明書のパスワードです。クライアント証明書が使用される場合は常に必要です。
2. [設定] ダイアログボックスの [リソース] セクションで、サービスの Web サービスクライアント定義に位置付けてください。[構成] 画面で、[セキュリティ] タブを選択してください。[セキュリティレベル] を **Transport**、[認証タイプ] を **SSL** に変更してください。これらの設定がクライアント証明書を使用する時に必要になります。

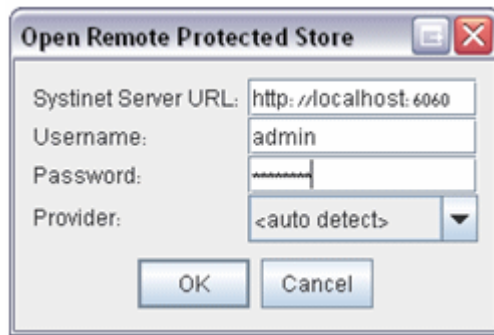
SSL サーバ証明書を変更するには：

デフォルトの Systinet Server 証明書は、**machineName.Domain** として発行されます。

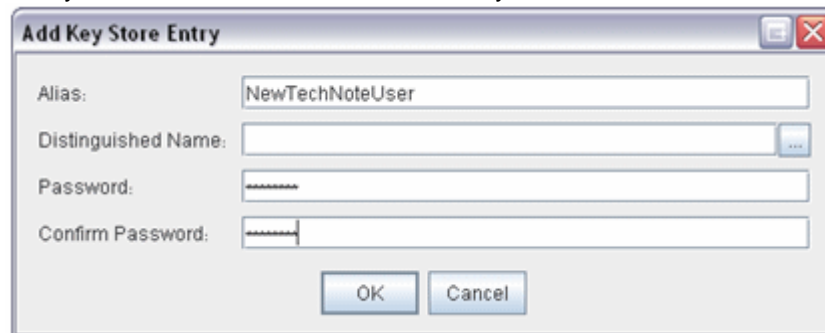
これは他のどの証明書にも変更することが可能で、以下の処理を実行する必要があります。

1. **%WASP_HOME%\bin\PStoreTool.bat --gui** コマンドを使用して PStore ツールを開いてください。

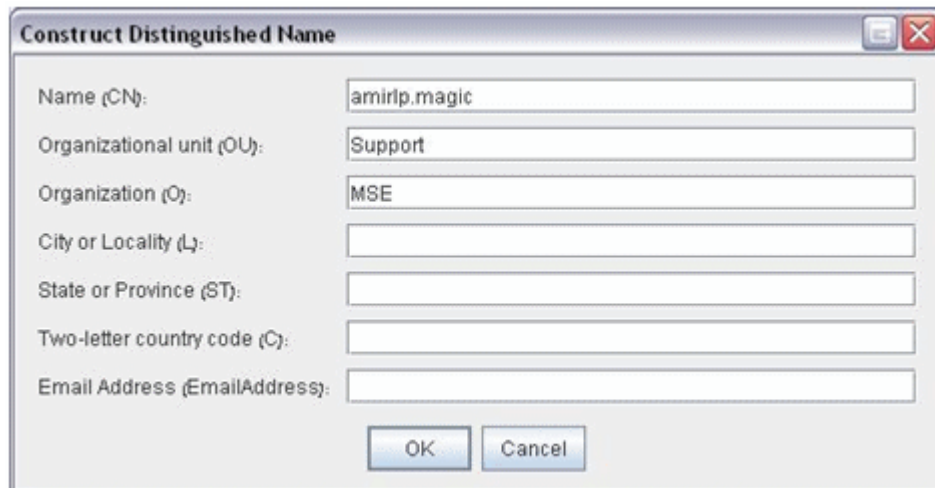
2. **PStore** → **Open Remote** を使用してください (利用可能なサーバが要求されます)。



3. [Key Store] メニューから新規の **Identity** を作成してください。

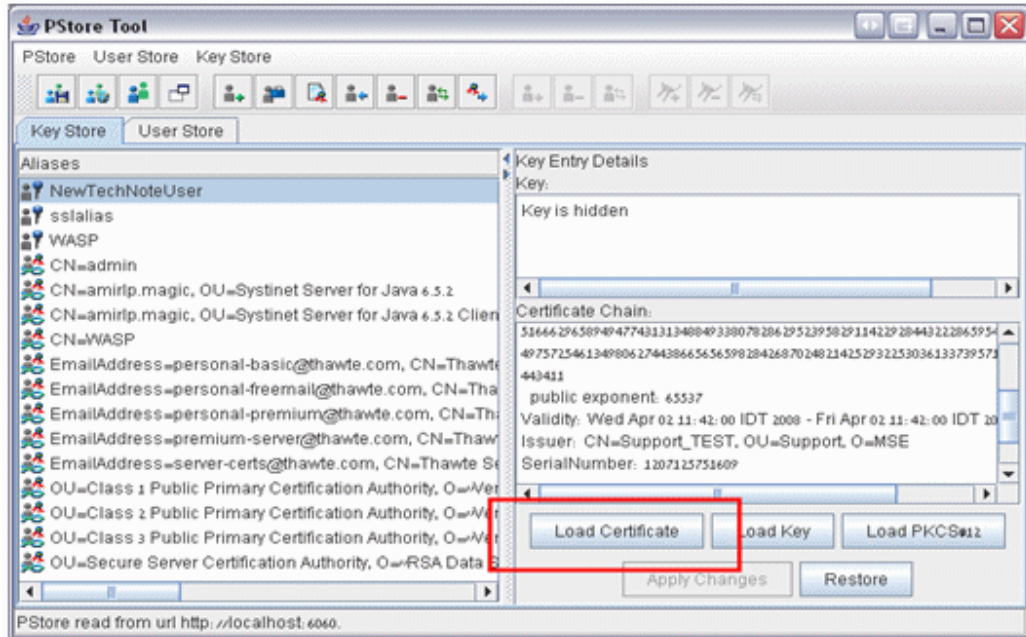


4. 名前とパスワードを入力してください。
5. [Add Key Store Entry] 画面で、[Distinguished Name] ボタンをクリックして、必要な情報を追加してください。

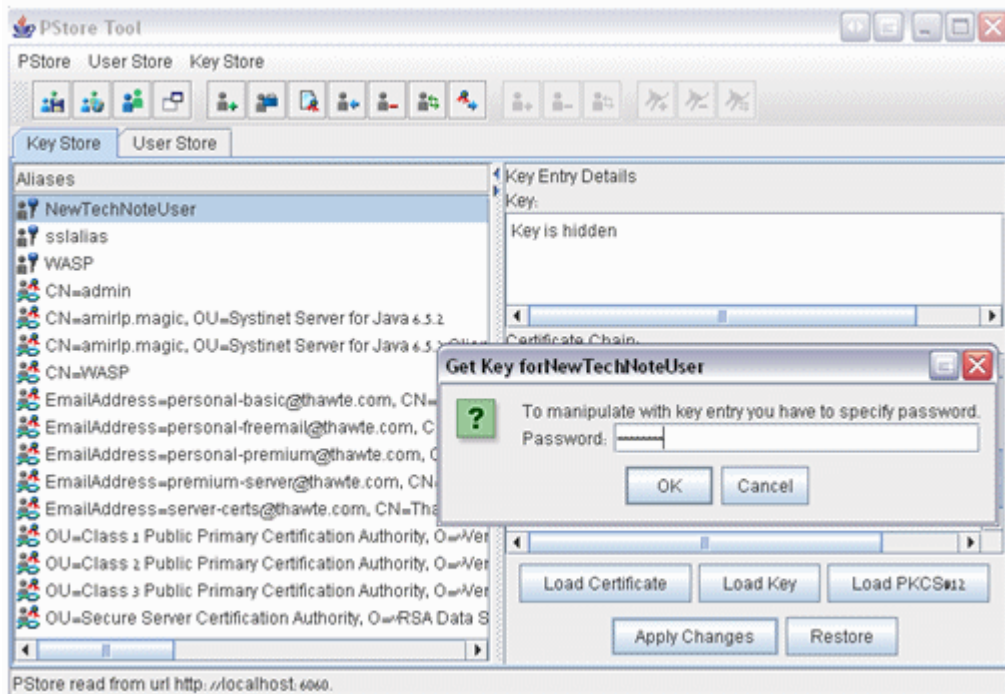


6. この新しい ID 上で右クリックしコンテキストメニューから [Certificate request] を選択してください。PC から証明書要求が発行されることは、極めて重要です。
7. 証明書リクエストを TXT ファイルとして保存して、CA (認証局)を使用している証明書を作成してください。

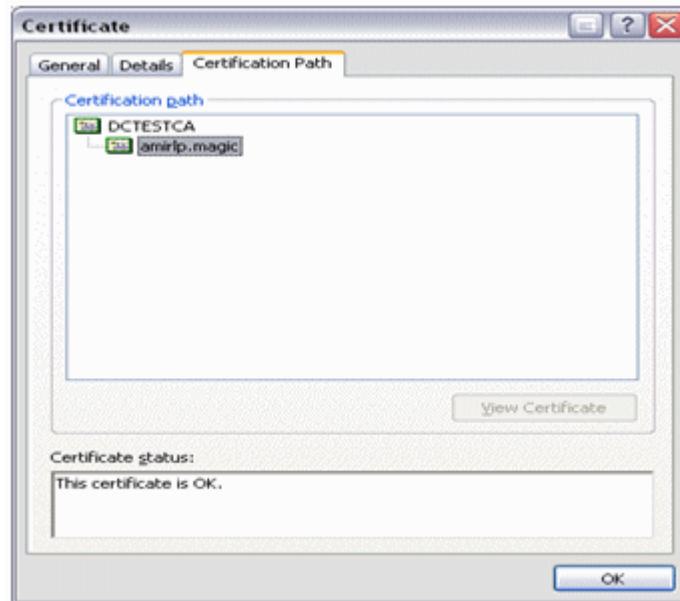
8. 新しい ID 上でパークしているとき、[Load Certificate] ボタンをクリックして、新しい ID に新しい証明書を読み込ませてください。



9. CA (認証局)によって作成されるファイル (*.cer)を選択してください。
10. [Apply Changes] ボタンをクリックすることで確認してください (識別パスワードが必要です)。



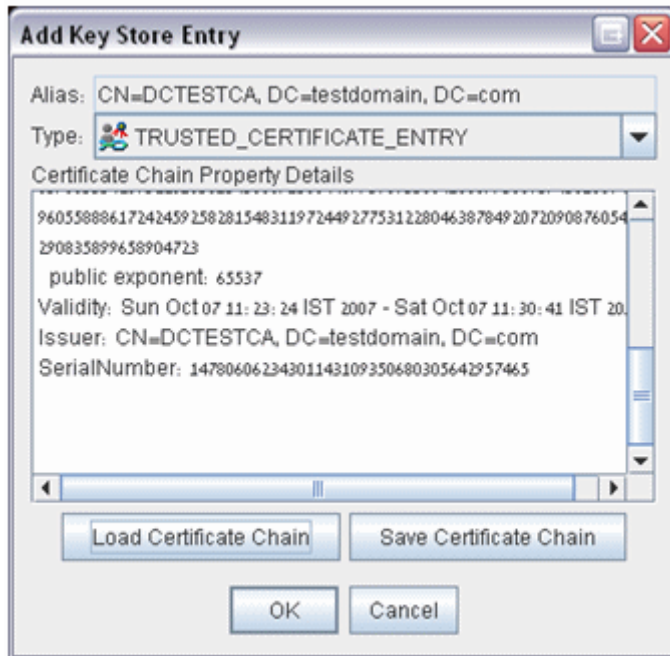
11. 証明書の発行者が信頼されていることを確認してください。下の図のように証明書の発行者が **DCTESTCA** であることに注意してください。



証明書の発行者が“信用される”ようにするために必要なステップは、以下の通りです。

- a. (上記に表示されているような)証明書を開いて、発行者行の上でパークしてください。
[View Certificate] ボタンが、利用できるようにならなければなりません。これをクリックして、2 番目の証明書を開いてください。
- b. [Detail] タブを開き、[Copy to file] をクリックして *.cer ファイルを保存してください。

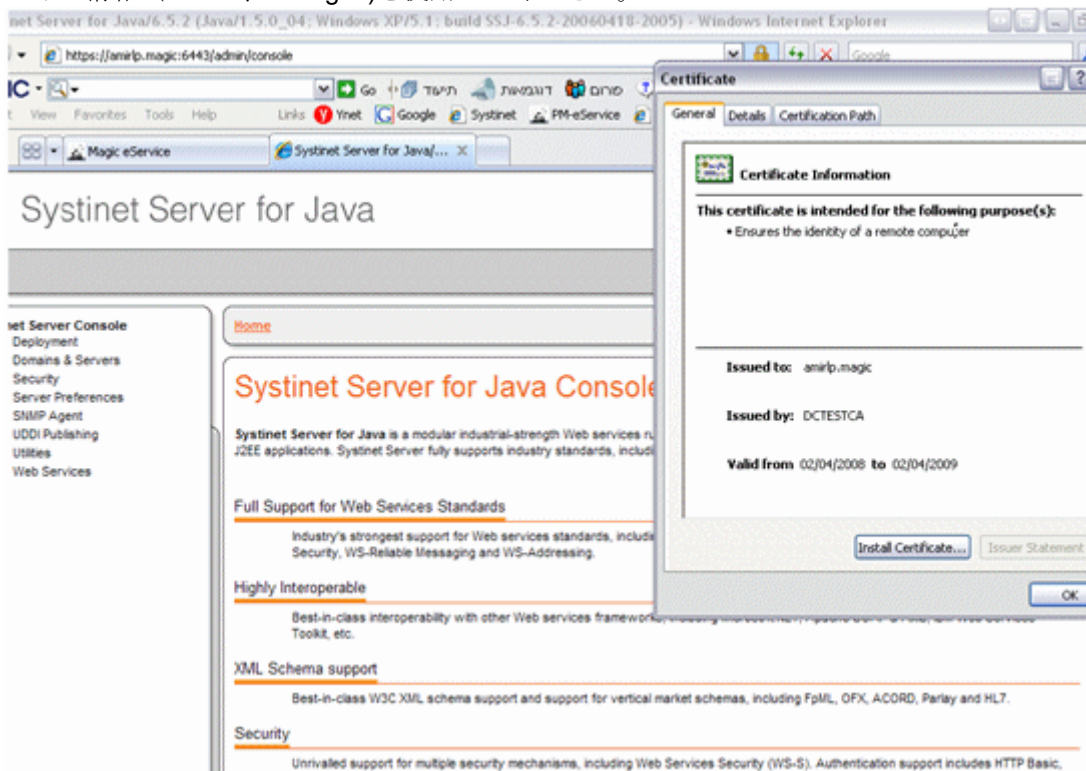
- c. PStore で **KeyStore** → **Import alias** を選択して、Type を **TRUSTED_CERTIFICATE_ENTRY** に変更します。[Load Certificate Chain] ボタンをクリックして、.cer ファイルを選択します。



- d. **SSJ\Conf** ディレクトリの **ServerConf.xml** ファイルを開いてください。
- e. エイリアスとして新しい ID を、パスワードとしての新しい ID のパスワードを指定することで、[httpsPreferences] セクションを変更してください。:

```
<httpsPreferences name="https">
  <alias>NewTechNoteUser</alias>
  <defaultTimeout>40000</defaultTimeout>
  <maxIdleTime>500</maxIdleTime>
  <maxReadTime>40000</maxReadTime>
  <maxThreads>255</maxThreads>
  <minThreads>5</minThreads>
  <needsClientAuth>false</needsClientAuth>
  <nonProxyHosts>localhost</nonProxyHosts>
  <password>TechPass</password>
  <port>6443</port>
</httpsPreferences>
```


- f. Systinet を再起動して、管理コンソールを開いてください。ログオンする際、最初のログイン情報 (Admin、changeit) を使用してください。



IIS7 で証明書を処理して、Systinet に SOAP リクエストを送るには

このトピックは、Systinet に SOAP リクエストを送るために IIS7 のルーティング機能を使用する方法を説明しています。そして、2048 ビットのキー長の証明書の使用を有効にしています。

PStore ツールは、1024 バイトの証明キーを作成することができます。その結果、2048 バイト証明書を使用したい場合、Systinet に SOAP リクエストを送る必要があります。

IIS のこの新しいアドオン (<http://www.iis.net/downloads/microsoft/application-request-routing>) を使用することでリクエストをルーティングすることができます。

以下のサイトでこれに関する詳細な情報を確認することができます。

[http://www.iis.net/learn/extensions/configuring-application-request-routing-\(arr\)/define-and-configure-an-application-request-routing-server-farm](http://www.iis.net/learn/extensions/configuring-application-request-routing-(arr)/define-and-configure-an-application-request-routing-server-farm)

[http://www.iis.net/learn/extensions/installing-application-request-routing-\(arr\)/install-application-request-routing](http://www.iis.net/learn/extensions/installing-application-request-routing-(arr)/install-application-request-routing)

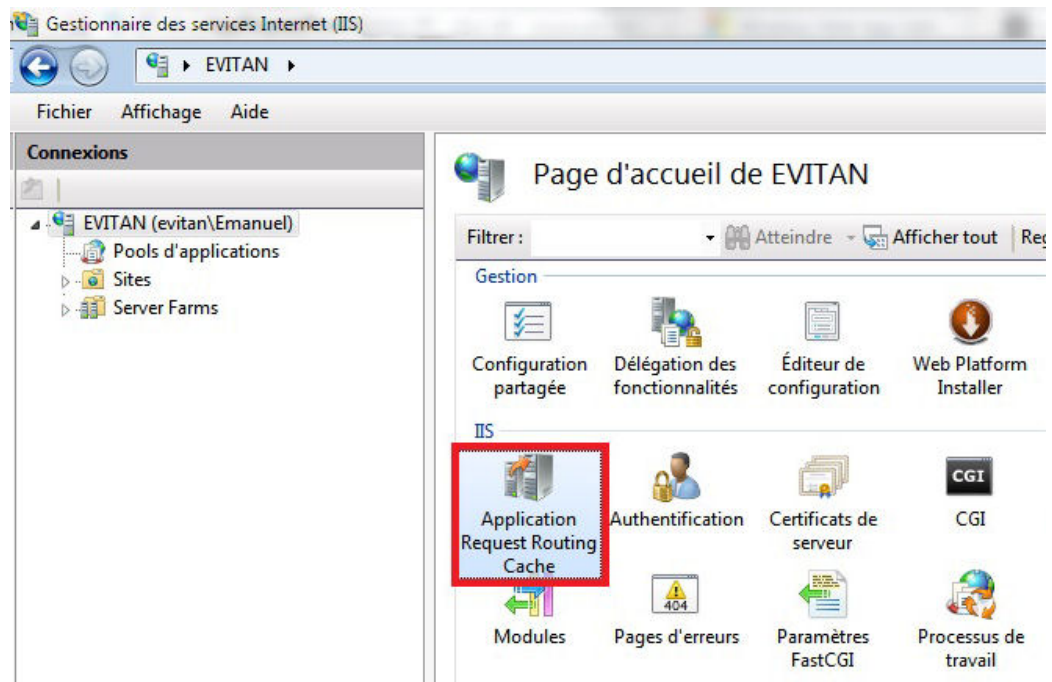
<http://blogs.iis.net/wonyoo/archive/2011/04/20/how-to-install-application-request-routing-arr-2-5-without-web-platform-installer-webpi.aspx>

<http://aaronwalthrath.wordpress.com/2010/05/28/installing-a-certificate-in-iis-7-5-from-a-public-certificate-authority/>

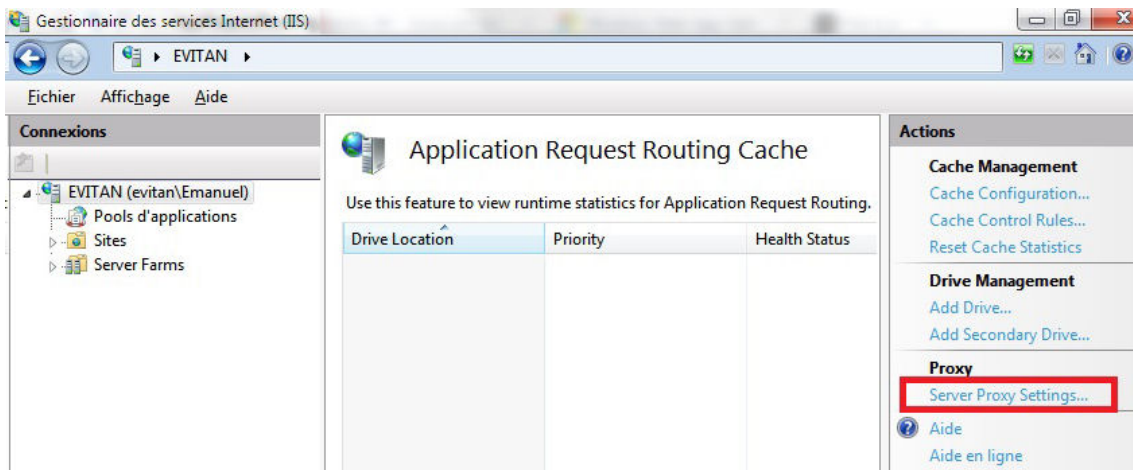
ARRv3.0.exe アドオンをインストールしたら、IIS7 の IIS マネージャーを起動して以下のステップに従ってください。

1. IIS マネージャーの [サーバ] ノードを選択してください (左側のツリービュー上のサーバ名)。

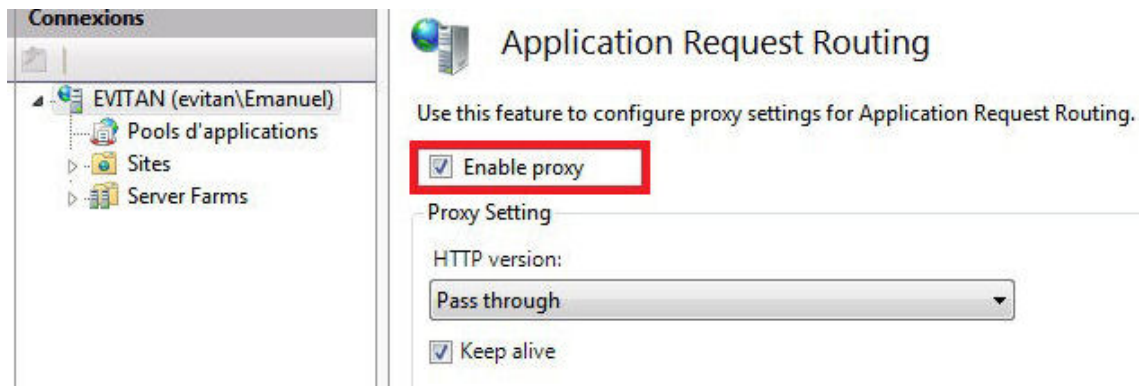
2. [Application Request Routing Cache] アイコンをダブルクリックしてください。



3. 右側の [操作] パネルの、[Server Proxy Settings] をクリックしてください。



4. [Enable proxy] のチェックボックスを選択します。



5. [Use URL Rewrite to inspect incoming requests] チェックボックスを選択してください。

Fichier Affichage Aide

Connexions

- EVITAN (evitan\Emanuel)
 - Pools d'applications
 - Sites
 - Server Farms

Application Request Routing

Use this feature to configure proxy settings for Application Request Routing.

Forwarding proxy header value:

Cache Setting

Memory cache duration (seconds):

☒ Enable disk cache

☐ Enable request consolidation

Query string support:

Ignore query string

Buffer Setting

Response buffer (KB):

Response buffer threshold (KB):

Proxy Chain

Proxy server:

Example: proxy.contoso.com:8080

Proxy Type

☒ Use URL Rewrite to inspect incoming requests

☒ Enable SSL offloading

Reverse proxy:

6. [Reverse proxy] フィールドに、リクエストを送りたいサーバを入力してください。
例、Systinet が 10.1.12.115:6060 (または、vmware7util06.frmagic : 6060) のアドレスで動作している場合、図のように入力してください。

7. SOAP UI で、エンドポイントを IP (またはホスト名) に変更してください。例：
10.1.12.115 (または、vmware7util06)+ WSDL 内のエンドポイントの残り。

アクセスする URL は、以下のように変更する必要があります。

変更前: "http: //10.1.12.115: 6060/MySolution/" (または http://vmware7util06.frmagic:
6060/MySolution/)

変更後: "http: //10.1.12.115/MySolution/" (または http://vmware7util06.frmagic/MySolution/)

これで、サーバの IP アドレス(またはホスト名)とポートは置き換えられます。

通常、Web サービスの仕様が頻繁に変わることはないため、Web サービスの WSDL も頻繁に変更されることはありません。

その結果、先に述べたようにエンドポイントの手動での変更に対応するには、結局 IIS の任意の仮想ディレクトリに置いたファイルに Systinet の WSDL を保存するようにみなすことができ、IIS の URL に基づくエンドポイントを手動で変更することができます。

索引

D

DB2/400 言語サポート 34

H

HTTP トリガ 18

I

IIS

Systinet に SOAP リクエストを送るには 288

IIS7 を使用して証明書を処理する 288

M

MSMQ 21

O

ODS 配列 - 使用する 6

S

SAP B1 - 複数のオブジェクト 211

SAP R/3

Magic xpi を起動させるために SAP R/3 システムを
設定する 213

マスタデータ分布 218

SQL 文 28

SQL 文を作成する 28

SSL

Web サービスのコンシューマ 279

U

Unicode か ANSI、UTF-8 のファイルをメモリに読み込
ませる 121

Unicode または UTF-8 BOM を BLOB ファイルに追加
123

UTF-8 の内容を含むバイナリ BLOB を Unicode BLOB
に変換する 126

W

Web サービス

Web Services Over SSL のコンシューマ 279

Web サービス Over SSL のコンシューマ 279

え

エンコード

方法 114

Unicode か ANSI、UTF-8 のファイルをメモリに
読み込ませる 121

Unicode か UTF-8 BOM を BLOB ファイルに追加
するには 123

UTF-8 の内容を含むバイナリ BLOB を Unicode
BLOB に変換するには 126

データマッピングを使用する場合、Unicode か ANSI、
UTF-8 のフラットファイルでレコードを処理する
122

ファイルが指定されたエンコードで定義されるこ
とを確認するには 114

指定されたエンコードでファイルを作成する 117

エンコードの問題 114

た

ダイナミックなテキストファイル 35

タイムアウト - フロー 150

て

データベース 70

動的 WHERE 句 70

データマッピング 30

SQL 文 28

データマッピングを使用する場合、Unicode か ANSI、
UTF-8 のフラットファイルでレコードを処理する
122

データ共有 6

テーブルカラム

更新 37

テーブルカラムの更新 37

ふ

ファイルが指定されたエンコードで定義されることを
確認する 114

フロータイムアウト 150

ほ

ポーリング 4

る

ループ 9

漢字

証明書

IIS7 で証明書を処理する 288

長いトランザクション (ワークフロー) 3

特定のエンコードでファイルを作成 117

保証された配達 2