Micro Focus Enterprise Developer チュートリアル

メインフレーム PL/I 開発: CICS

Eclipse 編

1. 目的

本チュートリアルでは、PL/I 言語で書かれた CICS 命令を含むソースをオープン環境へ移行後、Eclipse を使用してプロジェクトの作成、コンパイル、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。

2. 前提

- Windows 開発環境に Enterprise Developer 2.3 for Eclipse がインストール済であること。
- Micro Focus Rumba などの TN3270 エミュレーターがインストール済で稼働実績があること。

3. チュートリアル手順の概要

- 1. チュートリアルの準備
- 2. Eclipseの起動
- 3. メインフレーム PL/I プロジェクトのインポート
- 4. プロジェクトプロパティの確認
- 5. ビルドの実行
- 6. Enterprise Server の設定
- 7. Enterprise Server 開始と確認
- 8. CICS の実行
- 9. PL/I ソースのデバッグ
- 10. 終了処理



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) Eclipse のワークスペースで使用する「work」フォルダを C ディレクトリ直下に作成します。
- 2) 製品をインストールしたフォルダ配下に含まれているサンプルプログラム「CICSDEMO」フォルダを作成した C:¥work へ コピーします。
 - 例) C:¥Users¥Public¥Documents¥Micro Focus¥Enterprise Developer¥Samples¥PLI-Eclipse¥CICSDEMO



3) コピーした「CICSDEMO」を展開して「CICS-Subs」フォルダを開き、「.project」ファイルを右クリック後、テキストエ ディターを開きます。

work → C	CICSDEMO → CICS-	-Subs		
,	* 名前	*		
pliBuild 碑 .pliProj				
	.project			
	STAFF04.p	ノロクラムから開く(H)		
	STAFF06.p	SAKURAで開<(E)		

4) 「CICS」フォルダに含まれるプロジェクトと同名になることを避けるため、「name」タグに囲まれた文字列を "CICS" から "CICS-Subs" へ変更後、保存してエディタを閉じます。

<pr< th=""><th>ojectDescription≻↔</th><th></th><th><projectdescription>↔</projectdescription></th></pr<>	ojectDescription≻↔		<projectdescription>↔</projectdescription>
^	<name>CICS</name> ←		^ <name>CICS-Subs</name> ↔
<u> </u>	<comment></comment> ↔	\rightarrow	<pre><comment></comment>↔</pre>



3.2 Eclipse の起動

1) Micro Focus Enterprise Developer for Eclipse を起動します。

WF	Enterprise Developer for Eclipse	👗 Adobe Reader XI	
MF	Enterprise Developer for Visual Studio	Enterprise Developer for Eclipse の起動	١
MF	ドキュメント - Eclipse	Micro Focus License Manager	۰
0	ドキュメント - Visual Studio 2012	Micro Focus Enterprise Developer	

2) 前項で作成した「C:¥work」をワークスペースへ指定して、[OK] ボタンをクリックします。

M.	ワークスペース・ランチャー		×
ワークスペースの選択			
Eclipse は、ワークスペースと呼ばれるフ このセッションに使用するワークスペース・	オルダにプロジェクトを保存します。 フォルダを選択してください。		
ワークスペース(<u>W</u>): C:¥work		✓ 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>) OK	キャンセル	

3) [ようこそ] タブが表示されますので、[Open PL/I Perspective] をクリックして、PL/I パースペクティブを開きます。



4) PL/I パースペクティブ表示後、[プロジェクト] プルダウンメニューの [自動的にビルド] を選択して、これをオフにします。

) 🕯	編集(E) ナビゲート(N) 検索	プロジェクト(P)) 編	集(E)	ナビゲート(N)	検索	プロジェクト(P)
	プロジェクトを開く(E) プロジェクトを閉じる(S)				カシ プロシ	ェクトを開く(E) ェクトを閉じる(S)		
010	すべてビルド(A) プロジェクトのビルド(B) ワーキング・セットのビルド(W) クリーン(N)	Ctrl+B ▶	_	810	すべて プロジ: ワーキ クリーン	ビルド(A) エクトのビルド(B) ング・セットのビル ン(N)	ř(W)	Ctrl+B
~	目動的にビルド(M)		\rightarrow		自動的	ዓመይካካዮ(M)		

5) 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。 [ウィンドウ] プロダウンメニューの [設定] > [Micro Focus] > [PL/I] > [指令の確定] > [指令の確定 を行う] 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。



- 3.3 X + X + Z = X +
 - 1) 用意したサンプルプロジェクトをインポートします。 [ファイル] プルダウンメニューから [インポート] を選択し、インポートウィン ドウにて [General] > [既存プロジェクトをワークスペースへ] を選択後 [次へ] ボタンをクリックします。

ME	インポート	- 🗆 🗙
選択 アーカイブ・ファイルまたに	まディレクトリーから新規プロジェクトを作成します。	Ľ.
インポート・ソースの選抜 フィルタ入力	₹(<u>⊆)</u> :	
 ▲ General ● アーカイフ ● アーカイノ ● ファイル・: ● 既存プロ ● 設定 	・ ・ファイル ンステム ジェクトをワークスペースへ	^
⊳ 🕞 CVS		~
?	< 戻る(B) 次へ(N) > 終了(E)	キャンセル



2) [ルート・ディレクトリの選択] へ「C:¥work¥CICSDEMO」を指定すると、このフォルダに含まれるプロジェクトが表示され ます。チェックをオンにした状態で [終了] ボタンをクリックします。

WE	インポート	- 🗆 🗙
プロジェクトのイン 既存の Eclipse プロジ	ボート 江クトを検索するディレクトリーを選択します。	
 ルート・ディレクトリー アーカイブ・ファイルの プロジェクト(<u>P</u>): 	の選択(工): C:¥work¥CICSDEMO の選択(<u>A</u>):	 ✓ 参照(<u>R</u>) ✓ 参照(<u>R</u>)
CICS (C:¥w	ork¥CICSDEMO¥CICS 直接アクセス) (C:¥work¥CICSDEMO¥CICS-Subs 直接アクセス	すべて選択(<u>S)</u> 選択をすべて解除(<u>D</u>)
< オプション オプション ネストしたプロジェグ フプロジェクトをワーク ワークスペースに既	→ たを検索(山) スペースにコピー(<u>C</u>) に存在するプロジェクトを隠す(i)	更新(<u>E</u>)
ワーキング・セット ワーキング・セット ワーキング・セット(<u>0</u>)	こプロジェクトを追加(工) ;	✓ 選択(E)
?	< 戻る(<u>B</u>) 次へ(<u>N</u>) > 終了(<u>F</u>)	キャンセル

3) [PL/I エクスプローラー] にインポートした 2 つのプロジェクトが表示されます。

🔓 PL/I Explorer 🛛	🔜 サーバー エクスプローラー
🖻 😂 CICS	
CICS-Subs	

4) [CICS] プロジェクトを展開すると PL/I ソースなどが確認できます。



3.4 プロジェクトプロパティの確認

このサンプルは [CICS-Subs] プロジェクトで作成される lib オブジェクトを [CICS] プロジェクトがリンクして dll を生成する 内容になっています。プロジェクトの設定値を確認していきます。

- [CICS-Subs] プロジェクトを右クリックし、[プロパティ] を選択すると [プロパティ ウィンドウ] が表示されます。
 64-bit 稼働が指定されていますが、ここでは 32-bit OS で実行することを考慮して 32-bit 稼働へ変更します。
 - ① [Micro Focus] > [ビルド構成] で [構成の管理] ボタンをクリックして構成管理ウィンドウを表示します。

フィルタ入力	ビルド構成	<
 リソース Micro Focus ビルドパス 	x64 [使用中]	✓構成の管理
▷ プロジェクト設定 Project Facets	出力ディレクトリ名: bin¥x64¥debug	参照

② [ビルドの構成管理] ウィンドウでは [x86] のチェックボックスをオンにして [完了] ボタンをクリックします。

₩.	ビルド構成の管理	×
ビルド構成の 現在使用中のビ	管理 ルド構成の選択やビルド構成の作成と削除をおこないます。	*
🖪 🗘 🖉 🗸	∕ ¥	
構成の名前		使用中
x64 x86		
K		
?		完了

③ [Micro Focus] > [ビルド構成] ウィンドウへ戻り [x86] へ変更されたことと、プロジェクト配下の「bin¥debug」 フォルダへ実行ファイルが出力されることを確認後 [適用] ボタンをクリックします。

ビルド構成			
	x86 [使用中]		
	u カディ カレロタ・ bin¥c	lebua	
	шлл 109794: 01140	icoug	

2) [Micro Focus] > [ビルド構成] > [PL/I リンク設定] を選択して内容を確認すると、「CICS-Subs」という名前で 32 ビット稼働する単一ライブラリとしてオブジェクトを生成することがわかります。

Micro Focus	x86 [使用中]	
ビルド構成 BMS		
 ▶ PL/I コンパイル設定 ▶ PL/I リンク設定 ▶ アセンブラ コンパイラ > アセンブラ コンパイラ 	設定: フィルタテキストを入力	
イベント	設定	値
⊿ プロジェクト設定	▲ 一般的なオプション	
BMS	プラットフォーム ターゲット	32 bit
⊳ IMS	出力タイプ	単一ライブラリ
▲ PL/I コンパイル設定	出力ファイル名	CICS-Subs
CICS プリプロセッ	システム	

 [Micro Focus] > [プロジェクト設定] > [PL/I コンパイル設定]を選択して内容を確認すると、サンプルの内容に沿って、 「システム」には「CICS」が設定されており、デバッグ用のファイルを生成することがわかります。最後に [OK] ボタンをクリ ックしてください。

 ▲ Micro Focus ビルドパス ▷ ビルド構成 ▲ プロジェクト設定 	設定: フィルタテキストを入力		
BMS	設定	値	
⊳ IMS	▲ 一般的なオプション		
▷ PL/I コンパイル設定	システム	CICS	
▷ アセンブラ コンパイラ	デバッグ用にコンパイル (-debug)	はい	
アセンブラ リンカ	リストファイルを出力 (-1)	いいえ	
Project Facets	最適化レベル (-opt)		
Task Repository	エンディアン (-bigendian)		
Task Tags	EXEC プリプロセッサ オプション (-optexec)		
> Validation	追加オプション		

4) 次に [CICS] プロジェクトも同様に [プロパティ ウィンドウ] を表示します。

64-bit 稼働が指定されていますが、[CICS-Subs] と同じビット数にする必要がありますので、32-bit 稼働へ変更します。

① [Micro Focus] > [ビルド構成] で [構成の管理] ボタンをクリックして構成管理ウィンドウを表示します。



② [ビルドの構成管理] ウィンドウでは [x86] のチェックボックスをオンにして [完了] ボタンをクリックします。

M.	ビルド構成の管理	×
ビルド構成の管理 現在使用中のビルド構成の選	択やビルド構成の作成と削除をおこないます。	*
다. 라 🥒 🛩 🗱		
構成の名前		使用中
x64		
x86		
?		完了

③ [Micro Focus] > [ビルド構成] ウィンドウへ戻り [x86] へ変更されたことと、プロジェクト配下の「bin¥debug」 フォルダへ実行ファイルが出力されることを確認後 [適用] ボタンをクリックします。

ビルド構成	
x86 [使用中]	
出力ディレクトリ名:	pin¥debug
_	

5) [Micro Focus] > [ビルド構成] > [PL/I リンク設定] を選択して内容を確認すると、「CICS」という名前で 32 ビット 稼働する実行可能ネイティブライブラリとしてオブジェクトを生成することがわかります。

⊿ Micro Focus ビルド パス	x86 [使用中]	~ 構成
 ▲ ビルド構成 BMS > PL/I コンパイル設定 > PL/I リンク設定 > アセンブラ コンパイラ アセンブラ リンカ 	設定: フィルタテキストを入力	
イベント ▲ プロジェクト設定	設定	値
BMS ▷ IMS ▲ PL/I コンパイル設定	 ブラットフォーム ターゲット 出力タイプ 出力ファイル名 	32 bit すべて実行可能ネイティブライブラリ CICS

6) [Micro Focus] > [ビルド構成] > [PL/I リンク設定] > [追加のリンクファイル] を選択すると、「CICS-Subs.lib」をリ ンクするように指定していることがわかります。

▲ Micro Focus ビルドパス	x86 [使用中]
▲ ビルト構成 BMS	
▶ PL/I コンパイル設定	リンクする追加項目 (obj、lib 等):
▲ PL/I リンク設定 追加のリンクファイ	EICS-Subs.lib - CICS-Subs/bin/deb
ヨカロックノックノア・17	

7) 「CICS-Subs.lib」を選択した状態で [編集] ボタンをクリックして、再度 [CICS-Subs.lib] を選択指定して [OK] ボタンをクリックします。前ウィンドウに戻りましたら [適用] ボタンをクリックします。

リンクする追加項目 (obj、lib 等):		
CICS-Subs.lib - CICS-subs/bin/deb	ファイルを追加(<u>F</u>)	🛭 🗁 CICS-subs
	外部ファイルを追加(X)	🔺 🗁 bin
		a 🗁 debug
	編集(<u>E</u>)	\rightarrow CICS-Subs.lib

8) [Micro Focus] > [ビルド構成] > [PL/I コンパイル設定] を選択して内容を確認すると、サンプルの内容に沿って、「シ ステム」には「CICS」が設定されており、デバッグ用のファイルを生成することがわかります。 [EXEC プリプロセッサ オプシ ョン] にはデバッグモード時に指定する "plitest" が指定されていますが、ここではクリアして [OK] ボタンをクリックしてくださ い。

▲ Micro Focus ビルド パス	×86 [使用]中]				
▲ ビルド構成 DMS						
PL/I コンパイル設定	▼構成の)固有な設定を可能にする(C)				
CICS プリプロセッ		1100000 C 11000 C (2)				
SQL プリプロセット				_		
マクロ プリプロセット	設定:					
▲ PL/1リンク設定 追加のリンクファイ	70	レタテキストを入力				
Pセンブラ コンパイラ					4 一般的なオプション	
アセンブラ リンカ	設定		値		システム	CICS
イベント	4.	一般的なオプション		-		cics the
⊿ プロジェクト設定		システム	CICS		テハック用にコンハイル (-debug)	はい
BMS		デバッグ用にコンパイル (-debug)	はい		リストファイルを出力 (-1)	いいえ
▷ IMS		リストファイルを出力 (-1)	いいえ	L.	最適化レベル (-opt)	-noopt
▷ PL/I コンパイル設定		最適化レベル (-opt)	-noopt	-	エンディアン (-bigendian)	
Pセンブラ コンパイラ		エンティアン (-Digendian)		L .	EVEC THETE that the style option of	
アセンフラ リンカ		EXEC フリフロセッサ オフション (-optexec)	plitest	\rightarrow	EXEC JUJU COURSEL	

MICRO FOCUS

- 3.5 ビルドの実行
 - 1) [プロジェクト] プルダウンメニューの [自動的にビルド] を選択して、これをオンすると自動的にビルドが実行されます。



2) [CICS] プロジェクトへ [CICS-Subs] で生成された新しい lib をリンクさせるため、再ビルドします。 [プロジェクト] プル ダウンメニューから [クリーン] を選択して [クリーンウィンドウ] を表示します。



3) [CICS] プロジェクトだけを再ビルドするように、下記のように指定して [OK] ボタンをクリックして成功を確認します。

			🖳 コンソール 🛛 🔝 問題 🔲 プロパティー
「クリーン」によって、ビルドにかかわるすべての問題と状態が破棄されます。プロジェクトは最初か らリビルドされます。			os.init.unix: init:
○ すべてのプロジェクトをクリーン(<u>A</u>)	●以下で選択したプロジェクトをクリーン(S)		post.build.cfg.x64:
			BUILD SUCCESSFUL Build finished with no errors.
CICS-Subs		\rightarrow	Total time: 0 seconds

4) 各プロジェクトの bin¥debug フォルダ配下に目的の lib、dll が作成されていることを確認してください。

work → CICSDEMO → CICS → bin →	debug		
名前	更新	CICSDEMO → CICS-Subs → bin → debug	
STAFF05.dll	201	▲ 名前 ▲	
STAFF03.dll	201 201	E CICS-Subs.lib	



3.6 Enterprise Server の設定

1) PL/I を実行するためのエンジンを搭載した Enterprise Server を作成します。 [サーバー エクスプローラー] タブの [ロ ーカル] を右クリックして [Administration ページを開く] を選択します。

🔓 PL/I Explorer 📃 サーバー	エクスプローラー 🛛 🛛 🗉	🕞
▶ 📃 ローカル [localhost:86]	新規作成(N)	•
	Administration ページを開く	Ctrl+F3

 C:¥work¥CICSDEMO¥CICS には Enterprise Server のサンプルが含まれており、これをインポートします。PL/I ア プリケーションは 32 ビット稼働を指定したため、「C:¥work¥CICSDEMO¥CICS¥staff_def」がインポート対象となり ます。64 ビットで稼働させる場合は「C:¥work¥JCLDEMO¥plijcl64_def」をインポートしてください。

Enterprise Server Administration 画面左側の [インポート] をクリックして、表示される下記項目へ前述のパスを 入力後、 [次へ] ボタンをクリックします。

サーバー情報のインボート (Page 1 of 4):

サーバーデー	タの復旧元ディレクトリの選択:
file:/// 🗸	C:\work\CICSDEMO\CICS\staff_def

3) 画面の Page 2/4、3/4、ではそのまま [次へ] ボタンを、Page 4/4 では [OK] ボタンをクリックすると、[STAFF] という 名前の 32 ビットアプリケーション稼働用 Enterprise Server が追加されます。



アプリケーション稼働ビット数 = Enterprise Server 稼働ビット数である必要があります。	

4) 設定を変更するため、[編集] ボタンをクリックします。



5) [サーバー] > [プロパティ] > [一般] タブで表示される画面の [構成情報] 欄を下記のように入力し、[Apply] ボタンを クリックします。

変更前;

[ES-Environment] CICSDEMO=C:¥Users¥Public¥Documents¥Micro Focus¥Enterprise Developer¥Samples¥"PLI-VS or PLI-Eclipse"¥CICSDEMO ES_SSTM_CICS="\$CICSDEMO¥sstmcics.jcl" CODEWATCH_NOTIF=Y

変更後;

[ES-Environment] CICSDEMO=C:¥work¥CICSDEMO ES_SSTM_CICS="\$CICSDEMO¥CICS¥sstmcics.jcl" CODEWATCH_NOTIF=Y

自情報

ES_SSTM_CICS 環境変数: CICS 環境で JCL を使用する際に SSTM CICS 環境の初期化に使用される JCL の所在地を指定します。 CODEWATCH_NOTIF 環境変数: デバッガを使用する開発用サーバーに指定します。本番サーバーにはパーフォーマンスの観点から排除することを推奨します。

6) [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確認します。

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。
システム初期化テーブル(SIT)	SIT 名称。
トランザクションパス	PCT やプログラムから呼ばれる実行ファイルが存在するパス。
File Path	FCT やプログラムからアクセスするファイルが存在するパス。
マップパス	マップ実行ファイルが存在するパス。
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。

CICS (♥) JES (♥) INIS PL/(♥) CICS 有効: マ システム初期化テーブル (SIT): STAFF トランザクションパス: \$CICSDEMO\CICS\bin\debug	
File Path: \$CICSDEMO\CICS\staff_base	Jルディスク (C:) → work → CICSDEMO → CICS →
マップバス:	名前 更新
<pre>\$CICSDEMO\CICS\staff_base\</pre>	\mu bin 201
レーレーレーン	staff_base 201
<pre>\$CICSDEMO\CICS\staff_base</pre>	istaff_def 201 istaff64_def 201

7) [サーバー] > [プロパティ] > [MSS] > [JES] > [一般] タブで表示される画面の各項目を確認します。

項目名	説明
ジョブ入力サブシステム 有効	[JES] タブ配下の設定をオン、オフ指定。
JES プログラム パス	実行ファイルが存在するパス。
システムカタログ	カタログファイルのパスとファイル名称。
データセットの省略時ロケーション	JCL などで指定するファイルのデフォルトパス。

CICS (✔) JES (✔) IMS PL/I (✔)		
──般 Initiators (1) Printers (0)		
ジョブ入力サブシステム有効: 🔽		
JESプログラムパス:		
\$CICSDEMO\CICS\bin\debug		
システム力タログ・	(C:) \mapsto work \mapsto CICSDEMO \mapsto CICS \mapsto staff_ba	se
\$CICSDEMO\CICS\staff_base\catalog.dat	^ 名前 [^]	更新日
データセットの省略時ロケーション:	🖺 DBA.dat	2015/
\$CICSDEMO\CICS\staff_base\	DBA.idx	2015/
	📋 DBA.pro	2015/



8) [サーバー] > [プロパティ] > [MSS] > [PL/I] > [一般] タブで表示される画面の各項目を確認します。

項目名	説明
PL/I 有効	[PL/I] タブ配下の設定をオン、オフ指定。
Codewatch ソース パス	デバッグで使用するソースファイルが存在するパス。
Codewatch STB パス	デバッグで使用するデバッグファイルが存在するパス。
PL/I 構成ディレクトリ	プロジェクトのパス。

CICS (✔) JES (✔) IMS PL/I (✔)	
一般	
PL/I 有効:	
Prompt for PLITEST attachment:	
Codewatch ソース パス:	
\$CICSDEMO\CICS;\$CICSDEMO\CICS-Subs	0
Codewatch STB バス:	
<pre>\$CICSDEMO\CICS\staff_base;\$CICSDEMO\CICS\bin\Debug;\$CICSDEM 0\CICS-Subs\bin\Debug</pre>	0
PL/I 構成ディレクトリ:	
\$CICSDEMO\CICS	~

9) [リスナー] タブで表示される TN3270 接続用のポート番号を確認します。サンプルでは 5150 ポートを使用します。

	編集	TN3270	tcp: 5150 (TOK-kt-W8v1.microfocus.com)
--	----	--------	---

10) 画面左上の [Home] をクリックして一覧画面に戻ります。



Home

3.7 Enterprise Server の開始と確認

- 1) サーバーエクスプローラ内に [STAFF] 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 [ローカル [localhost:86]] を右クリックし、 [更新] を選択してリフレッシュしてください。
- 2) [STAFF] サーバーを右クリックして [開始] を選択します。



3) 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	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 ‡ฑวชม	,

4) 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。



5) [サーバー] > [診断] > [ES コンソール] で [STAFF] 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	ス ダンプ ESコンソール CSコンソール
曲直	● Show entries from 1 to 10 ● Show last 10 lines of 54 total entries
Entry	Event Show Entire Log
45	151104 11070899 44852 STAFF JCLCM0199I JOB01000 SSTMCICS Program MFJBR14 is COBOL VSC2 ASCII Big-Endian AMODE31. 11:07:08
46	151104 11070720 44852 STAFF JES000040I SSTM environment established successfully. (CICS) Job #: 001000 11:07:07
47	151104 11070740 44852 STAFF CASSI5001I PLTPI Phase 1 - No PLT Specified 11:07:07
48	151104 11070762 44652 STAFF CASSI5040I Active SEP memory strategy set to x'00000001', retain count 100 11:07:07
49	151104 11070823 43312 STAFF CASSI1744I MFPLI support loaded successfully 11:07:08
50	151104 11070844 43312 STAFF CASBJ0005I Batch initiator started for job classes "ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789" 11:07:08
51	151104 11070885 44284 STAFF CASSI1744I MFPLI support loaded successfully 11:07:08
52	151104 11070886 44284 STAFF CASSI1600I SEP initialization completed successfully 11:07:08
53	151104 11070908 44284 STAFF JES000040I SSTM environment established successfully. (CICS) 11:07:08
54	151104 11070929 44284 STAFF CASSI5021I PLTPI Phase 2 - No PLT Specified 11:07:09

•	· 注意
いつけ	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。

6) 画面左上の [Home] をクリックして一覧画面に戻ります。



3.8 CICS の実行

1) TN3270 エミュレーターを使用して、Enterprise Server のポートへの接続を設定して、[接続] ボタンをクリックします。 ここでは Micro Focus Rumba を使用します。



2) CICS ログイン画面が表示されますので USERID と Password へ「SYSAD」を入力して Enter キーを押します。



3) ログインに成功したら、画面クリア後(CTL+SHIFT+Z)、PCT 名である「TTEA」を入力して Enter キーを押すと、 プログラムが起動されてアプリケーション画面が表示されます。





4) [Name] 欄へ "WILSON" と入力して Enter キーを押すと、ファイルからデータが読み込まれて表示されます。

		ノアイル(F) 構築(E) 表示(V) 接続(C) 輸送(R) オノション(D) ツール(1) ヘルノ(H)
		□·ㅋ···································
愛 Micro Focus データファイ)	レツール - [DBA.dat (固定 長さ 索引)]	- D X ジョン ジョン ション ション ション ション ション ション ション ション ション シ
● ファイル(E) 編集(E) 表示(Y) ファイル(E) 検索(S) オプション	(Q) ソール(王) ウィンドウ(型) ヘルプ(H)	Staff Directory - Staff Information Summary
		Name: WILSON, MICHAEL Pref: MIKE
ANGI ¥ ±4- ¥ EEE ¥ 6		Designations: GLU, FIIG
130001LEE JR	HPPH OK	Mailing address: 1115 BROADWAY AVE.
130002RUSSELL //FWP 音 130003ATKINSON 1 DBA_LAYOUT	783 2 , 18	City: LINCIN Charles Units and and
130004HUTTON 2 ST_STAFF_CODE	char(6) 130000	State. IL 21F. 24001-3462
120006SMTH 2 1AST	char(30) WILSON	Office address: 1115 BROADWAY AVE. SUITE A2
130007STROUP - 3 FIRST	char(15) MICHAEL	City: LINCOLN State: IL ZIP: 24061-3482
130009S0UIRE 2 BUS_ADDR	char(10) MINE	Phone: 000-000-0000
1300110L1FF0RD 3 L1NE1	char(30) 1115 BROADMAY AVE.	000-0000
1300120 SHIELDS 3 CITY	char(30) LINCOLN	
< > 3 STATE	char(2) IL	·
CEAds(Lata: 長さ素引) X		PF2=END PF3=Search Criteria Accept
WILSON (IPI\ 85 /		
3400K	N/A 121-11 長 400 (固定) N/	

3.9 PL/I ソースのデバッグ

- 1) デバッガを起動させてステップ実行するために、前項でクリアしたプロジェクトのプロパティを再指定します。PL/I エクスプローラ ーから [CICS] プロジェクトを右クリックして [プロパティ] を選択し、プロパティ ウィンドウを開きます。
- [Micro Focus] > [ビルド構成] > [PL/I コンパイル設定] を選択して [EXEC プリプロセッサ オプション] の値
 ペ "PLITEST" を入力後、[OK] ボタンをクリックしてください。自動的に再ビルドが実行されますので、正常終了を確認します。

▲ Micro Focus ドルドパス	x86 [使用中]		
 ■ ビルド構成 BMS > PL/I コンパイル設定 > PL/I リンク設定 > アセンブミコンパイニ 	☑ 構成の固有な設定を可能にする(<u>C</u>)		
アセンブラリンカ	- 設定:		📃 コンソール 🛛 🔝 問題 🔲 プロパティー
1ヘント ▶ プロジェクト設定	フィルタテキストを入力		Micro Focus Build
Project Facets	設定	庙	
Server		12	post.build.cfg.New_Configuration:
> Task Repository		CICC	
lask lags		cics	
Validation	テバック用にコンバイル (-debug)	はい	BUILD SUCCESSFUL
WikiText	リストファイルを出力 (-1)	いいえ	Build finished with no errors.
ビルダー	最適化レベル (-opt)	-noopt	
プロジェクト参照	エンディアン (-bigendian)		Total times & seconds
実行/デバッグ設定	EXEC プリプロセッサ オプション (-optexec)	PLITEST	→ Iotal time: 0 seconds

3) 前項の CICS 実行と同様に、TN3270 エミュレーターから「TTEA」を入力して Enter キーを押すと Codewatch デバッガが自動的に立ち上がってきます。

CodeWatch -								- ×		
<u>File</u> <u>E</u> dit <u>V</u> iew	Eile <u>E</u> dit <u>V</u> iew <u>A</u> ctions <u>H</u> elp									
💠 🔂 🚯 🕻	P Src:	C:#work#CICSDEMO#CICS#STAFF02pli	~	Env:	ST	AFF02 V	Find:			
22: /* 23: /*dea 24: /* (25: /* / 26: /* 27: /* 1 28: /* a 29: /*dea 30:	okokokokokokokokokokokokokokokokokokok	*/ */ (C) Micro Focus 1885-2013. */ ecode is supplied for demonstration purposes only on */ basis and is for use at your own risk. */ ************************************		^	 7 6 5 4 3 2	Owner Current STAFF02 MPLICPPN_STAFF02 pi_main 0x21e1000 0x1f73d10 0x1f73d10		From STAFF02#KENTRY MPLUCPPN_STAFF02#88 [pi_main+0x5a 0x21e1000 0x1f73d10 0x1f73d10 0x1f73d10 0x1f73d10		
32: D0 33: 0	L OMM AREA	CHAR(100) BASED(COMM PTR):			1	0×1f73d10				

4) ステップインを行うことにより、ステップ実行が可能です。

<u>F</u> ile	File <u>E</u> dit <u>Vi</u> ew <u>A</u> ctions <u>H</u> elp								
\$	0 +	የ)	Src:	C#work#CICSDEMO#CICS#STAFF02pli	~	Er			
	122: 123:	Step ir	to (F11)	VC.MENU_SELECT = 0; UC.SCRN_NUM = '1';					
	124:		COMM_STF	NC.ERR_MSG = (70)'';					
	125:		COMM_STR	NUC.RETURN_TU = (8)'''; NUC WHO EROM = (8)''':					
	127:		COMM_STF	UC.QUEUE_NAME = Q_NAME;					
	129:	/***		OBTAIN HOME STATE FROM EA301 *****/					
	130:								
	131:		HOME_STA	(TE_CODE = 113);					

5) ステップアウトを行い、アプリケーションを先に進めると前項同様の画面が表示されますので、再度 "WILSON" を入力して Enter キーを押します。

	Staff Directory — Search Criteria
	Type a name, city, or ZIP. Type name as last name, first name.
	Name: <u>WILSON</u>
	City:
수 규 관 (아 Src: C¥Users¥tar	State: <u>IL</u>
$\begin{array}{ c c c c } \hline 22: /* & \text{Step out (shift F11)} \\ \hline 23: /* & Action of the state of the stat$	ZIP:

6) 再度、Codewatch 画面が表示されますので、ステップインします。コード内の変数へマウスオーバーして右クリックし、 [View contents of '変数名'] を選択すると、値がポップアップウィンドウで参照できます。

TTEM MINU - 1.			Contents of 'ITEM_NUM'
EXEC	Toggle breakpoint Enable breakpoint		ITEM_NUM = -31998 {fixed binary (15)}
	View contents of 'ITEM_NUM'		Ok
	Add 'ITEM NUM' to Evaluation panel	\rightarrow	

7) 変数値を常に監視したい場合には、変数へマウスオーバーして右クリックし、 [Add '変数名' to Evaluation panel] を 選択すると、右側に表示されているパネルへ常に表示されるようになります。

TTEM NU	M - 14	
I TEM_NO	То	ggle breakpoint
EXEC CI	En	able breakpoint
	Vie	ew contents of 'ITEM NUM'
	Ad	d 'ITEM_NUM' to Evaluation panel

8) ステートメントの行をダブルクリックすることによりブレークポイントの設定が可能です。ブレークポイントには、該当行の左端に 赤丸が表示されます。解除も同様にダブルクリックを行います。

PRFNAMESO = DDP_STRUCB.PREF_NAME;



9) 何度か [Continue] アイコンをクリックすると、情報表示画面が表示されます。

	Micro Focus Rumba - 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) ヘルプ(H)	
	D • 🤜 🖏 🗟 🖓 // D D D 🔊 🚳 • 🔕 D D D 🗞 // 🖧 🌽 🐺 // 🔣 🕑 📗	
		B
	Staff Directory - Staff Information Summary	
	Name: WILSON, MICHAEL Pref: MIKE	
	Designations: CLU, FIIC	
	Mailing address: 1115 BROADWAY AVE. SUITE A2	
	City: LINCOLN State: IL ZIP: 24061-3482	
	Office address: 1115 BROADWAY AVE. SUITE A2	
File Edit View Act	City: LINCOLN State: IL ZIP: 24061-3482	
아 한 한 수	Phone: 000-000-0000 000-000-0000	
	PF2=END PF3=Search Criteria Ac	cept
Continue (F5) →	レディ 実行中 SSL が NUMFLE NETCO00 OVR CAP NUM W 1,1 127.4	0.0.1

10) TN3270 エミュレーターを切断して、デバッグを終了します。

3.10 終了処理

1) [サーバー エクスプローラー] 内で [STAFF] を右クリックして [停止] を選択し、開始中のサーバーを停止します。



2) [STAFF] サーバーの停止状態を確認後に、Eclipse を終了します。

WHAT'S NEXT

- メインフレーム PL/I 開発: CICS Eclipse 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。