
Micro Focus Enterprise Developer チュートリアル

メインフレーム COBOL 開発 : IMS

Eclipse 編

1. 目的

本チュートリアルでは、Eclipse 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。

2. 前提

- 本チュートリアルで使用したマシン OS : Windows 10 Enterprise
- 使用マシンに Micro Focus Enterprise Developer 7.0 for Eclipse 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

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 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. 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
7. Enterprise Server インスタンスの設定
8. Enterprise Server インスタンスの開始と確認
9. IMS リソースの確認
10. IMS トランザクションの実行
11. プログラムの動的デバッグ
12. Enterprise Server インスタンスの停止

3.1 チュートリアル準備

例題プログラムに関連する資源を用意します。

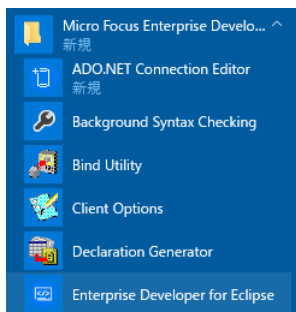
- 1) 使用する例題プログラムは、キットに添付されている IMStutorial.zip に圧縮されています。これを C:¥ 直下に解凍します。



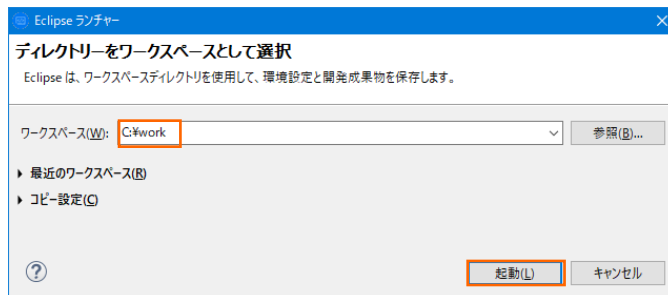
- 2) Eclipse のワークスペースで使用する work フォルダを C:¥ 直下に作成します。

3.2 Eclipse の起動

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



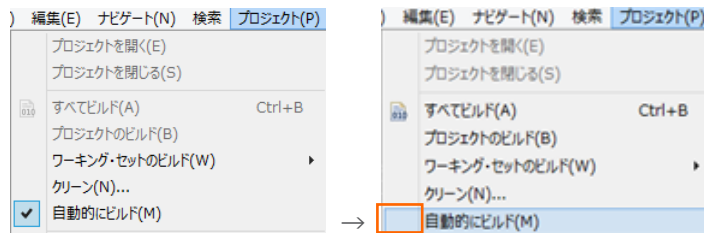
- 2) 前項で作成した C:¥work をワークスペースへ指定して、[起動] ボタンをクリックします。



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

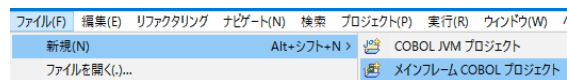


- 4) パースペクティブ表示後、[プロジェクト] プルダウンメニューの [自動的にビルド] を選択して、これをオフにします。

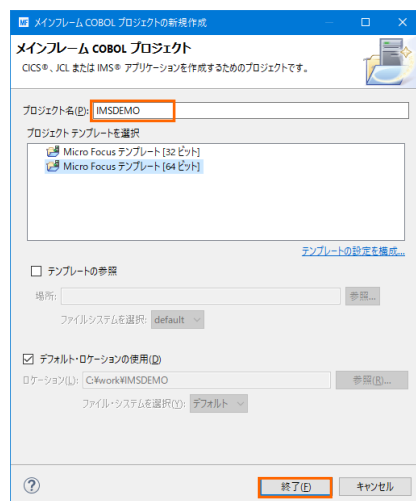


3.3 メインフレーム COBOL プロジェクトの作成

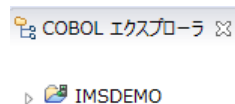
- 1) Eclipse のプロジェクトを作成します。[ファイル] プルダウンメニューから [新規] > [メインフレーム COBOL プロジェクト] を選択します。



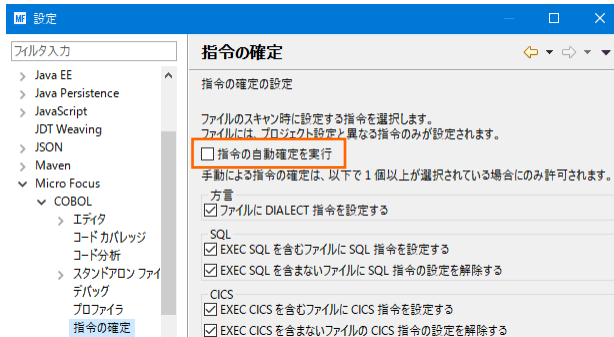
- 2) [プロジェクト名] は任意ですが、ここでは IMSDEMO を入力後、[64 ビット] テンプレートを選択して [終了] ボタンをクリックします。



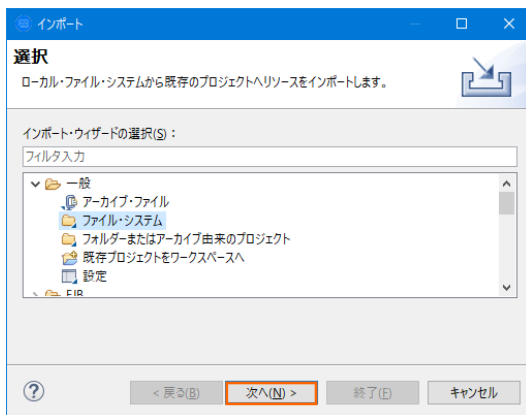
- 3) COBOL エクスプローラーへ作成したプロジェクトが表示されます。



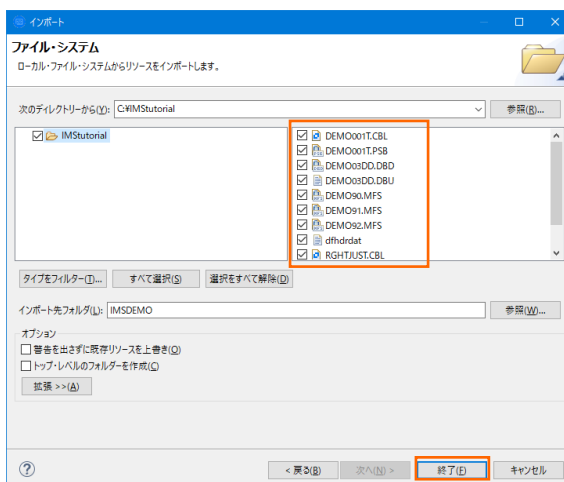
- 4) 既存ファイルのインポート時、自動的にコンパイル指令が指定される機能が用意されていますが、本チュートリアルではこれを解除します。[ウインドウ] プロダクションメニューの [設定] > [Micro Focus] > [COBOL] > [指令の確定] > [指令の自動確定を実行] チェックボックスをオフにして [適用して閉じる] ボタンをクリックします。



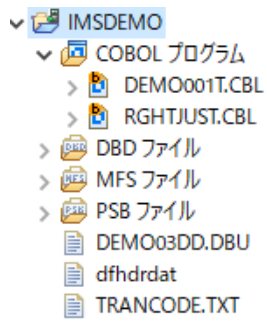
- 5) 用意した例題プログラム類をインポートします。IMSDEMO プロジェクトを右クリックして [インポート] > [インポート] を選択し、インポートウィンドウにて [一般] > [ファイル・システム] を選択後 [次へ] ボタンをクリックします。



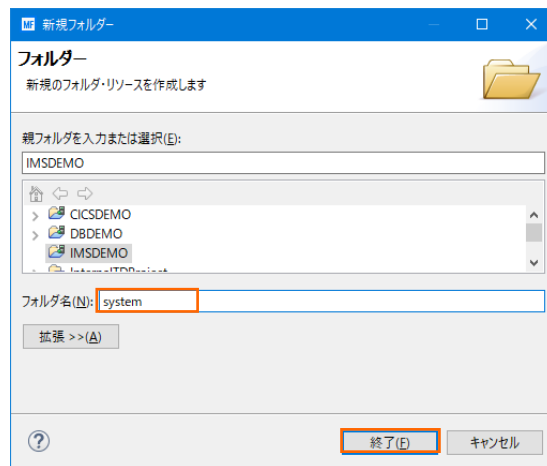
- 6) C:\¥IMStutorial を [次のディレクトリから] へ指定すると内容が表示されますので、全てのファイルをオンにして [終了] ボタンをクリックします。この実行により、プロジェクトフォルダへ例題プログラムが配置されます。



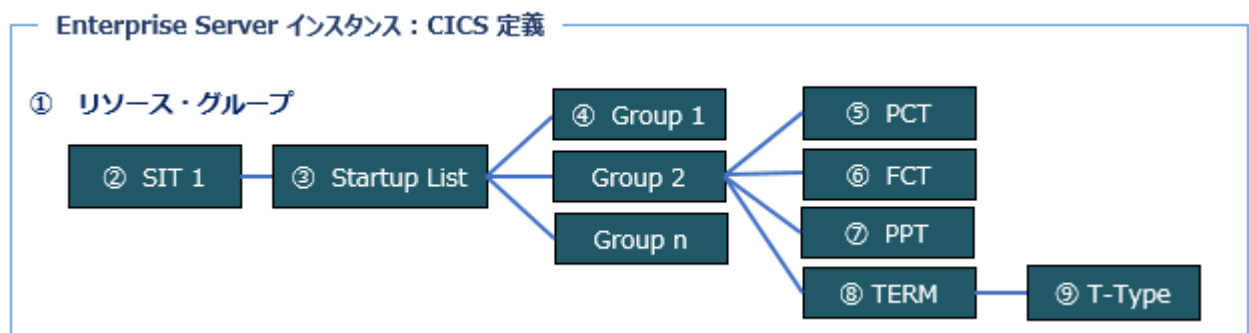
- 7) COBOL エクスプローラー内に表示されている IMSDEMO プロジェクトにインポートしたファイルが表示されていることを確認します。



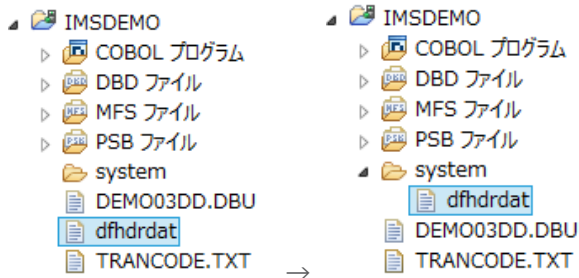
- 8) IMSDEMO プロジェクトへリソース定義ファイルを配置するフォルダを作成します。IMSDEMO プロジェクトを右クリックして [新規作成] > [フォルダ] を選択し、新規フォルダウィンドウにて [フォルダ名] に system を入力して [終了] ボタンをクリックします。



- 9) Enterprise Server インスタンスの Mainframe Subsystem Support (MSS) は“リソース”と呼ぶ CICS アプリケーションに使用するソフトウェアやハードウェアの項目を定義、制御、および監視するための機能を備えています。IMS アプリケーションに必要なリソースは、この機能に含まれるよう拡張された経緯から IMS においても CICS と同様にリソース定義ファイルを配置します。リソース構成の詳細については [CICS SIT 構築チュートリアル] をご参照ください。

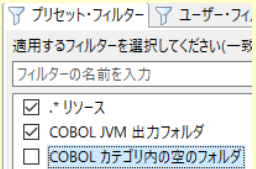


COBOL エクスプローラー内に表示されている IMSDEMO プロジェクトに存在するリソース定義ファイル dfhdrdat を、作成した system フォルダへドラッグ、ドロップしてファイルを移動します。



注意

フォルダ内が空のため作成したフォルダが表示されない場合があります。その際は COBOL エクスプローラー右上の [ビュー・メニュー] > [フィルタとカスタマイズ] を選択して [プリセット・フィルター] タブ内の [COBOL カテゴリ内の空のフォルダ] のチェックをオフにしてください。



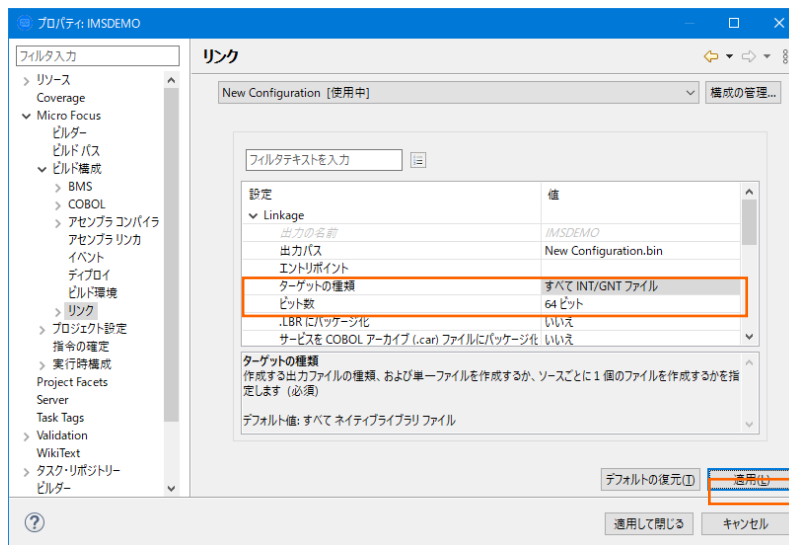
ビュー・メニュー → フィルタとカスタマイズ(F)... →

3.4 プロジェクトプロパティの設定

例題には 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。

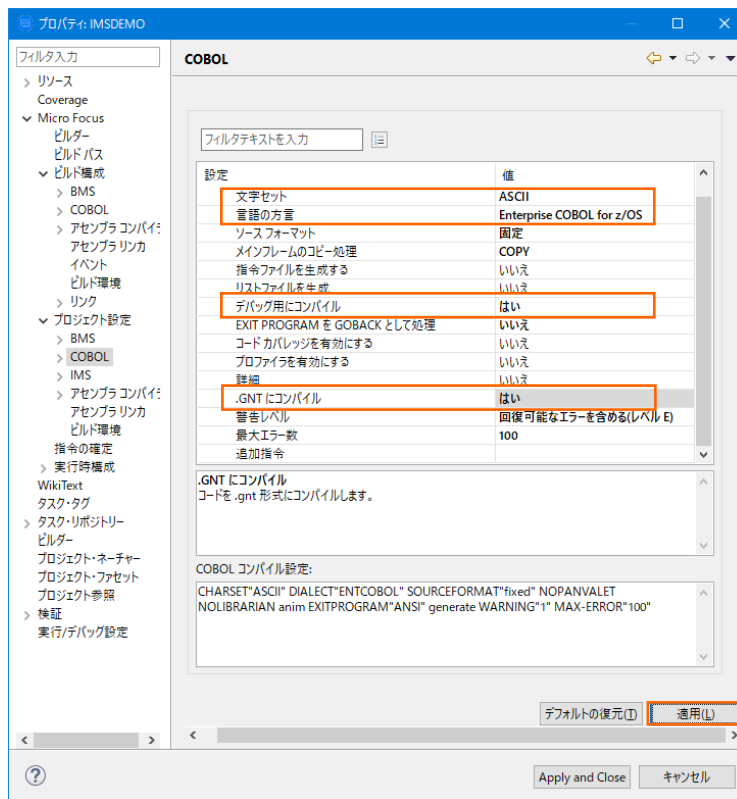
- 1) COBOL エクスプローラー内の IMSDEMO プロジェクトを右クリックして [プロパティ] を選択します。
- 2) 左側ツリービューの [Micro Focus] > [ビルド構成] > [リンク] を選択して、下記項目を指定します。指定後は [適用] ボタンをクリックしてください。

項目名	説明
ターゲットの種類	実行ファイル形式を指定。ここでは [全て INT/GNT ファイル] を選択します。
プラットフォーム ターゲット	稼働ビット数を指定。ここでは [64 ビット] を指定します。

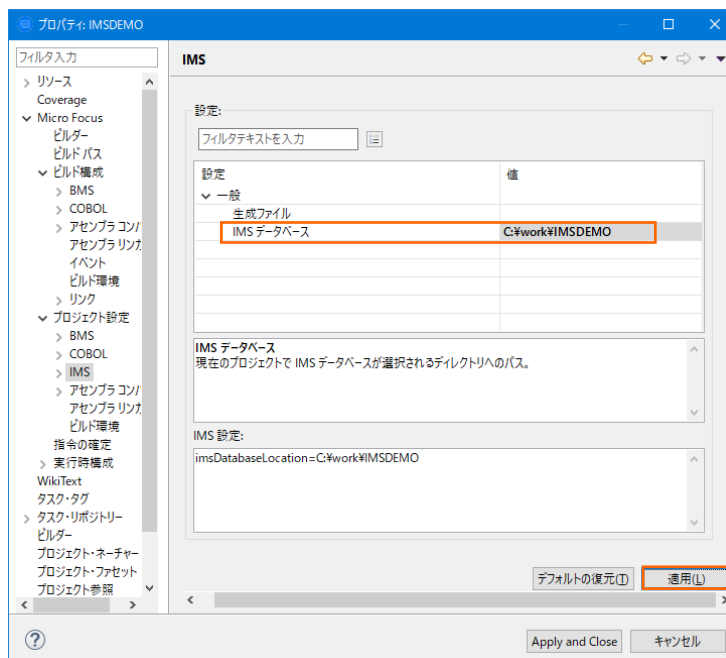


- 3) 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [COBOL] を選択して、下記項目を指定します。指定後は [適用] ボタンをクリックしてください。

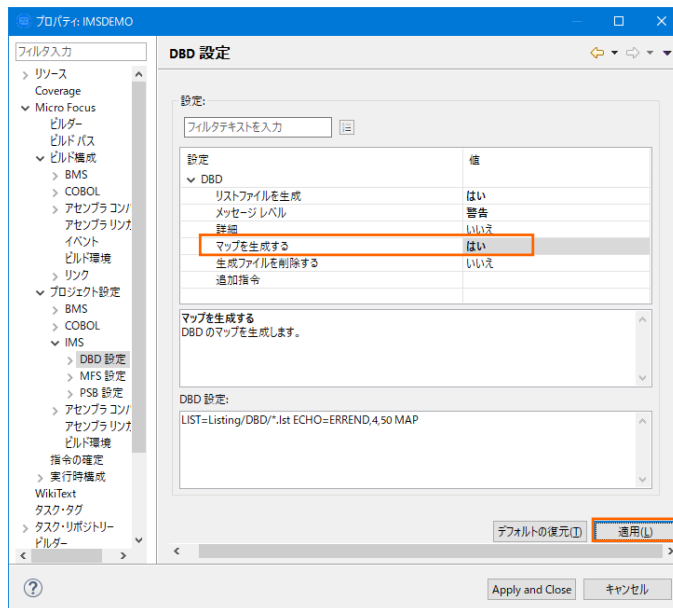
項目名	説明
文字集合	EBCDIC または ASCII を指定。ここでは [ASCII] を選択します。
言語方言	COBOL 言語方言を指定します。 例題プログラムは IBM Enterprise COBOL の方言を使用しているため、ここでは [Enterprise COBOL for z/OS] を指定します。
デバッグ用にコンパイル	デバッグ実行時に使用するファイルを生成するように指定します。
.GNT にコンパイル	実行ファイル形式を GNT に指定します。
追加指令	ここでは指定しません。



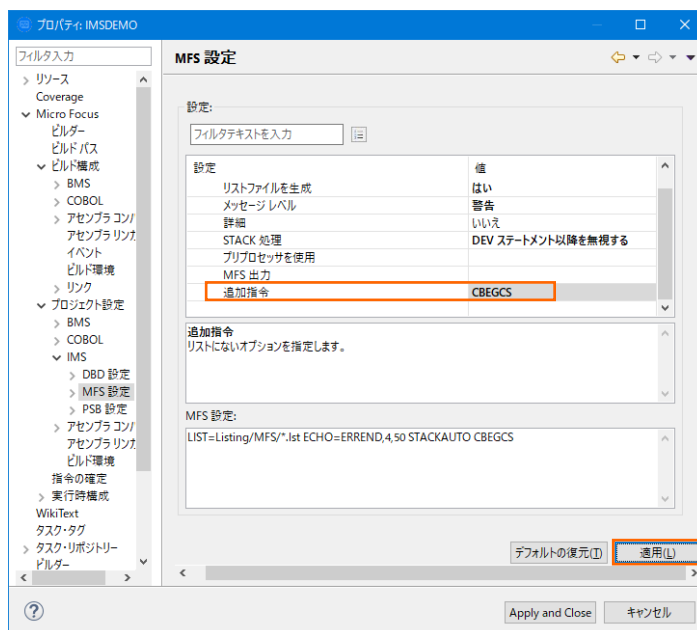
- 4) 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [IMS] を選択して、[IMS データベース] へ C:\work\IMSDEMO を指定して [適用] ボタンをクリックしてください。



- 5) 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [IMS] > [DBD 設定] を選択して、[マップを生成する] へ [はい] を選択して [適用] ボタンをクリックしてください。



- 6) 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [IMS] > [MFS 設定] を選択して、[追加指令] へ CBEGCS を入力して [適用] ボタンをクリックしてください。



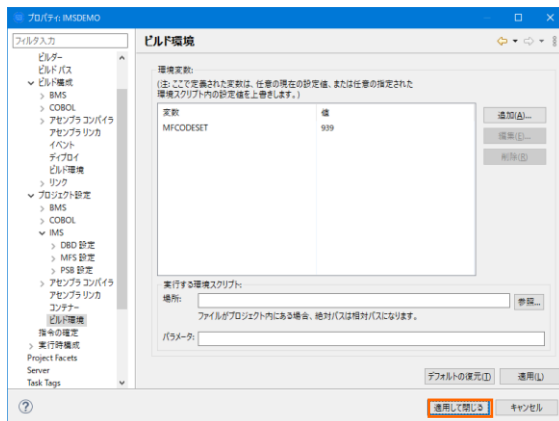
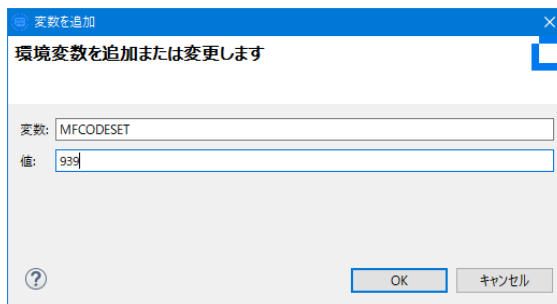
これは MFS 定義へ EATTR=EGCS と EATTR=MIX を指定時に追加する指令です。

- ① EGCS = 固定文字の場合は 'G'あいう' と指定し、前後のシフトイン、シフトアウト文字は排除します。
- ② MIX = 固定文字の場合は '<A>DD - 追加' と指定し、ダブルバイト文字前後のシフトイン、シフトアウト文字は排除します。

詳細に関しては製品ヘルプをご参照ください。

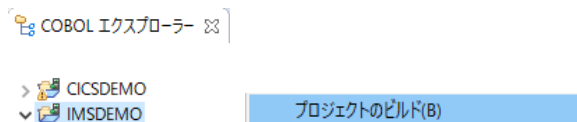
- 7) 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [ビルド環境] を選択後、[追加] ボタンをクリックしてコードセット値を設定します。追加後は [適用して閉じる] ボタンをクリックしてください。

項目名	説明
変数	ビルドに使用する環境変数を指定します。ここでは画面表示に使用するコードセット MFCODESET を指定します。
値	939 を指定します。 939 --- 英大小文字を使用する際に指定します。 9122 --- 英大文字と半角カタカナを使用する際に指定します。 半角カナを使用する際はこの値を指定します。

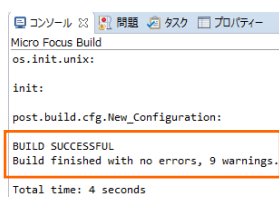


3.5 ビルドの実行

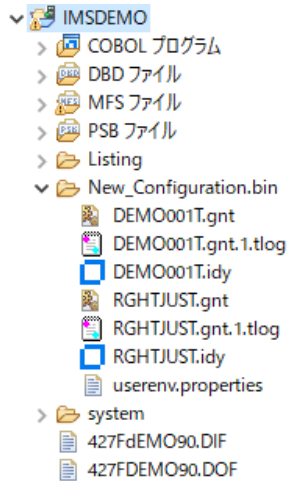
- 1) COBOL エクスプローラー内のプロジェクトを右クリックして [プロジェクトのビルド] を選択するとビルドが実行されます。



- 2) [コンソール] タブでビルドの成功を確認します。ワーニングメッセージは無視して問題ありません。



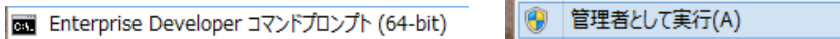
- 3) COBOL エクスプローラーのプロジェクト内に存在する New_Configuration.bin フォルダ配下に実行ファイル (.gnt ファイル) が生成され、プロジェクト直下には ACB, DIF, DOF, MAP, MFSX, MID, MOD や DBDGEN2.DAT, DBDGEN2F.DAT, PSBGEN3.DAT が PSB, DBD, MFS などのマクロから生成されています。



3.6 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) 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。



- 2) 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:\work\IMSDEMO
c:\work\IMSDEMO>
```

- 3) 例題に含まれている 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:\work\IMSDEMO>MFIMS IMSDBU LOAD DEMO03DD NOCLS
```

↓

```
13:24:51 DEMO03DD LOAD started
13:24:53 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
DEMO03DD.LST
IMSDBU.LOG
```

コマンドの詳細に関しては製品ヘルプをご参照ください。

- 4) 例題に含まれている TRANCODE.TXT ファイルは、メインフレームから抽出した Stage1 ファイルを Enterprise Server インスタンスの IMS 機能がサポートするファイル形式に “stage1ext” コマンドを使用して変換したものです。

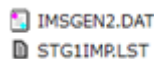
TRANCODE.TXT ファイルから IMS トランザクションのリストを設定するために、次のコマンドを実行します。

コマンド) MFIMS STAGE1IMP TRANCODE.TXT

```
MPP MFDEMO DEM0001TDEM0001T0000100000000000NYSSR001NPU
MPP TESTMENUTEST001TTEST001T0000101000000000NYSSR001NPU
MPP TESTMAINTEST002TTEST002T0000101000000000NYSSR001NPU

c:\work\%IMSDEMO>MFIMS STAGE1IMP TRANCODE.TXT
Importing transactions
Number of definitions inserted:
BMP : 00000.
QBMP : 00000.
MPP : 00003.
NRMP : 00000.
LTERM: 00000.
Classes used:
Class: 1 used by: 3 transactions
```

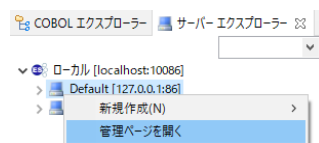
コマンドにより下記ファイルが作成されます。



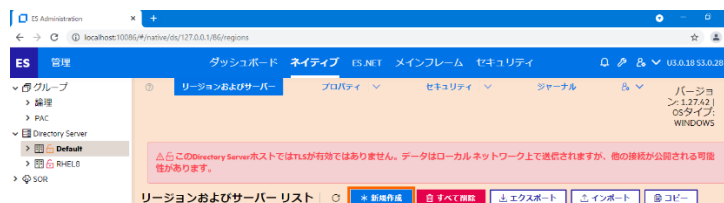
3.7 Enterprise Server インスタンスの設定

Enterprise Developer は CICS や IMS のエミュレーション機能を搭載している開発用 Enterprise Server インスタンスを内包しており、各開発者がこのインスタンスを占有してメインフレームアプリケーションのテスト実行やデバッグを行うことができます。本番環境にはコンパイラなどを含まない実行環境製品 Enterprise Server をインストールし、本番用インスタンス上でアプリケーションを稼働させます。

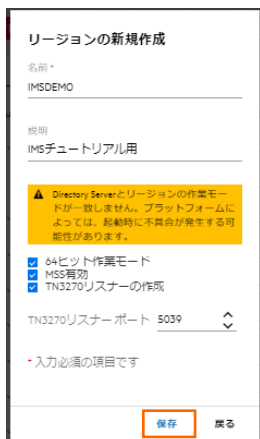
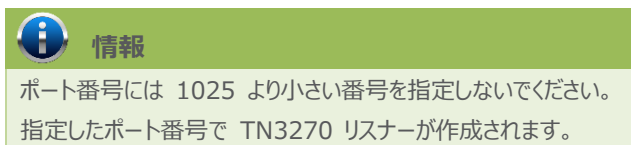
- 1) Enterprise Server インスタンスを作成します。Eclipse の [サーバー エクスプローラー] タブの [ローカル] > [Default] を右クリックして [管理ページを開く] を選択します。



- 2) ブラウザが立ち上がり、Enterprise Server インスタンスを運用、管理する Enterprise Server Common Web Administration (以降 ESCWA) が表示されます。画面の中央にある [新規作成] ボタンをクリックします。



- 3) [リージョンの新規作成] 項目の [名前]、[説明] は任意ですが、ここでは IMSDEMO と IMS チュートリアル用と入力します。Eclipse の実行可能ファイルは 64 ビットを指定してコンパイルしたため、稼働させる Enterprise Server インスタンスも同様に [64 ビット作業モード] ヘチェックを入れます。これにより警告が表示されますが無視して先に進んでください。[MSS 有効]、[TN3270 リスナーの作成] にチェックが入っていることを確認し、[TN3270 リスナーポート] へ 5039 を指定して [保存] ボタンをクリックします。

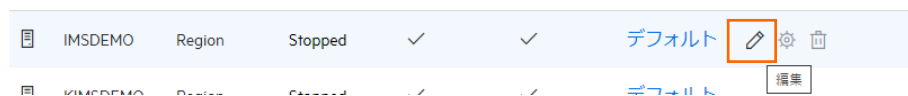



- 4) 64 ビットアプリケーション稼働用の IMSDEMO インスタンスが作成され、一覧に表示されます。

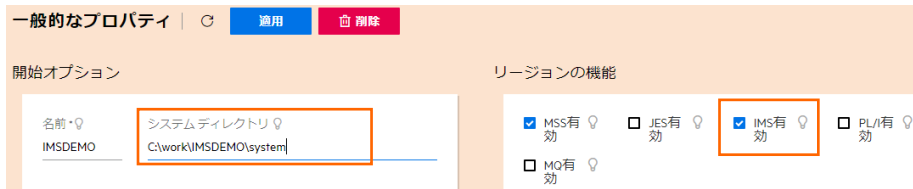


名前	説明	PAC	エンドポイント	タイプ	ステータス	64ビット	MSS有効	セキュリティ
IMSDEMO	Region				Stopped	✓	✓	デフォルト

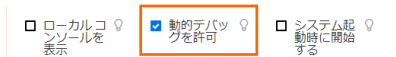
- 5) IMSDEMO インスタンスにカーソルを合わせ、[編集] アイコンをクリックします。



- 6) IMSDEMO インスタンスのログなどが出力される [システムディレクトリ] には Eclipse で作成した system フォルダを指定し、[リージョンの機能] の [IMS 有効] ヘチェックします。



Eclipse を使用したデバッグを実施するため、[動的デバッグを許可] にチェックします。



- 7) [追加設定] の [構成情報] にはプロジェクトのパス変数と、Eclipse で指定した MFCODESET を指定後、[適用] ボタンをクリックします。

設定値)

[ES-Environment]
 IMSPROJ=C:\work\IMSDemo
 MFCODESET=939

追加設定

構成情報
 [ES-Environment]
 IMSPROJ=C:\work\IMSDemo
 MFCODESET=939



重要

入力値は全て半角英数字で指定してください。

- 8) [CICS] プルダウンメニューから [構成] を選択します。

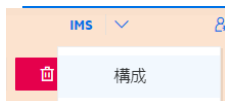


- 9) [CICS の構成] 項目に次の値を入力後、[適用] ボタンをクリックします。

項目名	説明
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。ここでは例題に含まれている IMSSIT を指定します。
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 Eclipse で作成した \$IMSPROJ\system を指定します。



- 10) [IMS] プルダウンメニューから [構成] を選択します。



- 11) [IMS 一般] 項目へ次の値を入力後、[適用] ボタンをクリックします。

項目名	説明
デフォルトコードセット	ASCII を指定します。
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。

IMS一般

デフォルトコードセット

ACB ファイルディレクトリ

GEN ファイルディレクトリ

12) [DB] タブをクリックし、次の項目を入力後、[適用] 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。

一般 DB TM

DB一般

データベースパス

13) [TM] タブをクリックし、次の項目を入力後、[適用] ボタンをクリックします。

項目名	説明
アプリケーションパス	ビルドした実行可能ファイルが存在するパスを指定します。 ここでは \$IMSPROJ¥New_Configuration.bin を指定します。
MFS パス	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。

IMSの構成 | 適用

一般 DB TM

アプリケーション

アプリケーションパス

末尾スペース 秒
トランザクションしきい値

MFS /パス

14) 同画面の下部の [メッセージ処理領域] にある [新規作成] ボタンをクリックし、メッセージ処理リージョン (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低 1 つは必要です。

メッセージ処理領域 | * 新規作成

名前	クラス	説明
名前	クラス	説明

15) [MPR プロパティ] に次の項目を入力後、[保存] ボタンをクリックします。

項目名	説明
名前	任意ですが、ここでは MPR01 を指定します。
トランザクション クラス	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。

MPR プロパティ

名前

クラス

説明

*入力必須の項目です

保存 戻る

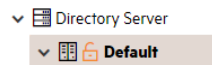
名前	クラス	説明
MPR01	1	MPR for class1

情報

MPR とは :

IMS TM アプリケーションを実行する際に必要となる、特殊なサービス実行プロセス (SEP) を指します。

16) 画面左側ペインの [Directory Server] > [Default] をクリックして一覧画面に戻ります。

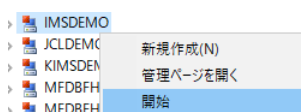


3.8 Enterprise Server インスタンスの開始と確認

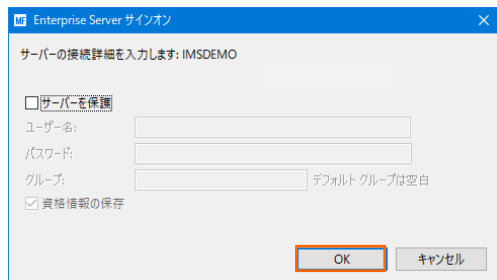
- 1) Eclipse に戻り、サーバー エクスプローラー内に IMSDEMO インスタンスが表示されていることを確認します。表示されていない場合は [ローカル] を右クリックし、[更新] を選択してリフレッシュしてください。
- 2) サーバー エクスプローラー内の IMSDEMO インスタンスを右クリックし、[プロジェクトと関連付ける] > [IMSDEMO] を選択します。これにより IMSDEMO プロジェクトから実行されるアプリケーションは IMSDEMO インスタンスで処理されることとなります。



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



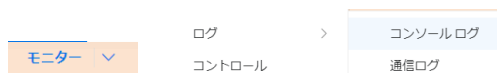
- 4) 下記ウィンドウが表示された場合は、ここではユーザーによる制限を行わないため [OK] ボタンをクリックします。



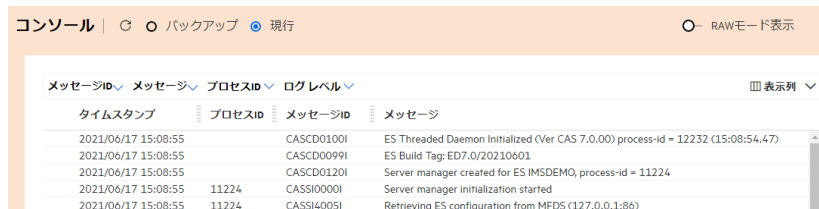
- 5) ESCWA で開始状態であることを確認後、[編集] アイコンをクリックします。

名前	タイプ	ステータス	64ビット	MSS有効	セキュリティ...
IMSDEMO	Region	Started	✓	✓	デフォルト

- 6) [モニター] プロダウンメニューから [ログ] > [コンソールログ] を選択し、IMSDEMO インスタンスのコンソールログを確認します。



- 7) コンソールログでインスタンスが正常に開始されたことを確認します。



タイムスタンプ	プロセスID	メッセージID	メッセージ
2021/06/17 15:08:55	CASCD01001	ES Threaded Daemon Initialized (Ver CAS 7.0.00) process-id = 12232 (15:08:54.47)	
2021/06/17 15:08:55	CASCD00991	ES Build Tag: ED7.0/20210601	
2021/06/17 15:08:55	CASCD01201	Server manager created for ES IMSDEMO, process-id = 11224	
2021/06/17 15:08:55	CASSI00001	Server manager initialization started	
2021/06/17 15:08:55	11224	CASSI40051	Retrieving ES configuration from MFDS (127.0.0.1:86)

【IMS 機能の正常開始ログ抜粋】

CASCD1042I	IMS TM Control Process created for Server IMSDEMO, process-id = 8588
CASTS1002I	ES TRC Service Process initialization complete
CASTS5115I	ES TSC Log facility file: C:\work\IMSDEMO\system\LOGTSTDQ.dat
CASCD1043I	IMS DB Control Process created for Server IMSDEMO, process-id = 11860
CASTM5001I	IMS TM Control Process initialization started
CASTS5113I	ES TSC Log facility cold started
CASCD1059I	IMS MPR Process created for server IMSDEMO, process-id = 10960



注意

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。

3.9 IMS リソースの確認

Enterprise Server インスタンスでは 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) ESCWA の [IMS] プルダウンメニューから [トランザクション] を選択します。



- 2) 登録されているトランザクション一覧が表示されますので、[MFDEMO] トランザクションにカーソルを合わせて [編集] アイコンをクリックします。

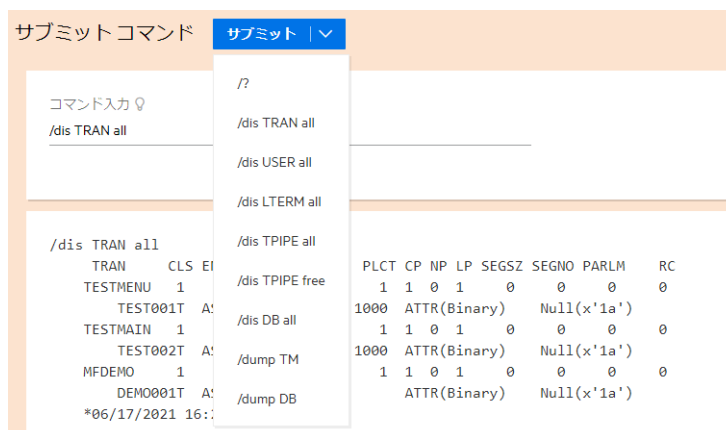
名前	PSB 名	説明	
MFDEMO	DEMO001T	MPP	
TESTMAIN	TEST002T	MPP	

編集

- 3) 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。



- 4) 再度 [IMS] プルダウンメニューから [コントロール] を選択すると IMS コントロール画面が表示されます。画面下部の [サブミット コマンド] ではコマンドを選択または入力して、トランザクションの状態などが確認できます。



3.10 IMS トランザクションの実行

現在 IMSDEMO インスタンスが稼働していますので、例題プログラムを実行することができます。

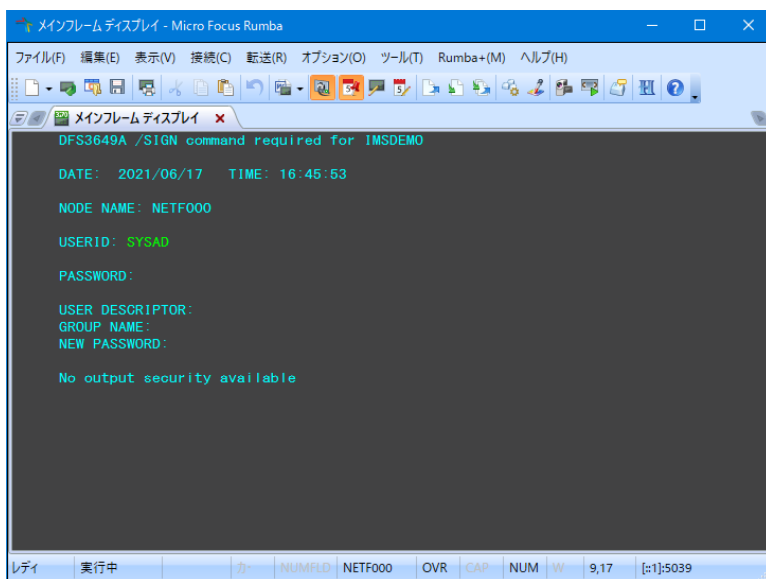
ご使用の TN3270 エミュレータを、前項で作成した TN3270 リスナーポート (localhost:5039) へ接続します。

補足) 後続の手順にクリアキーの押下があるため、TN3270 エミュレータで使用しているキーボード設定をご確認ください。

Rumba の例)



- 1) 下記は Micro Focus Rumba を使用した画面です。例題の SIT に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 を入力して実行キーを押します。



- 2) 正常にサインオン後、クリアキー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、実行キーを押します。(末尾に空白あり)

コマンド) /STA TRAN MFDEMO

```
/STA TRAN MFDEMO
```

- 4) 正常に開始されると下記メッセージが表示されます。

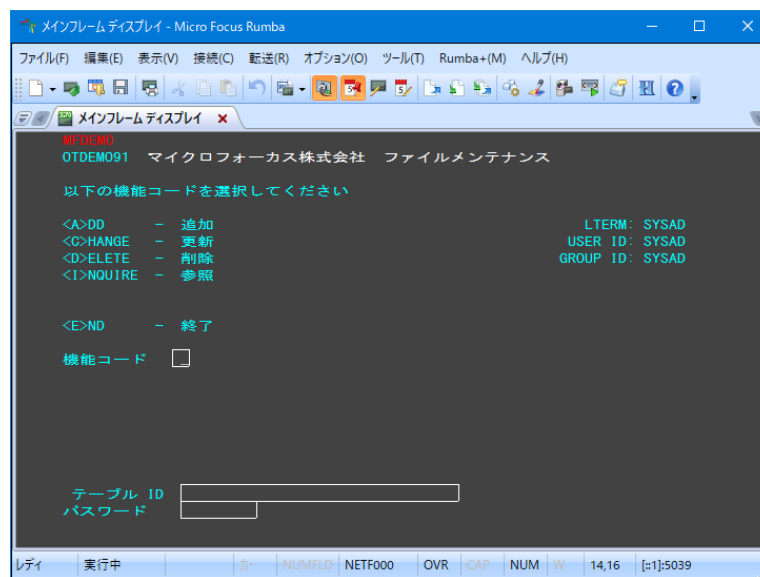
```
DFS058I 16:47:11 START command completed
```

- 5) 再度、画面をクリアキーでクリアします。
- 6) MFDEMO (末尾ブランク 1 文字)を入力して実行キーを押します。

```
MFDEMO
```

- 7) 下記のように例題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。

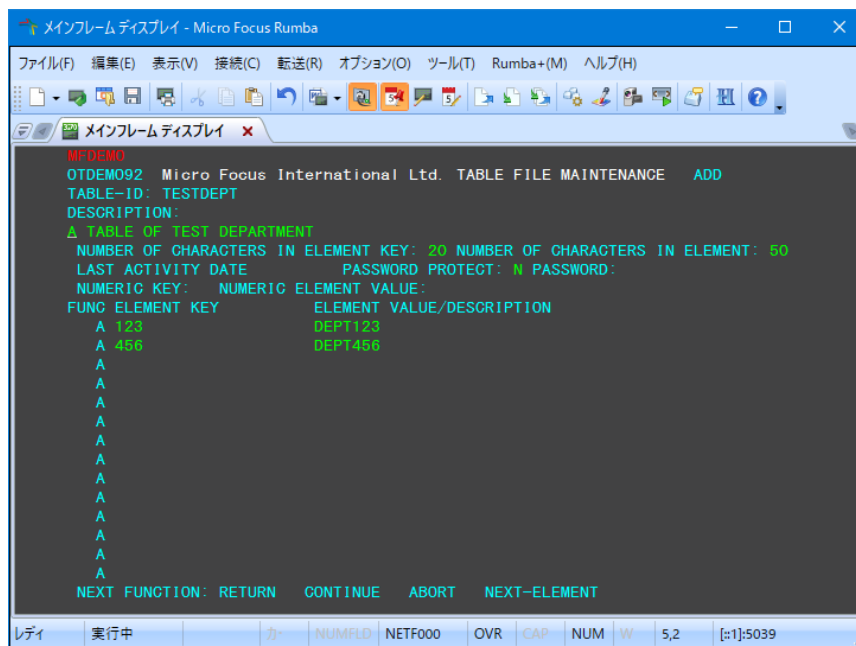
タブキーで項目間を移動して、[機能コード] へは A を [テーブル ID] へは TESTDEPT を入力後、実行キーを押します。



- 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、実行キーを押します。

なお、この例題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 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 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。



```

メインフレーム ディスプレイ - Micro Focus Rumba
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) Rumba+(M) ヘルプ(H)
メインフレーム ディスプレイ x
MFDemo
OTDEMO92 Micro Focus International Ltd. TABLE FILE MAINTENANCE ADD
TABLE-ID: TESTDEPT
DESCRIPTION:
A TABLE OF TEST DEPARTMENT
NUMBER OF CHARACTERS IN ELEMENT KEY: 20 NUMBER OF CHARACTERS IN ELEMENT: 50
LAST ACTIVITY DATE          PASSWORD PROTECT: N PASSWORD:
NUMERIC KEY: NUMERIC ELEMENT VALUE:
FUNC ELEMENT KEY          ELEMENT VALUE/DESCRIPTION
A 123                      DEPT123
A 456                      DEPT456
A
A
A
A
A
A
A
A
A
A
A
NEXT FUNCTION: RETURN CONTINUE ABORT NEXT-ELEMENT
レディ 実行中 かん NUMFLD NETF000 OVR CAP NUM W 5,2 [::]:5039
  
```

- 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、実行キーを押すとメニュー画面へ戻ります。

```

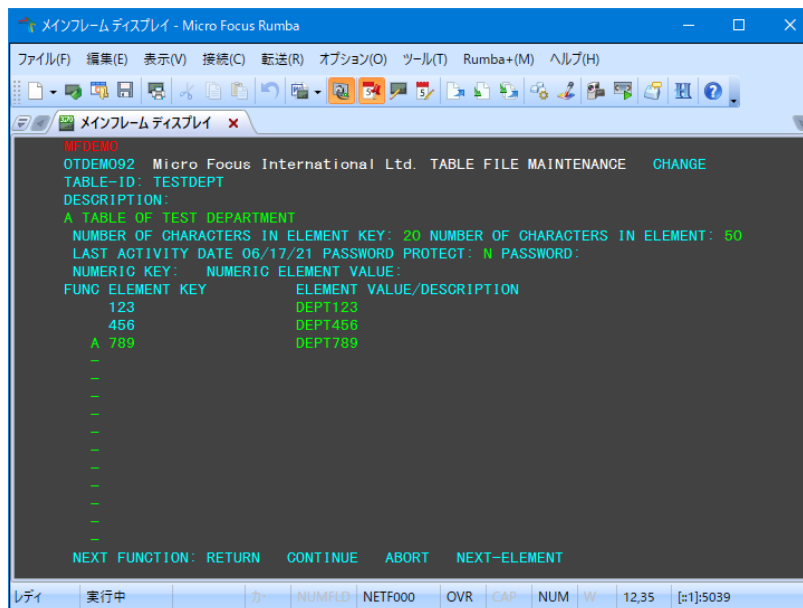
NEXT FUNCTION: RETURN X CONTINUE ABORT NEXT-ELEMENT
  
```

- 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、実行キーを押します。



- 11) テーブル要素の 3 行目に下記を追加して実行キーを押します。

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。



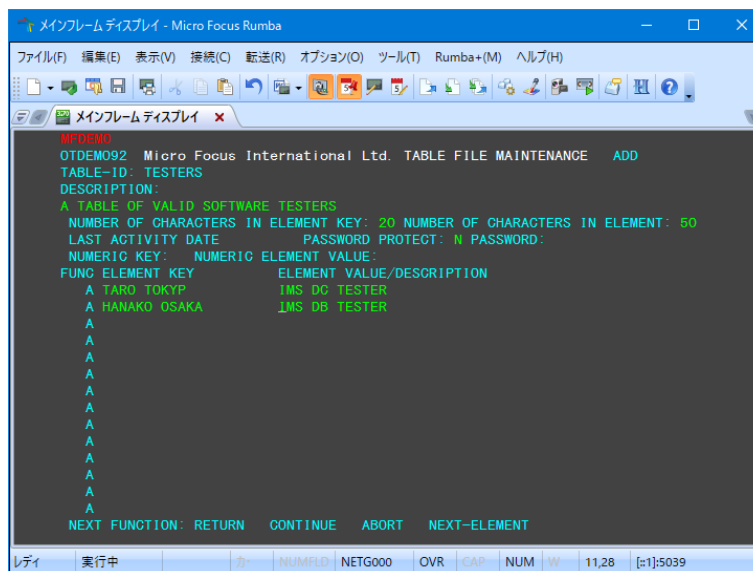
- 12) 画面下部の [RETURN] までタブキーで移動し、X を入力します。ほかの選択肢に X が入っていないことを確認後、実行キーを押すとメニュー画面へ戻ります。

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、実行キーを押します。



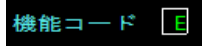
14) タブキーで項目間を移動して、TESTERS テーブルの要素を入力後、実行キーを押します。

項目名	説明
DESCRIPTION	説明を入力します。 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。

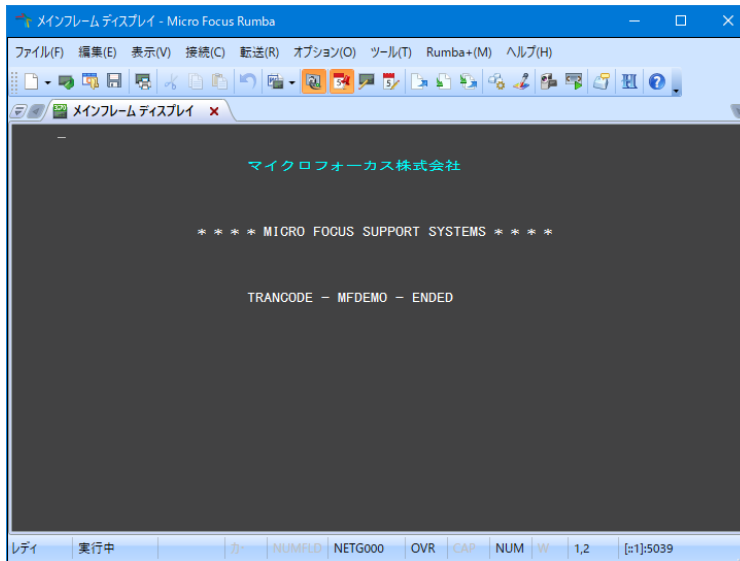


15) 前項と同じ要領でメニュー画面へ戻ります。

16) [機能コード] へ E を入力後、実行キーを押してトランザクションを終了します。



17) 例題プログラムの終了画面が表示されます。

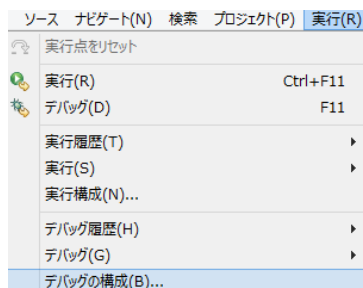


18) TN3270 エミュレータを切断します。

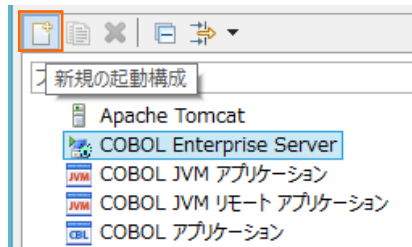
3.11 プログラムの動的デバッグ

Eclipse を使用して、例題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 インスタンスの [動的デバッグを許可] へのチェックと IMSDEMO インスタンスと Eclipse プロジェクトの関連付けが必要です。

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。

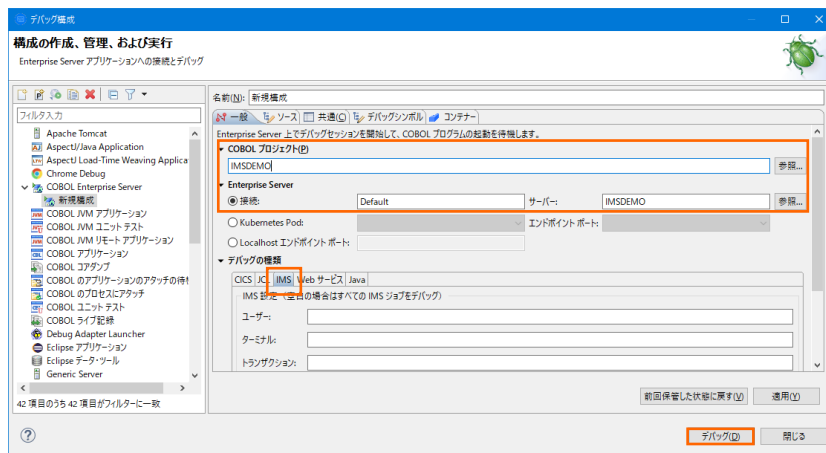


- 2) 左側のメニューから [COBOL Enterprise Server] を選択して、左上の [新規の起動構成] アイコンをクリックします。

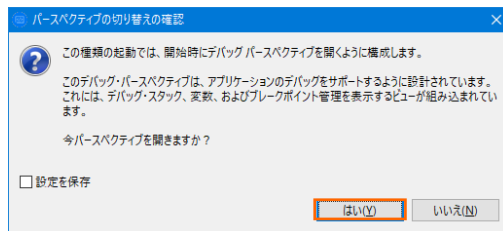


- 3) [COBOL プロジェクト] へ対象となる IMSDEMO を入力し、[Enterprise Server] へ実行させる IMSDEMO インスタンスを指定します。

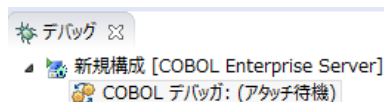
[デバッグの種類] は「IMS」タブを選択した状態で、[デバッグ] ボタンをクリックします。



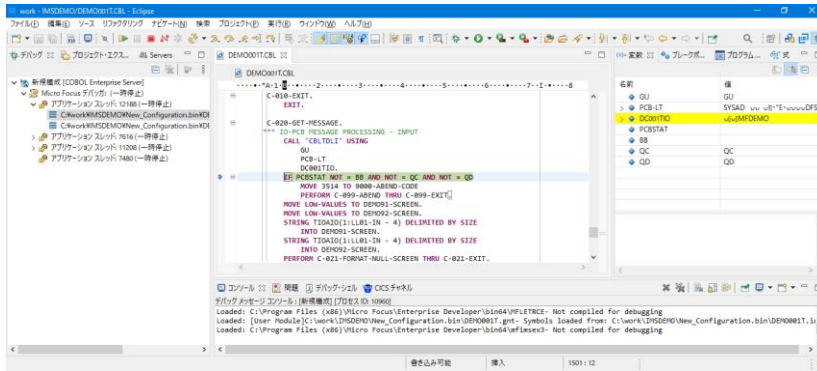
- 4) パースペクティブの切り替え確認ウィンドウでは [はい] ボタンをクリックし、デバッグ用のパースペクティブを開きます。



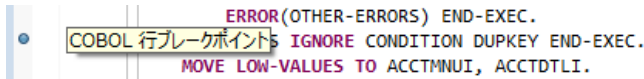
- 5) デバッグタブで [アタッチ待機] 状態になったことを確認します。



- 6) 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します。
 - 7) 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の値が確認できます。
- なお、この例題プログラムでは 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。



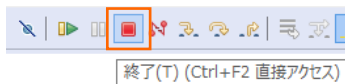
- 8) 希望のステップの左端をダブルクリックすることにより、ブレークポイントを設定することも可能です。



- 9) 先に進める場合は画面上部の再開アイコンをクリックします。

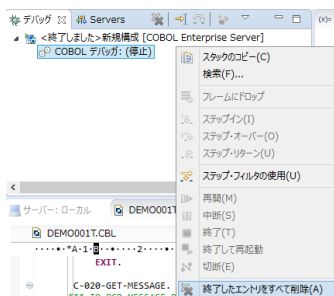


- 10) デバッグを終了させるため、画面上部の終了アイコンをクリックします。

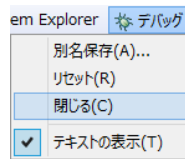


- 11) TN3270 エミュレータを切断します。

- 12) デバッグ履歴を消去するには、[デバッグ] パースペクティブの [デバッグ] タブ内で右クリックし、[終了したエントリをすべて除去] を選択します。

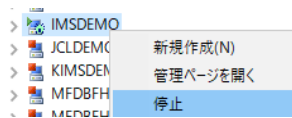


13) 右上の [デバッグ] パースペクティブを右クリックし、[閉じる] を選択して [COBOL] パースペクティブに戻ります。



3.12 Enterprise Server インスタンスの停止

1) IMSDEMO インスタンスを停止します。



2) IMSDEMO インスタンスの停止を確認後、Eclipse を終了します。

4. 免責事項

本チュートリアル の例題ソースコードは機能説明を目的としたサンプルであり、無謬性を保証するものではありません。例題ソースコードは弊社に断りなくご利用いただけますが、本チュートリアルに関わる全てを対象として、二次的著作物に引用する場合は著作権法 の精神に基づき適切な扱いを行ってください。

WHAT'S NEXT

- メインフレーム COBOL 開発 : JCL Eclipse 編
- メインフレーム COBOL 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。