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 RESTful 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 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 にディプロイし、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 11 Professional Edition(64bit) COBOL 開発環境製品 Micro Focus Visual COBOL 8.0 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

D MICRO[®] Focus

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ィプロイ先変更
 - 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 ヘディプロイ
 - 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
 - [スタート] メニュー > [すべてのアプリ] > [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。

Eclipse ランチャー		Х
ディレクトリーをワークスペースとして選択		
Eclipse uses the workspace directory to store its preferences and development artifacts.		
ワークスペース(<u>W)</u> : C:¥Work¥RESTful	~	参照(<u>B</u>)
□この選択をデフォルトとして使用し、今後この質問を表示しない(<u>U</u>)		
▶ 最近のワークスペース(<u>R</u>)		
起	助(<u>L</u>)	キャンセル

- 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
 - [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。※テンプレートは 32 ビットで構いません。
 - ② 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] > [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。

ゴンポート	
選択 ローカル・ファイル・システムから既存のブロジェクトへリソースをインボートします。	Ľ
<u>S</u> elect an import wizard: ブルタスカ	
 ← 一般 ・ アーカイブ・ファイル ・ アーカイブ・ファイル ・ ファイル・システム ・ ファイル ・ ・システム ・ ファイル・システム ・ ファイル・システム ・ ファイル・システム ・ ・システム ・ ・・・・・・・・・・・・・・・・・・・・・・・	^
途中省略	
? < 戻る(B) 次へ(N) > 終了(E)	キャンセル

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。



[四] インボート		_		Х
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。				
次のディレクトリーから(Y): C.¥Work		~	参照(<u>R</u>)	
> 🔳 🔁 Work	BOOK-INFO.cpy BOOK.cbl BOOKSCRN.cbl			

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。

ե	С	23	ዬ ታ	28	Α	<u>_</u> t	,	
				~	E	ل	$\downarrow^a_{\mathbf{Z}}$	₽ @
\sim	e	Nat	tiveCOB	OL				
	~	Þ	COBOL	プロク	156			
		>	🖻 BOC	OK.cb	ol 👘			
	~		コピーファ	11				
			BOC	DK-IN	IFO.c	ру		

- 3) ビルドオプションの変更
 - 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、コンテクストメニューから [プロパティ]を選択します。
 - 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。

✓ Micro Focus ビルダー ビルドパス	New Conngulation [IC/ID/T]	* (第1934
 ✓ ビルド構成 > COBOL イベント ビルド環境 	74ルタテキストを入力	
> リンク > プロジェクト設定	設定	値
> COBOL	出力の名前	NativeCOBOL
ビルド環境	出力パス	New Configuration.bin
指令の確定	エントリポイント	
> 実行時構成	ターゲットの種類	すべて INT/GNT ファイル
MibiTevt	17 1 444	22.1



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、[適用して閉じる] ボタンをクリックします。

✓ Micro Focus ビルダー ビルドパス	✓構成の固有な設定を可能にする(C)	
 ビルド構成 > COBOL イベント ディブロイ 	フルタテキストを入力	值
ビルト環境 > リンク	✓ 一般 文字セット	ASCII
 > ブロジェクト設定 指令の確定 	COBOL 方言 ソースフォーマット	Micro Focus 固定
≫ 実行時構成 WikiText		はい
タスク・タグ > タスク・リポジトリー	EXIT PROGRAM @ GOBACK 2 D C処理 詳細	いいえ
ビルダー プロジェクト・ネーチャー	.GNT にコンパイル ◆ 出力 .GNT にコンパイル:は	(はい い
プロジェクト・ファセット	指令ファイルを生成する リストファイルを生成	いいえ いいえ
	コードカバレッジを有効にする プロファイラを有効にする	false false
美行/デバック設定	✓ エラ-/警告 等告レベル	回復可能なエラーを含める(レベル E)
	最大IJ-数 ・ 2 fm 半へ	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。



- 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。

<mark>ို</mark> း Cobol 🔀	≌. ナビゲーター	Part Applica	💻 サーバー	·		
_			~ Ē ∳-	-/(- :	・ エクスプローラー	
V 🛃 NativeCOB	OL プロガニル					



② Visual COBOL にビルドインされている Enterprise Server 「ESDEMO」を右クリックし、コンテクストメニューから
 [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「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。
 - ✓ □-カル [localhost:10086]
 ✓ Default [127.0.0.1:86]
 ESDEMO
 ESDEMO64
- 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。
 - 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」を選択します。





③ [プリセットフィルター]タブの[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) ディプロイ用フォルダの作成
 - ① 「NativeCOBOL」プロジェクト上で右クリックし、コンテクストメニューから [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。

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。	
次のディレクトリーから(Y): C:¥Program Files (x86)¥Micro Focus¥Visual COBOL¥deploy ~	参照(R)
■ > deploy	
タイプをフィルター(T) すべて選択(S) 選択をすべて解除(D)	
インポート先フォルダ(L): NativeCOBOL/deploy	参照(W)

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、コンテクストメニューから [管理ページを開く] を選択します。

🔁 С 🕾 N 😤 А	黒 サ窓 [≫] 1 □ □	
	♥ 🖻 🖇	
✔ 🚳 ローカル [localh	ost:10086]	
🗸 🛃 Default [12]	7. 新規作成(N)	>
📜 ESDEMC		
🚪 ESDEMO	De 管理ベージを開く	

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。

ES	管理	5	ブッシュフ	ボード	ネイティン	C ES .NE	ד אראס	レーム セニ	キュリティー。	<i>P</i> 2, ∨ U4.0.9	9 S4.0.24
∨ ∰ク	リループ									wind	lows
> 1	俞理		▲ 🖨	このDirec	tory Serverホス	トでは TLS が有効	めではありませんが	、ループバック	トラフィックの受信	のみに制限されている	ます
> F	AC		リーシ	ジョン	およびサ・	ーバー リ	スト C	* 新規作成	💼 すべて)	削除 🔄 エク	スポート
v目D	irectory Server										
~ [🗄 🔓 Default	C	Ĺ Ŷ,	インボー			り更新				?
	🖪 ESDEMO	\triangleright									
	ESDEMO64										
> ∲ S	OR			7 2	前		▽ 説明		PAC		
				γд	ニンドポイント		タイプ 🗸 ステ・	ータス∨ 64	ビットマ MSS	3有効 🗸	
										四 表示列	히 🗸
				\$	乙前	タイプ	ステ	64ビ	MSS	セキ	
				Ē E	SDEMO	Region	Started			デフォルト	^
				Ξ. E	SDEMO64	Region	Stopped	\checkmark		デフォルト	
			4								* •

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] → [ESDEMO]をクリックします。

∨ ⊞	Directory Server	
~	🔃 덢 Default	
		\triangleright

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。



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、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。

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

① 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。

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 Eclipse IDE の 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、コンテクス 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。



REST Web サービスの新規作成 このページで REST Web サービスを新規作成します	₩
Web サービス名: BOOKREST マッピング: 〇 デフォルト	
マップするプログラム: NativeCOBOL/BOOK.cbl	参照

- 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、コンテクストメニューから [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。

一般	HTTP	ユーザ出口	
オペレー	ションは選	択されたエントリポイ	イントインターフェイスを使用して COBOL プログラムを起動するのに使用されます
オペレー	ション名:	SEARCHBOOK	
エントリ	ポイント:	BOOK	~

- 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 [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 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。

o co	DBOL 割当て	プロパティ	—		×
名前:	LNK-FUNC	TION		1	編集
值:	1				
		OK		キャンセ	.JL

⑤ 「SERCHBOOK オペレーション – インターフェイス フールド」にて右クリックし、コンテクストメニューからを [インターフェ イス グループの新規作成] をクリックします。



SEARCHBOOK 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 CTOC/200 ■ インターフェー	っ イス フィールドの新;	規作成
■ インターフェイ	イス グループの新規	化 作成
F ()		

⑥ [グループプロパティ] ウィンドウにて "input_root" と入力、[方向] を [入力] にして [OK] ボタンをクリックします。

名前:	input_root	
OCCURS	: 0	
方向:	О入力 ○出力	
場所:	О ボディ ○ パス ○ クエリ	
	ОК	キャンセル

- ⑦ 「SERCHBOOK オペレーション インターフェイス フールド」にて右クリックし、コンテクストメニューからを [インターフェ イス グループの新規作成] をクリックします。
- ⑧ [グループプロパティ] ウィンドウにて "output_root" と入力、[方向] を [出力] にして [OK] ボタンをクリックします。
- ⑨ [LINKAGE SECTION] 側にある「LNK_B_STOCKNO」を「input_root」にドラッグ&ドロップします。
- ⑩ [LINKAGE SECTION] 側にある「LNK-B-DETAILS」を「output_root」にドラッグ&ドロップします。
- ① 最後に [LINKAGE SECTION] 側にある「LNK-FILE-STATUS」を「output_root」にドラッグ&ドロップします。最終的には下のイメージのような構成になります。

SEARCHBOOK オペレーション - インタ	-71777-	ールド:		
名前	方向	型	OCCU	場所
✓ ☐ input_root	入力			ボディ
🛤 LNK_B_STOCKNO		string		
✓	出力			ボディ
✓ ➡ LNK_B_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		string		

- 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、コンテクストメニューから [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 「ADDBOOK オペレーション インターフェイス フールド」にて右クリックし、コンテクストメニューからを [インターフェイス



グループの新規作成]をクリックします。



④ [グループプロパティ] ウィンドウにて "input_root" と入力、[方向] を [入力] にして [OK] ボタンをクリックします。



- ⑤ 「ADDBOOK オペレーション インターフェイス フールド」にて右クリックし、コンテクストメニューからを「インターフェイス グループの新規作成」をクリックします。
- ⑥ [グループプロパティ] ウィンドウにて "output_root" と入力、[方向] を [出力] にして [OK] ボタンをクリックします。
- ⑦ [LINKAGE SECTION] 側にある「LNK-B-DETAILS」を「input_root」にドラッグ&ドロップします。
- ⑧ [LINKAGE SECTION] 側にある「LNK-FILE-STATUS」を「output_root」にドラッグ&ドロップします。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	OCCU	場所
✓	入力			ボディ
✓				
✓				
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		
✓	出力			ボディ
LNK_FILE_STATUS		string		

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

- 1) Enterprise Server へのディプロイ情報を指定
 - 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、コンテクストメニューから [プ ロパティ(P)] を選択します。



🔁 сов ಜ 🗞 ታ	ピグ	🧾 サーバ			Nookrest 🔀
		E	€ 5	\bigtriangledown	LINKAGE SECTION:
⊿ 🖉 NativeCOBOL					名前
⊿ 📠 COBOL 九	バラム				LNK-FUNCTION
▶ BOOK.¢	bl				INK-B-DETAILS
▶ a BOOKS	CRN.c	bl			LNK-FILE-STATUS
⊳ 🔑 ⊐ピーファイル					
🛛 🔄 Web サービ	z				
⊿ 😻 BOOKR	FST				
-🗟 SEA	я	新規作成(N)			•
-뎒 ADI	X	削除(D)			
▷ ▷ New_Cont ▷ ▷ repos	5	^ו ם/(ד₁(P)			

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。

ディプロイメントサーバー	アプリケーションファイル	エグジット ポイント ハンドラー	CORS プロパティ		
Enterprise Server 名					
ESDEMO (localhost:50240) 変更				変更	
Contraction Contract	- 史仁味理培の体田				

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。

ディプロイ先の Er	nterprise Serv	erを選択してくた	รัชเง:		
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

- 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ィプロイする」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。

ディプロイメントサーバー	アプリケーションファイル	エグジット ポイント ハンドラー	CORS プロパティ		
レガシーアプリケーション	をディプロイ済みか、またし	はサーバーにディプロイする必要カ	があるかを指定してください。		
○ レガシーアプリケーシ ディプロイされたアご	ョンは既にディプロイ済み プリケーションのパス:				
● レガシーアプリケーシ アプリケーションファイル	● レガシーアプリケーションをディプロイする アプリケーションファイル:				
New_Configuratio New_Configuratio	n.bin/BOOK.gnt n.bin/BOOK.idy			ファイル追加 ファイル削除	

- 2) オリジン間リソース共有 (CORS) を許可
 - ① デフォルトではオリジン間リソース共有は許可されていません。もしこれに関するエラーが発生する場合、許可設定を行い ます。
 - ② 「BOOKREST」Web サービスを選択して、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。
 - ③ [CORS プロパティ]タブを選択し、運用用途に合わせた設定を行います。ここでは「アクセス制限チェックなしでクロスオリ ジンでリソースのフェッチを許可する(*)」を選択しています。



ディプロイメントサーバー	アプリケーションファイル	エグジット ポイント ハンドラー	🛞 CORS プロパティ
オリジンを許可する			
○ クロスオリジン要求	を許可しない(null)		
● アクセス制御チェッ	クなしでクロスオリジンで!	リソースのフェッチを許可する (*)	
○ 特定のオリジンから	らクロスオリジンでリソースの	のフェッチを許可する	
Allowed Origin:			

- ④ [OK] ボタンをクリックします。
- 3) RESTful Web サービスを Enterprise Server ヘディプロイ
 - 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、コンテクストメニューから [検査] を選択し、エラーが出ていないことを確認します。
 - ② 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、コンテクストメニューから [ディプロイ] を選択します。

⊿ 🕞 Web サー	UZ ZDCC	T	
-E S		新規作成(N)	•
-E A	×	削除(D)	
D Dev Contractor New Contractor N		プロパティ(P)	
		รังวับง	
_		検査	



- ③ ブラウザ上の「ESCWA サーバー」に切り替えます。
- ④ [一般]メニューから[サービス]をクリックします。
- ⑤ 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 RESTful Web サービスが追加されていることを確認します。

෯	✓ /temppath/BOOKR					
ଞ	#SEARCHBOOK	Available	Web	/temppath/BOOKREST/1.0	MFRHJSON	created 10:
ଞ	#ADDBOOK	Available	Web	/temppath/BOOKREST/1.0	MFRHJSON	created 10:

3.5. RESTful Web サービスのテスト

- 1) ディプロイした 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、コンテクストメニューから [クライアント生成...]を選択します。

✓ 100 Web サービス	ζ ΓCT		
		新規作成(N)	>
-🛱 SEA	×	削除	削除
> 🗁 repos	1	プロパティ(P)	
mccerror.b	c	รังวัยง	
		検査	
		開く	
		クライアント生成	

「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。

<u></u>	
◉ ナビゲータ上の現在のプロジェクト	
○ 開いている Eclipse プロジェクトを選択	
○ COBOL プロジェクトを新規作成	
生成されたクライアント用のプロジェクト: NativeCOBOL	選択
出力ディレクトリ: C:¥Work¥RESTful¥NativeCOBOL¥client	参照

- 2) Enterprise Server がアクセスするデータファイルを指定
 - ① ブラウザ上の ESCWA 画面に切り替えます。
 - ② [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。※ファイルはあらかじめこのディレクトリに必ずコピーしておいてください。

[ES-Environment]

BOOKINFO=C:¥work¥DAT¥BOOKINFO.DAT

追加設定

· # □· / = #2 ○	
BOOKINFO=C:\work\DAT\BOOKINFO.DAT	

③ 上にスクロールし、[適用]ボタンをクリックします。



?	一般			₹ 二 9- ~	·	&⊐-
一般的	はプロノ	(ティ	С	適用	茴 削除	
884/1						

④ 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。



- 3) Enterpriser 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。
 - [ESDEMO] 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。
- 4) 生成したテスト用 COBOL クライアントの実行(登録処理)

構成の作成、管理、および実行 COBOL プログラムを実行します

- 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、コンテクストメニューから [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、コンテクストメニューから [新規構成] を選択します。

📑 🖻 🐢 🖹 🖛 🏹 -	このダイアログから起動
7ィルタ入力	📑 - 選択した種類(
🗄 Apache Tomcat	P - 選択した種類(
AspectJ/Java Application	🖚 - 選択した構成
COBOL JVM アプリケーション	📄 - 選択した構成
COBOL JVM ユニット テスト	🛛 🗙 - 選択した構成:
INNA COBOL JVM リモート アノリク GEL COBOL アプリケーション	V-7/1.4.+12
😋 COBO 🎦 新規構成(W)	ļ
😧 Debug 📄 新規プロトタイ	プ(P)

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、Enter 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ポート 9003 でリッスンしているため、変更せず、クライアントの実行が行えます。このポートは、[リスナー]メニュー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。



- ⑤ Supplemental Query String (Optional) は変更しないのでそのまま Enter を押します。
- ⑥ 「Username」、「Password」は変更しないのでそのまま 2 回 Enter を押します。
- ⑦ 「Operation (1 = SEARCHBOOK, 2 = ADDBOOK」は、"2" を入力して Enter を押します。
- ⑧ 「lnk_B_TITLE」は "PLANET OF THE APES" を入力して Enter を押します。
- ⑨ 「lnk_B_TYPE」は "SCIENCE FICTION" を入力して Enter を押します。
- ⑩ 「lnk_B_AUTHOR」は "PIERRE BOULLE" を入力して Enter を押します。
- ⑪ 「lnk_B_STOCKNO」は、"5555" を入力して Enter を押します。
- ② 「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。
- ③ 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。



- 5) 生成したテスト用 COBOL クライアントの実行(検索処理)
 - 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、コンテクストメニューから [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

Service Address (Enter = http://localhost:9003):
Supplemental Query String (optional):
Username (optional):
Password (optional):
Operation (1 = SEARCHBOOK, 2 = ADDBOOK): 1
Body Parameters:
.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

- 6) Enterpiser Server の停止
 - ③ 「サーバーエクスプローラー」に切り替えます。
 - ④ 「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、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。