
Micro Focus Enterprise Developer チュートリアル

メインフレーム COBOL 開発 : CICS

Visual Studio 2019 編

1. 目的

本チュートリアルでは、Visual Studio 2019 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 10 Enterprise
- 使用マシンに Microsoft Visual Studio 2019 がインストールされていること
- 使用マシンに Micro Focus Enterprise Developer 6.0 for Visual Studio 2019 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

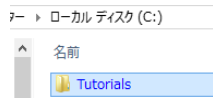
3. チュートリアル手順の概要

1. チュートリアルの準備
2. Visual Studio 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) Visual Studio のソリューションを保存する VS という名前のフォルダーを C:¥ 直下に作成します。

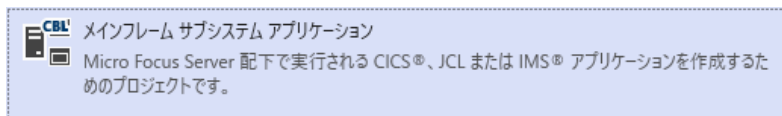
3.2 Visual Studio の起動

- 1) Visual Studio 2019 を起動します。



3.3 メインフレーム サブシステム アプリケーション プロジェクトの作成

- 1) 新しいソリューションと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 [新規作成] > [プロジェクト] を選択して [新しいプロジェクト] ウィンドウを表示し、[メインフレーム サブシステム アプリケーション] を選択後、[次へ] ボタンをクリックします。



- 2) 下記項目を入力後、[作成] ボタンをクリックします。

項目名	説明
プロジェクト名	任意ですが、ここでは CICSDEMO を入力します。
場所	前項で作成した C:¥VS を指定します。
ソリューションとプロジェクトを同じディレクトリに配置する	ここではチェックをオンにします。

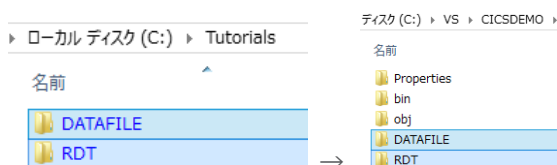


- 3) [ソリューション エクスプローラー] へ作成したプロジェクトが表示されます。



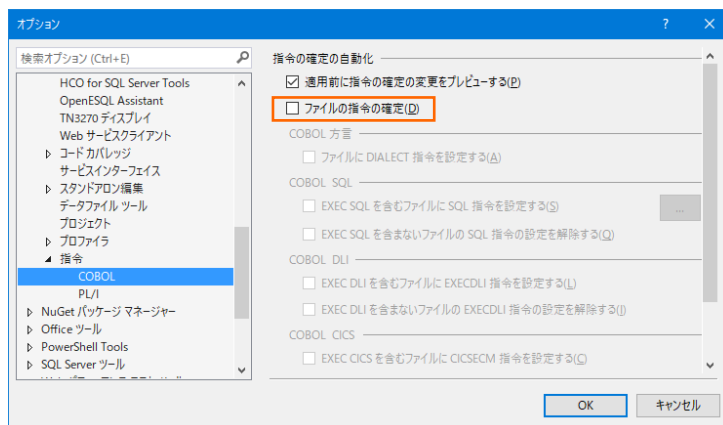
- 4) プロジェクトを作成したことにより C:\¥VS¥CICSDEMO フォルダが作成されていますので、C:\¥Tutorials フォルダ直下の DATAFILE と RDT フォルダを C:\¥VS¥CICSDEMO 配下へコピーします。

DATAFILE フォルダには例題プログラムで使用するファイルが、RDT フォルダ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

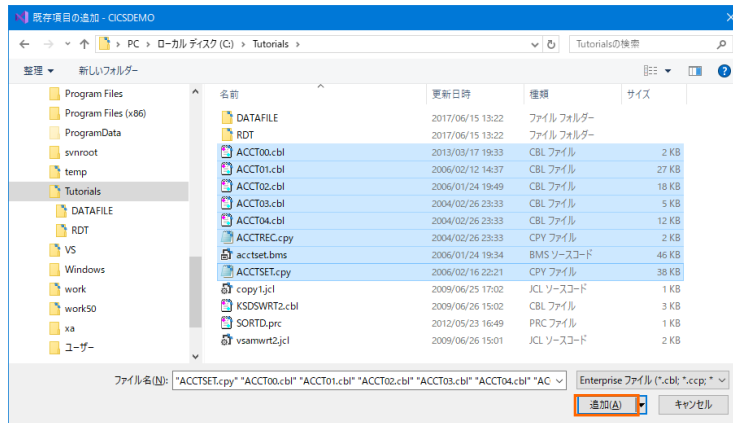


- 5) 既存ファイルのインポート時、自動的にコンパイル指令が指定される機能が用意されていますが、本チュートリアルではこれを解除します。[ツール] プロダクションメニューの [オプション] を選択してオプションウィンドウを表示します。

左側ツリービューの [Micro Focus Tools] > [指令] > [COBOL] > [ファイルの指令の確定] 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。



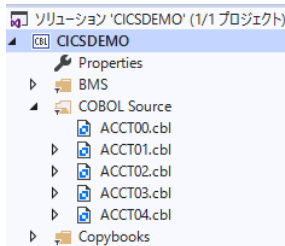
- 6) 用意した例題プログラム類をインポートします。CICSDEMO プロジェクトを右クリックして [追加] > [既存の項目] を選択し、既存項目の追加ウィンドウにて C:\¥Tutorials を指定すると内容が表示されます。ファイル名の先頭が ACCT の8ファイルを選択して [追加] ボタンをクリックします。この実行により、プロジェクトフォルダへ例題プログラムが配置されます。



- 7) 種類別に表示するため、[ソリューション エクスプローラー] 内の [仮想ビュー] アイコンをクリックします。



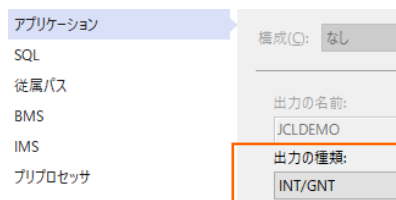
- 8) [ソリューション エクスプローラー] 内に表示されている CICSDEMO プロジェクトにインポートしたファイルが表示されていることを確認します。



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) [ソリューション エクスプローラー] 内の [Properties] をダブルクリックしてプロパティウィンドウを表示します。
- 2) 左側ツリービューの [アプリケーション] を選択して、生成する実行ファイルを GNT にするため [出力の種類] へ [INT/GNT] を選択します。

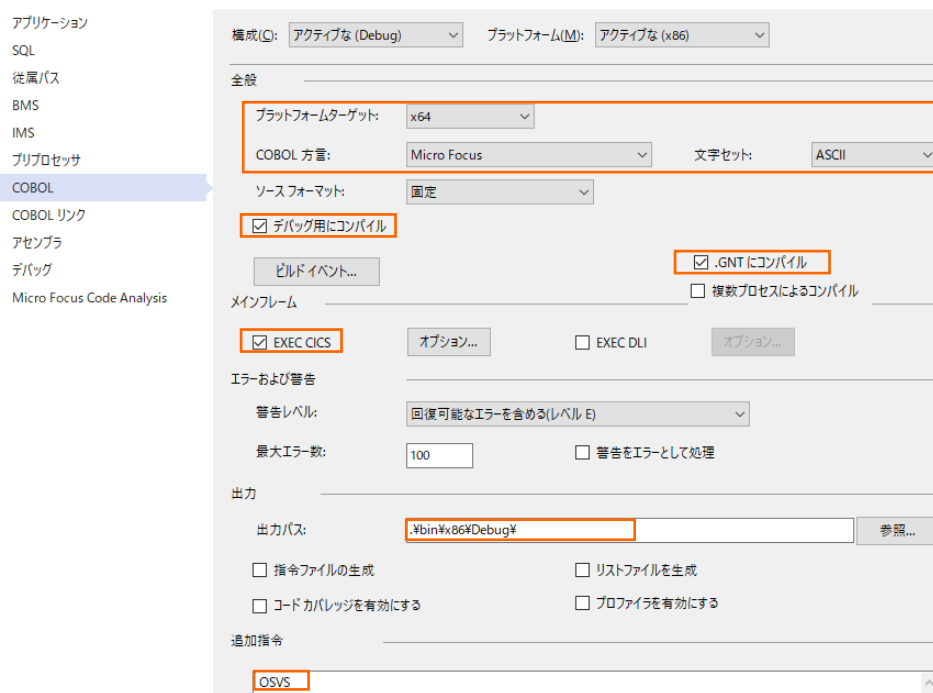


情報

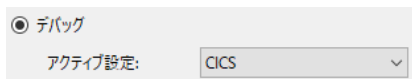
GNT は Micro Focus 独自のオブジェクトで、Micro Focus COBOL ランタイム環境下で実行可能となります。

3) 左側ツリービューの [COBOL] を選択して、下記項目を入力します。

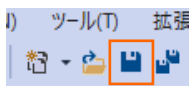
項目名	説明
プラットフォーム ターゲット	稼働ビット数を指定します。ここでは [x64] を指定します。
COBOL 方言	COBOL 言語方言を指定します。 例題プログラムは IBM OS/VS COBOL の方言を使用していますが、COPY 句に G 定数を使用しているためここでは [Micro Focus] を指定します。
文字セット	EBCDIC または ASCII を指定します。ここでは [ASCII] を選択します。
デバッグ用にコンパイル	デバッグ実行時に使用するファイルを生成するように指定します。
.GNT にコンパイル	実行ファイル形式を GNT に指定するためにチェックをオンにします。
メインフレーム EXEC CICS	プログラムに EXEC CICS 構文を含むため、チェックをオンにします。
出力パス	実行ファイルが出力されるパスを指します。任意に指定可能です。
追加指令	OSVS を指定します。



4) 左側ツリービューの [デバッグ] を選択して、[アクティブ設定] へ [CICS] を選択します。

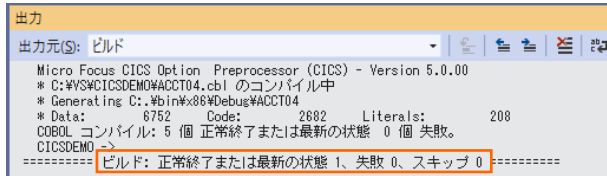


5) プロパティファイルを上書き保存します。

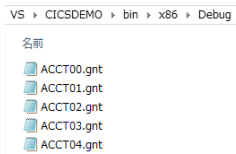


3.5 ビルドの実行

- 1) [ソリューション エクスプローラー] の CICSDemo ソリューションを右クリックして [ソリューションのビルド] を選択すると、コンパイル指定に沿ったビルドが実行されます。
- 2) [出力] ウィンドウで成功を確認します。



- 3) 前項で確認した出力パスへ、実行ファイル (.gnt) や マップファイル (.mod) が作成されていることを確認します。



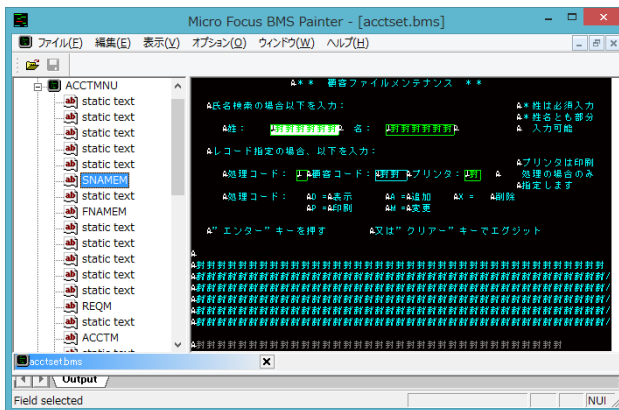
3.6 BMS 画面定義の確認

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が装備されています。

- 1) [ソリューション エクスプローラー] のプロジェクト内に存在する acctset.bms ファイルを右クリックして [BMS Painterを開く] を選択します。



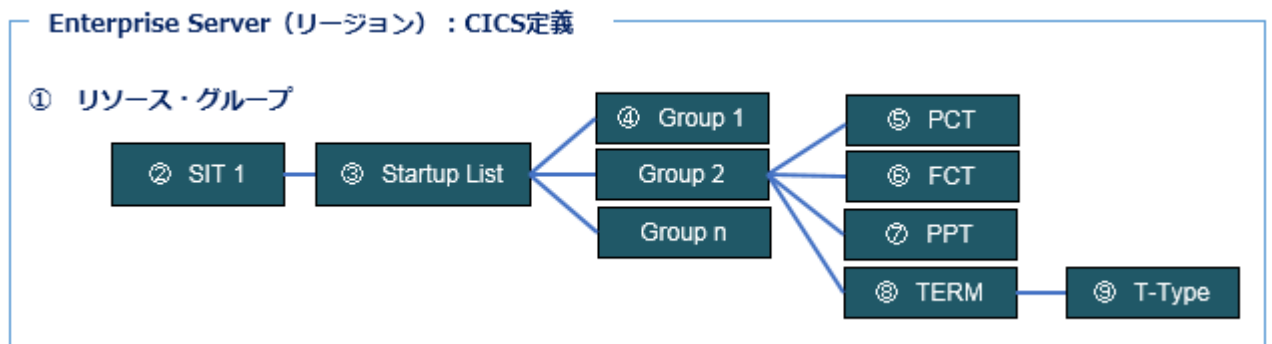
- 2) BMS ペインタウィンドウが表示され、画面定義内容をグラフィカルに確認できます。左側のツリービューでオブジェクトを選択すると右側のグラフィカルビュー内で対応するオブジェクトがハイライトされます。



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

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	CICS リソースのセットを指します。
② SIT	CICS の詳細設定が指定される、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。

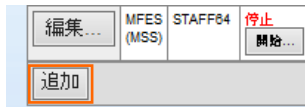
3.8 Enterprise Server インスタンスの設定

Enterprise Server インスタンスには CICS をエミュレーションする機能が搭載されており、この開発用インスタンスを使用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います。本番環境には実行製品である 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。[サーバー エクスプローラー] タブの [Micro Focus Server] を右クリックして [管理] を選択します。



- 2) Enterprise Server Administration 画面が表示され、Enterprise Server インスタンス一覧が表示されますので、画面の左下にある [追加] ボタンをクリックします。



- 3) サーバー名には CICSDEMO を入力、動作モードは 64-bit を指定して [次へ] ボタンをクリックします。

サーバー追加 (Page 1 of 3):

サーバー名:

動作モード:
 32-bit 64-bit
 You cannot change your choice of work

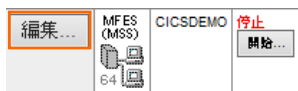
重要

実行ファイル生成に指定した稼働ビット数 = 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) 画面の Page 2/3 ではそのまま [次へ] ボタンを、Page 3/3 では [TN3270 リスナーの作成] チェックボックスがオンであることを確認して、[使用ポート] へ 9004 を指定後 [追加] ボタンをクリックすると、CICSDEMO という名前の 64 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。



- 5) 左にある [編集] ボタンをクリックします。



情報

指定したポート番号で TN3270 リスナーが作成されます。

6) [サーバー] > [プロパティ] > [一般] タブ内の下記項目を設定します。

- ① [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、Visual Studio からの動的デバッグが可能になります。

開始オプション:

共有メモリーブージ数:	512	サービス実行プロセス:	2
共有メモリックッション:	32	要求ライセンス:	10
ローカルコンソールを表示:	<input type="checkbox"/>	動的デバッグを許可:	<input checked="" type="checkbox"/>
Start on System Start:	<input type="checkbox"/>	64-Bit Working Mode:	<input checked="" type="checkbox"/>
以前のログを削除:	<input type="checkbox"/>	コンソールログサイズ (K):	0

- ② [構成情報] 欄に日本語半角カナを有効にするため下記内容を入力します。

[ES-Environment]

MFCODESET=9122

構成情報
[ES-Environment]
MFCODESET=9122



入力値は全て半角英数字で指定してください。

- ③ [適用] ボタンをクリックします。

7) [サーバー] > [プロパティ] > [MSS] > [CICS] タブで表示される画面の各項目を設定します。入力後は [Apply] ボタンをクリックします。

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。ここではオンへ指定します。
システム初期化テーブル (SIT)	CICS 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。
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出力パスを指定します。
File Path	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。
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。

メインフレーム サブシステム サポート有効:

CICS (✓) JES... IMS... PL/I

CICS 有効:

システム初期化テーブル (SIT):

トランザクションパス:

File Path:

マップパス:

リソース定義ファイルパス:



重要

入力値は全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。

- 8) 画面左上の [Home] をクリックして一覧画面に戻ります。

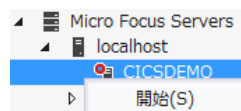


3.9 Enterprise Server インスタンスの開始と確認

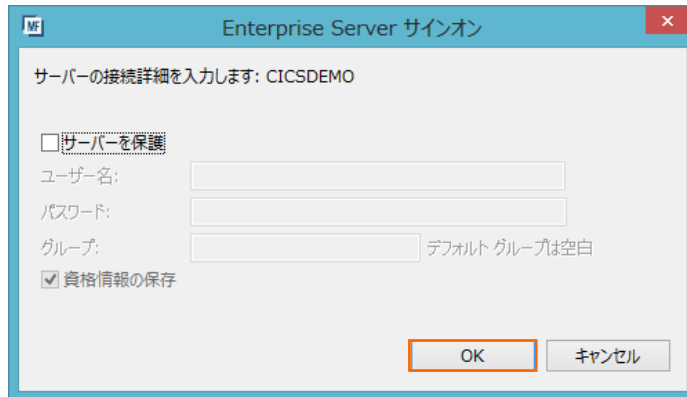
- 1) [サーバーエクスプローラー] 内に CICSDEMO インスタンスが表示されていることを確認します。表示されていない場合は [Micro Focus Server] を右クリックし、[最新の情報に更新] を選択してリフレッシュしてください。
- 2) [サーバーエクスプローラー] 内の CICSDEMO インスタンスを右クリックし、[プロジェクトと関連付ける] > [CICSDEMO] を選択します。これにより CICSDEMO プロジェクトから実行されるアプリケーションは CICSDEMO インスタンスで処理されることになります。



- 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。



Enterprise Server サインオン

サーバーの接続詳細を入力します: CICSDEMO

サーバーを保護

ユーザー名:

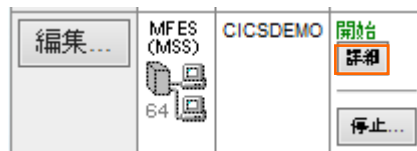
パスワード:

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

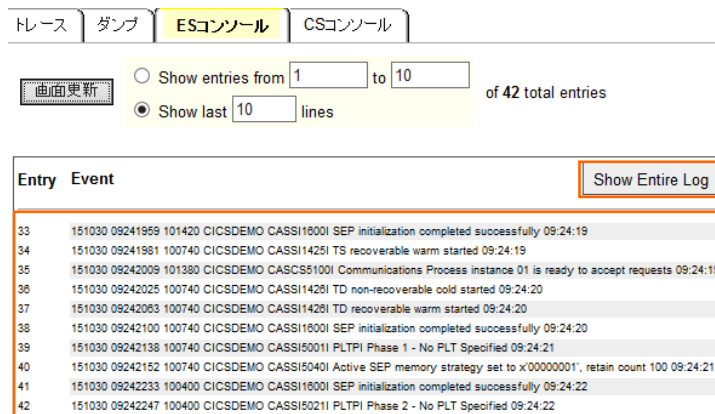
- 5) Enterprise Server Administration 画面へ移動して開始状態であることを確認後、[詳細] ボタンをクリックします。



編集... MFES (MSS) CICSDEMO 開始
64 [詳細]
停止...

- 6) [サーバー] > [診断] > [ES コンソール] で CICSDEMO 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。



ホーム 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 42 total entries

Show last 10 lines

Entry	Event
33	151030 09241959 101420 CICSDEMO CASSI1800I SEP initialization completed successfully 09:24:19
34	151030 09241981 100740 CICSDEMO CASSI1425I TS recoverable warm started 09:24:19
35	151030 09242009 101380 CICSDEMO CASC5100I Communications Process instance 01 is ready to accept requests 09:24:19
36	151030 09242025 100740 CICSDEMO CASSI1426I TD non-recoverable cold started 09:24:20
37	151030 09242083 100740 CICSDEMO CASSI1426I TD recoverable warm started 09:24:20
38	151030 09242100 100740 CICSDEMO CASSI1800I SEP initialization completed successfully 09:24:20
39	151030 09242138 100740 CICSDEMO CASSI5001I PLTPI Phase 1 - No PLT Specified 09:24:21
40	151030 09242152 100740 CICSDEMO CASSI5040I Active SEP memory strategy set to x'00000001', retain count 100 09:24:21
41	151030 09242233 100400 CICSDEMO CASSI1800I SEP initialization completed successfully 09:24:22
42	151030 09242247 100400 CICSDEMO CASSI5021I PLTPI Phase 2 - No PLT Specified 09:24:22

Show Entire Log

注意

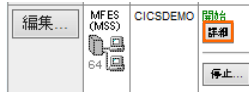
いくつかのサービス開始が失敗してもインスタンスは開始されますので、ログ内容を必ず確認してください。

- 7) 画面左上の [Home] をクリックして一覧画面に戻ります。

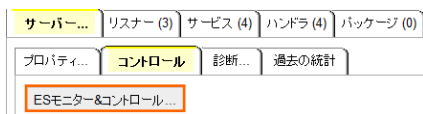
3.10 CICS リソース定義の確認

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ファイルを使用していますので、その内容を参照してみます。

- 1) Enterprise Server Administration 画面の CICSDEMO インスタンスの開始ステータス直下にある [詳細] ボタンをクリックします。



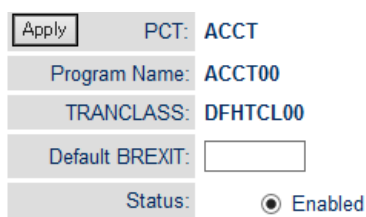
- 2) [ES モニター & コントロール] ボタンをクリックします。



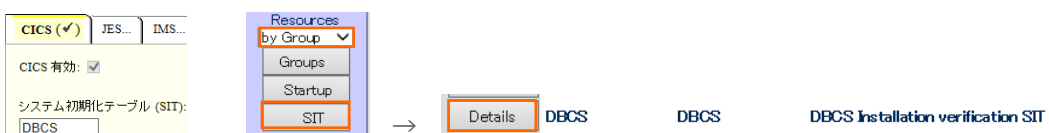
- 3) 画面左側中央にある [Resources] カテゴリ内のコンボボックスで [Active] を選択して、[PCT] ボタンをクリックすると現在アクティブな PCT 一覧が表示されますので、中央部の [ACCT] の虫眼鏡アイコンをクリックします。



- 4) 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 Enabled (有効) であることが確認できます。



- 5) 次に、画面左側中央にある [Resources] カテゴリ内のコンボボックスで [by Group] を選択し、[SIT] ボタンをクリックします。SIT 一覧が表示されますので、CICS タブの SIT で指定した DBCS の [Details] ボタンをクリックします。



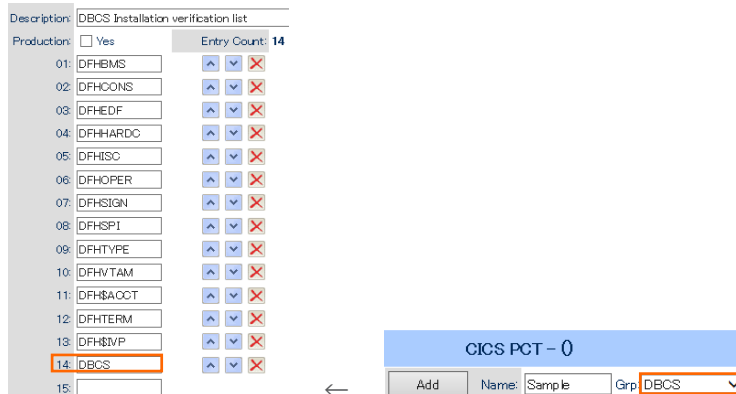
- 6) DBCS SIT には [Startup List] として同名の定義が、[Initial Tran ID] には CESN (CICS サインオン・トランザクション) が指定してあることが確認できます。



- 7) 上記で [Startup List] に指定してある DBCS の内容を確認します。画面左側中央にある [Resources] カテゴリ内の [Startup] ボタンをクリックすると Startup List 一覧が表示されますので指定した [DBCS] の [Details] ボタンをクリックします。



- 8) DBCS Startup List にはリソース定義類が属するグループが指定してあり、同じ名称の定義が異なるグループに存在する場合は後続グループが優位になります。この Startup List 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 リージョン構成チュートリアル] をご参照ください。



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。

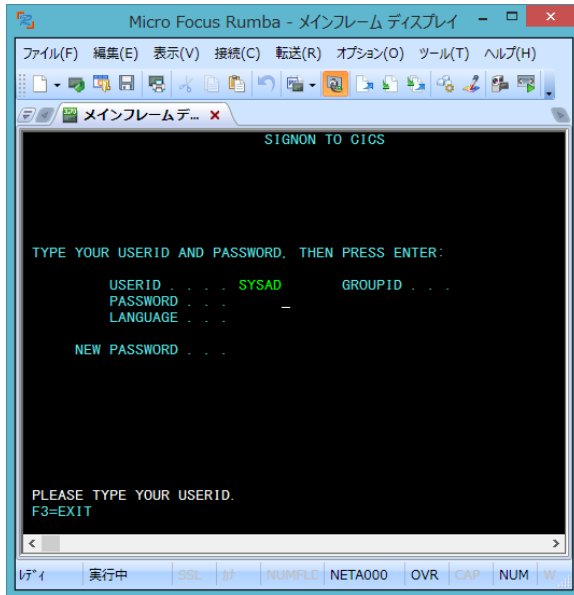
ご使用の TN3270 エミュレータを前項で作成した TN3270 リスナーポート (localhost:9004) へ接続します。

補足) 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。

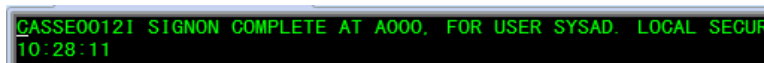
Rumba の例)



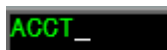
- 1) 下記は Micro Focus Rumba を使用した画面です。例題の SIT では初期トランザクションに CESN が指定されているため、接続後 CICS サインイン画面が表示されます。[USERID] と [PASSWORD] へ SYSAD を入力して実行キーを押します。



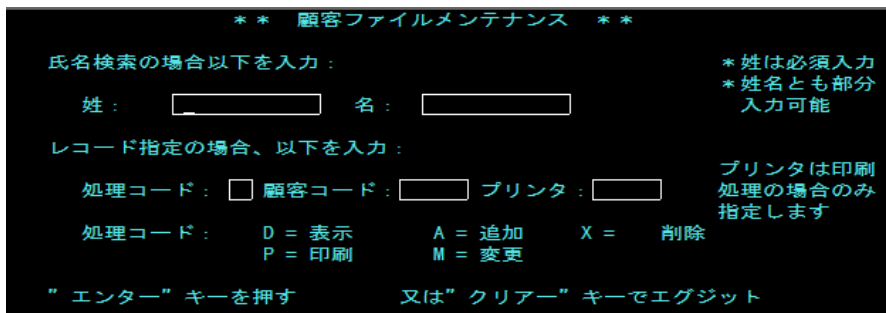
- 2) 正常にサインオンできましたら、クリアキーで画面をクリアします。



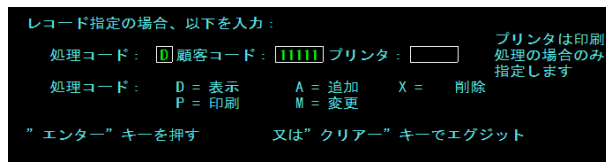
- 3) PCT に登録されていたトランザクションの ACCT を入力して実行キーを押します。



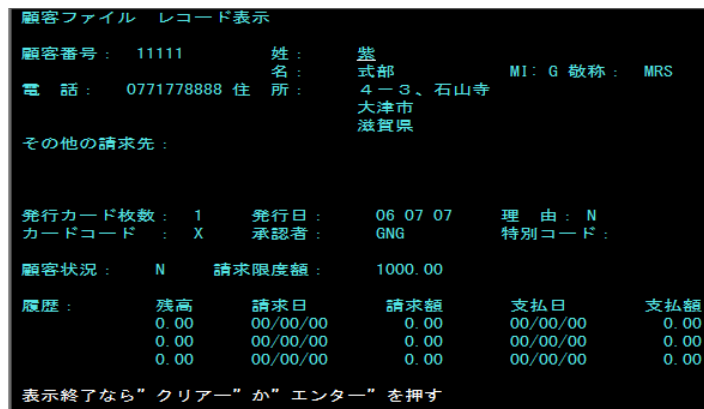
- 4) ACCT トランザクションからプログラムが呼ばれて、例題の BMS ファイルに定義されていた下記初期画面が表示されます。



- 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。



- 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。



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

3.12 CICS の動的デバッグ

Visual Studio 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] にチェックと CICSDEMO インスタンスと Visual Studio プロジェクトの関連付けが必要です。

- 1) [ソリューション 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。

```

ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
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。[デバッグ] プルダウンメニューの [ステップ イン] を選択します。

プロファイラー	
ステップイン(L)	F11
ステップオーバー(O)	F10

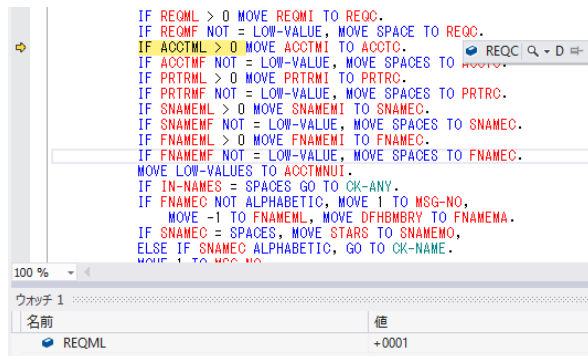
- 3) 画面左下がオレンジ色の [準備完了] となり、アタッチ待機状態になったことを確認します。



- 4) 前項と同様に TN3270 エミュレータから ACCT トランザクションを実行すると、プログラムのステップ実行が可能になります。F11 キーもしくは [デバッグ] 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ます。

マウスオーバーやウォッチタブを利用して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。



- 5) 希望のステップの左端を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。



- 6) 先に進める場合は画面上部の [続行] アイコンをクリックします。



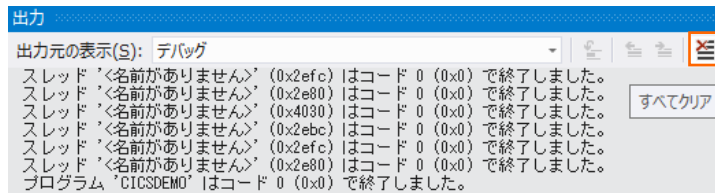
- 7) デバッグを終了させるため、画面上部の [デバッグ停止] アイコンをクリックします。



- 8) TN3270 エミュレータを切断します。
- 9) 画面左下が元の色の [準備完了] となり、アタッチ待機状態ではなくなったことを確認します。

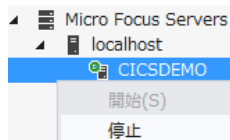


- 10) 出力タブに表示されているデバッグ履歴を消去するには、[全てクリア] アイコンをクリックします。

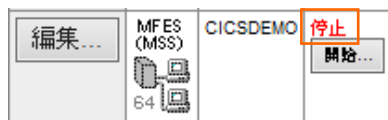


3.13 Enterprise Server インスタンスの停止

- 1) CICSDEMO インスタンスを停止します。



- 2) CICSDEMO インスタンスの停止を確認後、Visual Studio を終了します。



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。