Visual COBOL チュートリアル

JCA による JBOSS EAP 連携の設定と開発

1. 目的

Visual COBOL に付属する COBOL 専用のアプリケーションサーバー「Enterprise Server」は、ネイティブにコンパイルした COBOL のビジネスロジックを EJB として再利用し、Java EE ライアントから呼び出す機能を提供しています。EJB コンポーネントとして呼び出しを行う場合、Java アプリケーションサーバー上の Java EE クライアントは JCA の仕様にもとづいたリクエストを Enterprise Server に渡し、COBOL のビジネスロジックが処理をして結果を返します。また、Enterprise Server は、XA に準拠しているので同じく XA に準拠しているデータベースや他のシステムと協調してトランザクション処理を行うことができます。

このドキュメントでは Windows Server 上の JBOSS EAP 7.4 と Enterprise Server を JCA による連携を行い、Eclipse で開発する方法を説明します。

2. 前提条件

本チュートリアルは、下記の環境を前提に作成されています。サポートしているプラットフォームであれば他の Linux/UNIX でも利用可能です。

- アプリケーションサーバー側 ソフトウェア

OS	Windows Server 2022 Standard Edition（64bit）
AP サーバー製品	Red Hat JBoss EAP 7.4.6

- 開発クライアント ソフトウェア

OS	Windows Server 2022 Standard Edition（64bit）
COBOL 開発環境製品	Visual COBOL 9.0 for Eclipse

- チュートリアル用サンプルプログラム

  下記のリンクから事前にチュートリアル用のサンプルファイルをダウンロードして、任意のフォルダに解凍しておいてください。

  サンプルプログラムのダウンロード

内容

3. チュートリアル手順の概要

3.1. Visual COBOL for Eclipse を起動してソースコードのインポート作業を行う

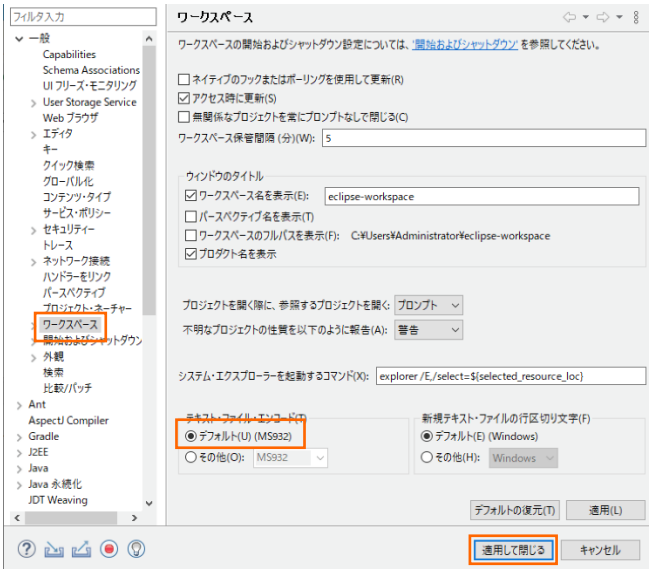
1) Visual COBOL for Eclipse を起動

① [スタート] メニュー > [Micro Focus Visual COBOL] > [Visual COBOL for Eclipse]を選択します。

② ワークスペースの選択画面にて 任意の作業フォルダを指定します。ここでは"C:¥work¥JCA" を指定します。

2) 文字コードの指定を行います。

① Visual COBOL 9.0 より Shift-JIS を指定して日本語を表示する場合、文字コードの指定を明確に行う必要があります。最初に、[Window(W)]メニュー > [設定(P)]より[一般] > [ワークスペース]とナビゲートし、テキストファイルエンコードを「MS932」に変更し、[適用して閉じる]ボタンをクリックします。

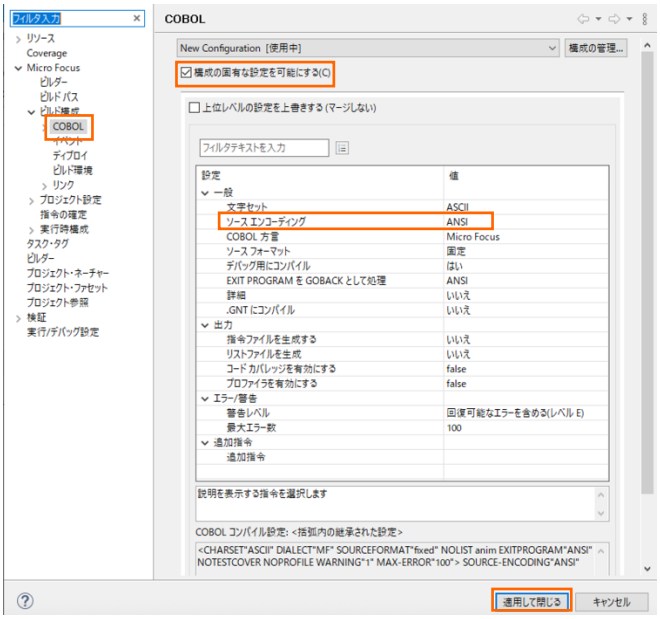


※Preference Recorder のダイアログが表示されたら [キャンセル] を選択してください。

3) ネイティブ COBOL プロジェクトの作成

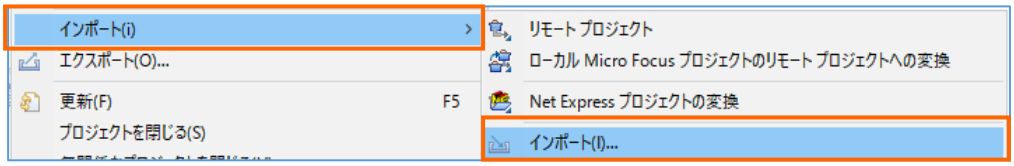
① [ファイル(F)]メニュー > [新規(N)] > [COBOL プロジェクト]を選択し、プロジェクト名に"NativeCOBOL"を入力し、プロジェクトテンプレートは「64 ビット」を指定して、[終了(F)] ボタンを押します。

② 次に作成した COBOL プロジェクトを選択した状態で、マウスの右クリックにてコンテクストメニューを表示し、「プロパティ」を選択します。[Micro Focus] > [ビルド構成] > [COBOL]とナビゲートし、「構成の固有な設定を可能にする(C)」にチェックを入れて[一般] > [ソース エンコーディング]を"UTF-8"から"ANSI"に変更し、[適用して閉じる]ボタンをクリックします。

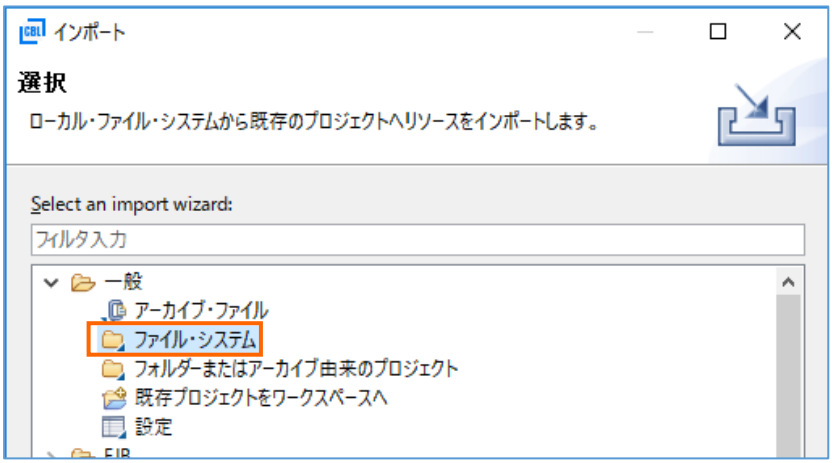


4) プログラムソースのインポート

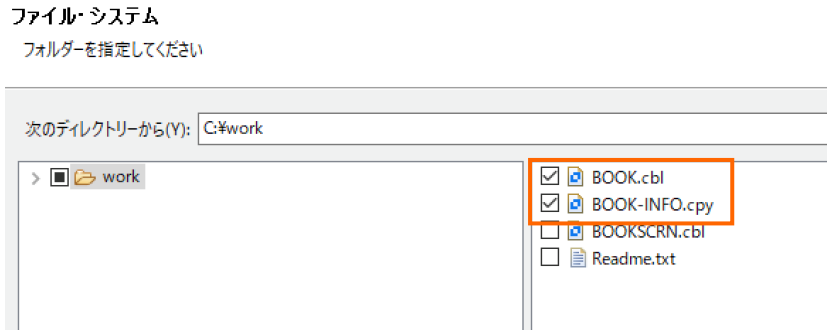
① COBOL エクスプローラーのパースペクティブを開き、COBOL エクスプローラービューにて プロジェクトフォルダを右クリックし、コンテクストメニューから [インポート(I)] > [インポート(I)] を選択します。



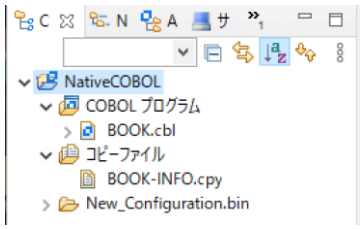
② 既存のソースコードをロードします。一般のフォルダを展開し、[ファイル・システム] を選択し、[次へ(N)] ボタンを押します。



③ インポートダイアログが表示されるので[参照(R)] ボタンを押してソースコードを解凍したフォルダを指定します。ここでは“C:¥Work”を指定しています。そこにある「BOOK.cbl」とコピーブック「BOOK-INFO.cpy」を指定し、[終了(F)] ボタンを押します。



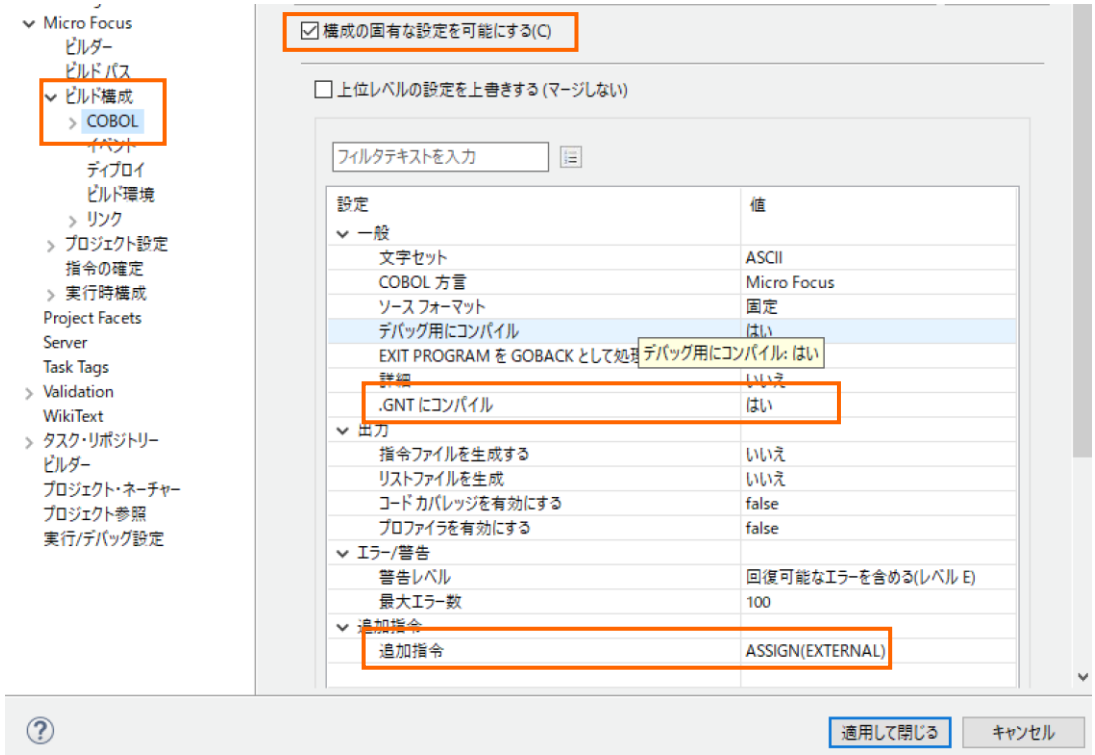
④ プロジェクトフォルダを展開し、２つのファイルが正常にロードされていることを確認します



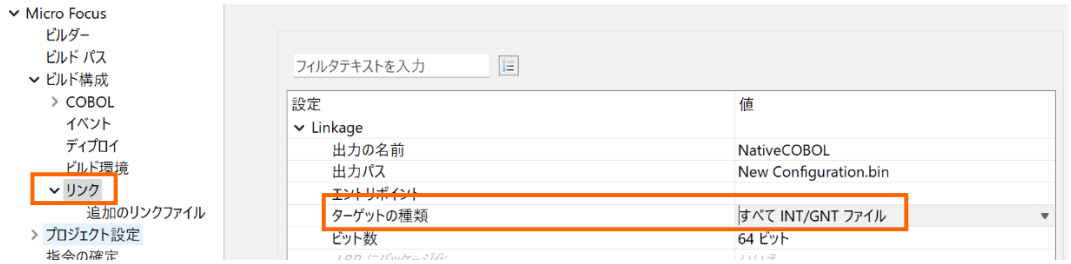
5) ビルドオプションの変更

① COBOL エクスプローラーにて作成したプロジェクトを右クリックし、[プロパティ(R)] を選択します。

② プロパティ設定ダイアログが表示されます。[Micro Focus] > [ビルド構成] > [COBOL] を選択します。次に [構成の固有な設定を可能にする(C)] にチェックを入れて、[.GNT にコンパイル] を“いいえ”から“はい”に変更します。次に[追加指令]の値フィールドに”ASSIGN(EXTERNAL)”を指定します。

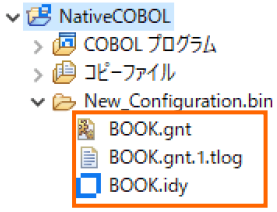


③ 次に [Micro Focus] > [ビルド構成] > [リンク] を展開します。[ターゲットの種類] を「すべて INT/GNT ファイル」に変更します。



④ 全ての設定が終了したら［適用して閉じる］ボタンを押します。

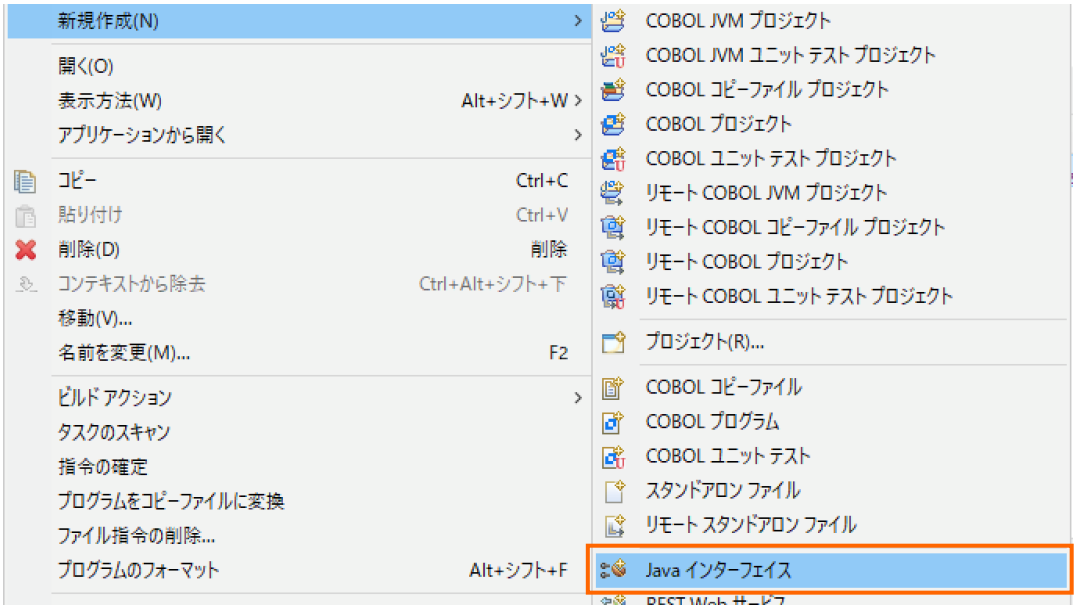
⑤ COBOL エクスプローラーにて"New_Configuration.bin"を展開して下記のファイルが作成されていることを確認します。



3.2. JCA 用のアプリケーションプロファイルを作成する

1) EJB/JCA アプリケーションのプロファイルを作成

① EJB/JCA アプリケーションとして利用する「BOOK.cbl」を右クリックし、コンテクストメニューから［新規作成(N)］>［Java インターフェイス］を選択します。



② ［Java インターフェイス名］欄には "bookEJB" を入力し、［マッピング］欄は「無し」を選択し、マッピングするプログラムは「BOOK.cbl」が選択されていることを確認し［終了］ボタンを押します。

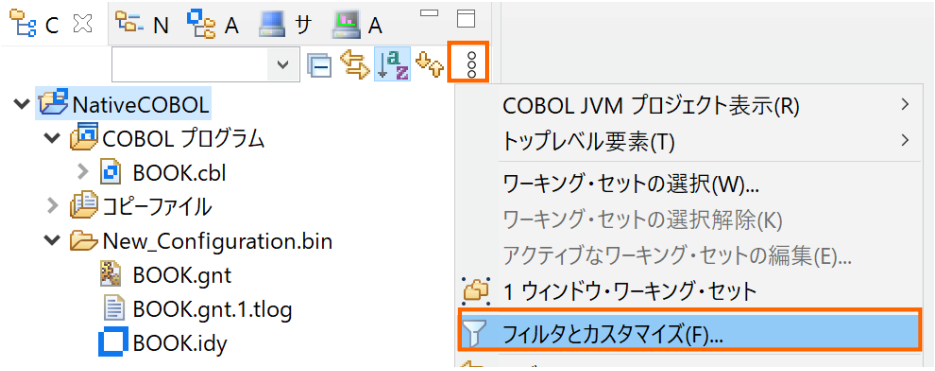
3.3. ディプロイ先の変更を行う

1) COBOL エクスプローラー表示設定の変更

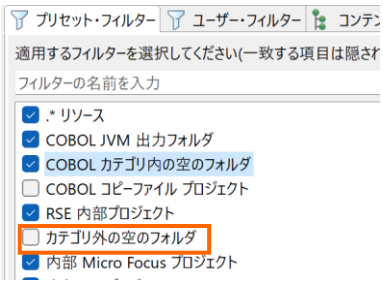
① COBOL エクスプローラーに戻ります。

② COBOL エクスプローラー右上の「↓↑」アイコンの右横にあるアイコンをクリックし、［フィルタとカスタマイズ(F)］を選択しま

す。



③ [カテゴリ外の空のフォルダ] にチェックされている場合は、チェックを外したのち、[OK] ボタンをクリックします。



2) ディプロイ用のフォルダの作成

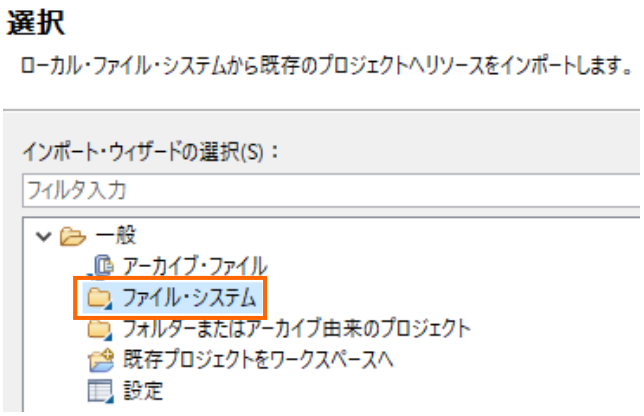
① 「NativeCOBOL」プロジェクト上で右クリックし、コンテクストメニューから [新規作成(N)]→[フォルダー] を選択します。

② 「NativeCOBOL」プロジェクトが選択されていることを確認の上、フォルダ名に ”deploy” を指定し、[終了(F)] ボタンをクリックします。

3) 「.mfdeploy」ファイルのインポート

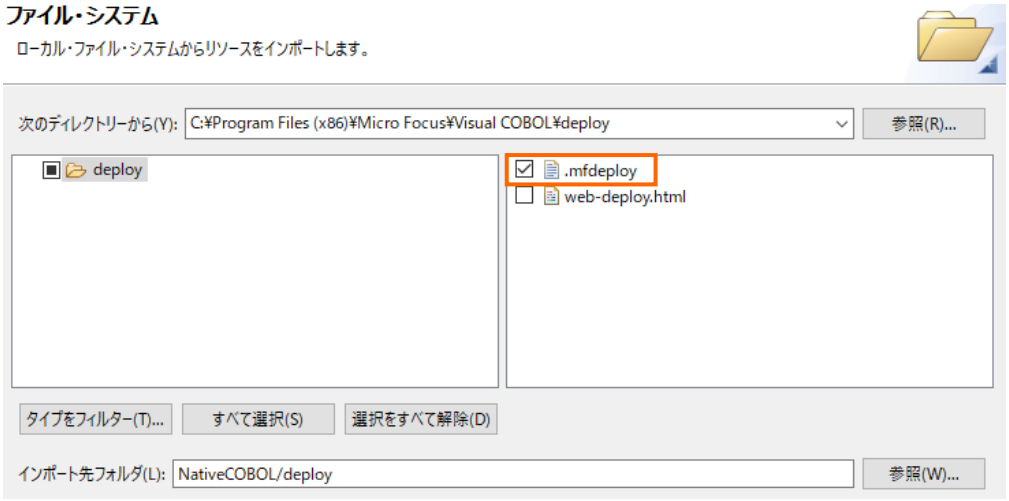
① 作成した「deploy」フォルダ上で右クリックし、コンテクストメニューから [インポート]→[インポート] を選択します。

② 「一般」配下の「ファイル・システム」を選択し、[次へ(N)] ボタンをクリックします。



③ [参照(R)] ボタンをクリックし、Visual COBOL インストールフォルダ¥deploy を選択します。

④ [.mfdeploy] ファイルにチェックを行い、[終了(F)] ボタンをクリックします。



補足）

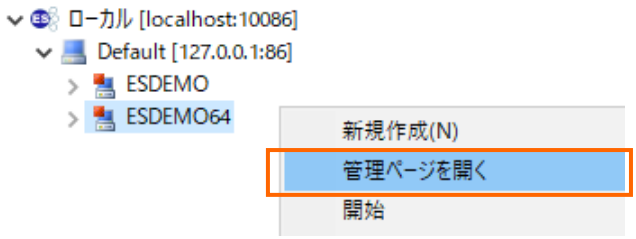
上記手順完了後も COBOL エクスプローラー上の NativeCOBOL¥deploy フォルダ配下に .mfdeploy ファイルは表示されませんが、これはフィルタによるものです。

正しくインポートされたことを確認する場合は、さきほど同様、[フィルタとカスタマイズ] を選択し [.* リソース] のチェックを外すことで表示されるようになります。しかし、リソースに関する設定情報が表示されるようになるため、通常はチェックを行い、非表示状態とすることを推奨します。

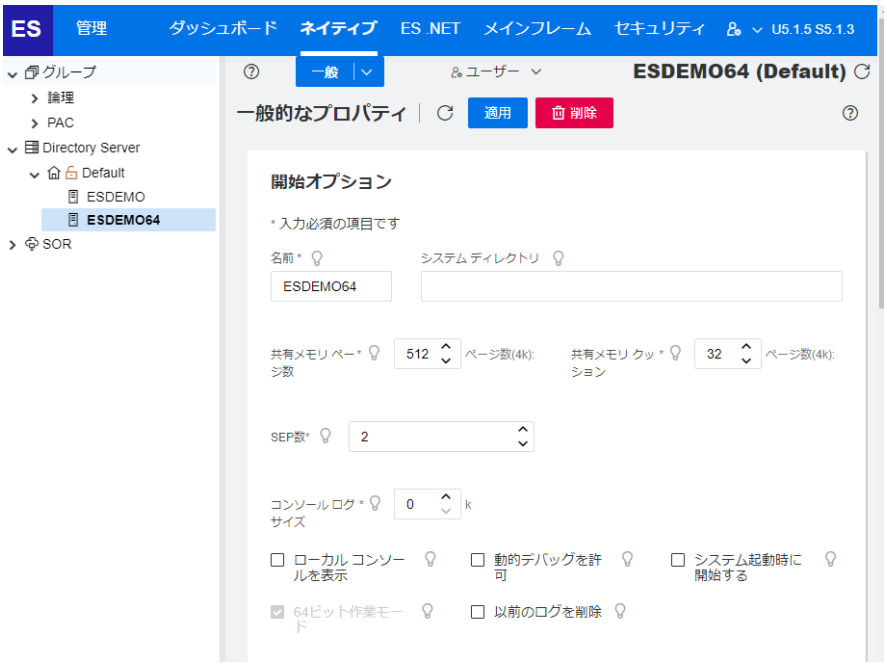
4) ディプロイ先の変更設定

① [サーバーエクスプローラー] タブを選択します。

② [ローカル localhost:86] 上で右クリックし、コンテクストメニューから [管理ページを開く] を選択します。

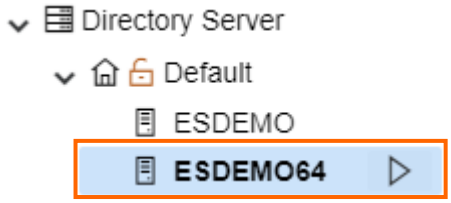


③ ESCWA (Enterprise Server Common Web Administration) 画面にてダッシュボードが表示されます。

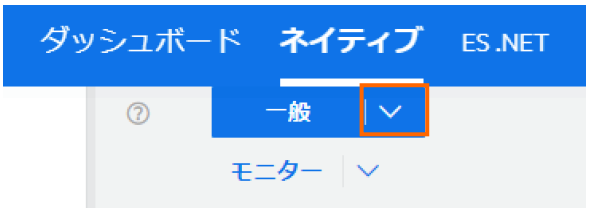


④ [ネイティブ]タブメニューをクリックします。

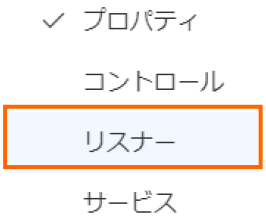
⑤ 次に左側メニューの[Directory Server]をドリルダウンして、[Default] → [ESDEMO64]をクリックします。



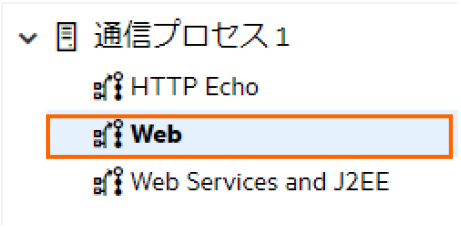
⑥ [一般]メニューが表示されるので横にある下向き記号をクリックします。



⑦ ドロップダウンメニューから[リスナー] をクリックします。



⑧ [通信プロセス１]の下に見えている[Web]リスナーをクリックします。



⑨ リスナープロパティが表示されます。横のスライドバーを下に下げていくとのカスタム構成情報が表示されています。デフォルトは「uploads=<ES>/deploy」となっています。この場合、Visual COBOL インストールディレクトリ配下の deploy フォルダがディプロイ用フォルダとして使用されます。通常、Program Files (x86)等のフォルダは管理者権限を持つユーザーでないと書き込みできないので変更を行います。

下記の例のように設定を変更し、[適用] ボタンをクリックします。

例：uploads=C:/work/JCA/NativeCOBOL/deploy

⑩ 正常に更新された旨のポップアップが表示されます。

※Enterprise Server を自分で作成した場合、[Web リスナー] を「Disable」から「Started」に変更してください。

⑪ 次に [Web Services and J2EE] リスナーをクリックします。

⑫ ポート番号を"*"から"9003"に変更し、スライドバーを上に移動して、[適用] ボタンをクリックします。

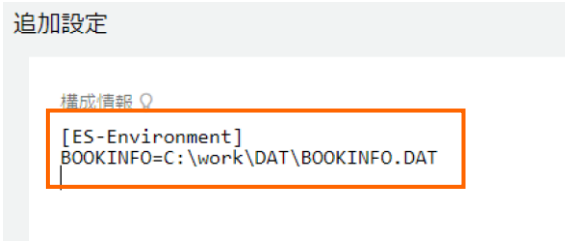
3.4. データファイルのアクセス指定を行う

1) Enterprise Server がアクセスするデータファイルを指定

① ESCWA 画面の[一般]をクリックし、[追加設定]フィールドに下記の命令を追加します。

[ES-Environment]

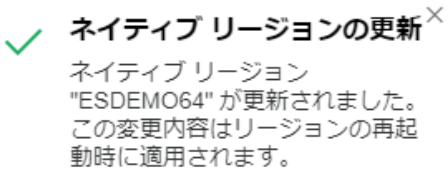
BOOKINFO=C:¥work¥DAT¥BOOKINFO.DAT



このパスは

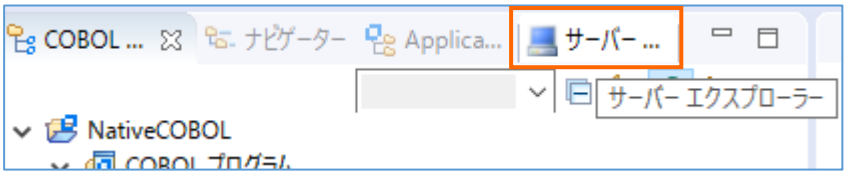
① 上にスクロールし、[適用] ボタンをクリックします。

② 更新された旨のポップアップメッセージが表示されます。

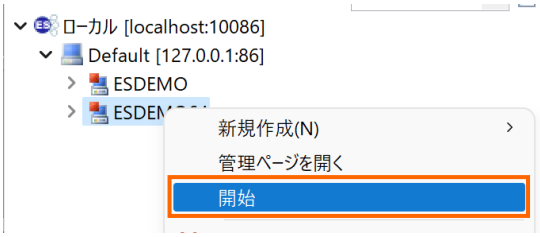


2) Enterprise Server の起動

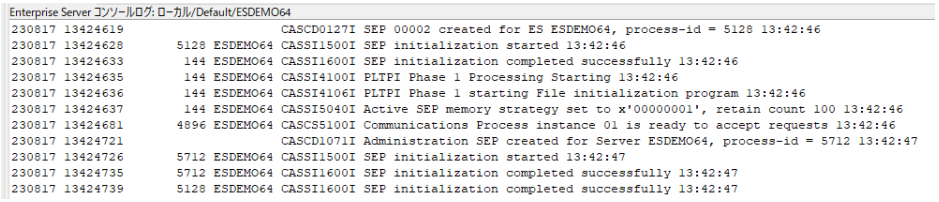
① [サーバーエクスプローラー] タブを選択します。



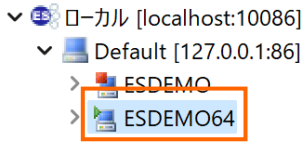
② ツリーを展開し、Visual COBOL にビルドインされている Enterprise Server「ESDEMO64」を右クリックし、[開始] を選択します。もし、ダイアログが表示されたらそのまま[OK]ボタンを押してください。



③ Eclipse の Secure Storage に関するダイアログが表示された場合、[いいえ] を選択してください。開始処理の状況は、[コンソール] ビューでモニターできます。



④ 正常に開始されると［サーバーエクスプローラー］ビュー上の「ESDEMO64」アイコンが起動されたことを示す緑色のアイコンに切り替わります。

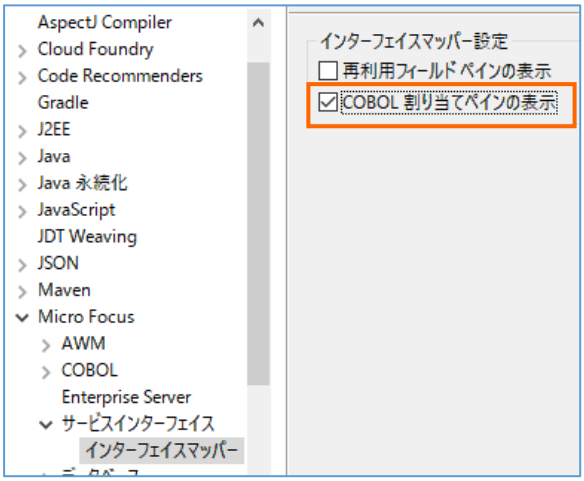


3.5. Visual COBOL for Eclipse にてマッピング指定を変更する

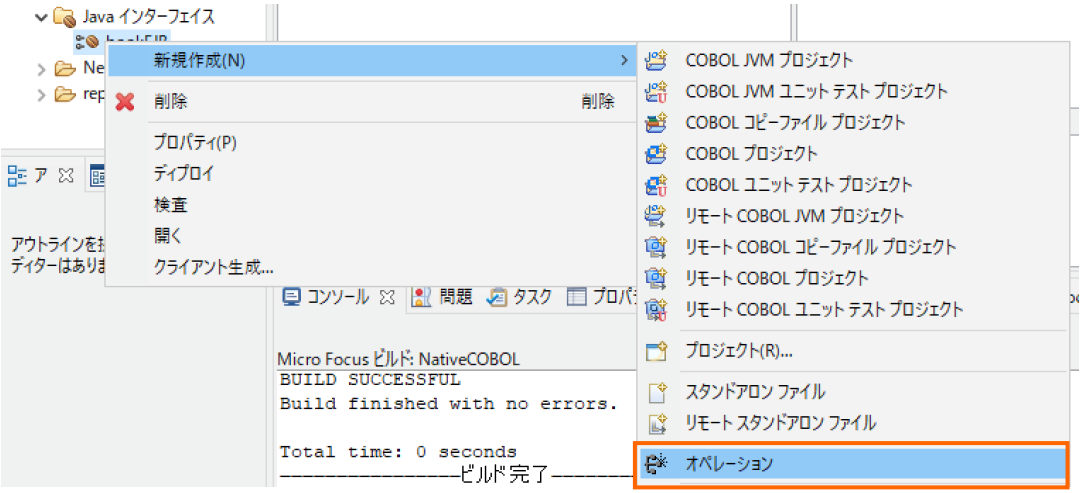
1) COBOL と Java の間の変換マッピングを定義、編集

① ［ウィンドウ(W)］メニューから［設定(P)］を選択し、[Micro Focus] > [サービスインターフェイス] > [インターフェイスマッパー］を選択します。[COBOL 割り当てペインの表示］にチェックを入れ［適用して閉じる］ボタンをクリックします。

※Preference Recorder のダイアログが表示されたら［キャンセル］を選択してください。



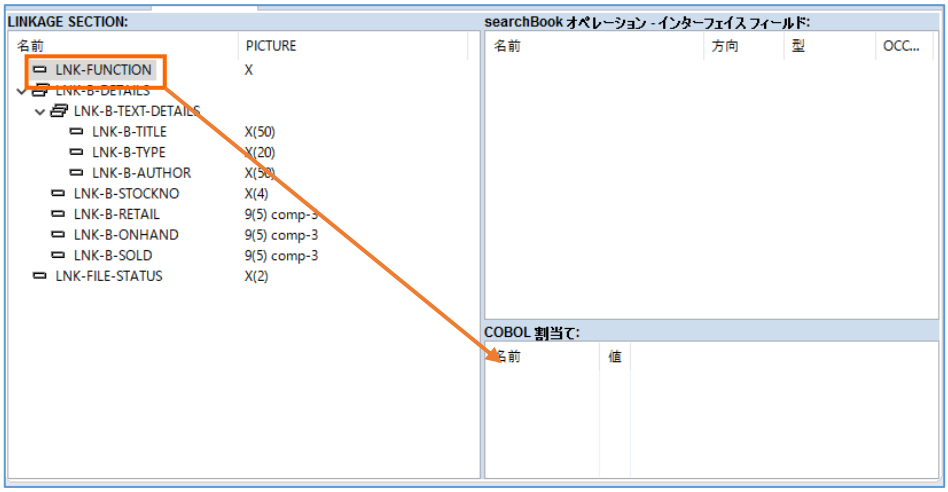
② 生成された bookEJB を右クリックし、コンテクストメニューから［新規作成(N)] > [オペレーション］を選択します。



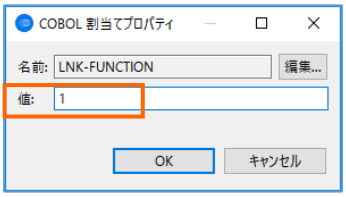
③ [オペレーション名] 欄に "searchBook" を指定し、[OK] ボタンを押します。

④ 生成されたオペレーション「searchBook」をダブルクリックします。

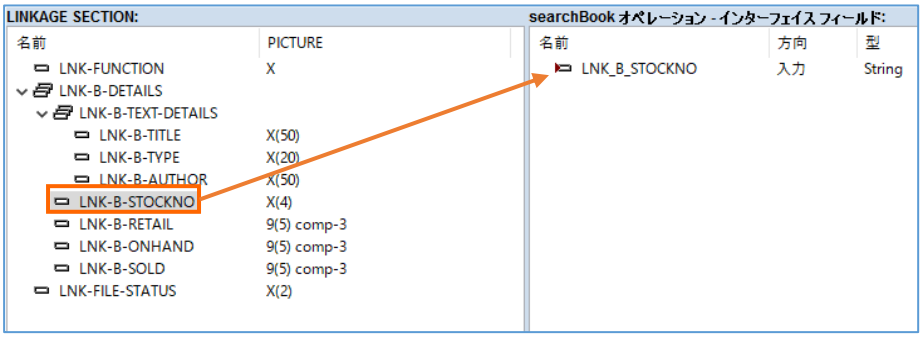
⑤ 生成されたインターフェイスマッピングを編集します。[LINKAGE SECTION] の COBOL 変数 「LNK-FUNCTION」を［COBOL 割当て］にドラッグ&ドロップします。



⑥ [COBOL 割当てプロパティ] ダイアログが表示されるので［値］に "1" を設定して[OK]ボタンをクリックします。



⑦ 次に「LNK_B_STOCKNO」を ［searchBook オペレーション – インターフェースフィールド］にドラッグ＆ドロップします。

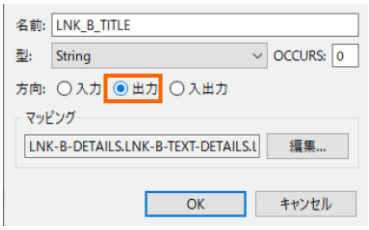


⑧ 同様の手順で、個別に以下のフィールドをドラッグ＆ドロップします。

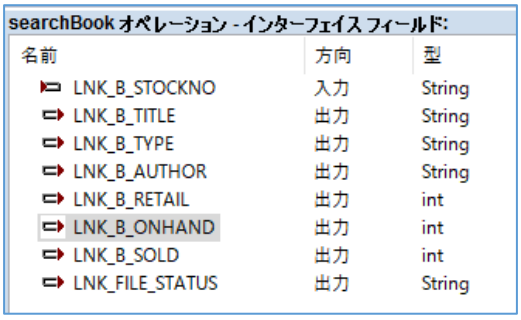
「LNK-B-TITLE」、「LNK-B-TYPE」、「LNK-B-AUTHOR」、

「LNK-B-RETAIL」、「LNK-B-ONHAND」、「LNK-B-SOLD」、「LNK-FILE-STATUS」

⑨ 「LNK_B_TITLE」をダブルクリックします。フィールドプロパティが表示されるので［入力］→［出力］に変更して[OK]ボタンを押します。



⑩ 同様の手順で「LNK_B_STOCKNO」以外の項目は全て方向を出力に変更してください。最終的には下のイメージのような構成になります。



3.6. リソースアダプタの編集と JBoss へディプロイを行う

1) リソースアダプタの編集とディプロイ

① Visual COBOL のマニュアルに従い、JBoss EAP の構成ファイル「standalone.xml」にリソースアダプタの指定をエディタ等で記述します。

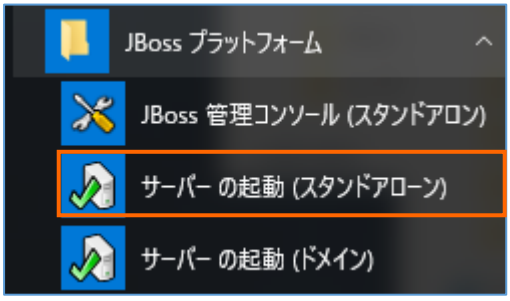
② https://www.amc.rocketsoftware.co.jp/manuals/VC90/Eclipse/vc90indx.html

③ <archive-validation> タグを修正します。

④ <subsystem> タグにてリソースアダプター mfcobol-notx.rar に関連する情報を指定します。

⑤ Visual COBOL インストールディレクトリ配下にある javaee フォルダ配下の javaee7 → jboss74EAP にある「mfcobol-notx.rar」ファイルを JBOSS インストールディレクトリ配下の standalone フォルダ → deployments 配下にコピーします。

⑥ JBoss EAP を起動するために Windows メニューを表示して［JBoss プラットフォーム］ > ［サーバーの起動（スタンドアローン］を選択します。



⑦ リソースアダプタ「mfcobol-notx.rar」ファイルが適切にデプロイされていることをコンソールメッセージで確認します。

※standalone.xml の編集の詳細はマニュアルの［ディプロイ］> [Enterprise Server へのディプロイ] > [モダナイズ

済みのアプリケーションのディプロイおよび構成] > [EJB およびリソース アダプターのディプロイ] > [EJB のディプロイ - 概要] > [JBoss へのディプロイ] を参照してください。

3.7. Visual COBOL for Eclipse にて JCA プロファイルを構成する

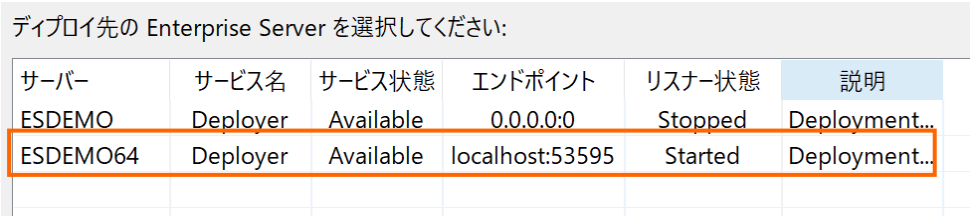
1) EJB/JCA アプリケーションのプロファイルを構成

① 「3.5 Visual COBOL for Eclipse にてマッピング指定を変更する」で作成したプロファイル「bookEJB」を右クリックし、コンテクストメニューから［プロパティ］を選択します。

② ［ディプロイメントサーバー］タブをクリックします。

③ ［Enterprise Server 名］欄にて［変更］ボタンを押します。

④ ディプロイする Enterprise Server の一覧が表示されるので［ESDEMO64］選択して［OK］ボタンを押します。



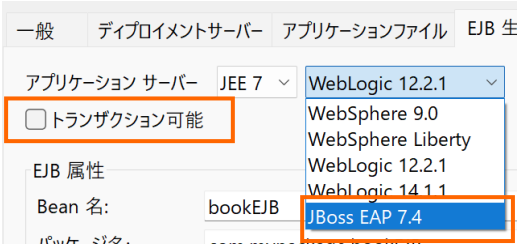
⑤ ［アプリケーションファイル］タブをクリックし、［レガシーアプリケーションをディプロイする］を選択します。

⑥ ［ファイル追加］ボタンを押します。

⑦ Ctrl を押しながら、「BOOK.gnt」及び「BOOK.idy」を選択し［開く］ボタンを押します。

⑧ ［EJB 生成］タブをクリックします。

⑨ ［アプリケーションサーバー］欄にて、「JEE 7」及び「JBoss EAP 7.4」を選択し、［トランザクション可能］のチェックを外します。



⑩ ［Java コンパイラ］欄は、インストールされている JDK を指定します。例えばここでは、チュートリアル用に別途インストールした ”C:¥Java¥JDK1.8.0¥bin” を指定しています。

⑪ ［Java EE クラスパス］欄は［参照］ボタンを押し、エクスプローラーにて下記の 3 つのクラスファイルを指定します。
C:/JBoss/EAP-7.4.0/modules/system/layers/base/javax/ejb/api/main/jboss-ejb-api_3.2_spec-2.0.0.Final-redhat-00001.jar

⑫ ⑪の要領で
C:/JBoss/EAP-7.4.0/modules/system/layers/base/javax/resource/api/main/jboss-connector-api_1.7_spec-2.0.0.Final-redhat-00001.jar

⑬ ⑪の要領で
C:/JBoss/EAP-7.4.0/modules/system/layers/base/javax/servlet/api/main/jboss-servlet-api_4.0_spec-2.0.0.Final-redhat-00001.jar

⑭ ［J2EE クラスパス］欄に jar ファイルが 3 つセミコロン「;」で区切られて追加されていることが確認できたら、［OK］ボタンを押します。

2) Enterprise Server の再起動

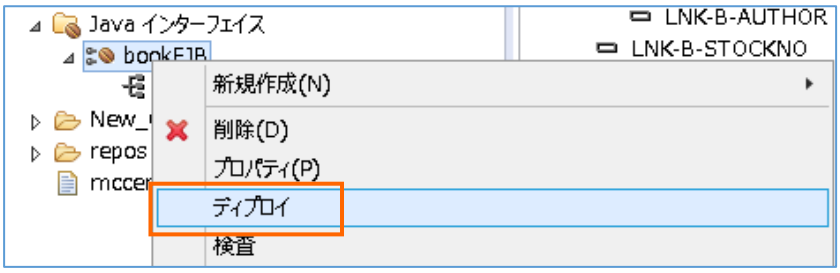
① Eclipse IDE に戻り、「サーバーエクスプローラー」に切り替えます。

② 「ESDEMO64」上で右クリックし、コンテクストメニューから［再起動］を選択し、Enterprise Server を停止/起動します。

3.8. テスト用クライアントアプリケーションを生成する

1) ディプロイした Java サービスをテストするための J2EE アプリケーションを生成する

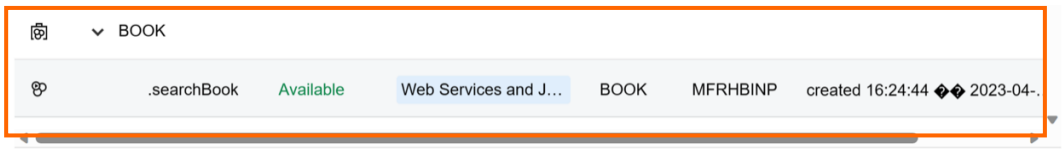
① “bookEJB” を右クリックし、コンテクストメニューから[ディプロイ] を選択します。



② さきほどブラウザー上で開いていた「管理画面」に切り替えます。

③ [一般]メニューから[サービス]をクリックします。

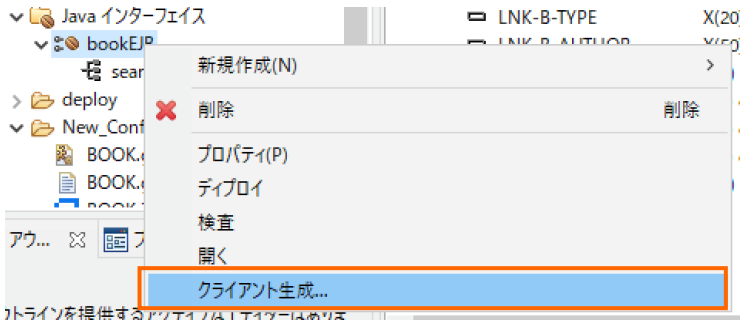
④ 画面を下にスクロールしていくと最下行にディプロイした JCA サービスが追加されていることを確認します。



3.9. テスト用アプリケーションの生成とテスト実行

1) ディプロイした Java サービスをテストするための Java EE アプリケーションを生成する

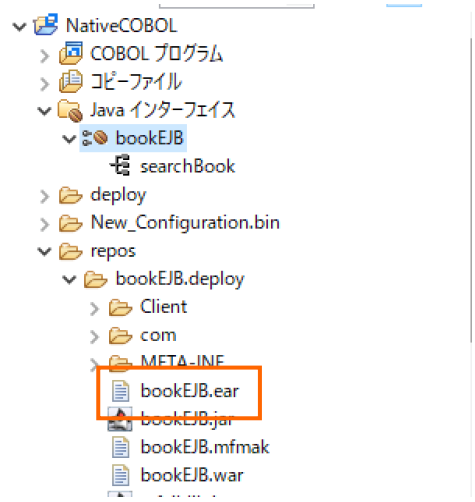
① 「bookEJB」を右クリックし、コンテクストメニューから[クライアント生成...] を選択します。



② [サービスインターフェイス コンソール] ビューに正常にクライアントアプリケーションが生成された旨のメッセージが出力されます。

```
マニフェストが追加されました
bookEJB.jarを追加中です(入=3056)(出=2747)(10%収縮されました)
bookEJB.warを追加中です(入=19427)(出=18513)(4%収縮されました)
mfejblib.jarを追加中です(入=11614)(出=10418)(10%収縮されました)
META-INF/application.xmlを追加中です(入=527)(出=261)(50%収縮されました)
META-INF/jboss-deployment-structure.xmlを追加中です(入=509)(出=201)(60%収縮されました)
C:\Program Files (x86)\Micro Focus\Visual COBOL\bin64
クライアント生成が正常終了しました。
```

③ 生成されたクライアントアプリケーション「bookEJB.ear」を JBoss EAP インストールディレクトリ配下の standalone フォルダ → deployments 配下にコピーします。



④ 正常にディプロイされたことを JBoss EAP の起動コンソール画面で確認します。

2) ディプロイした Java アプリケーションのテストを行う

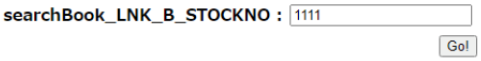
① ブラウザーを起動します。

② アドレスバーに http://localhost:8080/bookEJB/searchBook.jsp を入力します。

③ アプリケーションの動作を確認します。

④ LNK_B_STOCKNO_io に“1111”を指定して［Go!］ボタンを押します。

⑤ ブラウザーに COBOL プログラムがデータファイルより読み込んだ内容が返されます。



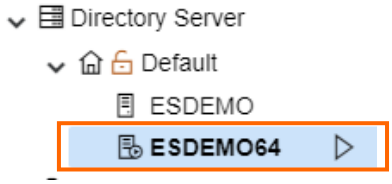
Result:

Variable	Value
LNK_B_TITLE	LORD OF THE RINGS
LNK_B_TYPE	FANTASY
LNK_B_AUTHOR	TOLKIEN
LNK_B_RETAIL	1500
LNK_B_ONHAND	4000
LNK_B_SOLD	3444
LNK_FILE_STATUS	00

3.10. Enterprise Server にディプロイしたアプリケーションのデバッグ

1) 動的デバッグの許可

① さきほどブラウザー上で開いた管理画面の左側より「ESDEMO64」の情報をクリックします。



② 画面の[一般]をクリックし、[動的デバッグを許可]をチェックしたうえで、[適用]ボタンをクリックします。

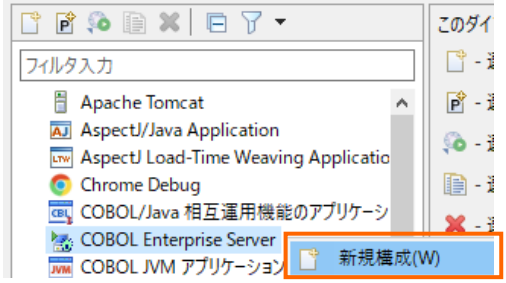
③ Eclipse IDE に戻り、「サーバーエクスプローラー」に切り替えます。

④ 「ESDEMO64」上で右クリックし、コンテクストメニューから［再起動］を選択し、Enterprise Server を再起動します。

2) アプリケーションのデバッグ準備

① [実行(R)]メニューより、[デバッグの構成(B)]を選択します。

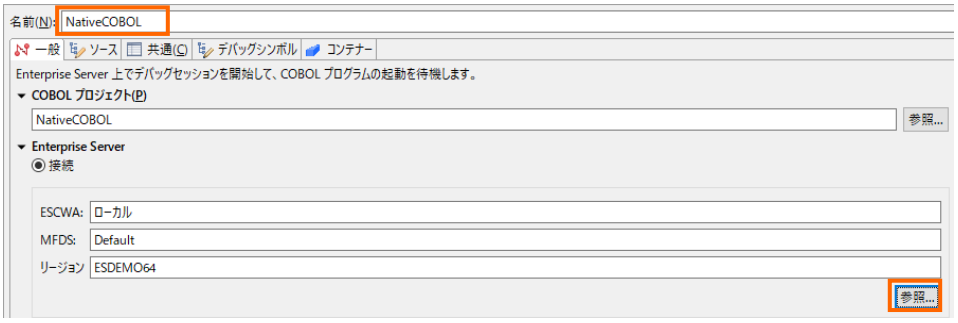
② 左側より、「COBOL Enterprise Server」上で右クリックし、コンテクストメニューから［新規構成(W)］を選択します。



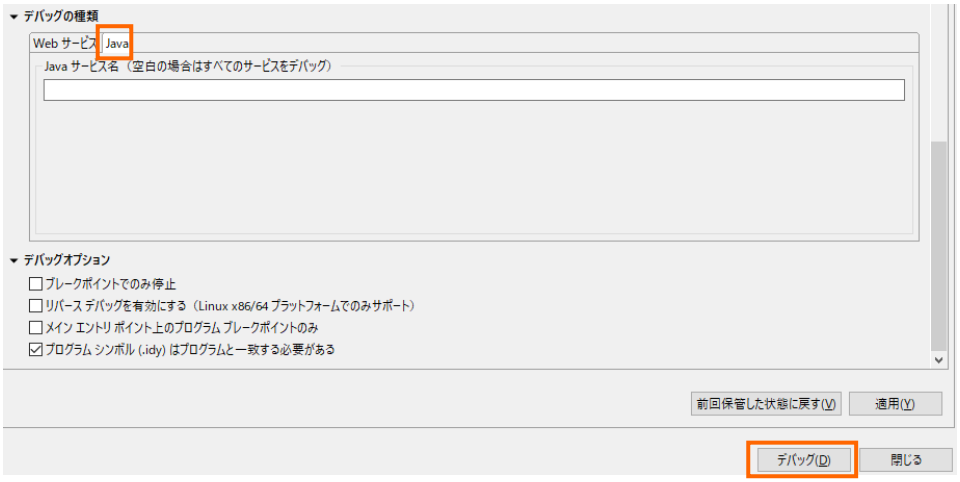
③ 以下の入力を行い、[デバッグ(D)] をクリックします。

名前： “NativeCOBOL”

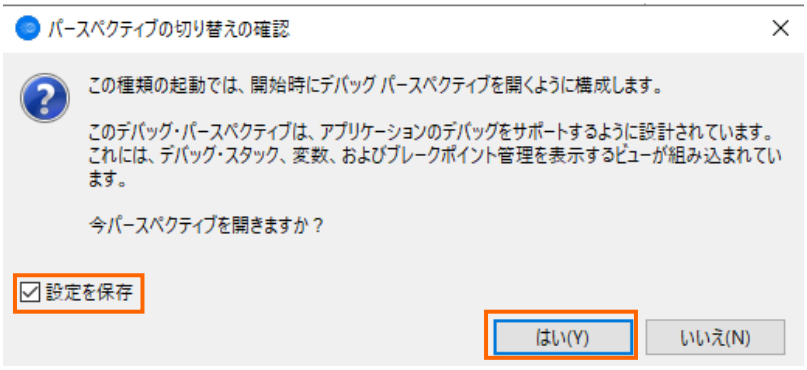
Enterprise Server を選択し、[参照] をクリックして、「ESDEMO64」を選択



デバッグの種類で「Java」を選択

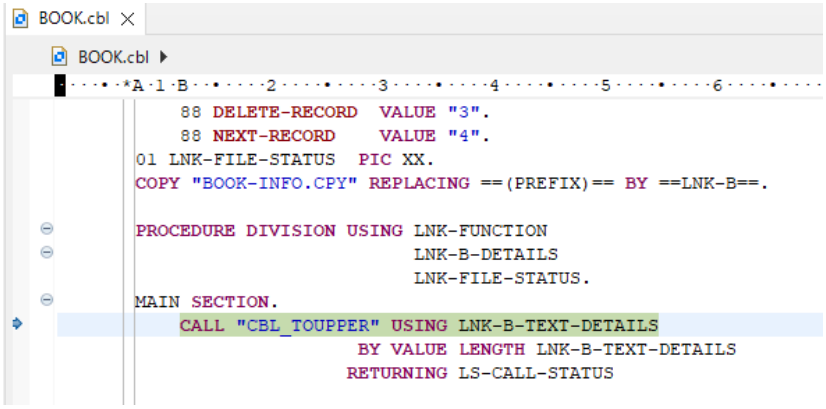


④ 以下のダイアログが表示された場合は、[設定を保存] にチェックをして [はい(Y)] をクリックします。



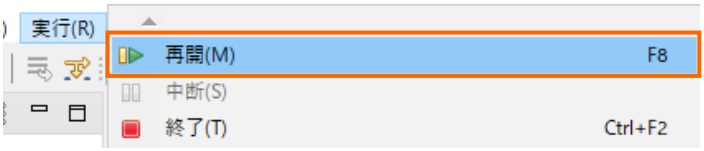
⑤ さきほど同様、ブラウザー上で http://localhost:8080/bookEJB/searchBook.jsp にアクセスし、”1111”を入力して、[Go!] をクリックします。

⑥ Eclipse IDE を見ると、デバッグパースペクティブ上で、BOOK.cbl で停止していることが分かります。



通常のコンソールアプリケーションと同様、項目値の確認や、ステップ実行などが行えます。

デバッグを終え、画面に応答結果を戻すためには、[実行(R)]メニューより［再開(M)］を選択してください。



⑦ デバッグ作業を終了する際は、[実行(R)］メニューより［終了(T)］を選択してください。

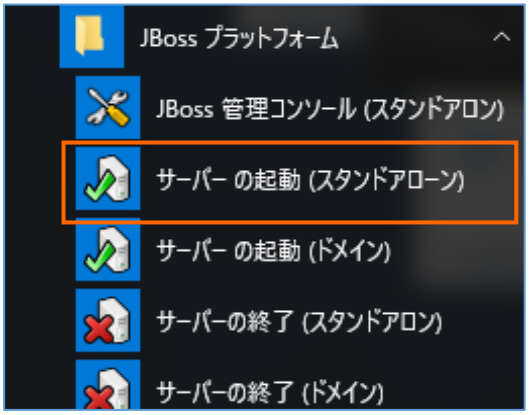
3.11. JBoss の停止と Enterprise Server インスタンスの停止

1) Enterprise Server の停止

① 「サーバーエクスプローラー」にて「ESDEMO」上で右クリックし、コンテクストメニューから［停止］を選択します。

2) JBoss EAP の停止

① Windows メニューを表示して［JBoss プラットフォーム］ > ［サーバーの終了（スタンドアローン）］を選択します。



② JBoss が起動されたコマンドプロンプト、JBoss を停止したコマンドプロンプトの両方の DOS プロンプトを閉じます。

WHAT'S NEXT

- 本チュートリアルで学習した技術の詳細については製品マニュアルをご参照ください。

JCA による JBOSS EAP 連携の設定と開発

免責事項

ここで紹介したソースコードは、機能説明のためのサンプルであり、製品の一部ではございません。ソースコードが実際に動作するか、御社業務に適合するかなどに関しまして、一切の保証はございません。 ソースコード、説明、その他すべてについて、無謬性は保障されません。
ここで紹介するソースコードの一部、もしくは全部について、弊社に断りなく、御社の内部に組み込み、そのままご利用頂いても構いません。
本ソースコードの一部もしくは全部を二次的著作物に対して引用する場合、著作権法の精神に基づき、適切な扱いを行ってください。