WebOTX V6 J2EEアプリケーションの
トラブルシューティング
(リソース参照、EJB参照)

改版履歴

2006 年 11 月　初版

目次

1 はじめに

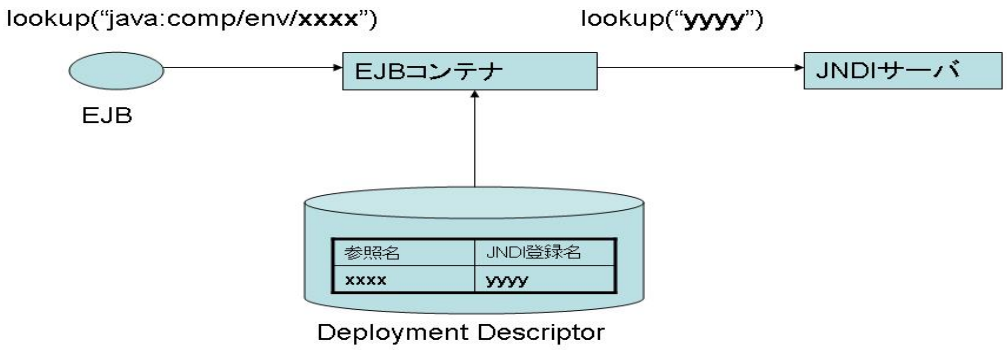
本資料では、WebOTX で J2EE アプリケーションを動作させるときに特に陥りやすい問題であるリソース参照、EJB 参照の動作概要と設定方法について説明しています。
なお本資料は WebOTX Ver6 を対象としています。Ver5 以前には適用できませんので注意してください。

2 リソース参照、EJB 参照について

J2EE アプリケーションの各コンポーネントは EJB、JDBC データソース、JavaMail などのさまざまなリソースを JNDI のインタフェース経由で取得します。J2EE 仕様ではこのようなリソースを参照するときに、"java:comp/env" という特殊なスキーマ名とコンテキスト名を持つ名前を参照する方法を定義しています。この "java:comp/env" が付いた名前で EJB を参照するものを EJB 参照、JDBC データソースや JCA コネクションファクトリ、JMS コネクションファクトリを参照するものをリソース参照といいます。

リソース参照, EJB 参照の特徴としては、JNDI の lookup メソッドに指定した名前と実際に JNDI サーバに登録される名前の関連付けを配備記述子(Deployment Descriptor)で指定するという点です。これにより、EJB やリソースの名前が変わった場合でも参照元のソースプログラムを変更する必要はなく、配備記述子だけの修正で対応することができます。

名前の関連付けはリソース参照、EJB 参照を使用した場合の JNDI に対するルックアップにコンテナが介在することで行われます。例えば EJB から "java:comp/env/..." の名前をルックアップすると、EJB コンテナがその名前を配備記述子で定義されている JNDI 登録名に置き換えて JNDI サーバに対するルックアップを実行します。



3 リソース参照、EJB 参照の設定に問題がある時のエラーと対処方法について

WebOTX 上で動作する J2EE アプリケーションから"java:comp/env/…" の名前をルックアップしたときに、配備記述子に対応するリソース参照、EJB 参照の定義が存在しないと以下のような例外が発生します。

javax.naming.NameNotFoundException: No object bound to name java:comp/env/...

また、リソース参照、EJB 参照が定義されていても配備記述子で関連付けされた名前が存在しない場合や、J2EE アプリケーションでないアプリケーションから "java:comp/env/…" の名前をルックアップした場合も同じ例外が発生します。
このときの対処は単純に JNDI サーバに問い合わせる名前が間違っているケースとは異なります。WebOTX 配備ツールを使用するかまたは配備記述子の XML ファイルを直接編集することによってリソース参照、EJB 参照の設定を行う必要があります。

4 設定方法

EJB や Web アプリケーションでリソース参照、EJB 参照を使用するための設定方法について説明します。WebOTX 配備ツールを使用する方法と、配備記述子のファイルを直接編集

する方法があります。

4.1　リソース参照

以下アプリケーションで “java:comp/env/MyDataSource” というリソース参照を用いて JNDI サーバに “jdbc/Oracle” という名前で登録されている JDBC データソースをルックアップしているものとします。

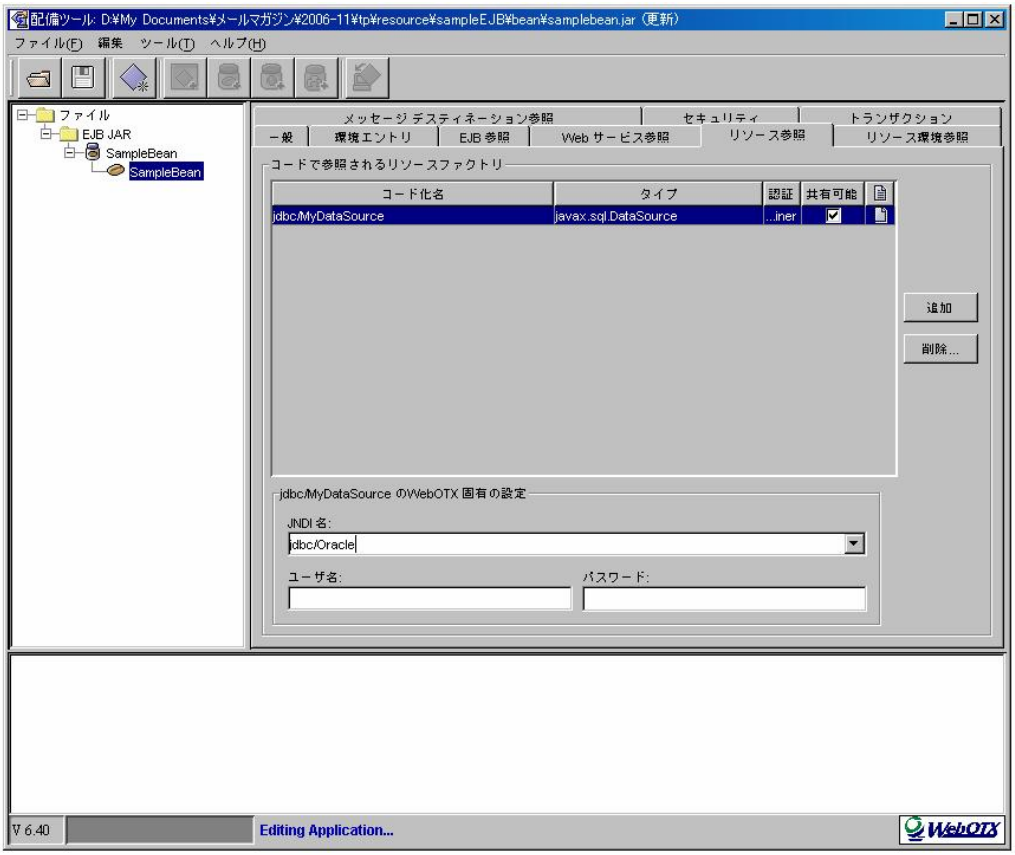
```
import javax.naming.InitialContext;
import java.sql.Connection;
import javax.sql.DataSource;
. . . .
InitialContext initCtx = new InitialContext();
DataSource ds =
(DataSource)initCtx.lookup(“java:comp/env/jdbc/MyDataSource”);
Connection conn = ds.getConnection();
```

ソースイメージ

4.1.1 WebOTX 配備ツールを使用する場合

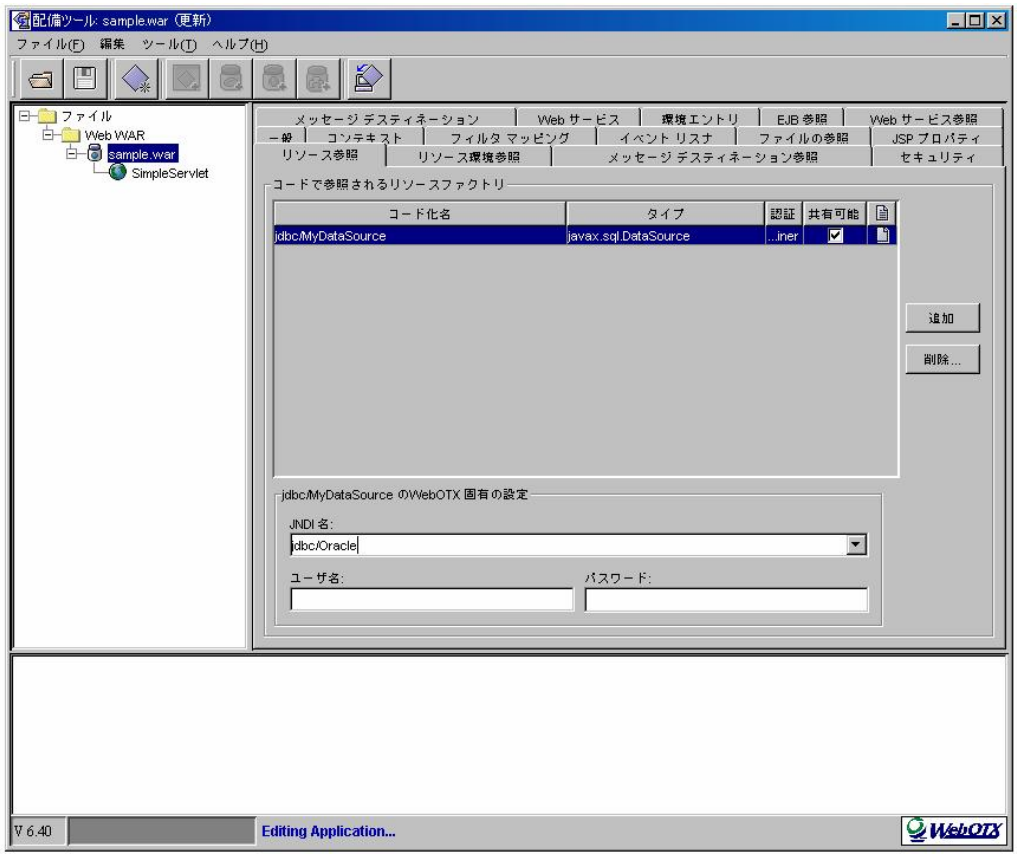
配備記述子は WebOTX 配備ツールの GUI 画面で編集することができます。このためリソース参照も GUI 画面から編集することができます。

EJB の場合、下の画面のように EJB のノードを選択して右側画面の「リソース参照」タブを選択するとリソース参照の設定画面となります。



- 「コード化名」には アプリケーションからルックアップしている名前の “java:comp/env/” より後ろの文字列を指定します。この例では”jdbc/MyDataSource” となります。
- 「タイプ」は参照するリソースのタイプを指定します。この例では java.sql.DataSource を選択します。
- 「認証」は Container か Applicetion を選択します。リソースに接続するときのユーザ名、パスワードの指定をアプリケーションから行う場合（getConnection メソッドをユーザ名、パスワードを指定して呼び出す場合）は Application を選択します。コンテナに任せる場合は Container を選択します。この例では getConnection でユーザ名、パスワードを指定していないため、Container を選択します。
- 「共有可能」は取得したコネクションを他のアプリケーションと共有するかどうかを指定します。通常は共有ありにします。
- 「WebOTX 固有の設定」の「JNDI 登録名」は JNDI サーバに登録される名前を指定します。この例では “jdbc/Oracle” となります。
- 「WebOTX 固有の設定」の「ユーザ名」、「パスワード」の設定は、認証が Container の場合に接続ユーザ名、パスワードを指定することができます。指定しない場合は参照するデータソースで設定されている値が使用されます。

war アプリケーションの場合も同様となります。以下の画面となります。



4.1.2 配備記述子を直接編集する場合

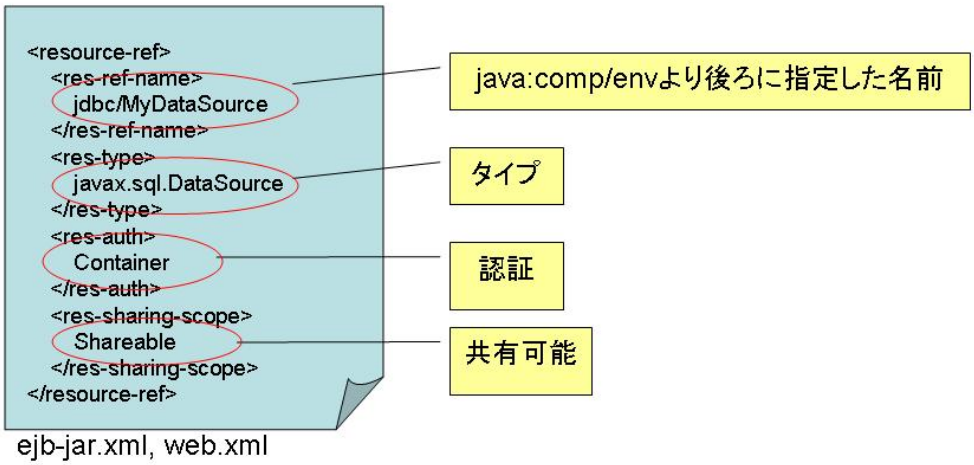
配備記述子を直接編集する場合、編集するファイルはアプリケーションの種類によって異なり、以下のようになっています。

アプリケーションの種類	関連するファイル
EJB	META-INF/ejb-jar.xml META-INF/nec-ejb-jar.xml
Web アプリケーション	WEB-INF/web.xml WEB-INF/nec-web.xml

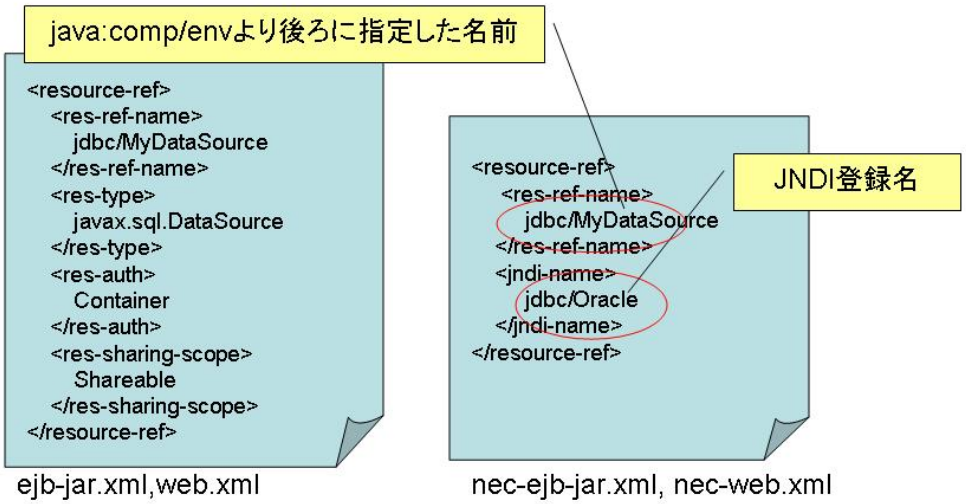
ejb-jar.xml,と web-xml は EJB, サーブレットの仕様で定義されているファイルです。
nec-ejb-jar.xml と nec-web.xml は WebOTX 独自のファイルです。
配備ツールで編集したときも結果はこのファイルに反映されますので、最初にリソース参照、EJB 参照の定義を作成するときは配備ツールを使用し、以後の修正は ejb-jar, war から XML ファイルを取り出して編集、再アーカイブすることで行うこともできます。

設定の仕方としては、まず ejb-jar.xml または web.xml で以下の要素を追加してリソース参照の定義を行ないます。

- resource-ref
  リソース参照を定義します。resource-ref-name, resource-type, resource-type, res-sharing-scope 要素を子供として含みます。
- resource-ref-name
  配備ツールの「コード化名」の設定に対応します。
- resource-type
  配備ツールの「タイプ」の設定に対応します。
- resource-type
  配備ツールの「認証」の設定に対応します。Container または Application を指定します。
- res-sharing-scope
  配備ツールの「共有可能」の設定に対応します。



次に nec-ejb-jar.xml または nec-web.xml に同じ res-ref-name を持つ resource-ref 要素を定義し、jndi-name 要素で JNDI 登録名を指定します。今回の例の場合、”jdbc/Oracle”を jndi-name に指定します。



4.2 EJB 参照（リモート）

EJB 参照も基本的にはリソース参照と同様ですが、EJB にはリモート、ローカルの二種類ありそれぞれ設定方法が少し異なります。まずリモートの場合について説明します。以下アプリケーションで “java:comp/env/MyEJB” というリソース参照を用いて ”SampleBean” という名前で JNDI サーバに登録されている EJB をルックアップしているものとします。

```
import javax.naming.InitialContext;
import javax.rmi.PortableRemoteObject;
. . . .
InitialContext initCtx = new InitialContext();
Object obj = initCtx.lookup("java:comp/env/ejb/MyEjb");
EjbRefSampleHome home =   (EjbRefSampleHome)
          PortableRemoteObject.narrow(obj,EjbRefSampleHome.class);
EjbRefSample remote = home.create();
```

ソースイメージ

4.2.1 WebOTX 配備ツールを使用する場合

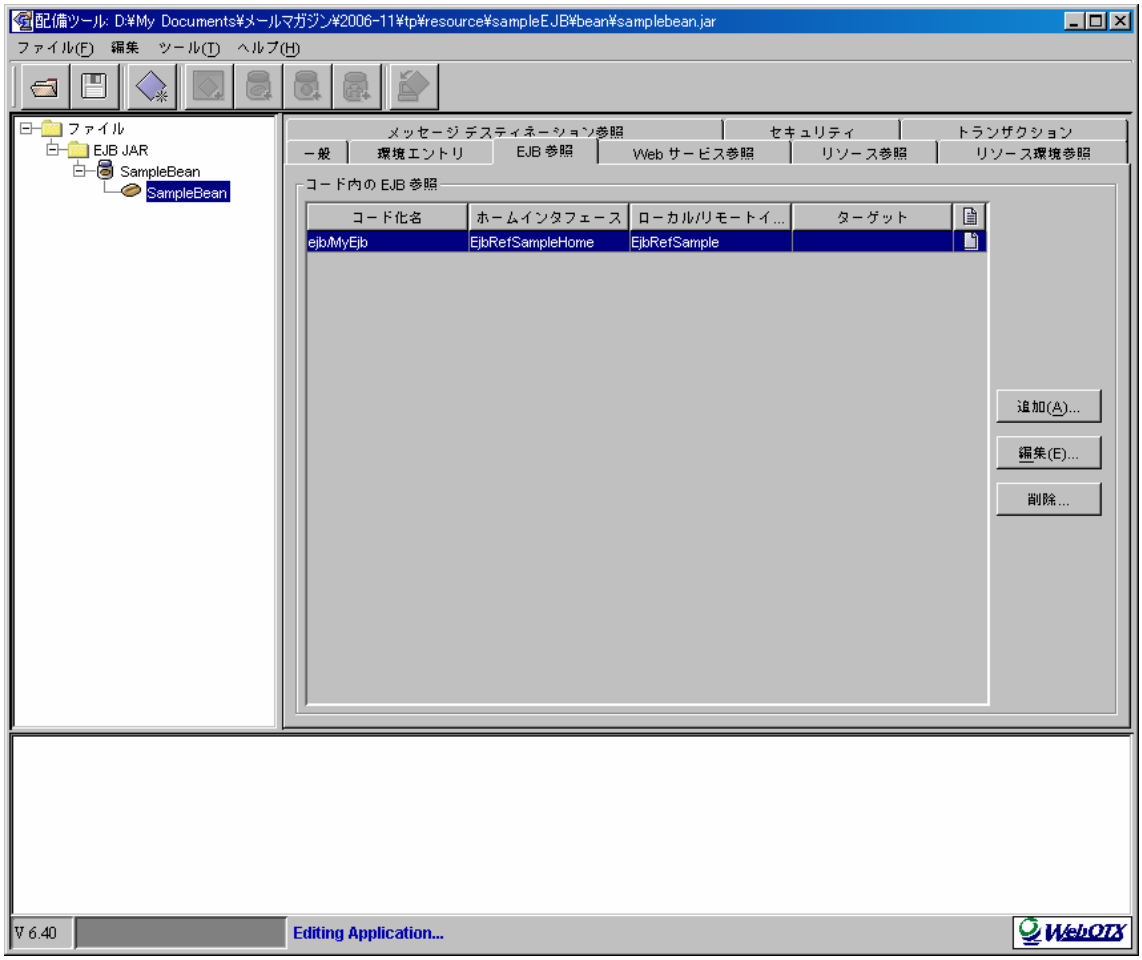
EJB の場合、下の画面のように EJB のノードを選択して右側画面の「EJB 参照」タブを選択するとリソース参照の設定画面となります。ここで「追加」ボタンを押すと以下のダイアログが現れ、これに入力すると EJB 参照が追加されます。

- 「コード化名」には アプリケーションからルックアップしている名前の “java:comp/env/” より後ろの文字列を指定します。この例では”ejb/MyEjb”となりま

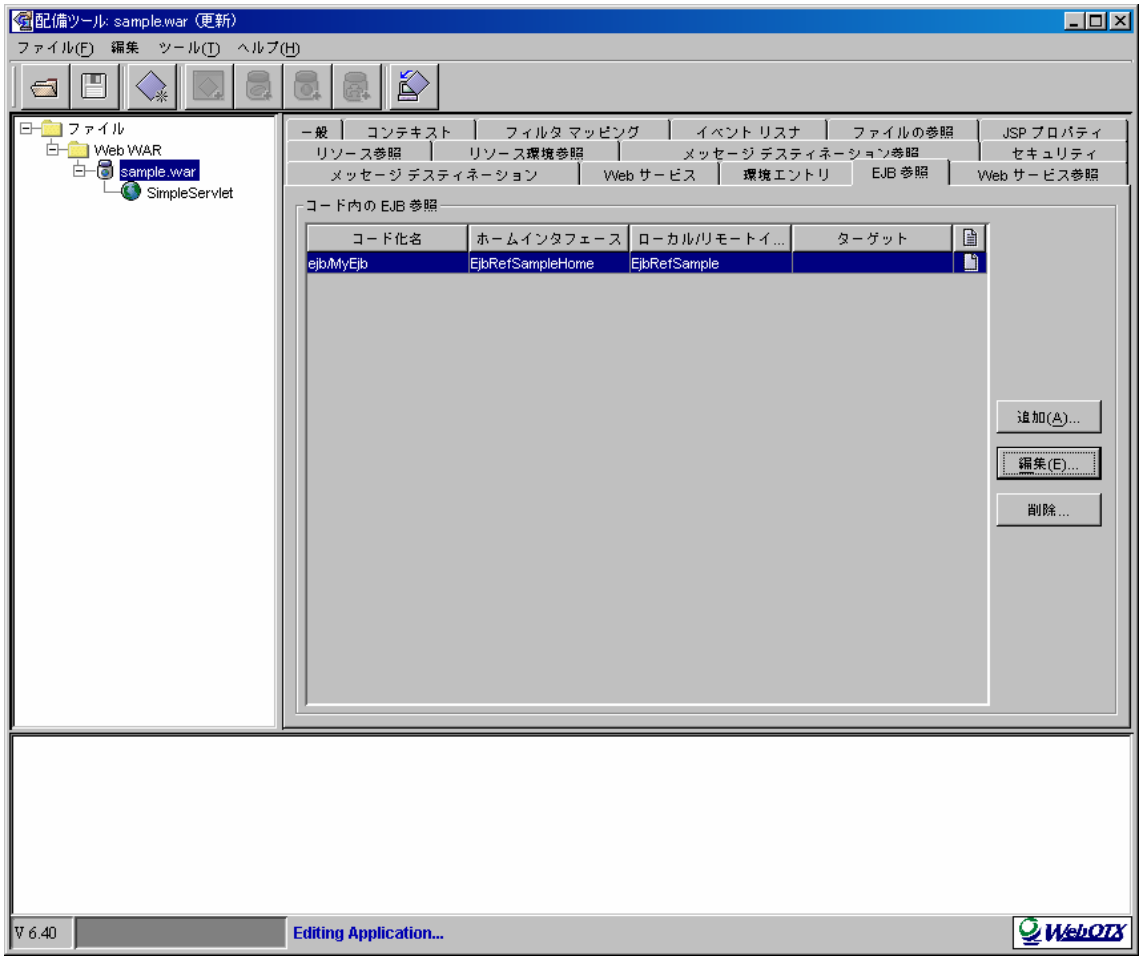
す。

- 「EJB タイプ」には参照する EJB の種類を Sesstion, Entity から選択します。
- 「インタフェース」には参照する EJB がリモートであるかローカルであるかを Remote, Local から選択して指定します。
- 「ホームインタフェース」には参照する EJB のホームインタフェースクラス名を指定します。
- 「ローカル/リモートインタフェース」には参照する EJB のリモートインタフェースのクラス名を指定します。
- 「ターゲット EJB」には参照する EJB を識別するための名前を指定します。リモートの場合 JNDI 名を指定します。この例では “SampleBean” となります。

ok ボタンを押すと、EJB 参照の定義が追加されます。



war アプリケーションの場合も同様となります。以下は追加後の画面となります。

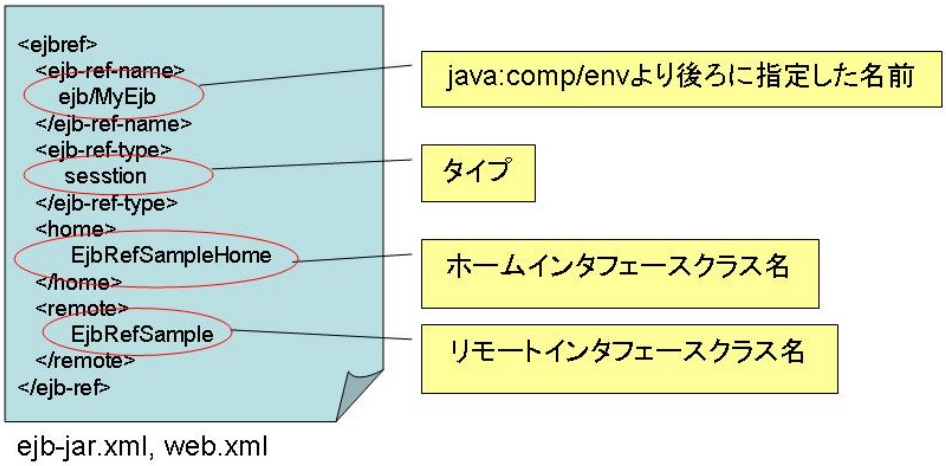


4.2.2 配備記述子を直接編集する場合

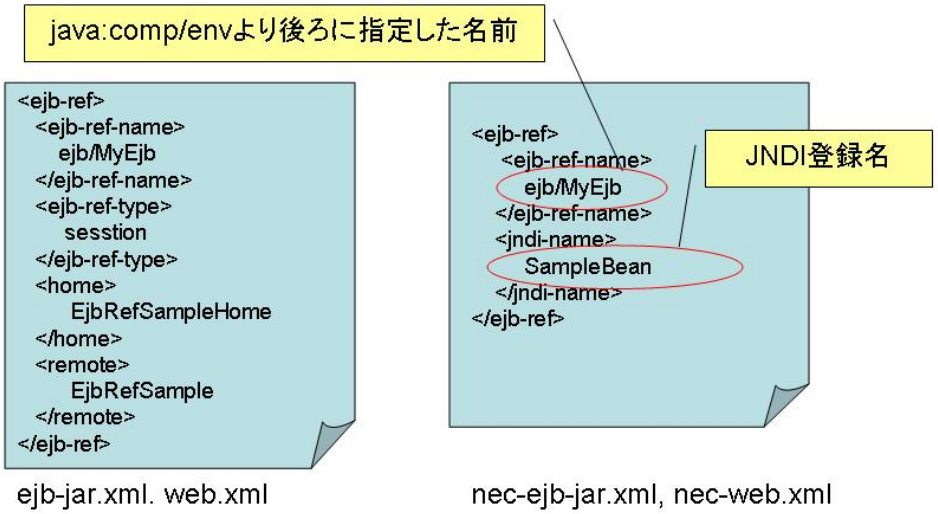
修正するファイルはリソース参照の場合と同じです。ejb-jar.xml または web.xml に追加する要素は以下のとおりです。

- ejb-ref
  EJB 参照を定義します。ejb-ref-name, ejb-ref-type,　home, remote, ejb-link 要素を子供として含みます。(ejb-link はここでは使用しません)
- ejb-ref-name
  配備ツールの「コード化名」の設定に対応します。
- ejb-ref-type
  配備ツールの「EJB タイプ」の設定に対応します。
- home
  配備ツールの「ホームインタフェース」の設定に対応します。
- remote
  配備ツールの「ローカル/リモートインタフェース」の設定に対応します。

ejb-ref 要素はリモート EJB 専用であるため、ローカル・リモートを指定する要素はありません。



次に nec-ejb-jar.xml または nec-web.xml にはリソース参照の場合と同じく、同じ ejb-ref-name を持つ ejb-ref を定義し、その中の jndi-name 要素で JNDI 登録名を指定します。今回の例では “SampleBean” を jnd-name に指定します。



4.3 EJB 参照（ローカル）

ローカル EJB の場合について説明します。以下アプリケーションで “java:comp/env/MyEJB” というリソース参照を用いて同じ ear ファイルの samplebean.jar に含まれる ”SampleLocalBean” という名前(ejb-jar.xml の ejb-name で定義している名前)の EJB をルックアップしているものとします。

```
import javax.naming.InitialContext;
. . . .
InitialContext initCtx = new InitialContext();
EjbRefSampleLocalHome home =
        (EjbRefSampleLocalHome)initCtx.lookup("java:comp/env/ejb/MyEjb");
EjbRefSampleLocal local = home.create();
```

ソースイメージ

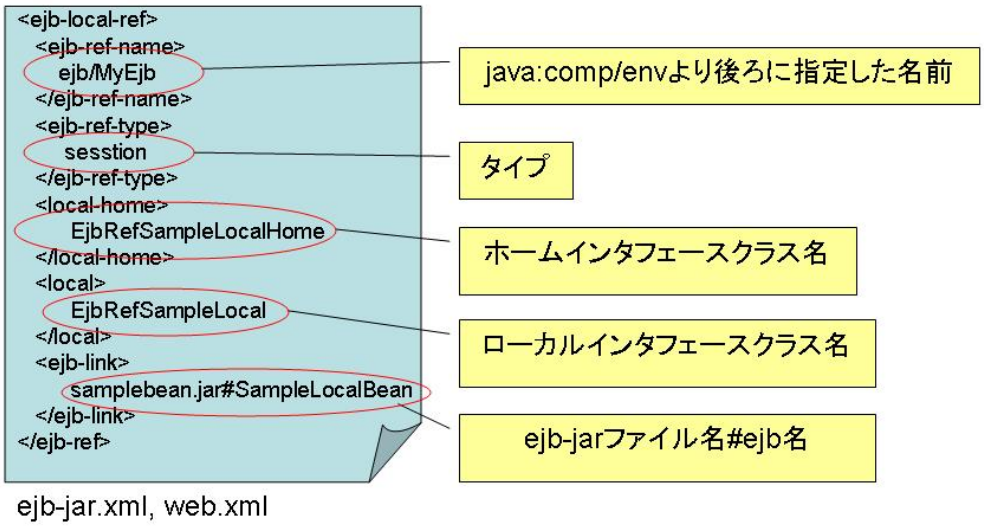
4.3.1 WebOTX 配備ツールを使用する場合

EJB 参照の追加ダイアログで、インタフェースは ”Local” を選択し、ターゲット EJB では「Enterprise Bean 名」を選択して “ejb-jar ファイル名#ejb 名” を選択します。

4.3.2 配備記述子を直接編集する場合

ejb-jar.xml または web.xml に追加する要素が”ejb-ref”でなく、”ejb-local-ref”になります。

- ejb-local-ref
  EJB 参照を定義します。ejb-ref-name, ejb-ref-type,　local-home, local, ejb-link 要素を子供として含みます。
- ejb-ref-name
  配備ツールの「コード化名」の設定に対応します。
- ejb-ref-type
  配備ツールの「EJB タイプ」の設定に対応します。
- local-home
  配備ツールの「ホームインタフェース」の設定に対応します。
- local
  配備ツールの「ローカル/リモートインタフェース」の設定に対応します。
- ejb-link
  EJB の名前を“ejb-jar ファイル名#ejb 名”で指定します。



nec-ejb-jar.xml または nec-web.xm には要素を追加する必要はありません。