
Micro Focus Enterprise Developer チュートリアル

メインフレーム COBOL 開発 : CICS

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、CICS 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 11 Pro
- 使用マシンに Micro Focus Enterprise Developer 8.0 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

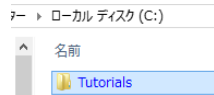
3. チュートリアル手順の概要

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. BMS 画面定義の確認
7. CICS リソース定義の概念
8. Enterprise Server インスタンスの設定
9. Enterprise Server インスタンスの開始と確認
10. CICS リソース定義の確認
11. CICS の実行
12. CICS の動的デバッグ
13. Enterprise Server インスタンスの停止

3.1 チュートリアルの準備

例題プログラムに関連するリソースを用意します。

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



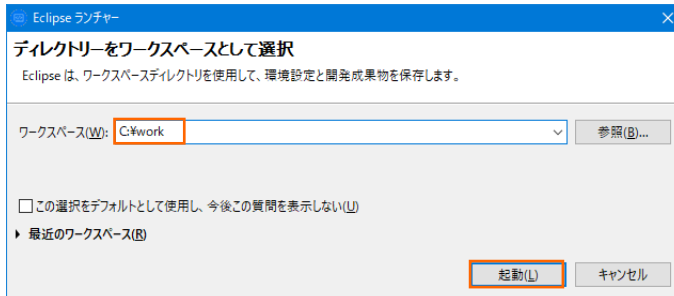
- 2) Eclipse のワークスペースで使用する work フォルダを C:¥ 直下に作成します。

3.2 Eclipse の起動

- 1) メニューから Enterprise Developer for Eclipse を起動します。



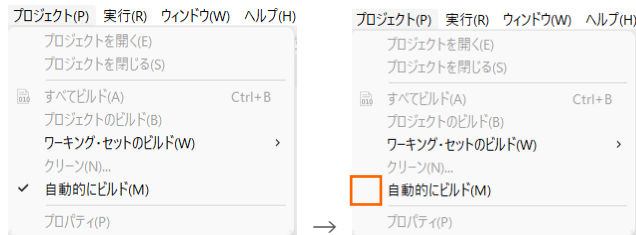
- 2) 前項で作成した C:¥work をワークスペースへ指定して、[起動] ボタンをクリックします。



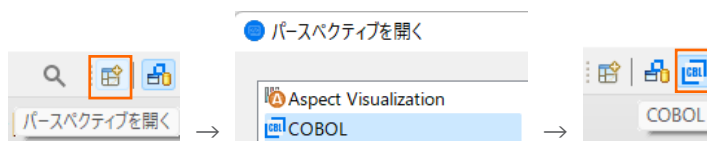
- 3) [ようこそ] タブが表示されたら [Open COBOL Perspective] をクリックして、COBOL パースペクティブを開きます。



- 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。

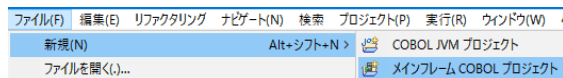


- 5) COBOL パースペクティブが開いていない場合は Eclipse 右上の [パースペクティブを開く] アイコンをクリックして表示後、[COBOL] を選択して [開く] ボタンをクリックします。

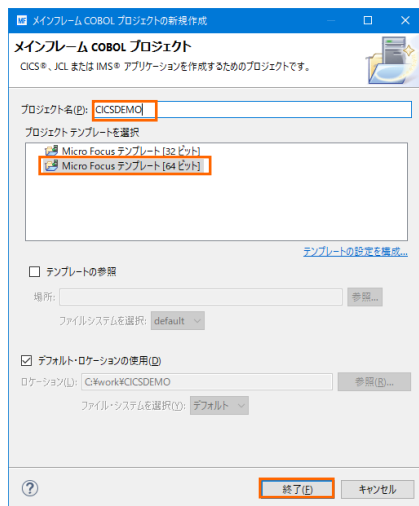


3.3 メインフレーム COBOL プロジェクトの作成

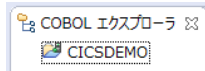
- 1) 用意した例題ソースをインポートするためのプロジェクトを作成します。[ファイル] プルダウンメニューから [新規] > [メインフレーム COBOL プロジェクト] を選択します。



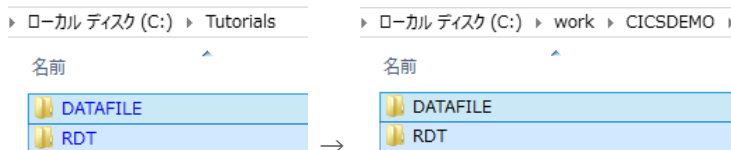
- 2) [プロジェクト名] は任意ですが、ここでは CICSDEMO を入力し、テンプレートは 64 ビットを選択後 [終了] ボタンをクリックします。



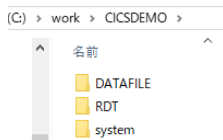
- 3) COBOL エクスプローラへ作成したプロジェクトが表示されます。



- 4) プロジェクトを作成したことにより C:\work\CICSDEMO フォルダが作成されていますので、C:\Tutorials フォルダ直下の DATAFILE と RDT フォルダを C:\work\CICSDEMO 配下へ Windows エクスプローラを使用してコピーします。DATAFILE には例題プログラムで使用するファイルが、RDT には例題プログラムで使用する定義済み CICS リソース定義テーブルが含まれています。



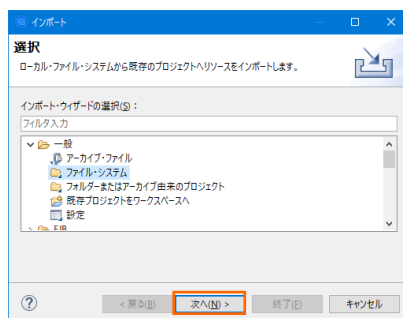
また、実行時に使用する system フォルダを新規作成してください。



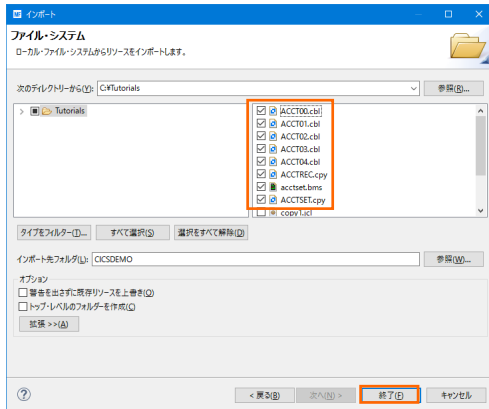
- 5) 既存ファイルのインポート時、自動的にコンパイル指令が指定される機能が用意されていますが、本チュートリアルではこれを解除します。[ウィンドウ] プルダウンメニューの [設定] > [Micro Focus] > [COBOL] > [指令の確定] > [指令の自動確定を実行] で [常になし] を選択し、[適用して閉じる] ボタンをクリックします。



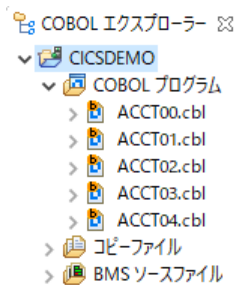
- 6) 用意した例題プログラム類をインポートします。CICSDEMO プロジェクトを右クリックして [インポート] > [インポート] を選択し、インポートウィンドウにて [一般] > [ファイル・システム] を選択後 [次へ] ボタンをクリックします。



- 7) C:\¥Tutorials を [次のディレクトリーから] へ指定すると内容が表示されますので、ファイル名の先頭に ACCT が付く上部 8 ファイルのチェックをオンにして [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。



- 8) COBOL エクスプローラー内に表示されている CICSDEMO 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。

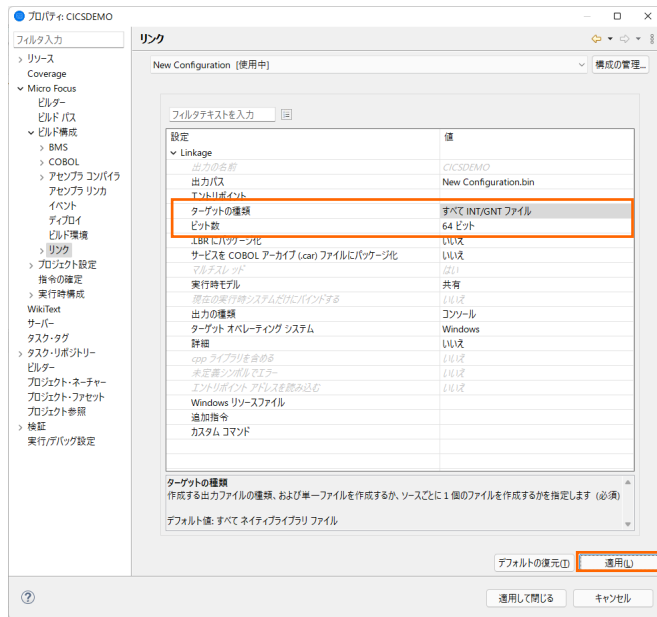


3.4 プロジェクトプロパティの設定

この例題は BMS 画面定義、EXEC CICS 文を含むプログラム、COPY メンバーが含まれています。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 [CICSDEMO] 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] を選択後、下記項目を指定して [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します。ここでは [すべて 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します。ここでは [64 ビット] を選択します。

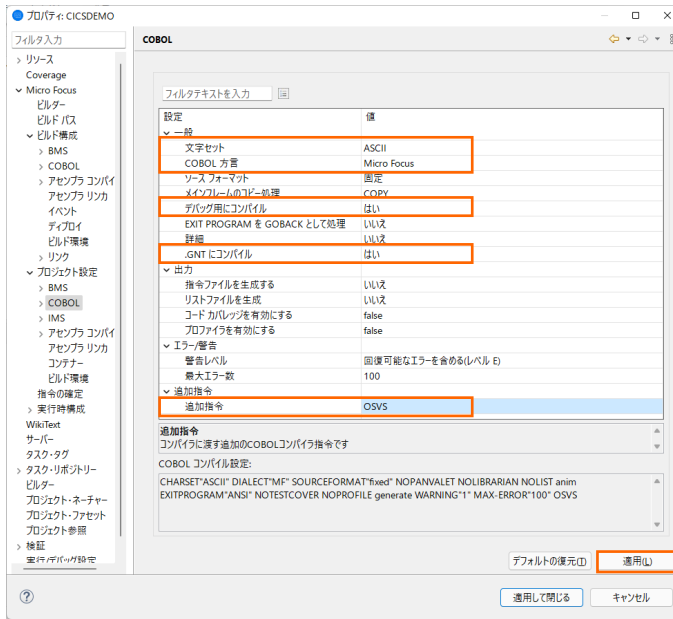


- 3) 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [COBOL] を選択して、下記項目を指定します。指定後は [適用] ボタンをクリックしてください。

項目名	説明
文字セット	EBCDIC または ASCII を指定します。ここでは [ASCII] を選択します。
言語方言	COBOL 言語方言を指定します。 例題プログラムは IBM OS/VS COBOL の方言を使用していますが、COPY 句に G 定数を使用しているためここでは [Micro Focus] 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 [はい] を選択します。
.GNT にコンパイル	実行ファイル形式を GNT に指定するため [はい] を選択します。
追加指令	ここでは OSVS を入力します。

情報

“.GNT” は Micro Focus 独自の動的にロード可能なファイルであり、実行可能ファイルにリンクする必要がありません。

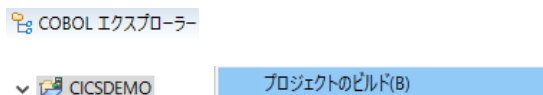


- 4) 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [COBOL] > [CICS プリプロセッサ] を選択して、[CICS プリプロセッサの使用] チェックボックスをオンにします。[適用して閉じる] ボタンをクリックしてウィンドウを閉じます。

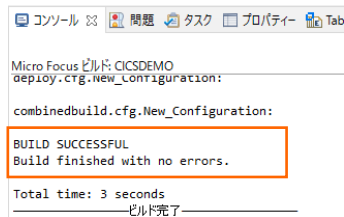


3.5 ビルドの実行

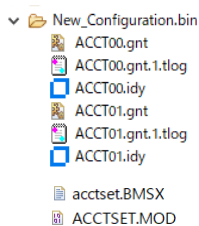
- 1) COBOL エクスプローラー内のプロジェクトを右クリックして [プロジェクトのビルド] を選択するとビルドが実行されます。



- 2) コンソールタブで成功を確認します。



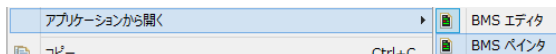
- 3) COBOL エクスプローラーのプロジェクト内に存在する New_Configuration.bin フォルダ配下の実行ファイル (.gnt ファイル) や BMS マップ (.mod ファイル) が作成されていることを確認してください。



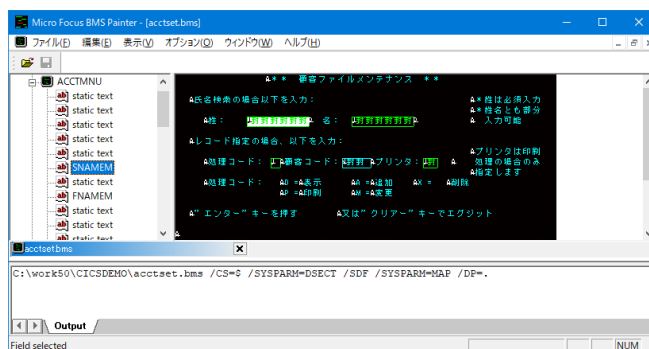
3.6 BMS 画面定義の確認

Enterprise Developer には CICS 開発者のために BMS 画面を対話型で編集するユーティリティやビューアーが装備されています。

- 1) COBOL エクスプローラーのプロジェクト内に存在する BMS ソースファイル フォルダの acctset.bms ファイルを右クリックして [アプリケーションから開く] > [BMS ペインタ] を選択します。

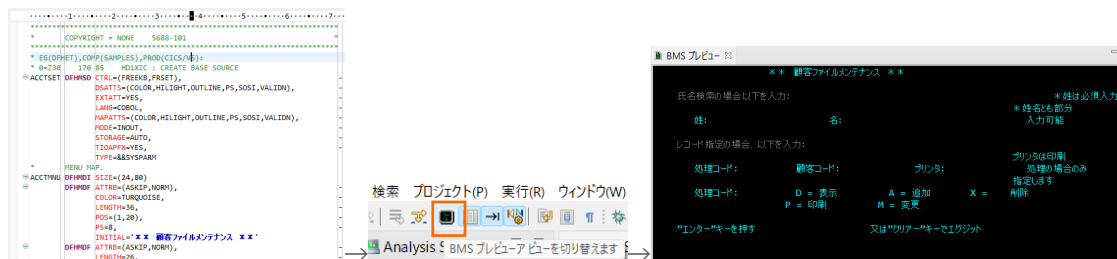


- 2) BMS ペインタウィンドウが表示され、画面定義内容をグラフィカルにメンテナンスできます。左側のツリービューでオブジェクトを選択すると右側のグラフィカルビュー内で対応するオブジェクトがハイライトされます。



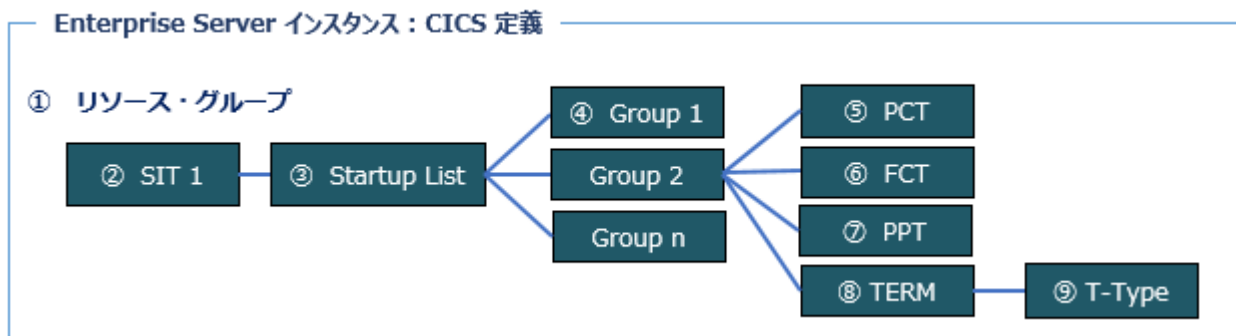
- 3) [ファイル] プルダウンメニューの [終了] を選択して BMS ペインタウィンドウを終了します。

- 4) BMS ソースをエディタで開き [BMS プレビュー] アイコンをクリックすると、画面のプレビューを表示させることもできます。表示させたいマップにカーソルを合わせると表示画面が切り替わります。



3.7 CICS リソース定義の概念

CICS ではアプリケーションで使用するソフトウェアやハードウェアの項目をリソースと呼び、Enterprise Server インスタンスの Mainframe Subsystem Support (MSS) は、このリソースを定義、制御、および監視するための機能を備えています。



項目名	説明
① リソース・グループ	SIT に関連する CICS リソースのセットを指します。
② SIT	CICS インスタンスの詳細設定を指定する、システム初期化テーブルを指します。
③ Startup List	CICS インスタンス起動時、自動的にロードされるグループ一覧を指します。
④ Group	PCT などの制御テーブルが所属するグループを指します。
⑤ PCT	CICS で使用するトランザクション制御テーブルを指します。
⑥ FCT	CICS で使用するファイル制御テーブルを指します。
⑦ PPT	CICS で使用するプログラム制御テーブルを指します。
⑧ TERM	端末定義を指します。
⑨ T-Type	端末タイプを指します。

3.8 Enterprise Server インスタンスの設定

Enterprise Developer は CICS 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ランタイムのみの実行環境製品 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) 実行する開発用 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。Default に登録されているインスタンスが表示エラーになる場合は、Windows の Micro Focus Directory Server サービスが開始されているか確認し、停止している場合は開始してください。



- 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、ここでは名前に CICSDEMO、説明に CICS 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] ヘ 9004 を指定して [保存] ボタンをクリックします。

名前
CICSDEMO

説明
CICSチュートリアル用

⚠ Directory Serverとリージョンの作業モードが一致しません。プラットフォームによっては、起動時に不具合が発生する可能性があります。

64ビット作業モード
 MSS有効
 TN3270リスナーの作成

TN3270リスナーポート 9004

* 入力必須の項目です

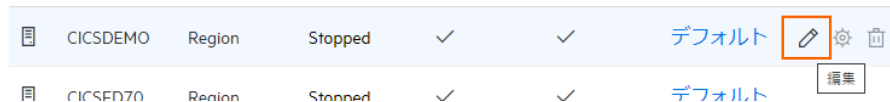
情報

ポート番号には 1025 より小さい番号を指定しないでください。
 指定したポート番号で TN3270 リスナーが作成されます。

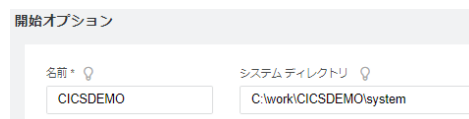
- 4) 64 ビットアプリケーション稼働用の CICSDEMO インスタンスが作成され、一覧に表示されます。



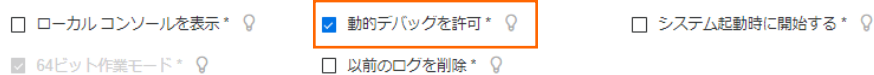
- 5) CICSDEMO インスタンスにカーソルを合わせ、[編集] アイコンをクリックします。



- 6) CICSDEMO 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] には前項で作成した system フォルダを指定します。



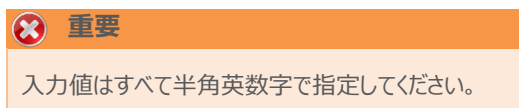
- 7) 表示画面の下にある [動的デバッグを許可] チェックボックスをオンにします。この指定により、Eclipse から動的デバッグが可能になります。



- 8) 例題では日本語半角カナ表示を有効にするため [追加設定] の [構成情報] 欄に下記内容を入力します。また、プロジェクトのパスを指定する環境変数も設定し、最後に [適用] ボタンをクリックします。

入力値)

[ES-Environment]
MFCODESET=9122
proj=C:\work\CICSDEMO



- 9) 画面上部の [CICS] ブルダウンメニューから [構成] を選択し、表示される画面の各項目を設定します。構成情報に指定した環境変数を使用して値を入力後、[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)	CICS インスタンス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 DBCS を指定します。
リソース定義ファイルパス	CICS リソース定義ファイルのパスを指定します。ここでは例題で用意されているリソース定義ファイルのパスを指定します。
トランザクションパス	実行される CICS プログラムの探索パスを指定します。ここでは .gnt ファイルが生成されているパスを指定します。
ファイルパス	データセットのデフォルトパスを指定します。ここでは例題で用意されている VSAM ファイルの置かれているパスを指定します。
マップパス	コンパイル済み BMS マップセットのパスを指定します。ここでは .MOD ファイルが生成されているパスを指定します。

CICSの構成 |

システム初期化テーブル(SIT) リソース定義ファイルパス

トランザクションパス

ファイルパス

マップパス

EZASOCKET サポート *

重要
 入力値はすべて半角英数字で指定してください。
 これらのフィールドでは改行を入れないように注意してください。

- 10) セキュリティ観点から、Web リスナーのデフォルトステータスは [Disabled] になっています。安全を確認したうえで、[一般] プルダウンメニューから [リスナー] を選択し、表示された Web リスナーのステータスを [Stopped] へ変更後、[適用] ボタンをクリックします。

TLS設定

ステータス	ステータスの設定	実際のアドレス
Disabled	Stopped	tcp:0.0.0.0:0

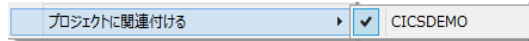
- 11) 画面左側ペインの [Default] をクリックして一覧画面に戻ります。

- ▼ Directory Server
 - ▼ Default
 - CICSDEMO

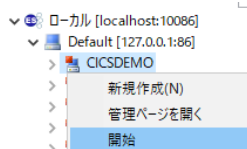
3.9 Enterprise Server インスタンスの開始と確認

- 1) Eclipse に戻り、サーバーエクスプローラー内に CICSDEMO 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 [Default] 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。

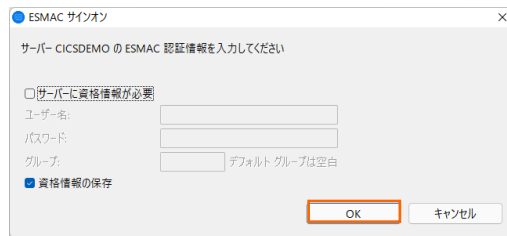
- 2) サーバーエクスプローラー内の CICSDEMO インスタンスを右クリックし、[プロジェクトに関連付ける] > [CICSDEMO] を選択します。これにより CICSDEMO プロジェクトから実行されるアプリケーションは CICSDEMO インスタンスで処理されることとなります。



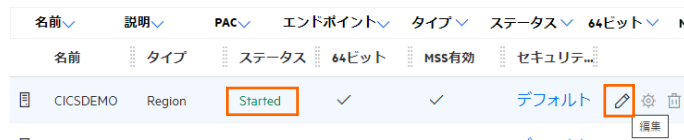
- 3) CICSDEMO インスタンスを右クリックして [開始] を選択します。



- 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 へ移動して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。



- 6) 画面上部の [モニター] プルダウンメニューから [ログ] > [コンソールログ] を選択し、正常に開始されたか確認します。

ログレベルが I はインフォメーション、S や E の場合はエラー表示されます。

コンソール | < > | バックアップ | 実行 | << >> | ページ: 1 / 1 | 51 行 / 51 | 表示

メッセージID	メッセージ	プロセスID	ログレベル		
タイムスタンプ	プロセスID	リージョン	メッセージID	ログレベル	メッセージ
2022/05/25 14:35:26			CASCD1095I	I	ES TRC Service Process created for Server CICSDEMO, process-id = 4668
2022/05/25 14:35:26			CASCD1075I	I	ES TSC Service Process created for Server CICSDEMO, process-id = 9400
2022/05/25 14:35:27			CASCD1038I	I	ES Communications Server created, ES CICSDEMO, process-id = 8944
2022/05/25 14:35:27	1764	CICSDEMO	CASKC6027I	I	Enterprise Server time keeper daemon initialised
2022/05/25 14:35:27	4700	CICSDEMO	CASKC1000I	I	ES concurrent request limit: 000000010
2022/05/25 14:35:27	4700	CICSDEMO	CASSI1000I	I	Server Manager initialization completed successfully
2022/05/25 14:35:27			CASCD0127I	I	SEP 00001 created for ES CICSDEMO, process-id = 1736
2022/05/25 14:35:27	4668	CICSDEMO	CASTS1007I	I	ES TRC Service Process initialization started
2022/05/25 14:35:27			CASCD0127I	I	SEP 00002 created for ES CICSDEMO, process-id = 6180
2022/05/25 14:35:27	9400	CICSDEMO	CASTS0007I	I	ES TSC Service Process initialization started
2022/05/25 14:35:27	9400	CICSDEMO	CASTS5115I	I	ES TSC Log facility file: C:\work\CICSDEMO\system\LOGTSTDQ.dat
2022/05/25 14:35:27	9400	CICSDEMO	CASTS5113I	I	ES TSC Log facility cold started
2022/05/25 14:35:27	9400	CICSDEMO	CASTS5116I	I	ES TSC Log facility file size: 4 blocks (316,236 bytes)
2022/05/25 14:35:27	9400	CICSDEMO	CASTS5117I	I	ES TSC Log facility is running with 4 buffers of 63,028 bytes each
2022/05/25 14:35:27	1736	CICSDEMO	CASSI1500I	I	SEP initialization started
2022/05/25 14:35:27	4668	CICSDEMO	CASTS1002I	I	ES TRC Service Process initialization complete
2022/05/25 14:35:27	8944	CICSDEMO	CASC5001I	I	Communications interface 01 initialization started
2022/05/25 14:35:27	6180	CICSDEMO	CASSI1500I	I	SEP initialization started
2022/05/25 14:35:27	9400	CICSDEMO	CASTS0002I	I	ES TSC Service Process initialization complete
2022/05/25 14:35:27	8944	CICSDEMO	CASC5003I	I	Communications interface 01 initialization complete



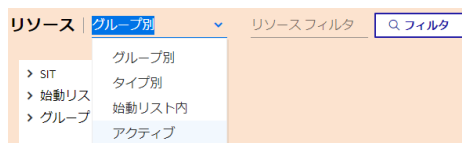
注意

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。

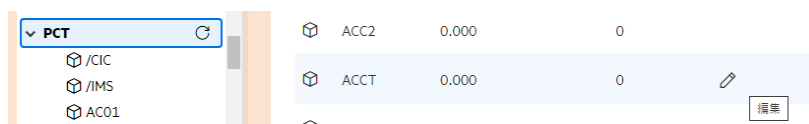
3.10 CICS リソース定義の確認

メインフレームの CICS と同様に Enterprise Server インスタンスでも各種リソース定義をオンラインで参照・更新・追加・削除することが可能です。本チュートリアルでは定義済みのリソース定義テーブルを使用していますので、その内容を参照してみます。

- 1) ESCWA の上部にある [CICS] プルダウンメニューから [リソース] を選択後、表示形式に [アクティブ] を指定します。



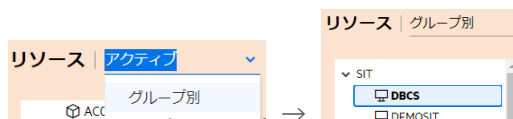
- 2) 左側ペインの [PCT] をクリックすると現在アクティブな PCT 一覧が表示されますので、[ACCT] の編集アイコンをクリックします。



- 3) ACCT トランザクションを呼び出すと ACCT00 プログラムが呼び出されることやステータスが有効であることが確認できます。



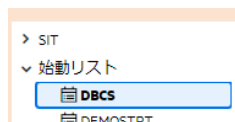
- 4) 次に、表示形式を [グループ別] に変更し、左側ペインから [SIT] を展開後、SIT で指定した [DBCS] をクリックします。



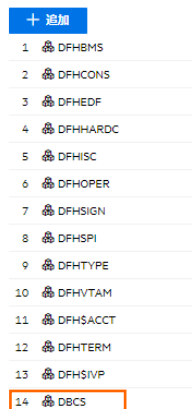
- 5) DBCS SIT には [始動リスト] として同名の定義が、[開始 Tran] には CESN (CICS サインオン・トランザクション) が指定してあることが確認できます。



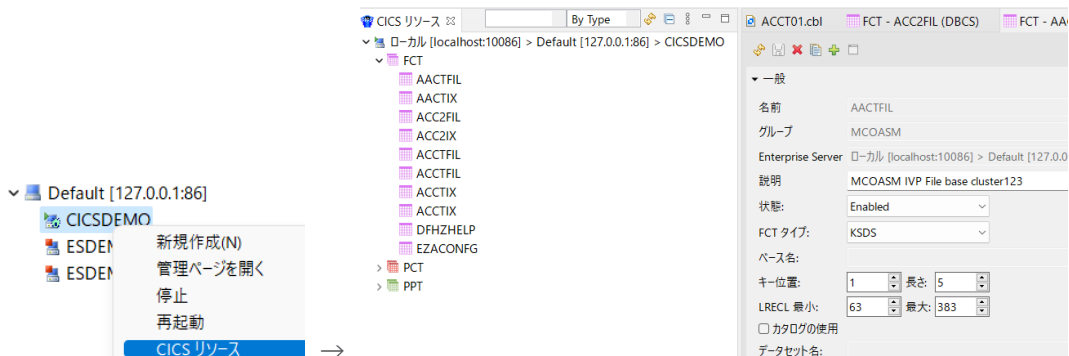
- 6) 上記で [始動リスト] に指定してある DBCS の内容を確認します。左側ペインの [始動リスト] を展開し、[DBCS] をクリックします。



- 7) DBCS 始動リストにはリソース定義類が属するグループが指定しており、同じ名称の定義が異なるグループに存在する場合は後述グループが優位になります。この始動リストでは 14 番目に指定されている DBCS グループに属するリソース定義類が最も優位となります。詳しくは [CICS SIT 構成チュートリアル] をご参照ください。



- 8) Enterprise Server インスタンスが開始状態であれば、Eclipse から一部の CICS リソースをメンテナンスすることができます。



3.11 CICS の実行

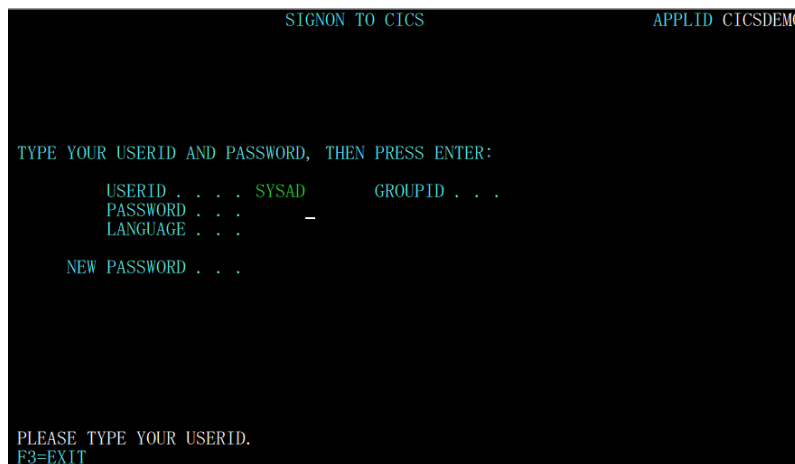
現在 CICSDEMO インスタンスが稼働していますので、例題プログラムを実行することができます。

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:9004) へ接続します。

補足) クリアキーを使用するため、TN3270 エミュレータのキーボード設定をご確認ください。



- 1) 下記は IBM パーソナル・コミュニケーションズを使用した画面です。例題の SIT では初期トランザクションに CESN が指定されているため、接続後 CICS サインイ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) 正常にサインオンできましたら、クリアキーで画面をクリアします。



- 3) PCT に登録されていたトランザクションの ACCT を入力して実行キーを押します。



- 4) ACCT トランザクションからプログラムが呼ばれて、例題の BMS ファイルに定義されていた下記初期画面が表示されます。

```

** 顧客ファイルメンテナンス **

氏名検索の場合以下を入力：
姓： [ ] 名： [ ]
*姓は必須入力
*姓名とも部分
 入力可能

レコード指定の場合、以下を入力：
処理コード： [ ] 顧客コード： [ ] プリンタ： [ ]
プリンタは印刷
処理の場合のみ
指定します
処理コード： D = 表示      A = 追加      X = 削除
               P = 印刷      M = 変更

”エンター” キーを押す      又は”クリアー” キーでエグジット
  
```

- 5) Tab キーで入力フィールドを移動して、[処理コード] へ D を、[顧客コード] へ 11111 を入力して実行キーを押します。

```

レコード指定の場合、以下を入力：
処理コード： [D] 顧客コード： [11111] プリンタ： [ ]
処理コード： D = 表示      A = 追加      X = 削除
               P = 印刷      M = 変更
  
```

- 6) 例題ファイルから指定顧客コードを持つデータが検索され、表示されます。

```

顧客ファイル レコード表示
顧客番号： 11111      姓：      整
                  名：      式部      MI: G 敬称： MRS
電 話： 0771778888 住 所：      4- 3、石山寺
                  大津市
                  滋賀県
その他の請求先：

発行カード枚数： 1      発行日： 06 07 07      理 由： N
カードコード： X      承認者： GNG      特別コード：

顧客状況： N      請求限度額： 1000.00

履歴：
残高      請求日      請求額      支払日      支払額
0.00      00/00/00      0.00      00/00/00      0.00
0.00      00/00/00      0.00      00/00/00      0.00
0.00      00/00/00      0.00      00/00/00      0.00

表示終了なら”クリアー” か”エンター” を押す
  
```

- 7) 実行キーを押して前画面へ戻り、TN3270 エミュレータを切断します。

3.12 CICS 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、CICSDEMO インスタンスの [動的デバッグを許可] へチェックと CICSDEMO 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) COBOL エクスプローラー内に存在する CICSDEMO プロジェクトの ACCT00.cbl をダブルクリックして内容を表示します。EXEC CICS 構文を使用して MAP の SEND と、次トランザクションを呼び出していることがわかります。

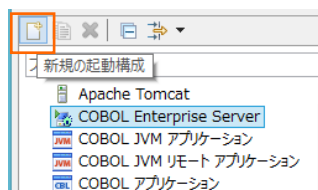
```

PROCEDURE DIVISION.
INITIAL-MAP.
    EXEC CICS SEND
        MAP('ACCTMNU')
        MAPSET('ACCTSET') FREEKB
        ERASE MAPONLY
    END-EXEC
    EXEC CICS RETURN TRANSID('AC01') END-EXEC
GOBACK.
    
```

- 2) これらのプログラムをステップ実行します。[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。

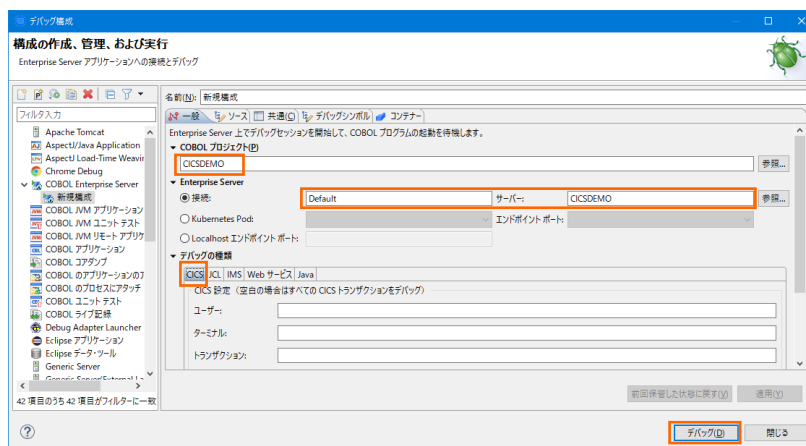


- 3) 左側のツリービューから [COBOL Enterprise Server] を選択して、左上の [新規の起動構成] アイコンをクリックします。

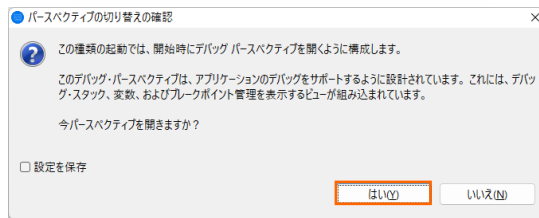


- 4) [COBOL プロジェクト] へ対象となる CICSDEMO プロジェクトを入力し、[Enterprise Server] へ実行させる CICSDEMO インスタンスを指定します。

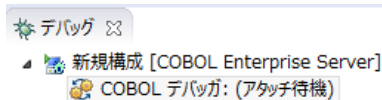
[デバッグの種類] は「CICS」タブを選択した状態で、[デバッグ] ボタンをクリックします。



- 5) 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。

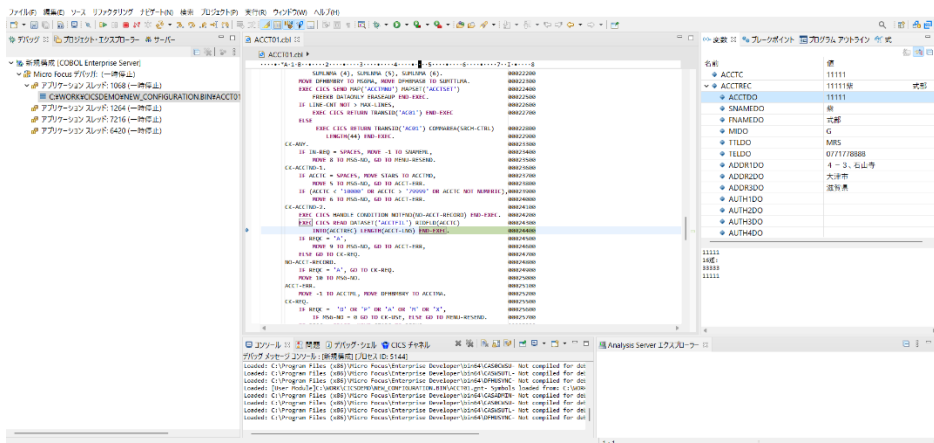


- 6) デバッグタブで [アタッチ待機] 状態になったことを確認します。

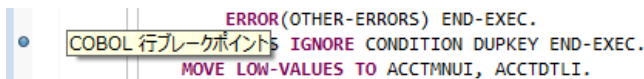


- 7) プログラムのステップ実行が可能になります。[F5] キーもしくは [実行] プルダウンメニューから [ステップイン] を選択してステップを進めることができ、変数タブでは使用している変数の値が確認できます。

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。

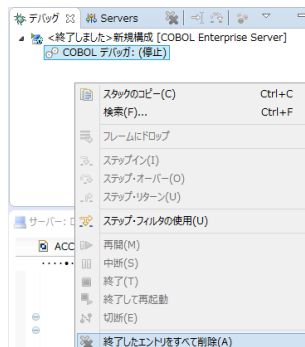


10) デバッグを終了させるため、画面上部の終了アイコンをクリックします。

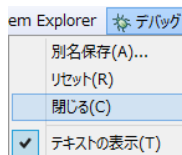


11) TN3270 エミュレータを切断します。

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削除] を選択します。

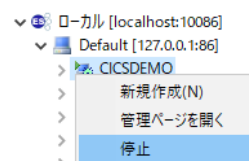


13) 右上の [デバッグ] パースペクティブを右クリックし、[閉じる] を選択して COBOL パースペクティブに戻ります。

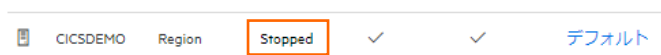


3.13 Enterprise Server インスタンスの停止

1) COBOL パースペクティブへ戻り、CICSDEMO インスタンスを停止します。



2) ESCWA から CICSDEMO 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すべてを対象として、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 : CICS SIT 構築
- 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。