Micro Focus Enterprise Developer チュートリアル

メインフレーム COBOL 開発: IMS

Visual Studio 2012 編

4. 目的

本チュートリアルでは、Visual Studio 2012 を使用したメインフレーム COBOL プロジェクトの作成、コンパイル、IMS を使用したトランザ クションの実行、デバッグまでを行い、その手順の習得を目的としています。

5. 前提

- 本チュートリアルで使用したマシン OS : Windows 8 Enterprise
- 使用マシンに Micro Focus Enterprise Developer 2.3 for Visual Studio 2012 がインストールされていること
- 使用マシンに TN3270 エミュレータがインストールされており、稼働実績があること

6. チュートリアル手順の概要

- 1. チュートリアルの準備
- 2. Enterprise Developer の起動
- 3. メインフレーム COBOL プロジェクトの作成
- 4. プロジェクトプロパティの設定
- 5. ビルドの実行
- 6. リソース定義ファイルの更新
- 7. IMS データベースのロードおよびトランザクションリスト設定
- 8. Enterprise Server の設定
- 9. Enterprise Server の開始と確認
- 10. IMS 資源定義の確認
- 11. IMS トランザクションの実行
- 12. プログラムの動的デバッグ
- 13. Enterprise Server の停止



3.1 チュートリアルの準備

例題プログラムに関連する資源を用意します。

1) 使用する例題プログラムは、キットに添付されている IMStutorial.zip に圧縮されています。これを C:¥ 直下に解凍します。



2) Visual Studio のソリューションを保存する「VS」フォルダーを C:¥ 直下に作成します。

3.2 Enterprise Developer の起動

1) Micro Focus Enterprise Developer for Visual Studio 2012 を起動します。

ME Enter	prise Developer for Visual Studio 2012	Å	Adobe Reader XI
🔤 1 [°] ‡1)	ント - Eclipse		IBM WebSphere MQ
😮 אב‡ז 😮	ント - Visual Studio 2012		Micro Focus Enterprise Developer

3.3 メインフレーム COBOL プロジェクトの作成

1) 新しいソリューションとプロジェクトを作成します。 [ファイル] プルダウンメニューから [新規作成] > [プロジェクト] を選択して [新しいプロジェクト] ウィンドウを表示し、下記項目を指定後 [OK] ボタンをクリックします。

項目名	説明
左側メニュー	[インストール済] > [テンプレート] > [COBOL プロジェクト] > [Native] を選択します。
中央リスト	[メインフレーム サブシステム アプリケーション] を選択します。
名前	任意ですが、ここでは "IMSDEMO" を入力します。
場所	前項で作成した "C:¥VS" を指定します。
ソリューションのディレクトリ ーを作成	ここではチェックをオフにします。

	新しいプロジェクト		? 🗙
▶ 最近使用したファイル	.NET Framework 4.5 · 並べ替え基準: 民定	-	₩ 🗄 インストール済み テンプレート の検診 ρ・
▲ インストール済み	Enterprise Server アプリケーション	COBOL プロジェクト	種類: COBOL プロジェクト
▲ デンプレート ▲ COBOL プロジェクト Cloud	■二 CBL Windows アプリケーション	COBOL プロジェクト	Micro Focus Server 配下で実行される CICS®、JCLまたは IMS® アプ・ケーションを 作成するためのプロジェクトです。
Database	📑 🚆 メインフレーム サブシステム アプリケーション	COBOL プロジェクト	
Native Web	こ コンソール アプリケーション	COBOL プロジェクト	
▷ Visual C#	リンク ライブラリ	COBOL プロジェクト	
LightSwitch ト 他の童語	50 室のプレジェクト	COBOL プロジェクト	
♦ Other Project Types サンプル	Micro Focus INT/GNT	COBOL プロジェクト	
▶ オンライン	Web Service Client Application from	COBOL プロジェクト	
名前(N): IMSDEMO			
場所(L): c:¥VS			参照(日)
9月1-5月2名(M): IMSDEMO		L	1 901-5820770719841B(D)
			OK キャンセル

2) [ソリューション エクスプローラー] へ作成したプロジェクトが表示されます。

▲ IMSDEMO
🔑 Properties



 プロジェクトを作成したことにより C:¥VS¥IMSDEMO フォルダーが作成されていますので、このフォルダー直下に C:¥VS¥IMSDEMO¥system フォルダーを作成します。前項で作成した C:¥IMStutorial フォルダーにある [dfhdrdat] ファイルを C:¥VS¥ IMSDEMO¥system ヘコピーします。このファイルには IMS で使用するリソース定義 が含まれています。

ディスク (C:) → IMStutorial		(C:) ► VS	▶ IMSDEMO → system
名前		名前	^
dfhdrdat	\rightarrow	📄 dfhdrda	at

4) 既存ファイルのインポート時、自動的にコンパイル指令が指定される機能が用意されていますが、本チュートリアルではこれを 解除します。 [ツール] プロダウンメニューの [オプション] を選択してオプションウィンドウを表示します。

左側メニューの [Micro Focus] > [指令] > [COBOL] > [ファイルの指令の確定] チェックボックスをオフにして [OK] ボタンをクリックします。

MICRO FOCUS	
±nx Enterprise Server	
HCO for SQL Server Tools	COBOL 方言
OpenESQL Assistant	✓ ファイルに DIALECT 指令を設定する(A)
TN3270 Display	COBOL SQL
Web Service Clients	✓ EXEC SQL を含むファイルに SQL 指令を設定する(S) …
▷ コードカバレッジ	✓ EXEC SQL を含まないファイルの SQL 指令の設定を解除する(Q)
サービスインターフェイス	
テータファイル ツール	▼ EXEC DLI を含むファイルに EXECDLI 指令を設定する(L)
▷ ノロノバ1フ 、 当会	EVEC DLT を含またいつっていの EVEC DLT 場合の設定を紹発する(1)
COBOL	
PL/I	COBOL CICS
単一ファイルのコンパイル	
> Office ツール	▼ EXECCICS (E3.400)/1/0/ NOCICS 1976(10)/2/00/10/
。SQL Server ツール	
Web パフォーマンス テスト ツール	■ EXECULS を含むJP1/Nc ULS 指令を設定する(E)
Windows フォーム デザイナー	v

5) 用意したサンプルプログラム類をインポートします。 [IMSDEMO] プロジェクトを右クリックして [追加] > [既存の項目] を 選択し、既存項目の追加ウィンドウにて C:¥IMStutorial を指定すると内容が表示されますので、[すべてのファイル] を 対象にして、[dfhdrdat] ファイル以外を選択後 [追加] ボタンをクリックします。この実行により、プロジェクトフォルダーへサ ンプルプログラムが配置されます。

M	既存項目の追加 - IMS	SDEMO		×	
⊛ ∋ ▼ ↑ 퉬 « 🛛	ーカル ディスク (C:) → IMStutorial	v C IMS	tutorialの検索	Q	
整理 ▼ 新しいフォルダー			-		
^	名前	更新日時	種類	サイズ	
	DEMO001T.CBL	2012/01/23 17:40	CBL ファイル	72	
Supversion	DEMO001T.PSB	2008/01/03 5:36	PSB ファイル	1	
P+1X2P	DEMO03DD.DBD	2012/01/23 17:40	DBD ファイル	2	
	DEMO03DD.DBU	2006/03/31 21:35	DBU ファイル	10	
	DEMO90.MFS	2012/05/24 13:49	MFS ファイル	3	
	DEMO91.MFS	2012/05/24 13:49	MFS ファイル	5	
A + 1 B + 7	DEMO92.MFS	2012/01/23 17:40	MFS ファイル	17	
₩ <u>₩</u> -14970-2	📄 dfhdrdat	2013/12/17 12:38	ファイル	141	
📕 TNRA – b	RGHTJUST.CBL	2012/01/23 17:40	CBL ファイル	3	
	TRANCODE.TXT	2006/04/04 23:27	テキスト ドキュメント	1	
₩ CD 1 717 (D-) V <					
ファイノ	レ名(N): "TRANCODE.TXT" "DEMO001T.CBI	_" "DEMO001T. 🗸 すべ	てのファイル (*.*)	~	
	L	追	加(<u>A)</u> 本	Pンセル:	



6) [ソリューション エクスプローラー] 内に表示されている [IMSDEMO] にインポートしたファイルが表示されていることを確認 します。



3.4 プロジェクトプロパティの設定

このサンプルには COBOL プログラム, DBD, MFS, PSB ファイルが含まれており、プログラム内容に沿ったプロジェクトのプロパティを設定します。

- 1) [ソリューション エクスプローラー] 内の [Properties] をダブルクリックしてプロパティウィンドウを表示します。
- 2) 左側メニューの [アプリケーション] を選択して、生成する実行ファイルを GNT にするため [出力の種類] へ [INT/GNT] を選択します。

アプリケーション	構成(C)· 該当村,
SQL	149/44(<u>C</u>).
従属パス	114 0 7
BMS	出力の名前: CICSDEMO
IMS	出力の種類:
Preprocessors	INT/GNT

	情報					
"GN ⁻ COB	T″は OL ラ	Micro Focus ンタイム環境下・	独自のオブラで実行可能と	ジェクトで、N こなります。	Aicro Foc	us

3) 左側メニューの [COBOL] を選択して、下記項目を入力します。

項目名	説明
プラットフォーム ターゲット	稼働ビット数を指定します。ここでは [x64] を指定します。
COBOL 方言	COBOL 言語方言を指定します。 サンプルプログラムはIBM Enterprise COBOL の方言を使用しているため、ここで は [Enterprise COBOL for z/OS] を指定します。
文字セット	EBCDIC または ASCIIを指定します。ここでは [ASCII] を選択します。
デバッグ用にコンパイル	デバッグ実行時に使用するファイルを生成するように指定します。
.GNT にコンパイル	実行ファイル形式を GNT に指定するためにチェックをオンにします。
出カパス	実行ファイルが出力されるパスを指します。任意に指定可能です。
追加指令	ここでは何も指定しません。

501	構成(<u>C</u>): アクティブ (Debug) v プラットフォー	ム(<u>M</u>): アクティブ (x8	6) ~	
従属パス	全般				
BMS	プラットフォームターゲット:	v64 v			
IMS	· · · · · · · · · · · · · · · · · · ·	X07 Y			
プリプロセッサ	COBOL 方言:	Enterprise COBOL for	z/OS 🗸	文字セット:	ASCII
COBOL	ソース フォーマット:	固定	~		
COBOL リンク	✓ デバッグ用にコンパイル	,	✓ .GNT にコンパ	YJL	
アセンブラ					
デバッグ	ビルドイベント				
Micro Focus Code Analysis	メインフレーム				
	EXEC CICS	オプション	EXEC DLI	オプション	
	エラーおよび警告				
	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	□ 警告をエラーと	して処理	
	出力				
	出力パス:	.¥bin¥x86¥Debug¥			参照
	□ 指令ファイルの生成		□ リストファイルを	生成	
	コードカバレッジを有効	めにする	🗌 プロファイラを有	効にする	
	10 to the CA				

4) 左側メニューの [IMS] を選択して、IMS 資産をビルドした際に生成されるファイルの出力パスを確認します。デフォルトはソ リューションが存在するパスになっていますが、任意に変更することも可能です。

従属パス	出力フォルダ	
BMS	生成ファイル パス: .¥	
IMS		
プリプロセッサ	MFS 出力パス: .¥	
COBOL	データベース パス: .¥	
COBOL リンク	構成ファイル パス: .¥	
アセンブラ		
$\pm m_0 R$	 9へ(の) DBD/PSB 生成 JP1ルも削除9る 	

5) 左側メニューの [IMS] を選択して表示されている [DBD ビルド設定] の [DBD のマップを生成] チェックをオンにしま す。

DBD ビルド設定		
メッセージ レベル:	警告	~
🗌 リストファイルを生成		
✓ DBD のマップを生成		

6) 左側メニューの [デバッグ] を選択して、[アクティブ設定を選択] へ [IMS] を選択します。

7) プロパティファイルを上書き保存します。

ファイル(F)	編集(E)	表示(V)
G - O	8 - 省	1



3.5 ビルドの実行

- 1) [ソリューション エクスプローラー] の [IMSDEMO] ソリューションを右クリックして [ソリューションのビルド] を選択すると、コ ンパイル指定に沿ったビルドが実行されます。
- 2) [出力] ウィンドウで成功を確認します。



 前項で確認した出カパスへ実行ファイルに指定した ".gnt" ファイルが作成され、ACB, DIF, DOF, MAP, MFSX, MID, MOD や DBDGEN2.DAT, DBDGEN2F.DAT, PSBGEN3.DAT などの IMS 関連生成ファイルはソリューションと同じ フォルダーへ出力されています。



3.6 リソース定義ファイルの更新

IMS サポート機能は CICS モードと IMS モードを切り替える "/CIC" と "/IMS" 2 つのトランザクションを含んでおり、これ らを利用できるように caspcupg コマンドを使用してリソース定義ファイルを更新する必要があります。

Windows のプログラムメニューから [Micro Focus Enterprise Developer] > [ツール] > [Enterprise Developer コマンドプロンプト(64-bit)] を右クリックして [管理者として実行] を選択します。

📷 Enterprise Developer コマンドプロンプト (64-bit)	管理者として実行(A)

2) 表示されたコマンドプロンプトから、リソース定義ファイルのパスを指定した下記コマンドを実行して、[return-code=0000] となることを確認します。

コマンド) caspcupg/dp=C:¥VS¥IMSDEMO¥system C:¥Users¥tarot¥Documents>caspcupg /dp=C:¥VS¥IMSDEMO¥system ↓ CASRB0025I Upgrade log file: C:¥VS¥IMSDEMO¥system¥rdoupgrd.log CASRB0056I Processing completed with return-code=0000



コマンドの詳細に関しては下記アドレスを参照してください。

http://documentation.microfocus.com/help/topic/com.microfocus.eclipse.infocenter.enterprisedeveloper.vs2 012/HRMTRHCOMM23.html

3.7 IMS データベースのロードおよびトランザクションリスト設定

MFIMS コマンドを使用して、IMS データベースのロードと IMS トランザクションリストの設定を行います。

1) 前項で使用したコマンドプロンプトで、作成した COBOL プロジェクトのパスへ移動します。

C:¥Users¥tarot¥Documents>cd C:¥VS¥IMSDEMO C:¥VS¥IMSDEMO>_

2) サンプルに含まれている IMS データベースをロードするために、下記のコマンドを実行して成功を確認します。

コマンド) MFIMS IMSDBU LOAD DEMO03DD NOCLS

C:¥VS¥IMSDEMO>MFIMS IMSDBU LOAD DEMO03DD NOCLS

 \downarrow

15.11.10/11:08:01 DEMO03DD LOAD completed successfully

コマンドにより下記ファイルが作成されます。

DEMO03DD.DAT	2015/11/10 11:08	DAT ファイル
DEMO03DD.LST	2015/11/10 11:08	MASM Listing
IMSDBU.LOG	2015/11/10 11:08	テキスト ドキュメント

コマンドの詳細に関しては下記アドレスを参照してください。

http://documentation.microfocus.com/help/topic/com.microfocus.eclipse.infocenter.enterprisedeveloper.vs2 012/GUID-3C4AE8AF-8742-45F3-95C4-1BA495BB91A0.html?resultof=%22%6d%66%69%6d%73%22%20 %22%69%6d%73%64%62%75%22%20%22%63%6f%6d%6d%61%6e%64%22%20

3) サンプルに含まれている TRANCODE.TXT ファイルに記述されている IMS トランザクションのリストを設定するために、次のコマンドを実行します。

コマンド) MFIMS STAGE1IMP TRANCODE.TXT





コマンドにより下記ファイルが作成されます。

🖺 IMSGEN2.DAT	2015/11/10 11:18	DAT ファイル
STG1IMP.LST	2015/11/10 11:18	MASM Listing



コマンドの詳細に関しては下記アドレスを参照してください。

http://documentation.microfocus.com/help/topic/com.microfocus.eclipse.infocenter.enterprisedeveloper.vs2 012/GUID-42AFF73E-E5A0-40AD-AB1B-32AC2F32D772.html?resultof=%22%6d%66%66%6d%63%22%20 %22%73%74%61%67%65%31%69%6d%70%22%20%22%63%6f%6d%6d%61%6e%64%22%20

3.8 Enterprise Server の設定

Enterprise Server には CICS や IMS をエミュレーションする機能が搭載されており、この開発用サーバーを使用してメイン フレームアプリケーションの実行やデバッグを行います。マイグレーションにおいては本番実行用の Enterprise Server 製品を使用します。

1) Enterprise Server を作成します。 [サーバー エクスプローラー] タブの [Micro Focus Servers] を右クリックして [管理] を選択します。



2) Enterprise Server Administration 画面へ遷移して、Enterprise Server 一覧が表示されますので、画面の左下 にある [追加] ボタンをクリックします。



サーバー追加 (Page 1 of 3):

3) サーバー名には [IMSDEMO] を入力、動作モードは 64-bit を指定して [次へ] ボタンをクリックします。

サーバー名: IMSDEMO	
動作モード:	
○ 32-bit	
You cannot change your choice of wor	
キャンセル 次へ>>	
● 重要	
実行ファイル生成に指定した稼働ビット数 = Enterprise	Server 稼働ビット数である必要があります。

 画面の Page 2/3 では、CICS や JCL を実行可能な機能を持つ [Micro Focus Enterprise Server with Mainframe Subsystem Support] が選択されていることを確認後、[次へ] ボタンをクリックします。

1	サーバータイプ:		
	0	MFES	Micro Focus Enterprise Server An enterprise server that provides an execution environment for COBOL application programs running as services in a service orientated architecture.
	۲		Micro Focus Enterprise Server with Mainframe Subsystem Support An enterprise server that also provides an execution environment for CICS applications that have been migrated from the mainframe.



5) Page 3/3 では [TN3270 リスナーの作成] のチェックがオンであることを確認して、[Using port] へ "5039" を指定 します。[追加] ボタンをクリックすると、[IMSDEMO] という名前の 64 ビットアプリケーション稼働用 Enterprise Server が追加されます。 "IMS リージョン" と同意語です。

4	上成オプション: 編集 MFE3	IMSDEMO	<u>停止</u> 開始
	TN3270リスナーの作成 🗹 using port 5039 →		
	• 情報		
	ポート番号には 1025 より小さい番号を指定しないでください。		
	指定したポート番号で TN3270 リスナーが作成されます。		

6) 左にある [編集] ボタンをクリックします。



- 7) [サーバー] > [プロパティ] > [一般] タブ内の下記項目を設定します。
 - ① [動的デバッグを許可] チェックボックスをオンにします。この指定により、Visual Studio からの動的デバッグが可能に なります。

開始オプション:	
共有メモリページ数: 512	サービス実行プロセス: 2
共有メモリクッション: 32	要求ライセンス: 10
ローカルコンソールを表示: 🗌	動的デバッグを許可: 🔽
Start on System Start:	64-Bit Working Mode:
以前のログを削除: 🗌	コンソールログサイズ (K): 0

② [構成情報]欄にパス変数と、IMS マップの日本語半角カナを有効にするため下記内容を入力します。



③ [Apply] ボタンをクリックします。



8) [サーバー] > [プロパティ] > [MSS] > [CICS] タブで表示される画面の各項目を設定します。入力後は [Apply] ボ タンをクリックします。

項目名	説明
メインフレーム サブシステム サポート有効	[MSS] タブ配下の設定をオン、オフ指定します。ここではオンを指定します。
システム初期化テーブル(SIT)	IMS リージョン設定の詳細が提供されるシステム初期化テーブルを指定します。ここではサンプルに含まれている "IMSSIT" を指定します。
トランザクションパス	実行プログラムの探索パスを指定します。ここでは指定しません。
File Path	データセットのデフォルトパスを指定します。ここでは指定しません。
マップパス	コンパイル済み BMS マップセットのパスを指定します。ここでは指定しません。
リソース定義ファイルパス	リソース定義ファイルが存在するパスを指定します。ここでは前項で作成した "\$IMSPROJ¥system"を指定します。

メインフレーム サブシステム サポート有効: 🔽 1

CICS (✔)	JES	IMS	PL/I
CICS 有効:	~		
システム初期 IMSSIT	líヒテーブル	(SIT):	
トランザクション	ンパス:		
File Path:			
マップパス:			
リソース定義が	ファイルバス	:	
\$IMSPROJ\system			
構成情報			
[ES-Environment]			
MFCODESET=9122			

🛞 重要

入力値は全て半角英数字で指定してください。

これらのフィールドでは改行を入れないように注意してください。



9) [サーバー] > [プロパティ] > [MSS] > [IMS] > [General] タブで表示される画面の各項目を設定します。入力後は [Apply] ボタンをクリックします。

項目名	説明
IMS 有効	[IMS] タブ配下の設定をオン、オフ指定します。ここではオンにします。
ACB ファイル ディレクトリ	ACB ファイルが存在するパスを指定します。ここでは "\$IMSPROJ" を指定します。
Gen ファイル ディレクトリ	GEN ファイルが存在するパスを指定します。ここでは "\$IMSPROJ" を指定します。
CICS (✔) JES IMS	
General DB TM	
IMS 有効: 🔽	
Codeset bias: ASCII 🗸	
Configuration directory:	
ACB ファイル ディレクトリ:	
\$IMSPROJ	(C:) → VS → IMSDEMO → (C:) → VS → IMSDEMO →
<mark>Gen ファイル ディレクトリ:</mark> \$IMSPROJ	名前 名前 DEMO001T.ACB ③ DBDGEN2.DAT DEMO03DD.ACB ⑤ DBDGEN2F.DAT

10) [サーバー] > [プロパティ] > [MSS] > [IMS] > [DB] タブで表示される画面の各項目を設定します。入力後は [Apply] ボタンをクリックします。

項目名	説明
データベース パス	IMS データベースが存在するパスを指定します。ここでは "\$IMSPROJ" を指定します。
CICS (✔) JES ĬIM 一般 DB TM 一般 圧縮 ロック データベースパス: \$IMSPROJ	<mark>S (✓)</mark> 復旧

11) [サーバー] > [プロパティ] > [MSS] > [IMS] > [TM] タブで表示される画面の各項目を設定します。入力後は [Apply] ボタンをクリックします。

項目名	説明
MFS path	MFS ファイルが存在するパスを指定します。ここでは "\$IMSPROJ" を指定します。
Application path	ビルドした実行ファイルが存在するパスを指定します。
	ここでは "\$IMSPROJ¥¥bin¥x86¥Debug" を指定します。

CRO
General MPRs (1)
Message queue:
Use default queue name: 🔽
Name:
Maximum blocks: 32
Queue buffer count: 2 Cold start: None
MFS path:
, indiated
MFS attribute bias: Binary
MFS null character:
O Character:
Activity keypoint frequency: 64 Trailing space: 🔽
Transaction threshold: 0 (seconds)
Application path:
\$IMSPROJ\\bin\x86\Debug

12) [サーバー] > [プロパティ] > [MSS] > [IMS] > [TM] > [MPRs] タブへ表示されている [追加] ボタンをクリックして、 メッセージ処理領域 (MPR) を作成します。IMS アプリケーションを実行するためには最少 1 つは必要です。

項目名	説明
名前	任意ですが、ここでは "MPR01" を指定します。
Transaction Class	実行するトランザクションクラスを指定します。ここでは "001"を指定します。
説明	任意ですが、ここでは "MPR for class1" を指定します。

一般 DB TM					
一般 MPRs (0)					
Add Message Processing Region					
名前: MPR01					
Transaction Class: 001 說明:		一般	MPRs (1)	
MPR for class1			名前	クラス	説明
キャンセル 追加	\rightarrow	編集	MPR01	001	MPR for class1

13) 画面左上の [Home] をクリックして一覧画面に戻ります。

追加画面にて下記項目を入力後、[追加] ボタンをクリックします。



3.9 Enterprise Server の開始と確認

1) [サーバー エクスプローラー] 内に [IMSDEMO] サーバーが表示されていることを確認します。表示されていない場合は [Micro Focus Servers] を右クリックし、[最新の情報に更新] を選択してリフレッシュしてください。



2) [サーバー エクスプローラー] 内の [IMSDEMO] サーバーを右クリックし、[プロジェクトに関連付ける] > [IMSDEMO] を選択します。これにより [IMSDEMO] プロジェクトから実行されるアプリケーションは [IMSDEMO] サーバーで処理され ることになります。

プロジェクトに関連付ける ▶ IMSDEMO

3) [IMSDEMO] サーバーを右クリックして [開始] を選択します。

Þ	IMSDEMO	
	開始(S)	

4) 下記ウィンドウが表示された場合は、ここではユーザーによる制限を行わないため [OK] ボタンをクリックします。

ME	Enterprise Server サインオン	×
サーバーの接続詳細をス	入力します:	
□サーバーを保護		
ユーザー名;		
パスワード:		
グループ:	デフォルト グループは空白	
✓ 資格情報の保存		
	OK キャンセル	

5) Enterprise Server Administration 画面へ移動して開始状態であることを確認後、[詳細] ボタンをクリックします。



6) [サーバー] > [診断] > [ES コンソール] で [IMSDEMO] サーバーのコンソールログをリアルタイムにチェックすることができ ます。また [Show Entire Log] をクリックしてログ全体を表示させることも可能です。

正常に開始されたことを確認します。

トレース	ダンプ ESコンソール CSコンソール					
● Show entries from 1 to 10 of 66 total entries ③ Show last 10 lines						
Entry	Event Show Entire Log					
57	151110 13545577 49980 IMSDEMO CASSI1428I TD recoverable cold started 13:54:55					
58	151110 13545577 49980 IMSDEMO CASSI1734I MFIMS Logon Successful 13:54:55					
59	151110 13545577 49980 IMSDEMO CASSI1600I SEP initialization completed successfully 13:54:55					
60	151110 13545801 49980 IMSDEMO CASSI5001I PLTPI Phase 1 - No PLT Specified 13:54:55					
61	151110 13545601 49980 IMSDEMO CASSI5040I Active SEP memory strategy set to x'00000001', retain count 100 13:54:56					
62	151110 13545643 50988 IMSDEMO CASSI1734I MFIMS Logon Successful 13:54:56					
63	151110 13545643 50988 IMSDEMO CASSI1600I SEP initialization completed successfully 13:54:56					
64	151110 13545643 50988 IMSDEMO CASSI5021I PLTPI Phase 2 - No PLT Specified 13:54:56					
65	151110 13545665 50544 IMSDEMO CASIP0005I IMS Message Processing Region started for classes "001" 13:54:56					
66	151110 13545685 50544 IMSDEMO CASIP0013I IMS Message Processing Region registered for classes "001" 13:54:56					

メインフレーム COBOL 開発: IMS Visual Studio 2012 編



注意 いくつかのサービス開始が失敗してもサーバーは開始されますので、ログ内容を必ず確認してください。

7) 画面左上の [Home] をクリックして一覧画面に戻ります。

3.10 IMS 資源定義の確認

Enterprise Server では IMS トランザクションをオンラインで参照・更新・追加・削除することが可能です。また IMS コントロール画面ではコマンドによる状態の確認が可能です。

1) Enterprise Server Administration 画面の [IMSDEMO] サーバーのステータス (開始) 直下にある [詳細] ボタ ンをクリックします。



2) [ES モニター & コントロール] ボタンをクリックします。



3) 画面左側中央にある [Resources] カテゴリ内のコンボボックスで [IMS] を選択して、[Transact] ボタンをクリックする と一覧が表示されますので、[MFDEMO] の左にある [Details] ボタンをクリックします。

		_			IMS Transa	ctions	
Г	Resources			New			
			Details	MFDEMO	DEMO001T	MPP	
	Transact.			Details	TESTMAIN	TEST002T	MPP
	Control		\rightarrow	Details	TESTMENU	TEST001T	MPP

4) [MFDEMO] トランザクションの詳細を表示すると、呼び出すプログラムや関連する PSB ファイルが確認できます。

IMS Tra	Refresh		
Apply Name:	MFDEMO		
Description:			
PSB Name:	DEMO001T	Class:	1
Program Name:	DEMO001T	Schedule:	Parallel 🗸



5) 次に [IMS] の [Control] ボタンをクリックすると IMS コントロール画面が表示されます。ボタンをクリックまたはコマンド入 力でトランザクションの状態などが確認できます。

IMS	Cont	rol			Refresh	Interval (Secs)
Commands:	/dis	: TRAN all		dis USER all	/dis DB all	/dump TM
Enter		/dis TRAN a	all			
/dis TRAN all						
TRAN CLS	ENQCT	QCT LCT	PLCT	CP NP LP SEGSZ	SEGNO PARLM	RC
TESTMENU 1	0	0 1	1	1010	0 0	0
TEST001T	ASCII	SPA =	1000	ATTR(Binary)	Null(x'1a')	
TESTMAIN 1	0	0 1	1	1010	0 0	0
TEST002T	ASCII	SPA =	1000	ATTR(Binary)	Null(x'1a')	
MFDEMO 1	0	0 1	1	1010	0 0	0
DEMO001T	ASCII			ATTR(Binary)	Null(x'la')	

3.11 IMS トランザクションの実行

現在 [IMSDEMO] サーバーが稼働していますので、サンプルプログラムを実行することができます。

ご使用の TN3270 エミュレータを、前項で作成した TN3270 リスナーポート (localhost:5039) へ接続します。

1) 下記は Micro Focus Rumba を使用した画面です。サンプルの SIT では初期トランザクションに [/IMS] が指定されて いるため、接続後 IMS サインオン画面が表示されます。[USERID] と [PASSWORD] へ "SYSAD" を入力して Enter キーを押します。

CICSS	IT - IMSSIT	[Refresh		
Apply Name:	IMSSIT				
Description:	ES/IMS example 3	SIT			
Startup List:	IMSSTRT 🗸	Work Area: 51	2 (CV)		
SysID:	EIMS	Initial Tran ID: /I	MS		
R.	Micro Focus R	umba - メインフレー	ムディスプレイ		×
ファイル(F) 編集(E)	表示(V) 接続(C)	転送(R) オプション(O) ツール(T) /	√レプ(H)	
🗋 - 🤜 🖏 🗄	s 2 0 0	ର 🖻 • 🔯 🕞 🔊	😜 🗞 🎸	🚰 🗣 🗳 🎛	-
🖅 🖉 🚆 メインフレ	ームデ × \				
DFS3649A /SIG	N COMMAND REQU	IRED FOR IMSDEMO)		
DATE: 2015/11	1/10 TIME: 1	4:23:27			
NODE NAME: NET	TA000				
USERID: SYSAD					
PASSWORD :					
USER DESCRIPTO	DR:				
GROUP NAME: NEW PASSWORD:					
NO OUTPUT SECU	JRITY AVAILABL	E			
<					>
レディ 実行中	SSL ht	NUMFLE NETA000	OVR CAP	NUM W 9,	17



- 2) 正常にサインオンできましたら、クリアキー(CTL+SHIFT+Z)で画面をクリアします。
- 3) IMS トランザクションを開始する下記コマンドを入力後、Enterキーを押します。

コマンド) /STA TRAN MFDEMO

/STA TRAN MFDEMO

4) 1) 正常に開始されると下記メッセージが表示されます。

DFS058I 13:02:03 START COMMAND COMPLETED

- 5) 再度、画面をクリア(CTL+SHIFT+Z)します。
- 6) "MFDEMO" (末尾ブランク)を入力して Enter キーを押します。

MFDEMO _

7) 下記のようにサンプルの初期画面が表示されます。ここでは TESTDEPT テーブルを追加します。

タブキーで項目間を移動して、[機能コード] へは "A" を [テーブル ID] へは "TESTDEPT" を入力後、Enter キー を押します。





8) タブキーで項目間を移動して、TESTDEPT テーブルの要素を入力後 Enter キーを押します。

なお、このサンプルは日本語入力に対応していません。

項目名	説明
DESCRIPTION	説明を入力します。ここでは "A TABLE OF TEST DEPARTMENT" を指定します。
NUMBER OF CHARACTERS IN ELEMENT KEY	ここでは "20" を指定します。
NUMBER OF CHARACTERS	ここでは "50"を指定します。
PASSWORD PROTECT	ここでは "N" を指定します。
FUNC	"A"を確認します。
ELEMENT KEY(1 行目)	ここでは "123" を指定します。
ELEMENT VALUE(1 行目)	ここでは "DEPT123" を指定します。
ELEMENT KEY(2 行目)	ここでは "456" を指定します。
ELEMENT VALUE(2 行目)	ここでは "DEPT456" を指定します。

ファイル(F) 編集(E) 表示(V) 接続(C) 転送(R) オプション(O) ソール(T) ヘルプ(H)	
D - 🤜 🖽 🗔 😽 🖉 D 🛍 🗂 🛍 - 🔕 D 🕒 🔂 🍕 🎜 👫 🗐 🖧 🖉 !	
<u> ティ</u> 響 メインフレームデ… ×	Þ
TABLE-ID: TESTDEPT	
DESCRIPTION:	
NUMBER OF CHARACTERS IN ELEMENT KEY: 20 NUMBER OF CHARACTERS IN ELEMENT:	50
LAST ACTIVITY DATE PASSWORD PROTECT: N PASSWORD:	
FUNC ELEMENT KEY ELEMENT VALUE/DESCRIPTION	
A 123 DEPT123	
A 450 DEP1450_ A	
A	
Â	
A	
Â	
A	
A	
A	
NEXT FUNCTION: RETURN CONTINUE ABORT NEXT-ELEMENT	
レデ*イ 実行中 SSL が NUMFLE NETC000 OVR CAP NUM W 11,35 127.0	.0.1

9) 画面下部の [RETURN] までタブキーで移動し、"X" を入力します。ほかの選択肢に "X" が入っていないことを確認後 Enter キーを押すとメニュー画面へ戻ります。

NEXT FUNCTION: RETURN X CONTINUE ABORT NEXT-ELEMENT



10) 作成した TESTDEPT テーブル要素を変更します。[機能コード] へは "C" を [テーブル ID] へは "TESTDEPT" を 入力後、Enter キーを押します。



11) テーブル要素の 3 行目に下記を追加して Enter キーを押します。

項目名	説明
FUNC	ここでは "A" を指定します。
ELEMENT KEY(3 行目)	ここでは "789"を指定します。
ELEMENT VALUE(3 行目)	ここでは "DEPT789" を指定します。



- 12) 画面下部の [RETURN] までタブキーで移動し、"X" を入力します。ほかの選択肢に "X" が入っていないことを確認後 Enter キーを押すとメニュー画面へ戻ります。
- 13) 次に TESTERS テーブルを作成します。 [機能コード] へは "A" を [テーブル ID] へは "TESTERS" を入力後、 Enter キーを押します。





14) タブキーで項目間を移動して、TESTERS テーブルの要素を入力後 Enter キーを押します。

項目名	説明
DESCRIPTION	説明を入力します。ここでは "A TABLE OF VALID SOFTWARE TESTERS"を指定します。
NUMBER OF CHARACTERS IN ELEMENT KEY	ここでは "20" を指定します。
NUMBER OF CHARACTERS IN ELEMENT	ここでは "50" を指定します。
PASSWORD PROTECT	ここでは "N" を指定します。
FUNC	"A"を確認します。
ELEMENT KEY(1 行目)	ここでは "TARO TOKYO" を指定します。
ELEMENT VALUE(1 行目)	ここでは "IMS DC TESTER" を指定します。
ELEMENT KEY(2 行目)	ここでは "HANAKO OSAKA" を指定します。
ELEMENT VALUE(2 行目)	ここでは "IMS DB TESTER" を指定します。



- 15) 前項と同じ要領でメニュー画面へ戻ります。
- 16) [機能コード] ヘ "E" を入力後、Enter キーを押してトランザクションを終了します。





17) サンプルプログラムの終了画面が表示されます。



18) TN3270 エミュレータを切断します。

3.12 プログラムの動的デバッグ

Visual Studio を使用して、サンプルプログラムのデバッグを行います。前項で実施しましたが、[IMSDEMO] サーバーの [動 的デバッグを許可] へのチェックと [IMSDEMO] サーバーと Visual Studio プロジェクトの関連付けが必要です。

1) [デバッグ] プルダウンメニューの [ステップ イン] を選択します。

ファー	1/l(F)	編集(E)	表示(V)	プロジェクト(P)	ビルド(B)	デバッグ(D)
	ウィンド	ウ(W)				
	グラフィッ	ックス				
•	デバッグ	'開始(S)				F5
▶	デバッグ	なしで開始(I	н)			Ctrl+F5
	パフォー	マンス分析の	開始(A)			Alt+F2
2	一時停	止したパフォー	-マンス分析(D開始(Y)		Ctrl+Alt
¢®	プロセス	、にアタッチ(P)				
	インスト	ールされてい	るアプリケーシ	ヨン パッケージのデノ	(ッグ(D)	
	例外(X	()				Ctrl+D,
ς.	ステップ	イン(I)				F11
Ċ,	ステップ	オーバー(0)				F10

2) 画面左下がオレンジ色の [準備完了] となり、アタッチ待機状態になったことを確認します。





3) 前項と同様に TN3270 エミュレータから IMS トランザクションを実行すると、プログラムのステップ実行が可能になります。 [F11] キーもしくは [デバッグ] プルダウンメニューから [ステップ イン] を選択してステップを進めることができます。

マウスオーバーやウォッチタブを利用して変数の値が確認できます。なお、このサンプルプログラムでは TN3270 エミュレータとの画面送受信がありますので、その都度、表示を切り替えてデバッグします。

IF PCBSTAT NOT = BB AND NOT =	QC AND NOT = QD
MOVE 3514 TO 9000-ABEND-CO	DE
PERFORM C-099-ABEND THRU C	-099-EXIT
MOVE LOW-VALUES TO DEMO91-SCRE	EN.
MOVE LOW-VALUES TO DEMO92-SCRE	EN.
STRING TIOAIO(1:LL01-IN - 4) D	ELIMITED BY SIZE
INTO D 🥥 TIOAIO 🔍 - MFDEMO)
STRING TIUHIOGI - LEGT IN 47 0	CEIMITED DI DIZE
INTU DEMUSZ-SUREEN.	
DEDEODW 0-022-FORMAT-NULL-SORE	EN THRU CHUZIHEAII.
MOVE CRT-LENGTH TO LL01-IN.	EN THRU C-022-EATT.
C-020-EXIT.	
EXIT.	
C-021-FORMAT-NULL-SCREEN.	
*** SCREEN HANDLING FOR DYNAMIC ATT	RIBUTE MODIFICATION
MOVE LOW-VALUES TO DEMOST-FUNC	I I UNC
2 · · · ·	
£1	
ΰ	値
DC001TIO	{長さ=3004}: "MFDEMO
LL01-IN OF DC001TIO	+00011
Z001 OF DC001TIO	
Z0012 OF DC001TIO	
TIOAIO OF DC001TIO	MEDEMO

4) 希望のステップの左端をクリックすることにより、ブレークポイントを設定することも可能です。

PERFORM C-021-FORMAT-NULL-SCREEN THRU C-021-EXIT.

5) 先に進める場合は画面上部の [続行] アイコンをクリックします。

▶ 続行(<u>C</u>) -

6) デバッグを終了させるため、画面上部の [デバッグ停止] アイコンをクリックします。



- 7) TN3270 エミュレータを切断します。
- 8) 画面左下が水色の [準備完了] となり、アタッチ待機状態ではなくなったことを確認します。
 エラーー覧 プロジェクト詳細ウィントウ 準備完了



9) 出力タブに表示されているデバッグ履歴を消去するには、[全てクリア] アイコンをクリックします。

出力		**********		
出力元の表示(<u>S</u>): デバッグ	-	£ 2	≊	ab 🗗
スレッド (名前がありません)(0x8134) はコード 0 (0x0) で終了 スレッド (名前がありません)(0x7e74) はコード 0 (0x0) で終了 スレッド (名前がありません)(0x8168) はコード 0 (0x0) で終了 フレッド (名前がありません)(0x8168) はコード 0 (0x0) で終了	しました。 しました。 しました。			
- プログラム 'IMSDEMO' はコード 0 (0x0) で終了しました。	02.0/20			

3.13 Enterprise Server の停止

1) [IMSDEMO] サーバーを停止します。

\triangleright	IMSDEMO
	開始(<u>S</u>)
	停止

2) [IMSDEMO] サーバーの停止を確認後、Visual Studio を終了します。

編集	MFES (MSS)	IMSDEMO	停止
	64 0		M \$2

WHAT'S NEXT

- メインフレーム COBOL 開発 : JCL Visual Studio 2012 編
- メインフレーム COBOL 開発 : CICS Visual Studio 2012 編
- 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。