
Micro Focus Visual COBOL チュートリアル

RESTful Web サービスによる COBOL 資産の再利用 Eclipse 編

1. 目的

Micro Focus 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 REST API を利用し Web サービスとして呼び出す機能を提供しています。RESTful の Web サービスとして呼び出しを行う場合、JSON 形式でやり取りが可能であれば呼び出し側のプログラムに依存することなく連携できるようになります。

このドキュメントでは COBOL のソースコードに一切手を加えることなくビジネスロジックとして Enterprise Server にデプロイし、それを Visual COBOL のクライアント生成機能を使って動作確認用のクライアントを作成し連携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 Linux/UNIX でも利用可能です。

- 開発クライアント ソフトウェア

- OS Windows Server 2016 Standard Edition (64bit)

- COBOL 開発環境製品 Micro Focus Visual COBOL 6.0J 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

1. 目的
2. 前提条件
3. チュートリアル手順の概要
 - 3.1. Windows クライアントでの開発準備作業
 - 3.2. Enterprise Server に登録するサービスのデプロイ先変更
 - 3.3. RESTful Web サービスの開発作業
 - 3.4. コンパイルした COBOL アプリケーションを Enterprise Server へデプロイ
 - 3.5. RESTful Web サービスのテスト
 - 3.6. インスタンスの停止

3. チュートリアル手順の概要

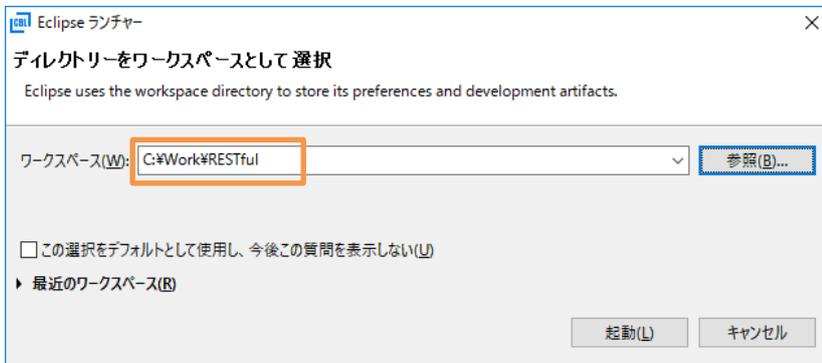
3.1. Windows クライアントでの開発準備作業

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。

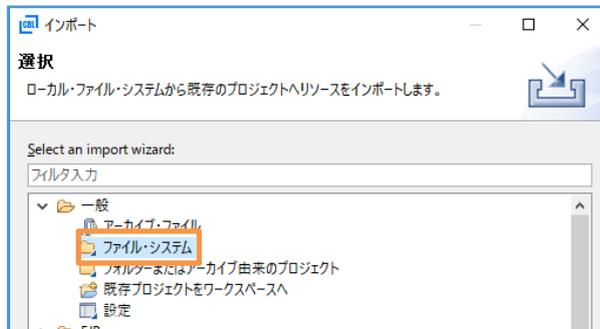


- ② ワークスペースの選択画面にて “C:\Work\RESTful” を指定し、[起動] ボタンをクリックします。



2) ネイティブ COBOL プロジェクトの作成とプログラムソースのインポート

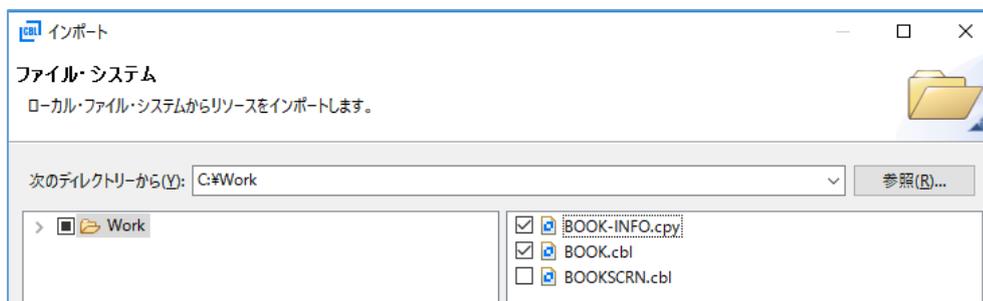
- ① [ファイル]メニュー > [新規] > [COBOL プロジェクト] を選択し、プロジェクト名に “NativeCOBOL” を指定して、[終了(F)] ボタンをクリックします。
- ② COBOL エクスプローラーのパーспекティブを開き、COBOL エクスプローラービューにて プロジェクトフォルダを右クリックし、コンテキストメニューから [インポート] > [インポート] を選択します。
- ③ 既存のソースコードをロードします。一般のフォルダを展開し、[ファイル・システム] を選択し、[次へ(N)] ボタンをクリックします。



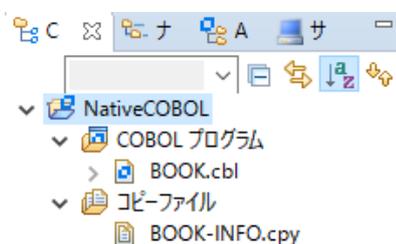
途中省略



- ④ チュートリアル用のファイルをインポートします。インポートダイアログが表示されるので [参照(R)] ボタンをクリックし、任意のフォルダにダウンロードしたファイルを解凍し、そのフォルダを指定します。下図では “C:\work” に解凍したファイルを指定しています。ここで “BOOK-INFO.cpy” と “BOOK.cbl” を指定し、[終了(F)] ボタンをクリックします。

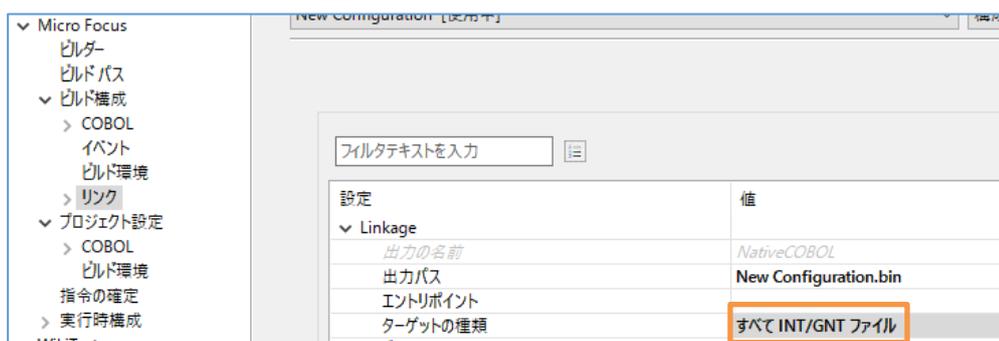


- ⑤ プロジェクトフォルダを展開し、2つのファイルが正常にロードされていることを確認します。



3) ビルドオプションの変更

- ① プロジェクトの構成を変更します。COBOL エクスプローラーにて作成した「NativeCOBOL」プロジェクトを右クリックし、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] > [リンク] をクリックし、[ターゲットの種類] を「すべて INT/GNT ファイル」に変更します。



- ③ 次に[Micro Focus] > [プロジェクト設定] > [COBOL] をクリックし、[.GNT にコンパイル] を「はい」に変更し、[追加指令]に “ASSIGN(EXTERNAL)” を指定し、[Apply and Close] ボタンをクリックします。



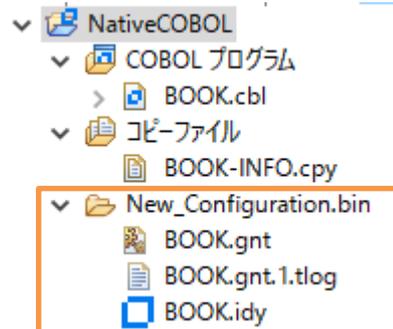
Micro Focus
ビルダー
ビルドパス
ビルド構成
COBOL
イベント
デプロイ
ビルド環境
リンク
プロジェクト設定
指令の確定
実行時構成
WikiText
タスク・タグ
タスク・リポジトリ
ビルダー
プロジェクト・ネーチャー
プロジェクト・ファセット
プロジェクト参照
検証
実行/デバッグ設定

構成の固有な設定を可能にする(C)

フィルタテキストを入力

設定	値
一般	
文字セット	ASCII
COBOL 方言	Micro Focus
ソース フォーマット	固定
デバッグ用にコンパイル	はい
EXIT PROGRAM を GOBACK として処理	いいえ
詳細	いいえ
.GNT にコンパイル	はい
出力	
命令ファイルを作成する	いいえ
リストファイルを作成	いいえ
コードカバレッジを有効にする	false
プロファイラを有効にする	false
エラー/警告	
警告レベル	回復可能なエラーを含める(レベル E)
最大エラー数	100
追加指令	ASSIGN(EXTERNAL)

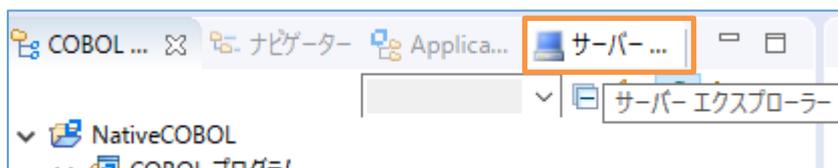
- ④ COBOL エクスプローラーにて “New_Configuration.bin” を展開して下記のファイルが作成されていることを確認します。



NativeCOBOL
COBOL プログラム
BOOK.cbl
コピーファイル
BOOK-INFO.cpy
New_Configuration.bin
BOOK.gnt
BOOK.gnt.1.tlog
BOOK.idy

4) 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。

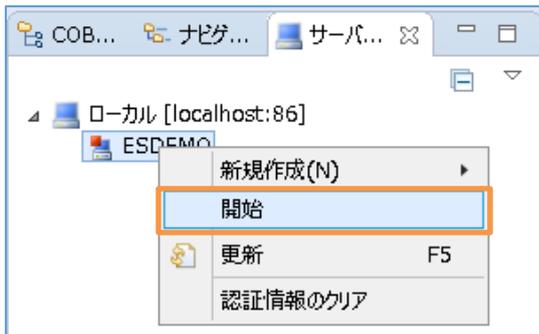


COBOL ... ナビゲーター Applica... サーバー ...

サーバー エクスプローラー

NativeCOBOL
COBOL プログラム

- ② Visual COBOL にビルドインされている Enterprise Server 「ESDEMO」 を右クリックし、コンテキストメニューから [開始] を選択します。

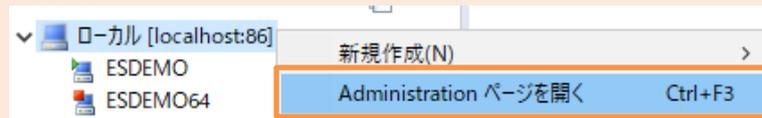


注意)

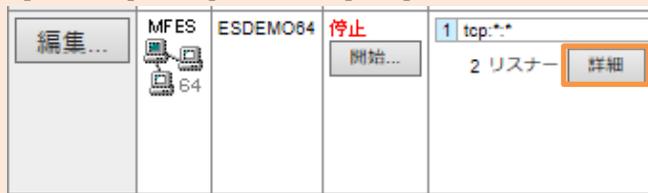
予めセットアップされた「ESDEMO」ではなく、自分で作成した Enterprise Server を使用する場合は、以下の手順にて、サーバー起動前に Web リスナーの「新ステータス」を“停止”に変更してください。この手順を実施していない場合、後の手順で行う RESTful Web サービスのデプロイに失敗します。

なお、ここでは、自分で作成した Enterprise Server 名を「ESDEMO64」としています。

- 1) 「ローカル[localhost:86]」を右クリックし、コンテキストメニューから [Administration ページを開く] を選択します。



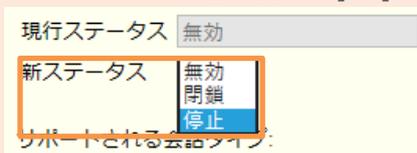
- 2) Enterprise Server Administration コンソール画面にて「ESDEMO64」サーバー欄の [通信プロセス] 列中の [リスナー] の横にある [詳細] ボタンをクリックします。



- 3) リスナーの構成情報が表示されるので [名前] 列中の [Web] の [編集] ボタンをクリックします。

	名前	アドレス	ステータス
	Web Services and J2EE	tcp:.*	停止
	Web	tcp:127.0.0.1:*(localhost)	無効

- 4) 新ステータスを「停止」に変更し、[OK] ボタンをクリックします。



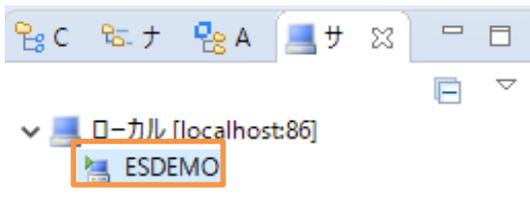
- ③ Eclipse の Secure Storage に関するダイアログが表示された場合、[いいえ] を選択してください。開始処理の状況は、[コンソール] ビューでモニターできます。

```

コンソール 問題 タスク プロパティ
Enterprise Server
開始サーバー: ESDEMO
....
CASCD0167I ES Daemon successfully auto-started 11:23:07
CASCD0050I ES "ESDEMO" initiation is starting 11:23:07

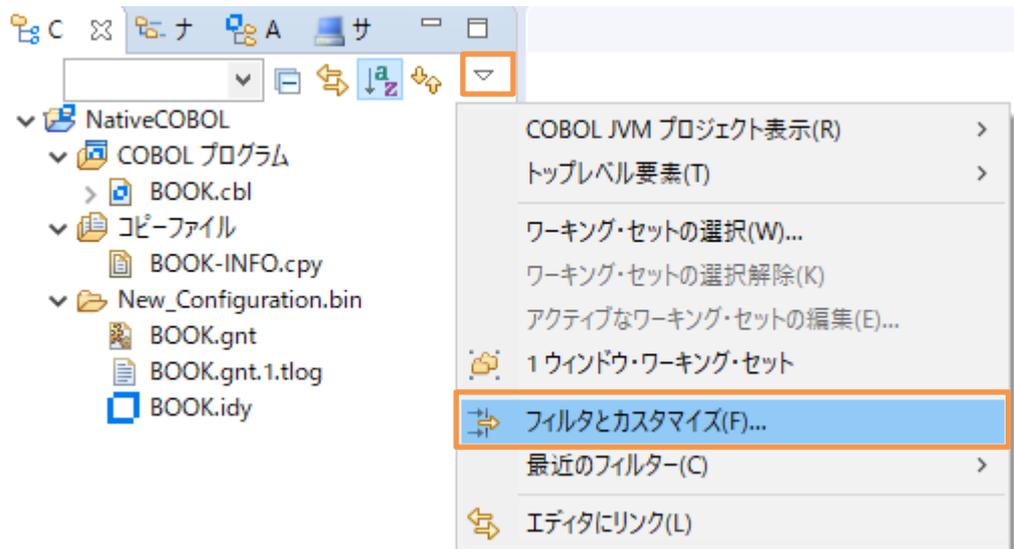
サーバー: ESDEMO 正常に起動されました
  
```

- 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「ESDEMO」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。

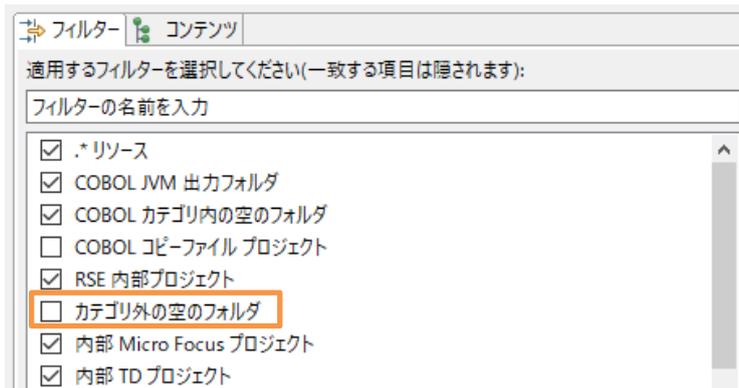


5) COBOL エクスプローラー表示設定の変更

- 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「▽」アイコンをクリックし、[フィルタとカスタマイズ] 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。



6) デploy用フォルダの作成

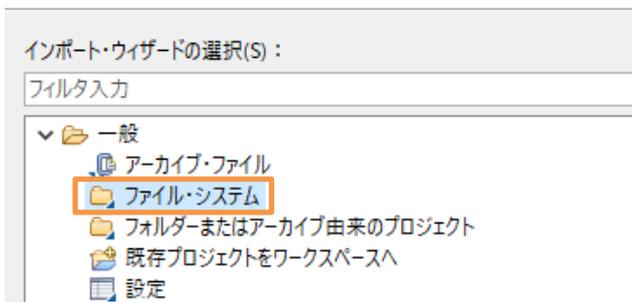
- ① 「NativeCOBOL」プロジェクト上で右クリックし、コンテキストメニューから [新規作成]→[フォルダー] を選択します。
- ② 「NativeCOBOL」プロジェクトが選択されていることを確認の上、フォルダ名に “deploy” を指定し、[終了(F)] ボタンをクリックします。

7) 「.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、コンテキストメニューから [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。

選択

ローカル・ファイル・システムから既存のプロジェクトへリソースをインポートします。

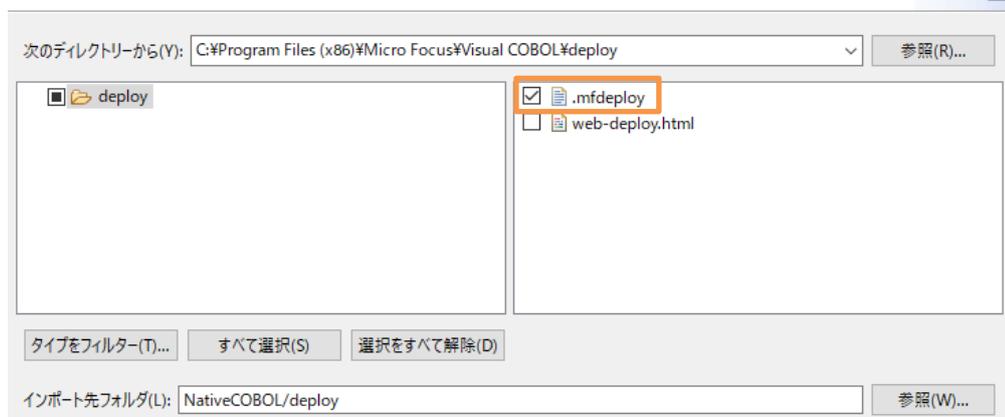


- ③ [参照(R)] ボタンをクリックし、Visual COBOL インストールフォルダ¥deploy を選択します。

- 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。

ファイル・システム

ローカル・ファイル・システムからリソースをインポートします。



補足)

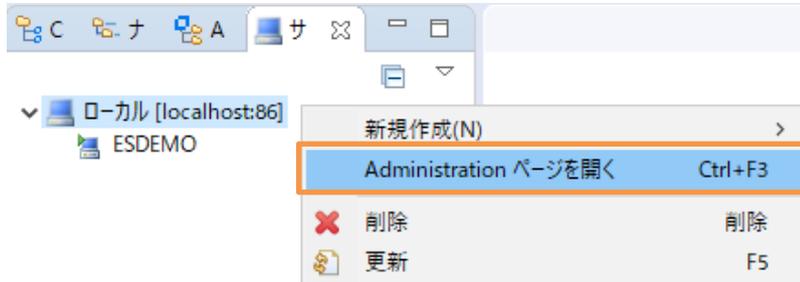
上記手順完了後も、COBOL エクスプローラー上の NativeCOBOL\deploy フォルダ配下に .mfdeploy ファイルは表示されませんが、これはフィルタによるものです。

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。

3.2. Enterprise Server に登録するサービスのディプロイ先変更

1) RESTful Web サービスのディプロイ先フォルダの変更

- 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、コンテキストメニューから [Administration ページを開く] を選択します。



- ③ Enterprise Server Administration コンソール画面にて [通信プロセス] 列中の [リスナー] の横にある [詳細] ボタンをクリックします。



- ④ リスナーの構成情報が表示されるので [名前] 列中の [Web] の [編集] ボタンをクリックします。

編集...	HTTP Echo	tcp:192.168.56.1*:9002 (192.168.56.1)	開始
編集...	Web	tcp:192.168.56.1*:51231* (192.168.56.1)	開始
編集...	Web Services and J2EE	tcp:192.168.56.1*:9003 (192.168.56.1)	開始

- ⑤ 構成情報フィールドを見るとデフォルトは「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。

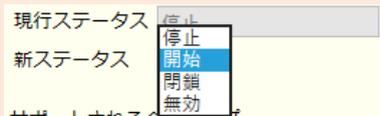
下記のように設定を変更し、[OK] ボタンをクリックします。

例：uploads=c:/work/RESTful/NativeCOBOL/deploy

注意)

Web リスナーが開始状態で、この変更を行うと Web リスナーが「停止」状態になります。

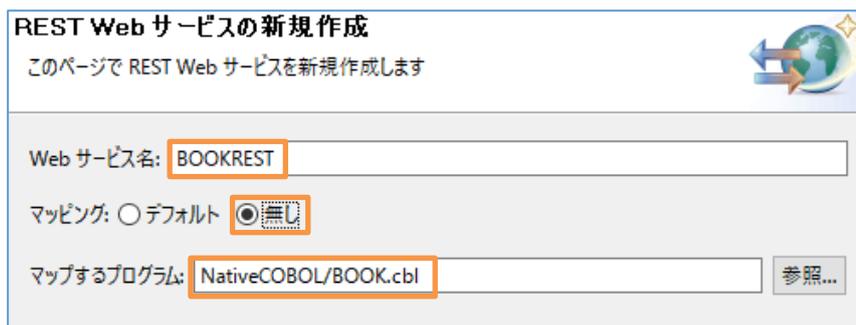
このため、④の手順を再度行い、新ステータスを「開始」に変更した上で、[OK] ボタンをクリックします。



3.3. RESTful Web サービスの開発作業

1) RESTful Web サービスのプロファイル作成

- ① COBOL エクスプローラーに戻ります。
- ② RESTful Web サービスとして利用するビジネスロジックを処理するプログラム「BOOK.cbl」を右クリックし、コンテキストメニューから [新規作成] > [REST Web サービス] を選択します。
- ③ REST Web サービスの新規作成ウィザードが表示されます。[Web サービス名] 欄に “BOOKREST” を指定します。[マッピング] 欄は「無し」を選択、[マップするプログラム] 欄には「NativeCOBOL/BOOK.cbl」が選択されていることを確認し [終了(F)] ボタンをクリックします。



2) 書籍データ検索機能のオペレーションを作成

- ① 1) で作成した Web サービスプロファイル「BOOKREST」を右クリックし、コンテキストメニューから [新規作成] > [オペレーション] を選択します。
- ② オペレーションプロパティウィンドウが表示されます。[オペレーション名] 欄に “SEARCHBOOK” を入力します。

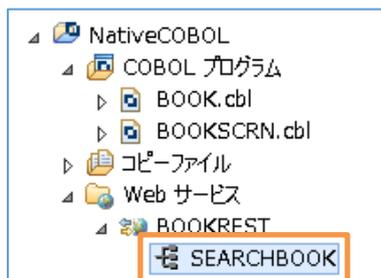


- ③ 次に [HTTP] タブを選択します。HTTP メソッドが [POST] になっていることを確認し、[OK] ボタンをクリックします。

3) 書籍データ検索機能オペレーションのインターフェイスマッピングを定義

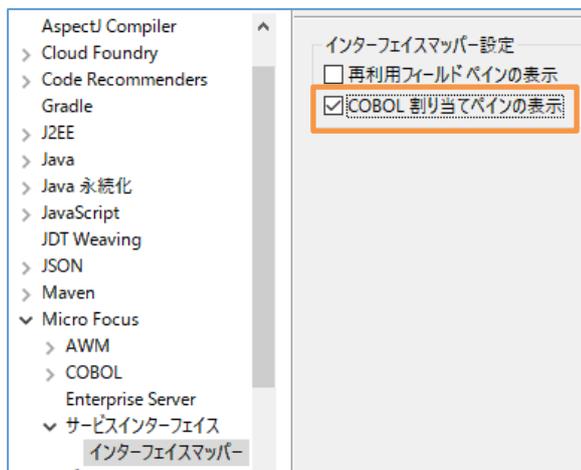
Enterprise Server は Web サービス側のデータ型と COBOL のデータ型を相互に自動変換させる機能を装備しています。この機能により Web サービスコンシューマー側と COBOL 側はそれぞれ相手のデータ型を意識することなく透過的にデータ変換が処理されやりとりができます。

- ① 「SEARCHBOOK」オペレーションをダブルクリックします。

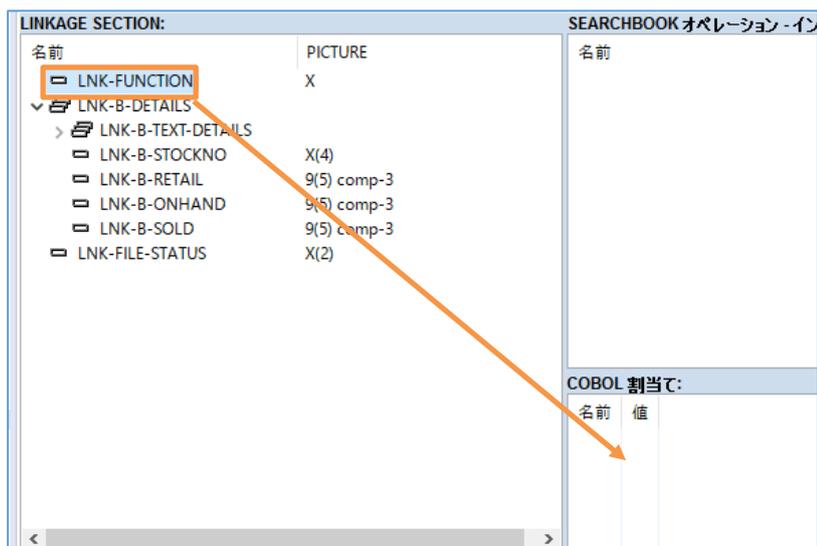


- ② [ウィンドウ] メニューから [設定] を選択し、[Micro Focus] > [サービスインターフェース] > [インターフェイスマッパー] を選択します。[COBOL 割り当てペインの表示] にチェックを入れ [Apply and Close] ボタンをクリックします。

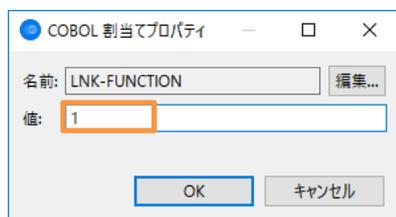
※Preference Recorder のダイアログが表示されたら [キャンセル] を選択してください。



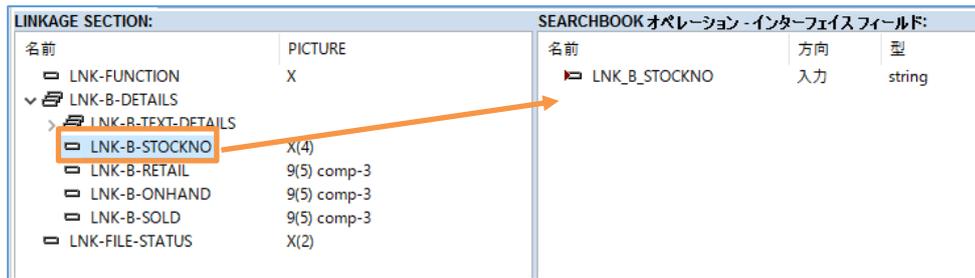
- ③ LINKAGE SECTION の COBOL の変数「LNK-FUNCTION」を [COBOL 割り当て] にドラッグ&ドロップします。



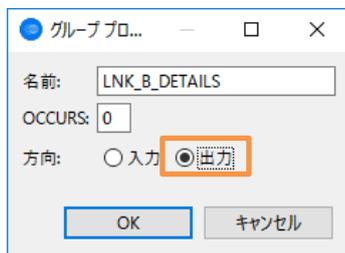
- ④ [COBOL 割当てプロパティ] ダイアログが表示されるので [値] に "1" を設定して [OK] ボタンをクリックします。



- ⑤ 次に「LNK-B-STOCKNO」を [SEARCHBOOK オペレーション - インターフェイスフィールド] にドラッグ&ドロップします。



- ⑥ 「LNK-B-DETAILS」もドラッグ&ドロップします。
 ⑦ ドラッグ&ドロップされた [LNK_B_DETAILS] をダブルクリックします。フィールドプロパティダイアログが表示されるので [方向] を「出力」へ変更します。



- ⑧ 最後に「LNK-FILE-STATUS」をドラッグ&ドロップし、[方向] を「出力」に変更し、[OK] ボタンをクリックします。最終的には下のイメージのような構成になります。



- 4) 書籍データ追加機能のオペレーションを追加
- ① 1) で作成した Web サービスプロファイル「BOOKREST」を右クリックし、コンテキストメニューから [新規作成] > [オペレーション] を選択します。
 - ② オペレーションプロパティウィンドウが表示されます。[オペレーション名] 欄に "ADDBOOK" を入力します。
 - ③ 次に [HTTP] タブを選択します。HTTP メソッドが [POST] になっていることを確認し、[OK] ボタンをクリックします。
- 5) 書籍データ登録機能オペレーションのインターフェイスマッピングを定義
- ① 「ADDBOOK」オペレーションをダブルクリックします。

- ② 「LNK-FUNCTION」を COBOL 割当てにドラッグ&ドロップし、[値] には “2” を指定します。
- ③ 「LNK-B-DETAILS」、「LNK-FILE-STATUS」それぞれを右側の [ADDBOOK オペレーション - インターフェイス フィールド] ヘッドラッグ&ドロップします。
- ④ 「LNK_FILE_STATUS」の方向を「出力」へ変更します。最終的には下のイメージのような構成になります。

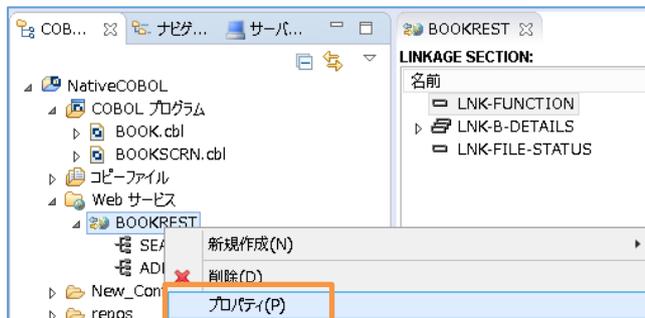
ADDBOOK オペレーション - インターフェイス フィールド:		
名前	方向	型
LNK_B_DETAILS	入力	
LNK_B_TEXT_DETAILS		
LNK_B_TITLE		string
LNK_B_TYPE		string
LNK_B_AUTHOR		string
LNK_B_STOCKNO		string
LNK_B_RETAIL		integer
LNK_B_ONHAND		integer
LNK_B_SOLD		integer
LNK_FILE_STATUS	出力	string

COBOL 割当て:	
名前	値
LNK-FUNCTION	2

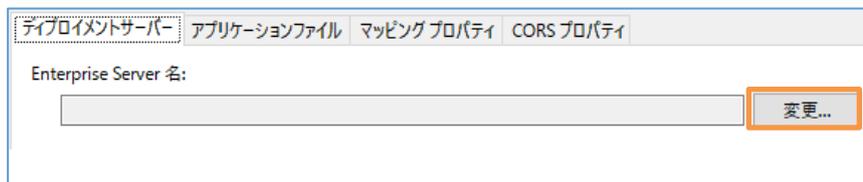
3.4. コンパイルした COBOL アプリケーションを Enterprise Server へデプロイ

1) Enterprise Server へのデプロイ情報を指定

- ① COBOL エクスプローラーにて追加した Web サービス「BOOKREST」を右クリックし、コンテキストメニューから [プロパティ(P)] を選択します。



- ② [デプロイメントサーバー] タブを選択し、[変更] ボタンをクリックします。



- ③ 起動済みの Enterprise Server 「ESDEMO」 を選択し、[OK] ボタンをクリックします。

デプロイ先の Enterprise Server を選択してください:

サーバー	サービス名	サービス状態	エンドポイント	リスナー状態	説明
ESDEMO	Deployer	Available...	192.168.17.13...	BitMode=32-Bit	Deployment file-upload service

注意)

Enterprise Server インスタンスが起動した後、Web リスナーが起動するまで、エンドポイントにアドレスが設定されません。アドレスが未設定の場合、Web リスナーの設定を確認してください。

なお、Enterprise Server インスタンスを予めセットアップされた「ESDEMO」ではなく、自分で作成したインスタンスを利用する場合は、3.1 に記載されている手順にて Web リスナーを有効化してください。

- ④ 次に [アプリケーションファイル] タブを選択し、「レガシーアプリケーションをデプロイする」を選択します。
- ⑤ [ファイル追加] ボタンを押して、プロジェクトディレクトリ配下の「New_Configuration.bin」に生成された「BOOK.gnt」および「BOOK.idy」を選択し、[開く(O)] ボタンをクリックします。
- ⑥ アプリケーションファイルが入ったマッピングプロパティ画面に戻るので [OK] ボタンをクリックします。

デプロイメントサーバー アプリケーションファイル マッピングプロパティ CORS プロパティ

レガシーアプリケーションをデプロイ済みか、またはサーバーにデプロイする必要があるかを指定してください。

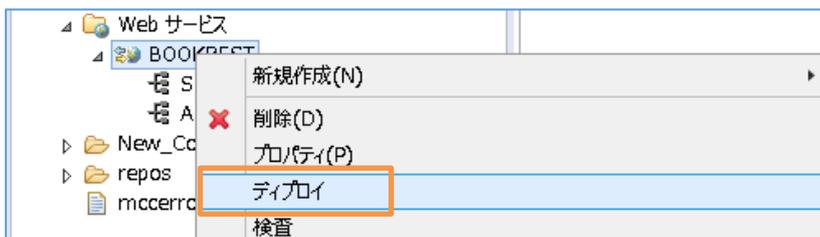
レガシーアプリケーションは既にデプロイ済み
 デプロイされたアプリケーションのパス:

レガシーアプリケーションをデプロイする

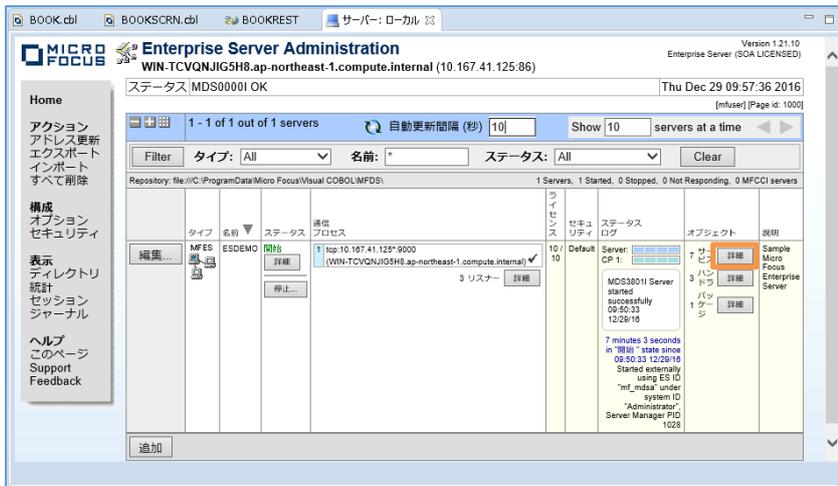
アプリケーションファイル:

2) RESTful Web サービスを Enterprise Server ヘデプロイ

- ① COBOL エクスプローラーにて作成した Web サービス「BOOKREST」を右クリックし、コンテキストメニューから [デプロイ] を選択します。



- ② 「Enterprise Server Administration」 コンソール画面より [Home] をクリックし、「名前」列が「ESDEMO」となっている [サービス] の横にある [詳細] ボタンをクリックします。



補足)

サービス数や下記のサービス一覧で表示される内容は、Enterprise Server に登録されたサービス内容により異なります。

- ③ 最下行にデプロイした RESTful Web サービスが追加されていることを確認します。

削除...		2 of 2 operations shown							
#SEARCHBOOK	編集...	1	1	CP 1	Web Services and J2EE	MFRHJJSON	/temp/path/BOOKREST/1.0	Available	OK
#ADDBOOK	編集...	1	1	CP 1	Web Services and J2EE	MFRHJJSON	/temp/path/BOOKREST/1.0	Available	OK

補足)

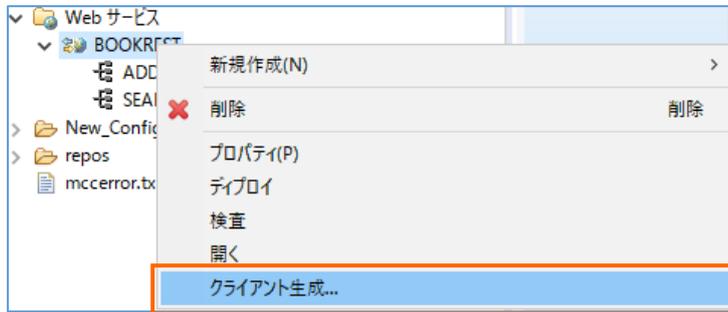
登録サービス数が多い場合は、画面右上の「ページ内表示数」の変更や、[<< 前へ]、[次へ >>] などで表示位置を最後に変更してください。

ページ内表示数 << 前へ 次へ >>

3.5. RESTful Web サービスのテスト

1) デployした RESTful Web サービスをテストするための COBOL アプリケーションを生成する

- ① 「BOOKREST」を右クリックし、コンテキストメニューから「クライアント生成...」を選択します。

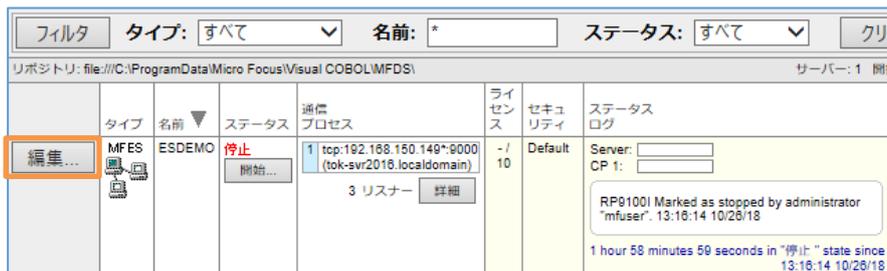


- ② 「クライアントを生成」ダイアログが表示されるので「ナビゲータ上の現在のプロジェクト」を選択し、[OK] ボタンをクリックします。



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

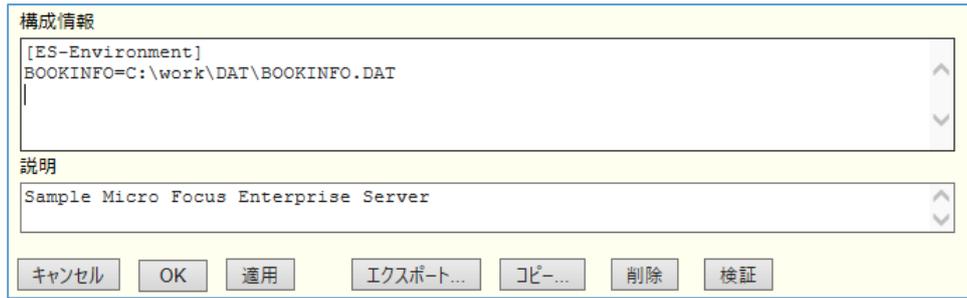
- ① 「サーバーエクスプローラー」に切り替えます。
- ② 「ESDEMO」上で右ボタンをクリックし、コンテキストメニューから「停止」を選択し、Enterprise Server を停止します。
- ③ 「Enterprise Server Administration」コンソール画面より「名前」列が「ESDEMO」の [Home] > [編集] ボタンをクリックします。



- ④ [サーバー] > [プロパティ] > [一般] タブの [構成情報] に以下の値を設定します。

[ES-Environment]

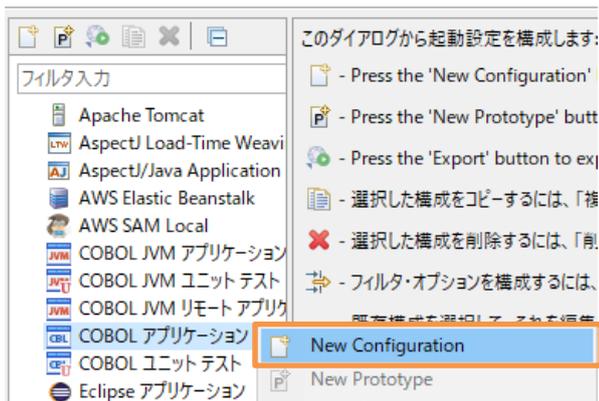
BOOKINFO=C:\work\DAT\BOOKINFO.DAT



- ⑤ [OK] ボタンをクリックし、画面を閉じたら、[サーバーエクスプローラー] に切り替えて、「ESDEMO」上で右ボタンをクリックし、コンテキストメニューから [開始] を選択し、Enterprise Server を起動します。
- 3) 生成したテスト用 COBOL クライアントの実行（登録処理）
- ① COBOL エクスプローラーにて「NativeCOBOL」プロジェクトを右クリックし、コンテキストメニューから [実行] > [実行の構成] を選択します。
 - ② 左側より「COBOL アプリケーション」を右クリックし、コンテキストメニューから [New Configuration] を選択します。

構成の作成、管理、および実行

COBOL プログラムを実行します



- ③ [一般] タブの 以下の入力を行い、[実行] ボタンをクリックします。
[名前] にて “NativeCOBOL-client”
[主プログラム] にて “New_Configuration.bin/BOOKREST-app.gnt”
- ④ 「Service Address」は、今回変更不要のため、そのまま Enter キーを押します。

補足)

Web サービスの要求を受け付けるリスナーが以下のようにポート 9003 でリスンしているため、変更せず、クライアントの実行が行えます。このポートは、[編集] ボタンをクリックした後の画面にて任意に変更できます。変更した場合は、クライアント実行時に変更後のポートを指定する必要があります。

編集...	Web Services and J2EE	tcp:192.168.56.1*:9003 (192.168.56.1)	開始
-------	-----------------------	---------------------------------------	----

- ⑤ 「Username」、「Password」は変更しないのでそのまま 2 回 Enter を押します。
- ⑥ 「Operation (1 = SEARCHBOOK, 2 = ADDBOOK)」は、“2” を入力して Enter を押します。
- ⑦ 「lnk_B_TITLE」は “PLANET OF THE APES” を入力して Enter を押します。
- ⑧ 「lnk_B_TYPE」は “SCIENCE FICTION” を入力して Enter を押します。

- ⑨ 「lnk_B_AUTHOR」は "PIERRE BOULLE" を入力して Enter を押します。
- ⑩ 「lnk_B_STOCKNO」は、"5555" を入力して Enter を押します。
- ⑪ 「lnk_B_RETAIL」は、"1000"、「lnk_B_ONHAND」は、"3000"、「lnk_B_SOLD」は、"2333" を入力して Enter を押します。
- ⑫ RESTful の Web サービスが実行されます。「lnk_FILE_STATUS」に "00" が返ってきたら登録成功です。

```

ca. "C:\Program Files (x86)\Micro Focus\Visual COBOL\bin\run.exe" "C:\S
Service Address (Enter = http://localhost:9003):
Username (optional):
Password (optional):
Operation (1 = SEARCHBOOK, 2 = ADDBOOK): 2
LNK_B_DETAILS:
.LNK_B_TEXT_DETAILS:
..LNK_B_TITLE: PLANET OF THE APES
..LNK_B_TYPE: SCIENCE FICTION
..LNK_B_AUTHOR: PIERRE BOULLE
.LNK_B_STOCKNO: 5555
.LNK_B_RETAIL: 1000
.LNK_B_ONHAND: 3000
.LNK_B_SOLD: 2333
LNK_FILE_STATUS: 00
続行するには何かキーを押してください . . .

```

4) 生成したテスト用 COBOL クライアントの実行 (検索処理)

- ① COBOL エクスプローラーにて「ネイティブ COBOL」プロジェクトを右クリックし、コンテキストメニューから [実行] > [実行の構成] を選択します。
- ② 左側の [COBOL アプリケーション] 配下の「NativeCOBOL-client」を選択し、[実行] ボタンをクリックします。
- ③ 「Service Address」は、デフォルトのまま使いたいので Enter キーを押します。
- ④ 「Username」、「Password」は変更しないのでそのまま 2 回 Enter を押します。
- ⑤ 「Operation (1 = SEARCHBOOK, 2 = ADDBOOK)」は、"1" を入力して Enter を押します。
- ⑥ 「lnk_FUNCTION」は、"5555" を入力して Enter を押します。
- ⑦ RESTful の Web サービスが実行され、データが返ってきます。以下のような内容が確認できます

```

ca. "C:\Program Files (x86)\Micro Focus\Visual COBOL\bin\run.exe" "C:\Seminar
Service Address (Enter = http://localhost:9003):
Username (optional):
Password (optional):
Operation (1 = SEARCHBOOK, 2 = ADDBOOK): 1
LNK_B_STOCKNO: 5555
LNK_B_DETAILS:
.LNK_B_TEXT_DETAILS:
..LNK_B_TITLE: PLANET OF THE APES
..LNK_B_TYPE: SCIENCE FICTION
..LNK_B_AUTHOR: PIERRE BOULLE
.LNK_B_STOCKNO: 5555
.LNK_B_RETAIL: 01000
.LNK_B_ONHAND: 03000
.LNK_B_SOLD: 02333
LNK_FILE_STATUS: 00
続行するには何かキーを押してください . . .

```

3.6. インスタンスの停止

- 1) Enterpiser Server の停止
 - ① 「サーバーエクスプローラー」に切り替えます。
 - ② 「ESDEMO」上で右クリックし、コンテキストメニューから [停止] を選択し、Enterprise Server を停止します。

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。

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。