

Fiorano®

Fiorano SOA プラットフォーム バージョン管理とデプロイメント ルール

対象バージョン : 2007 SP7 および 9.0.0

はじめに

このガイドブックは、Fiorano SOA プラットフォームにおけるバージョン管理とデプロイメント ルールについて説明するものです。

このガイドブックは、以下のガイドブックで説明されている知識を有していることを前提としています。

- Fiorano SOA プラットフォームの起動方法
- Fiorano SOA プラットフォームのアーキテクチャ概要
- Fiorano オーケストレータによるビジネス プロセス構築の概要 (基礎編)

目次

1 デプロイメント管理の重要性.....	3
2 コンポーネント フローのバージョンニング	3
2.1 リポジトリ上での管理	3
2.2 バージョン番号の付与方法.....	4
2.2.1 最初のバージョン	4
2.2.2 コメントの記入	6
2.2.3 バージョン アップ (新規バージョンの作成).....	7
3. デプロイメント管理.....	9
3.1 デプロイメント ルール	9
3.1.1 ラベル.....	9
3.1.2 ルール シンタックス	10
3.2 例題.....	12
4. Deployment Manager の操作.....	14
4.1 Deployment Manager の起動	14
4.2 ルールの設定方法.....	16
4.2.1 新規ルールの作成	16
4.2.2 ルールのテスト.....	20

1 デプロイメント管理の重要性

Fiorano SOA プラットフォームの最も優れた特徴の一つは、ビジネス プロセス変更の柔軟性にあります。

ビジネス側の要求に俊敏に応えていくことは、既コンポーネント フローの絶え間ない変更を意味します。このことは、コンポーネント フローの開発者によるバージョン管理のみならず、システム管理者によるコンポーネント フローのデプロイメント管理が重要であることを示しています。

Fiorano SOA プラットフォーム は、コンポーネント フローのデプロイメントを信頼のおける方法で実行する強力なツールを備えています。Deployment Manager と呼ばれるこのツールは、ルール エンジンをベースとしたもので、本番環境での実行ルールにしたがってコンポーネント フローやサーボス コンポーネントを指定されたピア サーバーに確実に間違いなくデプロイします。

このガイドブックでは、Studio におけるコンポーネント フローのバージョンングと Deployment Manager によるデプロイメント管理の方法を説明します。

2 コンポーネント フローのバージョンング

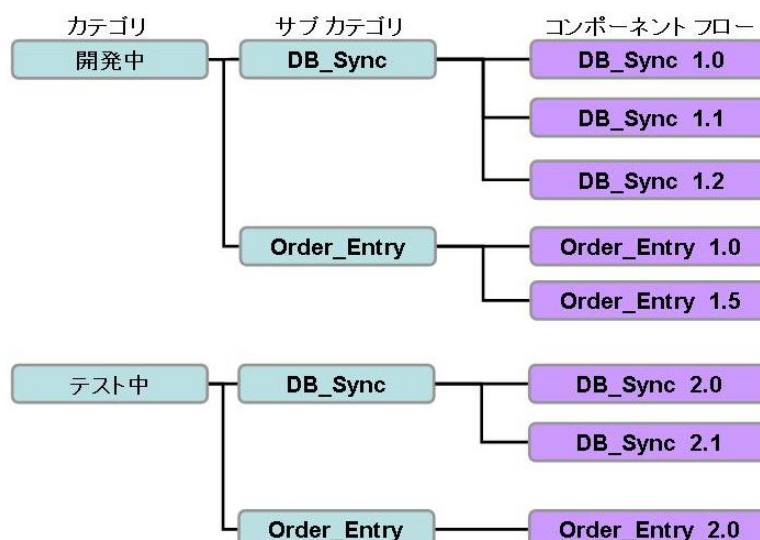
2.1 リポジトリ上での管理

コンポーネント フローには、開発者が任意のバージョン番号を付けて、ESB サーバー (Enterprise Server) のリポジトリに格納することができます。

バージョンの番号付けルールと保存フォルダー (カテゴリ) の作成ルールを工夫することで、より効果的な管理が行えるようになります。

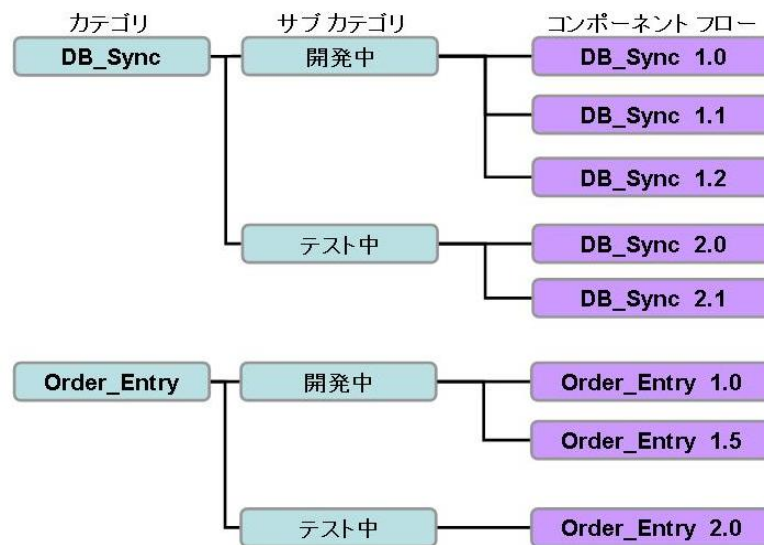
次の例 1 では、バージョン番号 1.x が開発中のバージョンを、2.x がテスト中のバージョンを示すようルールを設定しています。また、コンポーネント フローの保存フォルダー (カテゴリ) を、開発中、テスト中などのステージ別に分けています。さらに、ステージ カテゴリの下にコンポーネント フロー毎のサブ カテゴリを設けています。

例 1



例 2 では、バージョン番号のルールは 例 1 と同じですが、カテゴリをコンポーネント フロー毎に設け、その下にステージを区分けするサブ カテゴリを設定しています。

例 2



[注意]

コンポーネント フローが実行される際には、GUID (コンポーネント フローの名前) とバージョン番号によって識別され、リポジトリの格納場所 (カテゴリ) は識別の対象となりません。したがって、カテゴリが異なるからといって、同じ GUID とバージョン番号を持つコンポーネントを 2 つ以上のカテゴリに置くことはできません。

例えば、「開発中」カテゴリにコンポーネント フロー “DB_Sync 1.0” を置き、「テスト中」カテゴリにも “DB_Sync 1.0” を置くことはできません。別の言い方をすれば、開発中のバージョン 1.0、テスト中のバージョン 1.0 という番号付けルールは使用できないということです。

[注意]

バージョン番号として許されているのは、0 から 99 までです。ピリオド (小数点) 以下も、0 から 99 までです。ただし、0.0 はバージョン番号として許されていません。

2.2 バージョン番号の付与方法

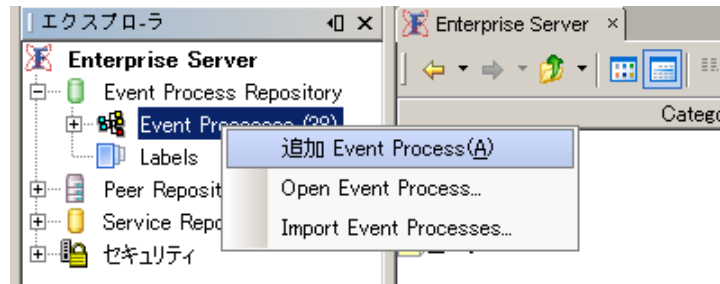
2.2.1 最初のバージョン

コンポーネント フローを新たに作成する方法には、次の 2 つがあります。

- スクラッチから作成
- 既存のコンポーネント フローを利用 (コピー) することで作成

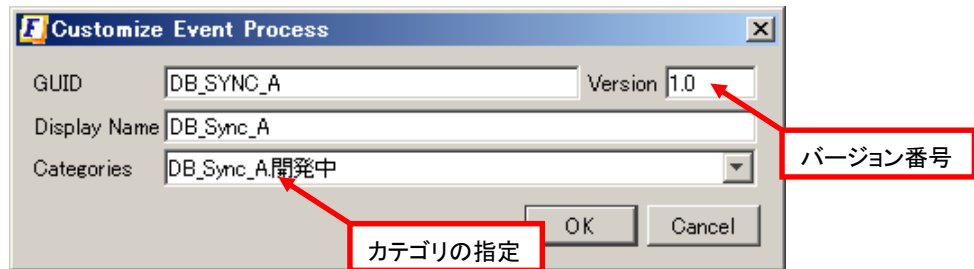
スクラッチから作成する場合

下のキャプチャ画面のように Event Process Repository の下の Event Processes ノードを右クリックして、[追加 Event Process] を選択します。



次のダイアログ ボックスが表示されます。このダイアログ ボックスでは、次の項目を指定します。

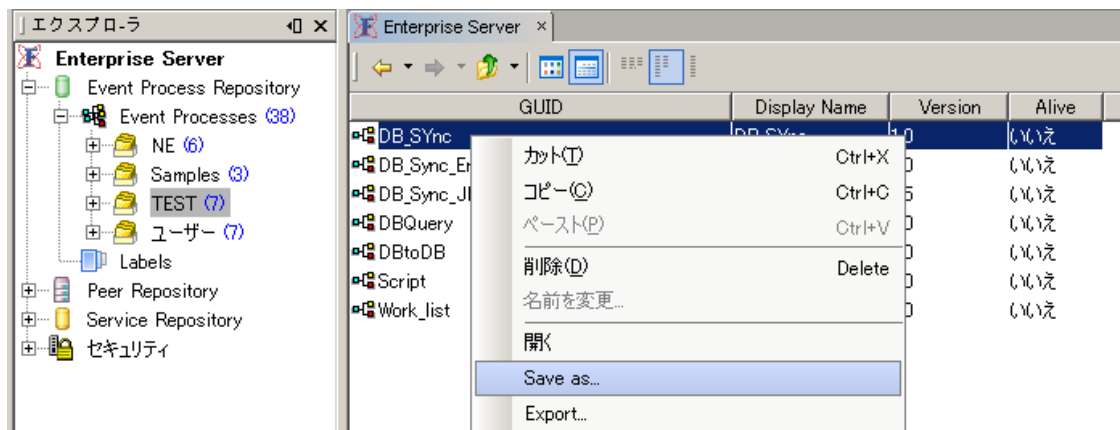
- GUID** : 全体で一意的コンポーネント フロー名 (大文字と一部の記号)
- Display Name** : ツールなどに表示される名前 (GUID と同一の名前であるが小文字も可)
- Categories** : リポジトリに格納される際のカテゴリ (フォルダー)
 サブ カテゴリは . (ピリオド) で区切って指定する
 例 : カテゴリ.サブ.サブサブ
- Version** : バージョン番号 (デフォルトの番号は 1.0)
 ここで、初期バージョン番号を指定することができます。



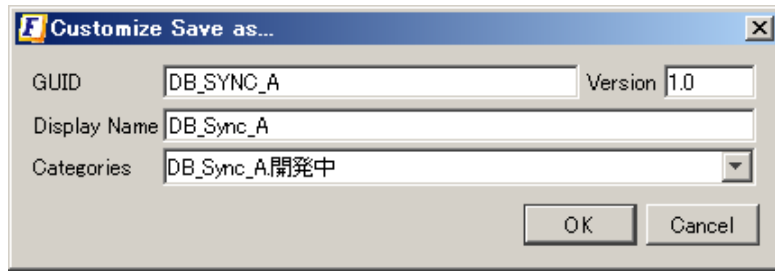
[OK] ボタンをクリックすると、新規のコンポーネント フローが作成されます。ただし、中身は空です。

既存のコンポーネント フローを利用 (コピー) することで作成する場合

既存のコンポーネント フローを利用して最初のバージョンを作成する場合には、利用するコンポーネント フローを右クリックし、メニューから [Save as ...] を選択します。

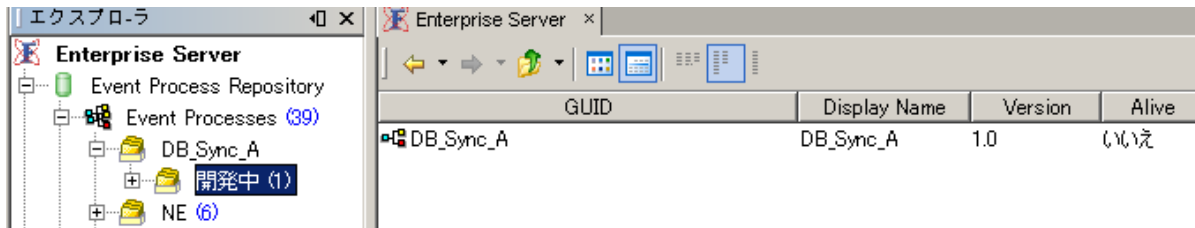


次のダイアログ ボックスが表示されますので、必要な情報を入力します。



- GUID** : 全体で一意的コンポーネント フロー名 (大文字と一部の記号)
- Display Name** : ツールなどに表示される名前 (GUID と同一の名前であるが小文字も可)
- Categories** : リポジトリに格納される際のカテゴリ (フォルダー)
 サブ カテゴリは . (ピリオド) で区切って指定する
 例 : カテゴリ.サブ.サブサブ
- Version** : バージョン番号 (デフォルトの番号は 1.0)
 ここで、初期バージョン番号を指定することができます。

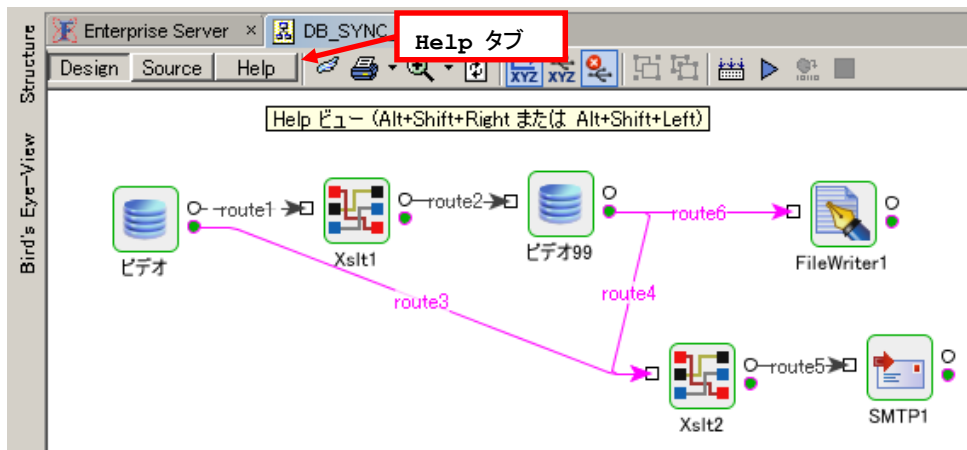
[OK] ボタンをクリックすると、次の画面のようにコンポーネント フローが作成されます。コンポーネント フローの内容は、元のフローと全く同じです。



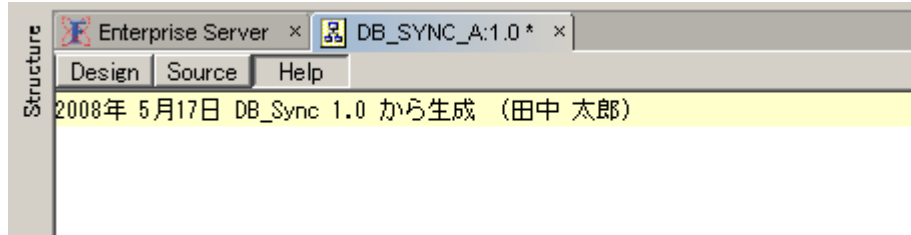
2.2.2 コメントの記入

コンポーネント フローには、コンポーネント フローについての説明を記述することができます。これを利用して、バージョンの毎の説明や前バージョンからの変更点を記しておくことができます。

説明を記述するには、[Help] タブをクリックします。

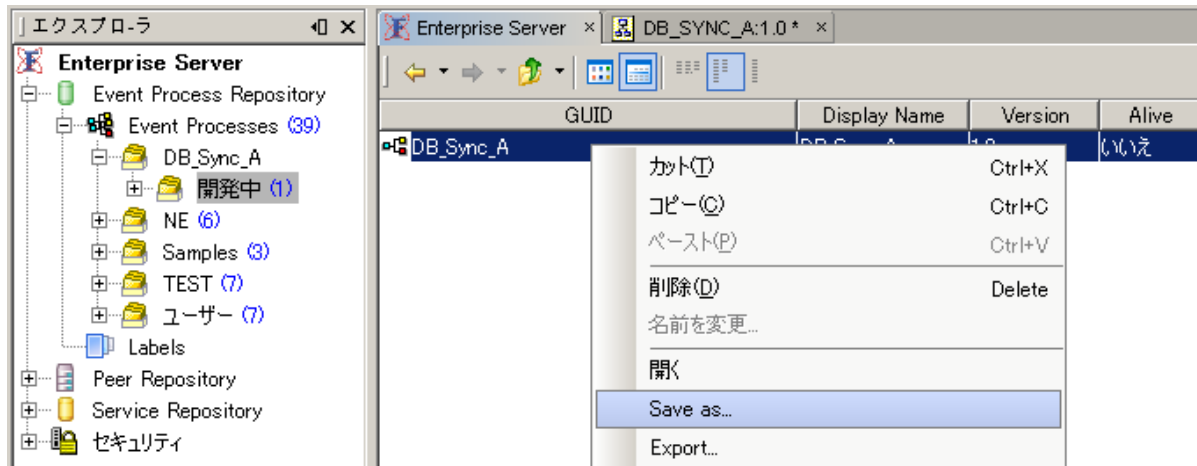


Help の内容は自由に記述できます。

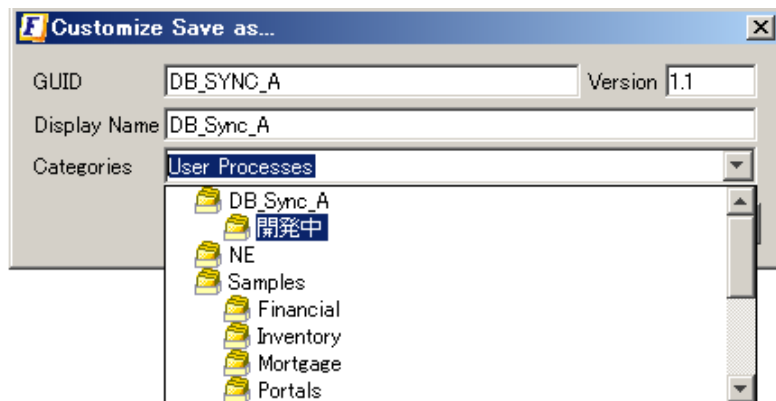


2.2.3 バージョン アップ (新規バージョンの作成)

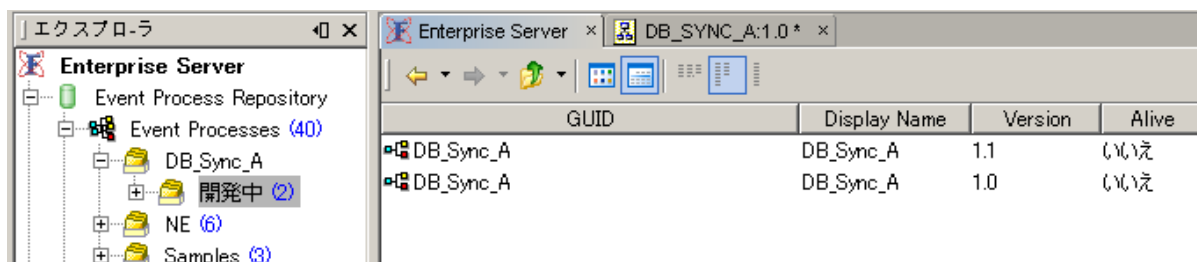
新たなバージョンを作成するには、元とするバージョンのコンポーネント フローを右クリックし、メニューから **[Save as ...]** を選択します。



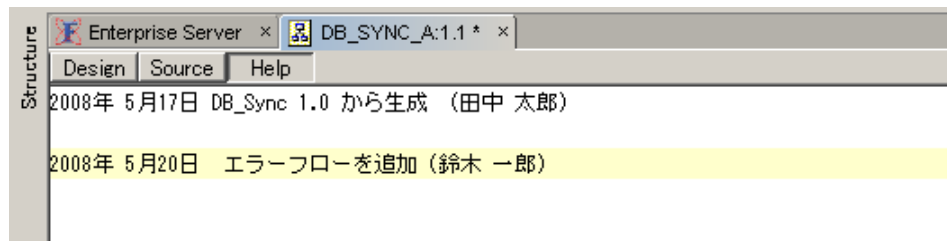
表示されるダイアログ ボックスに値を入力します。下のキャプチャ画面では、バージョン番号を 1.1 に変更し、カテゴリをプルダウン メニューから選択しています。



[OK] ボタンをクリックすると、次の画面のようにバージョン 1.1 が作成されます。



コンポーネント フローを開き、[Help] タブを選択します。旧バージョンの Help 記述もコピーされていますので、このバージョンに対する説明を追記します。



3. デプロイメント管理

Fiorano SOA プラットフォーム には、指定されたルールに基づいてコンポーネント フローをデプロイメントする Deployment Manager が備わっています。Deployment Manager は、ルールを解釈し、実行するルール エンジンです。

3.1 デプロイメント ルール

Deployment Manager のルールは、GUID、バージョン番号、ラベル の 3 つをデプロイメント条件に使用します。

3.1.1 ラベル

すべてのコンポーネント フローおよびサービス コンポーネントには、ラベルが付けられています。

コンポーネント フローとユーザーが開発したサービス コンポーネントには、下記のラベルを任意に選択して変更することができます。

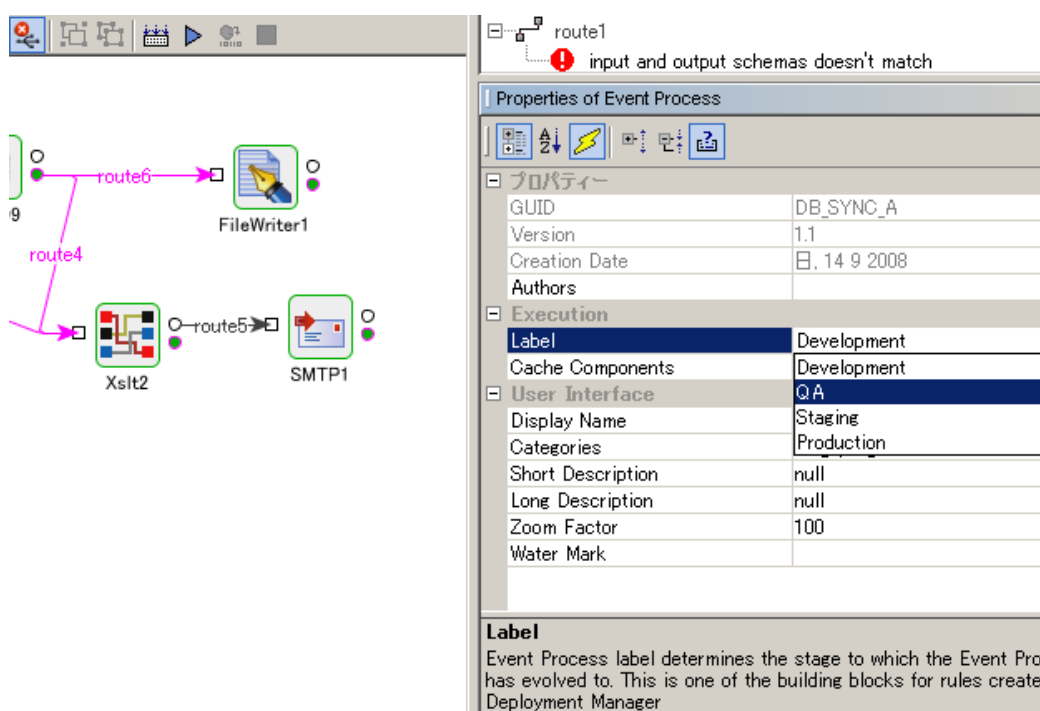
- **Development** : 開発中 (デフォルトのラベル)
- **QA** : テスト中
- **Staging** : ステージング (本番待機中)
- **Production** : (本番用)

Fiorano 社がプリビルトして製品に組み込んでいるサービス コンポーネント (プリビルト コンポーネント) は、Production のラベルが付いており、これを変更することはできません。

コンポーネント フローのラベル変更

コンポーネント フローのラベルを変更するには、コンポーネント フローを開き、表示されるプロパティ ウィンドウで指定します。

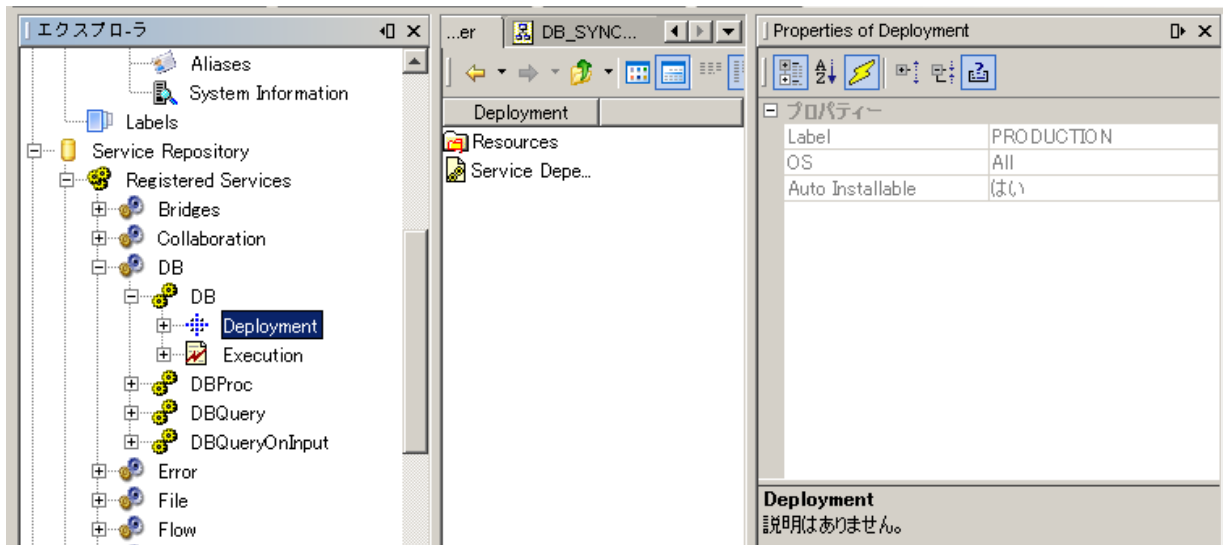
[Label] 項目の入力エリアのプルダウン メニューからラベルを選択します。



サービス コンポーネントのラベル

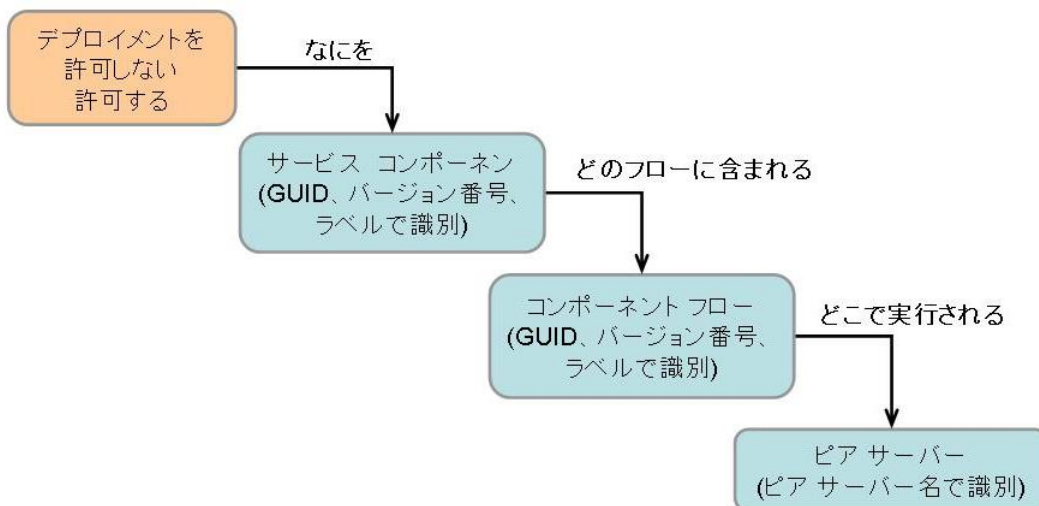
サービス コンポーネントのラベルは、エクスプローラ ウィンドウで **Service Repository** → **Registered Service** と展開すると、システムに登録されているサービス コンポーネントがカテゴリ別に表示されます。登録済みのユーザー開発コンポーネントもここに表示されます。コンポーネントを選択し、**Deployment** ノードを選択すると、右側のプロパティ ウィンドウにコンポーネントの **Label** が表示されます。

下のキャプチャ画面は、DB コンポーネントの **Deployment** を選択した状態です。Label が **PRODUCTION** になっていることがわかります。(すべて大文字表示なっていますが、ラベルは大文字、小文字を区別しません。)



3.1.2 ルール シンタックス

ルールのシンタックスは、次の図のように表すことができます。



図中水色で示している対象 (サービス コンポーネント、コンポーネント フロー、ピア サーバー) の指定は、省略することができます。

例えば、すべての対象を指定したルールは、(文章で表すと) 次のようになります。

『 GUID が “ABC” のコンポーネント フローに含まれる、GUID が “XYZ” のコンポーネントを、ピア サーバー FPS1 へデプロイメントすることを許可しない 』

この例は、サービス コンポーネント単位のデプロイメント条件となります。

サービス コンポーネントを省略したルールの例は、次のようになります。

『 ラベル “Development” のコンポーネントを、ピア サーバー FPS ヘデプロイメントすることを許可しない 』

このルールではコンポーネント フロー単位の条件となり、そこに含まれるサービス コンポーネントはすべて一様にデプロイメントが許可されません。

ピア サーバーを省略したルールの例は、次のようになります。

『 GUID “ABC” のコンポーネント フローのデプロイメントを許可しない 』

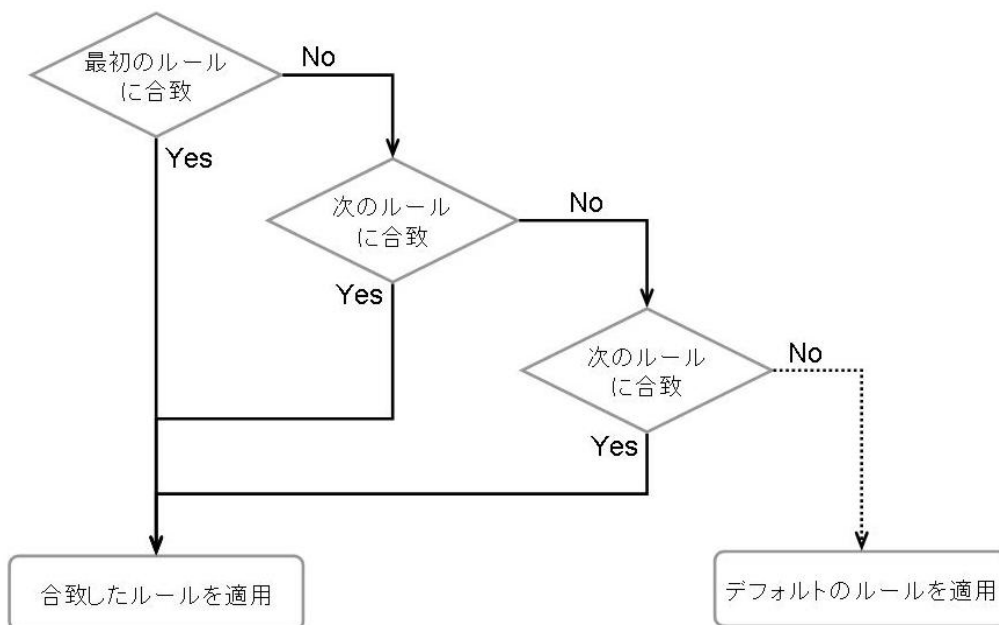
また、対象を識別するための識別条件は複数指定できます。例えば、

『 GUID が “ABC” で、ラベルが “Development” と “QA” と “Staging” 』
 といった指定が可能です。

このように、ルール対象を省略するほど、ルールの適用範囲が広がります。ルール対象や識別条件を多く指定するほど、ルールは複雑になり、適用範囲も広がります。

ルールの順序

ルールは、複数設定できます。複数のルールが設定されている場合、次のフローチャートのような設定の古い順に解釈され、合致したルールが適用されます。すべてのルールに合致しなかった場合には、デフォルトのルールが適用されます。

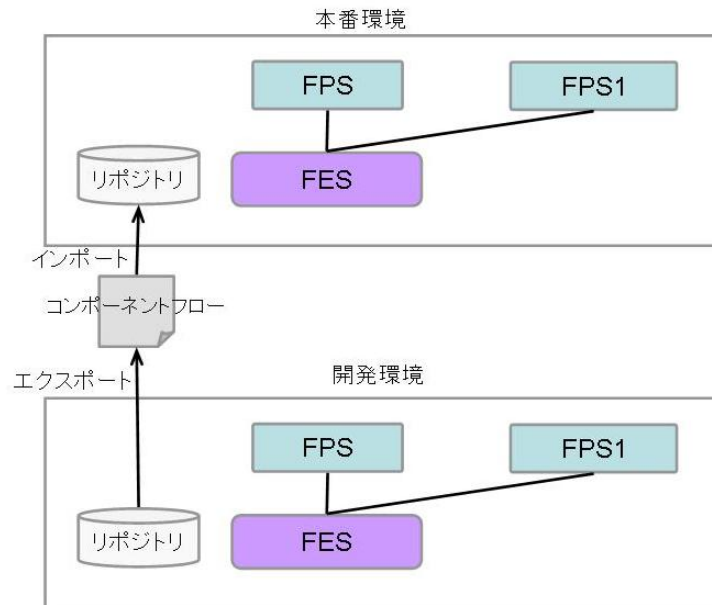


デフォルト (なにもルールが設定されていない状態) では、すべてのサービス コンポーネント、コンポーネント フローがどのピア サーバーへもデプロイメントが許可されています。

3.2 例題

例題 1

下図のように開発環境と本番環境が別個に設けられています。開発およびテストが完了したコンポーネント フローは、開発環境からエクスポートし、本番環境にインポートされます。



開発環境からテスト未完了のコンポーネントが間違えてエクスポートされてきてもいいように、本番環境のデプロイメント ルールを定めます。

まず、開発およびテストが完了したコンポーネント フローのラベルは必ず “PRODUCTION” に変更するという運用ルールを定めます (これは、Fiorano SOA で体系的に定めることはできず、人的な作業でカバーします)。次に、本番環境のデプロイメント ルールとして、『ラベルが “PRODUCTION” コンポーネント フローのみを許可する』を設定します。

『すべてを許可』というのがデフォルトのルールですので、このデフォルト ルールが適用されないようにします。

『ラベル “PRODUCTION” のコンポーネント フローを許可』というルールを設定しても、このルールに合致しないコンポーネント フローも最後にデフォルトのツールが適用され、すべてのコンポーネント フローのデプロイメントが可能となってしまいます。前セクションのフローチャートを参照してください。

この例題のルールは、

『ラベルが “Development”、“QA”、“Staging” のコンポーネント フローは許可しない』

とする必要があります。

このルールによって、“PRODUCTION” 以外のラベルを持ったコンポーネント フローはデプロイメントできなくなります。“PRODUCTION” ラベルのコンポーネント フローは、このルールに合致せず、デフォルトのルールによって許可されることとなります。

このルールでは、ラベルが “PRODUCTION” であれば、どのコンポーネント フローもデプロイメントできます。特定のコンポーネント フロー (例えば GUID が “ABC”) のみを許可する場合には、さらに細かなルールを設定します。

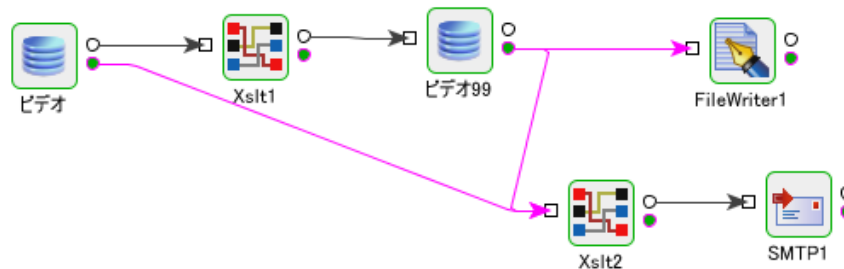
これを可能とするルールは、次のように 2 つのルールを設定する必要があります。

次の順番でルールを設定します。

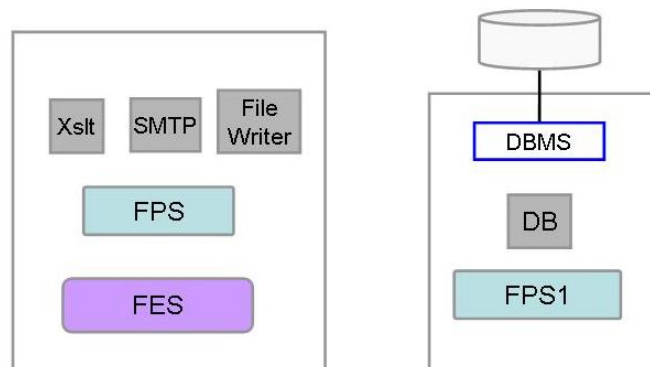
1. 『 GUID “ABC”、ラベル “PRODUCTION” のコンポーネント フローを許可する 』
2. 『 ラベル “DEVELOPMENT”、”PRODUCTION”、”QA”、”STAGING” のコンポーネント フローを許可しない 』

例題 2

下に示すコンポーネント フロー “DB_Sync” をデプロイメントするルールを考えます。



コンポーネント フローは、FPS と FPS1 という 2 つのピア サーバー上で実行されます。ピア サーバー FPS1 が稼働しているマシンでは、データベースも稼働しています。処理パフォーマンスを考慮し、FPS1 は DB コンポーネント専用のピアサーバーとします。



設定するルールは、FPS1 上に DB コンポーネント以外のコンポーネントがデプロイメントできないようにするルールです。

1. 『 GUID “DB” のコンポーネントを “FPS1” にデプロイメントすることを許可 』
2. 『 すべてのサービス コンポーネントの “FPS1” へのデプロイメントを許可しない 』

この 2 つのルールによって、DB コンポーネント以外のコンポーネントが FPS1 にデプロイメントされることが防げます。これをベースにさらなる条件を追加していくことで、より細かな管理が行えます。

4. Deployment Manager の操作

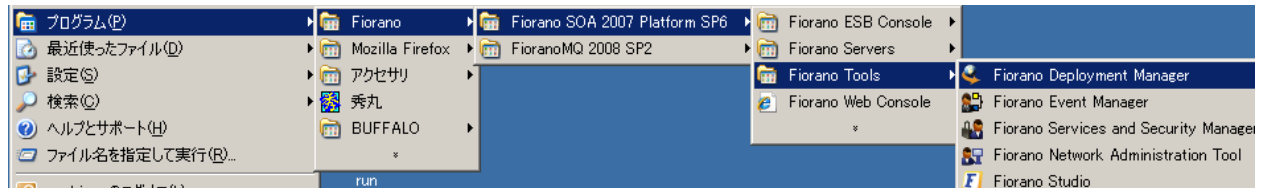
4.1 Deployment Manager の起動

Deployment Manager は、スクリプト `dm.bat` (Unix / Linux 版の場合は、`dm.sh`) によって起動できます。

場所

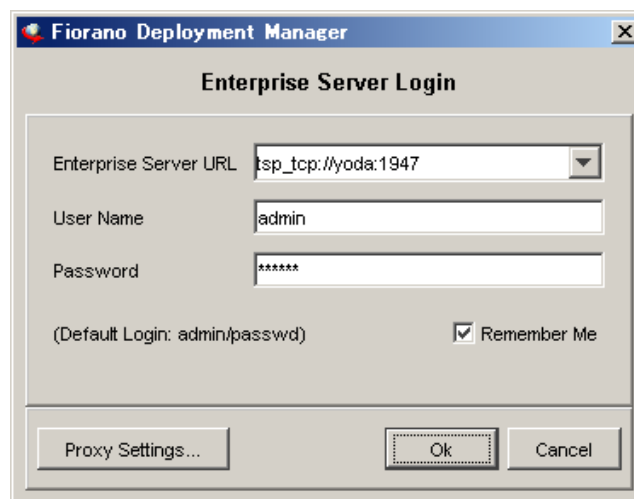
<インストール ディレクトリ>/`esb/tools/dm/bin`

Windows 版では、スタート メニューからも起動できます。



Deployment Manager が起動すると、次のログイン画面が表示されます。入力情報については、下記を参照してください。

ESB サーバーが起動していないとログインできませんので、注意してください。



Enterprise Server URL

ログインする ESB サーバー (Enterprise サーバー) の URL を指定します。

デフォルト値として、次の値があらかじめ設定されています。

`tsp_tcp://localhost:1947`

デフォルトの ESB サーバー コンフィグレーションでは、各ツールは ポート 1947 に TCP プロトコルによって接続するよう設定されています。ESB サーバーのコンフィグレーション設定を変更している場合は、それに合わせて URL を変更します。また、別マシン上の ESB サーバーにログインする場合には、localhost ではなく、ESB サーバーの IP アドレス (またはリモート ホスト名) を指定します。

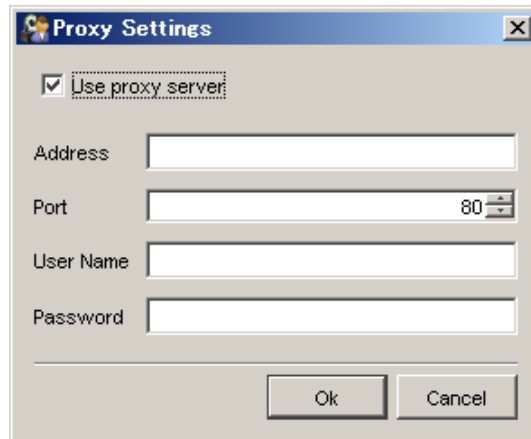
User Name、Password

登録されているユーザー名とパスワードを指定します。デフォルト設定では admin ユーザーとパスワード “passwd” でログインできます。

Proxy Settings

プロキシ サーバーを介して ESB サーバーに接続する場合には、[Proxy Settings ...] ボタンをクリックします。表示されるダイアログ ボックスに、プロキシ サーバーへの接続情報を入力します。

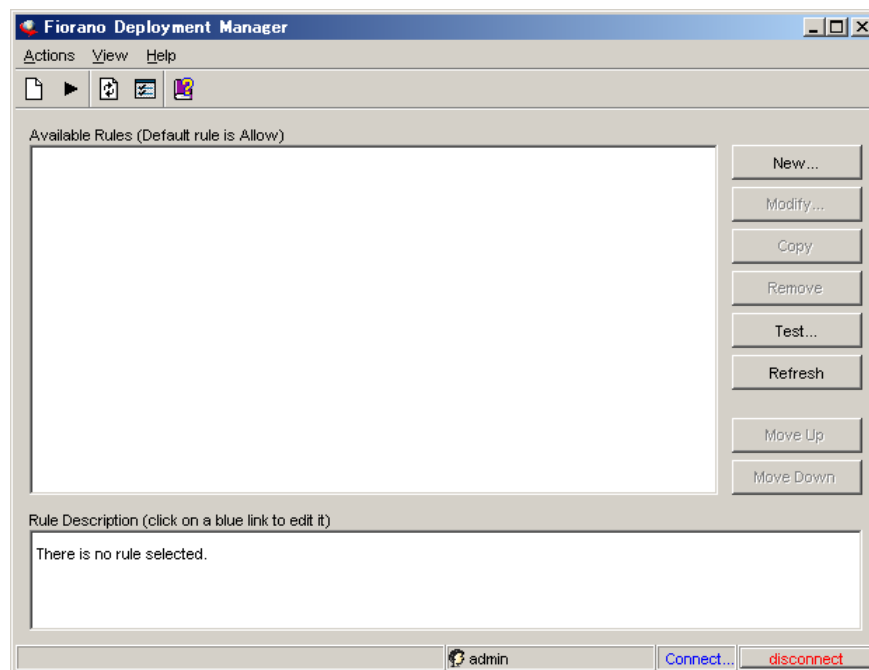
情報を入力したら、[Ok] ボタンをクリックし、最初のログイン画面に戻ります。



The image shows a 'Proxy Settings' dialog box with the following fields and controls:

- Use proxy server
- Address:
- Port:
- User Name:
- Password:
- Buttons: Ok, Cancel

ログインに成功すると、次の画面が表示されます。初期状態では、ルールはなにも設定されていません。



The image shows the 'Fiorano Deployment Manager' main window with the following elements:

- Menu: Actions, View, Help
- Toolbar: File, Play, Refresh, Print, Help
- Available Rules (Default rule is Allow): An empty list box.
- Buttons on the right: New..., Modify..., Copy, Remove, Test..., Refresh, Move Up, Move Down
- Rule Description (click on a blue link to edit it): A text area containing 'There is no rule selected.'
- Status bar: admin, Connect..., disconnect

4.2 ルールの設定方法

前章で説明した例題を例にして、ルール設定の方法を説明します。

例題 1 のルールを設定します。例題 1 のルールは次の 2 つからなっています。

1. 『 GUID “ABC”、ラベル “PRODUCTION” のコンポーネント フローを許可する 』
2. 『 ラベル “DEVELOPMENT”、”PRODUCTION”、”QA”、”STAGING” のコンポーネント フローを許可しない 』

4.2.1 新規ルールの作成

[New] ボタンをクリックすると、ルール作成ウィンドウが表示されます。

The screenshot shows a 'New Rule' dialog box with the following details:

- Radio buttons: Allow, Disallow
- Section: Execution of Business Component[s]
 - where business component guid contains guid
 - where business component version matches version
 - where business component label contains label
- Section: As part of Event Process[s]
 - where event process guid contains guid
 - where event process version matches version
 - where event process label matches label
- Section: On peer server(s)
 - where peer server name contains name
 - where peer server label contains label
- Section: Rule Description (click on a blue link to edit it)

Allow
- Section: Name of the Rule

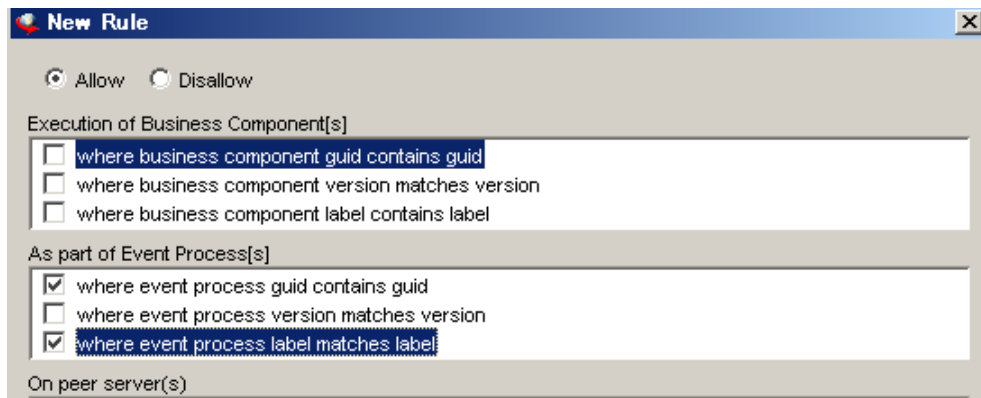
[Empty text box]
- Buttons: Ok, Cancel

まず、最初のルールである『 GUID “ABC”、ラベル “PRODUCTION” のコンポーネント フローを許可する 』を作成します。

許可するルールであるため、最上段にある [Allow] を選択します。

This close-up shows the top portion of the 'New Rule' dialog box, specifically the radio button selection area where Allow is selected and Disallow is unselected.

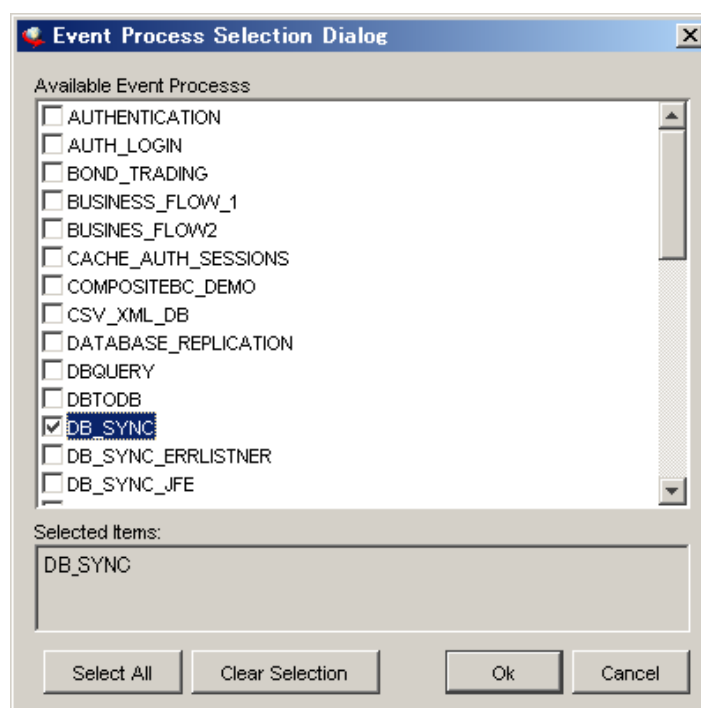
次に、コンポーネント フローの条件を指定します。条件は、GUID とラベルとなっていますので、中断にある [As part of Event Process(s) の `[where event process guid contains guid]` のチェックボックスをチェックします。続いて、`[where event process label matches label]` をチェックします。



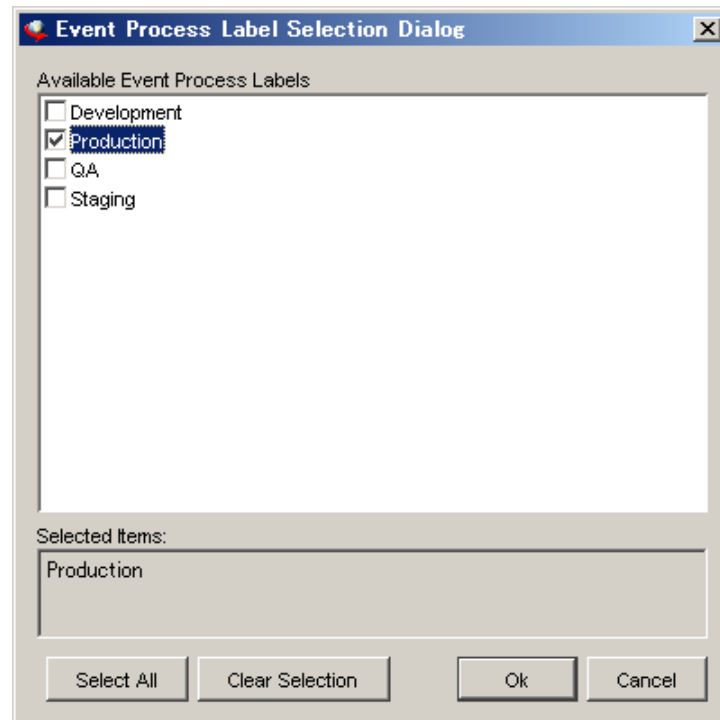
上の画面のようにチェックすると、下段にある Rule Description (Click on blue link to edit it) のペインが次のキャプチャ画面ように表示されます。



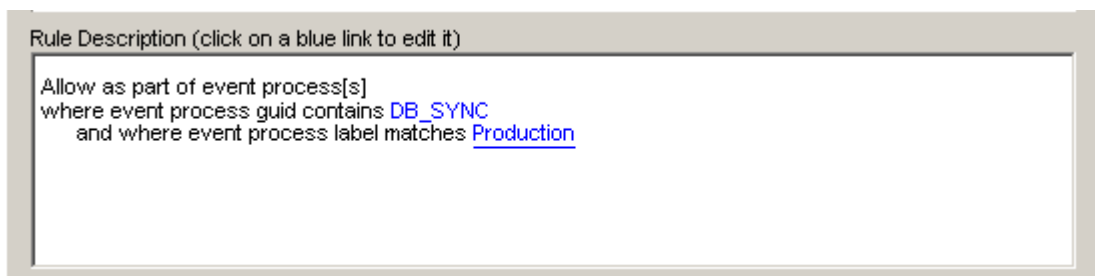
青字となっている guid をクリックします。次のようにコンポーネント フローの GUID を指定するダイアログ ボックスが現れます。ここでは、DB_Sync をチェックします (例題では、GUID を特定していませんでしたが、便宜上 DB_Sync を指定します)。[Ok] ボタンをクリックして確定します。



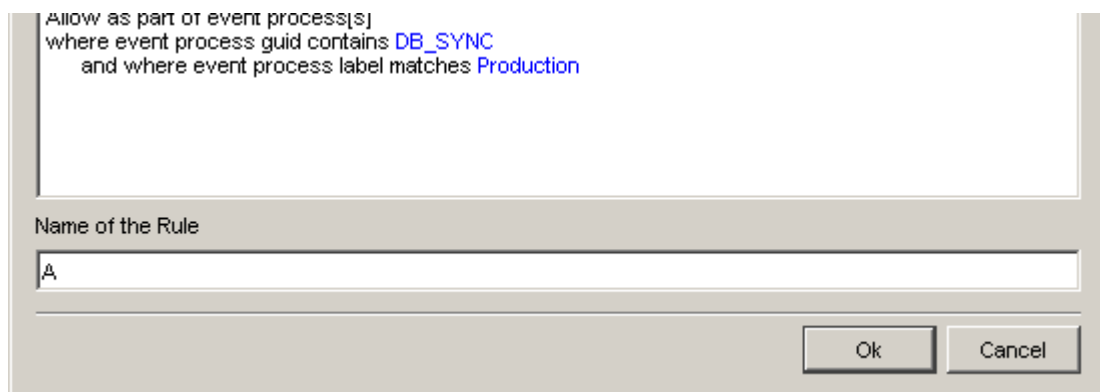
同様に、label をクリックして、ラベルを指定します。Production をチェックし、[ok] ボタンをクリックします。



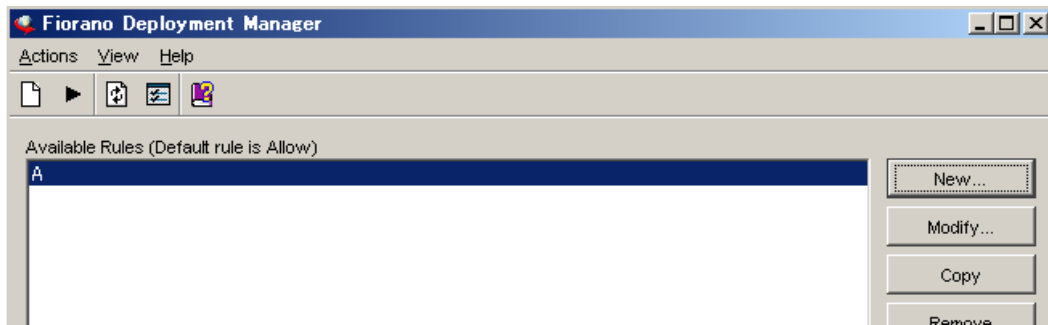
Rule Description (Click on blue link to edit it) のペインは、次の画面のように変更されます。



最後にルール名を指定します。最下段の Name of the Rule ペインに任意の名前を指定します。下の画面では、「A」としてありますが、ルール内容がわかる名前のほうがよいでしょう。



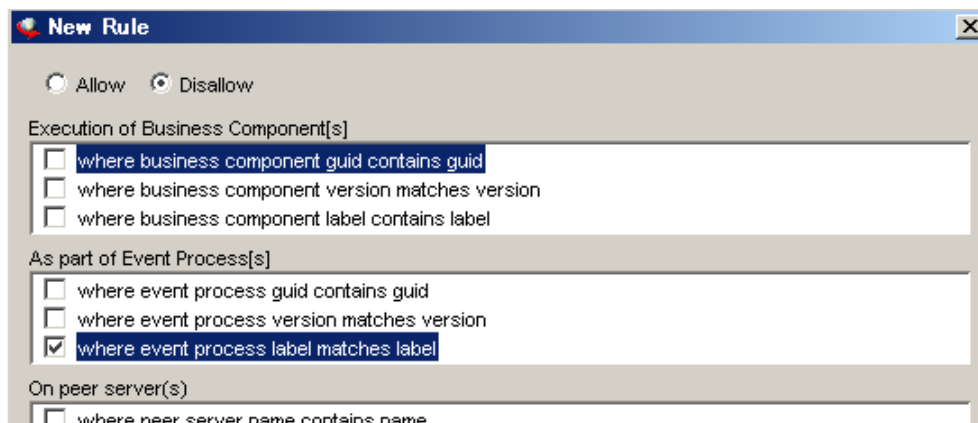
[ok] ボタンをいクリックすれば、ルールが作成されます。



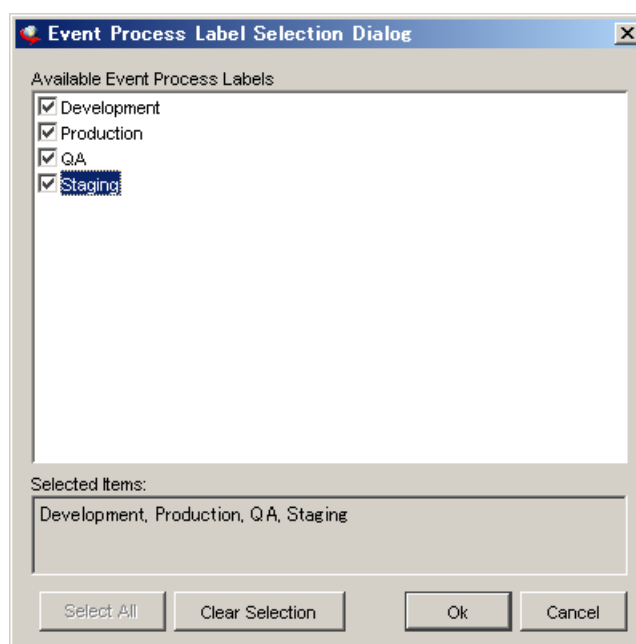
次に、『ラベル “DEVELOPMENT”、“PRODUCTION”、“QA”、“STAGING” のコンポーネント フローを許可しない』ルールを作成します。

[New] ボタンをクリックして、ルール作成ウィンドウを表示します。

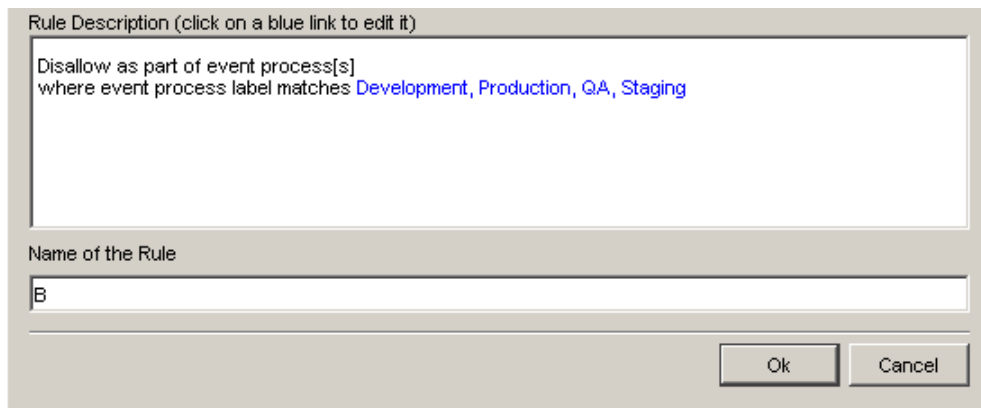
このルールでは、[Disallow] を選択し、[where event process label matched label] をチェックします。



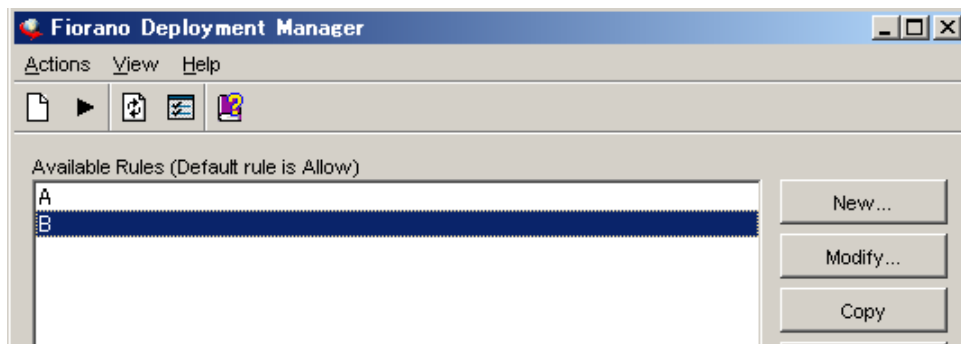
Rule Description (Click on blue link to edit it) のペインで、青字で表示されている label をクリックし、ラベル指定のダイアログ ボックスを表示します。すべてのラベルをチェックし、[Ok:] ボタンをクリックします。



このルールに名前を付け、[Ok] ボタンで確定します。



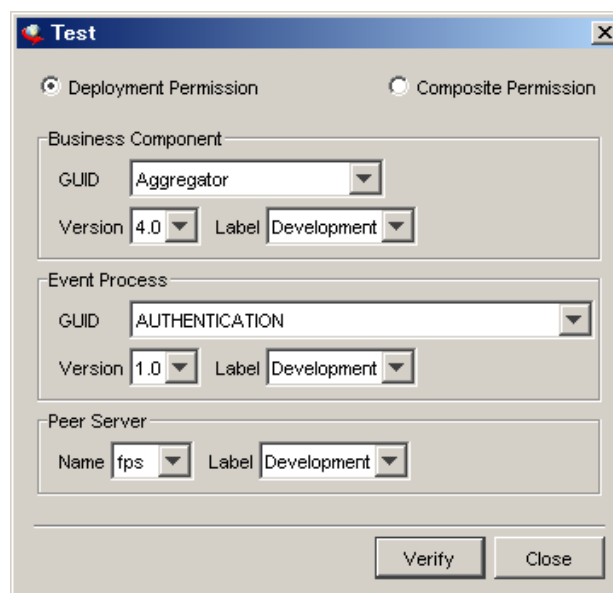
これで、2 つのルールが設定できました。



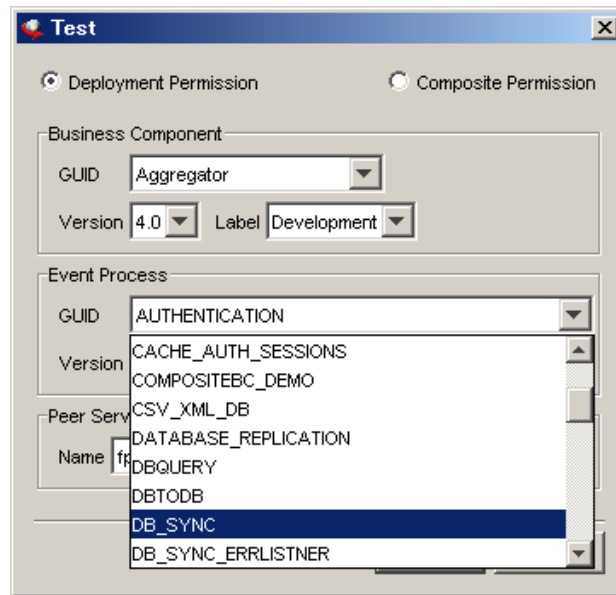
4.2.2 ルールのテスト

Deployment Manager には、作成したルールのテスト機能が備わっています。

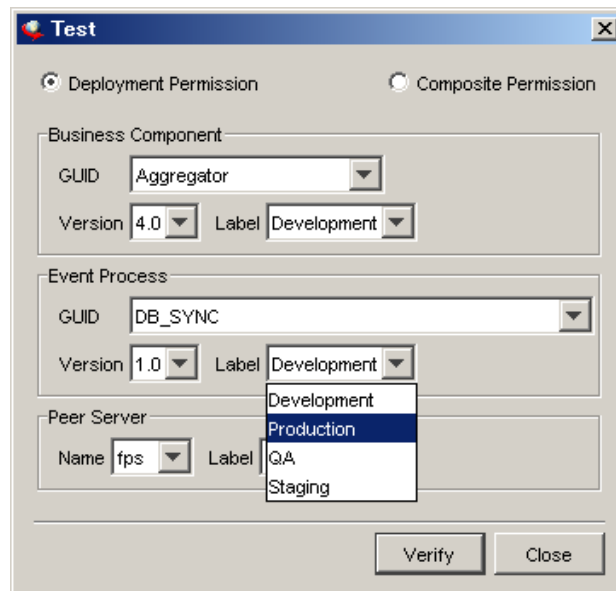
[Test ...] ボタンをクリックし、テスト条件を指定するダイアログ ボックスを表示します。



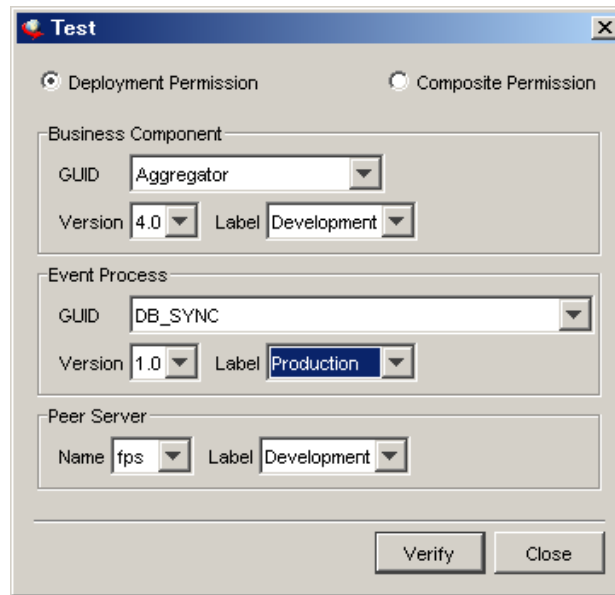
Event Process の欄で、GUID のプルダウンメニューから DB_Sync を選択します。



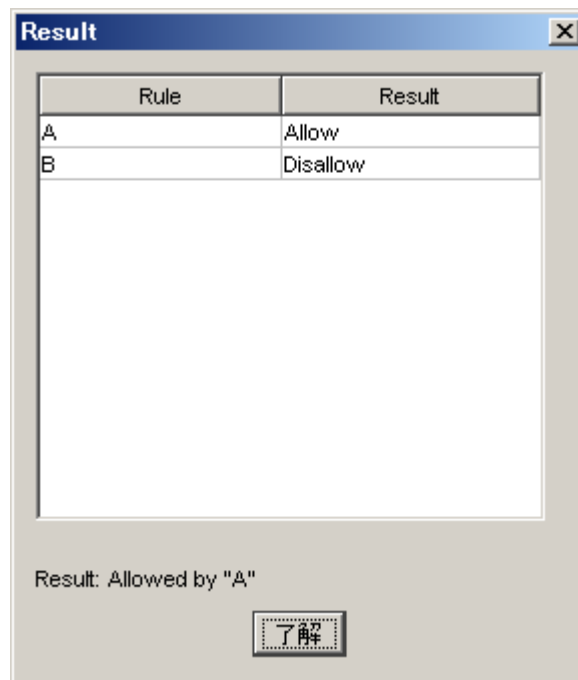
次に、Label のプルダウン メニューから Production を選択します。



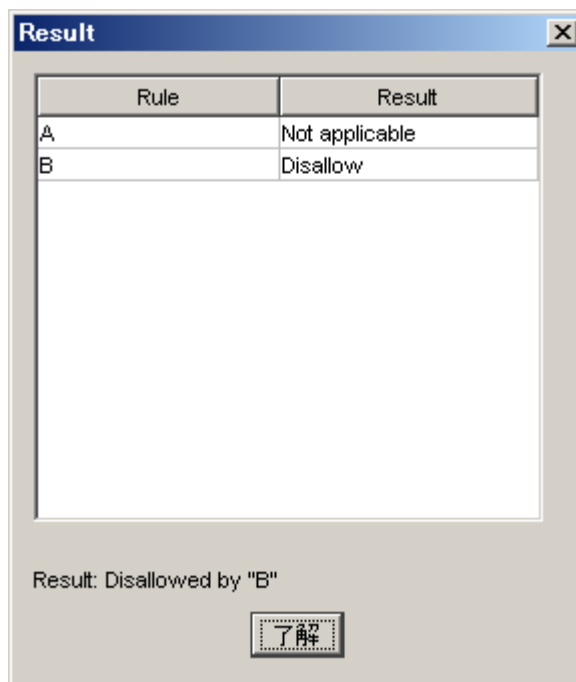
次の画面のように設定したら、[Verify] ボタンをクリックします。



次に示すようにテスト結果が表示されます。表示内容は、ルール A によって許可されたことを示しています。



次のテスト結果は、上記とは異なるコンポーネント フローを指定してテストをしたものです。ルール B によってデプロイメントが許可されなかったことが分かります。



次のエラー メッセージは、ラベルが Production となっていない DB_Sync を実行しようとした際に表示されるものです。



ラベルが Development となっているため、デプロイメントが許可されなかったことが分かります。