

---

# 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 4.0 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

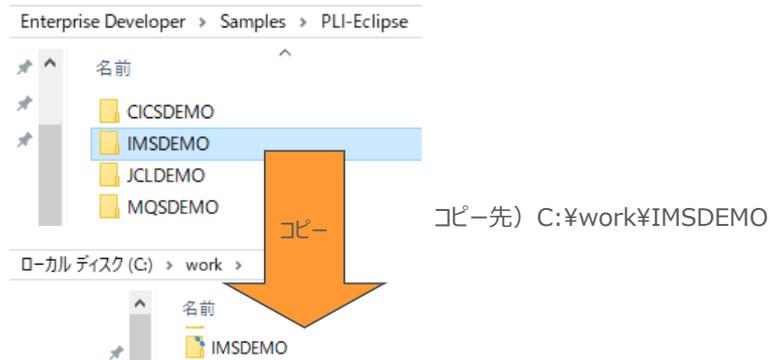
1. チュートリアルの準備
2. Eclipse の起動
3. メインフレーム PL/I プロジェクトのインポート
4. プロジェクトプロパティの確認
5. 実行ファイルの生成
6. Enterprise Server インスタンスの設定
7. Enterprise Server インスタンス開始と確認
8. IMS データの作成
9. IMS トランザクションの実行
10. PL/I ソースのデバッグ
11. 終了処理

### 3.1 チュートリアルの準備

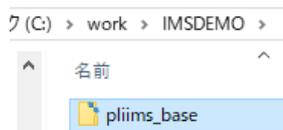
例題プログラムに関連するリソースを用意します。

- 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例題プログラム IMSDEMO 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¥IMSDEMO

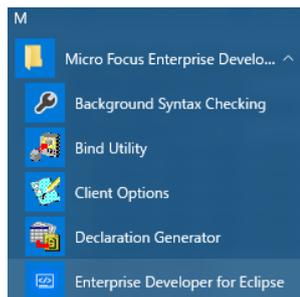


- 3) コピーした IMSDEMO フォルダの配下に pliims\_base フォルダを作成します。

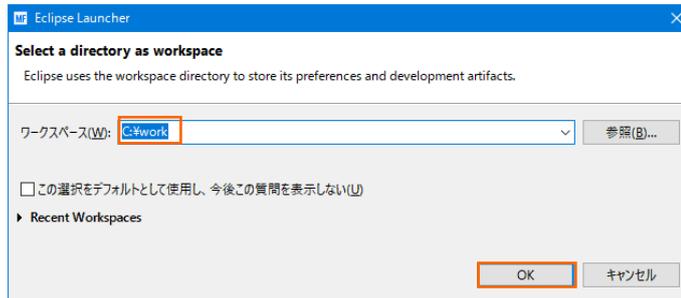


### 3.2 Eclipse の起動

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



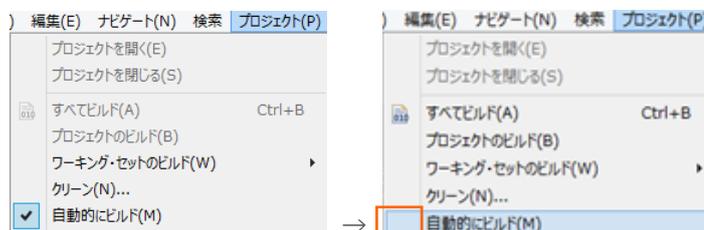
- 2) 前項で作成した C:\work をワークスペースへ指定して、[OK] ボタンをクリックします。



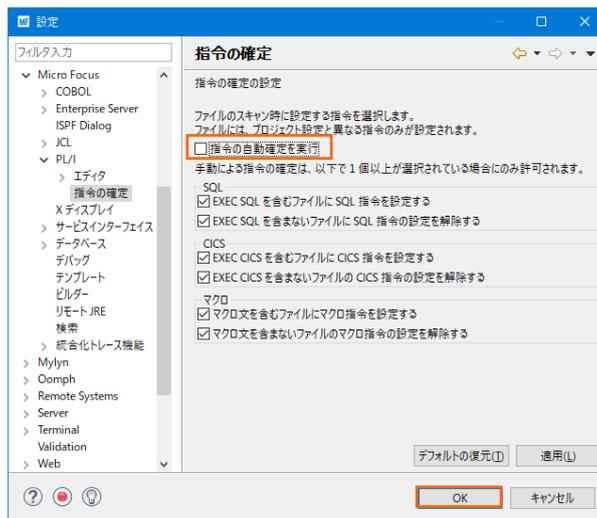
- 3) [ようこそ] タブが表示されますので、[Open PL/I Perspective] をクリックして、PL/I パースペクティブを開きます。



- 4) 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] ブ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。

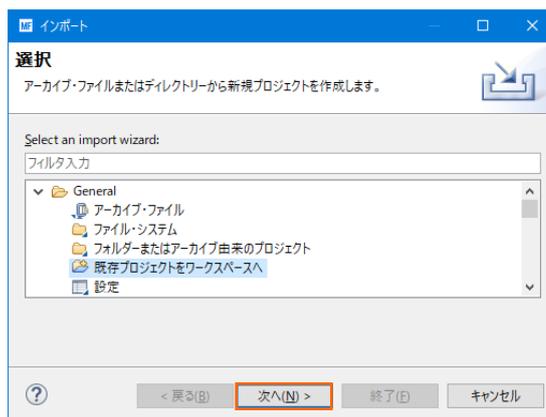


- 5) 既存ファイルのインポート時、自動的にコンパイル指令が指定される機能が用意されていますが、本チュートリアルではこれを解除します。[ウィンドウ] ブルダウンメニューの [設定] > [Micro Focus] > [PL/I] > [指令の確定] > [指令の確定を行う] チェックボックスをオフにして [OK] ボタンをクリックします。

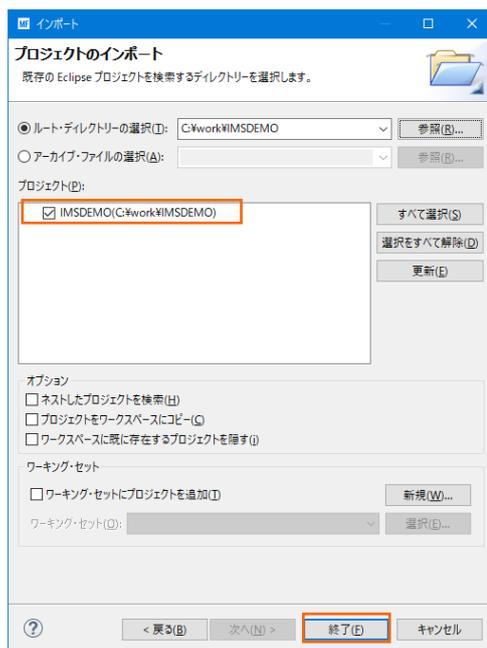


### 3.3 メインフレーム PL/I プロジェクトのインポート

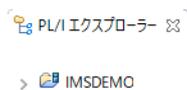
- 1) 用意した例題プロジェクトをインポートします。[ファイル] プルダウンメニューから [インポート] を選択し、インポートウィンドウにて [General] > [既存プロジェクトをワークスペースへ] を選択後 [次へ] ボタンをクリックします。



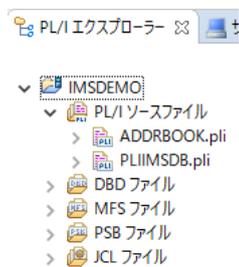
- 2) [ルート・ディレクトリの選択] ^ C:\work\IMSDEMO を指定すると、このフォルダーに含まれるプロジェクトが表示されます。チェックをオンにした状態で [終了] ボタンをクリックします。



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ます。



4) [IMSDemo] プロジェクトを展開すると 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。

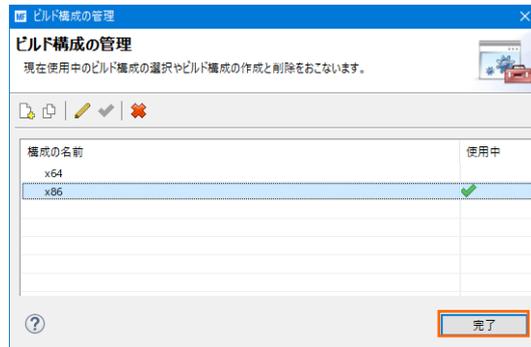
1) IMSDEMO プロジェクトを右クリックして [プロパティ] を選択するとプロパティウィンドウが表示されます。

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。

- ① 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。



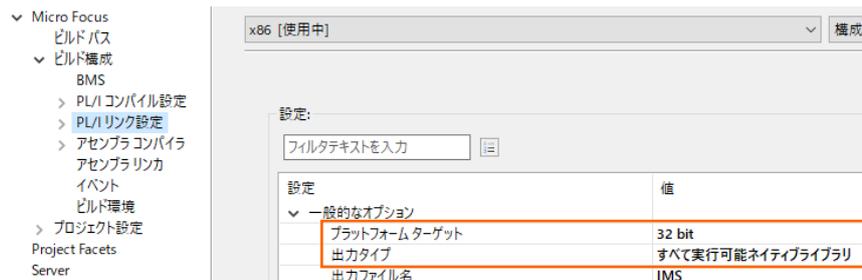
- ② [ビルドの構成管理] ウィンドウでは [x86] のチェックボックスをオンにして [完了] ボタンをクリックします。



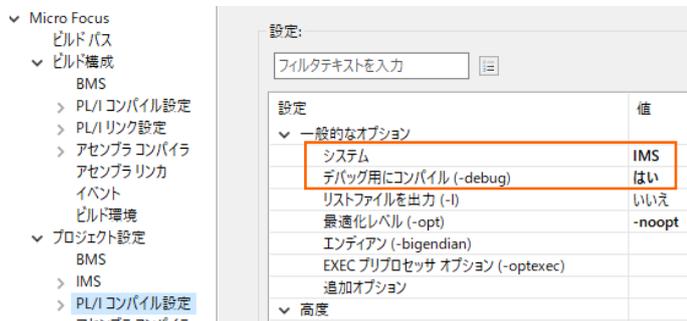
- ③ [Micro Focus] > [ビルド構成] ウィンドウへ戻り [x86] へ変更されたことと、プロジェクト配下の「bin\debug」フォルダーへ実行ファイルが出力されることを確認後 [適用] ボタンをクリックします。



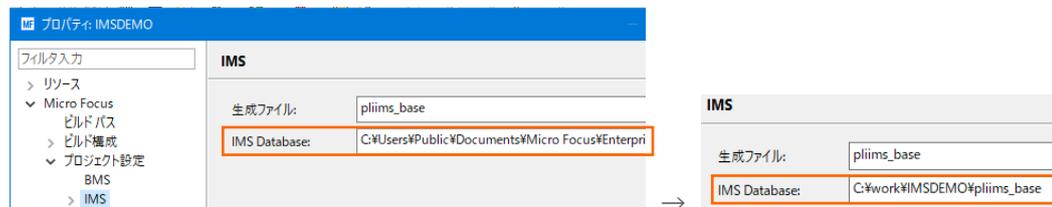
2) [Micro Focus] > [ビルド構成] > [PL/I リンク設定] を選択して内容を確認すると、32 ビット稼働する実行可能ネイティブライブラリを実行ファイルタイプとして生成することがわかります。



- 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。



- 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 Database] 項目の [参照] ボタンから、前項で作成した pliims\_base フォルダを選択して [適用] ボタンをクリックします。

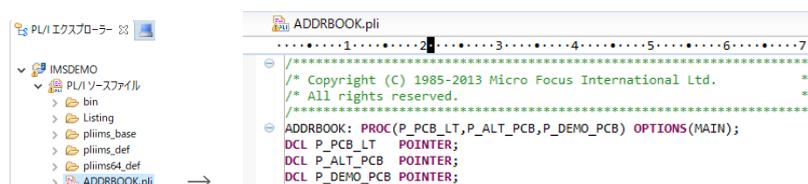


- 5) MFS のパスを変更します。[Micro Focus] > [プロジェクト設定] > [IMS] > [MFS 設定] を選択して [MFS 出力] 項目の [参照] ボタンから、前項で作成した pliims\_base フォルダを選択して [適用] ボタンをクリック後、[Apply and Close] ボタンをクリックしてウィンドウを閉じます。



### 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。



- 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】

```

NA_ADDRESS =
trim(substr(work_string,1,index(work_string,',') - 1));
work_string =
substr(work_string,index(work_string,',') + 2);
→
NA_ADDRESS =
trim(substr(work_string,1,index(work_string,',') - 1));
work_string =
substr(work_string,index(work_string,',') + 1);

```

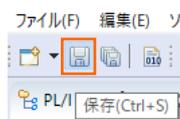
【342 行目】

```

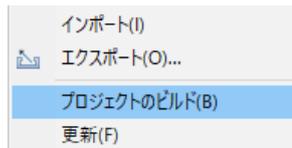
NA_CITY =
trim(substr(work_string,1,index(work_string,',') - 1));
work_string =
substr(work_string,index(work_string,',') + 2);
→
NA_CITY =
trim(substr(work_string,1,index(work_string,',') - 1));
work_string =
substr(work_string,index(work_string,',') + 1);

```

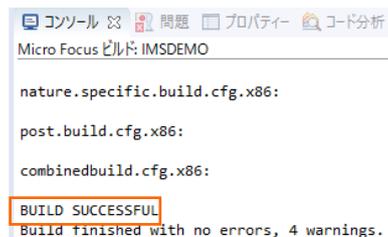
【保存】



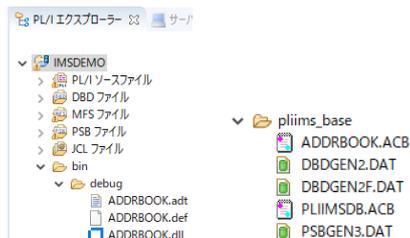
- 3) プロジェクトのビルドを行います。[PL/I エクスプローラー] に表示されている [IMSDEMO] プロジェクトを右クリックして [プロジェクトのビルド] を選択します。



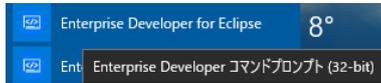
- 4) [コンソール] タブでビルドの成功を確認します。ビルドのログを見ると DBDGEN や PSBGEN が成功していることが確認できます。



- 5) プロジェクトの bin\debug フォルダ配下に目的の実行ファイルが、pliims\_base フォルダ配下には 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。



- 7) コマンドプロンプトで pliims\_base フォルダへ位置づけます。

コマンド) cd C:¥work¥IMSDEMO¥pliims\_base

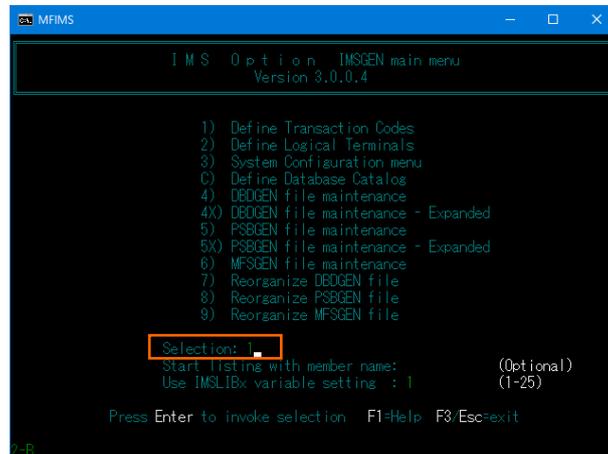


- 8) ツールを起動させます。

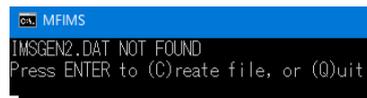
コマンド) mfims imsgen



- 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。

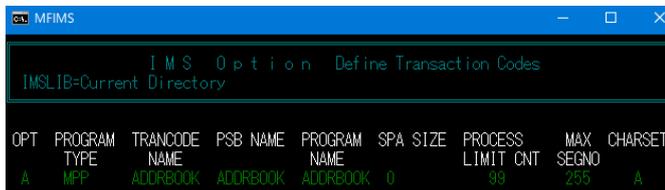


- 10) IMSGEN2.DAT ファイルがないとのメッセージが表示されますので、作成の意味を持つ “C” を入力します。



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。

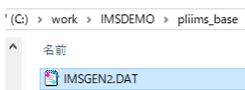
項目名	入力値
OPT	A
PROGRAM TYPE	MPP
TRANCODE NAME	ADDRBOOK
PCB NAME	ADDRBOOK
PROGRAM NAME	ADDRBOOK
SPA SIZE	0
PROCESS LIMIT CNT	99
MAX SEGNO	255
CHARSET	A



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。

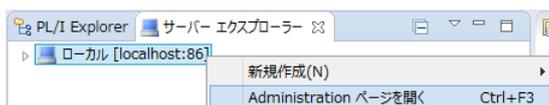


13) C:\work\IMSDEMO\pliiims\_base に IMSGEN2.DAT が作成されています。



### 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。



- 2) 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。

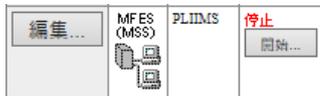
Enterprise Server Administration 画面左側の [インポート] をクリックして、表示される下記項目へ前述のパスを入力後、[次へ] ボタンをクリックします。

サーバー情報のインポート (ページ 1 / 4):

復旧するサーバー データが含まれるソース ディレクトリの選択:

file:///

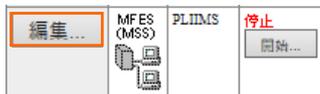
- 3) 画面の Page 2/4、3/4、ではそのまま [次へ] ボタンを、Page 4/4 では [OK] ボタンをクリックすると、[PLIIMS] という名前の 32 ビットアプリケーション稼働用 Enterprise Server インスタンスが追加されます。



**重要**

アプリケーション稼働ビット数 = Enterprise Server インスタンス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 [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"
CODEWATCH_NOTIF=Y
```

変更後；

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

 **情報**

CODEWATCH\_NOTIF 環境変数について：  
CodeWatch デバッガを使用する際に開発用インスタンスに指定します。本番インスタンスにはパフォーマンスの観点から排除することを推奨します。ここでは Eclipse 上のデバッガを使用するため削除します。

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 (✓) **JES... (✓)** IMS... (✓) PL/I (✓)

**一般** 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\plims\_base\catalog.dat

データセットの省略時ロケーション:  
\$PROJECT\plims\_base\

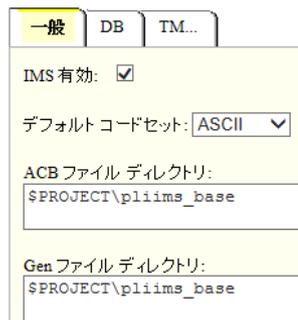
システム プロシージャライブラリ:  
SYS1.PROCLIB

- 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。



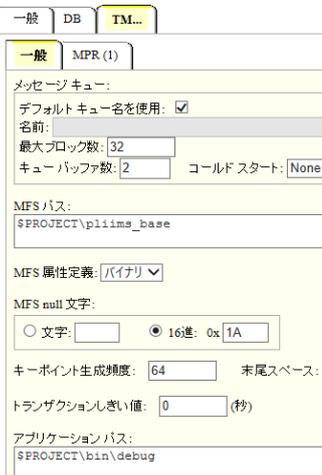
- 10) [サーバー] > [プロパティ] > [MSS] > [IMS] > [DB] > [一般] タブで表示される設定値を確認します。

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。



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。



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。

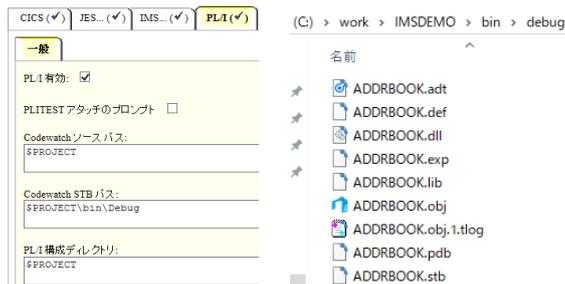
メッセージ処理領域の定義が一覧で表示されます。ここでは 1 つのメッセージ処理領域が既に定義されています。



名前	クラス
MPR01	1

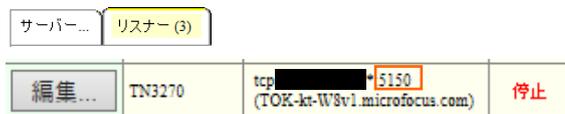
13) [サーバー] > [プロパティ] > [MSS] > [PL/I] > [一般] タブで表示される画面の各項目を確認します。

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 ソース パス	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 STB パス	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。



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。



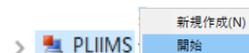
### 3.7 Enterprise Server インスタンスの開始と確認

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。

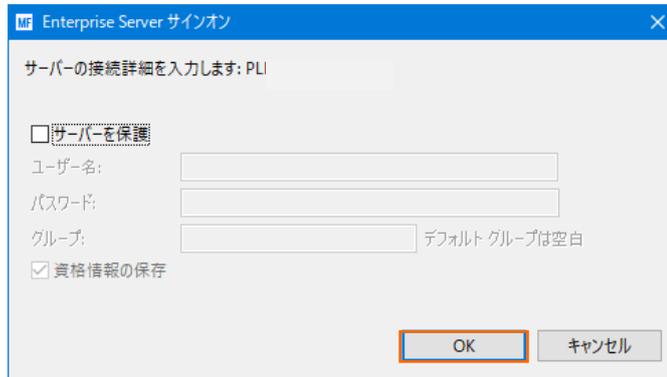
2) サーバーエクスプローラ内の [PLIIMS] インスタンスを右クリックし、[プロジェクトに関連付ける] > [IMSDEMO] を選択します。これにより Eclipse 内の [IMSDEMO] プロジェクトから実行するアプリケーションは [PLIIMS] インスタンスを対象として実行されます。



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



- 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

サーバーの接続詳細を入力します: PLI

[サーバーを保護]

ユーザー名:

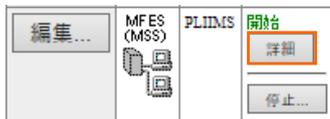
パスワード:

グループ:  デフォルトグループは空白

資格情報の保存

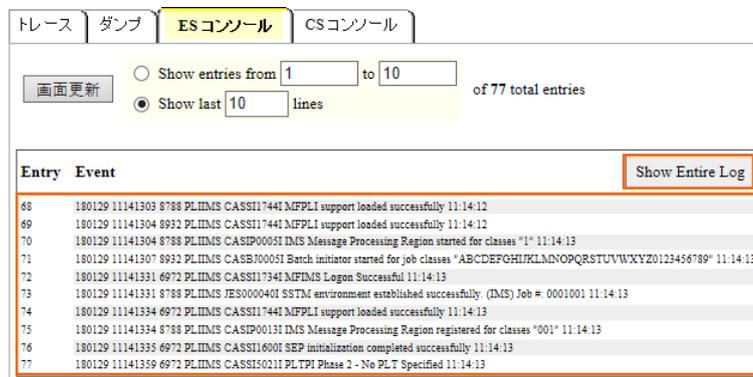
OK キャンセル

- 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。



トレース ダンプ ES コンソール CS コンソール

画面更新

Show entries from 1 to 10 of 77 total entries

Show last 10 lines

Entry	Event	Show Entire Log
68	180129 11141303 8788 PLIIMS CASSI17441 MFPLI support loaded successfully 11:14:12	
69	180129 11141304 8932 PLIIMS CASSI17441 MFPLI support loaded successfully 11:14:12	
70	180129 11141304 8788 PLIIMS CASIP0005I IMS Message Processing Region started for classes "1" 11:14:13	
71	180129 11141307 8932 PLIIMS CASB70005I Batch initiator started for job classes "ABCDEFGHIJKLMNQRSTUWXYZ0123456789" 11:14:13	
72	180129 11141331 6972 PLIIMS CASSI1734I MFIMS Logon Successful 11:14:13	
73	180129 11141331 8788 PLIIMS JES00040I SSTMI environment established successfully (IMS) Job #. 0001001 11:14:13	
74	180129 11141334 6972 PLIIMS CASSI1744I MFPLI support loaded successfully 11:14:13	
75	180129 11141334 8788 PLIIMS CASIP0013I IMS Message Processing Region registered for classes "001" 11:14:13	
76	180129 11141335 6972 PLIIMS CASSI1800I SEP initialization completed successfully 11:14:13	
77	180129 11141339 6972 PLIIMS CASSI502I PLTPI 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 データベースヘデータを挿入する 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
/*
//ZERO01 EXEC PGM=MFDBUJCL,PARM='ZEROLOAD,ADDRBOOK'
//SYSOUT DD SYSOUT=*,HOLD=Y
```

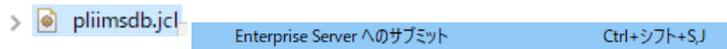
[BBOJ12 ステップ] SYSUT1 のデータが IMS データベースへ登録されます。

```
//BBOJ12 EXEC PGM=DFSRR00,
// PARM=(DLI,PLIIMSDB,PLIIMSDB)
//BBLOG DD DSN=DMF.BBLOG.ADDRBOOK,DISP=(NEW,CATLG,DELETE)
//IMSTRACE DD SYSOUT=*,HOLD=Y
//PLIDUMP DD SYSOUT=*,HOLD=Y
//SYSOUT DD SYSOUT=*,HOLD=Y
//SYSPRINT DD SYSOUT=*,DCB=(RECFM=LSEQ,LRECL=0),HOLD=Y
//SYSUT1 DD *
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 OH43233
ISRT Brewer Anthony 92 Rodent Lane Santa Clara CA45233
ISRT Fendick Andrew Caravan Keep Lancaster PA78383
ISRT Rafiq Irfan 87 Badminton Newbury UK78383
ISRT Kemanetzis Charis 3 Headbanger Way Newbury UK78383
LIST
/*
```

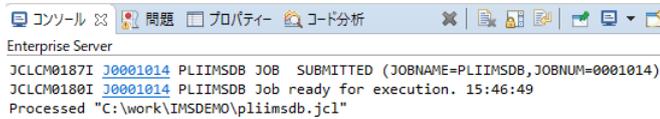
[START01 ステップ] 対象の IMS データベースを開始します。

```
//START01 EXEC PGM=MFDBUJCL,PARM='/STA DB ADDRBOOK'
//SYSOUT DD SYSOUT=*,HOLD=Y
```

- 2) 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。



- 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。



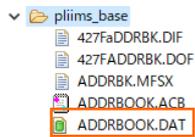
```
Enterprise Server
JCLCM0187I J0001014 PLIIMSDB JOB SUBMITTED (JOBNAME=PLIIMSDB,JOBNUM=0001014)
JCLCM0180I J0001014 PLIIMSDB Job ready for execution. 15:46:49
Processed "C:\work\IMSDemo\pliimsdb.jcl"
```

- 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。



```
x SYSPRINT BBOJ12 → starting pliimsdb 32 Bit ASCII ...
demo_dbd_name: ADDRBOOK
Successfully Inserted: Dave Fahringer
Successfully Inserted: Clive Poole
Successfully Inserted: Davinder Athwal
Successfully Inserted: Paul Belcher
Successfully Inserted: Jim Smith
Successfully Inserted: Anthony Brewer
Successfully Inserted: Andrew Fendick
Successfully Inserted: Irfan Rafiq
Successfully Inserted: Charis Kemanetzis
Listing all the Segments in Alpha Order(Last Name)
```

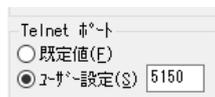
- 5) [PL/I エクスプローラー] 直下の [pliims\_base] を右クリック後 [更新] を選択してリフレッシュすると、生成された ADDRBOOK.DAT ファイルが確認できます。



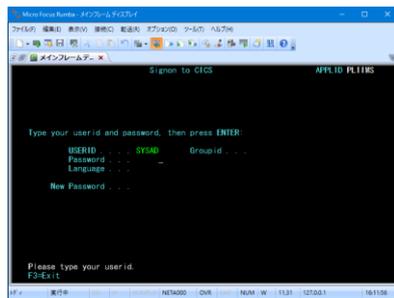
次に、オンライン処理からデータの確認と登録を行います。

### 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。

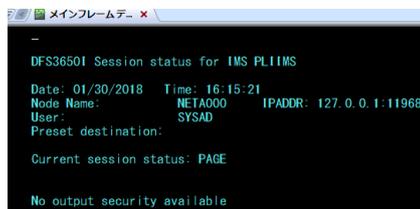
コマンド) /IMS



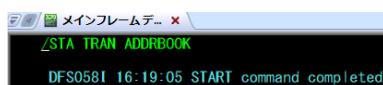
- 3) IMS モードの初期画面が表示されますので、これから実行するトランザクションを開始します。

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。



↓



- 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。

```

ADDRBOOK
Micro Focus (IP) Limited - Address Book
      A = Add, D = Delete
- 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 Kemanetzi   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
  
```

- 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

```

- Jim Smith      22 Royal Watcher, London, OH 43233
A 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。

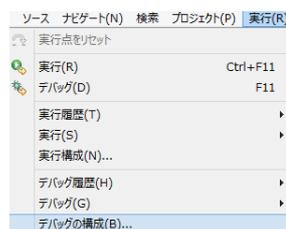
```

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

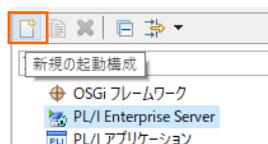
### 3.10 PL/I ソースのデバッグ

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。

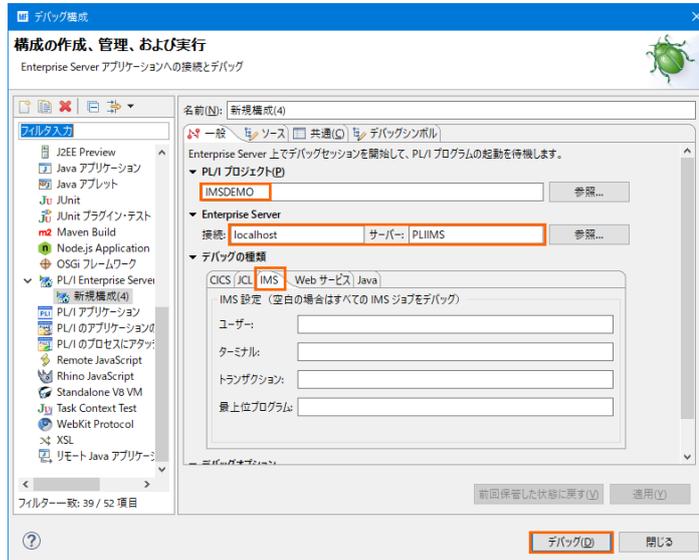
- 1) [実行] プルダウンメニューの [デバッグの構成] を選択します。



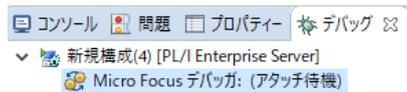
- 2) 左側のツリービューから [PL/I Enterprise Server] を選択して、左上の [新規の起動構成] アイコンをクリックします。



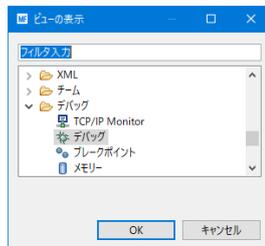
- 3) [PL/I プロジェクト] へ対象となる IMSDEMO を入力し、[Enterprise Server] へ実行させる PLIIMS インスタンスを指定します。[デバッグの種類] は [IMS] タブを選択した状態で、[デバッグ] ボタンをクリックします。



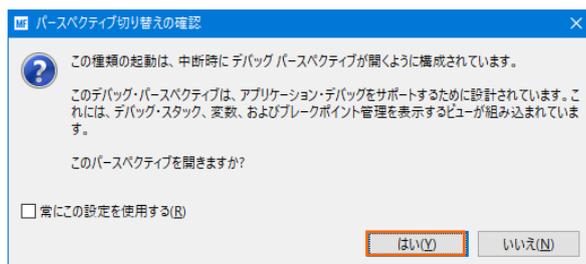
- 4) デバッグタブで [アタッチ待機] 状態になったことを確認します。



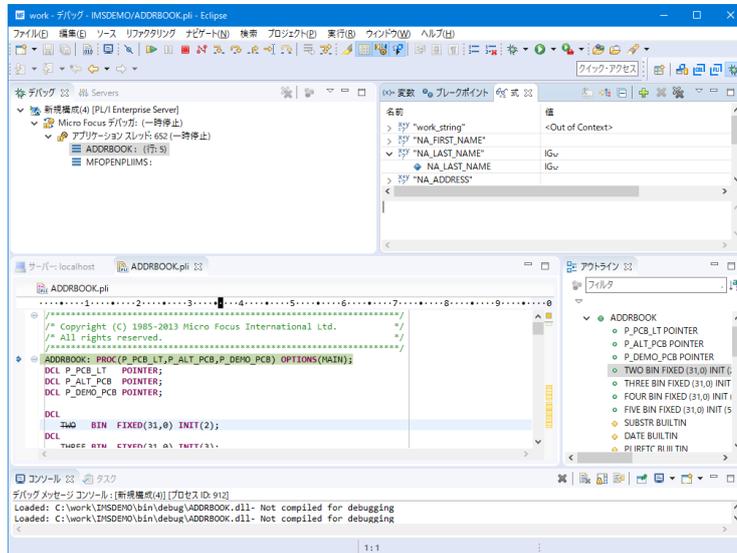
デバッグタブが表示されていない場合は [ウィンドウ] > [ビューの表示] > [その他] を選択して表示されるウィンドウで [デバッグ] > [デバッグ] を選択して [OK] ボタンをクリックするとデバッグタブが現れます。



- 5) 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。  
 6) 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。



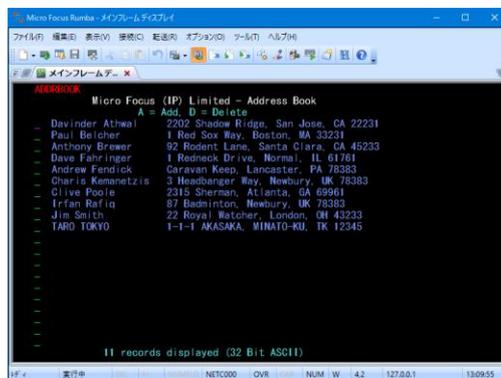
- 7) 少し待つとデバッグセッションが開始して、プログラムのステップ実行が可能になります。[F5] キーもしくは [実行] プルダウンメニューから [ステップイン] を選択してステップを進めることができ、[式] タブでは使用している変数を設定して値が確認できます。



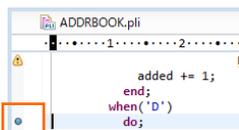
- 8) [再開] アイコンをクリックしてデータ一覧を表示させます。



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。



- 9) プログラムの 357 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。

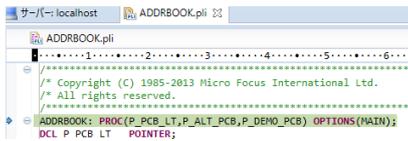


10) TN3270 エミュレータから、追加したデータ行へ “D” を入力して [Enter] キーを押下します。

```

ADDRBOOK
Micro Focus (IP) Limited - Address Book
A = Add, D = Delete
- 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 IARO TOKYO 1-1-1 AKASAKA, MINATO-KU, TK 12345
  
```

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

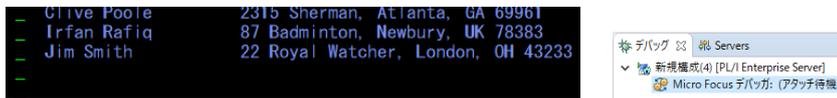


12) 削除ルーチンへ入り、項目値の確認ができます。



名前	値
work_string	TOKYO 1-1-1 AKASAKA, MINATO-KU, TK 12345
work_string	TOKYO 1-1-1 AKASAKA, MINATO-KU, TK 12345

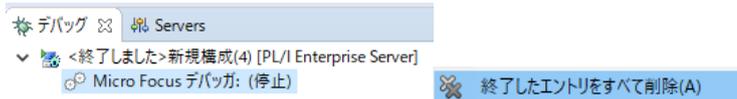
13) 先に進めるために [再開] アイコンをクリックすると指定データが削除された一覧が表示され、デバッガはアタッチ待機状態に戻ります。



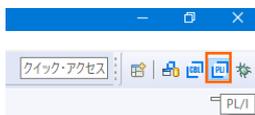
14) TN3270 エミュレータの接続を切断後、デバッグを終了させるため、画面上部の終了アイコンをクリックします。



15) 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。

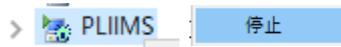


16) 右上のパースペクティブ切り換えで PL/I アイコンをクリックして、PL/I パースペクティブへ戻ります。

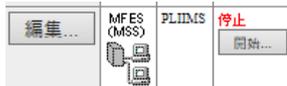


### 3.11 終了処理

- 1) サーバークラスター内で [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。



- 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。