Micro Focus Enterprise Developer チュートリアル

メインフレーム PL/I 開発: IMS

Eclipse 編

1. 目的

本チュートリアルでは、Eclipse を使用した PL/I プロジェクトの作成、コンパイル、JCL とオンライン処理の実行、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。

2. 前提

• Windows 開発環境に Enterprise Developer 6.0 for Eclipse がインストール済であること。

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. 実行ファイルの生成
- 6. Enterprise Server インスタンスの設定
- 7. Enterprise Server インスタンス開始と確認
- 8. IMS データの作成
- 9. IMS トランザクションの実行
- 10. PL/I ソースのデバッグ
- 11. 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リソースを用意します。

- 1) Eclipse のワークスペースで使用する work フォルダーを 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ー配下に含まれている例題プログラム IMSDEMO フォルダーを、作成した C:¥work ヘコ ピーします。
 - 例) C:¥Users¥Public¥Documents¥Micro Focus¥Enterprise Developer¥Samples¥PLI-Eclipse¥IMSDEMO



3.2 Eclipse の起動

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



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。

📴 Eclipse Launcher		×
Select a directory as workspace		
Eclipse uses the workspace directory to store its preferences and development artifacts.		
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)		
Recent Workspaces		
ОК		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。

	500××8= &
Enterprise Developer for Eclipse	にようこそ
 	ブァースト・ステップ ファースト・ステップ ファースト・ステップ ファースト・ステップ ファースト・ステップ ファースト・ステップ マック マッ マッ マッ マッ マック マック マック マック マック マック マッ マック マッ マッ マッ マッ マック マッ マッ
Web リノース Web 上の器鍵目的	マイグレーション 彩泉ルースへのマイグレーション
Open Team Developer Perspective Team Developer Perspective Team Developer 1-スペクラィブを加えます。 このバースペクラィブをは、 パインレームのリンースへのアウセスを開始してあるとかのフール本 COBOL たんざやエノブシームの定義的であるためのワールを解除し、	Open COBOL Perspective COBOL / ース・ペクチィンを描述ます。このパース・ペクチィンでは、COBOL アプリアーンペンド研究するためのワールを提供します。
URT.	Open PL/I Perspective R.11パースペクティンを開きます。このパースペクティンをは、R.11アッリ ケーションの解除するためのケールを提供します。

4) 後述でビルドを行うため、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)			プロジェ プロジェ	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)	Ctrl+B	8) すべて プロジン	ビルド(A) エクトのビルド(B)		Ctrl+B
	クリーン(N) 自動的にビルド(M)	•		ワーキ	ング・セットのビルド ン(N)	(W)	,
×			\rightarrow	目動的	9ICE/UF(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] > [指令の自動 確定を実行] 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。

MF 設定	— D X
フィルタ入力 ・ Micro Focus > COBOL > Enterprise Server ISPF Dialog > ICI	指令の確定 指令の確定の設定 アイルのスキャン時に設定する指令を選択します。 アイルには、フロンアクト設定と異なる指令のみが設定されます。 フマイルには、フロンアクト設定と異なる指令のみが設定されます。 フマイルには、アロンアクト設定と異なる指令のみが設定されます。
 > Iディタ 1ディタ 指令の確定 Xディスプレイ > サービスインターフェイス > データベース 	□. 読者の自動電気を受け 手動による指令の確定は、以下で1 個以上が選択されている場合にのみ許可されます。 SQL ☑ CKEC SQL を含むアメイルに SQL 指令を設定する ☑ EXEC SQL を含えないファイルに SQL 指令の設定を解除する CICS CICS
デバッグ テンプレート ビルダー リモート JRE 検索) 統合化トレース機能	 ☑ EXEC CICS を含むファイルに CICS 指令を設定する ☑ EXEC CICS を含まないファイルの CICS 指令の設定を解除する マクロ ☑ マクロ文を含むファイルにマクロ指令を設定する ☑ マクロ文を含まないファイルのマクロ指令の設定を解除する
> Mylyn > Oomph > Remote Systems > Server > Terminal Validation	
→ Web	デフォルトの復元(1) 週用(1) OK キャンセル



3.3 メインフレーム PL/I プロジェクトのインポート

1) 用意した例題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 [一般] > [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。

() インポート		×
選択 アーカイブ・ファイルまたはディレクトリーから新規プロジェクトを作成します。	Ľ	5
インボート・ウィザードの選択(<u>S</u>):		
 フィルタ入力 アー設 アーカイブ・ファイル ファイル・システム フォルグーまたはアーカイブ由来のプロジェクト 授長デフロジェクトをワークスペースへ 説定 		<
(ア) (マス) (ア) (P) (P)<	キャンセ	JL

2) [ルート・ディレクトリの選択] へ C:¥work¥IMSDEMO を指定すると、このフォルダー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。

④ インポート	—		×
プロジェクトのインボート 既存の Eclipse プロジェクトを検索するディレクトリーを選択します。			
 ・ディレクトリーの選択(①): C:¥work¥IIMSDEMO ・アーカイブ・ファイルの選択(Δ): ・ ・<!--</td--><td>~</td><td>参照() 参照(]</td><th><u>R</u>)</th>	~	参照() 参照(]	<u>R</u>)
MS(Cł¥work¥IMSDEMO)	選抄	すべて選択 Rをすべて角 更新(F)	?(<u>S</u>) 祥除(<u>D</u>)
オブション コネストレたプロジェクトを検索(出) コプロジェクトをワークスペースにコピー(C) Close newly imported projects upon completion ロワークスペースに既に存在するプロジェクトを隠す(i)			
ワーキング・セット			
□ ワーキング・セットにプロジェクトを追加(<u>T</u>)		新規(<u>W</u>).	
ワーキング・セット(<u>0</u>):	~	選択(<u>E</u>)	
? <戻3(B) 次へ(N) > 終了(E)		キャント	セル

3) [PL/I エクスプローラー] にインポートしたプロジェクトが表示され、PL/I ソースや JCL 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

プロジェクトの設定値を確認していきます。

- 1) IMS プロジェクトを右クリックして [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 ビット稼働が指定されていますが、リホスト後に 32 ビット で実行することを想定して 32 ビット稼働へ変更します。
 - ・ 左側ツリービュー [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
▷ ビルド構成 ▷ 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。	
D. & 🖉 🛩 🗰	
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」 フォルダー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: bin¥debug

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、32 ビット稼働する実行可能ネイ ティブライブラリを実行ファイルタイプとして生成することがわかります。

x86 [使用中]	~ 構成
設定: フィルタテキストを入力 []	
 設定 一般的なオプション 	値
ブラットフォーム ターゲット 出力タイプ	32 bit すべて実行可能ネイティブライブラリ
	×86 [使用中] 設定: マイルタテキストを入力 設定 マ 一般的なオブション ブラットフォームターゲット 出力タイブ リーク



3) [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、例題の内容に沿って、 [システム] には IMS が設定されており、デバッグ実行用ファイルを生成することがわかります。

イベント		
ビルド環境	設定	値
▼ プロジェクト設定	マ 一般的なオプション	
> BMS	システム	IMS
> IMS	デバッグ用にコンパイル (-debug)	はい
> PL/I コンパイル設定	データ収集の有効化 (-dc)	はい
> アセンブラ コンパイラ	リストファイルを出力 (-1)	いいえ
アセンブラリンカ	最適化レベル (-opt)	-noopt
1/11 L'122477	-	

4) IMS データベースのパスを変更します。[Micro Focus] > [プロジェクト設定] > [IMS] を選択して [IMS データベース] 項目のパスへ、作成した C:¥work¥IMSDEMO を指定して [適用] ボタンをクリックします。



5) MFS のパスを変更します。 [Micro Focus] > [プロジェクト設定] > [IMS] > [MFS 設定] を選択して [MFS 出 力] 項目へ pliims_base を指定して [Apply and Close] ボタンをクリックしてウィンドウを閉じます。

	ノリノロセッサを使用	
> MFS 設定 > PSB 設定	STACK 処理 ゴリゴロセッサも使用	
> DBD 設定	詳細	いいえ
✓ IMS	メッセージ レベル	
> BMS	リストファイルを生成	はい
ビルド環境 ・ プロジェクト設定	設定 V MFS	值

6) ADDRBOOK.pli ファイルにデバッグモードが指定されていますので、この指定を削除します。ADDRBOOK.pli ファイルを 右クリックして [プロパティ] を選択します。 [PL/I コンパイル設定] > [マクロ プリプロセッサ] を選択して [マクロ定義 (-define)] 項目に指定されている debugit を削除して [Apply and Close] ボタンをクリックします。後述で実施する デバッグ時にはこの値を設定しますので、テキストエディターなどへ退避しておいてください。





3.5 実行ファイルの生成

1) 入力項目の桁数を揃えるため、サンプルプログラム [ADDRBOOK.pli] のコードを修正します。 [PL/I エクスプローラー] に表示されている該当ソースをダブルクリックしてソースコードを表示します。

		ADDRBOOK.pli 1
✓ ご IMS < ○ (品) PL/1ソースファイル > (品) ADDRBOOK.pli > (品) PLIIMSDB.pli	\rightarrow	<pre> /* Copyright (C) 1985-2013 Micro Focus International Ltd. */ /* All rights reserved. */ /* All rights reserved. */ /* ADOR800K: PROC(P_PCB_LT,P_ALT_PCB,P_DEMO_PCB) OPTIONS(MAIN); DCL P_PCB_LT POINTER; DCL P_DEMO_PCB POINTER; DCL P_DEMO_PCB POINTER; </pre>

2) 文字列操作のコードを一部変更します。 327 行目と 342 行目の "2"を "1" に変更して保存します。

右下に行数と桁数が表示されますので参考にしてください。 323:65			
【327 行目】			
<pre>NA_ADDRESS = trim(substr(work_string,1,index(work_string,',') - 1)); work_string = substr(work_string,index(work_string, ',') +2; </pre>	<pre>NA_ADDRESS = trim(substr(work_string,1,index(work_string,',') - 1)); work_string = substr(work_string,index(work_string, ',') +1);</pre>		
【342 行目】			
<pre>NA_CITY = trim(substr(work_string,1,index(work_string,',') - 1)); work_string = substr(work_string,index(work_string, ',') +2); →</pre>	<pre>NA_CITY = trim(substr(work_string,1,index(work_string,',') - 1)); work_string = substr(work_string,index(work_string, ',') +1;</pre>		
【保存】			
ファイル(F) 編集(E) ソ ご マ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □			

3) プロジェクトのビルドを行います。 [PL/I エクスプローラー] に表示されている [IMS] プロジェクトを右クリックして [プロジェ クトのビルド] を選択します。



4) [コンソール] タブでビルドの成功を確認します。ビルドのログを見ると DBDGEN や PSBGEN が成功していることが確認 できます。

📃 ンソール 🛿 訳 問題 🥅 プロパティー 🖄 コード分析
Micro Focus ビルド: IMSDEMO
nature.specific.build.cfg.x86:
post.build.cfg.x86:
combinedbuild.cfg.x86:
BUILD SUCCESSFUL
Build finished with no errors, 4 warnings.



5) プロジェクトの bin¥debug フォルダー配下に目的の実行ファイルが、pliims_base フォルダー配下には GEN ファイルが生成されていることを確認してください。



6) IMS トランザクションを生成します。 Windows メニューの [Enterprise Developer for Eclipse] > [Enterprise Developer コマンドプロンプト (32 bit)] を選択します。

9	Ente	erprise Developer for Eclipse	8°
	Ent	Enterprise Developer コマンドプロン	ップト (32-bit)

7) コマンドプロンプトで pliims_base フォルダーへ位置づけます。

コマンド) cd C:¥work¥IMSDEMO¥pliims_base

(C:)	> work > IM	SDEMO >		
	名前	^		
*	pliims_bas	e		
C:¥	Users¥taro	ot¥Documen	nts>cd c:¥work¥imsc	lemo¥pliims_base
c:¥	/work¥IMSDE	EMO¥pliims	_base>_	

8) ツールを起動させます。

コマンド) mfims imsgen



9) メニューが表示されますので [Selection] へ "1" を入力して [Enter] キーを押下します。

MFIMS -		×
IMS Option IMSGEN main menu Version 3.0.0.4		
 Define Transaction Codes Define Logical Terminals System Configuration menu Define Database Catalog DBDGEN file maintenance XDBDGEN file maintenance SKORD File maintenance SKORD File maintenance Reorganize DBOEN file Reorganize DBOEN file Reorganize MESGEN file 		
Selection: 1 Start listing with member name: (Op Use IMSLIBx variable setting : 1 (1-	otional) 25)	
Press <mark>Enter</mark> to invoke selection - F1=Help - F3/Esc=exit 2-B		



10) IMSGEN2.DAT ファイルがないとのメッセージが表示されますので、作成の意味を持つ "C" を入力します。

IMSGEN2.DAT NOT FOUND Press ENTER to (C)reate file, or (Q)uit

11) 入力画面が表示されます。[Tab] キーで項目を移動しながら以下のように入力後 [Enter] キーを押下します。

項目名	入力値
OPT	A
PROGRAM TYPE	MPP
TRANCODE NAME	ADDRBOOK
PCB NAME	ADDRBOOK
PROGRAM NAME	ADDRBOOK
SPA SIZE	0
PROCESS LIMIT CNT	99
MAX SEGNO	255
CHARSET	А
Can MFIMS	- 🗆 X
IMSOption IMSLIB=Current Directory	
OPT PROGRAM TRANCODE PSB NAME P TYPE NAME A MPP ADDREDOK ADDREDOK A	ROGRAM SPA SIZE PROCESS MAX CHARSET NAME LIMIT ONT SEGNO DORBOOK 0 39 255 A

12) 入力が完了したら [Esc] キーを2回押下してツールを終了します。

Enterprise Developer コマンドブロンプト (32-bit) :: ¥work¥1MSDEMO¥p | i i ms_base>

13) C:¥work¥IMSDEMO¥pliims_base に IMSGEN2.DAT が作成されています。

' (C:)	> work > IMSDEMO > pliims_base
^	名前
	IMSGEN2.DAT

3.6 Enterprise Server インスタンスの設定

1) PL/I を実行するためのエンジンを搭載した Enterprise Server インスタンスを作成します。 [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	- エクスプローラー 🛙		~	P
▷ 📃 ローカル [localhost:86				
	#/7981FDX(IN)			
	Administration	1 ページを開く	Ctrl+F	3



 C:¥work¥IMSDEMO には Enterprise Server インスタンスのサンプルが含まれておりますので、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¥IMSDEMO¥pliims_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¥IMSDEMO¥pliims64_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 (ページ 1 / 4):



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、[PLIIMS] とい う名前の 32 ビットアプリケーション稼働用 Enterprise Server インスタンスが追加されます。





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 [動的デバッグを許可] 欄のチェックをオンにします。これにより、Eclipse から動的デバッグが行えるようになります。

動的デバッグを許可: 🔽

6) 前項と同じタブの [構成情報] 欄を下記のように変更し、[適用] ボタンをクリックします。

変更前;

```
[ES-Environment]
PROJECT=C:¥Users¥Public¥Documents¥Micro Focus¥Enterprise
Developer¥Samples¥"PLI-VS or PLI-Eclipse"¥IMSDEMO
ES_SSTM_IMS="$PROJECT¥sstmims.jcl"
```

変更後;

[ES-Environment] PROJECT =C:¥work¥IMSDEMO ES_SSTM_IMS="\$PROJECT¥sstmims.jcl"



7) [サーバー] > [プロパティ] > [MSS] > [JES] > [一般] タブで表示される設定値を確認します。

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、オフ指定します。
ジョブ入力サブシステム 有効	[JES] タブ配下の設定をオン、オフ指定します。
JES プログラム パス	実行ファイルが存在するパスを指定します。
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を指定します。
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: 🗹

$CICS(\checkmark) \qquad JES(\checkmark) \qquad IMS(\checkmark) \qquad PL/I(\checkmark)$
→般 イニシエータ(1) プリンター(0)
ジョブ入力サブシステム有効: 🗹
JESプログラムパス: \$PROJECT\bin\debug
システム カタログ: \$PROJECT\pliims_base\catalog.dat
データセットの省略時ロケーション:
<pre>\$PROJECT\pliims_base\</pre>
システム プロシージャライブラリ: SYS1.PROCLIB

8) [サーバー] > [プロパティ] > [MSS] > [JES] > [イニシエータ] タブでイニシエータ定義を確認します。 A ~ 9 までの クラスに対するイニシエータが設定されています。



9) [サーバー] > [プロパティ] > [MSS] > [IMS] > [一般] タブで表示される設定値を確認します。

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。
デフォルト コードセット	使用するコードセットを指定します。 (ASCII or EBCDIC)
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。
Gen ファイル ディレクトリ	DBDGEN2 などの GEN ファイルが存在するパスを指定します。



一般	DB TM
IMS 有效	h: ☑
デフォル	トコードセット: ASCII 🗸
ACB 77 \$PROJE	マイル ディレクトリ: CCT\pliims_base
Genファ \$PROJE	イルディレクトリ: ECT\pliims_base

10) [サーバー] > [プロパティ] > [MSS] > [IMS] > [DB] > [一般] タブで表示される設定値を確認します。

項目名	説明
データベース パス	IMS データファイルが存在するパスを指定します。
一般 DB TM 一般 圧縮 ロック 復旧 データベースパス: \$PROJECT\pliims_base	

11) [サーバー] > [プロパティ] > [MSS] > [IMS] > [TM] > [一般] タブで表示される設定値を確認します。

項目名	説明
MFS パス	MFSGEN により生成されるファイルのパスを指定します。
アプリケーション パス	プログラムの実行ファイルが存在するパスを指定します。
 一般 DB TM 一般 MPR (1) メッセージ キュー: デフォルト キュー名を使用: ☑ 名前: 最大プロック数: 32 キュー、バッファ数: 2 コールドスタート: None MFS N(2: SPROJECT\plims_base MFS mult 文字: ③ 文字: ④ 16進: 0x 1A キーボイント生成頻度: 64 末尾スペース: トランザクションしきい値: 0 (秒) 	
アプリケーション パス: \$PROJECT\bin\debug	

12) [サーバー] > [プロパティ] > [MSS] > [IMS] > [TM] > [MPR] タブで表示される設定値を確認します。 メッセージ処理領域の定義が一覧で表示されます。ここでは 1 つのメッセージ処理領域が既に定義されています。



CICS (✔) JES (✔)	IMS (✔)				
一般 DB TM					
一般 MPR (1)					
名前	クラス				
編集 MPR01	1				

13) [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、オフ指定します。
CodeWatch $M = 7 l^{6}7$	CodeWatch デバッガで使用するソースファイルパスを指定します。
	本チュートリアルでは Eclipse デバッガを使用します。
CodeWatch CTP 11°7	CodeWatch デバッガで使用するデバッグファイルパスを指定します。
	本チュートリアルでは Eclipse デバッガを使用します。
PL/I 構成ディレクトリ	プロジェクトのパスを指定します。

$\operatorname{CICS}\left(\overset{\checkmark}{}\right) \hspace{0.1cm} \operatorname{JES}_{\cdots}\left(\overset{\checkmark}{}\right) \hspace{0.1cm} \operatorname{IMS}_{\cdots}\left(\overset{\checkmark}{}\right) \hspace{0.1cm} \begin{array}{ llllllllllllllllllllllllllllllllllll$	(C:) > work > IMSDEMO > bin > debug
一般 PL1有効: ☑ PL1557 7タッチのプロンプト □ Codewatb ノース バス: Ismos zec	名前 ^ # @ ADDRBOOK.adt # ① ADDRBOOK.def # @ ADDRBOOK.dll
Codematch STB バス: SPROJECT\bin\Debug PLT 構成ディレクトリ: SPROJECT	ADDRBOOK.exp ADDRBOOK.lib ADDRBOOK.obj ADDRBOOK.obj.1.tlog ADDRBOOK.pdb ADDRBOOK.stb

14) TN3270 エミュレータと接続するリスナー設定を確認します。

[リスナー] タブをクリックして、[TN3270] リスナーのポート番号が [5150] であることを確認します。



15) 画面左上の [Home] をクリックして一覧画面に戻ります。





3.7 Enterprise Server インスタンスの開始と確認

- 1) サーバーエクスプローラ内に [PLIIMS] インスタンス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] を右クリック後 [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内の [PLIIMS] インスタンスを右クリックし、[プロジェクトに関連付ける] > [IMS] を選択します。こ れにより Eclipse 内の [IMS] プロジェクトから実行するアプリケーションは [PLIIMS] インスタンスを対象として実行され ます。

プロジェクトに関連付ける
・
く
IMS

3) [PLIIMS] インスタンスを右クリックして [開始] を選択します。

		新規作成(N)
>	PLIIMS	開始

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。

MF Enterprise Server サ	インオン	\times
サーバーの接続詳細を入	力します: PLIIMS	
□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 [PLIIMS] インスタンス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。

画面	● Show entries from 1 to 10 更新 ● Show last 10 lines of 77 total entries
Entry	Event Show Entire Log
58	180129 11141303 8788 PLIIMS CASSI1744I MFPLI support loaded successfully 11:14:12
9	180129 11141304 8932 PLHIMS CASSI1744I MFPLI support loaded successfully 11:14:12
0	180129 11141304 8788 PLIIMS CASIP0005I IMS Message Processing Region started for classes "1" 11:14:13
1	180129 11141307 8932 PLIIMS CASBJ0005I Batch initiator started for job classes "ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789" 11:14:13
2	180129 11141331 6972 PLIIMS CASSI1734I MFIMS Logon Successful 11:14:13
3	180129 11141331 8788 PLIIMS JES000040I SSTM environment established successfully. (IMS) Job #: 0001001 11:14:13
4	180129 11141334 6972 PLIIMS CASSI1744I MFPLI support loaded successfully 11:14:13
5	180129 11141334 8788 PLIIMS CASIP0013I IMS Message Processing Region registered for classes "001" 11:14:13
6	180129 11141335 6972 PLIIMS CASSI1600I SEP initialization completed successfully 11:14:13
	180120 11141359 6972 PLIIMS CASSI50211 PLTPL Phase 2 - No PLT Specified 11:14:13

いくつかのサービス開始が失敗してもインスタンスは開始されますので、ログ内容を必ず確認してください。

3.8 IMS データの作成

1) IMS データベースへデータを挿入する JCL を確認します。 [PL/I エクスプローラー] に表示されている [pliimsdb.jcl] をダブルクリックしてコードを確認します。

【STOP01 ステップ】 対象の IMS データベースを停止します。

【ZERO01 ステップ】対象の IMS データベースの ZEROLOAD を実行します。

//STOP01 EXEC PGM=MFDBUJCL,PARM='/STO DB ADDRBOOK'
//SYSOUT DD SYSOUT=*,HOLD=Y
//*
//ZER001 EXEC PGM=MFDBUJCL,PARM='ZEROLOAD,ADDRBOOK'
//SYSOUT DD SYSOUT=*,HOLD=Y

【BBOJ12 ステップ】 SYSUT1 のデータが IMS データベースへ登録されます。

//BB0J12 EXEC // //BB0L0G DD // //IMSTRACE DD 9 //PLIDUMP DD 9 //SYSOUT DD 9 //SYSPRINT DD 9	PGM=DFSRRC00, PARMe (DLI, PLIIMSDB, F DSN=DMF.BBOLOG.ADDRE SYSOUT=*, HOLD=Y SYSOUT=*, HOLD=Y SYSOUT=*, HOLD=Y SYSOUT=*, DCB=(RECFM: *	PLIIMSDB) 300K, <mark>DISP</mark> =(NEW,CATLG -LSEQ,LRECL=0),HOLD=	,DELETE) Y			
ISRT Fahringer	Dave	1 Redneck Drive	Normal	IL61761		
ISRT Poole	Clive	2315 Sherman	Atlanta	GA69961		
ISRT Athwal	Davinder	2202 Shadow Ridge	San Jose	CA22231		
ISRT Belcher	Paul	1 Red Sox Way	Boston	MA33231		
ISRT Smith	Jim	22 Royal Watcher	London	0H43233		
ISRT Brewer	ISRT Brewer Anthony 92 Rodent Lane Santa Clara					
ISRT Fendick	Andrew	Caravan Keep	Lancaster	PA78383		
ISRT Rafiq	Irfan	87 Badminton	Newbury	UK78383		
ISRT Kemanetzis LIST /*	Charis	3 Headbanger Way	Newbury	UK78383		
【START01 ステップ】 対象の IMS データベースを開始します。						
//START01 EXEC PGM=MFDBUJCL,PARM='/STA DB ADDRBOOK' //SYSOUT DD SYSOUT=*,HOLD=Y						



 確認した JCL を実行します。 [PL/I エクスプローラー] に表示されている [pliimsdb.jcl] を右クリックして [Enterprise Server へのサブミット] を選択します。

A plumedbuck		
🖉 🖉 piinisubijei	Enterprise Server A (0#15wb	CHRUSZINS I
	Enterprise server (10) 9 7 2 7 1	CUT+271-+3,1

3) 「コンソール」タブに実行された JCL のジョブ番号が表示されますので、リンクをクリックして結果を確認します。

א-עעב 📃	🖳 問題	🔲 วือパティ	- 🔯	ード分析		X [à 🔠	₽ 🛃		- 🗂
Enterprise Server										
JCLCM0187I J0	001014	PLIIMSDB J	OB SU	BMITTED	(JOBNAM	IE=PLI	IMSDB,	JOBNUM=	00010	914)
JCLCM0180I J0	001014	PLIIMSDB J	ob rea	dy for (executio	on. 15	:46:49)		
Processed "C:	\work\I	MSDEMO\pli	imsdb.	jcl"						

4) ジョブが正常に終了したことを確認後、[BBOJ12] ステップの [SYSPRINT] をクリックすると登録したデータが確認できます。
 starting plijnsch 32 Bit ASCII

				Starting prinisub 02 bit Aborr	
х	SYSPRINT	BBOJ12	\rightarrow	demo_dbd_name: AUUKBUUK Successfully Inserted: Dave F:	ahringer
				Successfully Inserted: Clive P Successfully Inserted: Davinder A Successfully Inserted: Paul B Successfully Inserted: Jim S Successfully Inserted: Anthony B Successfully Inserted: Andrew F Successfully Inserted: Irfan R Successfully Inserted: Irfan R Successfully Inserted: Charis K Successfully Inserted: Charis Alpha Order(oole thwal elcher mith rewer endick afiq emanetzi last Nam

5) [PL/I エクスプローラー] 直下の [pliims_base] を右クリック後 [更新] を選択してリフレッシュすると、生成された ADDRBOOK.DAT ファイルが確認できます。

~	🗁 pliims_base
	427FaDDRBK.DIF
	427FADDRBK.DOF
	ADDRBK.MFSX
	ADDRBOOK.ACB
	ADDRBOOK.DAT

次に、オンライン処理からデータの確認と登録を行います。

3.9 IMS トランザクションの実行

1) ご利用の TN3270 エミュレータを使用して、開始した PLIIMS インスタンスのポートへ接続します。本チュートリアルでは Rumba を使用して接続します。



CICS サインオン画面では USERID, Password 共に SYSAD でログインします。





2) IMS モードへの切り替えトランザクション名を入力します。

コマンド) /IMS

© ♥ メインフレームデ... × \ /IMS_

3) IMS モードの初期画面が表示されますので、これから実行するトランザクションを開始します。

コマンド) /STA TRAN ADDRBOOK

注意)コマンド最後尾に空白文字があります。

🖅 🎬 メインフレームデ	. × \
_	
DFS36501 Sessi	on status for IMS PLIIMS
Date: 01/30/20 Node Name: User: Preset destina	18 Time: 16:15:21 NETAOOO IPADDR: 127.0.0.1:11968 SYSAD tion:
Current sessio	n status: PAGE
No output secu	rity available
\downarrow	
🖅 🖉 🚆 メインフレー	۵デ ×
_STA TRAN	DDRBOOK
DFS0581 1	:19:05 START command completed

4) トランザクションを実行します。入力画面が表示され、JCL から登録したデータが確認できます。

コマンド) ADDRBOOK

注意)コマンド最後尾に空白文字があります。

₴ਫ਼∕ 🚆 メインフレームデ 🗙 🔪	
ADDRBOOK	
Micro Focus	(IP) Limited - Address Book
A =	Add, $D = De ete$
_ Davinder Athwal	2202 Shadow Ridge, San Jose, CA 22231
_ Paul Belcher	1 Red Sox Way, Boston, MA 33231
_ Anthony Brewer	92 Rodent Lane, Santa Clara, CA 45233
_ Dave Fahringer	1 Redneck Drive, Normal, IL 61761
_ Andrew Fendick	Caravan Keep, Lancaster, PA 78383
_ Charis Kemanetzis	3 Headbanger Way, Newbury, UK 78383
<pre>_ Clive Poole</pre>	2315 Sherman, Atlanta, GA 69961
_ Irfan Rafiq	87 Badminton, Newbury, UK 78383
_ Jim Smith	22 Royal Watcher, London, OH 43233

5) データを1 件追加します。 [Tab] キーで項目を移動し、最終行へ以下内容を入力後 [Enter] キーを押下します。

項目名	入力値
先頭項目	A=追加 D=削除。ここでは A を入力します。
入力データ	TARO TOKYO 1-1-1 AKASAKA, MINATO-KU, TK 12345
lim Smith	22 Roval Watcher London OH 43233

TARO TOKYO 1-1-1 AKASAKA, MINATO-KU, TK 12345

6) 入力データが登録されました。TN3270 エミュレータ接続を切断してください。

Jim Smith	22 Royal Watcher, London,	OH 43233
_ TARO TOKYO	1-1-1 AKASAKA, MINATO-KU,	TK 12345



3.10 PL/I ソースのデバッグ

TN3270 エミュレータから実行される PL/I プログラムをデバッグします。

1) 前項で削除した debugit を再指定して、プロジェクトを再度ビルドしてください。

∨ 高度	
インクルードファイルを含める (-incafter)	
デバッグ情報なし (-nodebuginfo)	いいえ
マクロを定義 (-define)	debugit
マクロを削除 (-undefine)	
バリアントを指定する (-variant)	

2) [実行] プルダウンメニューの [デバッグの構成] を選択します。

-y	-7	ナビゲート(N)	検索	プロジェクト(P)	実行(R)
R	実行	テ点をリセット			
Q,	実行	T(R)		Ctr	1+F11
检	デハ	ブッグ(D)			F11
	実行	亍履歴(T)			•
	実行実行	〒(S) 〒構成(N)			+
	デル	、ッグ履歴(H)			+
	デハ	、ッグ(G)			+
	デハ	、ッグの構成(B).			

3) 左側のツリービューから [PL/I Enterprise Server] を選択して、左上の [新規の起動構成] アイコンをクリックします。



4) [PL/I プロジェクト] へ対象となる IMS プロジェクトを入力し、[Enterprise Server] へ実行させる PLIIMS インスタンスを指定します。 [デバッグの種類] は [IMS] タブを選択した状態で、[デバッグ] ボタンをクリックします。

🥮 デバッグ構成					
構成の作成、管理、および実行 Enterprise Server アブリケーションへの接続と	テバッグ				Ť
 ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○	名前(<u>M</u>): 新規構成(1) <u>M</u> 一般 <u>4</u> ソース <u>一</u> 共: Enterprise Server 上でデバッグ PL/I プロジェクト(P) IMS	通(C) 😼 デバッグシンボル) 🥔 ユンラ セッションを開始して、 PL/I ブログラムの	・ナー 起動を待機します。		
PL/1のプリケーションのアタッチ(PL/1のプリケーションのアタッチ(PL/1のプロセスにアタッチ Wino JavaScript Standalone V8 VM WebKit Protocol XSL	 ▼ Enterprise Server ● 接続: ○ Kubernetes Pod: ○ Localhost エンドポイント ▼ デバッグの提留 	localhost ポート:	サーバー:	PLIIMS	
J ₁₀ ダスク・コンチネスト・テスト □ リモート Java アブリケーション ◆ リモートの JavaScript ■ 近期グループ □ 汎用サーバー □ 汎用サーバー(外部からの起動)	CICS (JCL IMS Web * - IMS 設定(空白の場合 ユーザー:	サービス) Java) はすべての IMS ジョブをデバッグ) ―			v
< > * * * * * * * * * * * * * * * * * *			前回保管し	た状態に戻す(⊻)	適用(<u>Y</u>)
0			I	デバッグ(<u>D</u>)	閉じる

5) デバッグタブで [アタッチ待機] 状態になったことを確認します。





デバッグタブが表示されていない場合は [ウィンドウ] > [ビューの表示] > [その他] を選択して表示されるウィンドウで [デ バッグ] > [デバッグ] を選択して [OK] ボタンをクリックするとデバッグタブが現れます。



- 6) 前項と同様に TN3270 エミュレータから接続を行い、ADDRBOOK トランザクションを実行します。
- 7) パースペクティブの切り替え確認ウィンドウが表示されますので、[はい] ボタンをクリックし、デバッグ用のパースペクティブを開きます。

MF パーフ	スペクティブ切り替えの確認 X				
\bigcirc	この種類の起動は、中断時にデバッグパースペクティブが開くように構成されています。				
このデバッグ・パースペクティブは、アブリケーション・デバッグをサポートするために設計されていますれには、デバッグ・スタック、変数、およびブレークポイント管理を表示するビューが組み込まれてす。					
	このパースペクティブを開きますか?				
 二常に 	この設定を使用する(R)				
	(<u>まい(y)</u> いいえ(<u>N</u>)				

8) 少し待つとデバッグセッションが開始して、プログラムのステップ実行が可能になります。[F5] キーもしくは [実行] 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、 [変数] タブでは使用している変数値が確認できます。

work - 77777 - iwiddeino/Abbitudookipii - eciipse						
ファイル(E) 編集(E) ソース リファクタリング ナビゲート(N) 検	索 プロジェクト(<u>P</u>) 実行(<u>R</u>) ウ	ィンドウ(<u>W)</u> ヘルプ(<u>H</u>)				
📑 🛨 🔛 🐚 🔛 🔛 🖎 🕨 💷 🛤 🌫 🦘 .	e 🗏 🔿 🗟 🕫 🏄 🛄	📲 🗣 🖻 🔳 🔳	1 🙀 🛊 🕶 🔘 🕶	🂊 + 🍅 😂 🛷 +		
£ - 2 - 4 + 4 + 4 +				クイック・アクセス	e 8 e e	*
参デバッグ 23 #6 Servers	% » × = =	(x)- 変数 💁 ブレークポイ	가 %(式 23	🏷 🤫 🕞 🍦	X 🔆 🗸 🖻	
 新規構成(4) [PL/I Enterprise Server] 		名前 > 詳 "work_string" > 詳 "NA FIRST NAM	值 <ou E'</ou 	t of Context>		^
ADDRBOOK: (17:5) MFOPENPLIIMS:		 NA_LAST_NAM NA_LAST_NAM NA_LAST_NA XY "NA_ADDRESS" 	E" IGu ME IGu			
		<			:	>
		L				^
						~
		(¢				2
🧾 サーバー: localhost 🛛 🔝 ADDRBOOK.pli 😒				計 アウトライン 😂		
ADDRROOK ali				い フィルタ		lla
1 ADDINGCOLPH				~		1.1
/** Copyright (C) 1985-2013 Micro Focus I /* All rights reserved.	nternational Ltd.	*****/ */ */ */		 ADDRBOOM P_PCB_ P_ALT_ P_DEM 	K LT POINTER PCB POINTER	^
DCL P_ALT_PCB LT POINTER; DCL P_ALT_PCB POINTER; DCL P_ALT_PCB POINTER; DCL P_DEMO_PCB POINTER;	_PCB) OPTIONS(MAIN);			TWO BI THREE FOUR B FIVE BIN	IN FIXED (31,0) INIT BIN FIXED (31,0) IN JIN FIXED (31,0) INIT N FIXED (31,0) INIT	і П Т ((5
DCL TWO BIN FIXED(31.0) INIT(2):				SUBSTR	BUILTIN	
DCL				OATE B	UILTIN	
THDEE RTN CTVEN/31 A) TNTT/3).			×	PI IRFT(C BUILTIN	. `
				¢		>
リーンパール XX 20 タスク			3	। 🗟 🚮 🐼 🛃	🕒 🕶 📩 🕶 🚍	
デパッグ メッセージ コンソール : [新規構成(4)] [プロセス ID: 912]						
Loaded: C:\work\IMSDEMO\bin\debug\ADDRBOOK.dll Loaded: C:\work\IMSDEMO\bin\debug\ADDRBOOK.dll	 Not compiled for debug Not compiled for debug 	ging				0
<						2
	1:	1				

9) [再開] アイコンをクリックしてデータ一覧を表示させます。





前項で追加したデータが表示されているのを確認できます。

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) オブション(O) ツール(T) ヘルプ(H)			
D-9 16 문 사이에 이 6 - 2 15 이 5 16 4 4 4 17 년 원 0 1			
王 / · · · · · · · · · · · · · · · · · ·			
Micro Focus (IP) Limited - Address Book A = Add, D = Delete Davides Athmai Paul Beleher Anthony Pereer Dave Fahringer Dave Fahringer Charle Kennenztzis Gharle Kennenztzis Charle Kennen			
1ディ 実行中 155 In MANUEL OVR CAP NUM W 4,2 127.0.0.	1	13.0	9.55

10) プログラムの 368 行目の左端をダブルクリックしてブレイクポイントを設定します。

	<pre>if index(work_string, ' ') > 0 then</pre>
۲	SSA_SEARCH.SSA_NA_FLDVALUE =
	<pre>trim(substr(work_string,</pre>

11) TN3270 エミュレータから、追加したデータ行へ "D" を入力して [Enter] キーを押下します。

	Micro Focus	(IP) Limited - Address Book
	A =	Add, $D = De ete$
	Davinder Athwal	2202 Shadow Ridge, San Jose, CA 22231
	Paul Belcher	1 Red Sox Way, Boston, MA 33231
	Anthony Brewer	92 Rodent Lane, Santa Clara, CA 45233
	Dave Fahringer	1 Redneck Drive, Normal, IL 61761
	Andrew Fendick	Caravan Keep, Lancaster, PA 78383
	Charis Kemanetzis	3 Headbanger Way, Newbury, UK 78383
	Clive Poole	2315 Sherman, Atlanta, GA 69961
	Irfan Rafiq	87 Badminton, Newbury, UK 78383
	Jim Smith	22 Royal Watcher, London, OH 43233
D	<u>T</u> ARO TOKYO	1–1–1 AKASAKA, MINATO–KU, TK 12345

12) プログラムの先頭に位置づけられますが、[再開] アイコンクリックしてブレイクポイントまで進みます。

<u>-</u> サ-	/(-: localhost 🕅 ADDRBOOK.pli 🔀
love love	ADDRBOOK.pli
	••••••1•••••2•••••3•••••4•••••5••••6•••
Θ	/* Copyright (C) 1985-2013 Micro Focus International Ltd. /* All rights reserved.
• 😑	ADDRBOOK: PROC(P_PCB_LT,P_ALT_PCB,P_DEMO_PCB) OPTIONS(MAIN); DCL_P_PCB_LTPOINTER:

13) 削除ルーチンへ入り、項目値の確認ができます。

ADDRROOK -E

Ī	when('D') do;			
Θ	<pre>/* Simple algorithm to try to find the "key" - aka last name */ work string = addrbook, field(ndx);</pre>	(×)= 変数 🕒 ブレークポイント 🚱 式 😒		
	<pre>work_string = trim(work_string); work_string = substr(work_string,index(work_string,' ') + 1);</pre>	名前 > ^{Xty} "work string"	值 TOKYO	1-1-1 AKASAKA MINATO-KU TK 12345
	if index(work_string, ' ') > 0 then	t ty hongsting	TOILTO	
- T	SSA_SEARCH.SSA_NA_FLDVALUE =	work_string	TOKYO	1-1-1 AKASAKA, MINATO-KU, TK 12345

14) 先に進めるために [再開] アイコンをクリックすると指定データが削除された一覧が表示され、デバッガはアタッチ待機状態に 戻ります。





15) TN3270 エミュレータの接続を切断後、デバッグを終了させるため、画面上部の終了アイコンをクリックします。



16) デバッガが停止状態になったのを確認後、右クリックして [終了したエントリをすべて削除] を選択し、これを削除します。

☆ デバッグ ♡	해 Servers							
✔ 😹 <終了しました>新規構成(4) [PL/I Enterprise Server]								
₀ [©] Mic	ro Focus デバッガ: (停止)	Sec.	終了したエントリをすべて削除(A)					

17) 右上のパースペクティブ切り換えで PL/I アイコンをクリックして、PL/I パースペクティブへ戻ります。



3.11 終了処理

1) サーバーエクスプローラ内で [PLIIMS] を右クリックして [停止] を選択し、開始中のインスタンスを停止します。

> 🔚 PLIIMS 停止

2) [PLIIMS] インスタンス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。